THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :
Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Signal, Image, Acoustique et Optimisation

Présentée et soutenue par :
Cécile BAZOT

le : vendredi 27 septembre 2013

Titre :

Méthodes bayésiennes pour l'analyse génétique

Ecole doctorale :
Mathématiques Informatique Télécommunications (MITT)

Unité de recherche :
Institut de Recherche en Informatique de Toulouse (IRIT)

Directeur(s) de Thèse :
Jean-Yves Tourneret, Professeur, INP-ENSEEIHT, Toulouse
Nicolas Dobigeon, Maître de Conférences, INP-ENSEEIHT, Toulouse

Rapporteurs :
Sophie Lambert-Lacroix, Professeur, Université Pierre Mendès-France, Grenoble
Cédric Richard, Professeur, Université de Nice Sophia-Antipolis

Autre(s) membre(s) du jury

Wojciech Pieczynski, Professeur, TELECOM SudParis (Examinateur)

Remerciements

Je voudrais exprimer ici mes remerciements les plus sincères aux personnes qui ont contribué, de près ou de loin, à la réussite de cette thèse.

Tout d'abord, mes remerciements s'adressent à mon directeur de thèse, Jean-Yves Tourneret, et mon co-directeur, Nicolas Dobigeon, pour leur confiance, leur rigueur et leur disponibilité malgré des emplois du temps souvent bien chargés. Leur clairvoyance et leur compétences scientifiques m'ont beaucoup appris. Je remercie également Alfred Hero pour l'idée de cette thèse et son expérience en génétique.

Je remercie cordialement M. Wojciech Pieczynski pour l'intérêt qu'il a accordé à mon travail en acceptant de le juger et de présider mon jury de thèse. J'adresse aussi mes sincères remerciements à Mme Sophie Lambert-Lacroix et M. Cédric Richard pour avoir accepté d'examiner mon travail.

Ma gratitude s'adresse également à l'ensemble des membres de l'équipe SC de l'IRIT, enseignants-chercheurs ou doctorants passés et présents, pour leur convivialité et leur sympathie. Je ne vais pas citer tout le monde pour ne pas risquer à en oublier. Je remercie en particulier Yoann Altmann et Abderrahim Halimi, pour leur soutien et la bonne humeur que nous avons partagé durant trois ans dans le même bureau. Je voudrais aussi remercier les doctorants du laboratoire TeSA et ceux de l'équipe IRT, croisés dans les couloirs et avec qui j'ai pu partagé certaines pauses cafés ou déjeuners.

Je remercie aussi profondément Lucie Campagnolo pour les nombreuses pauses cafés réconfortantes. Tu as toujours été à l'écoute et a su trouver les mots pour me rassurer et me remotiver dans les moments difficiles.

Enfin les mots me manquent pour remercier à leur juste valeur mon père, ma mère et ma soeur.

En effet, même si vous n'avez pas toujours compris ce que je faisais, vous avez toujours été là pour me soutenir et m'encourager durant toutes mes années d'études. Un immense merci à vous trois. Pour finir, je me dois aussi de remercier Jérémy pour toutes les bonnes choses qu'il m'a apporté au quotidien et surtout durant ces derniers mois de thèse.

Résumé

Ces dernières années, la génomique a connu un intérêt scientifique grandissant, notamment depuis la publication complète des cartes du génome humain au début des années 2000. A présent, les équipes médicales sont confrontées à un nouvel enjeu : l'exploitation des signaux délivrés par les puces ADN. Ces signaux, souvent de grande taille, permettent de connaître à un instant donné quel est le niveau d'expression des gênes dans un tissu considéré, sous des conditions particulières (phénotype, traitement, ...), pour un individu. Le but de cette recherche est d'identifier des séquences temporelles caractéristiques d'une pathologie, afin de détecter, voire de prévenir, une maladie chez un groupe de patients observés. Les solutions développées dans cette thèse consistent en la décomposition de ces signaux en facteurs élémentaires (ou signatures génétiques) selon un modèle bayésien de mélange linéaire, permettant une estimation conjointe de ces facteurs et de leur proportion dans chaque échantillon. L'utilisation de méthodes de Monte Carlo par chaînes de Markov sera tout particulièrement appropriée aux modèles bayésiens hiérarchiques proposés puisqu'elle permettra de surmonter les difficultés liées à leur complexité calculatoire.

Mots clés : analyse génétique, méthodes MCMC, inférence bayésienne, traitement du signal, données d'expression des gènes.

Abstract

In the past few years, genomics has received growing scientific interest, particularly since the map of the human genome was completed and published in early 2000's. Currently, medical teams are facing a new challenge: processing the signals issued by DNA microarrays. These signals, often of voluminous size, allow one to discover the level of a gene expression in a given tissue at any time, under specific conditions (phenotype, treatment, ...). The aim of this research is to identify characteristic temporal gene expression profiles of host response to a pathogen, in order to detect or even prevent a disease in a group of observed patients. The solutions developed in this thesis consist of the decomposition of these signals into elementary factors (genetic signatures) following a Bayesian linear mixing model, allowing for joint estimation of these factors and their relative contributions to each sample. The use of Markov chain Monte Carlo methods is particularly suitable for the proposed hierarchical Bayesian models. Indeed they allow one to overcome the difficulties related to their computational complexity.

Keywords: factor analysis, MCMC methods, Bayesian inference, signal processing, gene expression data.

Abréviations et notations

Abréviations

ACI	analyse en composantes indépendantes [HKO01]
ACP	analyse en composantes principales [Jol86]
BeG	Bernoulli-gaussien
BIC	critère d'information bayésien *Bayesian information criterion* [Sch78]
BLU	démélange linéaire bayésien (*Bayesian linear unmixing*) [DMC+09, HZR+11]
DAG	graphe acyclique orienté (*directed acyclic graph*)
FCLS	*fully constrained least squares* [HC01]
GB-GMF	*gradient-based algorithm for general matrix factorization* [NHN+11]
HMM	modèle de Markov caché (*hidden Markov model*) [Rab89]
i.i.d.	indépendants et identiquement distribués
MAP	maximum *a posteriori*
MCMC	méthodes de Monte Carlo par chaînes de Markov (*Monte Carlo Markov chains*)
MML