Application de méthodes de classification supervisée et intégration de données hétérogènes pour des données transcriptomiques à haut-débit

Vincent Guillemot

ORSAY
N° d'ordre :

Université de Paris XI
Centre d'Orsay

École Supérieure d'Électricité
SUPÉLEC

THÈSE

présentée pour obtenir le grade de

DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ DE PARIX XI ORSAY

Par

Vincent GUILLEMOT

APPLICATION DE MÉTHODES DE CLASSIFICATION SUPERVISÉE ET INTÉGRATION DE DONNÉES HÉTÉROGÈNES POUR DES DONNÉES TRANSCRIPTOMIQUES À HAUT-DEBIT

Thèse soutenue le 29 mars 2010

Commission d'examen :

.

M.	**AMBROISE**	**Christophe**	**Rapporteur**
Mme	**BOULESTEIX**	**Anne-Laure**	**Rapporteur**
M.	**FLEURY**	**Gilles**	**Directeur de thèse**
M.	**FROUIN**	**Vincent**	**Encadrant**
M.	**ROBIN**	**Stéphane**	**Examinateur**
M.	**WALTER**	**Éric**	**Président du jury**

Thèse préparée au sein du
Département Signaux et Systèmes Électroniques, Supélec
et du
Laboratoire d'Exploration Fonctionnelle des Génomes, CEA

Table des matières

Remerciements

Je tiens à remercier en premier lieu Anne-Laure Boulesteix et Christophe Ambroise d'avoir accepté de rapporter cette thèse, Eric Walter d'avoir accepté de présider le jury et Stéphane Robin d'avoir accepté d'être examinateur.

J'adresse également mes plus sincères remerciements à Vincent Frouin pour m'avoir encadré dans des conditions parfois périlleuses, tout d'abord au sein du Laboratoire d'Exploration Fonctionnelle des Génomes au CEA à Evry, puis à distance, lorsque ce dernier laboratoire a disparu. Je le remercie tout particulièrement pour sa pugnacité, et pour m'avoir montré à maintes reprises ce que signifie l'honnêteté scientifique. J'espère que je me souviendrai encore très longtemps de cet enseignement. Je remercie également Sylvain Baulande qui s'est occupé de moi au sein de PartnerChip, je le remercie pour son écoute, sa présence et ses conseils. Je remercie Laurent Le Brusquet pour son encadrement depuis Supélec et pour avoir eu le courage de relever le défi de faire partie d'un encadrement « pluriel ». Enfin, Je remercie Gilles Fleury pour avoir accepté de diriger cette thèse au sein du Département Signaux et Systèmes Electroniques de Supélec.

Je me dois également de remercier Arthur Tenenhaus, dont j'ai fait la connaissance lors de son Post-Doc au CEA et qui m'a énormément apporté tout au long de la thèse, même et surtout lors des moments les plus difficiles. J'espère avoir hérité d'un peu de la créativité qu'il allie avec une remarquable adresse à sa rigueur scientifique. J'espère également avoir appris un peu en le regardant jongler avec ses différents sujets de préoccupation, tant humains que professionnels. Et enfin je n'ose espérer avoir été contaminé par son inimitable style.

Merci à tous les collègues de la défunte équipe bioinfo. Merci Cathy, pour m'avoir tant apporté dans les projets que nous avons eus en commun, dont je garde un excellent souvenir. Merci également à Olivier, qui m'a appris que l'on peut tout à fait garder son calme dans les situations les plus critiques. Merci à Cyril, qui m'a montré le contraire. Merci à Cédric, mon prédécesseur auprès de Vincent.

Je remercie chaleureusement Pascal Soularue pour d'une part avoir financé une partie de la thèse, mais aussi pour avoir su faire preuve de son sens pratique, de sa répartie et/ou de son humour dans les situations qui en avaient besoin. Je remercie aussi tous mes collègues de PartnerChip : Linhda, Amandine, Audrey, Gwenaëlle, Nadia pour les collaborations intéressantes et accessoirement pour les moments de détentes à base de cocktails audacieux.

Je remercie tous mes compagnons de route du LEFG : David, Marie-Anne, Amélie,

Peggy, Simon, Guillaume, Valérie ; ceux des ailes voisines : Ghida, Loubna, Emma, les écureuils farceurs, Johnny, le poulpe etc. J'ai une pensée particulière pour Claude, compagnon de covoiturage, parfois sans phares dans la nuit noire, parfois à reculons sur les entrées du périphérique, mais toujours dans la bonne humeur. Je le remercie aussi pour son crédo : « Tu dois pouvoir expliquer le sujet de tes recherches à ta grand-mère ! ».

Merci à Elisabeth, Karine, Luc, toujours serviables et aimables malgré mes questions administratives, ou non, farfelues.

Il est impensable que je ne remercie pas Floriane et Rémi, qui ont été contraints de me subir ainsi que mes excentricités tout au long de la thèse. Je les remercie pour leur patience, pour leur amitié et leur soutien.

Merci Nicolas de m'avoir montré qu'on peut finir sa thèse de Mathématiques en 2 ans tout en restant préoccupé par son avenir, pour les moments passés ensembles, et pour les conseils en algèbre. Merci Claire d'avoir pensé à moi pour analyser tes données, pour m'avoir montré également un spécimen assez exotique de statisticien à moustache, pour m'avoir permis d'être présent à la réunion pendant laquelle il a été déclaré que « nous ne sommes pas à l'abri d'une bonne idée », j'en garderai toujours un souvenir ému, pour ton écoute et tes bonnes bouteilles. Merci à toute la clique des anciens colocs pour les toujours joyeuses retrouvailles.

Bien sûr, je remercie toute ma famille pour m'avoir montré le chemin, et m'avoir si souvent relevé et aidé à panser mes bobos après mes nombreuses chutes. Je remercie plus particulièrement mes parents, pour leur soutien, et je remercie ma sœur pour les longues heures thérapeutiques passées à raconter des bêtises. Je remercie mon frère pour m'avoir enseigné un certain esprit de compétition qui m'a certainement été salutaire pendant ces trois années.

Il faudrait beaucoup plus que deux pages pour remercier tous ceux que je n'ai pas pu citer ici. Qu'ils sachent que je pense à tout ce qu'ils m'ont apporté et que je leur en suis infiniment reconnaissant.

Introduction

Les puces à ADN sont des outils permettant de visualiser à un instant donné l'activité transcriptomique à l'équilibre d'un échantillon de cellules, pour un organisme modèle. L'activité transcriptomique porte l'empreinte des interactions moléculaires essentielles à la vie de la cellule. Cela se modélise classiquement par un **réseau de régulations génétiques** (RRG).

Pendant la thèse, nous nous sommes intéressés à l'utilisation de puces à ADN dans trois types de problèmes :

(a) pour déterminer, entre deux situations biologiques, quels sont les gènes qui sont différemment exprimés dans une situation plutôt qu'une autre, on parle alors d'**analyse différentielle** ;

(b) pour apprendre les différences existant entre deux classes d'individus afin de pouvoir prédire par la suite la classe de nouveaux individus, on parle alors de **classification** ;

(c) enfin, pour déterminer un réseau de régulations génétiques à partir d'un échantillon homogène, on parle alors d'**inférence** de RRG.

Ces trois questions sont bien connues des statisticiens dans le cas où le nombre d'individus est inférieur au nombre de variables. Cependant, dans le cas des expériences transcriptomiques, le nombre d'individus (de l'ordre de la dizaine) est très inférieur au nombre de variables (de l'ordre de la dizaine de milliers). On doit ainsi mettre en œuvre des méthodes de sélection d'attributs et des méthodes de classification ou de régression régularisées.

De plus, ces problèmes ne sont pas indépendants. D'une part, des méthodes d'analyse différentielle sont utilisées pour réduire la dimension des jeux de données dans la classification de données transcriptomiques. On y trouve deux classes de méthodes : les approches *filter* basées sur des tests d'hypothèses et les approches *wrapper* utilisant des méthodes de classification. D'autre part, les graphes inférés sur des données transcriptomiques peuvent également être utilisés dans des méthodes de classification et dans des méthodes d'analyse différentielle.

L'objectif de la thèse est de proposer une nouvelle méthode permettant d'intégrer l'information contenue dans un RRG dans un processus de classification. Les points abordés sont représentés schématiquement sur la figure 1.

– Sur la droite de cette figure est représentée la méthodologie de classification que nous avons adoptée. La classification est effectuée dans un contexte de validation croisée, et la sélection de variables intervient uniquement sur les données d'apprentissage, comme cela est recommandé dans [Ambroise and McLachlan, 2002, Boulesteix and Strimmer, 2005].
– Sur la gauche est représenté le processus d'obtention du réseau de régulations génétiques à intégrer dans la classification. Les méthodes de classification existantes permettant d'intégrer un graphe utilisent toujours des graphes issus de bases de données publiques d'interactions entre gènes et produits de gènes. Cependant ces bases de données ne sont pas toujours adaptées à l'expérience biologique menée. Nous proposons donc d'inférer un ou plusieurs réseaux de régulations génétiques sur des données transcriptomiques indépendantes des données utilisées pour la classification.

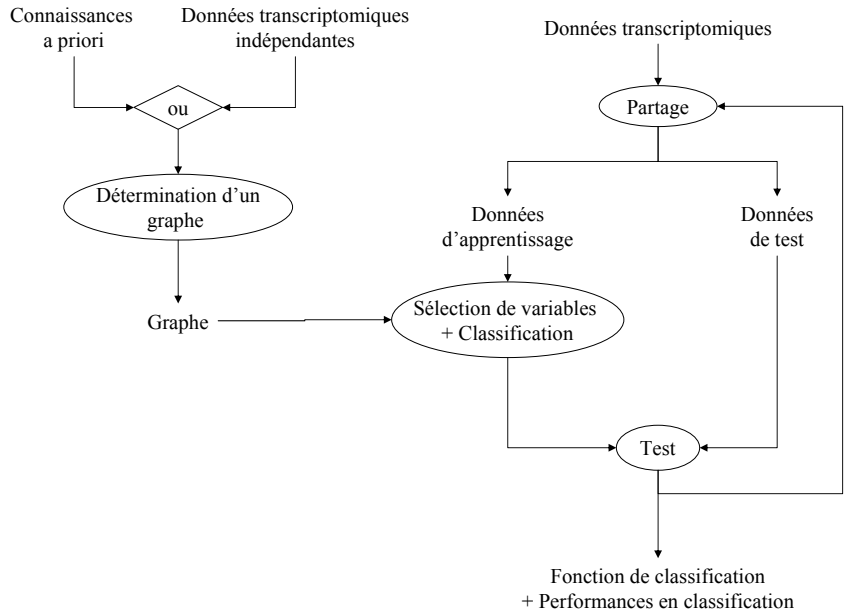


FIGURE 1 – Points abordés durant la thèse. L'objectif principal est de réaliser l'intégration d'un graphe dans un processus de classification.

La première partie de la thèse présente les méthodes de classification applicables aux expériences de transcriptomique haut-débit. En effet, du fait du nombre d'individus très faible par rapport au nombre de variables, les méthodes de classification échouent à proposer au clinicien ou au biologiste des fonctions de classification suffi-

samment performantes. Il faut donc non seulement mettre en œuvre des méthodes régularisées mais également des méthodes de réduction de dimension. Dans cette partie, nous mettons également en valeur l'importance, pour les méthodes de classification, de l'estimation des matrices de variance covariance entre les expressions des gènes à travers les individus des différentes classes.

La deuxième partie porte sur l'intégration à proprement parler d'un ou plusieurs graphes dans la classification. Les méthodes de l'état de l'art se concentrent sur la contrainte suivante : deux variables connectées dans le graphe à intégrer doivent avoir des poids identiques dans la classification. Cette contrainte est intégrée dans une méthode de classification comme les *Support Vector Machines* (SVM) [Rapaport et al., 2007, Zhu et al., 2009] ou dans une méthode de régression classique comme la régression LASSO [Li and Li, 2008]. Or, ces méthodes ne montrent pas d'amélioration notable des performances en classification. Nous proposons une méthode de classification basée sur l'analyse discriminante de Fisher, que nous avons appelée *graph constrained discriminant analysis* (gCDA), qui permet d'intégrer l'information contenue dans un RRG. L'intégration est effectuée lors des estimations des matrices de variance covariance effectuées dans l'algorithme de l'analyse discriminante. Nous montrons sur des données simulées que les performances en classification sont significativement améliorées.

La troisième partie traite la détermination d'un graphe à partir de données et l'adéquation d'un graphe à des données. Le premier point a été abordé lors d'un travail mené par Arthur Tenenhaus et a débouché sur l'implémentation d'une méthode d'inférence de réseaux. Le deuxième point apporte un complément d'une part aux méthodes d'inférence de réseau, car mesurer l'adéquation d'un graphe à des données permet de sélectionner parmi plusieurs graphes le « meilleur », et d'autre part à gCDA, car cela permet aussi de déterminer le modèle existant entre le graphe à intégrer et la matrice de variance covariance des données.

La quatrième partie présente l'application de gCDA à des données transcriptomiques réelles issues de bases de données publiques. Cela nécessite d'articuler tous les outils présentés précédemment, de la classification régularisée avec sélection d'attributs à l'inférence de réseau. Nous comparons gCDA aux méthodes de l'état de l'art permettant l'intégration d'un graphe et aux LP-SVM. Nous montrons une amélioration des performances de classification sur certains de ces jeux de données.

Obtenir un RRG et des données transcriptomiques nécessite une collaboration étroite avec les biologistes en charge de la partie expérimentale et des traitements statistiques sur les données brutes. Pour ne pas nuire au fil conducteur de ce mémoire, nous avons reporté en annexe tous les travaux effectués pendant la thèse qui ne sont pas directement en rapport avec l'intégration d'un graphe dans un processus de classification. Les problématiques et données biologiques que j'ai étudiées pendant la thèse proviennent du Laboratoire d'Exploration Fonctionnelle des Génomes (LEFG) du CEA et de la société PartnerChip (start-up du CEA). Ces deux organismes ont participé au financement de la thèse. Le LEFG a hébergé une plateforme d'hybridation de puces à ADN ; il proposait un service d'analyse de données acquises sur des modèles biologiques internes au LEFG ou provenant d'équipes de recherche extérieures. Part-

nerChip est une société de service d'hybridation de puces Affymetrix qui offre également une prestation d'analyse de données. J'ai donc pu participer à de nombreux projets, complémentaires à mon sujet de thèse, portant sur les points suivants :

- la normalisation de puces à ADN Affymetrix. Nous avons effectué une étude bibliographique de différentes méthodes de normalisation et les avons implémentées sous R. Des traitements statistiques de routine ont été rajoutés et le tout a été compilé dans une librairie R. Cette librairie est actuellement utilisée par les ingénieurs de PartnerChip pour la production des analyses. Une partie de mon travail a également consisté à mettre à jour cette librairie en fonction des besoins de PartnerChip.
- l'analyse de données toxicogénomiques. Ce point concerne le travail effectué pendant mon stage de Master, présenté à la conférence BIOTECHNO 2008 [Guillemot et al., 2008b].

D'autres collaborations ont été entreprises pendant la thèse et ne seront pas explicitées dans ce rapport. Elles portent sur :

- l'analyse de données transcriptomiques de patients en choc septique sévère [Lukaszewicz et al., 2007].
- la localisation de territoires chromosomiques [Heride et al., 2010].

Chapitre 1

État de l'art des méthodes de classification de données transcriptomiques

Les méthodes de classification de puces à ADN permettent d'exhiber les différences pouvant exister entre des classes d'individus au niveau transcriptomique. Ces différences sont résumées par une signature moléculaire constituée d'un ensemble de gènes discriminants. De plus, les méthodes de classification permettent l'estimation d'une fonction de classification et l'évaluation du pouvoir prédictif de la signature moléculaire. Classiquement, ce pouvoir prédictif est quantifié par le taux d'erreurs, qu'il est fondamental d'estimer sur un échantillon indépendant de celui qui a servi à l'apprentissage de la fonction de classification. La classification de puces à ADN est donc une première approche « haut débit » permettant de livrer à l'expérimentateur (le clinicien ou le biologiste) un sous-ensemble de gènes impliqués dans les processus biologiques étudiés. La signature moléculaire est ensuite validée sur des échantillons indépendants avec des techniques « bas débit » comme par exemple la *quantitative polymerase chain reaction* (PCR) ou des mesures de cytométrie. Le fait que le taux d'erreur doive être fiable et la signature moléculaire validable sur une cohorte indépendante amène à utiliser dans le processus de classification des méthodes de validation croisée qui seront présentées dans cette partie. Nous nous plaçons dans le cas de la classification de deux classes d'individus.

Les données de puces à ADN comportent classiquement de l'ordre de la dizaine de milliers de variables (p) pour au plus une centaine d'individus (n). Les algorithmes associés aux méthodes de classification standard comme l'analyse discriminante [Fisher, 1936] ou la régression logistique (voir [Tenenhaus, 2007] pour une présentation de ces méthodes standard) ne sont pas adaptés à ce cadre car nécessitant l'inversion de matrices mal conditionnées. Dans ce chapitre, nous présentons les méthodes de classification régularisées adaptées au cadre $p > n$. L'application de ces méthodes régularisées au contexte $n \ll p$ nécessite néanmoins de se ramener à $n < p$ en mettant en œuvre des stratégies de sélection d'attributs [Cornuéjols et al., 2002] pour garantir le carac-

tère généralisable de la fonction de classification sur des cohortes indépendantes, des temps de calcul raisonnables et des modèles plus interprétables par les biologistes.

Nous présentons les méthodes de classification et de régression de sorte à montrer que l'estimation régularisée des matrices de variance covariance est centrale dans le processus de classification dans le contexte $n < p$.

Notations

Une table des notations utilisées dans le rapport est disponible en D.4.

On considère p variables (les gènes) et n individus (ou mesures) répartis en deux classes. Notons $\boldsymbol{x}^{(k)}$ la matrice de dimensions $n_k \times p$, dont le coefficient $x_{ij}^{(k)}$ est l'expression du gène j pour l'individu i appartenant 0 la classe k. L'ensemble des expressions de tous les gènes de l'individu i est noté $x_{i.}^{(k)}$, et, de façon équivalente, l'ensemble des expressions du gène j pour tous les individus est noté $x_{.j}^{(k)}$.

L'ensemble des mesures de la classe $k = 1, 2$ est notée C_k et contient n_k individus :

$$C_k = \left\{ x_{i.}^{(k)}, i = 1, ..., n_k \right\}.$$

Cet ensemble est supposé être un échantillonnage de l'ensemble $\mathcal{C}_k$ contenant toutes les mesures possibles de la classe k.

Les lignes de $\boldsymbol{x}^{(k)}$ sont supposées être des réalisations d'une variable aléatoire réelle multivariée $\boldsymbol{X}^{(k)}$. La $j$ème composante de $\boldsymbol{X}^{(k)}$, notée $X_j^{(k)}$, est la variable aléatoire réelle modélisant le comportement de l'expression du gène j dans la classe k. De plus, les variables $\boldsymbol{X}^{(k)}$ sont supposées être des variables de densité $f_k : \mathbb{R}^p \mapsto \mathbb{R}$.

Les variables aléatoires que nous considérons admettent toujours un moment d'ordre 2 :

- pour une variable univariée X, la variance $\text{var}(X)$ est notée $\sigma^2 = E\left((X - E(X))^2\right)$,
- pour une variable multivariée $\boldsymbol{X}$, la matrice de variance covariance $\text{var}(\boldsymbol{X})$ est notée $\Sigma = E\left((\boldsymbol{X} - E(\boldsymbol{X}))(\boldsymbol{X} - E(\boldsymbol{X}))^\top\right)$.

Lorsque les variables considérées sont supposées gaussiennes, les notations adoptées sont les suivantes :

(a) du point de vue univarié, pour le gène j et la classe k, $X_j^{(k)} \sim \mathcal{N}(\mu_j^{(k)}, \sigma_j^{(k)\,2})$;

(b) du point de vue multivarié, pour la classe k, $\boldsymbol{X}^{(k)} \sim \mathcal{N}(\boldsymbol{\mu}^{(k)}, \Sigma^{(k)})$.

La moyenne d'une variable est estimée par la moyenne arithmétique :

$$\overline{\boldsymbol{x}}^{(k)} = \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{x}_{i.}^{(k)},$$

dont la $j$ème composante représente la moyenne du gène j :

$$\overline{x}_j^{(k)} = \frac{1}{n_k} \sum_{i=1}^{n_k} x_{ij}^{(k)}.$$

L'estimateur de la moyenne totale est

$$\overline{\boldsymbol{x}} = \frac{1}{n} \sum_{k=1}^{2} \sum_{i=1}^{n_k} \boldsymbol{x}_{i.}^{(k)}.$$

L'estimation usuelle non biaisée de la matrice de variance covariance de l'échantillon $\boldsymbol{x}^{(k)}$ est notée :

$$S^{(k)} = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} \left(\boldsymbol{x}_{i.}^{(k)} - \overline{\boldsymbol{x}}^{(k)} \right)^\top \left(\boldsymbol{x}_{i.}^{(k)} - \overline{\boldsymbol{x}}^{(k)} \right),$$

dont le $j$ème coefficient diagonal est noté $s_j^{(k)\,2}$ et représente la variance empirique du gène j :

$$s_j^{(k)\,2} = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} \left(x_{ij}^{(k)} - \overline{x}_j^{(k)} \right)^2.$$

L'estimation usuelle non biaisée de la matrice de variance covariance totale est notée :

$$S = \frac{1}{n} \sum_{k=1}^{2} \sum_{i=1}^{n_k} \left(\boldsymbol{x}_{i.}^{(k)} - \overline{\boldsymbol{x}} \right)^\top \left(\boldsymbol{x}_{i.}^{(k)} - \overline{\boldsymbol{x}} \right).$$

Enfin, la variable modélisant l'appartenance d'un échantillon à une classe donnée est notée Y. C'est une variable discrète à deux modalités $y_1 = -1$ et $y_2 = 1$. Ainsi la probabilité pour un individu d'appartenir à la classe k s'exprime comme étant la probabilité que la variable Y soit égale à y_k, elle est notée π_k :

$$\pi_k = P(Y = y_k).$$

1.1 Méthodes de classification et de régression régularisées

Nous nous intéressons aux méthodes de classification régularisées. Classer des individus demande de disposer d'une base d'apprentissage, dont les classes sont parfaitement connues, sur laquelle apprendre une fonction de classification qui permettra de prédire la classe de nouveaux individus. Nous notons la fonction de classification δ. δ est telle qu'elle permet d'attribuer la classe 1 ou 2 à un nouvel individu z :

$$\begin{aligned} \mathbb{R}^p &\to \{y_1, y_2\} \\ z &\mapsto \begin{cases} y_2 \text{ si } \delta(z) > 0 \\ y_1 \text{ sinon} \end{cases} \end{aligned}$$

Le cas qui nous intéresse dans ce paragraphe est le cas où la fonction δ est une fonction affine, et peut donc se mettre sous la forme

$$\delta : z \mapsto \beta^\top z + \beta_0, \tag{1.1}$$

avec $\beta \in \mathbb{R}^p$ et $\beta_0 \in \mathbb{R}$ les paramètres de la fonction de classification. Le but des méthodes de régression et de classification présentées ci-après est de déterminer les paramètres (β, β_0).

Nous entendons par méthode régularisée toute méthode de régression ou de classification qui permet de déterminer une fonction de classification tout en contrôlant sa norme. Dans une première partie, nous présentons l'exemple de la régression aux moindres carrés ordinaire (Ordinary Least Squares, noté OLS dans la suite) pour mesurer l'intérêt de la régularisation, suivie de méthodes de régressions régularisées classiques. Il est à noter que les méthodes de régression permettent de déterminer β en considérant la variable des classes comme une variable continue. La prédiction $\widehat{y} = \beta^\top z + \beta_0$ sera donc continue. La méthode choisie pour discrétiser la prédiction est présentée à la suite des méthodes de régressions régularisées. Dans une seconde partie, nous présentons les machines à vecteurs de support (SVM), méthode de classification régularisée couramment utilisées dans l'analyse de puces à ADN [Speed, 2003]. Enfin, les techniques de validation croisée que nous utilisons pour déterminer le paramètre de régularisation des méthodes présentées ainsi que pour estimer les performances en classification sont détaillées.

1.1.1 Méthodes de régression

Exemple de la régression OLS : besoin de régularisation. Soit $\boldsymbol{X} = (X_j, j = 1, ..., p)$ et Y des variables aléatoires ayant un moment d'ordre 2. Sans perte de généralité, nous supposons que les variables $\boldsymbol{X}$ et Y sont centrées. On cherche à prédire Y en fonction des variables présentes dans le vecteur $\boldsymbol{X}$.

Définition 1 (*Régression OLS*)

On appelle prédicteur OLS de Y en fonction de $\boldsymbol{X}$ la variable aléatoire $\gamma(\boldsymbol{X})$, combinaison linéaire des variables $X_j, j = 1, ..., p$, qui minimise le critère suivant

$$E\left((\gamma(\boldsymbol{X}) - Y)^2\right). \tag{1.2}$$

Il est noté $\widehat{Y}(\boldsymbol{X})$.

Théorème 1 (*Coefficients de régression*)

Les coefficients du prédicteur OLS de $\widehat{Y}(\boldsymbol{X})$ sont notés β et valent :

$$\beta = \mathrm{var}(\boldsymbol{X})^{-1} \mathrm{cov}(\boldsymbol{X}, Y). \tag{1.3}$$

Démonstration : La fonction γ minimisant le critère 1.2 est linéaire. $\gamma(\boldsymbol{X})$ sera donc noté

$\gamma^\top \boldsymbol{X}$ en considérant que $\gamma \in \mathbb{R}^p$. Ainsi,

$$\begin{aligned} f(\gamma) \triangleq E\left((\gamma(\boldsymbol{X}) - Y)^2\right) = E\left((\gamma^\top \boldsymbol{X} - Y)^2\right) &= E\left(\left(\gamma^\top \boldsymbol{X}\right)^2 - 2\gamma^\top \boldsymbol{X}Y + Y^2\right) \\ &= \gamma^\top E(\boldsymbol{X}^\top \boldsymbol{X})\gamma - 2\gamma^\top E(\boldsymbol{X}Y) + E(Y^2) \\ &= \gamma^\top \operatorname{var}(\boldsymbol{X})\gamma - 2\gamma^\top \operatorname{cov}(\boldsymbol{X}, Y) + E(X^2). \end{aligned}$$

La fonction f à minimiser étant quadratique (et convexe) en γ, il suffit d'annuler sa dérivée pour trouver son minimum :

$$\frac{\partial f}{\partial \gamma}(\gamma) = 2\operatorname{var}(\boldsymbol{X})\gamma - 2\operatorname{cov}(\boldsymbol{X}, Y) = 0.$$

Nécessairement, les coefficients de l'application linéaire minimisant l'erreur quadratique sont :

$$\beta = \arg\min_{\gamma \in \mathbb{R}^p} f(\gamma) = \operatorname{var}(\boldsymbol{X})^{-1} \operatorname{cov}(\boldsymbol{X}, Y).$$

■

L'estimateur classique pour β est l'estimateur

$$\widehat{\beta}^{ols} = \left(\boldsymbol{x}^\top \boldsymbol{x}\right)^{-1} \boldsymbol{x}^\top y, \tag{1.4}$$

avec $\boldsymbol{x}$ la matrice des mesures préalablement centrées.

L'équation (1.3) montre qu'estimer β peut se faire en calculant l'inverse de l'estimation de la matrice de variance covariance à partir des échantillons $\boldsymbol{x}$. Or , dans le cas où $n < p$ ou en présence de multicolinéarité entre variables, la matrice de variance covariance empirique pour des variables centrées $S = 1/(n-1)\boldsymbol{x}^\top \boldsymbol{x}$ n'est pas inversible, et utiliser l'estimateur $\widehat{\beta}^{ols}$ de l'équation (1.4) n'est plus possible. Il faut donc utiliser une pseudo-inverse ou introduire un système de contraintes. Ce second cas est étudié dans la suite de ce chapitre au travers des méthodes de régression LASSO [Tibshirani, 1996], de régression Ridge RR [Hoerl and Kennard, 1970] et de régression aux moindres carrés partiels (ou Partial Least Squares Regression notée PLS-R dans la suite) [Höskuldsson, 1988].

Le double intérêt de ces contraintes sera d'une part de résoudre le problème du mauvais conditionnement des données et d'autre part de diminuer l'erreur quadratique moyenne (Mean Square Error, MSE) de l'estimateur ainsi transformé. La MSE permet de mesurer la précision d'un estimateur et peut se décomposer en deux termes :

$$\begin{aligned} MSE(\widehat{\beta}) &= E\left[(\widehat{\beta} - \beta)^\top (\widehat{\beta} - \beta)\right] \\ &= \underbrace{\left(E[\widehat{\beta}] - \beta\right)^\top \left(E[\widehat{\beta}] - \beta\right)}_{\left\|\text{biais de } \widehat{\beta}\right\|^2} + \underbrace{E\left[(\widehat{\beta} - E[\widehat{\beta}])^\top (\widehat{\beta} - E[\widehat{\beta}])\right]}_{\text{terme de variance}}. \end{aligned} \tag{1.5}$$

Parmi tous les estimateurs sans biais, l'estimateur $\widehat{\beta}^{ols}$ de l'équation (1.4) est celui de variance minimale. Les estimateurs que nous allons détailler dans la suite sont volontairement de biais non nul mais de variance inférieure à celle de l'estimateur $\widehat{\beta}^{ols}$

de sorte à avoir une MSE inférieure. Nous montrons également, pour la RR et la régression LASSO, que cette estimation de β est essentiellement basée sur un estimateur différent de var$(\boldsymbol{X})$ qui a une MSE inférieure à celle de l'estimateur non biaisé classique S. Cette présentation est à rapprocher d'un travail récent de Witten et Tibshirani [Witten and Tibshirani, 2009].

Régression Ridge. Hoerl et Kennard en 1970 [Hoerl and Kennard, 1970] présentent la régression Ridge permettant de déterminer les coefficients de l'estimateur OLS dans le cadre $n < p$. Cette méthode est basée sur la résolution du problème d'optimisation suivant :

$$\min_{\beta} \|y - x\beta\|^2 \text{ t.q. } \|\beta\|^2 \leq t,$$

avec t un paramètre réel positif. La forme duale de ce problème d'optimisation est

$$\min_{\beta} \|y - x\beta\|^2 + \lambda \|\beta\|^2,$$

avec λ un nouveau paramètre réel positif.

On peut montrer que la solution de ce problème d'optimisation quadratique est

$$\widehat{\beta}^{ridge} = (x^\top x + \lambda I)^{-1} x^\top y.$$

Le fait d'introduire une contrainte sur la norme de β est donc équivalent à proposer un estimateur inversible

$$S + \frac{\lambda}{n-1} I \tag{1.6}$$

pour la matrice de variance covariance de $\boldsymbol{X}$.

Une alternative à la RR est la régression LASSO [Tibshirani, 1996] qui permet de sélectionner des variables d'intérêt tout en contraignant la norme de β.

Régression LASSO. La régression LASSO optimise le critère suivant :

$$J(\lambda) = \min_{\beta \in \mathbb{R}^p} \|y - x\beta\|^2 + \lambda |\beta|,$$

où $|\beta| = \sum_{j=1}^{p} |\beta_j|$. Minimiser ce critère ne peut se faire par des méthodes d'optimisation de critères quadratiques sous contraintes linéaires. L'algorithme le plus répandu pour calculer la solution de ce problème d'optimisation est basé sur la « régression aux moindres angles » (*Least Angle Regression*) [Efron et al., 2002]. La régression LASSO a comme propriété intéressante d'ajuster un modèle linéaire parcimonieux aux données.

La régularisation de l'estimation de la matrice de variance covariance apparaît plus clairement dans l'égalité vérifiée par $\widehat{\beta}^{lasso}$ [Tibshirani, 1996] :

$$\widehat{\beta}^{lasso} = \left(x^\top x + \lambda W^-\right)^{-1} x^\top y,$$

avec W la matrice diagonale contenant les coefficients $\left|\widehat{\beta}_i^{lasso}\right|$ et W^- l'inverse généralisée de W.

Régression Partial Least Squares. Dans cette partie, nous introduisons la régression PLS-R et nous concentrons sur les propriétés de régularisation de l'estimateur de la PLS-R pour β. Les mesures $\boldsymbol{x}$ représentent l'espace des prédicteurs alors que le vecteur y est la réponse à prédire. Nous considérons dans la suite que $\boldsymbol{x}$ et y sont centrées.

La PLS-R est utilisée pour résumer les données $\boldsymbol{x}$ sous la forme d'un ensemble de variables latentes $T = (t_1, ..., t_m)$ qui serviront à prédire y. Les variables latentes sont construites de sorte que leur covariance avec y soit maximale. Höskuldsson [Höskuldsson, 1988] montre que la première composante PLS-R $t_1 = \boldsymbol{x}w_1$ est obtenue en maximisant le critère de Tucker suivant [Tucker, 1958] :

$$w_1 = \underset{w/\|w\|=1}{\arg\max}\, \mathrm{cov}^2(\boldsymbol{x}w, y). \tag{1.7}$$

Les composantes suivantes t_h, $h = 2, ..., m$ sont également choisies de sorte à maximiser leur covariance avec y sous la contrainte supplémentaire que toutes les composantes soient orthogonales deux à deux. La contrainte d'orthogonalité est respectée par une procédure de « déflation » de $\boldsymbol{x}$, *i.e.* en éliminant dans les données l'information obtenue par projection de $\boldsymbol{x}$ sur l'espace engendré par $t_1, ..., t_{h-1}$: $\boldsymbol{x}_{h-1} = \boldsymbol{x} - \mathcal{P}^{\perp}_{t_1,...,t_{h-1}}\boldsymbol{x}$ avec $\mathcal{P}^{\perp}$ l'opérateur de projection orthogonale. La composante t_h est ensuite obtenue en résolvant le problème d'optimisation (1.8).

$$w_h = \underset{w/\|w\|=1}{\arg\max}\, \mathrm{cov}^2(\boldsymbol{x}_{h-1}w, y). \tag{1.8}$$

Enfin, le vecteur des poids de la régression $\hat{\beta}^{PLS}(h)$ défini ci-dessous (1.9) est déterminé

$$\hat{y}^{PLS}_h = \mathcal{P}^{\perp}_{t_1,...,t_h} y = \boldsymbol{x}\hat{\beta}^{PLS}(h). \tag{1.9}$$

Il servira à prédire le vecteur des classes d'un modèle à h composantes.

Proposition 1

Soit $s = \boldsymbol{x}^T y$, $C = \boldsymbol{x}^T\boldsymbol{x}$ et $K_h = \left[s\ Cs\ C^2s\ ...\ C^{h-1}s\right]$. Le vecteur $\hat{\beta}^{PLS}(h)$ peut s'exprimer ainsi :

$$\hat{\beta}^{PLS}(h) = K_h(K_h^{\top}CK_h)^{-1}K_h^{\top}s. \tag{1.10}$$

La preuve de cette proposition 1 est donnée par exemple dans [Manne, 1987] et [Helland, 1988]. Le corollaire suivant en est déduit :

Proposition 2

Le vecteur $\hat{\beta}^{PLS}(h)$ peut être calculé en résolvant le problème d'optimisation suivant (1.11) :

$$\hat{\beta}^{PLS}(h) = \underset{\beta\in\mathcal{K}_h(C,s)}{\mathrm{argmin}}\, \|y - \boldsymbol{x}\beta\|^2. \tag{1.11}$$

où $\mathcal{K}_h(C, s) = vect\{s,\ Cs,\ C^2s,\ ...,\ C^{h-1}s\}$

Ce corollaire, dont la preuve est détaillée dans [Tenenhaus et al., 2008], montre que PLS-R peut s'exprimer comme une version régularisée de la régression aux moindres carrés ordinaires, tout comme la régression LASSO ou la régression Ridge. Cette régularisation s'effectue de manière implicite et peut s'interpréter comme l'optimisation d'un critère des moindres carrés dont la solution est recherchée dans un espace de dimension plus petite que $\mathbb{R}^p$.

Obtenir une fonction de classification avec une méthode de régression. Pour les méthodes de régression, calculer β ne suffit pas lorsque le caractère à prédire Y est un caractère nominal. Il faut encore déterminer un seuil c à appliquer à la prédiction $\widehat{y} = \widehat{\beta}^\top z + \widehat{\beta}_0$ (z étant un nouvel individu dont la classe est à prédire) pour obtenir la fonction de classification suivante :

$$z \in \mathbb{R}^p \mapsto \begin{cases} 1 \text{ si } \delta(z) = \widehat{\beta}^\top z + \widehat{\beta}_0 > c \\ -1 \text{ sinon} \end{cases}$$

Pour positionner ce seuil, nous nous servons de courbes ROC (Receiver Operating Characteristic) (voir par exemple [Lasko et al., 2005]) : le seuil c doit optimiser à la fois la spécificité et la sensibilité de la fonction de classification. La spécificité spe et la sensibilité sen sont définies à partir du tableau (1.1) :

$$spe = \frac{VN}{VN + FP} \text{ et } sen = \frac{VP}{VP + FN}.$$

	$\boldsymbol{x}_i \in C_1$	$\boldsymbol{x}_i \in C_2$
$\beta x_i + \beta_0 > c$	VP	FP
$\beta x_i + \beta_0 \leq c$	FN	VN

TABLE 1.1 – Notations utilisées pour qualifier la prédiction de la classe de l'individu $\boldsymbol{x}_i$ en fonction d'un certain seuil β_0 à fixer. Les cases du tableau se lisent avec le code suivant : V = vrai, F = faux, P = positif et N = négatif. Par exemple, VP correspond donc au nombre de vrais positifs.

Il est classique d'utiliser le seuil optimal c_{opt} suivant :

$$c_{opt} = \arg\max_{c \in \mathbb{R}} \sqrt{(1 - sen(c))^2 + (1 - spe(c))^2},$$

ce qui permet d'obtenir une valeur de c pour laquelle à la fois la sensibilité et la spécificité sont proches de 1.

La détermination de ce seuil c n'est pas nécessaire lorsqu'une méthode de classification, comme les SVM, est utilisée.

1.1.2 Les Support Vector Machines : une méthode de classification régularisée de référence

Les SVM sont une méthode de classification introduites par [Cortes and Vapnik, 1995] basée sur la notion d'hyperplan séparateur optimal. On parle également de séparateurs à vastes marges, car cette méthode permet de maximiser la distance des observations à l'hyperplan séparateur (la marge) [Burges, 1998].

La fonction de classification recherchée est affine $\delta(\boldsymbol{x}) = \beta^\top \boldsymbol{x} + \beta_0$ et permet de décider qu'une mesure appartient à la classe k si $\delta(\boldsymbol{x})$ est du signe de y_k. La distance d'une mesure $\boldsymbol{x}_{i.}^{(k)}$ de la classe C_k à l'hyperplan (β_0, β) vaut :

$$\frac{y_k(\beta^\top \boldsymbol{x}_{i.}^{(k)} + \beta_0)}{\|\beta\|},$$

avec $\|\beta\|^2 = \sum_{j=1}^{p} \beta_j^2$. Choisir l'hyperplan de marge maximale revient à résoudre le problème suivant (dans le cas où les classes sont linéairement séparables)

$$\arg\max_{(\beta_0,\beta)} \frac{1}{\|\beta\|} \min_{k,i} y_k(\beta^\top \boldsymbol{x}_{i.}^{(k)} + \beta_0).$$

Notons les deux hyperplans caractérisant les marges de la façon suivante ($z \in \mathbb{R}^p$) :

$$\beta z + \beta_0 = 1 \text{ et}$$
$$\beta z + \beta_0 = -1.$$

On peut alors montrer que la distance entre ces deux hyperplans vaut l'inverse de $\frac{1}{2}\|\beta\|$. Minimiser $\frac{1}{2}\|\beta\| = \frac{1}{2}\sqrt{\sum \beta_i^2}$ ou $\frac{1}{2}\|\beta\|^2$ est théoriquement équivalent, mais le deuxième problème est plus facile à résoudre numériquement par des méthodes d'optimisation quadratique. Le facteur $\frac{1}{2}$ a été rajouté pour des commodités d'écriture. Ainsi, dans le cas où les classes sont linéairement séparables, le problème initial de maximisation des marges entres les mesures et l'hyperplan devient

$$\arg\min \frac{1}{2}\|\beta\|^2$$
$$\text{t.q. } \forall k = 1,2, \forall i = 1,...,n_k,\ y_k(\beta^\top \boldsymbol{x}_{i.}^{(k)} + \beta_0) \geq 1$$

Quand les classes ne sont pas séparables linéairement, il faut introduire des variables $\xi_i^{(k)} \geq 0$ qui permettent de relâcher les contraintes et donc de réaliser un compromis entre largeur de marge et taux d'erreur :

$$y_k(\beta^\top \boldsymbol{x}_{i.}^{(k)} + \beta_0) \geq 1 - \xi_i^{(k)}.$$

Le problème s'écrit alors

$$\arg\min \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{n} \xi_i^{(k)} \tag{1.12}$$
$$\text{t.q.} \quad \forall k = 1,2, \forall i = 1,...,n_k,\ y_k(\beta^\top \boldsymbol{x}_{i.}^{(k)} + \beta_0) \geq 1 - \xi_i^{(k)}$$
$$\forall k = 1,2, \forall i = 1,...,n_k,\ \xi_i^{(k)} \geq 0$$

Ainsi posé, c'est un problème d'optimisation d'un critère convexe sous contraintes linéaires, qui admet une unique solution [Faurre, 1988]. La résolution pratique de ce problème d'optimisation s'effectue usuellement sous sa forme duale (nous avons utilisé l'implémentation de l'algorithme LIB-SVM effectuée dans la librairie R `e1071` [Dimitriadou et al., 2006]).

Une variante des SVM [Bradley and Mangasarian, 1998] propose de résoudre le problème d'optimisation linéaire sous contraintes suivant :

$$\begin{aligned} & \arg\min |\beta| + C \sum_{i=1}^{n} \xi_i \\ \text{t.q.} \quad & \forall k = 1,2, \forall i = 1,...,n_k,\ y_i(\beta^\top x_i + \beta_0) \geq 1 - \xi_i^{(k)} \\ & \forall k = 1,2, \forall i = 1,...,n_k,\ \xi_i^{(k)} \geq 0 \end{aligned} \tag{1.13}$$

avec $|\beta| = \sum_{j=1}^{p} |\beta_j|$. Le problème (1.13) est plus adapté que les SVM classiques à des problèmes parcimonieux (en β). Nous noterons cette méthode LP-SVM (Linear Programming Support Vector Machines), en rapport avec le fait que des algorithmes de programmation linéaire sont mis en œuvre pour résoudre le problème d'optimisation 1.13. Nous avons utilisé la librairie R `lpSolve` pour implémenter les LP-SVM.

1.1.3 Validation croisée

Les algorithmes de classification présentés dépendent d'un paramètre de régularisation qui a une influence importante sur les performances du classifieur calculé. Si on note dans le cas général C ce paramètre, on aura donc, à la fin de l'algorithme, une estimation de l'hyperplan séparateur :

$$(\widehat{\beta}(C), \widehat{\beta}_0),$$

ce paramètre doit être déterminé en utilisant des techniques de validation croisée, de façon à éviter le « sur-apprentissage » de la fonction de classification sur les données de la base d'apprentissage. Ceci peut se faire en évaluant les performances de classification sur un échantillon n'ayant pas participé à la construction du modèle.

Pour estimer un taux d'erreur, nous avons choisi la validation croisée dite de Monte Carlo (Monte Carlo cross validation, MCCV), qui, à chaque itération de l'algorithme présenté sur la figure 1.2, choisit les échantillons d'apprentissage et de test par partition aléatoire de l'échantillon initial. Cette procédure est répétée $B = 100$ fois, ce qui est suffisant pour estimer en moyenne un taux d'erreur de classification et permet d'avoir une précision suffisante pour comparer deux méthodes de classification. Cette méthode permet de déterminer précisément le taux d'erreur, mais pour déterminer l'ensemble des paramètres de régularisation, nous préférons utiliser une validation croisée du type k-fold (avec $k = 10$), qui présente l'avantage de pouvoir estimer grossièrement le taux d'erreur bien plus rapidement que la MCCV. L'algorithme à la base de ces deux méthodes est le même, et est représenté figure (1.1). La différence se situe au niveau de l'étape de partage en jeux d'apprentissage et de test.

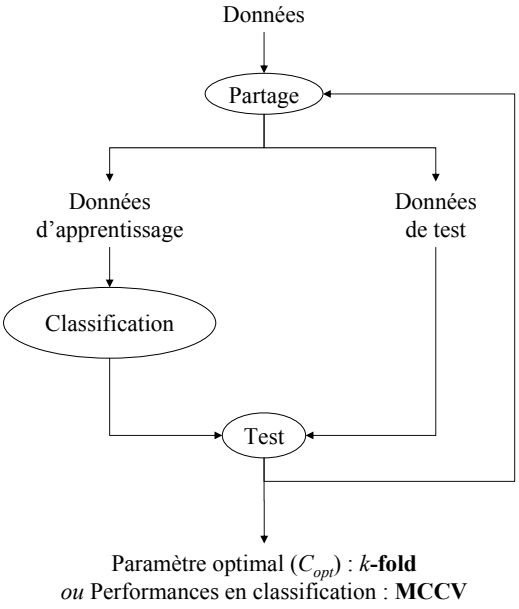


FIGURE 1.1 – Validation croisée pour déterminer un paramètre de l'algorithme de classification ou pour calculer un taux d'erreur.

L'algorithme utilisé pour mettre en œuvre la validation croisée à ces deux niveaux est représenté sur la figure 1.2 : la MCCV est utilisée pour estimer le taux d'erreur de classification (avec un nombre d'itérations $B = 100$) et les k-fold pour déterminer à chaque itération de la MCCV pour estimer l'ensemble des paramètres de la méthode de classification utilisée.

Dans un contexte de classification en très grande dimension, les méthodes de classification présentées ne suffisent plus à déterminer un classifieur performant. En effet, dans le cas des puces à ADN, le nombre de variables est de l'ordre de plusieurs dizaines de milliers (par exemple $p = 54675$ pour la puce hgu133plus2 d'Affymetrix), pour un nombre d'individus de l'ordre de la centaine, voire de la dizaine. Il faut donc accompagner la classification d'une réduction importante de la dimension du problème pour garantir un bon pouvoir de généralisation de la fonction de classification.

1.2 Gestion de la très grande dimension : de $n \ll p$ à $n < p$

Dans [Cornuéjols et al., 2002], les méthodes permettant de réduire la dimension du problème sont appelées choix des attributs de description des données. Ces nouveaux

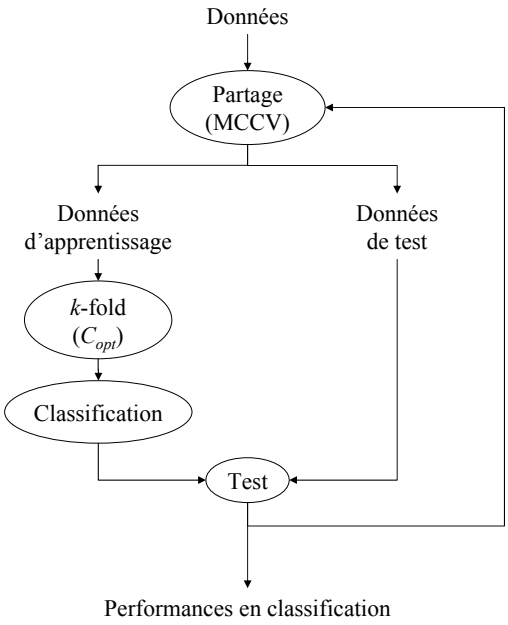


FIGURE 1.2 – Validation croisée pour déterminer un paramètre de l'algorithme de classification ou pour calculer un taux d'erreur.

attributs sont déterminés soit par *sélection*, soit par *extraction*.

La sélection d'attributs consiste, dans le cas de la classification binaire, à associer à chaque variable un score représentant son pouvoir de discrimination entre les deux classes. Pour ce faire, on utilise

(i) soit des méthodes *filter*, c'est-à-dire des méthodes statistiques basées sur des tests d'hypothèses univariés ou multivariés, le score associé à chaque variable est une *p-value* ;

(ii) soit des méthodes *wrapper*, c'est-à-dire des méthodes de classification, le score associé à chaque variable est son « importance » dans la fonction de décision calculée,

(iii) soit des méthodes dites *embedded* comme l'élimination récursive d'attributs (*Recursive Feature Elimination*) présentée par Guyon *et al.* [Guyon and Elisseeff, 2003].

L'extraction d'attributs consiste à appliquer une transformation linéaire ou non linéaire aux variables considérées de sorte à obtenir de nouvelles variables (le plus souvent appelées *composantes*) en moins grand nombre. Parmi ces méthodes d'extraction d'attributs on trouve :

(i) l'analyse en composantes principales ;

(ii) l'analyse en composantes indépendantes ;

(iii) la PLS-R ;

(iv) les cartes auto-organisatrices de Kohonen.

Dans le cadre de la thèse, nous nous sommes limités à la réduction de dimension par sélection d'attributs à l'aide de méthodes *filter* ou *wrapper*. Ces deux types de méthodes ont comme objectif d'identifier les gènes différentiellement exprimés, c'est-à-dire les gènes qui ont une moyenne significativement différente d'une classe à l'autre. Une comparaison de ces méthodes de sélection d'attributs est faite sur des donnée simulées. Les critères de comparaison sont la robustesse de la méthode par rapport aux paramètres en œuvre dans la simulation des données et l'identification des gènes différentiellement exprimés.

1.2.1 Méthodes *filter*

Les tests d'hypothèse utilisés en analyse différentielle et dans les logiciels d'analyse de profils d'expression, comme ArrayAssist (le logiciel d'Affymetrix) ou GeneSpring (le logiciel d'Agilent), sont des tests univariés classiques paramétriques, comme le test de Student, ou non paramétriques, comme le test de Mann-Whitney. Ces tests sont détaillés par exemple dans [Saporta, 2006] ou [Zar, 1999]. D'autres tests ont été développés spécifiquement pour les puces à ADN, utilisant le fait que les gènes ont des expressions qui ne sont pas indépendantes les unes des autres [Smyth, 2004], [Zuber and Strimmer, 2009]. Ces derniers tests sont parfois disponibles dans les logiciels d'analyse commerciaux, mais ils sont surtout disponibles librement sous la forme de librairies R.

Nous supposons être dans le cas de données non appariées, c'est à dire que les deux classes d'individus sont indépendantes ; très pratiquement, cela signifie que les patients recrutés dans chaque classe sont « différents » d'une classe à l'autre.

Ratio d'expression. La mesure communément admise chez les biologistes de cette différence est le ratio d'expression, également appelé *Fold-Change* (et donc abrégé en FC par la suite). Pour des données en échelle logarithmique :

$$FC = \overline{x}_1 - \overline{x}_2,$$

ou même parfois

$$FC = \text{signe}(\overline{x}_1 - \overline{x}_2) 2^{|\overline{x}_1 - \overline{x}_2|},$$

dépendant de la normalisation appliquée au jeu de données.

Statistique basée sur les moyennes géométriques. Une première méthode de filtrage est présentée dans [Breitling et al., 2004] et est très similaire au FC. La procédure décrite dans cet article est assez intuitive : elle consiste à calculer des ratios d'expression (RE) pour tous les individus. Ces RE servent ensuite à calculer des rangs qui sont ensuite moyennés pour obtenir un rang global, appelé dans l'article Rank Product

(RP). Ainsi plus ce dernier est petit, plus le gène en question est différentiellement exprimé.

On note i_k l'individu courant de la classe k : pour chaque individu de la classe C_1, on calcule un RE avec chaque individu de la classe C_2. Alors pour le gène j :

$$\overline{rp}_j = \prod_{i_1=1}^{n_1} \prod_{i_2=1}^{n_2} r_j(i_1, i_2),$$

avec $r_j(i_1, i_2) = x_{i_1 j}^{(1)} / x_{i_2 j}^{(2)}$. La significativité de ces $\overline{rp}_j$ est calculée par une procédure de permutations.

Test de Mann-Whitney. Le test de Mann-Whitney (ou test de Wilcoxon-Mann-Whitney) teste si deux échantillons *i.i.d.* de mesures effectuées sur un même gène j $x_{ij}^{(1)}, i = 1, ..., n_1$ et $x_{ij}^{(2)}, i = 1, ..., n_2$ ont des médianes significativement égales (voir [Mann and Whitney, 1947] ou plus récemment [Zar, 1999]). Les hypothèses nulles et alternatives sont respectivement :

$$\mathcal{H}_0 : \text{« médianes identiques »}, \mathcal{H}_A : \text{« médianes différentes »}.$$

C'est un test non paramétrique qui ne nécessite donc aucune hypothèse sur la loi suivie par les deux échantillons. Ce test est un test bilatéral au niveau de confiance α.

On calcule la statistique suivante

$$u_1 = n_1 n_2 + \frac{n_1(n_1+1)}{2} - r_1,$$

avec r_1 la somme des rangs attribués aux mesures de la classe C_1 dans le classement par valeurs croissantes des deux échantillons fusionnés.

Les p-valeurs associées à la statistique u_1 sont soit tabulées, soit calculables par l'approximation suivante sous l'hypothèse nulle :

$$\left(\frac{12}{n_1 n_2 (n_1 + n_2 + 1)}\right)^{-1/2} \left(U_{n_1,n_2} - \frac{n_1 n_2}{12}\right) \to \mathcal{N}(0, 1).$$

Lorsque l'on ne peut formuler d'hypothèses sur la distribution des variables aléatoires sous-jacentes , il est d'usage d'utiliser un test non paramétrique comme le test de Mann-Whitney que nous venons de voir. Cependant, lorsque les données sont gaussiennes, on perd en puissance à utiliser ce test non paramétrique plutôt que son équivalent paramétrique, le test de Student (*t-test* en anglais).

Test de Student Contrairement au test de Mann-Whitney, le test de Student fait l'hypothèse que les deux échantillons de mesures sur le gène j $\boldsymbol{x}_{ij}^{(1)}, i = 1, ..., n_1$ et $\boldsymbol{x}_{ij}^{(2)}, i = 1, ..., n_2$ suivent respectivement une loi gaussienne $\mathcal{N}(\mu_j^{(1)}, \sigma_j^{(1)\,2})$ et $\mathcal{N}(\mu_j^{(2)}, \sigma_j^{(2)\,2})$. Les hypothèses nulle et alternative sont les suivantes :

$$\mathcal{H}_0 : \mu_j^{(1)} = \mu_j^{(2)}, \mathcal{H}_A : \mu_j^{(1)} \neq \mu_j^{(2)}.$$

Supposons que, comme c'est le cas pour l'analyse de puces à ADN, $\sigma_j^{(1)\,2}$ et $\sigma_j^{(2)\,2}$ sont inconnues. On distingue alors deux cas :

1. $\sigma_j^{(1)\,2} = \sigma_j^{(2)\,2}$, la statistique utilisée sera :

$$T_1 = \frac{\overline{\boldsymbol{x}}_j^{(1)} - \overline{\boldsymbol{x}}_j^{(2)}}{\sqrt{\frac{(n_1-1)s_j^{(1)\,2}+(n_2-1)s_j^{(2)\,2}}{n_1+n_2-2}\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}} \sim \mathcal{T}(n_1+n_2-2)$$

2. $\sigma_j^{(1)\,2} \neq \sigma_j^{(2)\,2}$, la statistique utilisée sera :

$$T_2 = \frac{\overline{\boldsymbol{x}}_j^{(1)} - \overline{\boldsymbol{x}}_j^{(2)}}{\sqrt{\frac{s_j^{(1)\,2}}{n_1}+\frac{s_j^{(2)\,2}}{n_2}}} \sim \mathcal{T}(m),$$

   où le nombre de degrés de liberté m est l'entier se rapprochant le plus de la quantité :

$$m \approx \frac{c^2}{n_1-1} + \frac{1-c^2}{n_2-1}, \text{ avec } c = \frac{s_j^{(1)\,2}/n_1}{s_j^{(1)\,2}/n_1 + s_j^{(2)\,2}/n_2}.$$

La deuxième version de ce test est également appelée ***test d'Aspin-Welch***.

Le test de Student est robuste à la violation des hypothèses de normalité des échantillons et au fait que les échantillons soient petits [Zar, 1999]. C'est un test très utilisé pour identifier des gènes différentiellement exprimés.

Test de Student modéré Le test de Student modéré décrit dans [Smyth, 2004] est basé sur l'hypothèse que l'estimation de la variance de chaque gène doit prendre en compte l'ensemble des valeurs d'expression. Cela permet d'éviter le fait que des sondes d'expression parfois très faibles se retrouvent considérées comme différentiellement exprimées.

Pour chaque gène $j = 1, ..., p$, on a

$$\mathcal{H}_0 : \mu_j^{(1)} = \mu_j^{(2)} \text{ et } \mathcal{H}_A : \mu_j^{(1)} \neq \mu_j^{(2)},$$

La statistique modérée est notée $\widetilde{T}$,

$$\widetilde{T} = \frac{\overline{\boldsymbol{x}}_j^{(1)} - \overline{\boldsymbol{x}}_j^{(2)}}{\sqrt{\widetilde{s}_j^2\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}},$$

avec

$$\widetilde{s}_j^2 = \frac{d s_j^2 + d_0 s_0^2}{d + d_0}.$$

Les paramètres s_0 et d_0 sont estimés à partir de l'ensemble des mesures, d est le nombre de degrés de libertés de la statistique non modérée et s_j^2 la variance poolée.

Cette statistique est encore une statistique de Student, mais elle n'est plus radicalement univariée, elle prend en compte le niveau d'expression des autres gènes.

Test du T^2 de Hotelling Les tests univariés font l'hypothèse que les gènes sont indépendants. Or cette hypothèse n'est pas en accord avec l'existence de réseaux de régulations génétiques. Les statistiques multivariées de Hotelling ([Lu et al., 2005] ,[Guillot et al., 2007]) permettent de tenir compte des relations existants entre gènes. Pour réaliser ce test, on se place dans le cadre multivarié : soit deux variables multivariées gaussiennes $\boldsymbol{X}^{(1)} \sim \mathcal{N}(\boldsymbol{\mu}^{(1)}, \Sigma^{(1)})$ et $\boldsymbol{X}^{(2)} \sim \mathcal{N}(\boldsymbol{\mu}^{(2)}, \Sigma^{(2)})$. Les hypothèses nulle et alternative sont

$$\mathcal{H}_0 : \boldsymbol{\mu}^{(1)} = \boldsymbol{\mu}^{(2)} \text{ et } \mathcal{H}_A : \boldsymbol{\mu}^{(1)} \neq \boldsymbol{\mu}^{(2)},$$

La statistique utilisée est

$$T^2 = \left(\overline{\boldsymbol{x}}^{(1)} - \overline{\boldsymbol{x}}^{(2)}\right)^\top W^{-1} \left(\overline{\boldsymbol{x}}^{(1)} - \overline{\boldsymbol{x}}^{(2)}\right),$$

avec W un estimateur de la matrice de variance covariance poolée. On peut montrer que cette statistique suit une loi de Fisher.

Cette statistique ne peut être calculée que sur un sous-ensemble très restreint de gènes, car il est nécessaire pour la calculer d'inverser une matrice de variance covariance empirique (ce qui n'est pas possible si le nombre de gènes considérés est trop important). Le but de la procédure est finalement de trouver le plus grand sous ensemble de gènes pour lequel le test du T^2 de Hotelling dit que $\boldsymbol{\mu}^{(1)} \neq \boldsymbol{\mu}^{(2)}$ de façon significative.

La procédure décrite dans [Lu et al., 2005] est coûteuse en temps de calcul car elle nécessite de parcourir l'ensemble des groupes de gènes possibles. Pour utiliser la procédure de [Guillot et al., 2007], l'utilisateur doit lui-même fixer la taille des sous-ensembles de gènes à considérer ainsi que le nombre de ces sous ensembles. Pour ces raisons, bien que ces tests soient prometteurs, ils sont en pratique difficilement utilisables.

Classement des variables par « cat-score ». Calculer une statistique de Student pour chacune des variables revient en fait à calculer un vecteur $\boldsymbol{t}$ de taille $p \times 1$ tel que

$$\boldsymbol{t} = \left[\left(\frac{1}{n_1} + \frac{1}{n_2}\right)\widehat{V}\right]^{-1/2} (\overline{\boldsymbol{x}}^{(1)} - \overline{\boldsymbol{x}}^{(2)}),$$

avec $\widehat{V}$ une matrice diagonale contenant les estimations des variances poolées de chaque gène. Zuber et Strimmer [Zuber and Strimmer, 2009] proposent de considérer que $\widehat{V}$ correspond au cas où les variables sont décorrélées. Pour prendre en compte la

corrélation existant entre les variables, matérialisée par la matrice de corrélation P, ils proposent la statistique :

$$\boldsymbol{t} = \widehat{P}^{-1/2} \left[\left(\frac{1}{n_1} + \frac{1}{n_2} \right) \widehat{V} \right]^{-1/2} (\overline{\boldsymbol{x}}^{(1)} - \overline{\boldsymbol{x}}^{(2)}),$$

avec $\widehat{P}$ l'estimée de la matrice de corrélation.

Dans le cas $n < p$, l'estimateur classique de la matrice de corrélation n'est plus défini positif. L'estimateur $\widehat{P}$ utilisé dans [Zuber and Strimmer, 2009] est celui présenté par Schäfer *et al.* dans [Schäfer and Strimmer, 2005b].

Rappelons que le but des tests présentés est de sélectionner un nombre de variables très inférieur au nombre initial de variables. À l'issue des tests proposés, un ordonnancement des gènes est proposé. Nous n'appliquons pas de correction des tests multiples car cet ordre n'est pas impacté par ce type de correction.

1.2.2 Méthodes *wrapper*

Les méthodes de classification utilisées comme méthodes de sélection d'attributs sont appelées méthodes *wrapper*. On se sert de l'importance qu'a une variable dans la fonction de classification pour déterminer si le gène correspondant est significativement différentiellement exprimé : plus $|\beta_j|$ est grand, plus son rôle est important dans la fonction de classification et plus le gène j est considéré comme différentiellement exprimé.

Le tableau 1.2 résume les différents problèmes d'optimisation associés à chacune des méthodes précédemment décrites.

Méthode de classification	Expression du problème d'optimisation
SVM	$\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \beta^\top \boldsymbol{x}_i) + \lambda \|\beta\|^2$
LP-SVM	$\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \beta^\top \boldsymbol{x}_i) + \lambda \|\beta\|$
RR	$\min_{\beta \in \mathbb{R}^p} \|y - \beta^\top \boldsymbol{x}\|^2 + \lambda \|\beta\|^2$
PLS-R	$\min_{\beta \in K_h} \|y - \beta^\top \boldsymbol{x}\|^2$
LASSO	$\min_{\beta \in \mathbb{R}^p} \|y - \beta^\top \boldsymbol{x}\|^2 + \lambda \|\beta\|$

TABLE 1.2 – Expression des problèmes d'optimisation pour chaque méthode de classification.

1.2.3 Comparaison des différentes méthodes d'analyse différentielle sur des données simulées

La capacité des méthodes présentées à identifier correctement des gènes différentiellement exprimés sera évaluée à l'aide de courbes ROC.

Pour simuler des données, nous avons utilisé deux méthodes. Elles consistent à tirer des réalisations indépendantes d'une variable aléatoire gaussienne $\mathcal{N}(\mu^{(k)}, \Sigma^{(k)}), k = 1, 2$, avec

1. $\mu^{(1)} = 0_{\mathbb{R}^p}$ et $\mu^{(2)}$ qui varie selon la méthode de simulation utilisée :
   (a) différences discrètes, pour chaque gène j, la différence entre les deux moyennes suit une loi de Bernoulli de paramètre noté $\varpi \in]0;1[$, $\boldsymbol{\mu}_j^{(2)} - \boldsymbol{\mu}_j^{(1)} \sim \mathcal{B}(\varpi)$,
   (b) différences gaussiennes, $\boldsymbol{\mu}^{(2)} - \boldsymbol{\mu}^{(1)} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$;
2. les $\Sigma^{(k)}$ qui sont des matrices de variance-covariance inspirées directement de données réelles (les données Golub) : ce sont des sous-matrices de taille $p \times p$ des matrices de variance covariance empiriques de chaque classe des données. La méthode utilisée pour réaliser cette inférence est la méthode régularisée de [Schäfer and Strimmer, 2005b].

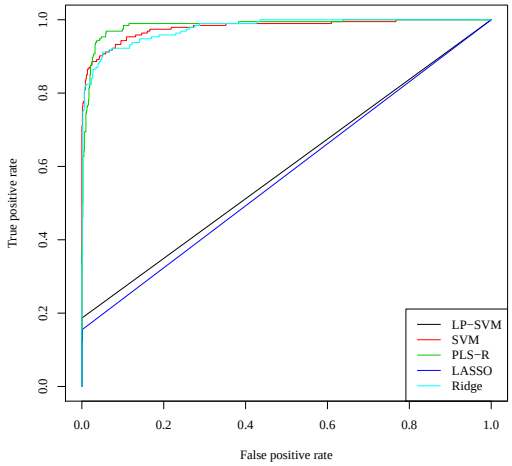
La différence entre ces deux types de simulation se situe au niveau de $\mu^{(2)}$. Les deux modèles de différences sont tels que le nombre de gènes différentiellement exprimés soit de 5%.

Nous avons fixé $n = 50$ et $p = 1000$. Pour le modèle de différences discrètes, les résultats sont matérialisées par les courbes ROC des figures 1.3(a) et 1.3(b). Pour le modèle de différences gaussiennes, les courbes ROC sont représentées sur les figures 1.3(c) et 1.3(d). D'autres simulations ont été effectuées en faisant varier les valeurs des paramètres p et n (cf. annexe A).

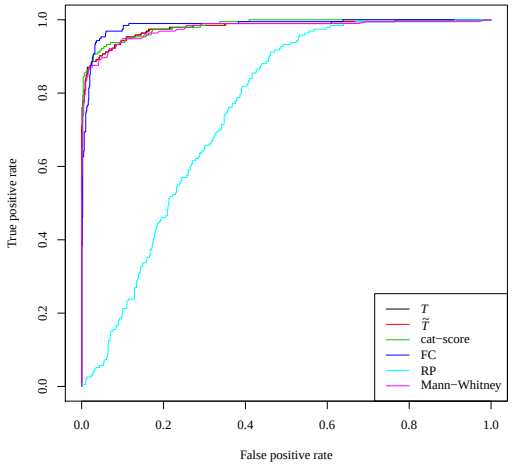
Ces figures nous montrent que le comportement de toutes ces méthodes dépend beaucoup de la différence existant réellement entre les deux classes d'individus. Il est difficile de dire quel est le modèle qui a le plus de pertinence biologique, mais il est certain qu'il faut choisir la méthode la plus robuste par rapport aux différentes situations que nous avons choisies de modéliser. Les méthodes *wrapper* ne sont jamais meilleures que les méthodes *filter* et elles sont de plus gourmandes en temps de calcul à cause de la nécessité de faire de la validation croisée, elles ne sont donc pas retenues. Les méthodes de filtrage par FC ou par statistique de Student sont les plus intéressantes. La première car elle est très facile et rapide à calculer et parce qu'elle est robuste « biologiquement » et la deuxième pour sa robustesse. Nous retenons donc en premier lieu le filtrage par FC, accompagné, si besoin d'un filtrage par test de Student.

1.3 Démarche complète de classification de données transcriptomiques

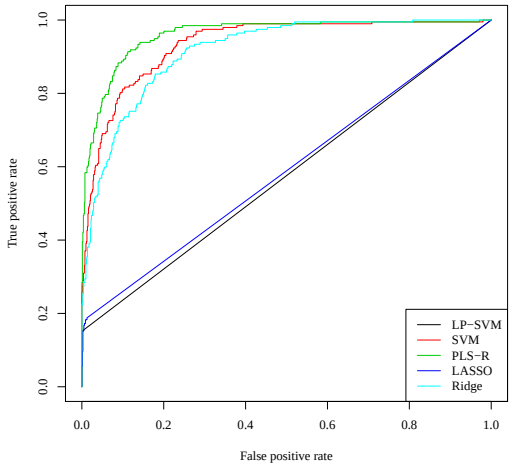
Dans cette section, nous présentons la méthodologie utilisée par la suite pour la classification des données microarray selon les recommandations formulées dans



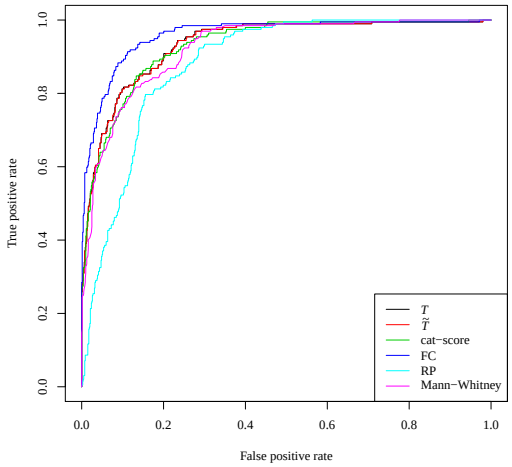
(a) Méthodes *wrapper*, modèle discret



(b) Méthodes *filter*, modèle discret



(c) Méthodes *wrapper*, modèle gaussien



(d) Méthodes *filter*, modèle gaussien

FIGURE 1.3 – Comparaison des performances des méthodes de sélection d'attributs sur deux modèles de simulations : en haut les différences entre les moyennes des deux classes sont discrètes, tandis qu'en bas les différences sont continues. Les figures de gauche représentent la comparaison des méthodes de type *wrapper*, celles de droites des méthodes de type *filter*. Les données simulées utilisées pour cette comparaison comportent $n = 50$ individus (25 par classe) pour $p = 100$ variables.

[Ambroise and McLachlan, 2002],[Zhu et al., 2008], [Boulesteix et al., 2008]. Ces articles donnent comme règle d'or d'intégrer la sélection d'attributs dans la validation croisée. L'algorithme choisi est représenté sur la figure 1.4. Après sélection de variables sur le jeu d'apprentissage (par une méthode de type *filter* ou *wrapper*), une fonction de classification est calculée puis testée sur le jeu de données de test.

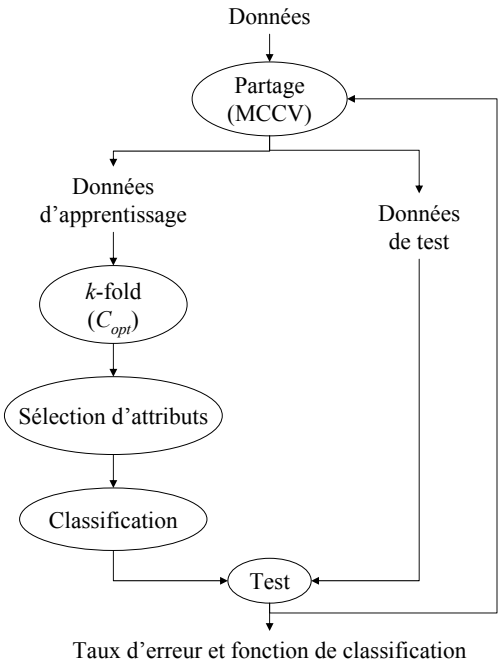


FIGURE 1.4 – Algorithme de classification de données transcriptomiques.

Les paramètres de cet algorithme sont :

(a) l'ensemble des paramètres de régularisation de la méthode de classification ou de régression utilisée, déterminés par *k*-fold, *k* étant fixé à 10.

(b) le nombre de gènes retenus lors de la sélection d'attributs. Nous appliquons de façon générale un filtrage basé sur le FC avec un nombre fixe de gènes sélectionnés à chaque itération.

(c) le nombre de gènes dans la signature moléculaire. Le nombre de gènes dans la signature moléculaire n'est pas fixé *a priori*. On choisit usuellement de ne garder que les gènes qui apparaissent dans plus de 50 % des listes de gènes différentiellement exprimés calculés pendant les phases d'apprentissage. Cette méthode empirique est justifiée par le diagramme présenté figure 1.5, qui montre que le nombre d'occurrences des gènes sélectionnés par la méthode de filtrage est une variable grossièrement bimodale avec deux modes situés en 0 % et environ 90 %.

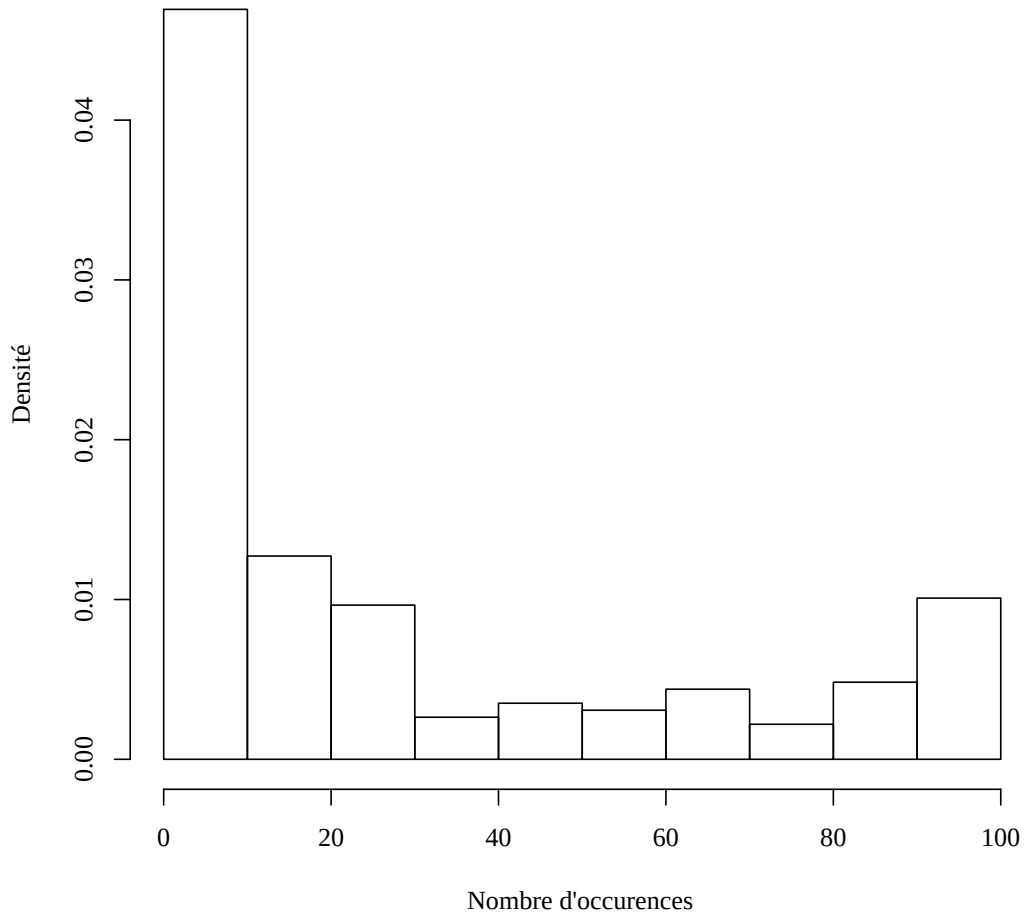


FIGURE 1.5 – Histogramme des occurrences de chaque gène sélectionné dans au moins une étape de sélection d'attributs. Exemple sur les données réelles de Golub et *et al.* ALL/AML [Golub et al., 1999].

1.4 Discussion et perspectives

Les méthodes de régression régularisées détaillées dans cette partie font appel à l'inférence de matrice de variance covariance, comme le montre notamment l'équation 1.3. Nous montrons dans la partie suivante comment nous utilisons cette caractéristique pour intégrer l'information contenue dans un graphe.

Nous avons vu dans cette partie une méthodologie permettant de classer des don-

nées de puce à ADN. Cette méthodologie a été en partie déterminée par une étude sur des données simulées. La structure de covariance a été intégrée de manière empirique dans le modèle en inférant pour chaque classe une matrice de variance covariance à partir d'un jeu de données réelles. Nous montrons dans la partie suivante un modèle de simulation de données gaussiennes prenant en compte un graphe d'indépendances conditionnelles. Il conviendra, dans un travail ultérieur, d'enrichir ce modèle de simulation en intégrant des sources de bruit biologique.

Chapitre 2

Intégration de réseaux de régulations génétiques dans la classification de données transcriptomiques

Nous avons vu dans la partie précédente que la classification de données microarray est fortement contrainte par le fait que le nombre de variables est très supérieur au nombre de mesures. Toute information supplémentaire est donc bienvenue pour améliorer les performances de la fonction de classification. Les réseaux de régulations génétiques (RRG) sont une de ces sources d'information que nous nous proposons d'intégrer.

Le dogme central de la biologie, présenté sur la figure 2.1, définit comment l'information contenue dans l'ADN, les gènes, est transformée en une protéine opérationnelle permettant de déclencher le phénotype pour lequel le gène code.

Sans remettre en cause ce dogme, il est possible de décrire la dynamique de synthèse des protéines comme un ensemble d'interactions entre de multiples molécules : métabolites, protéines, ARN (ARNs messagers, petits ARNs interférents, ARNs de transfert etc.). Un RRG correspond à l'impact qu'ont toutes ces interactions sur le transcriptome. Nous supposons que chaque RRG est particulier d'une situation biologique. Ainsi, dans une expérience à deux classes, on peut disposer de deux RRG différents.

Les méthodes de l'état de l'art intégrant un RRG dans un processus de classification ([Rapaport et al., 2007], [Li and Li, 2008], [Zhu et al., 2009], [Binder and Schumacher, 2009]) ne sont capables d'intégrer qu'un seul RRG supposé commun à toutes les classes. Elles réalisent de plus l'intégration de ce graphe en contraignant la détermination de la fonction de classification de telle sorte que deux variables connectées dans le graphe aient des poids proches dans la fonction de classification. Ces méthodes n'ont pas comme but d'améliorer la qualité de la prédiction, mais l'amélioration de l'interprétabilité du modèle.

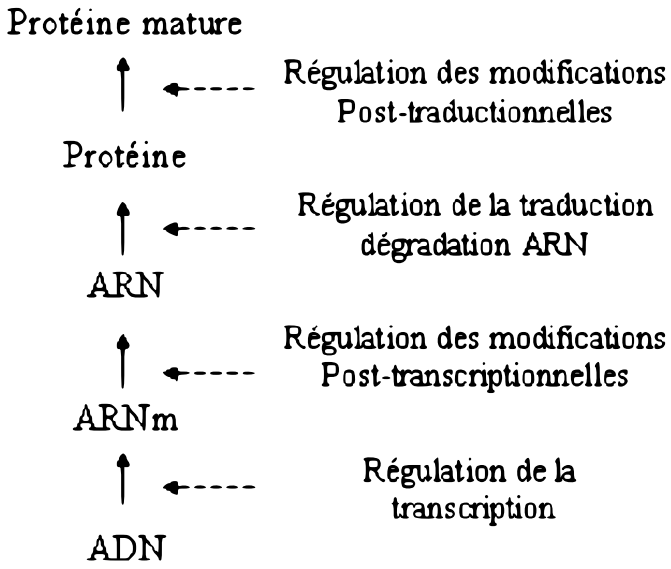


FIGURE 2.1 – Dogme central de la biologie moléculaire : l'information contenue dans la molécule d'ARN est transcrite dans une molécule d'ADN. Après épissage, l'ARN est ensuite traduit en une suite d'acides aminés : une protéine. La protéine obtenue subit éventuellement un ensemble de traitements avant de devenir opérationnelle. Chacune de ces étapes peut être régulée par d'autres molécules produites par d'autres gènes.

Nous proposons de réaliser l'intégration d'un ou plusieurs graphes dans une analyse discriminante de Fisher en utilisant l'information contenue dans ces graphes pour régulariser l'estimation des matrices de variance covariance. Cette manière d'intégrer le graphe ne respecte pas la contrainte imposée par les méthodes de l'état de l'art, à savoir que deux variables connectées dans le graphe doivent avoir des poids proches dans la fonction de classification. Mais nous montrons que sous certaines conditions, les performances de classification sont améliorées significativement.

2.1 Méthodes de l'état de l'art

Les graphes que nous considérons dans la suite de l'exposé sont définis comme suit.

Définition 2

Un graphe $\mathcal{G}$ fini, non dirigé, non valué d'ordre $p \in \mathbb{N}^*$ est défini par l'ensemble de ses nœuds $V = \{1, ..., p\}$ (vertices) *et de ses arêtes $E \subset \{\{i, j\}, i \in V, j \in V\}$* (edges)*. Deux nœuds i et j sont reliés (on dit également adjacents) dans $\mathcal{G}$ dès que $\{i, j\} \in E$.*

(i) on notera $i \sim j$ lorsque deux nœuds i et j sont reliés dans le graphe $\mathcal{G}$.

(ii) $\mathcal{G}$ est vide dès que l'ensemble de ses arêtes est vide.

(iii) $\mathcal{G}$ est complet dès que $E = \{\{i, j\}, i \in V, j \in V\}$.

(iv) *La matrice d'adjacence de $\mathcal{G}$ est une matrice de dimensions $p \times p$, notée A :*

$$[A]_{i,j} \begin{cases} = 1 & \text{si } i \sim j \\ = 0 & \text{sinon} \end{cases}$$

(v) *Le degré d'un nœud est le nombre de nœuds qui lui sont adjacents.*

(vi) *Le Laplacien d'un graphe $\mathcal{G}$ de matrice d'adjacence A et de matrice diagonale des degrés D est noté $L_{\mathcal{G}}$:*

$$L_{\mathcal{G}} = D - A.$$

En génomique fonctionnelle, le problème de l'intégration des co-régulations entre gènes se pose de manière récurrente : cette information peut provenir soit de bases de données alimentées par les biologistes [1] soit d'autres expériences transcriptomiques réalisées sur les mêmes problématiques biologiques. Dans le dernier cas, il faut mettre en œuvre des méthodes d'inférence de RRG ([Tenenhaus et al., 2008], [Schäfer and Strimmer, 2005b], etc.) pour extraire un graphe de ces données supplémentaires.

Des méthodes existent déjà permettant d'intégrer un graphe dans un processus de classification [Li and Li, 2008], [Rapaport et al., 2007], [Zhu et al., 2009]. Les deux premières méthodes utilisent le Laplacien du graphe $\mathcal{G}$ *a priori*. La troisième méthode [Zhu et al., 2009] propose également de réaliser cette intégration sans utiliser le Laplacien du graphe, mais directement la présence ou non d'interaction entre deux variables.

2.1.1 Intégration du réseau par des méthodes à noyau : transformation spectrale du Laplacien du graphe

Dans [Rapaport et al., 2007], une transformation spectrale est appliquée au Laplacien $\mathcal{L}_{\mathcal{G}}$ du graphe $\mathcal{G}$. On note $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_p$ les valeurs propres de $\mathcal{L}_{\mathcal{G}}$ et $e_1, ..., e_p$ les vecteurs propres correspondants. Une décomposition sur les sous-espaces propres de $\mathcal{L}_{\mathcal{G}}$ est similaire à une décomposition de Fourier d'un signal. Si l'on projette un vecteur sur les espaces propres correspondant aux valeurs propres les plus petites, on peut « adoucir » ses variations tout le long du graphe, c'est-à-dire que les composantes du vecteur correspondant à des nœuds adjacents dans le graphe seront rendues plus proches.

Rapaport *et al.* propose de transformer à l'aide d'une fonction ϕ les valeurs propres λ_j, ce qui a pour but d'atténuer l'influence des valeurs propres les plus grandes. Deux fonctions ϕ ont été proposées par Rapaport *et al.*

– une fonction de seuillage :

$$\phi : \lambda \mapsto \begin{cases} 1 \text{ si } 0 \leq \lambda \leq \lambda_0 \\ 0 \text{ sinon} \end{cases},$$

1. Par exemple KinBase : http://kinase.com/kinbase, Biocarta : http://www.biocarta.com/, Pathway Interaction Database : http://pid.nci.nih.gov/

qui élimine toutes les valeurs propres supérieures au seuil $\lambda_0 > 0$.

– une fonction d'atténuation exponentielle :

$$\phi : \lambda \mapsto \exp(-\gamma\lambda),$$

avec γ un paramètre réel positif.

Une fois cette transformation effectuée, la fonction suivante est appliquée aux profils d'expression :

$$z \in \mathbb{R}^p \mapsto \sum_{j=1}^{p} z_j \phi(\lambda_j) e_j, \tag{2.1}$$

avec z un profil d'expression et z_j l'expression du gène j. On peut montrer qu'appliquer cette fonction revient à envoyer les profils d'expressions dans un espace de représentation dual défini par le noyau K_ϕ :

$$K_\phi = \sum_{j=1}^{p} \phi(\lambda_j)^2 e_j e_j^\top.$$

Le résultat de l'utilisation de la transformation proposée dans [Rapaport et al., 2007] est représenté sur la figure 2.2. Un profil d'expression z et un graphe $\mathcal{G}$ ont été générés indépendamment et aléatoirement. Les nœuds du graphe ont été colorés (cf. fig 2.2(a)) suivant les valeurs de z. Comme attendu, on note de fortes variations de couleur sur le graphe. Dans un second temps, la transformation proposée par Rapaport *et al.* (*cf.* équation (2.1)) est appliquée au vecteur z. Le résultat est présenté sur la figure 2.2(b) et on constate un adoucissement notable des variations le long du graphe.

Rapaport *et al.* mettent en œuvre des SVM à noyaux utilisant le noyau K_ϕ. Les auteurs ne prétendent pas améliorer les performances en classification, mais l'interprétabilité biologique du classifieur au sens où si deux gènes sont connectés dans le réseau, leurs poids dans la fonction de classification sont proches.

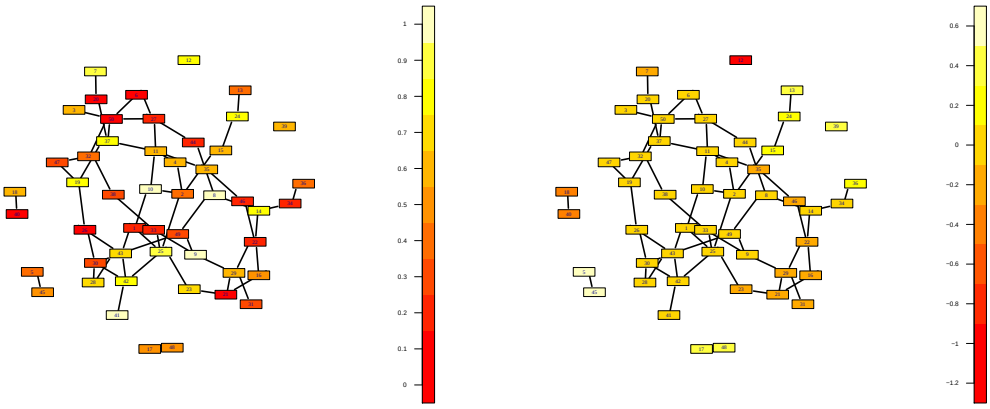
2.1.2 Intégration du Laplacien du graphe dans la partie quadratique de la contrainte d'une régression elastic-net

Dans [Li and Li, 2008], les auteurs proposent d'ajuster un modèle linéaire $y = \boldsymbol{x}\beta$ aux données $\boldsymbol{x}$ pour prédire y. Leur approche consiste à optimiser le critère suivant :

$$J(\beta, c_1, c_2) = \|y - \boldsymbol{x}\beta\|^2 + c_1 |\beta| + c_2 \beta^\top \mathcal{L}_\mathcal{G} \beta$$

en fonction de β, où $\mathcal{L}_\mathcal{G}$ est le Laplacien du graphe *a priori* $\mathcal{G}$. La solution de ce problème d'optimisation est calculée avec la méthode LASSO et mène à un modèle parcimonieux. Les coefficients non nuls de la fonction de classification permettent aux auteurs de mettre en évidence les *pathways* à l'œuvre dans les processus biologiques étudiés.

Les résultats présentés dans l'article [Li and Li, 2008] sur des données simulées (inspirées de [Efron et al., 2002] et détaillées en section 2.2.4.1) montrent que cette méthode est plus performante que les méthodes LASSO et *elastic - net* [Zou and Hastie, 2005].



(a) Sans transformation.

(b) Avec transformation.

FIGURE 2.2 – Exemple d'application d'une transformation sur le Laplacien d'un graphe pour « adoucir » un vecteur de valeurs à travers un graphe donné. Un graphe est généré aléatoirement d'une part. Un vecteur est généré aléatoirement d'autre part. Sur la figure de gauche, le graphe est coloré en fonction des valeurs de ce vecteur aléatoire. Sur la figure de droite, les nœuds du graphe sont colorés en fonction des valeurs prises par le vecteur transformé selon l'équation (2.1).

En revanche l'amélioration des performances en classification de données réelles n'est pas démontrée. Nous reprenons pour cette méthode l'acronyme adopté par les auteurs pour l'identifier : Net.

2.1.3 Intégration des arêtes du graphe dans la contrainte de l'optimisation de SVM linéaires

La méthode *network-based Support Vector Machines* (NB-SVM) est présentée dans [Zhu et al., 2009] et consiste à optimiser le critère suivant :

$$\min_{\beta_0,\beta} \sum_{i=1}^{N} \xi_i + \lambda \sum_{j\sim k} M(j,k)$$

$$\forall i = 1,...,N,\ y_i(\beta_0 + x_i^\top \beta) \geq 1 - \xi_i$$

$$\forall j \sim k,\ \begin{array}{rcl} \left|\frac{\beta_j}{w_j}\right| & \leq & M(j,k) \\ \left|\frac{\beta_k}{w_k}\right| & \leq & M(j,k) \end{array},$$

où $M(j,k) = max\left(\left|\frac{\beta_j}{w_j}\right|, \left|\frac{\beta_k}{w_k}\right|\right)$ et le coefficient w_k est soit égal à d_k, le degré de la variable k dans le graphe utilisé, soit $\sqrt{d_k}$, soit simplement 1. D'après les observations de Zhu *et al.* sur des données simulées et des données réelles, le choix $w_k = d_k$, que nous adoptons pour la comparaison, semble donner de meilleurs résultats en classification.

Les résultats obtenus sur les données simulées utilisées dans [Li and Li, 2008] montrent une amélioration des performances de classification par rapport aux LP-SVM. En revanche, les auteurs ne constatent pas d'amélioration notable sur des données réelles.

2.1.4 Une contrainte commune

Ces trois méthodes de classification permettent d'intégrer le graphe en ayant pour objectif la même propriété de la fonction de classification : deux variables proches dans le graphe doivent avoir des poids similaires dans la classification. Réaliser cette propriété apparaît clairement sous la forme d'une contrainte supplémentaire lorsque ces méthodes sont formulées de la façon suivante :

(i) Net :

$$\min_{\beta_0,\beta} \|y - x\beta - \beta_0\|^2$$

$$\text{t.q. } \alpha\,|\beta| + (1-\alpha)\beta^\top \mathcal{L}_\mathcal{G}\beta \leq t$$

(ii) Méthode de Rapaport *et al.* :

$$\min_{\alpha,x} \sum_{i=1}^{N} \alpha_i^2 + C\xi$$

$$\text{t.q. } \forall i = 1,...,N,\ y_i(\beta_0 + \sum_{j=1}^{N} \alpha_i K(x_i, x_j)) \geq 1 - \xi_i,$$

avec K le noyau dépendant du Laplacien du graphe.

(iii) NB-SVM :

$$\min_{\beta_0,\beta} \|\beta\| + C\sum_{i=1}^{N} \xi_i$$

$$\text{t.q. } \forall i = 1,...,N,\ y_i(\beta_0 + x_i^\top \beta) \geq 1 - \xi_i$$

$$\forall j \sim k, \begin{array}{rcl} \left|\frac{\beta_j}{w_j}\right| & \leq & M(j,k) \\ \left|\frac{\beta_k}{w_k}\right| & \leq & M(j,k) \end{array},$$

où $M(j,k) = max\left(\left|\frac{\beta_j}{w_j}\right|, \left|\frac{\beta_k}{w_k}\right|\right)$.

Ces formulations montrent que les trois méthodes tendent toutes à résoudre un problème de classification avec comme contrainte supplémentaire que deux variables connectées dans le graphe devront avoir des coefficients proches, voire identiques,

dans la fonction de classification. Cette contrainte améliore l'interprétation par le biologiste de la fonction de classification, comme le montrent les résultats présentés par Rapaport *et al.*, Li et Li et Zhu *et al.* sur des données transcriptomiques réelles. Il n'y a cependant pas de raisons particulières pour que cela améliore les performances en classification.

2.2 Approche proposée

Nous proposons une méthode inspirée de l'analyse discriminante, *graph Constrained Discriminant Analysis* [Guillemot et al., 2008a] (notée gCDA dans la suite), qui prend en compte l'information *a priori* contenue dans un RRG. La contrainte explicite dans les méthodes de l'état de l'art imposant à des coefficients de la fonction de décision d'être identiques s'ils sont reliés dans le graphe donné *a priori* n'est pas explicitement imposée à gCDA. Nous réalisons l'intégration de l'information contenue dans le RRG en l'incorporant dans l'estimation de la matrice de variance covariance intra classes utilisées dans l'algorithme de la FDA. Nous montrons sur des données simulées pour lesquelles le graphe à intégrer est connu que cette méthode améliore significativement les performances en classification.

2.2.1 Analyse Discriminante

Les méthodes d'analyse discriminante sont des méthodes de classification supervisée abondamment décrites dans la littérature ([Fisher, 1936], [Saporta, 2006]) et plébiscitées par certains auteurs [Hand, 2006]. Elles sont basées soit sur la maximisation du ratio des variances inter et intra classes pour l'Analyse Discriminante de Fisher [Fisher, 1936], soit sur l'expression d'une fonction de décision optimale au sens du maximum de vraisemblance pour l'Analyse Discriminante du maximum de vraisemblance. Nous nous concentrons sur l'analyse discriminante de Fisher. Elle se déroule en deux étapes :

1. la première consiste à déterminer une transformation linéaire permettant de minimiser la dispersion à l'intérieur de chaque classe et de maximiser la dispersion entre les deux classes,
2. la deuxième à construire une fonction de classification permettant d'attribuer une classe à un nouvel individu.

Pour intégrer l'information contenue dans les RRG, nous nous plaçons dans le cadre des modèles graphiques gaussiens GGM. Nous faisons donc l'hypothèse que les profils d'expression sont des réalisations de variables aléatoires gaussiennes multivariées de moyennes $\mu^{(1)}$ et $\mu^{(2)}$ et de matrices de variance covariance $\Sigma^{(1)}$ et $\Sigma^{(2)}$.

2.2.1.1 Optimiser le ratio de la variance inter classes sur la variance intra classes

Dans ce paragraphe, une transformation linéaire permettant de maximiser le rapport entre variance inter classes et variance intra classes est présentée. Les matrices de

variance covariance inter et intra classes sont définis de la façon suivante :

Définition 3

La matrice de variance covariance inter classe Σ_b vaut

$$\begin{aligned}\Sigma_b &= \pi_1(\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{tot})(\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{tot})^\top + \pi_2(\boldsymbol{\mu}^{(2)} - \boldsymbol{\mu}^{tot})(\boldsymbol{\mu}^{(2)} - \boldsymbol{\mu}^{tot})^\top \\ &= \pi_1\pi_2(\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)})(\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)})^\top.\end{aligned}$$

avec $\boldsymbol{\mu}^{tot} = \pi_1\boldsymbol{\mu}^{(1)} + \pi_2\boldsymbol{\mu}^{(2)}$ la moyenne totale et $\pi_k = P(Y = y_k)$.

La matrice de covariance intra classes Σ_w (ou encore matrice de covariance poolée*) vaut*

$$\Sigma_w = \pi_1\Sigma^{(1)} + \pi_2\Sigma^{(2)}.$$

Par construction, la matrice Σ_b est une matrice symétrique semi définie positive et la matrice Σ_w est symétrique définie positive.

Le théorème suivant permet de déterminer une transformation maximisant le ratio entre variance inter classes et variance intra classes.

Théorème 2

L'application

$$\begin{aligned}\mathbb{R}^p &\to \mathbb{R} \\ u &\mapsto \frac{u^\top\Sigma_b u}{u^\top\Sigma_w u}\end{aligned}$$

admet un maximum sur $\mathbb{R}^p$ qui est atteint pour $u \in E_{max}$, l'espace propre de la matrice $\Sigma_w^{-1}\Sigma_b$ engendré par le vecteur $\Sigma_w^{-1}(\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)})$.

Démonstration : Comme $\Sigma^{(1)}$ et $\Sigma^{(2)}$ sont supposées inversibles, Σ_w est également inversible, donc la solution proposée dans le théorème existe toujours. Nous supposons que $\boldsymbol{\mu}^{(1)} \neq \boldsymbol{\mu}^{(2)}$.

On veut maximiser la fonctionnelle

$$J(u) = \frac{u^\top\Sigma_b u}{u^\top\Sigma_w u}.$$

Ce problème est équivalent au problème suivant :

$$\begin{aligned}&\min_u -\frac{1}{2}u^\top\Sigma_b u, \\ &\text{t.q. } u^\top\Sigma_w u = 1\end{aligned}$$

et peut se réécrire de la façon suivante en utilisant les multiplicateurs de Lagrange (les facteurs $1/2$ ont été rajoutés par commodité)

$$\mathcal{L}(u, \lambda) = -\frac{1}{2}u^\top\Sigma_b u + \frac{1}{2}\lambda\left(u^\top\Sigma_w u - 1\right).$$

En dérivant cette expression par rapport à u et λ, on en arrive au problème suivant :

$$\begin{aligned} \Sigma_b u &= \lambda \Sigma_w u \\ \text{et } u^\top \Sigma_w u &= 1. \end{aligned} \tag{2.2}$$

Comme Σ_w est inversible, l'équation 2.2 est équivalente à l'équation suivante :

$$\Sigma_w^{-1} \Sigma_b u = \lambda u \tag{2.3}$$

Les solutions de (2.3) appartiennent à l'ensemble $\mathrm{Im}(\Sigma_w^{-1})$ et sont donc de la forme $\Sigma_w^{-1} v$, avec $v \in \mathbb{R}^p$. En remplaçant dans (2.3), on obtient

$$\begin{aligned} &\Sigma_w^{-1} \Sigma_b \Sigma_w^{-1} v = \lambda \Sigma_w^{-1} v \\ \Leftrightarrow &\Sigma_b \Sigma_w^{-1} v = \lambda v \\ \Leftrightarrow &\pi_1 \pi_2 (\mu^{(1)} - \mu^{(2)}) \underbrace{(\mu^{(1)} - \mu^{(2)})^\top \Sigma_w^{-1} v}_{\text{scalaire}} = \lambda v. \end{aligned}$$

Nécessairement, l'égalité est vérifiée dès que v est proportionnel au vecteur $\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)}$. Les solutions de (2.2) sont donc dans l'espace $\mathrm{Vect}\, \Sigma_w^{-1}(\mu^{(1)} - \mu^{(2)})$, vecteur propre de la matrice $\Sigma_w^{-1}\Sigma_b$. De plus, comme Σ_b est de rang 1, $\Sigma_w^{-1}\Sigma_b$ est aussi de rang 1. Donc l'espace vectoriel $\mathrm{Vect}\, \Sigma_w^{-1}(\mu^{(1)} - \mu^{(2)})$ est l'espace propre associé à l'unique valeur propre de la matrice $\Sigma_w^{-1}\Sigma_b$. ■

Le sous espace propre ainsi défini est de dimension 1. Ainsi, après cette étape de maximisation du ratio des variances inter et intra classes, on aura également procédé à une réduction de dimension : les données à p variables seront résumées à une seule droite, appelée l'axe discriminant de Fisher. Un exemple de projection des mesures sur l'axe discriminant de Fisher est présenté figure 2.3.

Cette transformation permet de déterminer un axe discriminant le long duquel les deux classes de mesures sont le plus aisément discernables. Il s'agit ensuite de déterminer une fonction de classification sur cet axe qui permettra d'attribuer une classe à un nouvel individu.

2.2.1.2 Déterminer la classe d'un nouvel individu

Soit un nouvel individu $z \in \mathbb{R}^p$, on pose $z_F = V^\top z$, avec V les coordonnées du vecteur normal à l'axe discriminant de Fisher (autrement dit V est l'application linéaire permettant de maximiser le ratio entre variance intra classes et variance inter classes). Plusieurs fonctions de classification sont possibles une fois que le nouvel individu z est projeté sur l'axe discriminant :

1. une fonction de distance simple entre le nouvel individu et le centre des classes : on attribue à x la classe la plus proche,

$$d(z) = (\mu^{(2)} - z)^2 - (\mu^{(1)} - z)^2 = 2(\mu^{(2)} - \mu^{(1)}) \left(z - \frac{\mu^{(1)} + \mu^{(2)}}{2} \right).$$

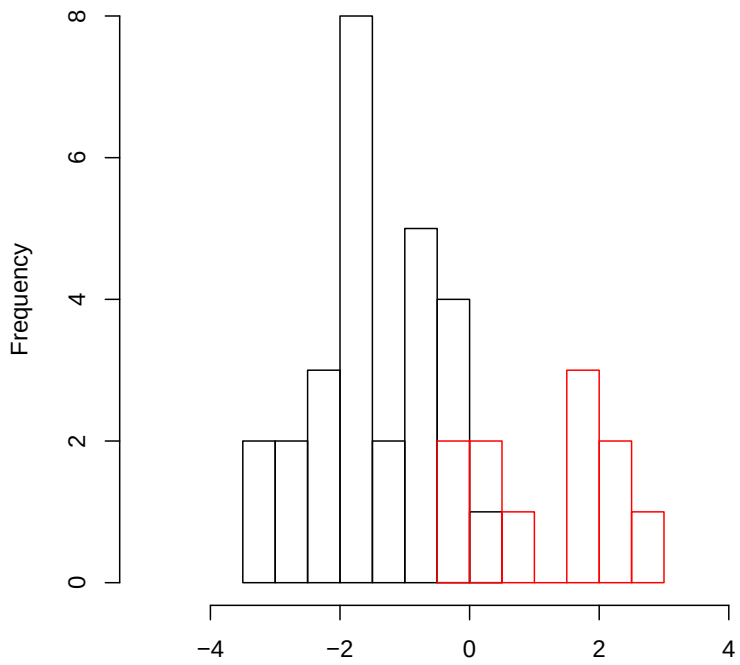


FIGURE 2.3 – Projections des échantillons d'apprentissage sur l'axe discriminant de Fisher pour les données de [Golub et al., 1999]. Les échantillons de la classe 1 sont représentés par le diagramme en noir et les échantillons de la classe 2 en rouge.

2. une distance basée sur la maximisation de la probabilité *a posteriori*, lorsque les données sont supposées gaussiennes.

La deuxième approche permet de déterminer une fonction de classification optimale au sens de l'erreur bayésienne.

On note $Z_F^{(k)}$ la variable aléatoire univariée modélisant le comportement de z_F. Sa densité s'exprime en fonction de la densité de $\boldsymbol{X}^{(k)}$:

$$Z_F^{(k)} = V^\top \boldsymbol{X}^{(k)} \sim \mathcal{N}\left(\mu^{(k)} = V^\top \boldsymbol{\mu}^{(k)}, \sigma^{(k)2} = V^\top \Sigma^{(k)} V\right).$$

On peut appliquer à une nouvelle réalisation z_F de cette variable la fonction de classification suivante :

$$\delta(z_F) = \left(-\frac{(z-\mu^{(1)})^2}{2\sigma^{(1)2}} - \ln \sigma^{(1)} + \ln \pi_1\right) - \left(-\frac{(z-\mu^{(2)})^2}{2\sigma^{(2)2}} - \ln \sigma^{(2)} + \ln \pi_2\right).$$

On peut montrer que cette fonction de classification est la fonction de décision qui minimise le risque bayésien. Ainsi, si $\delta(z_F) > 0$, on décide que z appartient à la classe 1, et si $\delta(z_F) \leq 0$, on décide que z appartient à la classe 2. De plus, en remplaçant z_F par $V^\top z$, on peut exprimer la fonction de classification $\delta(z)$ (par abus de notation, les deux fonctions sont notées δ) en fonction des paramètres des lois des $\boldsymbol{X}^{(k)}$ en utilisant la relation suivante :

$$\frac{(z-\mu^{(k)})^2}{2\sigma^{(k)2}} = \frac{(z-\boldsymbol{\mu}^{(k)})^\top VV^\top(z-\boldsymbol{\mu}^{(k)})}{2V^\top\Sigma^{(k)}V}.$$

La fonction de classification devient :

$$\begin{aligned}\delta(z) =& \frac{(z-\boldsymbol{\mu}^{(2)})^\top VV^\top(z-\boldsymbol{\mu}^{(2)})}{2V^\top\Sigma^{(2)}V} - \frac{(z-\boldsymbol{\mu}^{(1)})^\top VV^\top(z-\boldsymbol{\mu}^{(1)})}{2V^\top\Sigma^{(1)}V} \\ &+ \frac{1}{2}\ln V^\top\Sigma^{(2)}V - \frac{1}{2}\ln V^\top\Sigma^{(1)}V + \ln\pi_1 - \ln\pi_2.\end{aligned}$$

La fonction de classification δ est donc linéaire quand $\Sigma^{(1)} = \Sigma^{(2)} = \Sigma$:

$$\begin{aligned}\delta(z) =& \frac{(\boldsymbol{\mu}^{(2)} - \boldsymbol{\mu}^{(1)})^\top VV^\top}{V^\top\Sigma V}z + \frac{\boldsymbol{\mu}^{(2)\top}VV^\top\boldsymbol{\mu}^{(2)} - \boldsymbol{\mu}^{(1)\top}VV^\top\boldsymbol{\mu}^{(1)}}{V^\top\Sigma V} \\ &+ \ln V^\top\Sigma^{(2)}V - \ln V^\top\Sigma^{(1)}V + \ln\pi_1 - \ln\pi_2.\end{aligned}$$

$\delta(z)$ peut s'écrire sous la forme $\delta(z) = \beta_0 + \beta^\top z$.

Dans le cas où $\Sigma^{(1)} \neq \Sigma^{(2)}$, δ est une fonction quadratique de z.

2.2.1.3 Estimations des paramètres de δ

Les estimateurs utilisés classiquement pour les matrices de covariances inter et intra classes sont les suivants :

$$S_b = \frac{1}{n}\sum_{k=1}^{2} n_k(\overline{\boldsymbol{x}}^{(k)} - \overline{\boldsymbol{x}})(\overline{\boldsymbol{x}}^{(k)} - \overline{\boldsymbol{x}})^\top \text{ et}$$

$$S_w = \frac{1}{n-2}\left((n_1-1)\widehat{\Sigma}^{(1)} + (n_2-1)\widehat{\Sigma}^{(2)}\right),$$

avec $\overline{\boldsymbol{x}}^{(k)}$ la moyenne empirique de la classe k et $\overline{\boldsymbol{x}}$ la moyenne empirique totale.

Les estimateurs usuels sont utilisés dans le cas $p > n$:

- pour π_k : $\widehat{\pi}_k = \frac{n_k}{n_1+n_2}$,
- pour $\boldsymbol{\mu}^{(k)}$: $\overline{\boldsymbol{x}}^{(k)}$,
- et pour $\Sigma^{(k)}$: $\widehat{\Sigma}^{(k)} = S^{(k)}$.

Dans le cas $n \ll p$ qui nous intéresse, les estimateurs $S^{(k)}$ ne sont pas inversibles, et donc il n'est plus possible de calculer la fonction de classification de l'analyse discriminante.

2.2.2 Analyse discriminante régularisée

Si $n \ll p$, alors il faut régulariser l'estimation de Σ_w, ce qui usuellement effectué par la régularisation de l'estimation de $\Sigma^{(1)}$ et $\Sigma^{(2)}$.

Dans [Dudoit et al., 2002], la méthode *Diagonal Discriminant Analysis* (DDA) est présentée. Elle consiste à supposer que tous les termes non diagonaux de la matrice de variance covariance sont nuls :

$$\widehat{\Sigma}^{dda} = diag(\sigma_1^2, ..., \sigma_p^2).$$

Dans [Friedman, 1998], les auteurs proposent, dans le cadre de leur méthode appelée *Regularized Discriminant Analysis* (RDA), de reprendre l'estimateur classique à la diagonale duquel est rajoutée un terme constant :

$$\widetilde{\Sigma}^{rda} = \alpha \widehat{S} + (1 - \alpha) I. \tag{2.4}$$

Les estimateurs utilisés dans DDA est RDA sont biaisés mais présentent une variance moindre que l'estimateur S.

D'autres estimateurs régularisés de matrices de variance covariance sont présentés dans la littérature, mais ne sont pas encore implémentés dans un cadre d'analyse discriminante. On peut par exemple citer les estimateurs présentés dans [Schäfer and Strimmer, 2005b]. Ils sont de la même forme que l'estimateur utilisé dans la méthode RDA :

$$\widetilde{\Sigma}^{schaefer} = \alpha \widehat{S} + (1 - \alpha) T, \tag{2.5}$$

avec T une matrice cible, calculée à partir des corrélations et variances des échantillons considérés. L'intérêt du travail mené par Schäfer et Strimmer est double car il présente d'une part 6 matrices cibles T et également une méthode permettant de déterminer analytiquement une valeur du paramètre α minimisant la MSE.

Nous proposons un nouvel estimateur pour Σ qui utilise l'information contenue dans le graphe $\mathcal{G}$, ce qui permet d'obtenir un estimateur inversible et régularisé tout en intégrant $\mathcal{G}$ dans l'algorithme de classification.

2.2.3 Intégration de $\mathcal{G}$ dans l'estimation de Σ

Soit un graphe $\mathcal{G} = \{E, V\}$ à intégrer dans l'estimation de la matrice de variance covariance Σ d'une variable $\boldsymbol{X} = (X_1, ..., X_p)$ gaussienne multivariée. Dans le cadre des GGM, $\mathcal{G}$ décrit les relations d'indépendances conditionnelles entre les variables $(X_1, ..., X_p)$. Lorsqu'il n'y a pas d'arête entre les nœuds j_1 et j_2, les variables X_{j_1} et X_{j_2} sont indépendantes conditionnellement à toutes les autres, ce qui est noté

$$X_{j_1} \perp\!\!\!\perp X_{j_2} | \left\{ X_j, j \neq j_1, j_2 \right\}.$$

On admettra [Whittaker, 1990] que dans ce cas, le coefficient en place (j_1, j_2) de la matrice de précision $\Omega = \Sigma^{-1}$ de $\boldsymbol{X}$ doit être nul. La donnée de $\mathcal{G}$ contraint ainsi fortement la matrice de covariance Σ, sans toutefois la déterminer de façon unique dans le cas général.

Les propositions existantes pour un modèle entre $\mathcal{G}$ et Σ dépendent directement de la décomposition du graphe en cliques, s'il est décomposable. Deux exemples simples permettent d'évaluer le nombre de degrés de liberté qu'il faut fixer pour des graphes d'indépendance conditionnelle très particuliers.

$\mathcal{G}$ a une forme particulière

1. $\mathcal{G}$ est vide. Dans ce cas la matrice Σ correspondant est une matrice diagonale avec des termes diagonaux positifs : il y a p coefficients (diagonaux) libres.
2. $\mathcal{G}$ est complet. Tous les coefficients de la matrice de précision sont libres : il y a $p(p+1)/2$ degrés de liberté.

Ainsi, plus le nombre de cliques d'un graphe $\mathcal{G}$ décomposable est grand [Letac and Massam, 2007] plus il y a de paramètres libres à déterminer pour en déduire une matrice de précision acceptable. Mais bien que le nombre de paramètres libres soit grand, il est possible de les déterminer de façon unique [Letac and Massam, 2007], [Rajaratnam et al., 2008].

$\mathcal{G}$ est quelconque

La plupart des méthodes permettant de déterminer Σ à partir de $\mathcal{G}$ ne peuvent s'appliquer que dans le cas où le graphe est décomposable. Une condition nécessaire pour qu'un graphe soit décomposable est qu'il ne comporte aucun cycle sans corde d'ordre 4 (voir par exemple [Whittaker, 1990]). Cette condition est trop restrictive pour l'appliquer de façon générale à des graphes de régulations génétiques.

Les travaux récents de Rajaratnam *et al.* [Rajaratnam et al., 2008] permettent d'étendre ces méthodes aux cas où le graphe $\mathcal{G}$ n'est pas décomposable. Ils peuvent ainsi fournir des pistes sur le nombre de paramètres libres lorsque l'on veut passer de $\mathcal{G}$ à Σ en fonction des propriétés de $\mathcal{G}$.

Déterminer un candidat Σ en adéquation avec $\mathcal{G}$

Bien que les travaux présentés ci-dessus permettent de déterminer les paramètres libres de Σ, ils ne fournissent pas de cadre analytique aisément implémentable. Le prix de cette détermination est de plus un coût en temps de calcul élevé. Nous allons donc, par souci de simplicité, restreindre au maximum le nombre de degrés de liberté.

Nous avons déterminé deux contraintes sur Σ pour minimiser le nombre de degrés de liberté de notre modèle :

1. la matrice de précision doit être telle que $\Omega_{i,j} = 0$ si et seulement si $i \nsim j$ dans $\mathcal{G}$,
2. Σ doit être définie positive, ce qui est équivalent à avoir Ω définie positive.

La première contrainte est respectée par la matrice d'adjacence de $\mathcal{G}$. Pour obtenir une matrice respectant les deux contraintes, il suffit de lui rajouter d'abord la matrice diagonale des degrés de chaque nœud, et ensuite une constante strictement positive.

Dans tout la suite du rapport, nous considérerons le modèle suivant : $\Omega = D - A + I = L + I$, avec D la matrice diagonale des degrés, A la matrice d'adjacence et $L_{\mathcal{G}}$ le

Laplacien de $\mathcal{G}$. Ω respecte les deux contraintes énoncées plus haut. Nous avons choisi de garder dans Ω le Laplacien du graphe pour garder une homogénéité méthodologique avec les méthodes présentées dans [Rapaport et al., 2007] et [Li and Li, 2008].

Intégration de $\mathcal{G}$ dans l'estimation de Σ

Dans la continuité du formalisme de [Friedman, 1998, Schäfer and Strimmer, 2005b], nous proposons d'intégrer le graphe $\mathcal{G}$ en considérant l'estimateur suivant pour Σ :

$$\widehat{\Sigma}^{(gcda)} = \alpha S + (1-\alpha)\left(D - A + I\right)^{-1}. \tag{2.6}$$

Le nouvel estimateur ainsi obtenu est défini positif et réalise un compromis entre l'information contenue dans les données (S) et celle contenue dans le graphe $\mathcal{G}$.

2.2.4 Intégration du graphe *a priori* dans la méthode gCDA

Nous avons vu que dans le cadre des GGM, le graphe d'indépendances conditionnelles contient une information centrale sur la matrice de variance covariance du graphe. Or l'estimation de la matrice de variance covariance intervient dans le calcul de la fonction de classification. On peut vérifier expérimentalement qu'une mauvaise estimation de la matrice de variance covariance conduit à de mauvaises performances en prédiction de l'analyse discriminante. Plus précisément, on vérifie cette dégradation des qualités de prédiction en augmentant le nombre de variables à nombre d'individus constant. Les résultats sont présentés figure 2.4. Cette figure montre des résultats obtenus sur des données simulées selon le modèle décrit 2.2.4.2 : nous avons généré deux classes de profils d'expression et construits deux classifieurs différents testés ensuite sur des données indépendantes. Le premier classifieur est calculé de façon classique, c'est à dire que la matrice de variance covariance est inconnue et doit donc être estimée (classiquement ici avec un l'estimateur S non-biaisé). Pour le deuxième classifieur, on suppose que l'on connait déjà la matrice de variance covariance. Les performances en classification sont excellentes lorsque l'on considère la variance connue dans le cas où le nombre de variables vaut $p = 200$ pour un nombre d'individus deux fois plus petit $n = 100$. En revanche, quand le nombre de variables est raisonnablement plus petit que le nombre d'individus, l'estimateur S est suffisamment précis.

Dans gCDA, l'estimation de la matrice de variance covariance est régularisée de la façon suivante :

$$\widehat{\Sigma}^{gcda} = \alpha S + (1-\alpha)\left(D - A + I\right)^{-1}.$$

Ainsi, l'estimateur de la matrice de variance covariance intra classes devient :

$$\begin{aligned}\widehat{\Sigma}_w^{gcda}(\alpha) &= \frac{n_1}{n}\widehat{\Sigma}_1^{gcda}(\alpha) + \frac{n_2}{n}\widehat{\Sigma}_2^{gcda}(\alpha) \\ &= \alpha\left(\frac{n_1}{n}S^{(1)} + \frac{n_2}{n}S^{(2)}\right) + (1-\alpha)\left(\frac{n_1}{n}\left(D^{(1)} - A^{(1)} + I\right)^{-1} + \frac{n_2}{n}\left(D^{(2)} - A^{(2)} + I\right)^{-1}\right)\end{aligned}$$

avec $D^{(k)}$ et $A^{(k)}$ les matrices des degrés et d'adjacence du graphe associé à la classe k.

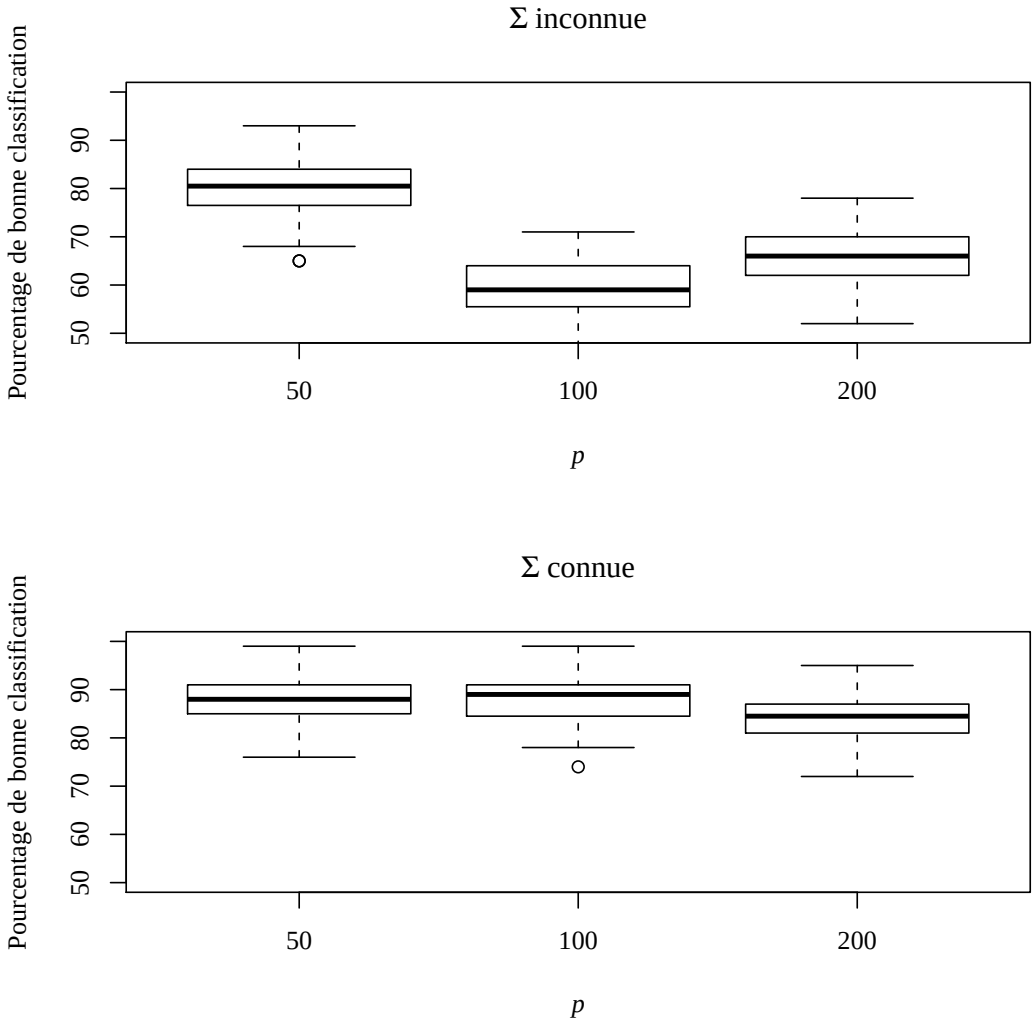


FIGURE 2.4 – Comparaison des performances en classification de l'analyse discriminante classique (Σ inconnue) et de l'analyse discriminante quand *Sigma* est connue. Le nombre d'individus est constant ($n = 100$) et réparti entre les deux classes.

On peut ainsi mener un analyse discriminante linéaire ou quadratique tout en prenant en compte la structure de corrélation existant dans $\mathcal{G}$:

(i) gCDA linéaire : chaque classe est supposée avoir la même covariance et donc le même graphe de régulations :

$$\widetilde{\Sigma}_w(\alpha) = \alpha \left(\frac{n_1}{n} S^{(1)} + \frac{n_2}{n} S^{(2)} \right) + (1 - \alpha) \left(D - A + I \right)^{-1}.$$

(ii) gCDA quadratique : chaque classe présente une structure de corrélation qui lui est propre :

$$\widetilde{\Sigma}_w(\alpha) = \alpha \left(\frac{n_1}{n} S^{(1)} + \frac{n_2}{n} S^{(2)} \right) + (1-\alpha) \sum_{k=1}^{2} \frac{n_k}{n} \left(D^{(k)} - A^{(k)} + I \right)^{-1}.$$

Le paramètre α est déterminé par une validation croisée de type k-fold.

Les connaissances dont on dispose sur les processus biologiques que l'on veut comparer ne nous permettent pas toujours de déterminer quelle version de gCDA utiliser. Des tests multivariés permettent de contrôler l'homogénéité de 2 matrices de variances covariances en testant l'hypothèse nulle suivante :

$$\mathcal{H}_0 : \Sigma^{(1)} = \Sigma^{(2)}.$$

L'intérêt de ce genre de tests est donc tout trouvé avant de mener gCDA : si $\mathcal{H}_0$ est rejetée, on appliquera la version quadratique, sinon, la version linéaire. Le test proposé par Schott en 2007 [Schott, 2007] est particulièrement adaptée au cas où $n \ll p$.

Nous nous sommes intéressés à deux types de jeux de données simulées. Le premier est directement inspiré de [Li and Li, 2008] (qui lui même prend son inspiration chez [Efron et al., 2002]), avec le paramétrage de [Zhu et al., 2009]. Le deuxième est inspiré des modèles graphiques gaussiens.

2.2.4.1 Modèle de simulation issu de [Li and Li, 2008]

Le premier modèle utilisé correspond à un graphe d'indépendances entre variables très simple : il y a dans le jeu de données K gènes, appelés les facteurs de transcription, qui sont reliés directement à 10 gènes chacun, les gènes régulés. On a donc un graphe de $11K$ gènes avec $10K$ arêtes [2]. Le niveau d'expression X_t des facteurs de transcription suit une loi normale $X_t \sim \mathcal{N}(0,1)$, les niveaux d'expression des facteurs de transcription sont indépendants deux à deux. Chaque facteur de transcription est modélisé par une variable $X_t \sim \mathcal{N}(0,1)$. Il régule 10 autres gènes (qui ne sont pas eux même des facteurs de transcription) de telle sorte que le couple facteur de transcription, gène régulé ait une expression suivant une loi gaussienne bivariée de covariance 0.7. Sachant le niveau d'expression X_t du facteur de transcription, le niveau d'expression du gène régulé suit une loi gaussienne $\mathcal{N}(0.51X_t, 0.7)$. On peut montrer [Monfort, 1996] que la variable aléatoire multivariée $\boldsymbol{X}$ constituée par l'ensemble de ces facteurs de transcription et de gènes régulés est une variable aléatoire gaussienne.

On construit ensuite la réponse y à partir d'une réalisation $\boldsymbol{x}$ de $\boldsymbol{X}$ de sorte que $y = \boldsymbol{x}\beta + \beta_0$ avec la constante $\beta_0 = 2$ et le vecteur β tel que ([Zhu et al., 2009], premier

2. Ce qui représente un taux d'arêtes présentes de $20/(11(11K-1))$, soit environ 1.6 % pour $K = 10$, 0.16 % pour $K = 100$ etc.

modèle de simulation) :

$$\beta = \left(5, \underbrace{\frac{5}{\sqrt{10}}, ..., \frac{5}{\sqrt{10}}}_{\text{10 fois}}, -5, \underbrace{\frac{-5}{\sqrt{10}}, ..., \frac{-5}{\sqrt{10}}}_{\text{10 fois}}, \underbrace{0, ..., 0}_{11(K-2) \text{ fois}} \right).$$

Pour discrétiser y, on lui applique une fonction sigmoïde. Dans cette simulation, β est épars. Une bonne méthode de classification permettrait d'identifier les variables associées aux coefficients non nuls de β. Une méthode intégrant le graphe connu des régulations devrait en outre permettre de retrouver les deux premières composantes connexes du graphe impliquées dans la différence entre les deux classes.

2.2.4.2 Modèle de simulation proposé

Pour le deuxième modèle, nous faisons l'hypothèse qu'à chaque classe est associée un graphe d'indépendances (aléatoire et généré grâce à l'algorithme d'Erdős-Rényi [Erdős and Rényi, 1959]). De ce graphe $\mathcal{G}_k, k = 1, 2$ nous déduisons une matrice de précision $\Omega_k = D_k - A_k + I$ et $\Sigma_k = \Omega_k^{-1}$. Les différences entre les deux classes sont modélisés par une variable gaussienne : $\boldsymbol{\mu}^{(2)} - \boldsymbol{\mu}(1) \sim \mathcal{N}(0, \sigma^2 I)$.

Une fois $\Sigma^{(k)}$ déterminée à partir de $\mathcal{G}_k$, on peut générer un échantillon aléatoire de n_k réalisations d'une variable gaussienne multivariée de moyenne quelconque et de matrice de variance-covariance $\Sigma^{(k)}$. Nous avons choisi pour cela un algorithme basé sur une décomposition de Choleski de $\Sigma^{(k)}$.

Un exemple de graphes correspondant au deux modèles de simulation proposés est présenté sur les figures 2.5(a) et 2.5(b). On remarque que le graphe 2.5(a), associé au modèle de [Li and Li, 2008, Zhu et al., 2009], est assez caricatural (pour des données biologiques) avec notamment des nœuds qui sont soit de degré 10, soit de degré 1. Au contraire, la distribution des degrés des nœuds du graphe 2.5(b), associé au second modèle, correspond à la distribution « sans échelle » attendue pour des RRG.

2.2.4.3 Résultats sur les deux simulations

Nous choisissons de ne faire varier que le nombre de variables p. Le nombre d'individus est fixé à $n_1 = n_2 = 50$. Les résultats sur les données simulées sont présentés dans la table 2.1. Les taux de bonne classification sont calculés par validation croisée de type MCCV. Les paramètres de régularisation sont estimés par k-fold. Pour la méthode gCDA, nous avons considéré le modèle $\Omega = D - A + I$ de correspondance entre Σ et $\mathcal{G}$.

Les résultats obtenus sur les données simulées montrent le bon comportement de gCDA. Sur le modèle de simulation que nous proposons (qui correspond au type 2), les performances de gCDA sont meilleures que celles de LP-SVM, NB-SVM et Net. De plus, gCDA prend bien en compte le graphe qu'on lui propose, puisque l'on peut voir sur la figure 2.6 que les valeurs optimales sélectionnées pour α (dans le cas linéaire) sont proches de 0.

	LP-SVM	NB-SVM	gCDA linéaire	gCDA quadratique	Net
Simulation type 1					
$p = 55, K = 5$	95.1 (3.8)	80.3 (6.9)	94.9 (3.5)	- (-)	49.6 (6.9)
$p = 110, K = 10$	90.1 (5.1)	92.1 (5.1)	81.1 (6.2)	- (-)	48.6 (5.4)
$p = 220, K = 20$	83.4 (6.0)	90.7 (5.4)	81.1 (5.7)	- (-)	48.8 (5.4)
Simulation type 2 (linéaire)					
$p = 50$	70.7 (9.6)	64.7 (7.3)	89.3 (4.4)	- (-)	50.2 (6.5)
$p = 100$	95.2 (3.2)	89.8 (4.1)	96.3 (5.3)	- (-)	49.7 (5.3)
$p = 200$	73.4 (9.6)	64.9 (9.1)	85.2 (5.4)	- (-)	48.3 (7.1)
Simulation type 2 (quadratique)					
$p = 50$	61.9 (8.0)	56.7 (8.9)	- (-)	60.8 (7.4)	50.8 (6.6)
$p = 100$	66.3 (8.1)	65.8 (8.0)	- (-)	77.4 (7.2)	51.0 (6.2)
$p = 200$	65.8 (8.6)	64.1 (10.2)	- (-)	72.7 (6.0)	58.6 (5.9)

TABLE 2.1 – Résultats des simulations effectuées, les moyennes des taux de bonne classification sont reportés dans ce tableau, avec entre parenthèse les écarts-types associés. La méthode utilisée pour les SVM est la méthode LP-SVM, ce qui permet d'obtenir à la fin de l'algorithme une fonction de décision éparse. Cette méthode sert de référence sans intégration de graphe. Les autres méthodes présentées sont la méthode NB-SVM de Zhu *et al.*, la méthode Net de Li et Li et gCDA linéaire et quadratique. p est le nombre de variables, K le nombre de facteurs de transcriptions.

En revanche, les résultats obtenus sur les données simulées de [Li and Li, 2008] et [Zhu et al., 2009] (qui correspond au type 1) montrent que les méthodes LP-SVM et NB-SVM sont meilleures que notre méthode quand $p > n$. On peut s'interroger sur les mauvaises performances de gCDA. Nous avons identifié trois causes possibles :

- le modèle $\Omega = D - A + I$ n'est pas correct,
- pour $p = 220$, le vrai β contient 198 valeurs nulles, or l'analyse discriminante est moins adaptée que NB-SVM à des modèles épars,
- une seule variable aléatoire gaussienne est à l'origine des deux classes de ces données simulées.

Afin d'expliciter le modèle utilisé pour passer du graphe à la matrice de variance covariance dans les simulations proposées par Li et Li, nous nous proposons d'étudier un exemple simple d'un facteur de transcription régulant l'expression de deux gènes. Nous montrons l'expression de la matrice de variance covariance formée par les trois variables considérées. Soit $T \sim \mathcal{N}(0,1)$ modélisant l'expression du facteur de transcription, R_1 et R_2 les expressions de deux gènes régulés par ce facteur de transcription et telles que $R_1|T \sim \mathcal{N}(aT, b)$ et $R_2|T \sim \mathcal{N}(aT, b)$, avec $a \in \mathbb{R}$ et $b > 0$.

Les paramètres de la matrice de variance covariance s'expriment en fonction de a et b [Monfort, 1996] :

– $\text{var}(R_1) = E(\text{var}(R_1|T)) + \text{var}(E(R_1|T)) = b + a^2 = \text{var}(R_2)$,
– $\text{cov}(T, R_1) = E(TR_1) = E(TE(R_1|T)) = a = \text{cov}(T, R_2)$,
– $\text{cov}(R_1, R_2) = E(E(R_1|T)E(R_2|T)) = a^2$,

D'où la matrice de variance covariance Σ correspondant aux trois variables

$$\Sigma = \begin{bmatrix} 1 & a & a \\ a & a^2 + b & a^2 \\ a & a^2 & a^2 + b \end{bmatrix},$$

dont le déterminant vaut b^2. La matrice de précision vaut donc

$$\Sigma^{-1} = \begin{bmatrix} 1 + 2a^2/b & -a/b & -a/b \\ -a/b & 1/b & 0 \\ -a/b & 0 & 1/b \end{bmatrix}.$$

Ces résultats sont aisément généralisables au cas de K facteurs de transcription régulant l'expression de 10 gènes chacun, la matrice de précision Σ^{-1} de la variable considérée comprend donc K blocs diagonaux et des 0 partout ailleurs :

$$\Sigma^{-1} = \begin{bmatrix} B & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{0} & B \end{bmatrix}.$$

Chaque bloc B est une matrice carrée de taille 11×11 de la forme :

$$B = \begin{bmatrix} 1 + 2a^2/b & -a/b & \cdots & \cdots & -a/b \\ -a/b & 1/b & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -a/b & 0 & \cdots & 0 & 1/b \end{bmatrix}.$$

Pour obtenir une matrice Σ^{-1} de cette forme, nous avons supposé que les variables sont rangées de la même manière que celle qui est utilisée par Li et Li : elles sont rangées en K groupes de 11 variables, chaque groupe comportant en premier un facteur de transcription suivi des 10 gènes qu'il régule.

Nous remarquons que ce modèle est différent du modèle $\Omega = D - A + I$ que nous avons proposé pour simuler des données. Cependant, même lorsque le vrai modèle est intégré dans gCDA, les performances en classification ne sont pas améliorées, comme l'illustre la figure 2.7. Pour obtenir cette figure, nous avons choisi de ne comparer gCDA qu'avec la méthode NB-SVM pour la situation $n = 100$, $K = 20$ et $p = 220$.

Cette différence de performances en classification entre NB-SVM et gCDA persiste lorsque le vrai β n'est plus parcimonieux (données non représentées). Les mauvaises performances de gCDA par rapport à NB-SVM ne peuvent donc s'expliquer que par

le fait que les jeux de données simulées de Li et Li relèvent d'un problème de classification essentiellement différent de celui qui est résolu par analyse discriminante. L'analyse discriminante suppose en effet que les deux classes sont des réalisations de deux variables $\boldsymbol{X}^{(1)}$ et $\boldsymbol{X}^{(2)}$ différentes en moyenne alors que le modèle de simulation de données de [Li and Li, 2008] suppose que les deux classes sont des réalisations d'une même variable aléatoire $\boldsymbol{X}$, gaussienne, dont les moments sont détaillés plus haut. Ceci montre une des limites de la méthode gCDA.

2.3 Discussion et perspectives

Les méthodes de l'état de l'art permettent d'intégrer un graphe dans le processus de classification avec l'objectif d'obtenir des coefficients de β qui sont proches, voire identiques, si les variables correspondantes sont connectées dans le graphe à intégrer. Or, l'intégration effectuée par le biais de cette contrainte n'améliore pas significativement les performances en classification, et ces performances sont même inchangées quelle que soit la qualité du graphe intégré. Ainsi, comme le montre la figure 2.2, même s'il n'existe aucune relation entre un vecteur et un graphe, il est possible d'obtenir avec ce genre de contrainte des résultats qui semblent satisfaisants à « l'œil nu ». Il est donc très dangereux d'interpréter les coefficients de la fonction de classification obtenues avec ces méthodes sans avoir vérifié de façon préalable la qualité du graphe inféré.

Pour les données simulées, nous connaissions le graphe d'indépendances sous-jacent, alors que ce n'est jamais le cas pour des données transcriptomiques (ou même des données biologiques en général). Les méthodes de l'état de l'art présentées plus haut [Rapaport et al., 2007], [Li and Li, 2008] et [Zhu et al., 2009] utilisent pour l'intégration dans la classification des réseaux issus de bases de données publiques généralistes ou spécialistes des cancers. Or les informations présentes dans ces bases de données sont relativement parcellaires, et même si l'on fait l'hypothèse qu'elles sont complètes, elles sont très souvent de portée trop générale pour que les graphes extraits soient réellement en adéquation avec un jeu de données particulier.

Nous avons donc décidé de nous appuyer sur les méthodes d'inférence de graphe que nous verrons dans la partie suivante, appliquées à des jeux de données indépendants des jeux de données étudiés en classification mais en rapport avec le phénomène biologique étudié.

Par manque de temps, nous n'avons pu implémenter la méthode de Binder *et al.* [Binder and Schumacher, 2009], qui permet d'intégrer une information de réseau dans un algorithme de *boosting*.

Nous proposons également de montrer dans la suite du travail que les résultats de gCDA dépendent beaucoup de la qualité du graphe à intégrer et du modèle qui permet de déterminer la matrice de variance covariance à partir du graphe. Il est donc d'autant plus important de s'assurer que le graphe donné est en adéquation avec les données à classer.

Enfin, nous remarquons que si l'on considère les méthodes de Rapaport *et al.* et de Li et Li, le Laplacien $L_{\mathcal{G}}$ est respectivement homogène à la matrice de précision et à la matrice de variance covariance des données :

- Pour la méthode de Rapaport *et al.*, la distance entre deux profils d'expression f et g dans le nouvel espace est définie ainsi :

$$d(f,g) = f^\top K_\phi g.$$

avec K_ϕ le noyau utilisé. Lorsque ϕ est l'opérateur de seuillage, K_ϕ, vu comme un noyau de Mahalanobis (voir par exemple [Haasdonk and P ekalska, 2008]), est homogène à une matrice de précision. Par suite, le Laplacien du graphe, après seuillage de ses valeurs propres par la fonction ϕ, est considéré homogène à la matrice de précision des données.

- Pour la méthode de Li et Li, si l'on s'abstrait de la contrainte de parcimonie, le critère à optimiser est le suivant :

$$J(\beta,\lambda) = \|y - \boldsymbol{x}\beta\|^2 + \lambda\beta^\top L_{\mathcal{G}}\beta.$$

Comme $L_{\mathcal{G}}$ est semi définie positive, J est un critère convexe qui admet une unique solution dont l'expression s'obtient en résolvant l'équation suivante :

$$\begin{aligned}
&\frac{\partial J}{\partial \beta} = 2\boldsymbol{x}^\top y - 2\boldsymbol{x}^\top \boldsymbol{x}\beta + 2\lambda L_{\mathcal{G}}\beta = 0 \\
\Leftrightarrow &\left(\boldsymbol{x}^\top \boldsymbol{x} - \lambda L_{\mathcal{G}}\right)\beta = \boldsymbol{x}^\top y \\
\Rightarrow &\beta = \left(\boldsymbol{x}^\top \boldsymbol{x} - \lambda L_{\mathcal{G}}\right)^{-1} \boldsymbol{x}^\top y
\end{aligned}$$

Cette expression est à comparer à l'expression des vrais coefficients :

$$\beta = \mathrm{var}(\boldsymbol{X})^{-1}\,\mathrm{cov}(\boldsymbol{X}, Y).$$

Ainsi, résoudre le problème d'optimisation proposé par Li et Li nécessite de considérer que $L_{\mathcal{G}}$ est homogène à la matrice de variance covariance des données.

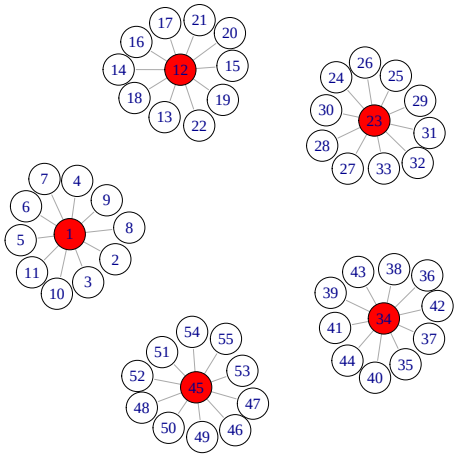
La manière de considérer le Laplacien dans ces travaux peut être étendue

- en intégrant $L_{\mathcal{G}}$ non plus sous une forme seuillée dans un noyau de Mahalanobis, mais sous la forme que nous avons proposée dans la méthode gCDA :

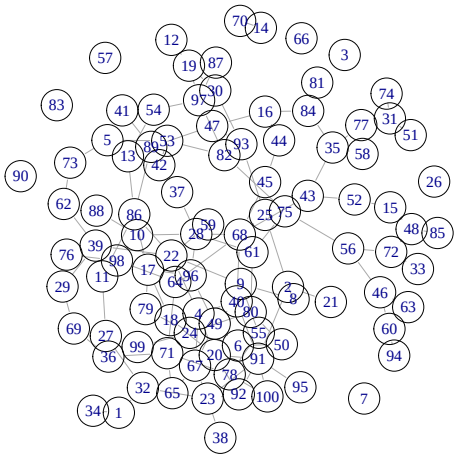
$$K = \left(\alpha S + (1-\alpha)(L+I)^{-1}\right)^{-1},$$

- ou en l'intégrant dans une RR de sorte à ce qu'il soit de nouveau homogène à une matrice de précision, ce qui mènerait par exemple à l'estimation des poids du modèle de régression suivante :

$$\widehat{\beta} = \left(\boldsymbol{x}^\top \boldsymbol{x} + \lambda\Omega^{-1}\right)\boldsymbol{x}^\top y \text{ avec } \Omega = L + I.$$



(a) Graphe correspondant aux simulations de [Li and Li, 2008].



(b) Graphe correspondant au modèle de simulation proposé.

FIGURE 2.5 – Exemples de graphes caractéristiques des modèles de simulation utilisés. La figure du haut représente un graphe caractéristique des simulations présentées par Li *et al.*, les facteurs de transcription sont colorés en rouge. La figure du bas représente un graphe généré aléatoirement avec les méthodes de la librairie `igraph`.

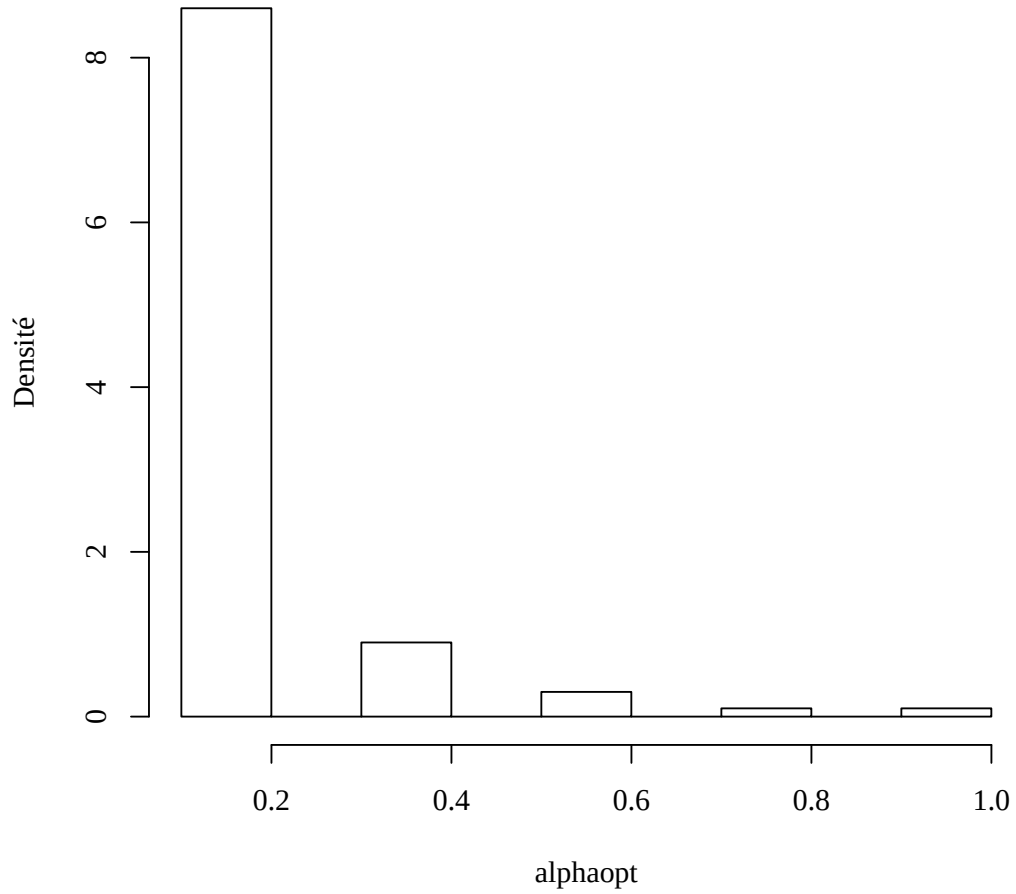


FIGURE 2.6 – Histogramme des valeurs optimales du paramètre α sélectionnée par k-fold à chaque itération de la validation croisée par MCCV.

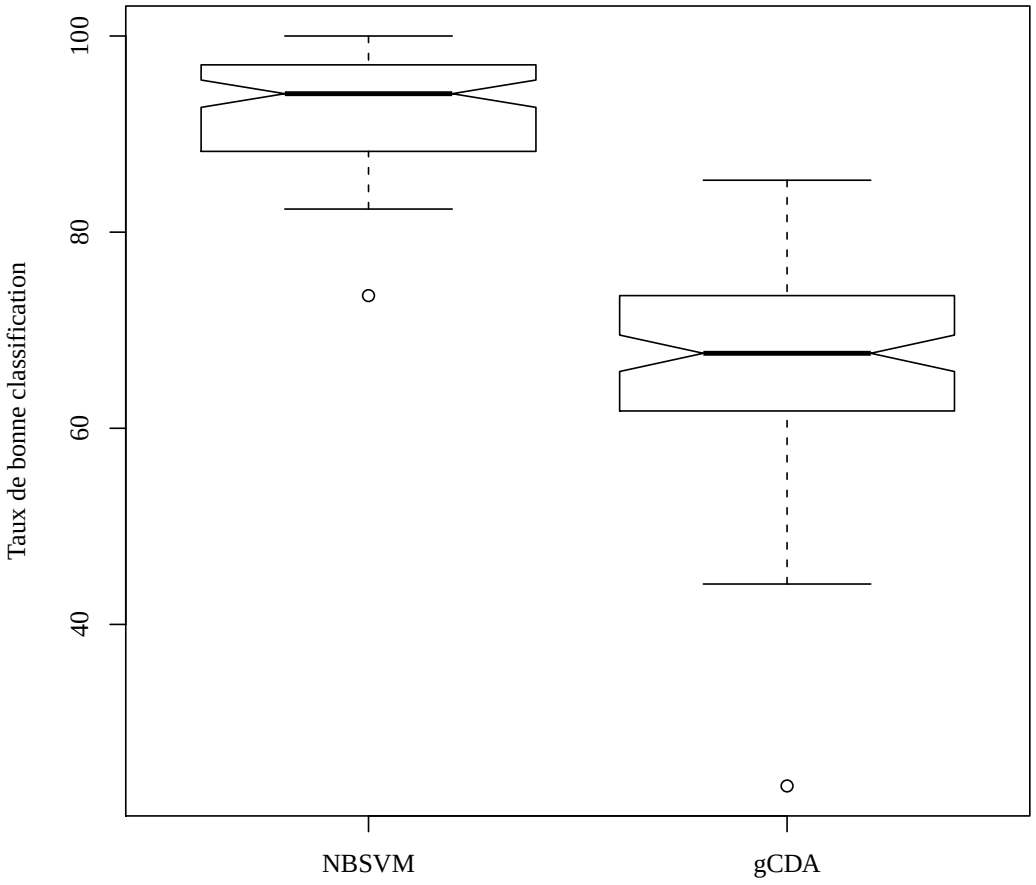


FIGURE 2.7 – Résultats complémentaires portant sur le modèle de simulation proposé dans [Li and Li, 2008].

Chapitre 3

Inférence de réseaux de régulations génétiques et adéquation de réseaux à des données transcriptomiques

Nous avons vu dans la partie précédente qu'il est capital de contrôler la qualité du graphe à intégrer dans le processus de classification. Comme les graphes issus des bases de données publiques ne nous permettent pas de le faire, nous nous tournons vers des méthodes statistiques permettant de déterminer un RRG à partir de données microarray. La figure 3.1 illustre les points qui seront abordés dans ce chapitre : l'inférence de RRG, mais aussi l'adéquation d'un réseau à des données.

Dans toute cette partie, nous nous plaçons dans le cadre des modèles graphiques gaussiens (*Graphical Gaussian Models* ou GGM) qui permettent d'associer un graphe d'indépendances conditionnelles à des profils d'expression. Plus précisément, la matrice de corrélations partielles Π a des coefficients $[\Pi]_{i,j} = \pi_{i,j}$ nuls lorsque les nœuds i et j ne sont pas reliés dans le graphe d'indépendances conditionnelles sous jacent $\mathcal{G}$ [Whittaker, 1990]. Inférer un graphe reviendra donc, dans notre cas, à calculer une matrice de corrélations partielles empirique et à identifier les coefficients significativement non nuls. Nous proposons un estimateur des coefficients de corrélation partielles basé sur des PLS-R successives [Tenenhaus et al., 2008] et en déduisons un graphe grâce à la procédure de sélection des coefficients π_{ij} significatifs décrite dans [Schäfer and Strimmer, 2
Cette procédure d'inférence est similaire à d'autres méthodes présentées dans la littérature comme la méthode *glasso* [Tibshirani, 1996] ou encore la méthode présentée par Krämer *et al.* [Krämer et al., 2009] basée sur des RR successives.

Comme nous manquons d'une méthode permettant de sélectionner parmi plusieurs graphes celui qui est le plus en adéquation avec des données simulées, nous proposons également dans cette partie un indicateur basé sur une statistique de sphéricité permettant de choisir parmi plusieurs graphes [Guillemot et al., 2009].

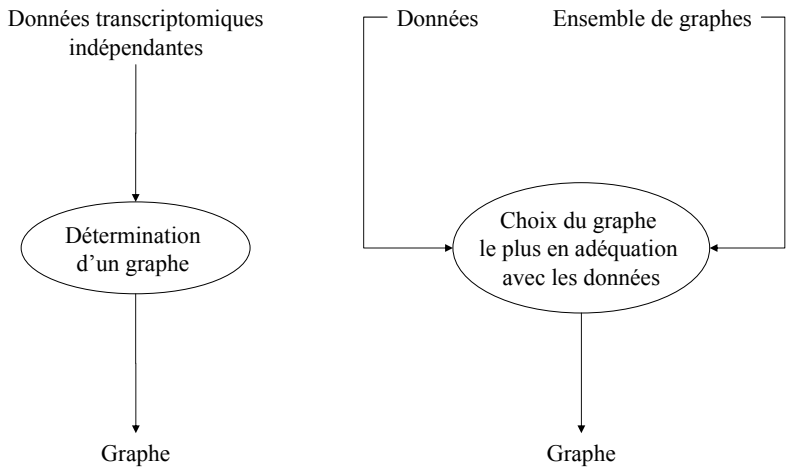


FIGURE 3.1 – Les points abordés dans ce chapitre sur les modèles graphiques gaussiens de façon schématique.

3.1 Coefficient de corrélation partielle

Nous montrons dans cette partie trois écritures équivalentes du coefficient de corrélation partielle :

- sa définition,
- une écriture utilisant des coefficients de régression linéaire.
- une écriture utilisant les coefficients de la matrice de précision,

L'intérêt d'introduire ces écritures est de montrer que dans le cas où $n \ll p$, il est possible de calculer des coefficients de corrélation partielle en ayant recours à la régularisation soit de l'estimation de la matrice de variance covariance empirique, avant de l'inverser, soit de l'estimation de coefficients de régression linéaire, par exemple avec les méthodes de régression régularisée RR, LASSO ou PLS-R vues précédemment. Les résultats présentés dans ce paragraphe sont extraits de [Whittaker, 1990].

Soit X, Y et $\boldsymbol{Z} = (Z_1, ..., Z_q)$ des variables aléatoires ayant un moment d'ordre 2 (avec les notations précédentes : $q + 2 = p$ le nombre total de gènes). On cherche à prédire X ou Y en fonction des variables présentes dans le vecteur $\boldsymbol{Z}$.

Il est à noter que dans cette partie, l'hypothèse que les données sont gaussiennes n'est pas nécessaire.

Définition 4

Le coefficient de corrélation partielle entre X et Y sachant $\boldsymbol{Z}$ mesure la corrélation entre les résidus des régressions OLS de X et Y sur $\boldsymbol{Z}$:

$$\text{cor}(X, Y|\boldsymbol{Z}) = \frac{\text{cov}(X - \widehat{X}(\boldsymbol{Z}), Y - \widehat{Y}(\boldsymbol{Z}))}{\sqrt{\text{var}(X - \widehat{X}(\boldsymbol{Z}))\,\text{var}(Y - \widehat{Y}(\boldsymbol{Z}))}}.$$

Ce coefficient permettra, lorsque les données sont supposées gaussiennes, de mesurer l'indépendance entre X et Y conditionnellement à $\boldsymbol{Z}$ et de faire correspondre à un jeu de données un graphe d'indépendances conditionnelles. Une arête est tracée dans ce graphe dès que le coefficient de corrélation partielle correspondant est significativement différent de 0.

3.1.1 Expression du coefficient de corrélation partielle à l'aide de régressions OLS

Rappelons la définition du prédicteur OLS déjà utilisé dans la partie 1.1.1 :

Définition 5 (*Régression OLS*)

Le prédicteur aux moindres carrés ordinaires de X en fonction de $\boldsymbol{Z}$ est la variable aléatoire $\gamma(\boldsymbol{Z})$, combinaison linéaire des variables $Z_i, i = 1, ..., q$, minimisant le critère suivant

$$E\left((\gamma(\boldsymbol{Z}) - X)^2\right),$$

avec E l'espérance. Il est noté $\widehat{X}(\boldsymbol{Z})$.

Un tel prédicteur $\widehat{X}(\boldsymbol{Z})$ possède les propriétés suivantes :

Proposition 3

$$\begin{aligned}
\text{cov}(\widehat{X}(\boldsymbol{Z}), \boldsymbol{Z}) &= 0 && (3.1)\\
\widehat{X}(\boldsymbol{Z}) &= \boldsymbol{Z}\,\text{var}(\boldsymbol{Z})^{-1}\,\text{cov}(\boldsymbol{Z}, X) && (3.2)\\
A \in GL_q(\mathbb{R}) &\Rightarrow \widehat{X}(A\boldsymbol{Z}) = \widehat{X}(\boldsymbol{Z}) && (3.3)\\
\text{cov}(U, V) = 0 &\Rightarrow \widehat{Y}(U, V) = \widehat{Y}(U) + \widehat{Y}(V), && (3.4)
\end{aligned}$$

avec $GL_q(\mathbb{R})$ l'ensemble (le groupe pour la multiplication) des matrices carrées inversibles à coefficients réels.

Il est possible de calculer un coefficient de corrélation partielle en utilisant les coefficients de régressions OLS. Pour montrer cette propriété, il faut tout d'abord énoncer le théorème suivant :

Théorème 3

$$\begin{aligned} \operatorname{cov}(\boldsymbol{Z}, Y - \widehat{Y}(\boldsymbol{Z})) &= 0, && (3.5)\\ \widehat{X}(\boldsymbol{Z}, Y) &= \widehat{X}(\boldsymbol{Z}) + \widehat{X}(Y - \widehat{Y}(\boldsymbol{Z})). && (3.6) \end{aligned}$$

Pour démontrer (3.5), on commence par utiliser la bilinéarité de la covariance :

$$\operatorname{cov}(\boldsymbol{Z}, Y - \widehat{Y}(\boldsymbol{Z})) = \operatorname{cov}(\boldsymbol{Z}, Y) - \operatorname{cov}(\boldsymbol{Z}, \widehat{Y}(\boldsymbol{Z})),$$

Or, d'après la propriété (3.2),

$$\widehat{Y}(\boldsymbol{Z}) = \boldsymbol{Z} \operatorname{var}(\boldsymbol{Z})^{-1} \operatorname{cov}(\boldsymbol{Z}, Y).$$

Nécessairement,

$$\begin{aligned} \operatorname{cov}(\boldsymbol{Z}, Y - \widehat{Y}(\boldsymbol{Z})) &= \operatorname{cov}(\boldsymbol{Z}, Y) - \operatorname{cov}\left(\boldsymbol{Z}, \boldsymbol{Z} \operatorname{var}(\boldsymbol{Z})^{-1} \operatorname{cov}(\boldsymbol{Z}, Y)\right) \\ &= \operatorname{cov}(\boldsymbol{Z}, Y) - \operatorname{cov}(\boldsymbol{Z}, \boldsymbol{Z}) \operatorname{var}(\boldsymbol{Z})^{-1} \operatorname{cov}(\boldsymbol{Z}, Y). \end{aligned}$$

En simplifiant, on obtient bien la propriété (3.5).

Pour démontrer (3.6), nous avons besoin de l'opérateur d'orthogonalisation suivant :

$$L : (U, V) \mapsto L(U, V) = (U, V - \widehat{V}(U)).$$

C'est une fonction linéaire et, de plus, inversible, comme on le voit si on explicite sa matrice associée (que l'on notera L également) :

$$L = \begin{bmatrix} I & 0 \\ -\operatorname{cov}(V, U) \operatorname{var}(U)^{-1} & I \end{bmatrix}.$$

D'une part, comme L est inversible et grâce à la propriété (3.3), $\widehat{X}(L(\boldsymbol{Z}, Y)) = \widehat{X}(\boldsymbol{Z}, Y)$. Et d'autre part, $\widehat{X}(L(\boldsymbol{Z}, Y)) = \widehat{X}(\boldsymbol{Z}, Y - \widehat{Y}(\boldsymbol{Z}))$ par définition de L. Or, la propriété (3.5) indique que la covariance entre $\boldsymbol{Z}$ et $Y - \widehat{Y}(\boldsymbol{Z})$ est nulle, on retrouve donc l'égalité recherchée en appliquant la propriété (3.4) :

$$\widehat{X}(\boldsymbol{Z}, Y) \overset{(3.3)}{=} \widehat{X}(L(\boldsymbol{Z}, Y)) = \widehat{X}(\boldsymbol{Z}, Y - \widehat{Y}(\boldsymbol{Z})) \overset{(3.5)+(3.4)}{=} \widehat{X}(\boldsymbol{Z}) + \widehat{X}(Y - \widehat{Y}(\boldsymbol{Z})).$$

Théorème 4

Le coefficient de corrélation partielle peut se calculer à l'aide de coefficients de régressions OLS. Si on pose

$$\begin{aligned} \widehat{X}(Y, \boldsymbol{Z}) &= \beta_1 Y + \sum_{j=1}^{q} \alpha_{X,j} Z_j \\ \widehat{Y}(X, \boldsymbol{Z}) &= \beta_2 X + \sum_{j=1}^{q} \alpha_{Y,j} Z_j \end{aligned} \qquad (3.7)$$

alors

$$\text{signe}(\beta_1) = \text{signe}(\beta_2) \text{ et } \text{cor}(X, Y|\boldsymbol{Z}) = \text{signe}(\beta_1)\sqrt{\beta_1\beta_2}.$$

D'après la propriété (3.2) :

$$\widehat{X}(Y - \widehat{Y}(\boldsymbol{Z})) = (Y - \widehat{Y}(\boldsymbol{Z}))\,\text{var}(Y - \widehat{Y}(\boldsymbol{Z}))^{-1}\,\text{cov}(Y - \widehat{Y}(\boldsymbol{Z}), X).$$

Or la propriété (3.5), alliée au fait que $\widehat{X}(\boldsymbol{Z})$ est une combinaison linéaire des composantes de $\boldsymbol{Z}$, nous donne $\text{cov}(\widehat{X}(\boldsymbol{Z}), Y - \widehat{Y}(\boldsymbol{Z})) = 0$, donc

$$\text{cov}(Y - \widehat{Y}(\boldsymbol{Z}), X) = \text{cov}(Y - \widehat{Y}(\boldsymbol{Z}), X - \widehat{X}(\boldsymbol{Z}))$$

et on peut écrire :

$$\widehat{X}(Y - \widehat{Y}(\boldsymbol{Z})) = (Y - \widehat{Y}(\boldsymbol{Z}))\,\text{var}(Y - \widehat{Y}(\boldsymbol{Z}))^{-1}\,\text{cov}(Y - \widehat{Y}(\boldsymbol{Z}), X - \widehat{X}(\boldsymbol{Z})).$$

On peut donc exprimer β_1, qui est en fait le coefficient en facteur de Y :

$$\beta_1 = \text{var}(Y - \widehat{Y}(\boldsymbol{Z}))^{-1}\,\text{cov}(Y - \widehat{Y}(\boldsymbol{Z}), X - \widehat{X}(\boldsymbol{Z})).$$

On remarque que $\text{var}(Y - \widehat{Y}(\boldsymbol{Z}))$, $X - \widehat{X}(\boldsymbol{Z})$ et $Y - \widehat{Y}(\boldsymbol{Z})$ sont des scalaires. Le même calcul est fait pour $\widehat{Y}(X, \boldsymbol{Z})$ et donne

$$\beta_2 = \text{var}(X - \widehat{X}(\boldsymbol{Z}))^{-1}\,\text{cov}(X - \widehat{X}(\boldsymbol{Z}), Y - \widehat{Y}(\boldsymbol{Z})).$$

Ainsi, par symétrie de la covariance (entre deux variables scalaires), $\text{signe}(\beta_1) = \text{signe}(\beta_2)$, ce signe étant de même égal au signe de la covariance entre les deux résidus. De plus :

$$\beta_1\beta_2 = \frac{\text{cov}(X - \widehat{X}(\boldsymbol{Z}), Y - \widehat{Y}(\boldsymbol{Z}))^2}{\text{var}(X - \widehat{X}(\boldsymbol{Z}))\,\text{var}(Y - \widehat{Y}(\boldsymbol{Z}))}.$$

Finalement :

$$\text{cor}(X, Y|\boldsymbol{Z}) = \text{signe}(\beta_1)\sqrt{\beta_1\beta_2}$$

3.1.2 Expression de la matrice de corrélation partielle à l'aide de la matrice de variance covariance

Le coefficient de corrélation partielle s'exprime ainsi en fonction de coefficients de régression OLS. Rappelons que X et Y sont de « dimension » 1 et Z est de « dimension » $q = p - 2$. Il peut également s'exprimer en fonction des coefficients de la matrice inverse de la matrice de variance covariance, également appelée la matrice de précision.

Proposition 4 (*Lemme d'inversion de la variance*)

Notons l'inverse de la matrice de variance covariance $D = var(X, Y, Z)^{-1}$, avec la

décomposition en blocs suivante :

$$D = \begin{pmatrix} D_{\{X,Y\},\{X,Y\}} & D_{\{X,Y\},Z} \\ D_{Z,\{X,Y\}} & D_{Z,Z} \end{pmatrix}.$$

(i) La matrice de variance covariance admet la décomposition de Cholesky par blocs suivante :

$$\begin{aligned} \operatorname{var}(\{X,Y\},Z) &= \begin{pmatrix} I & 0 \\ \operatorname{var}(Z)^{-1}\operatorname{cov}(Z,\{X,Y\}) & I \end{pmatrix} \\ &\quad \begin{pmatrix} \operatorname{var}(\{X,Y\}|Z) & 0 \\ 0 & \operatorname{var}(Z,Z) \end{pmatrix} \begin{pmatrix} I & \operatorname{cov}(\{X,Y\},Z)\operatorname{var}(Z)^{-1} \\ 0 & I \end{pmatrix} \end{aligned}$$

(ii) la décomposition précédente permet de montrer la propriété suivante pour la matrice de variance covariance inverse :

$$\operatorname{var}(\{X,Y\},Z)^{-1} = \begin{pmatrix} \operatorname{var}(\{X,Y\}|Z)^{-1} & * \\ * & * \end{pmatrix}$$

La preuve de ces deux propriétés utilise l'opérateur d'orthogonalisation que nous avons vu précédemment. Elles permettent de montrer le théorème suivant, explicitement démontré dans [Whittaker, 1990].

Théorème 5

Le coefficient non diagonal normalisé en position (i,j) de la matrice de précision est l'opposé du coefficient de corrélation partielle des deux variables correspondant aux numéros i et j conditionnellement à toutes les autres variables.

On note tout d'abord

$$D_{\{X,Y\},\{X,Y\}} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

La propriété d'inversion de la matrice de variance covariance nous donne

$$D_{\{X,Y\},\{X,Y\}} = \operatorname{var}(\{X,Y\}|Z)^{-1} = \begin{pmatrix} \operatorname{var}(X|Z) & \operatorname{cov}(X,Y|Z) \\ \operatorname{cov}(Y,X|Z) & \operatorname{var}(Y|Z) \end{pmatrix}^{-1}.$$

En « normalisant » les deux matrices de sorte à avoir des diagonales de 1, on obtient :

$$\operatorname{cor}(X,Y|Z) = -\frac{b}{\sqrt{ad}}.$$

Le théorème 5 permet ainsi faire le lien entre les coefficients de corrélation partielle et les coefficients de la matrice de variance-covariance.

3.1.3 Résumé

Numérotons maintenant les variables de 1 à p : $\boldsymbol{X} = (X_1, ..., X_p)$. Soit i et j deux entiers différents compris entre 1 et p. On pose de plus

$$\widehat{X}_i(X_j, X_{k\neq i,j}) = \beta_{i,j}X_j + \sum_{k\neq j}\beta_{k,j}X_k \tag{3.8}$$

$$\widehat{X}_j(X_i, X_{k\neq i,j}) = \beta_{j,i}X_i + \sum_{k\neq i}\beta_{k,i}X_k \tag{3.9}$$

Le coefficient de corrélation partielle entre deux variables X_i et X_j sachant un ensemble d'autres variables $X_k, k \in \{1, ..., p\} \setminus i, j$ peut s'écrire de deux façons [Whittaker, 1990] :

- la première basée sur des coefficients de régressions linéaires $X_i = \beta_{i,j}X_j + \sum_{k\neq i,j}\beta_{i,k}X_k$ et $X_j = \beta_{j,i}X_i + \sum_{k\neq i,j}\beta_{j,k}X_k$

$$\text{cor}(X_i, X_j|X_{k\neq i,j}) = \text{signe}(\beta_{i,j})\sqrt{\beta_{i,j}\beta_{j,i}}. \tag{3.10}$$

- la deuxième sur des coefficients de la matrice de précision $\Omega = \Sigma^{-1} = [\omega_{i,j}]$,

$$\text{cor}(X_i, X_j|X_{k\neq i,j}) = -\frac{\omega_{i,j}}{\sqrt{\omega_{i,i}\omega_{j,j}}}. \tag{3.11}$$

Nous adoptons de plus la notation

$$\pi_{ij} = \text{cor}(X_i, X_j|X_{k\neq i,j}).$$

Dans le cas où $n < p$, il n'est plus possible de calculer la corrélation partielle avec des estimateurs classiques des coefficients de régression linéaire aux moindres carrés (par exemple $\widehat{\beta}^{ols}$) ou de la matrice de variance-covariance. Des estimateurs régularisés peuvent être utilisés. Pour des estimateurs régularisés de la matrices de variance covariance, voir par exemple [Schäfer and Strimmer, 2005b], [Schäfer and Strimmer, 2005a]. Pour des estimateurs de corrélations partielles à base de régression régularisées, voir par exemple [Krämer et al., 2009], [Friedman et al., 2008], [Meinshausen and Bühlman, 2006], [Tenenhaus et al., 2008]. Nous nous intéressons dans la suite à un estimateur régularisé de la corrélation partielle utilisant des coefficients de PLS-R.

3.2 Estimation de coefficients de corrélation partielle lorsque $n \leq p$

Dans [Tenenhaus et al., 2008], nous présentons un algorithme d'inférence de graphes d'indépendances conditionnelles à partir de données transcriptomiques en utilisant la PLS-R. Le graphe est inféré pas en identifiant des coefficients significativement nuls dans la matrice de corrélations partielles notée $\Pi = \left[\pi_{i,j}\right]_{i,j}$:

$$\pi_{ij} = \text{cor}(X_i, X_j|X_k, k \neq i, j)$$

Pour estimer ces coefficients de corrélation partielle, nous utilisons l'équivalence (3.10) qui exploite les coefficients de régression linéaire :

$$\pi_{ij} = \text{signe}\,\beta_{ij}\sqrt{\beta_{ij}\beta_{ji}},$$

avec les coefficients β_{ij} et β_{ji} définis par les équations (3.8) et (3.9). Utiliser la PLS-R pour estimer ces coefficients β_{ij}, notés alors $\widehat{\beta}^{pls}_{ij}$, permet de régulariser l'estimation des coefficients de corrélation partielle. Le cadre $n < p$ provoque parfois des incohérences de signes entre $\widehat{\beta}^{pls}_{ij}$ et $\widehat{\beta}^{pls}_{ji}$, qui, d'après le théorème 4, devraient être égaux. En effet, il existe des indices i_0, j_0 tels que

$$\text{signe}\,\widehat{\beta}^{pls}_{i_0 j_0} \neq \text{signe}\,\widehat{\beta}^{pls}_{j_0 i_0}.$$

La stratégie choisie dans ce cas est de considérer que, les deux régressions effectuées menant à des résultats contradictoires, le coefficient doit être annulé. L'estimateur des coefficients de corrélation partielle utilisant la PLS-R est le suivant :

$$\widehat{\pi}^{pls}_{ij} = \begin{cases} \text{signe}\,\widehat{\beta}^{pls}_{ij}\sqrt{\widehat{\beta}^{pls}_{ij}\widehat{\beta}^{pls}_{ji}} & \text{si } \widehat{\beta}^{pls}_{ij}\widehat{\beta}^{pls}_{ji} > 0 \\ 0 & \text{sinon} \end{cases}.$$

Cet estimateur sera noté dans la suite PLS-PC.

Utiliser un estimateur régularisé pour la corrélation partielle signifie également introduire un paramètre de régularisation, Krämer *et al.* [Krämer et al., 2009] proposent de déterminer pour chaque régression ce coefficient de régularisation par validation croisée. Nous avons adopté une démarche différente, en imposant la même valeur valeur à tous les paramètres de régularisation des coefficients d'une matrice de corrélations partielles. Cela permet d'harmoniser la régularisation pour tous les $\widehat{\pi}^{pls}_{ij}$ que nous pouvons donc noter $\widehat{\pi}^{pls}_{ij}(h)$, avec h le nombre de composantes PLS retenues pour l'estimation. Cependant, il n'est plus possible d'utiliser des méthodes de validation croisée classiques comme le font Krämer *et al.* pour déterminer la valeur de ce nombre de composantes.

Nous avons imaginé une méthode empirique permettant de déterminer le paramètre h en fonction de la convergence des coefficients $\widehat{\beta}^{pls}_{ij}(h)$ vers une valeur fixe. On peut observer sur la figure 3.2 que tous les coefficients calculés atteignent un plateau pour une certaine valeur de h. Cette valeur est déterminée sur un ensemble de coefficients $\widehat{\beta}^{pls}_{ij}$ qui prennent des valeurs significativement non nulles.

Une fois calculée $\Pi^{pls} = \left[\widehat{\pi}^{pls}_{ij}\right]$, il faut, pour la représenter sous forme d'un graphe $\mathcal{G}$, déterminer quels sont les coefficients significativement non nuls à l'aide d'un test d'hypothèse. Or, dans une matrice de corrélation partielle de taille $p \times p$, il y a $p(p-1)/2$ tests à effectuer. Comme p peut être de l'ordre de 100 (par exemple 100 gènes

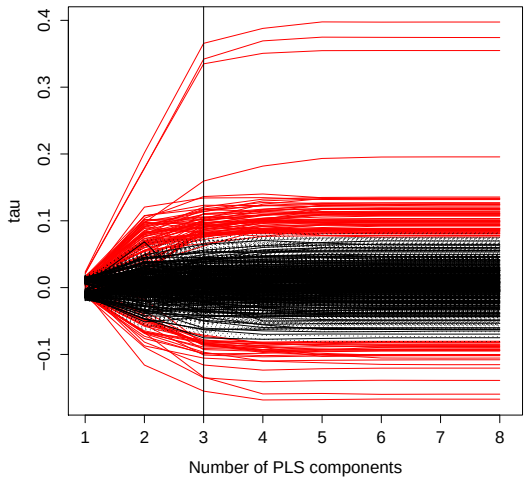


FIGURE 3.2 – Coefficients $\widehat{\beta}_{ij}^{pls}$ en fonction de h.

différentiellement exprimés), il faut donc corriger les p-values obtenues avec une stratégie de correction des tests multiples avant de conclure qu'un coefficient est significativement non nul. Comme indiqué précédemment, nous avons choisi la même correction pour les tests multiples que celle adoptée par [Schäfer and Strimmer, 2005a]. Nous envisageons cette question en testant l'hypothèse nulle suivante :

$$\mathcal{H}_0 : \pi_{i,j} = 0 \text{ versus } \mathcal{H}_1 : \pi_{i,j} \neq 0$$

On suppose que les p-values obtenues sont des réalisations d'une variable aléatoire dont la densité est le mélange suivant :

$$f(\tau) = p_0 f_0(\tau) + p_1 f_1(\tau) \tag{3.12}$$

avec p_0 et p_1 les *a priori* respectivement concernant la distribution nulle et la distribution alternative, f_0 et f_1, respectivement, avec $p_0 + p_1 = 1$ et $p_1 \ll p_0$. La probabilité *a posteriori* qu'une p-value soit significativement inférieure au seuil α donné sachant la valeur de la p-value (c'est la définition du taux de faux positif local (*local* FDR)) se calcule classiquement suivant la formule de Bayes :

$$P(\overline{pval < \alpha} | pval = v) \equiv fdr(v) = \frac{p_0 f_0(v)}{f(v)} \tag{3.13}$$

Le *local* FDR est basé sur l'estimation de f et $p_0 f_0$. Cette estimation se déroule suivant les étapes présentées ci-dessous :

(i) Estimer le mélange des densités $f(v)$ en ajustant un histogramme sur toutes les *p-values* calculées. Pour estimer les composantes d'un mélange, voir par exemple [Efron, 2005a].

(ii) Estimer $p_0 f_0$ à partir de l'histogramme autour de la valeur $v = 0$ de façon non paramétrique. On peut à ce niveau injecter un *a priori* sur la distribution des coefficients de corrélation partielle sous l'hypothèse nulle.

(iii) Conclure que le $\widehat{\pi}_{ij}^{pls}$ est significativement non nul si la probabilité *a posteriori* $\hat{p}_0 \hat{f}_0(v) / \hat{f}(v)$, est plus petite qu'un seuil prédéfini (par exemple 0.8 comme suggéré par [Efron, 2005b]).

Pour plus de détails sur la procédure du *local* FDR, voir [Efron, 2005a] et [Schäfer and Strimmer, 2005a].

Il faut souligner qu'un élément clef de cette procédure est que les réseaux d'interactions biologiques sont supposés posséder peu d'arêtes (on dira aussi que leur structure est parcimonieuse). Ainsi, plus p est grand, plus le nombre d'arêtes rejetées est grand, et plus il est facile d'estimer $p_0 f_0$.

Un résultat intéressant que nous avons obtenu avec la méthode PLS-PC est l'obtention, sur des données réelles d'expression chez *E. coli*, d'un graphe de régulations génétiques qui contient des motifs aisément reconnaissables par les biologistes (par exemple l'opéron lactose). Ce graphe est représenté sur la figure 3.3.

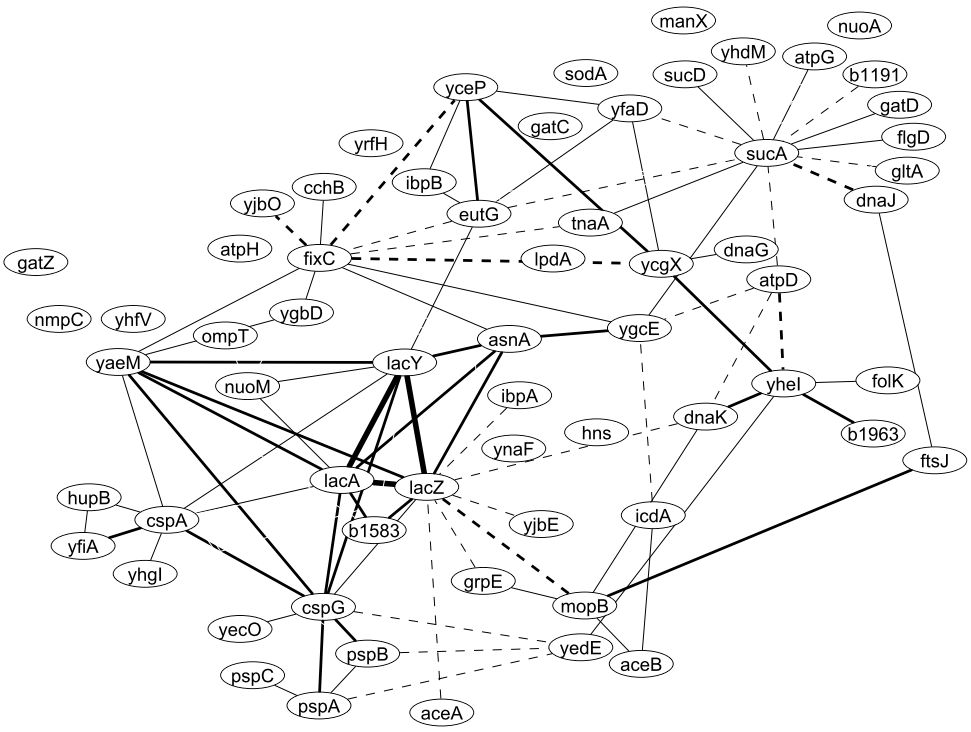


FIGURE 3.3 – Graphe obtenu sur des données réelles, voir détails dans [Tenenhaus et al., 2008].

Un exemple sur des données simulées avec la méthode proposée dans le paragraphe 2.2.4.2 est présenté sur la figure 3.4 :

- un échantillon gaussien multivarié a été généré, respectant la structure d'indépendances conditionnelles représentée par le graphe 3.4(a),
- deux matrices de corrélations partielles ont été inférés avec les méthode basées sur la RR d'une part et la méthode PLS-PC d'autres part,
- pour chacune de ces matrices, le graphe optimal en termes de spécificité et de sensibilité est déterminé et représenté respectivement figure 3.4(b) pour la méthode basée sur la RR et figure 3.4(c) pour la méthode PLS-PC. Le caractère optimal d'un graphe en fonction de la spécificité spe et de la sensibilité sen de la prédiction des arêtes est le graphe qui minimise la quantité

$$(1 - sen)^2 + (1 - spe)^2.$$

Les deux graphes inférés ont des allures très différentes, ils contiennent notamment bien plus d'arêtes que le vrai graphe. Cela est dû à l'utilisation de la spécificté comme indicateur de qualité d'un graphe. Ce point est discuté dans la partie 3.4.

Ainsi, des méthodes basées sur le même principe, l'inférence d'une matrice de corrélations partielles grâce à l'équation (3.10), donnent des graphes très différents. Il n'y a de plus pas de consignes claires sur le seuil à appliquer au *local* FDR. Enfin, d'autres méthodes d'inférence de RRG existent, basées sur l'équivalence (3.11) [Schäfer and Strimmer, 2005b] ou sur l'inférence d'information mutuelle [Margolin et al., 2006]. Toutes ces remarques soulignent le besoin qu'ont les bioinformaticiens d'un indicateur permettant de déterminer dans une famille de graphes lequel est le plus en adéquation avec un jeu de données. L'indicateur que nous présentons est inspirée de la théorie des tests de sphéricité.

3.3 Mesure de l'adéquation d'un graphe à un jeu de données

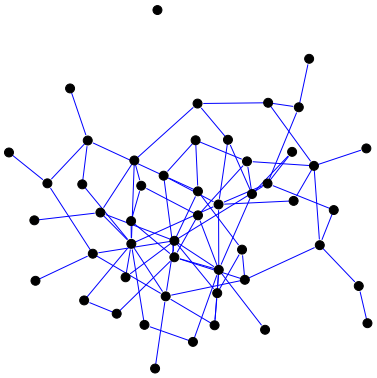
Nous proposons dans cette partie un indicateur simple d'utilisation permettant de montrer qu'un graphe est en adéquation avec un jeu de données. Cet indicateur permet pour une méthode d'inférence donnée, de déterminer les paramètres de régularisation optimaux, mais aussi de choisir la meilleure méthode d'inférence.

Test de sphéricité

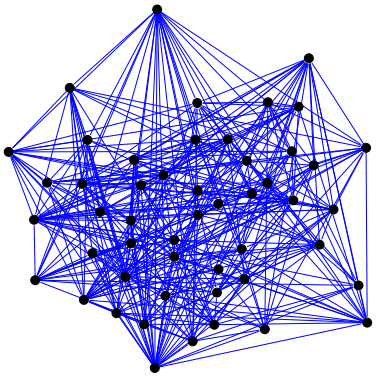
Un test de sphéricité consiste à vérifier que la matrice de variance-covariance d'un échantillon correspond de façon significative à la matrice identité. L'hypothèse nulle est la suivante :

$$\mathcal{H}_0 : \Sigma = I_p \text{ ou bien } \Sigma = \sigma^2 I_p.$$

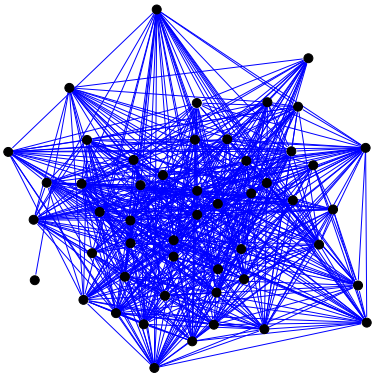
On trouve des descriptions de tels tests chez [Dagnelie, 1999] (lui même citant le travail très complet de [Nagao, 1973]) dans le cas où le nombre de variables est inférieur à la taille de l'échantillon. La plupart de ces tests sont basés sur le critère du rapport de



(a) Vrai graphe.



(b) Méthode Ridge.



(c) Méthode PLS.

FIGURE 3.4 – Comparaison des résultats obtenus par deux méthodes d'inférence de graphe : la méthode basée sur la RR (en bas à gauche) et la méthode PLS-PC (en bas). Les données ont été simulées à partir du graphe représenté en haut. Nous remarquons, et c'est une tendance générale que nous avons rencontrée tout au long de nos études sur des données simulées, que PLS-PC tend à produire des graphes contenant bien plus d'arêtes qu'ils ne devraient en contenir.

vraisemblance [Anderson, 2003]. Cependant, d'après Ledoit et Wolf [Ledoit and Wolf, 2002],

le test du rapport de vraisemblance est « dégénéré » dans le cas où $n < p$.

Une autre statistique est alors proposée par Ledoit et Wolf :

$$W = \frac{1}{p}\operatorname{tr}([S - I_p]^2) - \frac{p}{n}\left(\frac{1}{p}\operatorname{tr}(S)\right)^2 + \frac{p}{n},$$

avec S l'estimateur non biaisé de la matrice de variance covariance de l'échantillon considéré, $S = 1/(n-1)\sum_{i=1}^{n}(x_i - \overline{x})^\top(x_i - \overline{x})$.

Ce test est dédié à l'hypothèse nulle $H_0 : \Sigma = I_p$, pour le rendre valable dans le cas général, c'est-à-dire pour $\mathcal{H}_0 : \Sigma = \Sigma_0$ avec Σ_0 une matrice définie positive quelconque, nous appliquons la transformation préconisée par [Anderson, 2003] qui consiste à remplacer S par $S\Sigma_0^{-1}$.

Indicateur proposé

Nous nous plaçons ici dans le cas très particulier où le modèle de passage entre $\mathcal{G}$ et Σ est connu : $\Sigma^{-1} = \Omega(\mathcal{G})$ (par exemple $\Omega(\mathcal{G}) = D - A + I$).

La statistique proposée est inspirée de W, mais est généralisée au cas où Σ_0^{-1} n'est pas forcément égale à I grâce à la transformation proposée par [Anderson, 2003]. Cette nouvelle statistique est notée W' :

$$W' = \frac{1}{p}\operatorname{tr}([S\Sigma_0^{-1} - I_p]^2) - \frac{p}{n}\left(\frac{1}{p}\operatorname{tr}(S\Sigma_0^{-1})\right)^2 + \frac{p}{n}.$$

Nous utilisons cette statistique initialement dédiée à un test de sphéricité modifié pour mesurer l'adéquation d'une famille de graphes à un jeu de données avec comme but de sélectionner le « meilleur « (au sens de celui pour lequel W' est minimal). Nous avons montré sur des jeux de données simulées que la procédure ainsi définie permet d'identifier un « bon » graphe, bon au sens défini ci-après.

3.4 Résultats obtenus sur données simulées

Pour comparer des graphes inférés ou de la littérature avec un graphe de référence, nous disposons d'indicateurs bien caractérisés et définis à partir de la table de contingence suivante :

	$i \sim j$	$i \nsim j$
$w_{ij} \neq 0$	VP	FP
$w_{ij} = 0$	FN	VN

Comme conseillé dans [Vert, 2008], nous avons choisi la valeur prédictive positive (ou Positive Prédictive Value, notée PPV dans la suite) et la sensibilité. Ces quantités seront notées

1. $ppv = \frac{VP}{VP+FP}$ pour la PPV et
2. $sen = \frac{VP}{VP+FN}$ pour la sensibilité.

Les réseaux biologiques étant assez pauvres en arêtes, il vaut en effet mieux se concentrer sur des indicateurs à base de nombres d'arêtes, prédites ou vraies, comme la PPV et la sensibilité, plutôt que sur le nombre d'« absences d'arête », comme la spécificité.

Comme ces indicateurs sont calculés à partir de la vérité terrain, ils peuvent servir de référence pour notre statistique de sphéricité modifiée. Ainsi nous pouvons :

- nous donner un graphe de référence,
- générer un échantillon à partir de ce graphe,
- inférer des graphes de qualité diverse sur cet échantillon,
- caractériser ces graphes d'une part par rapport au graphe de référence (avec la PPV et la sensibilité) et d'autre part avec W'.

Pour pouvoir tester les méthodes présentées dans cette partie, nous nous sommes uniquement servi de données simulées générées à partir d'un graphe. La première étape a donc été de générer aléatoirement un graphe.

3.4.1 Simulation d'un graphe aléatoire

Tout d'abord il faut obtenir un graphe de référence $\mathcal{G}_{ref}$ qui permettra de générer l'échantillon gaussien. Nous avons choisi de le générer aléatoirement plutôt que d'en prendre des exemples dans les bases de données (ce qui aurait été tout à fait possible) : on peut contrôler avec finesse la structure, la distribution des degrés, la taille de graphes aléatoires alors que c'est beaucoup plus difficile de justifier et de rationaliser ces manipulations sur des graphes « réels ». De plus, rien ne garantit que ces graphes « réels » modélisent bien le vivant.

Les graphes aléatoires utilisés ont été générés par la méthode d'Érdős-Rényi : les arêtes sont ajoutées itérativement avec une certaine probabilité de sorte que la distribution des degrés de connectivité atteigne un objectif donné. L'objectif ici est d'avoir une distribution *scale free* pour les degrés du graphe. Cette méthode est implémentée dans la librairie `igraph`. Un exemple de ce genre de graphes est donné sur la figure 3.5.

3.4.2 Modèle de génération de données simulées

Une fois le graphe de référence, noté $\mathcal{G}_{ref}$, donné, nous pouvons générer des données simulées gaussiennes multivariées. Avec la méthode de Cholesky présentée précédemment, nous générons n réalisations d'une variable aléatoire $X \sim \mathcal{N}(0, \Sigma = (D - A + I)^{-1})$. Le fait de disposer de ce modèle nous permet de tester les algorithmes d'inférence de réseaux évoqués dans les parties précédentes et de tester l'adéquation d'un ou plusieurs graphes à un échantillon.

3.4.3 Comparaison de différentes méthodes d'inférence de réseaux

On observe sur la figure 3.6 que la méthode permettant d'atteindre les meilleures performances en termes de sensibilité et de PPV est la méthode dérivée de la RR. Nous n'avons pas inclus dans cette comparaison les méthodes utilisant des régressions de

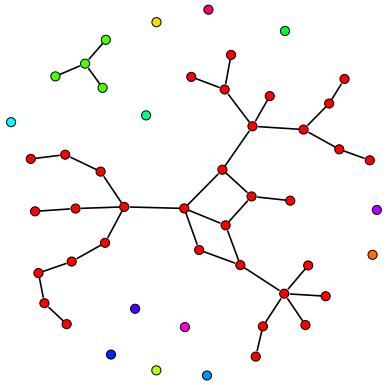


FIGURE 3.5 – Exemple de graphe aléatoire d'ordre $p = 50$ obtenu avec l'algorithme d'Erdos-Renyi. Les couleurs correspondent aux composantes connexes.

type LASSO car la méthode proposée pour fixer un seuil sur les matrices résultantes, reposant sur le *local* FDR, ne fonctionne plus. La méthode dérivée de la RR permet en outre d'obtenir des graphes dont la structure (cf. fig 3.7) est proche de celle du graphe de référence, tout en montrant des performances meilleures que celles des autres méthodes. Lors de l'application aux données réelles, c'est cette méthode que nous utiliserons.

3.4.4 Comparer des graphes inférés avec un graphe de référence

Il s'agit enfin de comparer l'indicateur W' que nous proposons à la PPV et la sensibilité que l'on peut calculer sur des données simulées, comme représenté sur la figure 3.8(a). Sur cette figure, la valeur de W' pour le graphe de référence est représentée par une ligne verticale en pointillés bleus foncés : cette limite n'est jamais atteinte par les graphes inférés, ce qui n'est pas surprenant vus les paramètres de la simulation $n = 100$ et $p = 100$. De plus, on observe sur les autres graphes inférés que le graphe choisi en minimisant W' réalise un compromis acceptable entre PPV et sensibilité.

3.5 Conclusion et Discussion

Dans cette partie, nous avons proposé une méthode permettant d'inférer des RRG à partir de données transcriptomiques utilisant la PLS-R. Cependant, des tests sur des données simulées ultérieurs aux travaux résumés dans [Tenenhaus et al., 2008] nous ont montré que la méthode basée sur la RR [Krämer et al., 2009] permet d'obtenir des

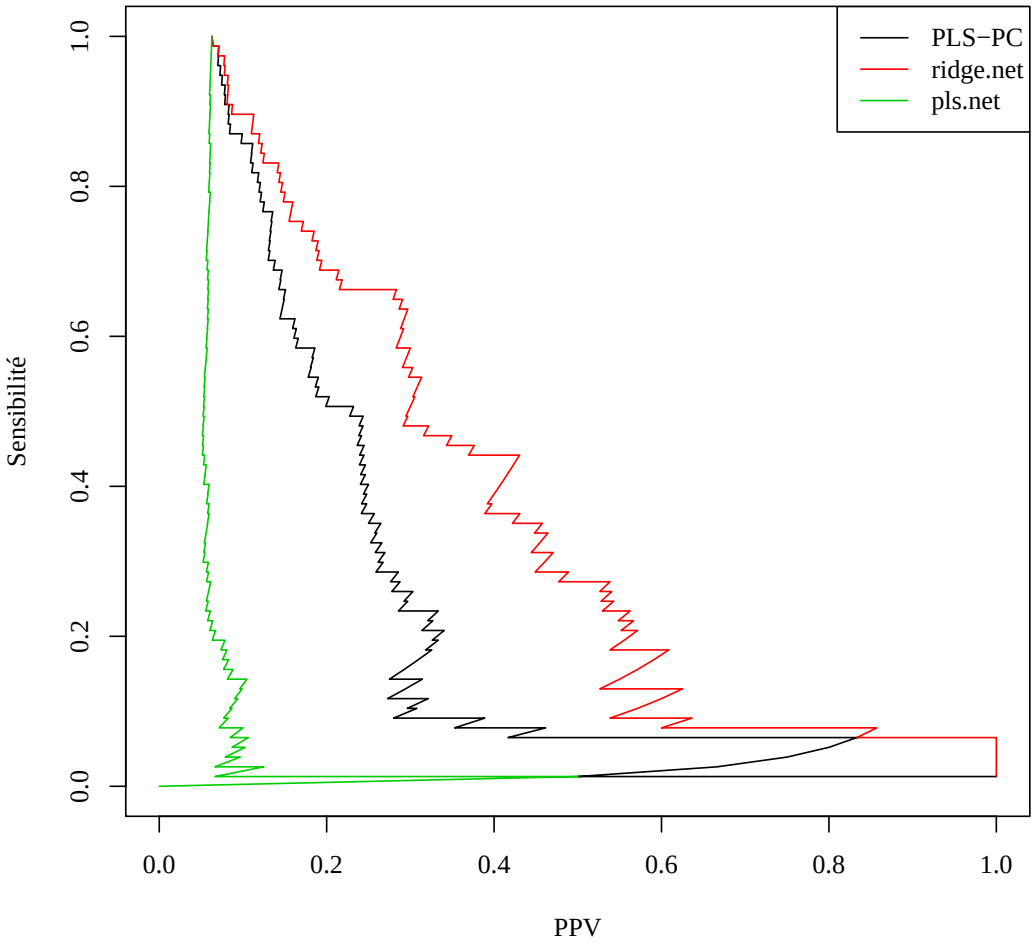


FIGURE 3.6 – Exemple d'inférence de réseaux avec trois méthodes : la méthode PLS-PC d'une part [Tenenhaus et al., 2008] ainsi que deux autres méthodes dérivées de la PLS-R (`pls.net`) et la RR (`ridge.net`)[Krämer et al., 2009]. Ces courbes sont tracées en faisant varier le seuil permettant de transformer la matrice de corrélations partielles inférées en graphe, la sensibilité et la PPV sont calculées pour chaque valeur de ce seuil. Nous observons que les deux méthodes utilisant la PLS-R montrent des performances inférieures à celles de la RR.

graphes dont la structure est plus proche de la réalité. Lors de l'application à des données réelles, c'est cette méthode que nous utiliserons.

Nous avons de plus présenté un indicateur, inspiré d'une statistique de sphéricité,

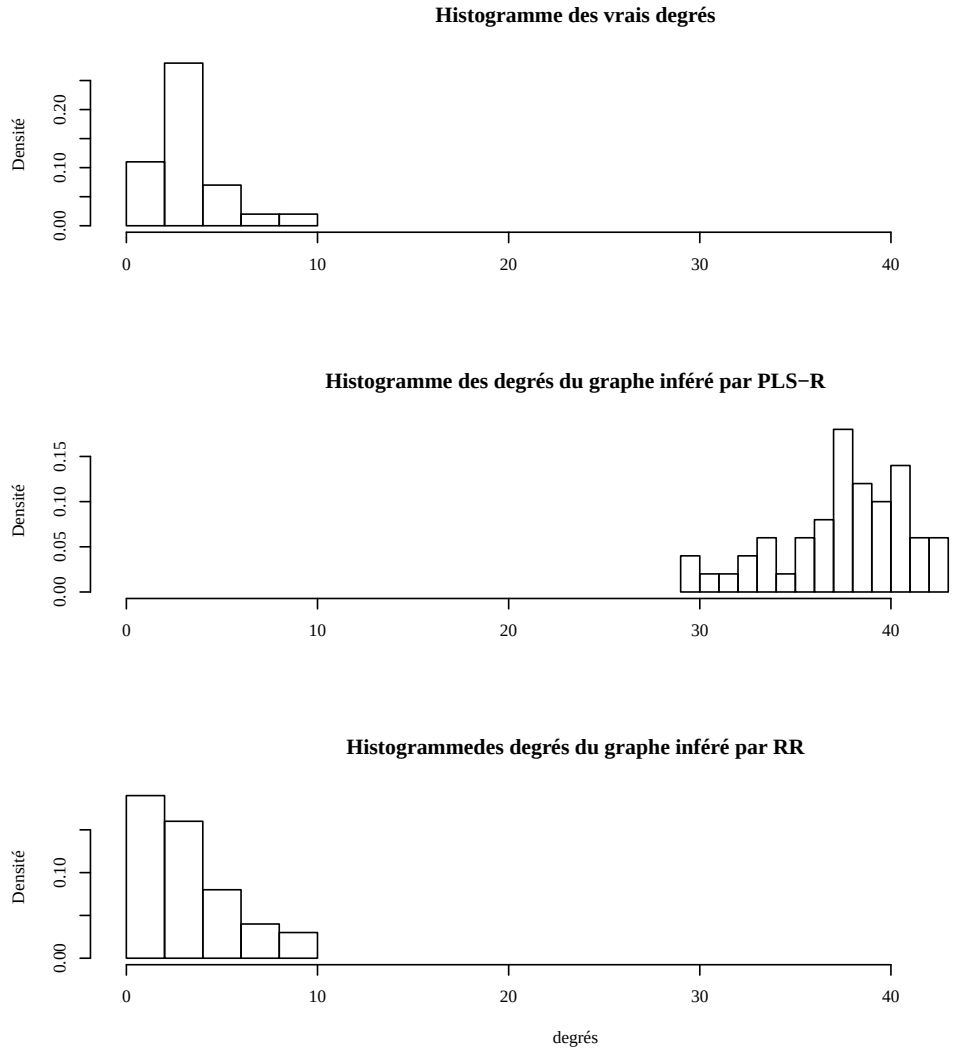


FIGURE 3.7 – Histogrammes des degrés du vrai graphe utilisé pour simuler les données (en haut), du graphe inféré par PLS-PC (au milieu) et du graphe inféré par la RR (en bas).

permettant de mesurer l'adéquation d'un graphe ou d'une famille de graphes à un jeu de données. Nous montrons sur des données simulées qu'il permet de sélectionner parmi plusieurs graphes celui qui réalise un compromis intéressant entre PPV et sensibilité. Il reste quelques points à éclaircir concernant cette mesure d'adéquation d'un graphe à un jeu de données avant de pouvoir appliquer cette méthode à des données réelles :

– la détermination des paramètres d'un modèle simple de passage entre graphe et matrice de variance covariance,

– l'utilisation dans cet indicateur d'un estimateur de la matrice de variance covariance plus adapté aux situations $n \ll p$ que S,
– l'interprétation des hypothèses suivantes présentées par Ledoit et Wolf pour déterminer la loi asymptotique de W

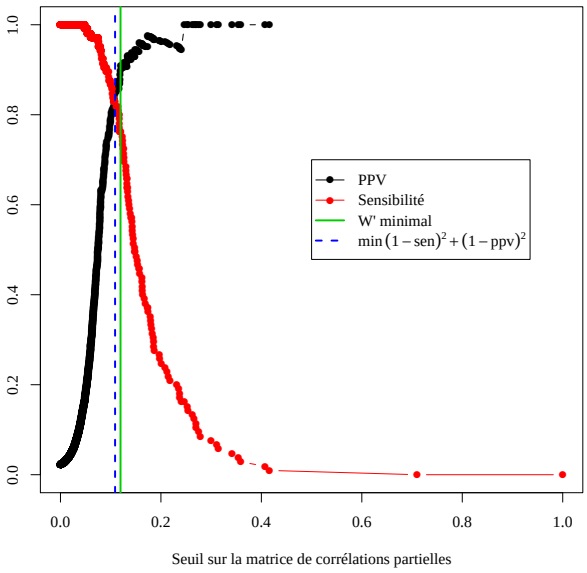
(a) le nombre de variables et d'individus sont des fonctions croissantes d'un indice entier k, $p = p_k$ et $n = n_k$ de sorte que $\lim_{k\to+\infty} p_k = +\infty$, $\lim_{k\to+\infty} n_k = +\infty$ et qu'il existe un réel $c > 0$ tel que $\lim_{k\to+\infty} p_k/n_k = c$;

(b) pour chaque indice k, X_k est une matrice $(n_k + 1) \times p_k$ de $n_k + 1$ observations i.i.d. d'une variable aléatoire gaussienne multivariée de moyenne μ_k et de matrice de variance-covariance Σ_k. Soit $\lambda_i^{(k)}, i = 1, ..., p_k$ les valeurs propres de Σ_k. On suppose que la valeur moyenne de ces valeurs propres $\overline{\lambda} = 1/p_k \sum_{i=1}^{p_k} \lambda_i^{(k)} > 0$ et que la dispersion $\delta^2 = 1/p_k \sum_{i=1}^{p_k} (\lambda_i^{(k)} - \overline{\lambda})^2$ sont toutes deux indépendantes de l'indice k.

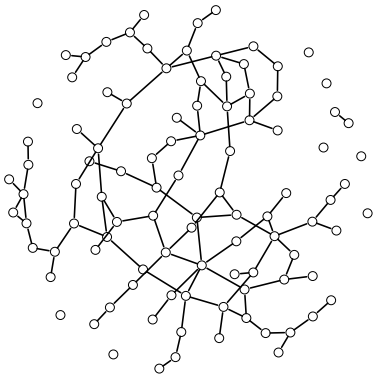
Sous ces deux hypothèses supplémentaires, et à condition que $(\overline{\lambda} - 1)^2 + \delta^2 = 0$, Ledoit et Wolf ont montré que

$$nW - p \xrightarrow{D} \mathcal{N}(1, 4), \text{ quand } n \to +\infty \text{ et } p \to +\infty, \tag{3.14}$$

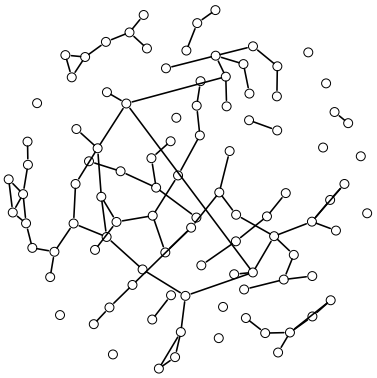
où $\xrightarrow{D}$ indique la convergence en distribution. Un travail complémentaire sera mené pour étudier l'influence de ces mêmes hypothèses sur la loi asymptotique de W' et leur signification sur la structure du graphe sous-jacent.



(a) Sensibilité et PPV de graphes inférés en fonction du seuil.



(b) $\mathcal{G}_{ref}$



(c) $\mathcal{G}$ ayant un W' minimal

FIGURE 3.8 – Représentation graphique de l'évolution de la sensibilité et de la PPV en fonction du seuil appliqué à la matrice des corrélation partielles estimée avec la méthode utilisant la régression Ridge [Krämer et al., 2009]. Les lignes verticales représentent le seuil à appliquer pour avoir W' minimal (ligne bleue en trait plein) et le seuil à appliquer pour obtenir une sensibilité et une PPV qui minimisent la quantité $(1 - sen)^2 + (1 - PPV)^2$ (ligne verte en pointillés). Le graphe de référence $\mathcal{G}_{ref}$ est représenté en bas à gauche, le graphe inféré ayant un W' minimal est représenté en bas à droite. $n = 100$ et $p = 100$.

Chapitre 4

Résultats de l'intégration d'un graphe dans un processus de classification sur des données transcriptomiques réelles

Trois jeux de données transcriptomiques ont été étudiés dans cette partie.

4.1 Sélection des Probe Sets correspondant aux gènes impliqués dans le cancer selon la base de données KEGG

Manipuler à la fois des réseaux de régulations génétiques et des jeux de données de microarray implique de savoir à quel gène correspond chaque variable du jeu de données, pour qu'il soit possible d'établir une bijection entre ces colonnes et les nœuds du graphe à inférer. Dans le cas particulier des puces à ADN Affymetrix, à chaque colonne correspond un identifiant constructeur prétendument unique. Cet identifiant est appelé Probe Set : il représente une synthèse des mesures d'expression effectuées sur un ensemble de sondes réparties tout le long d'une séquence cible du gène étudié. Les Probe Sets correspondant aux contrôles de qualité ne seront pas considérés dans la suite.

La sélection des sondes est décrite plus en détails en annexe C : les gènes correspondants au pathway KEGG hsa05200 (cancer chez l'homme) sont extraits, et les sondes correspondantes sur la puce sont identifiées. La gestion des sondes dupliquées se fait par leur position relative sur le gène et une fois que les sondes en « meilleure place » sont identifiées, les profils d'expression des doublons restants sont moyennés. On obtient donc à la fin de cet étape de filtrage un jeu de données pour lequel chaque variable correspond à un seul gène.

4.2 Inférence de réseaux de régulations génétiques

Pour chaque jeu de données, le fait d'avoir à disposition un réseau de régulations génétiques est modélisé par son inférence soit sur un jeu de données de même nature et indépendant soit sur une partie du jeu de données qui n'est pas utilisée pour la classification. La méthode utilisée pour l'inférence du réseau est la méthode `ridge.net` décrite dans la partie 3 [Krämer et al., 2009] qui permet de calculer la matrice des corrélations partielles transformée ensuite en graphe d'indépendances par un seuillage sur le *local* FDR (au seuil 0.8).

4.3 Description des données

Les jeux de données auxquels nous avons appliqué des méthodes de classification intégrant l'information contenue dans un RRG sont des données acquises dans le cadre d'étude sur le cancer. Toutes ont pour but d'aider le médecin dans le diagnostic d'un type de cancer particulier : les puces à ADN permettent d'identifier des marqueurs moléculaires de la différence entre deux classes d'individus.

4.3.1 Données de cancer de la prostate

Les données de cancer de la prostate sont issues d'un étude décrite dans [Singh et al., 2002]. Le jeu de données est constitué de 102 individus répartis en 52 individus atteints d'un cancer et 50 individus contrôles. Les résultats présentés par Singh *et al.* montrent une amélioration du pouvoir prédictif des marqueurs moléculaires identifiés par analyse différentielle par rapport aux indicateurs cliniques habituellement utilisés (comme le dosage de la PSA ou le score de Gleason).

Pour l'inférence de réseaux, deux jeux de données ont été extraits du même jeu de données [Chandran et al., 2007].

1. le premier pour les échantillons normaux : il s'agit des échantillons de tissu sain prélevé sur les patients cancéreux.
2. le deuxième pour les échantillons cancéreux : il s'agit d'échantillons tumoraux.

Les deux réseaux caractérisant les patients atteints d'un cancer de la prostate et les individus sains ont donc été obtenus de manière indépendante.

4.3.2 Données de cancer du colon

Les données sont celles décrites dans [Alon et al., 1999] ont été acquises dans le cadre d'une étude sur la cancer du colon. Le jeu de données est constitué de 62 individus répartis en 42 patients et 20 contrôles.

Pour l'inférence de réseaux, nous avons séparé le jeu de données en deux parties de tailles égales : la première partie sert à l'inférence des réseaux et la deuxième à la classification.

4.3.3 Données de cancer du poumon

Les données de cancer du poumon sont décrites dans [Lee et al., 2008] et sont disponibles sur le site GEO (sous l'identifiant GSE8894). Le jeu de données est constitué de 138 patients répartis en deux classes de taille égale : les patients ayant fait une rechute d'une part et les patients n'ayant pas fait de rechute d'autre part.

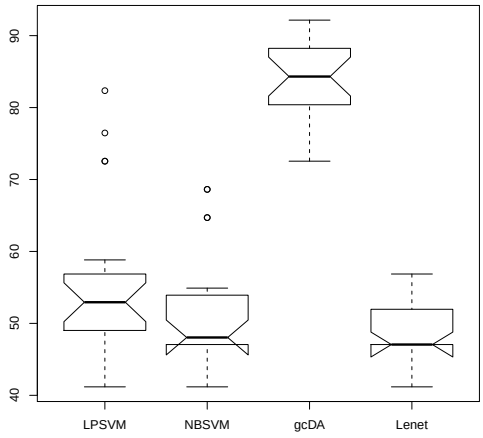
Pour l'inférence de réseaux, nous avons utilisé un seul jeu de données portant sur des individus atteints de cancer, dont le cancer du poumon (identifiant GSE8332).

4.4 Résultats de classification

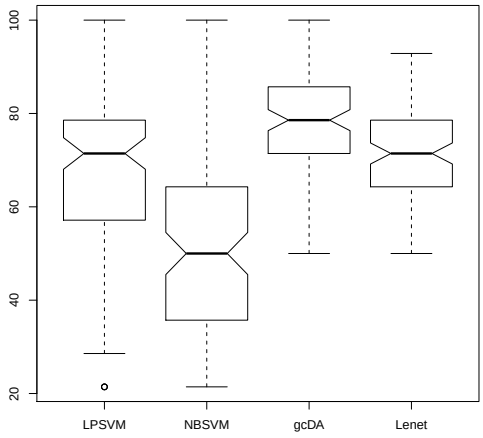
Nous avons appliqué à ces trois jeux de données les méthodes de classification avec intégration de RRG présentées dans la partie 2 : NB-SVM, Net et gCDA. La méthode LP-SVM est retenue comme méthode de classification de référence. Les résultats sont présentés sous forme de diagrammes en boites sur la figure 4.1.

Ces résultats montrent l'intérêt qu'il y a à intégrer dans la classification un réseau de régulations génétiques dont on dispose *a priori* : les trois figures 4.1(a), 4.1(b) et 4.1(c) montrent que gCDA permet d'améliorer les performances de classification. De plus, nous montrons également, et cela n'est pas en contradiction avec les déclarations des auteurs de ces méthodes, que les méthode de Zhu *et al.* et Li *et al.* échouent à améliorer les performances en classification sur des données transcriptomiques. La méthode gCDA permet d'obtenir des performances en classification significativement différentes d'une classification aléatoire, contrairement aux LP-SVM (cf. figure 4.1(c)).

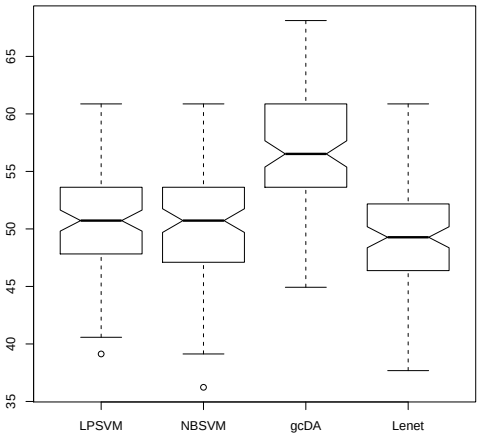
De plus, nous montrons également sur les figures 4.2(a), 4.2(b) et 4.2(c) que le graphe inféré par la méthode d'inférence basée sur la RR est bien intégré dans le processus de classification. Cela est en effet confirmé par le fait que les valeurs prises par les paramètres α, déterminées par k-fold, sont proches de 0.



(a) Cancer de la prostate

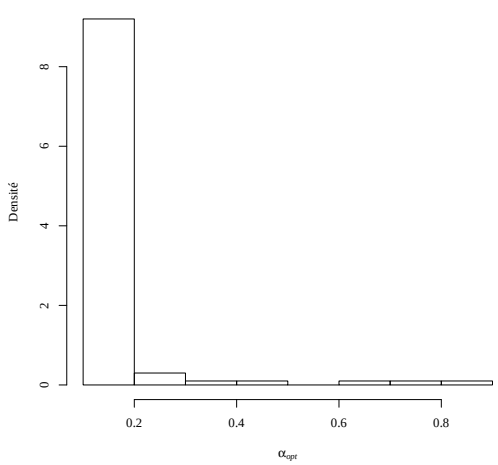


(b) Cancer du colon

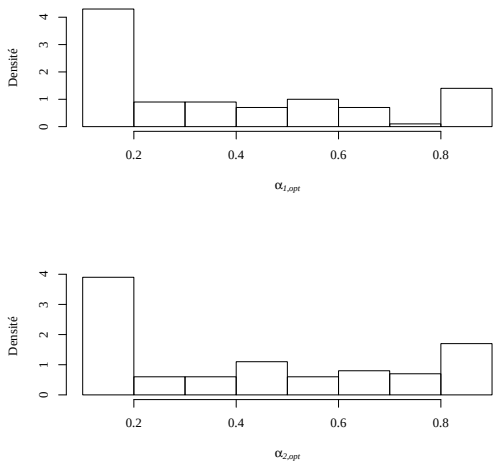


(c) Cancer du poumon

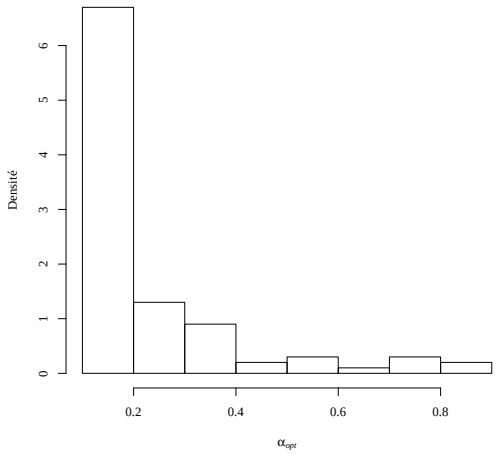
FIGURE 4.1 – Résultats de gCDA sur des données réelles et comparaison avec les méthodes de l'état de l'art. NB-SVM : méthode de Zhu *et al*. L-ENET : méthode de Li *et al*.. LP-SVM : SVM « linéaires ».



(a) Cancer de la prostate



(b) Cancer du colon



(c) Cancer du poumon

FIGURE 4.2 – Résultats de l'intégration des graphes dans gCDA. Les histogrammes représentent les valeurs prises par les paramètres α_1 et α_2 pour le cas quadratique (données de cancer du colon de Alon *et al.*) et par le paramètre α pour le cas linéaire. L'intégration s'est déroulée correctement : pour tous les jeux de données ces paramètres tendent à prendre des valeurs proches de 0.

Conclusion et perspectives

Conclusions

Classification de données transcriptomiques

Nous avons présenté une méthodologie de classification pour des données de puces à ADN reprenant un cadre de validation croisée permettant d'estimer correctement les performances en classification. Dans le cadre $n \ll p$ caractéristique des données transcriptomiques, afin de permettre aux méthodes de classification régularisées de déterminer une fonction de classification, il faut appliquer aux données d'apprentissage des méthodes de sélection d'attributs. Par ailleurs, la signature moléculaire obtenue à la fin de la classification doit être vérifiable par des techniques humides bas-débit.

Nous avons choisi la méthode de sélection d'attributs parmi des approches *wrapper* ou *filter* en fonction de leur capacité à identifier des gènes différentiellement exprimés. Sur des données simulées, nous avons montré que le FC permet d'identifier le mieux les gènes différentiellement exprimés. Notre choix du FC pour identifier des gènes différentiellement exprimés est également justifié par le fait que les signatures ainsi produites conviennent aux expérimentations bas débit classiquement menées à la suite des expériences microarray [Witten and Tibshirani, 2007].

Nous mettons également en évidence le rôle fondamental de l'estimation des matrices de variance covariance dans la classification.

Méthode gCDA

Nous avons proposé un cadre original d'intégration d'un RRG dans un processus de classification. L'information contenue dans le RRG est intégrée dans l'estimation des matrices de variance covariance utilisée dans l'analyse discriminante. Nous montrons sur des données simulées que gCDA atteint des performances meilleures que les méthodes de la littérature dans un cadre de classification où chaque classe est modélisée par une variable différente.

De plus, gCDA présente deux avantages par rapport aux méthodes de la littérature. D'une part il est possible d'intégrer un graphe par classe, ce qui se traduit par

une fonction de classification quadratique. D'autre part, la méthode proposée dépendant d'un hyperparamètre $\alpha \in [0;1]$ automatiquement estimé par validation croisée, il est possible de qualifier a posteriori l'intégration du graphe dans la classification : plus les valeurs prises par le paramètre α sont proches de 0, meilleure est l'intégration.

Inférence de RRG et adéquation d'un graphe à des données.

Pendant ma thèse, j'ai participé au développement d'une méthode d'inférence de RRG basée sur l'estimation de coefficients de corrélation partielle à l'aide de la PLS-R.

Nous avons également proposé un indice permettant de qualifier la qualité d'un graphe parmi une famille de graphes montrant des performances en termes de PPV et de sensibilité intéressantes sur des données simulées.

Application de gCDA à des données réelles.

Nous avons enfin appliqué à la fois des méthodes d'inférence de RRG et gCDA sur des données microarray issues de bases de données publiques.

La méthode gCDA montre de bonnes performances sur certaines de ces applications. Lorsque les performances de gCDA sont bonnes, nous montrons de plus que le graphe est bien intégré dans le processus de classification.

Perspectives

Simulations

Le modèle de simulation présenté au paragraphe 1.2.3, bien que prenant en compte la structure de corrélations existant entre variables, est encore assez frustre, il faudra l'enrichir en y intégrant par exemple une ou plusieurs sources de bruit.

Méthode gCDA

La méthode gCDA est une approche originale par rapport aux méthodes de l'état de l'art au sens où elle ne reprend pas la contrainte commune d'obliger des coefficients connectés dans le RRG à avoir des poids proches dans la fonction de classification [Rapaport et al., 2007] [Li and Li, 2008], [Zhu et al., 2009].

Comme nous l'avons vu au paragraphe 2.2.4, l'information contenue dans le graphe $\mathcal{G}$ a été intégrée dans gCDA en utilisant un nouvel estimateur de matrice de variance covariance :

$$\widehat{\Sigma}(\alpha) = \alpha S + (1-\alpha)\left(D - A + I\right)^{-1}.$$

Ainsi que nous l'avons suggéré au paragraphe 2.3, il est possible d'intégrer ce nouvel estimateur dans un noyau de Mahalanobis [Haasdonk and P ekalska, 2008].

Nous n'avons pas eu le temps de comparer gCDA à la méthode implémentée dans [Binder and Schumacher, 2009]. Cette méthode est basée sur une approche de boosting et est proposée comme une alternative à la méthode de Li *et al.* dans le cadre d'analyse de données de survie. Elle peut être utilisée dans un cadre de classification. Cela constitue une perspective évidente de notre travail.

Lien entre matrice de variance covariance et graphe

Durant toute la thèse, nous avons considéré que le graphe $\mathcal{G}$ était lié à la matrice de variance covariance Σ par la modèle $\Sigma^{-1} = D - A + I$, avec D la matrice des degrés et A la matrice d'adjacence du graphe. Nous envisageons d'enrichir ce modèle dans un premier temps en utilisant la statistique d'adéquation d'un graphe à des données que nous avons présentée pour ajuster le paramètre d'un modèle toujours inspiré du laplacien du graphe :

$$\Sigma^{-1} = D - A + \gamma I \text{ avec } \gamma > 0.$$

Nous envisageons également de nous inspirer des méthodes présentées par Letac *et al.* et Rajaratnam *et al.* pour déterminer Σ à partir de $\mathcal{G}$. La mise en œuvre de ces méthodes devrait être complétée par une étude plus poussée de l'influence du modèle existant entre Σ et $\mathcal{G}$ sur les performances en classification de gCDA. Cela pourrait se faire via un modèle paramétrique simple et robuste. Ce modèle serait appliqué à l'indicateur que nous avons proposé permettant de mesurer l'adéquation d'un graphe à des données, ce qui permettrait d'analyser des graphes et des données transcriptomiques réelles. L'indicateur ainsi adapté permettrait non seulement de positionner un seuil sur une seule matrice de corrélations partielles empiriques, mais aussi de choisir dans une famille de graphes celui qui est le plus en adéquation avec les données.

Parties annexes

Nous avons choisi de présenter dans le corps du mémoire tous les travaux effectués pendant la thèse en rapport avec l'intégration de l'information contenue dans un graphe dans un processus de classification. D'autres travaux ont également été menés car nécessaires à l'analyse de puces à ADN, mais que nous regroupons dans ces parties annexes afin de conserver un fil directeur clair. Dans son activité de plateforme, le LEFG a fabriqué des puces à ADN de type deux couleurs et a développé une bioinformatique dédiée à la génomique fonctionnelle à partir de puces une ou deux couleurs. ParnterChip est une société de services dont l'une des missions est d'hybrider et d'analyser des puces à ADN Affymetrix, donc à une couleur.

Dans ce cadre, nous avons contribué d'une part à étendre les connaissances bibliographiques de l'équipe de bioinformatique du LEFG à l'étude des puces à ADN Affymetrix. Cela a également permis à PartnerChip de mettre au point une routine d'analyse de puces Affymetrix, implémentée sous R. D'autre part, nous avons été partie prenante des expériences menées au LEFG, comme le montre notre participation au développement d'une nouvelle méthodologie appliquée aux données de toxicogénomique.

Les annexes sont structurées de la manière suivante. Tout d'abord, nous présentons des résultats complémentaires aux résultats présentés dans la partie 1. Ensuite, nous introduisons le concept des puces à ADN. Enfin, nous présentons plus particulièrement deux aspects sur lesquels nous avons travaillé durant la thèse.

- Le premier concerne la normalisation des données acquises sur les puces « une couleur » (type Affymetrix).
- Le second décrit une stratégie expérimentale d'étude d'un toxique dont les effets sont décrits uniquement au travers du phénotype général et d'un mécanisme moléculaire. Les effets sur le transcriptome de ce stress peuvent être étudiés mais le biologiste ne dispose alors que de peu d'information dosimétrique à ce niveau. Nous avons proposé une méthode qui considère une gamme de dose de toxique et reporte à l'étape d'analyse la sélection des doses d'intérêt pour l'analyse.

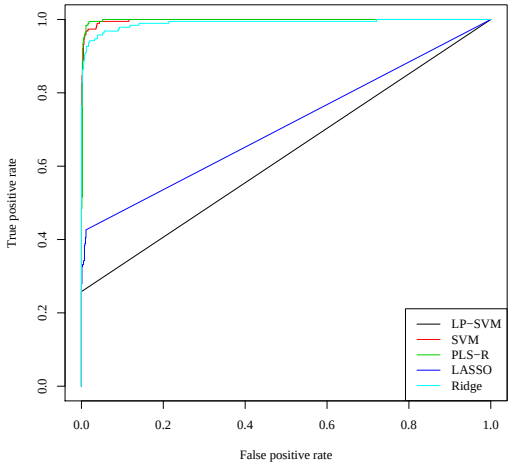
Annexe A

Résultats complémentaires

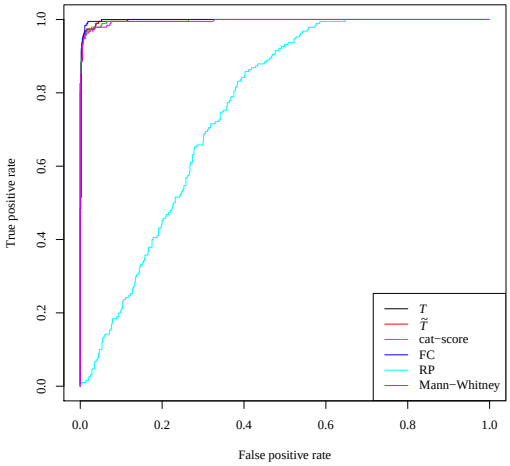
Nous présentons tout d'abord des résultats complémentaires aux résultats présentés sur des données simulées dans la partie 1. Les résultats que nous avons présentés dans cette partie concernaient la comparaison de méthodes d'analyse différentielle de type *filter* et *wrapper*. Les deux classes de données ont été générées selon une loi gaussienne multivariée avec une structure de corrélation inférée sur les données de [Golub et al., 1999].

Les figures suivantes présentent des résultats similaires pour un nombre d'individus plus grand que celui proposé dans le paragraphe 1.2.3 : $n = 100$ et 200.

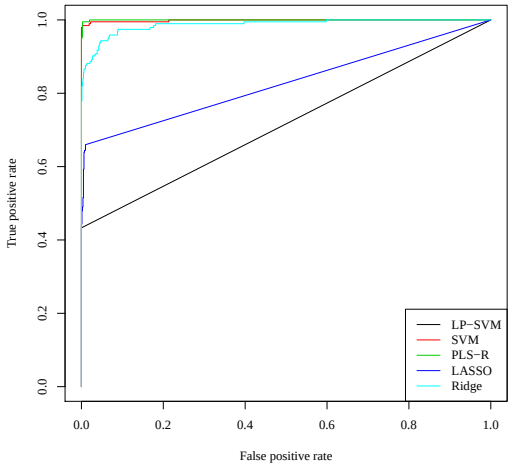
Ces figures nous montrent que l'augmentation du nombre d'individus, dans des proportions réalistes par rapport aux expériences de puces à ADN, n'influe pas sur les conclusions que nous avons tirées dans la partie 1. En effet, le FC reste toujours un indicateur robuste du caractère différentiellement exprimés des gènes.



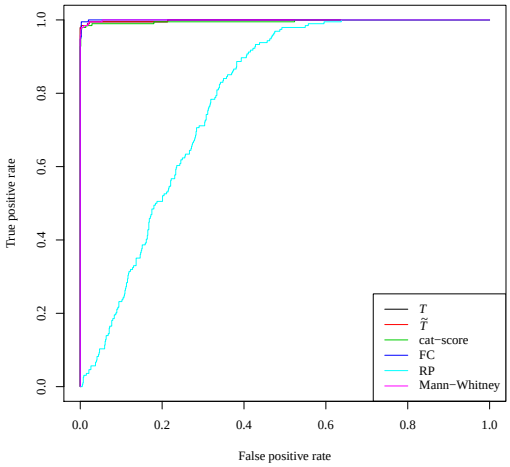
(a) Classification,$n = 100$



(b) Statistiques,$n = 100$

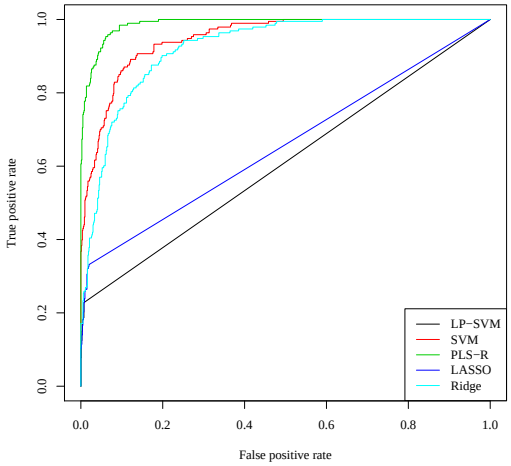


(c) Classification,$n = 200$

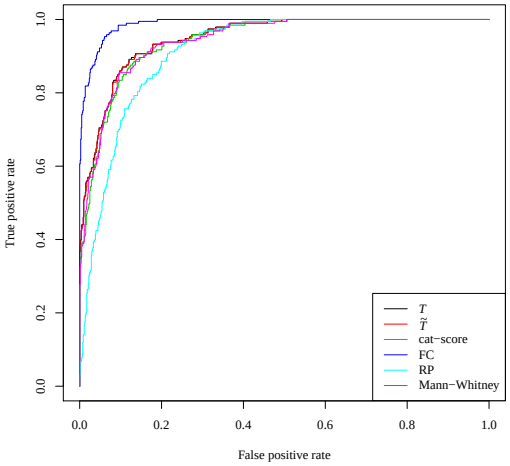


(d) Statistiques,$n = 200$

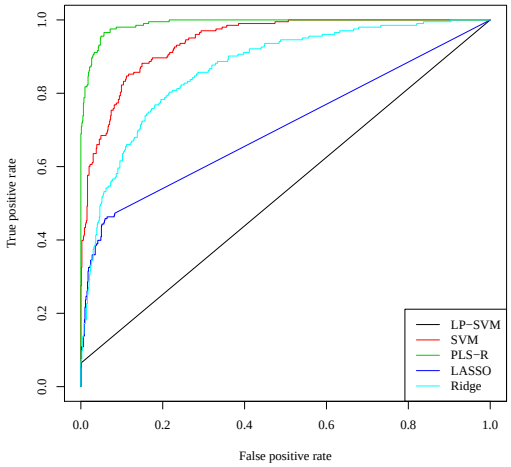
FIGURE A.1 – $n = 100, 200$ individus (50 par classe) et $p = 1000$. Modèle discret.



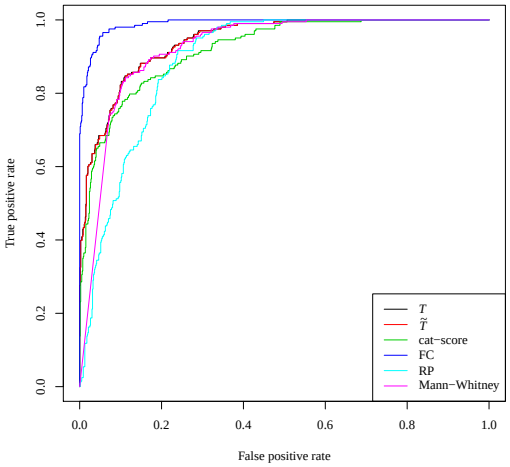
(a) Classification,$n = 100$



(b) Statistiques,$n = 100$



(c) Classification,$n = 200$



(d) Statistiques,$n = 200$

FIGURE A.2 – $n = 200$ individus (100 par classe) et $p = 1000$. Modèle normal.

Annexe B

Normalisation de puces à ADN

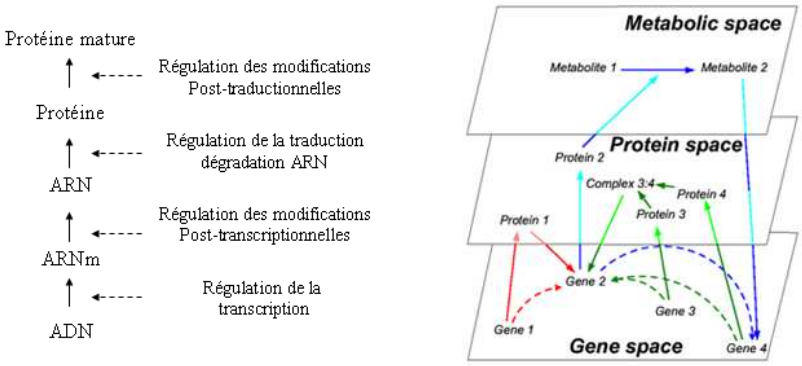
Dans cette annexe, nous présentons tout d'abord le contexte général de la fabrication des puces à ADN en nous focalisant sur les technologies propres au LEFG et à PartnerChip : les puces à ADN deux couleurs et les puces de la société Affymetrix. Ensuite, nous présentons une étude bibliographique des méthodes de normalisation des puces Affymetrix. Cette étude a bénéficié à l'équipe bioinformatique du LEFG, en s'ajoutant à l'étude bibliographique déjà effectuée sur la normalisation des puces à ADN deux couleurs. Elle a également bénéficié à PartnerChip, car elle a également donné lieu à l'implémentation dans le langage R d'une routine d'analyse aujourd'hui utilisée pour les projets de transcriptomique.

Une expérience de génomique fonctionnelle à « haut débit » consiste à extraire l'ensemble des transcrits (*i.e.* l'ensemble des acides ribonucléiques) d'un ensemble de cellules prélevées sur un individu afin de mesurer leurs concentrations. Cette opération se fait classiquement par dépôt des ARN prélevés (après amplification, marquage et souvent traduction en ADN complémentaire) sur une surface solide présentant des sondes caractéristiques de tous les gènes de l'organisme étudié. Nous rappelons dans un premier temps la structure d'une cellule eucaryote et dans un deuxième deux types de puces à ADN.

B.1 L'ARN

La plupart des êtres vivants sont constitués d'une ou plusieurs cellules. La cellule est définie comme un volume cloisonné par une membrane contenant toutes les molécules nécessaires à la vie de l'organisme. Les fonctions de la cellule vivante sont sous-tendues par des interactions entre des molécules d'ADN, d'ARN, les protéines, des molécules qui seront métabolisées etc. L'un des principaux paradigmes de la biologie moléculaire (voir figure B.1(a)) définit les différentes catégories de molécules présentes dans la cellule (ADN, ARN et protéines) et comment l'information transite entre ces catégories jusqu'à la synthèse des protéines. Ce que ne représente pas ce paradigme, ce sont les interactions entre ces différents niveaux (voir B.1(b)) : des protéines ou des molécules des ARN (dits interférants) régulent ou interfèrent avec la transcription ou

la traduction de gènes. Ce que nous avons appelé réseau de régulations génétiques (RRG) correspond ainsi à la projection du réseau de toutes ces interactions dans l'espace des mesures du transcriptome.



(a) Paradigme de la biologie moléculaire (b) Différents niveaux d'interaction

FIGURE B.1 – La planche (a) illustre un des grands paradigmes de la biologie moléculaire : les catégories de molécules et un petit nombre des mécanismes de contrôle sur le flux d'information. La planche (b) illustre comment le gène 1 interagit (flèche rouge pointillée) sur le gène 2 en passant par le niveau des protéines. Un exemple plus complexe est donnée pour les gènes 3 et 4 qui modulent le gène 2 par formation d'un complexe protéique. Communément on considère que ces interactions peuvent être projetées en un réseau d'interaction génétique qui rassemble les interactions du type flèches pointillées. Planche (b) d'après [Brazhnik et al., 2002]

B.2 Présentation générale des puces à ADN

Les étapes de la fabrication et de l'hybridation d'une puce à ADN sont en général toujours les mêmes :

1. fabrication de la puce,
2. extraction des ARNS des échantillons biologiques,
3. amplification(s), marquage(s) fluorescent(s) des ARNs,
4. dépôt sur la puce et hybridation,
5. lecture de la puce,
6. traitements manuels (caractérisation visuelle de la qualité d'hybridation) et numériques (normalisations)

Toutes ces étapes sont différentes d'un type de puces à un autre, nous détaillons rapidement ces différences pour les puces à ADN deux couleurs et une couleur.

Puces à ADN deux couleurs. Une puce à ADN deux couleurs est une lame de verre ou de polymère recouverte d'une substance permettant aux sondes ADN de s'accrocher. On dépose ainsi dans un premier temps un ensemble d'îlots correspondant chacun à une séquence spécifique du génome étudié. Le dépôt de ces îlots se fait à l'aide d'un robot (à l'aide d'aiguille, par projection etc.) et les sondes présentes dans chaque îlot sont issues d'une banque de sondes spécifique de la plate forme dans laquelle les puces sont fabriquées. Par exemple la plate forme du LEFG a contribué à la création des banques HuOli et MuOli [Brigand et al., 2006].

Après le dépôt de ces sondes sur la lame de verre, on peut déposer une solution obtenue à partir de deux échantillons de cellules. Cette solution a été obtenue après une série de transformations que l'on peut résumer en quelques étapes clé :

(a) extraction des ARN de l'échantillon de cellules,

(b) amplification,

(c) et marquage fluorescent.

Pour pouvoir différencier les ARN provenant des échantillons de cellules, on utilise deux marqueurs fluorescents différents : la cyanine 3 et la cyanine 5. Ces deux marqueurs émettent des rayonnements dans deux zones du spectre différentes et peuvent être excités par un même rayonnement source. La solution est déposée sur la puce puis mise dans des conditions de température favorables à l'hybridation des copies des transcrits aux sondes présentes dans les îlots. Cette hybridation est dire compétitive car des copies de transcrits marqués différemment (et donc issus de deux populations de cellules différentes) vont tenter de s'hybrider aux même îlots.

Ainsi, après l'hybridation compétitive, on peut mesurer à l'aide d'un scanner l'intensité lumineuse émise par chaque îlot pour chaque échantillon biologique. Ce genre d'hybridation est bien adapté dans le principe aux échantillons appariés (par exemple deux échantillons de cellules avant et après un traitement chimique ou encore deux échantillon prélevés sur un patient avant et après qu'il ait suivi un régime spécial). Quand les échantillons ne peuvent pas être appariés, on a recours à un « intermédiaire » : chaque échantillon est hybridé compétitivement avec un banque de sondes dite « universelle ».

Le résultat de l'hybridation et de la lecture par un scanner est une image ressemblant à l'image présentée figure B.2(a). Cette image doit encore être analysée à la fois manuellement (un opérateur vérifie la qualité de l'hybridation pour chaque îlot) et numériquement (on doit appliquer des méthode numériques de correction des biais techniques et de normalisation) avant de pouvoir obtenir la concentration supposée en transcrits pour chaque échantillon.

Pour les puces à deux couleurs, un profil d'expression le ratio des expressions dans l'échantillon 1 par rapport à l'échantillon 2.

Puces à ADN une couleur. Les puces à ADN une couleur sont également des support en polymère recouvert d'une substance permettant l'accrochage des sondes. Les sondes, des oligos courts de 25 bases, sont déposées à la surface de la puce par photo lithogravure (technologie utilisée également dans son principe pour fabriquer des circuit intégrés) ; base par base, un masque spécial permet de déposer la base qu'il faut (A, T, C ou G) où il faut. La densité des sondes déposées par ce procédé est bien supérieure à celle possible par le procédé précédent.

Une fois la puce fabriquée, une solution contenant des copies des transcrits présents dans un seul échantillon biologique y est déposée. Les copies des transcrits sont marqués avec de la biotine. Après hybridation et lecture, il faut encore procéder à la lecture de l'image et à des traitements numériques avant de pouvoir obtenir le vecteur des concentrations en transcrits. Une image d'une telle puce est représentée figure B.2(b).

Notons que ce ne sont pas les seules puces sur le marché permettant de réaliser des études transcriptomiques à haut débit : la société Illumina propose également des puces à microbilles, et les séquenceurs utilisés jusqu'à très récemment uniquement pour décoder des séquences ADN sont utilisés maintenant pour décoder des séquences ARN.

B.3 Le profil d'expression

À partir de la puce à ADN, nous obtenons pour chaque individu, ou pour chaque échantillon d'ARN, un vecteur de taille égale au nombre de sondes différentes présentes sur la puce. Ce nombre est du même ordre de grandeur que le nombre de gènes dans l'organisme étudié, on peut donc dire qu'on obtient un vecteur x de taille $1 \times p$ avec p de l'ordre de 10000 (gènes) : par exemple, l'organisme *Homo sapiens* est caractérisé par 25000 gènes.

Chaque composante de ce profil d'expression correspond à un identifiant de la banque de sondes utilisée pour les puces deux couleurs, à un Probeset Affymetrix, ou plus généralement à un identifiant de séquence sur le génome de l'organisme étudié qui correspond à un gène connu, un gène putatif ou une sonde servant au contrôle de qualité. Ces identifiants ont donc le plus souvent des correspondants dans les bases de données d'annotation publiques. On peut dénombrer plusieurs identifiants :

- le SYMBOL,
- le GENE NAME, très proche du SYMBOL, qui décrit la fonction identifiée du gène concerné,
- l'identifiant RefSeq (identifiant de séquence),
- l'identifiant ENTREZ,
- l'identifiant KEGG, qui permet entre autres de retrouver dans quels pathways de la base de données KEGG le gène en question a un rôle à jouer,
- l'identifiant GO qui permet de classer les gènes en fonction de leur(s) catégorie(s) fonctionnelle(s).

Un problème récurrent dans l'analyse de puces à ADN est que plusieurs Probe Sets

peuvent correspondre au même gène. Nous proposons en annexe C une méthodologie permettant de faire correspondre à un gène un seul Probe Set.

B.4 Normalisation

La technologie des puces à oligonucléotides développée par la société Affymetrix permet d'observer l'activité transcriptomique d'un ensemble de cellules prélevé sur un organisme eucaryote. Après amplification et marquage par un marqueur fluorescent, l'ARN extrait de ces cellules est déposé sur la puce, permettant aux brins d'ARN présents dans la préparation de s'hybrider aux sondes de 25 paires de bases présentes à la surface de la puce. Deux sortes de sondes appariées ont été déposées : à une sonde dite PM (Perfect Match), spécifique d'un transcrit, correspond une sonde MM (MisMatch) identique sauf au niveau de la 13^e paire de base, qui a été mutée. Ainsi les sondes PM s'hybrident spécifiquement avec un transcrit particulier, mais peuvent également s'hybrider avec des brins d'ARN non spécifiques de ce transcrit, ce phénomène, appelé hybridation non-spécifique, est mesuré à l'aide des sondes MM. Chaque transcrit est caractérisé par 11 ou 20 paires de sondes. Pour un même transcrit, on appelle l'ensemble des paires de sondes PM et MM le Probe set.

Le preprocessing des puces à oligonucléotides consiste en une succession d'étapes permettant une correction du bruit, une normalisation entre les lames d'un même échantillon, l'estimation de l'indice d'expression et enfin le résumé de l'ensemble des PM et MM d'un Probe set en une seule valeur caractéristique de l'expression d'un gène [Freudenberger, 2005]. L'étape de correction du phénomène d'hybridation non-spécifique semble être celle qui détermine le plus la qualité du preprocessing [Irizarry et al., 2006]. C'est donc à la modélisation de ce phénomène que nous allons nous intéresser dans ce document plutôt qu'au résumé ou à la normalisation.

Ces différentes étapes sont détaillées ci-dessous.

Correction du bruit de fond Le bruit mesuré sur la puce provient de phénomènes physiques et biologiques listés dans [Freudenberger, 2005] ; une hypothèse classique pour ce genre de bruit est qu'il est additif et gaussien.

Normalisation Le but de cette étape est de corriger les biais dus à la variabilité en abondance de matériel génétique, d'efficacité de marquage, d'hybridation ou d'amplification entre toutes les puces d'une expérience.

Correction du phénomène d'hybridation non-spécifique C'est l'étape critique du preprocessing. Elle permet entre autres de compenser le biais dû à l'hybridation non-spécifique.

Résumé On effectue le résumé des valeurs calculées pour toutes les paires de sondes d'un Probe set en une seule valeur caractéristique et, si possible, robuste : par exemple une moyenne robuste de Tukey biweight ou une valeur déterminée par l'algorithme median-polish, [1] ... [Freudenberger, 2005]

1. L'algorithme median-polish est utilisé par la méthode RMA pour résumer les Probe sets de toutes les puces de l'expérience considérée. Le principe est d'ajuster un modèle linéaire aux données d'expres-

Le résultat est une estimation de l'abondance de chaque transcrit représenté sur la puce. Cette estimation sera appelée dans la suite indice d'expression.

Pour savoir si une méthode de preprocessing permet d'estimer correctement l'abondance en transcrits dans une préparation, il faut la confronter à des données issues de préparations pour lesquelles on connaît la composition en ARN. Nous présentons dans la suite quelques-uns des jeux de données qui répondent à cette attente.

Les données de dilution ont été produites par GeneLogic à partir de deux sources d'ARN : deux lignées cellulaires de tissus humains hépatiques et du système nerveux central hybridées sur des puces HG-U95Av2. [Irizarry et al., 2003]. 6 groupes de puces ont été étudiés correspondant à des concentrations de 1.25, 2.5, 5.0, 7.5, 10.0 et 20.0 μg d'ARN total, au final, 60 lames ont été produites.

L'étude Spike in a été développée par Affymetrix pour valider l'algorithme MAS5 à partir de fragments d'ARN correspondant à 16 Probe sets de la puce GeneChip HG-U95A. Ces fragments ont été ajoutés à des préparations d'ARN dont les concentrations sont des puissances de 2 allant de 2 à 1024 pM[2]. La même source d'ARN a été utilisée dans tous les cas pour la préparation d'ARN. Ainsi, un petit nombre de gènes seront différentiellement exprimés tout en étant mélangés à une population classique d'ARN qui est identique pour toutes les puces.

Trois critères sont considérés pour comparer différentes méthodes de preprocessing. Ils consistent à évaluer :

- la précision de l'indice d'expression, évaluée par sa variabilité ;
- la consistance de l'estimation du fold-change ;
- la spécificité et la sensibilité de l'indice à détecter les gènes différentiellement exprimés.

Le package R `affycomp` , développé par Irizarry et al. ([Irizarry et al., 2003], [Wu et al., 2004], [Irizarry et al., 2006]) permet de générer automatiquement des représentations visuelles de ces critères que nous allons largement utiliser dans ce document.

Différents modèles ont été proposés pour l'influence du phénomène d'hybridation non spécifique sur le signal mesuré sur une puce.

Les notations employées dans tout le document sont celles adoptées par Irizarry et al. : $PM_{n,i,j}$ est l'intensité mesurée pour une sonde PM et $MM_{n,i,j}$ l'intensité mesurée pour une sonde MM, $i = 1, ..., I$ représente la puce, $j = 1, ..., J_n$ la sonde et $n = 1, ..., N$ le Probe set (ou gène).

Le nombre de gènes N est de l'ordre de 20 000, le nombre de puces I est le plus souvent de quelques dizaines et le nombre de paires de sondes J_n varie de 11 à 20. A partir des intensités des spots PM et MM, il faut estimer la quantité de transcrit présent dans la préparation déposée sur la puce. Dans la suite, on notera $\theta_{n,i}$ cette quantité pour le gène n et l'échantillon i et son estimation l'indice d'expression $\hat{\theta}_{n,i}$.

Modèle additif, MAS Par défaut, l'algorithme de preprocessing effectué sur les puces à oligonucléotides effectue le traitement suivant :

sion, de façon itérative, en considérant les médianes à travers les Probe sets.

2. pM est l'abréviation de picoMolaire. $1\ M = 1\ mol.L^{-1}$

$$\hat{\theta}_{n,i} = \frac{1}{J} \sum_j \left(PM_{n,i,j} - MM_{n,i,j} \right) \tag{B.1}$$

Il est basé sur le modèle simple additif suivant :

$$PM_{n,i,j} - MM_{n,i,j} = \theta_{n,i} + \epsilon_{n,i,j} \text{ avec } \epsilon_{n,i,j} \sim N(0, \sigma^2) \tag{B.2}$$

Cette différence est souvent négative. C'est par exemple flagrant sur le jeu de données disponible en ligne acquis par Golub et al. qui a subi cette transformation et qui présente de nombreuses valeurs négatives (environ 50 %). Ce jeu de données est disponible sous R dans le package multtest grâce à la commande $data(golub)$. Outre le fait qu'il est difficilement interprétable biologiquement d'avoir une abondance négative de transcrit dans une préparation, cela empêche également le passage au logarithme, qui est une transformation préconisée par nombre de méthodes de preprocessing car elle semble stabiliser la variance dans une certaine mesure, à la condition d'être effectuée sans correction du bruit d'hybridation non-spécifique [Durbin and al., 2002].

MAS5 est la version 5.0 de la solution "MicroArray Suite" proposée pour le preprocessing des puces à oligonucléotides par la société Affymetrix. L'indice d'expression $\hat{\theta}_{n,i}$ MAS5 est calculé suivant le protocole suivant pour un gène donné :

$$\hat{\theta}_{n,i} = \log^{-1}\left(\mathrm{T}_{\mathrm{bi}j} \left(\log \left(PM_{n,i,j} - MM^*_{n,i,j} \right) \right) \right) \tag{B.3}$$

où $MM^*_{n,i,j}$ est une valeur corrigée de l'intensité mesurée pour le spot MM dans le cas où $PM_{n,i,j} < MM_{n,i,j}$. Cette correction est due au fait que pour un grand nombre de sondes (jusqu'à 1/3), la quantité $PM_{n,i,j} - MM_{n,i,j}$ est négative. La fonction T_{bi} (Tukey Biweight) permet une moyenne pondérée des quantités $PM_{n,i,j} - MM^*_{n,i,j}$ pour un Probe set donné.

Malheureusement cette correction, qui est peu documentée, ne protège pas totalement contre la présence de valeurs négatives après preprocessing.

Modele multiplicatif [Li and Wong, 2001] Les solutions proposées par Affymetrix n'étant pas totalement satisfaisantes, le modèle suivant suivant a été développé :

$$y_{i,j} = PM_{i,j} - MM_{i,j} = \theta_i \phi_j + \epsilon_{i,j}, \ \sum_j \phi_j^2 = J, \ \epsilon \sim N(0, \sigma^2) \tag{B.4}$$

Ici, ϕ_j modélise la sensibilité d'une paire de sondes d'un Probe set au phénomène d'hybridation spécifique : plus ϕ_j est grand, meilleure est la réponse de la sonde PM. Ce modèle est basé sur l'observation que la variabilité entre sondes est 4 ou 5 fois supérieure à celle qui existe entre puces. Ce modèle prend donc mieux en compte le comportement différents des sondes d'un même Probe set. De façon plus générale, les hypothèses suivantes ont été effectuées sur les comportements des sondes PM et MM.

$$MM_{i,j} = \nu_j + \theta_i \alpha_j + \epsilon \tag{B.5}$$

$$PM_{i,j} = \nu_j + \theta_i \alpha_j + \theta_i \phi_j + \epsilon \tag{B.6}$$

Avec $\nu_j + \theta_i \alpha_j$ représentant la réponse de la $j^{\text{ème}}$ paire de sondes à une hybridation non-spécifique, pour l'échantillon i.

L'implémentation de ces modèles a été effectuée dans plusieurs langages par Lemon et al. dans un logiciel appelé DCHIP [Lemon et al., 2002].

Modèle log-linéaire additif GCRMA L'indice d'expression RMA est calculé grâce au modèle statistique suivant :

$$T(PM_{n,i,j}) = e_{n,i} + a_{n,j} + \epsilon_{n,i,j} \tag{B.7}$$

où T représente une transformation qui corrige le bruit de fond, normalise, et prend le logarithme des intensités pour les sondes PM, $e_{n,i}$ est le logarithme en base 2 de la valeur d'expression sur la puce i, $a_{n,j}$ modélise l'affinité pour les sondes j du Probe set considéré, et $\epsilon_{n,i,j}$ est un terme de bruit. Non seulement le modèle estimant l'indice d'expression est différent de celui utilisé par [Li and Wong, 2001], mais en plus les méthodes de correction de bruit de fond et de résumé des Probe sets sont également différentes. La séquence de la sonde a une importance, puisque toutes les bases A, T, G et C ne présentent pas la même affinité entre elles. Cela n'est pas négligeable et est pris en compte par l'algorithme GCRMA [Wu et al., 2004]. Le modèle de cette influence est le suivant :

$$\alpha = \sum_{k=1}^{25} \sum_{b \in \{A;T;G;C\}} \mu_{b,k} 1_{b_k=b} \text{ avec } \mu_{b,k} = \sum_{l=0}^{3} \beta_{b,l} k^l \tag{B.8}$$

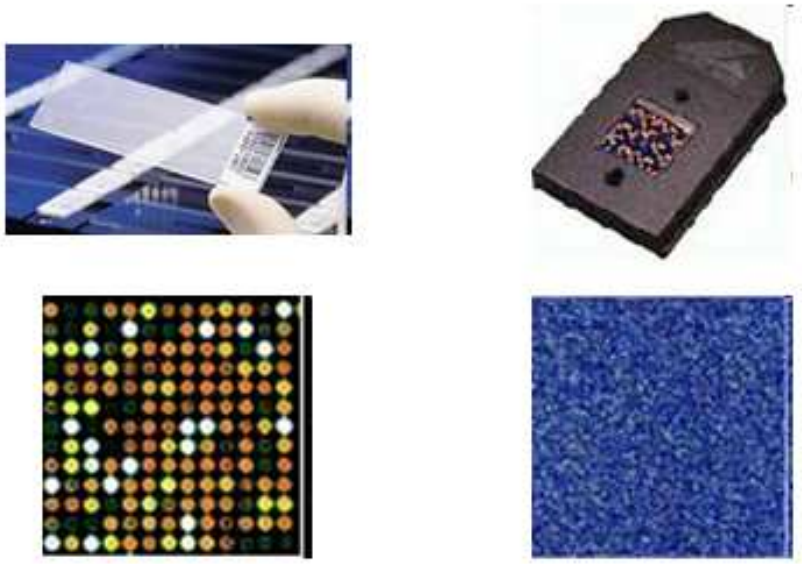
où $k = 1, ..., 25$ indique la position de la base considérée le long de la sonde, b la lettre de la base, b_k la base en position k, $1_{b_k=b}$ est une fonction indicatrice qui vaut 1 quand la base en position k est de type j, 0 sinon, et $\mu_{j,k}$ représente la contribution à l'affinité de la base b en position k. A b fixé, l'effet $\mu_{b,k}$ est un polynôme de degré 3.

Le preprocessing est une étape essentielle qui influe énormément sur les résultats issus d'études statistiques visant à détecter des gènes différentiellement exprimés, comme en témoigne notamment les études Spike in référencées par Irizarry et al.. Ce domaine de recherche est très actif et prolifique, et plusieurs questions restent ouvertes :

- Comment effectuer une correction du bruit de fond qui se trouve sur l'image de la puce ? Par bruit, on entend les biais techniques dus à l'amplification de l'ARN, aux lavage et séchage de la puce, au scanner utilisé etc.
- Les jeux de données Spike in et de dilution reflètent-ils bien ce qui peut se trouver dans un échantillon biologique classique ?
- L'ordre dans lequel est réalisé les différentes étapes du preprocessing est-il important ?

La société Affymetrix a récemment développé une nouvelle méthode de preprocessing, qui semble égaler les performances de la méthode GCRMA [Irizarry et al., 2006].

Elle est malheureusement peu documentée : il s'agit de la méthode PLIER (Probe Logarithmic Intensity Error [AffyTeam, 2005]).



(a) Puce à ADN deux couleurs (b) Puce à ADN une couleur

FIGURE B.2 – Deux technologies pour les puces à ADN. En (a) et en haut une lame de verre imprimées avec des oligonucléotides longs et en bas en fausses couleurs l'image scannée d'une portion hybrideés avec des extraits marqués en deux couleurs. En (b) et en haut une puce Affymetrix utilisant la photo lithogravure pour la synthèse *in situ* des sondes et en bas en fausses couleurs l'image scannée d'une portion de puces. On remarquera la différence de densité entre les deux technologies (1 à 2 ordres de grandeur).

Annexe C

Sélection des sondes impliquées dans le cancer selon la base de données KEGG

Nous proposons dans cette partie annexe une méthodologie permettant de sélectionner, parmi plusieurs identifiants Affymetrix ciblant le même gène, celui qui est le mieux positionné sur le gène. Cette méthodologie a été appliquée aux jeux de données réels utilisés dans le chapitre 4.

Manipuler à la fois des réseaux de régulations génétiques et des jeux de données de microarray implique de savoir à quel gène correspond chaque colonne du jeu de données, pour qu'il soit possible d'établir une bijection entre ces colonnes et les nœuds du graphe. Dans le cas particulier des puces à ADN Affymetrix, à chaque colonne correspond un identifiant constructeur prétendument unique. Cet identifiant est appelé Probe Set : il représente une synthèse des mesures d'expression effectuées sur un ensemble de sondes réparties tout le long d'une séquence cible du gène étudié. Les Probe Sets correspondant aux contrôles de qualité ne seront pas considérées dans la suite. Premièrement, le tableau C.1 montre les pourcentages de Probe Sets référencés dans la base de l'UCSC ciblant plusieurs gènes distincts. Ces Probe Sets étant non spécifiques, ils doivent être écartés de l'analyse. Les trois collections Affymetrix représentées dans ce tableau sont la HG-U95av2, la HG-U133A et la HG-U133+2.0. Ces différentes collections représentent chronologiquement l'évolution des puces transcriptomes humaines d'Affymetrix qui sont fonction de l'état des connaissances du génome humain du moment. Ainsi la puce HG-U95av2 est la plus ancienne, et HG-U133+2.0 la plus récente.

U95av2	U133A	U133+2.0
11.9 %	7.4 %	8.4 %

TABLE C.1 – Pourcentage des sondes artificiellement dupliquées pour 3 collections différentes.

Deuxièmement, différents Probe Sets peuvent correspondre au même gène. Le tableau C.2 représente pour un certain nombre d'occurrences k (k sondes différentes parmi les sondes de la puce correspondent au même gène) le nombre de sondes différentes concernées n_k. Le Probe Set qui intéresse le plus l'expérimentateur sera celui qui cible une séquence la plus proche possible de l'extrémité 3' de l'ARN messager du gène ciblé (ceci pour des raisons techniques inhérentes à la préparation des échantillons). Au niveau génomique, en fonction de l'orientation du gène (brin sens ou antisens), on choisira donc le Probe Set le plus proche de l'extrémité 3' de l'ARN messager et dans la même orientation que ce dernier. On dira donc que les Probe Sets les plus éloignés de l'extrémité 3' sont de moins bonne qualité. Ce renseignement est disponible, puce par puce, dans la base de données de l'UCSC et permet donc de sélectionner pour chaque gène le meilleur Probe Sets.

k	1	2	3	4	5	6	7	8	9	10	12	17	20
n_k	6556	1562	558	130	42	41	17	8	11	2	2	1	1

k	1	2	3	4	5	6	7	8	9	10	11	12	13	16	17	21	22
n_k	8040	2803	1301	485	167	111	26	19	22	7	2	10	1	2	1	1	1

k	1	2	3	4	5	6	7	8	9	10	11	12	13
n_k	8925	5309	2894	1493	717	416	164	89	59	33	11	20	7

k	14	15	16	18	22	23	24	27	30
n_k	3	6	3	3	4	1	2	1	1

TABLE C.2 – Occurence des SYMBOL

Troisièmement, la base de données publique KEGG contient une version de l'ensemble des gènes impliqués dans des processus biologiques liés au développement d'un cancer. Cet ensemble, appelé dans KEGG « pathway hsa05200 » est accompagné d'interactions documentées dans la littérature et représentées sous la forme d'un Réseau de Régulations Génétiques. Après avoir déterminé les Probe Sets correspondant aux gènes impliqués dans le RRG du cancer de KEGG, les Probe Sets correspondant à des sondes de mauvaise qualité ayant préalablement été éliminées, les gènes représentés par plusieurs Probe Sets sont beaucoup moins nombreux, comme en atteste le tableau C.3. Ces Probe Sets sont en fait des doublons qui ciblent des séquences à égale distance de l'extrémité 3' de l'ARN messager.

k	1	2	3	k	1	2	3	k	1	2	3
n_k	256	25	1	n_k	290	22	2	n_k	303	19	3

TABLE C.3 – Occurence des SYMBOL sur les Probe Sets présents dans le pathway KEGG du cancer chez l'être humain `hsa05200`.

Enfin, les profils d'expressions correspondant au doublons seront moyennés. Après ces différentes étapes, la fonction faisant la correspondance entre les noms des gènes et les colonnes du jeu de données de puces à ADN considéré est injective.

Annexe D

Analyse de données de toxicogénomique

La toxigénomique est l'ensemble des techniques qui permettent d'étudier l'effet d'une ou plusieurs substances sur un organisme modèle et plus particulièrement en terme de réseau de régulation, d'expression des gènes ou activité des protéines. L'organisme en question peut être par exemple la souris (*Mus musculus*) ou le rat (*Rattus norvegicus*) (entre autres) ou des cellules primaires extraites de tissus humains sains (par exemple des cellules du foie). Différentes doses de la substance toxique sont administrées à l'organisme modèle. Le but de certaines études menées au LEFG a été d'étudier l'effet de ces doses grâce à des puces à ADN.

Cependant, certaines doses ont un effet observable équivalent qui n'est pas prévisible par l'expérimentateur. Nous proposons une méthode qui permet de regrouper les doses ayant le même effet observable. Cette méthode permet de plus d'associer au problème de classification des profils d'expression en fonctions des doses

- une signature moléculaire permettant d'interagir avec les biologistes,
- un pouvoir de prédiction.

La partie suivante est la retranscription de l'article qui a été accepté à la conférence BIOTECHNO 2008 [Guillemot et al., 2008b].

The range of doses chosen in toxicogenomics studies does not always represent all the possible effects on gene expression : several doses of toxicant can lead to the same observable effect on the transcriptome. This makes the problem of dose exposure prediction difficult to address. We propose a strategy allowing to gather the doses with similar effects prior to the computing of a molecular signature. The different gatherings of doses are compared with criteria based on likelihood or Monte Carlo Cross Validation. The molecular signature is then determined via a voting algorithm. Experimental results point out that the obtained classifier has better prediction performances than the classifier computed according to the original labeling.

D.1 Introduction

Microarray gene-expression profiling is recognized to bring valuable information as regards diagnosis or prognosis (e.g. oncology, new drugs testing, etc.). Many works now aim at applying this high-throughput tool to toxicological studies [Bushel et al., 2007], [Fannin et al., 2005], for which the ultimate purposes are to know whether an individual has been intoxicated and if so, to identify the toxicant and possibly to predict the exposure level. Because the clinical signs are the same for a wide range of toxicants, a molecular imprint yielded by gene expression, the so called *molecular signature*, of each toxicant would help the design of a fast and efficient diagnostic tool. A typical toxicogenomic study consists in administering a toxicant to a model organism at different doses within a range, and getting the corresponding gene expression data. We address specifically here the determination of a molecular signature for dose exposure prediction in the case of a unique toxicant.

Some Machine Learning algorithms are dedicated to finding the molecular signature from gene expression. They have to be used cautiously to provide reliable results. The genes of the signature are determined in a cross-validation framework in order to limit the risk of over learning bias, as recommended in recent works [Ambroise and McLachlan, 2002, Michiels et al., 2005, Dupuy and Simon, 2007].

In this work, we consider as mandatory to explore the possibility that different doses have the same effect on the expression levels of genes. To perform this task, we propose two methods to apply prior to the determination of the molecular signature : one is a likelihood-based method and the other, introduced here, is based on the Monte Carlo Cross Validation (MCCV) algorithm. Then, once the doses of toxicant with similar effect are gathered, a machine learning algorithm is run, aiming at determining a molecular signature and its predictive power. Our approach is applied to two toxicogenomic studies. The obtained results demonstrate the possibility to gather similar doses, and the interest of this grouping in order to estimate a corresponding classifier with better prediction performances than the one related to the original labelling of doses.

The paper is organized as follows : in Section 2 the selection of an optimal partition thanks to the likelihood-based and the MCCV methods is detailed, then, the use of this optimal partition in a classification algorithm to compute a molecular signature is

presented ; in Section 3 the proposed methods are applied to two real toxicogenomic datasets.

D.2 Methods

We present here the notations adopted throughout this paper.

- n is the number of observations, g is the number of variables (genes), $g \gg n$ and N is the number of doses
- X is the $n \times g$ real matrix containing all the expression profiles
- x is a $g \times 1$ observation (or individual)
- $P_{orig} = (1, ..., N)$ is the vector of the N administered doses labels and is called the original partition.
- y is the $n \times 1$ vector of the administered dose of the toxicant for each observation. Each element of y takes its value in P_{orig}
- $P = (P^{(1)}, ..., P^{(N)})$ is a new labeling of the doses, consisting in a permutation with repetitions of the N elements of P_{orig} among its first $K < N$ elements : $(P^{(i)} \in \{1, ..., K\}$ with $i = 1, ..., N)$. $P^{(i)}$ is the new class label replacing the class label i in P_{orig}. Concretely, $P^{(1)}$ is the class label of the lowest dose and $P^{(N)}$ the class label of the highest dose. P is called a new partition
- z is the new vector of the observable doses, deduced from a new partition P
- g_{filt} is the number of genes kept after each filtering step (see figure D.1)
- g_s is the number of genes kept in the molecular signature

To clarify the notion of partition, let us consider an example of a five dose exposure experiment. The initial partition is $P_{orig} = (1, 2, 3, 4, 5)$ and $y = (1, 1, 2, 2, 3, 3, 4, 4, 5, 5)$ is the vector describing the class of each observation (two observations per dose). If a partition P consists in aggregating the weakest doses $P^{(1)}_{orig}$ and $P^{(2)}_{orig}$, it is noted $P = (1, 1, 2, 3, 4)$; the new vector of classes is $z = (1, 1, 1, 1, 2, 2, 3, 3, 4, 4)$. It is worth noticing that P could be indifferently noted $P = (1, 1, 2, 4, 3)$, or $P = (2, 2, 1, 3, 4)$. Finally, the partition $P_1 = (1, 1, 2, 3, 4)$ aggregates the weakest doses $P^{(1)}_{orig}$ and $P^{(2)}_{orig}$, whereas the partition $P_2 = (1, 2, 3, 4, 1)$ the weakest dose $P^{(1)}_{orig}$ with the strongest dose $P^{(5)}_{orig}$.

We present thereafter two different methods used to estimate a partition $\widehat{P}$ which describes the observable effects of the toxicant in the dataset X (see paragraphs D.2 and D.2). Then, given $\widehat{P}$, we classically determine a molecular signature and the test error rate (see paragraph D.2).

Best partition in a Classification Likelihood sense Let observation x be drawn from a multivariate mixture density :

$$f(x, \Theta) = \sum_{k=1}^{K} p_k f_k(x, \theta_k) \tag{D.1}$$

$\Theta = (p_k; \theta_k)_{k=1,...,K}$, p_k is the probability for an observation to be in the class k and θ_k is the parameter vector of f_k. The choice of a mixture model allows to derive a

Classification Log-Likelihood (CLL), as already proposed by [Bryant, 1991]. Let P be the current partition. This leads to a partition of the sample $X = [x_1; ...; x_n]$ into K classes $C_k, k = 1, ..., K$.

$$\mathcal{L}_c(X, P, \Theta) = \sum_{k=1}^{K} \sum_{x \in C_k} \log\left(p_k f_k(x|\theta_k)\right) \tag{D.2}$$

$$\mathcal{L}_c(X, P) = \sum_{k=1}^{K} \sum_{x \in C_k} \log\left(p_k f_k(x)\right) \tag{D.3}$$

Equation (D.2) is commonly used within the Classification Expectation-Maximization algorithm (CEM) [Celeux and Govaert, 1992]. We propose to use a Bayesian Information Criterion (BIC) based on the CLL (equation (D.4)) which will characterize the quality of P while taking into account the complexity of the corresponding mixture model.

$$BIC = -2\mathcal{L}_c + \nu \log(n) \tag{D.4}$$

where $\nu = gK + 1$ is a parameter depending on the complexity of the model, assuming that the observations follow a Gaussian mixture model.

The results of the BIC approach allow the selection of a presumably optimal partition. Yet, two characteristics of this criterion are debatable when considering the final objective of molecular signature finding :

- the BIC value does not have any signification, especially if one wants to characterize the test error rate associated to the partition P
- it depends strongly on the Gaussian mixture assumption

In the next section, a prediction model and its cross-validated error rate are computed for each partition. We investigate whether the partitions proposed by the BIC approach provide the smallest test error rates.

Best partition in an MCCV sense Some papers dealing with discrimination from microarray data have been severely criticized in recent works [Ambroise and McLachlan, 2002, Dupuy and Simon, 2007, Michiels et al., 2005]. For instance, Michiels *et al.* [Michiels et al., 2005] emphasizes the fact that numerous papers use methodology resulting in an overoptimistic estimation of the error rate. The approach proposed in this paper was designed to meet the quality requirements suggested in [Michiels et al., 2005] and advocates the use of validation by repeated random sampling, leading to an accurate methodology (cf. figure D.1) to get both a molecular signature and the discrimination model associated to a test error rate.

To obtain a robust estimation of the test error rate, the learning phase is embedded in a cross-validation framework, presented by figure D.1.

For each partition P, the test error rate evaluation is a 3-steps MCCV algorithm consisting in repeating B times :

(a) a split step : split randomly the dataset in a training and a testing set, respecting a 2 :1 ratio.

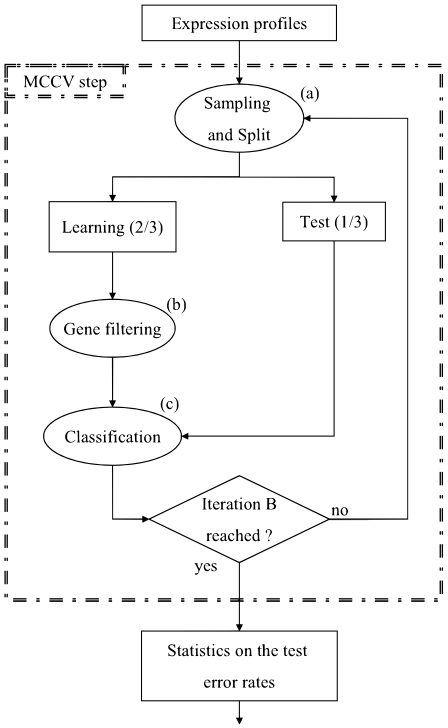


FIGURE D.1 – MCCV Algorithm, P being fixed.

(b) a filtering step : select the g_{filt} relevant variables from the learning dataset via a K-sample F-test with a Bayesian regularization [Smyth, 2004].

(c) a classification step : compute a prediction function. To deal with the $g \gg n$ setting, linear Support Vector Machines (SVM) [Cortes and Vapnik, 1995] were used to build the classifier. When a multiclass situation is encountered, a *One versus One* strategy is applied [Allwein et al., 2000]. The regularization parameter of the classifier is determined on the learning set by a Leave One Out Cross Validation technique (not shown on figure D.1).

g_{filt} is set to 200 for all P and MCCV iterations. Additional runs (not depicted in this paper) showed that g_{filt} has no influence on the ranking of P, it was chosen small enough with regard to g to significantly reduce the number of variables. B is set to 50, allowing the estimation of the mean test error rate. Each step of this MCCV algorithm respects the recommendations suggested in [Boulesteix et al., 2008].

To achieve the grouping of doses with same observable effect, we choose the partition $\widehat{P}$ as a trade-off between minimizing the BIC and the test error rate. Knowing $\widehat{P}$, we can estimate a classifier to discriminate between doses with observable effect.

Signature and classifier Given $P = \widehat{P}$, we compute a molecular signature of g_s genes. Between two MCCV iterations, the g_{filt} genes selected are not likely to be the same : it strongly depends on the split step D.1. Thus, the B lists of variables provided

TABLE D.1 – Description of the datasets

	Toxicant	
	Ricin (Tox_1)	Mustard Gas (Tox_2)
Organism	*Mus musculus*	*Rattus norvegicus*
Biological tissue	total blood	lung
# doses	5	4
Dosage	0, 1, 2, 4, 6 μg/kg	0, 1, 3, 6 mg/kg
# samples / dose	10, 7, 7, 9, 7	20, 10, 10, 11
# variables	24111	15923
# partitions	43	14

by the MCCV are very heterogeneous and to obtain a consensus list, a voting method is required.

We consider two voting techniques :

– "Unanimity" : the g_s genes which are selected unanimously by all the B iterations of the MCCV procedure

– "Quorum" : genes are sorted according to the number of occurrences in the B iterations. The first g_s genes of this sorting are selected.

Once the signature is determined, we finally build the classifier from the whole dataset and test it on an unseen set of observations, leading to an estimation of the generalization error associated to the g_s signature genes.

D.3 Results

Description of the datasets We applied our approach to two toxicogenomic datasets Tox_1 and Tox_2, described in table D.1. Tox_1 corresponds to an in-house experiment (not yet published data), and Tox_2 includes data described in [Dillman et al., 2005] available on the GEO repository (GSE1888). The animals have been sacrificed and messenger RNA (mRNA) has been extracted from the appropriate biological tissue according to usual protocols. The obtained samples have been hybridized on microarrays. For both cases the control sample consists in mRNA from animals injected with the vehicle of the toxicant.

Following classical experimental plans, the two selected experiments are designed as follows : among the doses of toxicants injected to the animals, at least one has known effects, for instance, the dose for which 50% of the exposed animals die (Lethal Dose 50, LD50). Exposure to all these doses can have very different effects on the tissue under study, with possibly no visible phenotype. As proposed earlier, the issue is then to cluster the doses which have the same effects on gene expression.

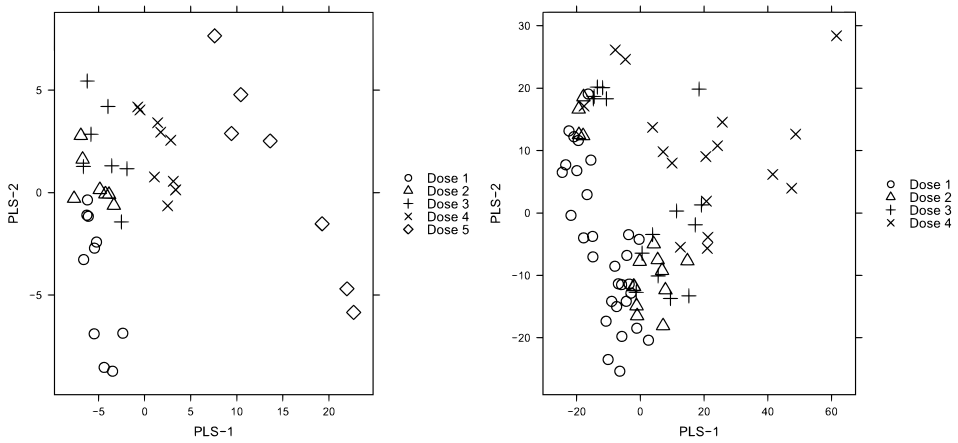
Figure D.2(a) and D.2(b) depict the projection of the n observations onto the 2-dimensional space spanned by the two first components of the Partial Least Squares

Regression (PLS-R) [Wold et al., 1983] of y on X. Symbols represent the class membership.

For Tox_1, we point out a clear linear discrimination between low and high doses. Samples corresponding to doses 2 and 3 seems to be quite similar in a transcriptomic point of view and can be considered as belonging to the same class (say the low level exposure class). Samples associated with doses 1, 4 and 5 constitute the null, medium and high level exposure classes respectively. The corresponding partition is $P = (1, 2, 2, 3, 4)$.

For Tox_2, we conjecture that samples corresponding to doses 2 and 3 are quite similar and can be considered as belonging to the same class (say the medium level exposure class) whereas samples associated with doses 1 and 4 constitute two other distinct classes, say the null level exposure class and the high level exposure class. The corresponding partition is $P = (1, 2, 2, 3)$.

Those remarks illustrate the need to formalize a way to characterize the "optimal" class structure by clustering the doses with the same observable effect.



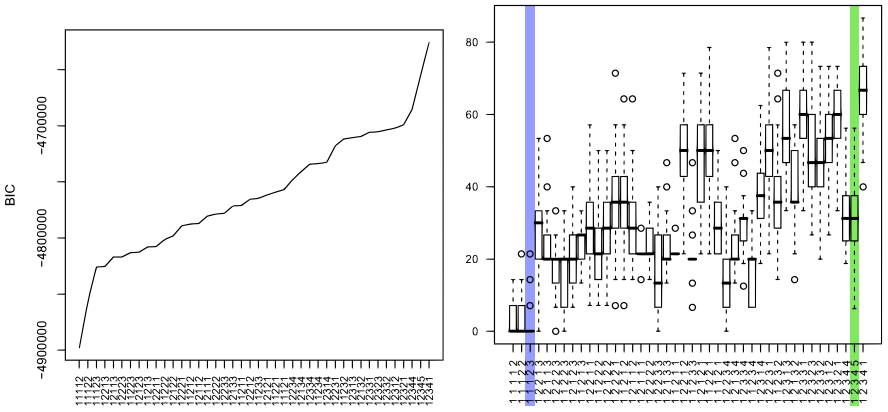
(a) Projection of the Tox_1 observations on the two first components of a PLS-R of y on X.

(b) Projection of the Tox_2 observations on the two first components of a PLS-R of y on X.

FIGURE D.2 – PLS-R projections of Tox_1 and Tox_2 observations.

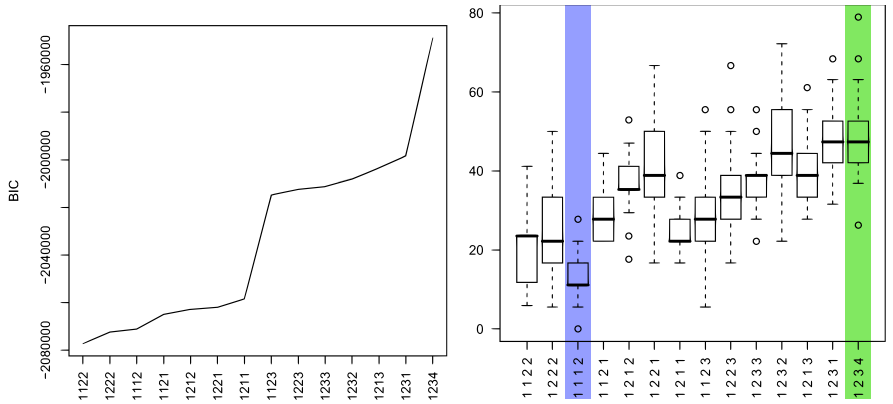
Choice of $\widehat{P}$ For each P, the BIC is estimated. Figures D.3(a) and D.3(c) depict BIC as a function of P for each dataset. BIC declared as optimal respectively the partitions $(1, 1, 1, 1, 2)$ and $(1, 1, 2, 2)$. The partitions suggested at the end of paragraph D.3 are ranked respectively 26^{th} out of 43 and 9^{th} out of 14 by BIC. We then compared the BIC and classification results in figures D.3(b) and D.3(d). As expected, the test error rates associated with the partitions top-ranked by BIC are the smallest. Yet the partitions corresponding to the lowest test error rates are $(1, 1, 1, 2, 3)$ for Tox_1 and $(1, 1, 1, 2)$ for Tox_2. Moreover these partitions are biologically interesting : from a range of 5 doses, we are able to deduce a range of 3 observable effects as regards gene expression for Tox_1 and from a range of 4 doses, to deduce a range of 2 observable effects for Tox_2.

In the Tox_1 case, this new partition is all the more interesting because it keeps apart samples associated with the LD50 from the non lethal doses.



(a) BIC as a function of P.

(b) Test error rates as a function of P : both $\widehat{P}$ and the original partition are highlighted.



(c) BIC as a function of P.

(d) Test error rates as a function of P : both $\widehat{P}$ and the original partition are highlighted.

FIGURE D.3 – BIC and test error rates for each partition for Tox_2.

Classification knowing $\widehat{P}$ Finally, we give the generalization error rates, as a function of g_{filt} and g_s, for $\widehat{P}$ in tables D.2(c) and D.2(d), obtained from 13 unseen observations for Tox_1 (resp. 50 for Tox_2). We set g_{filt} and g_s to sufficiently small values to significantly reduce the number of variables after the filtering step and to allow the biological validation of the signature.

It is worth pointing out that the number of genes kept by the filtering step has a non negligible effect on the determination of a consensus list of genes, although it has

TABLE D.2 – Generalization error rates.

(a) Unanimity for Tox_1.

	Number of filtered genes g_{filt}				
	25	50	100	150	200
Error rate	31%	23%	31%	31%	31%
g_s	12	17	39	66	80

(b) Unanimity for Tox_2.

	Number of filtered genes g_{filt}				
	25	50	100	150	200
Error rate			19%	15%	19%
g_s	0	0	1	4	2

(c) Quorum for Tox_1.

	Number of filtered genes g_{filt}				
g_s	25	50	100	150	200
5	54%	31%	23%	31%	31%
10	38%	23%	31%	38%	31%
15	15%	23%	31%	31%	31%
20	46%	62%	31%	31%	31%
25	23%	23%	23%	31%	31%
30	23%	23%	38%	38%	31%
40	23%	31%	23%	23%	38%

(d) Quorum for Tox_2.

	Number of filtered genes g_{filt}				
g_s	25	50	100	150	200
5	23%	23%	15%	12%	23%
10	19%	19%	23%	23%	27%
15	23%	23%	19%	23%	15%
20	35%	23%	27%	23%	12%
25	15%	27%	31%	23%	15%
30	15%	19%	15%	23%	19%
40	23%	15%	8%	8%	19%

no effect on the ranking of the partitions.

There is an optimal value (15%) when $g_s = 15$ for Tox_1. These genes are present in each signature of length greater than 15. The lowest generalization error rate for Tox_2 is 8% and is associated to a signature with $g_s = 40$. Our method provided both a classifier able to predict the dose exposure of a new observation and the best subset of genes in terms of prediction.

For both datasets, the original partitions $(1, 2, 3, 4, 5)$ and $(1, 2, 3, 4)$ show poor performances either according to their BIC ranking, to their ranking with the MCCV procedure (see figures D.3 and D.3). The associated generalization error rates are 37% and 38% respectively.

D.4 Conclusion

In the framework of toxicogenomics, studies aim at determining the molecular signature of a given toxicant from a tissue sample. We propose a two-fold methodology to be applied to usual dose-range gene expression experiments, consisting in : first, the discovery of sets of doses with the same observable expression effect and second, the determination of the molecular signature using a MCCV approach. The results presented on two datasets show the impact of the preliminary step on the generalization error. The results presented on two datasets show that gathering similar doses yields a classifier with better prediction performances than the one related to the original range of doses.

0. On the x axis, partitions are ranked according to increasing values of BIC.

Future work will focus on alternative methods to filter variables [Krishnapuram et al., 2004] and on the automatic selection of the best partitions. A special care will be given to the use of contingency tables rather than test and generalization error rate in order to better account for multiclass discrimination.

Notations

X	variable aléatoire réelle
$\boldsymbol{X}$	variable aléatoire réelle multivariée à p composantes
x	vecteur de n réalisations de la variable aléatoire X
$\boldsymbol{x}$	matrice, de taille $n \times p$, contenant n réalisations de la variable $\boldsymbol{X}$
$\boldsymbol{x}_{i.}$	ligne i de la matrice $\boldsymbol{x}$
$\boldsymbol{x}_{.j}$	colonne j de la matrice $\boldsymbol{x}$
$.^\top$	opérateur de transposition
cov	covariance

Glossaire

Le tableau suivant rassemble tous les acronymes utilisés dans le rapport.

Notation	Description
k-fold	*k*-fold Cross Validation. 20, 21, 30, 79
ADN	Acide désoxyribonucléique. 7, 9–11, 32, 33, 77, 78, 83, 87, 89, 93
ARN	Acide ribonucléique. 33, 93
DDA	Diagonal Discriminant Analysis. 44
FC	Fold Change. 23, 28, 30, 83, 89
FDA	Fisher Discriminant Analysis. 39
FDR	False Discovery Rate. 65–67, 71, 78
gCDA	Graph constrained Discriminant Analysis. 9, 39, 46, 48–53, 79, 83–85
GGM	Gaussian Graphical Models. 39, 44, 46, 57
KEGG	Kyoto Encyclopedia of Genes and Genomes, `http://www.genome.jp/kegg/`. 77
LASSO	Least Angle Regression SSO. 15, 16, 18, 27, 36, 58, 71
LEFG	Laboratoire d'Exploration Fonctionnelle des Génomes, ancien laboratoire de l'iRCM, institut de Radiobologie cellulaire et moléculaire.. 87, 93
LP-SVM	Linear Programming Support Vector Machines [Bradley and Mangasarian, 1998]. 9, 20, 27, 38, 49, 50, 79
MCCV	Monte Carlo Cross Validation. 20, 21, 49
MSE	Mean Square Error. 15, 16, 44

Notation	Description
NB-SVM	Network Based Support Vector Machines. 37, 38, 49–51, 79
Net	Méthode de [Li and Li, 2008]. 37, 38, 49, 50, 79
OLS	Ordinary Least Squares. 14, 16, 59, 60
PCR	Polymerase Chain Reaction. 11
PLS-PC	Partial Least Squares Regression - Partial Correlation [Tenenhaus et al., 2008]. 64, 66, 67
PLS-R	Partial Least Squares Regression. 15, 17, 18, 23, 27, 57, 58, 63, 64, 71, 84
PPV	Positive Predictive Value. 69–71, 73, 84
RDA	Regularized Discriminant Analysis. 44
RE	Ratio d'expression. 23, 24
ROC	Receiving Operator Characteristic. 18, 28
RP	Rank Products. 24
RR	Régression Ridge. 15, 16, 27, 53, 57, 58, 67, 70, 71, 79
RRG	Réseau de régulations génétiques. 7, 9, 33, 35, 39, 49, 57, 67, 71, 78, 79, 83, 84
SVM	Support Vector Machines. 9, 14, 18–20, 27, 36
UCSC	http://genome.ucsc.edu/. 103, 104

Références bibliographiques

[AffyTeam, 2005] AffyTeam (2005). Guide to probe logarithmic intensity error (plier) estimation. Technical report, Affymetrix.

[Allwein et al., 2000] Allwein, E. L., Schapire, R. E., and Singer, Y. (2000). Reducing multiclass to binary : A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1 :113–141.

[Alon et al., 1999] Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., and Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci U S A*, 96(12) :6745–6750.

[Ambroise and McLachlan, 2002] Ambroise, C. and McLachlan, G. J. (2002). Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc Natl Acad Sci U S A*, 99(10) :6562–6566.

[Anderson, 2003] Anderson, T. W. (2003). *An introduction to multivariate statistical analysis*. Wiley series in probability and statistics.

[Binder and Schumacher, 2009] Binder, H. and Schumacher, M. (2009). Incorporating pathway information into boosting estimation of high-dimensional risk prediction models. *BMC Bioinformatics*, 10 :18.

[Boulesteix and Strimmer, 2005] Boulesteix, A.-L. and Strimmer, K. (2005). Predicting transcription factor activities from combined analysis of microarray and ChIP data : a partial least squares approach. *Theor Biol Med Model*, 2 :23.

[Boulesteix et al., 2008] Boulesteix, A.-L., Strobl, C., Augustin, T., and Daumer, M. (2008). Evaluating microarray-based classifiers : An overview. *Cancer Inform*, 6 :77–97.

[Bradley and Mangasarian, 1998] Bradley, P. S. and Mangasarian, O. L. (1998). Feature selection via concave minimization and support vector machines. In *Machine Learning Proceedings of the Fifteenth International Conference (ICML'98)*, pages 82–90.

[Brazhnik et al., 2002] Brazhnik, P., de la Fuente, A., and Mendes, P. (2002). Gene networks : how to put the function in genomics. *Trends Biotechnol*, 20(11) :467–472.

[Breitling et al., 2004] Breitling, R., Armengaud, P., Amtmann, A., and Herzyk, P. (2004). Rank products : a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS Lett*, 573(1-3) :83–92.

[Brigand et al., 2006] Brigand, K. L., Russell, R., Moreilhon, C., Rouillard, J.-M., Jost, B., Amiot, F., Magnone, V., Bole-Feysot, C., Rostagno, P., Virolle, V., Defamie, V., Dessen, P., Williams, G., Lyons, P., Rios, G., Mari, B., Gulari, E., Kastner, P., Gidrol, X., Freeman, T. C., and Barbry, P. (2006). An open-access long oligonucleotide microarray resource for analysis of the human and mouse transcriptomes. *Nucleic Acids Res*, 34(12) :87.

[Bryant, 1991] Bryant, P. G. (1991). Large-sample results for optimization-based clustering methods. *Journal of Classification*, 8(1) :31–44.

[Burges, 1998] Burges, J. C. (1998). A tutorial on support vector machines for pattern recognition by christopher. *Data Mining and Knowledge Discovery*, 2 :121–167.

[Bushel et al., 2007] Bushel, P. R., Heinloth, A. N., Li, J., Huang, L., Chou, J. W., Boorman, G. A., Malarkey, D. E., Houle, C. D., Ward, S. M., Wilson, R. E., Fannin, R. D., Russo, M. W., Watkins, P. B., Tennant, R. W., and Paules, R. S. (2007). Blood gene expression signatures predict exposure levels. *Proc Natl Acad Sci U S A*, 104(46) :18211–18216.

[Celeux and Govaert, 1992] Celeux, G. and Govaert, G. (1992). A classification em algorithm for clustering and two stochastic versions. *Computational statistics and data analysis*, 14 :315–332.

[Chandran et al., 2007] Chandran, U. R., Ma, C., Dhir, R., Bisceglia, M., Lyons-Weiler, M., Liang, W., Michalopoulos, G., Becich, M., and Monzon, F. A. (2007). Gene expression profiles of prostate cancer reveal involvement of multiple molecular pathways in the metastatic process. *BMC Cancer*, 7 :64.

[Cornuéjols et al., 2002] Cornuéjols, A., Miclet, L., Kodratoff, Y., and Mitchell, T. (2002). *Apprentissage artificiel : Concepts et algorithmes*. Eyrolles.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20 :273–297.

[Dagnelie, 1999] Dagnelie, P. (1999). *Analyse statistique à plusieurs variables*. Les presses agronomiques de Gembloux, A. S. B. L.

[Dillman et al., 2005] Dillman, J. F., Phillips, C. S., Dorsch, L. M., Croxton, M. D., Hege, A. I., Sylvester, A. J., Moran, T. S., and Sciuto, A. M. (2005). Genomic analysis of rodent pulmonary tissue following bis-(2-chloroethyl) sulfide exposure. *Chem Res Toxicol*, 18(1) :28–34.

[Dimitriadou et al., 2006] Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., , and Weingessel, A. (2006). *e1071 : Misc Functions of the Department of Statistics (e1071), TU Wien*. R package version 1.5-16.

[Dudoit et al., 2002] Dudoit, S., Fridlyand, J., and Speed, T. P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97 :77–87.

[Dupuy and Simon, 2007] Dupuy, A. and Simon, R. M. (2007). Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting. *J Natl Cancer Inst*, 99(2) :147–157.

[Durbin and al., 2002] Durbin and al. (2002). A variance-stabilizing tansformation for gene-expression microarray data. *Bioinformatics*, 18 :105–110.

[Efron, 2005a] Efron, B. (2005a). Large-scale simultaneous hypothesis testing : the choice of a null hypothesis. *Journal of the American Statistical Association*, 99 :96–104.

[Efron, 2005b] Efron, B. (2005b). Local false discovery rates. Technical report, Department of Statistics, Stanford University.

[Efron et al., 2002] Efron, B., Hastie, T., Johnstone, L., and Tibshirani, R. (2002). Least angle regression. *Annals of Statistics*, 32 (2) :407–499.

[Erdős and Rényi, 1959] Erdős, P. and Rényi, A. (1959). On random graphs. *Publicationes Mathematicæ*, 6 :290–297.

[Fannin et al., 2005] Fannin, R. D., Auman, J. T., Bruno, M. E., Sieber, S. O., Ward, S. M., Tucker, C. J., Merrick, B. A., and Paules, R. S. (2005). Differential gene expression profiling in whole blood during acute systemic inflammation in lipopolysaccharide-treated rats. *Physiol Genomics*, 21(1) :92–104.

[Faurre, 1988] Faurre, P. (1988). *Analyse numérique : notes d'optimisation*. Ellipses.

[Fisher, 1936] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7 :179–188.

[Freudenberger, 2005] Freudenberger, J. M. (2005). *Comparison of background correction and normalization procedures for high-density oligonucleotide microarrays*. PhD thesis, Universität Leipzig.

[Friedman, 1998] Friedman, J. (1998). Regularized discriminant analysis. *Journal of the American Statististical Association*, 84 :165.

[Friedman et al., 2008] Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3) :432–441.

[Golub et al., 1999] Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D., and Lander, E. S. (1999). Molecular classification of cancer : class discovery and class prediction by gene expression monitoring. *Science*, 286(5439) :531–537.

[Guillemot et al., 2008a] Guillemot, V., Le Brusquet, L., Tenenhaus, A., and Frouin, V. (2008a). Graph-constrained discriminant analysis of functional genomics data. In *Proc. IEEE International Conference on Bioinformatics and Biomeidcine Workshops BIBMW 2008*, pages 207–210.

[Guillemot et al., 2008b] Guillemot, V., Philippe, C., Tenenhaus, A., Rollin, J., Gidrol, X., and Frouin, V. (2008b). Grouping levels of exposure with same observable effects before class prediction in toxicogenomics. (2008). In *Proc. of the International Conference on Biocomputation, Bioinformatics, and Biomedical Technologies*, pages 164–169.

[Guillemot et al., 2009] Guillemot, V., Tenenhaus, A., and Frouin, V. (2009). Une statistique de sphéricité pour l'adéquation d'un graphe à des données transcriptomiques. In *Poster présenté lors des Journées Ouvertes en Biologie, Informatique et Mathématiques (JOBIM) 2009*.

[Guillot et al., 2007] Guillot, G., Olsson, M., Benson, M., and Rudemo, M. (2007). Discrimination and scoring using small sets of genes for two-sample microarray data. *Math Biosci*, 205(2) :195 –203.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3 :1157–1182.

[Haasdonk and P ekalska, 2008] Haasdonk, B. and P ekalska, E. (2008). Classification with kernel mahalanobis distance classifiers. In *Proc. of the German Classification Society Annual Conference*.

[Hand, 2006] Hand, D. J. (2006). Classifier technology and the illusion of progress. *Statistical Science*, 21(1) :1–15.

[Helland, 1988] Helland, I. S. (1988). On the structure of Partial Least Squares regression. *Communications in Statistics Simulation and Computation*, 17 :581–607.

[Heride et al., 2010] Heride, C., Ricoul, M., Hase, J. V., Kiêu, K., Guillemot, V., Cremer, C., Dubrana, K., and Sabatier, L. (2010). Distance between homologous chromosomes results from chromosom positioning constraints. *to be submitted*.

[Hoerl and Kennard, 1970] Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression : biased estimation for nonorthogonal problems. *Technometrics*, 12 :55–77.

[Höskuldsson, 1988] Höskuldsson, A. (1988). PLS regression methods. *Journal of Chemometrics*, 2 :211–228.

[Irizarry et al., 2003] Irizarry, R. A., Bolstad, B. M., Collin, F., Cope, L. M., Hobbs, B., and Speed, T. P. (2003). Summaries of Affymetrix GeneChip probe level data. *Nucleic Acids Res*, 31(4) :15.

[Irizarry et al., 2006] Irizarry, R. A., Wu, Z., and Jaffee, H. A. (2006). Comparison of Affymetrix GeneChip expression measures. *Bioinformatics*, 22(7) :789–794.

[Krämer et al., 2009] Krämer, N., Schäfer, J., and Boulesteix, A.-L. (2009). Regularized estimation of large scale gene association networks using gaussian graphical models. *BMC Bioinformatics*, 10 :384.

[Krishnapuram et al., 2004] Krishnapuram, B., Carin, L., and Hartemink, A. (2004). *Gene expression analysis : Joint feature selection and classifier design*, chapter 14, pages 299–318. MIT press.

[Lasko et al., 2005] Lasko, T. A., Bhagwat, J. G., Zou, K. H., and Ohno-Machado, L. (2005). The use of receiver operating characteristic curves in biomedical informatics. *J Biomed Inform*, 38(5) :404–415.

[Ledoit and Wolf, 2002] Ledoit, O. and Wolf, M. (2002). Some hypothesis tests for the covariance matrix when the dimension is large compared to the sample size. *The Annals of Statistics*, 30 :1081–1102.

[Lee et al., 2008] Lee, E.-S., Son, D.-S., Kim, S.-H., Lee, J., Jo, J., Han, J., Kim, H., Lee, H. J., Choi, H. Y., Jung, Y., Park, M., Lim, Y. S., Kim, K., Shim, Y., Kim, B. C., Lee, K., Huh, N., Ko, C., Park, K., Lee, J. W., Choi, Y. S., and Kim, J. (2008). Prediction of recurrence-free survival in postoperative non-small cell lung cancer patients by

using an integrated model of clinical information and gene expression. *Clin Cancer Res*, 14(22) :7397–7404.

[Lemon et al., 2002] Lemon, W. J., Palatini, J. J. T., Krahe, R., and Wright, F. A. (2002). Theoretical and experimental comparisons of gene expression indexes for oligonucleotide arrays. *Bioinformatics*, 18(11) :1470–1476.

[Letac and Massam, 2007] Letac, G. and Massam, H. (2007). Wishart distribution for decomposable graphs. *Annals of Statistics*, 35(3) :1278–1323.

[Li and Li, 2008] Li, C. and Li, H. (2008). Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9) :1175–1182.

[Li and Wong, 2001] Li, C. and Wong, W. H. (2001). Model-based analysis of oligonucleotide arrays : expression index computation and outlier detection. *Proc Natl Acad Sci U S A*, 98(1) :31–36.

[Lu et al., 2005] Lu, Y., Liu, P.-Y., Xiao, P., and Deng, H.-W. (2005). Hotelling's T2 multivariate profiling for detecting differential expression in microarrays. *Bioinformatics*, 21(14) :3105–3113.

[Lukaszewicz et al., 2007] Lukaszewicz, A., Payen, D., Faivre, V., Megarbane, B., Fieux, F., Azoulay, E., Baulande, S., Guillemot, V., Tenenhaus, A., and Soularue, P. (2007). Canonical equation model of white cells gene expression for rapid (hours) detection of high risk of death at day0 of human septic shock. *Critical Care Medicine -Baltimore-*, 35(12) :A255.

[Mann and Whitney, 1947] Mann, H. B. and Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Ann. Math. Statist.*, 18(1) :50–60.

[Manne, 1987] Manne, R. (1987). Analysis of Two Partial Least Squares Algorithms for Multivariate Calibration. *Chemometrics and Intelligent Laboratory Systems*, 2 :187–197.

[Margolin et al., 2006] Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Favera, R. D., and Califano, A. (2006). Aracne : an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7 (Suppl. 1) :S7.

[Meinshausen and Bühlman, 2006] Meinshausen, N. and Bühlman, P. (2006). High dimensional graphs and variable selection with the lasso. *Annals of Statistics*, 34 :1436–1462.

[Michiels et al., 2005] Michiels, S., Koscielny, S., and Hill, C. (2005). Prediction of cancer outcome with microarrays : a multiple random validation strategy. *Lancet*, 365(9458) :488–492.

[Monfort, 1996] Monfort, A. (1996). *Cours de probabilités*. Economica, 3rd edition.

[Nagao, 1973] Nagao, H. (1973). On some test criteria for covariance matrix. *The Annals of Statistics*, 4 :700–709.

[Rajaratnam et al., 2008] Rajaratnam, B., Massam, H., and Carvalho, C. M. (2008). Flexible covariance estimation in graphical gaussian models. *Annals of Statistics*, 36(6) :2818–2849.

[Rapaport et al., 2007] Rapaport, F., Zinovyev, A., Dutreix, M., Barillot, E., and Vert, J.-P. (2007). Classification of microarray data using gene networks. *BMC Bioinformatics*, 8 :35.

[Saporta, 2006] Saporta, G. (2006). *Probabilités, analyse de données et statistiques*. Editions Technip.

[Schäfer and Strimmer, 2005a] Schäfer, J. and Strimmer, K. (2005a). An empirical Bayes approach to inferring large-scale gene association networks. *Bioinformatics*, 21 :754–764.

[Schäfer and Strimmer, 2005b] Schäfer, J. and Strimmer, K. (2005b). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Stat Appl Genet Mol Biol*, 4 :Issue 1, Article 32.

[Schott, 2007] Schott, J. R. (2007). A test for the equality of covariance matrices when the dimension is large relative to the sample sizes. *Computational Statistics & Data Analysis Volume*, 51 :6535–6542.

[Singh et al., 2002] Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A. A., D'Amico, A. V., Richie, J. P., Lander, E. S., Loda, M., Kantoff, P. W., Golub, T. R., and Sellers, W. R. (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2) :203–209.

[Smyth, 2004] Smyth, G. K. (2004). Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Stat Appl Genet Mol Biol*, 3 :Issue 1, Article 3.

[Speed, 2003] Speed, T. (2003). *Statistical Analysis of Gene Expression Microarray Data*. CRC Press Inc.

[Tenenhaus et al., 2008] Tenenhaus, A., Guillemot, V., Gidrol, X., and Frouin, V. (2008). Gene association networks from microarray data using a regularized estimation of partial correlation based on pls regression. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

[Tenenhaus, 2007] Tenenhaus, M. (2007). *Méthodes pour décrire, expliquer et prévoir*. Dunod, 2 edition.

[Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistics Society*, Series B 58 :267–288.

[Tucker, 1958] Tucker, L. (1958). An inter-battery method of factor analysis. *Psychometrika*, 23 :111–136.

[Vert, 2008] Vert, J. P. (2008). Reconstruction of biological networks by supervised machine learning approaches. Technical report, Mines ParisTech, Centre for Computational Biology.

[Whittaker, 1990] Whittaker, J. (1990). *Graphical models in applied multivariate statistics*. Wiley.

[Witten and Tibshirani, 2007] Witten, D. M. and Tibshirani, R. (2007). A comparison of fold-change and the t-statistic for microarray data analysis.

[Witten and Tibshirani, 2009] Witten, D. M. and Tibshirani, R. (2009). Covariance-regularized regression and classification for high dimensional problems. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 71 :615–636.

[Wold et al., 1983] Wold, S., Martens, L., and Wold, H. (1983). The multivariate calibration problem in chemistry solved by the PLS method. In *Proceedings Conf. Matrix Pencils, Ruhe A. & Kåstrøm B, Lecture Notes in Mathematics*, pages 286–293. Springer Verlag.

[Wu et al., 2004] Wu, Z., Irizarry, R. A., Gentleman, R., Murillo, F. M., and Spencer, F. (2004). A model based background adjustment for oligonucleotide expression arrays. *Journal of the American Statistical Association*, 99 :909.

[Zar, 1999] Zar, J. H. (1999). *Biostatistical Analysis*. Prentice-Hall, 4th edition.

[Zhu et al., 2008] Zhu, J., McLachlan, G., Jones, L. B.-T., and I.A.Wood (2008). On selection biases with prediction rules formed from gene expression data. *Journal of Statistical Planning and Inference*, 138 :374–386.

[Zhu et al., 2009] Zhu, Y., Shen, X., and Pan, W. (2009). Network-based support vector machine for classification of microarray samples. *BMC Bioinformatics*, 10 (Suppl. 1) :S21.

[Zou and Hastie, 2005] Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society : Series B*, 67 :301–320.

[Zuber and Strimmer, 2009] Zuber, V. and Strimmer, K. (2009). Gene ranking and biomarker discovery under correlation. *Bioinformatics*, 25(20) :2700–2707.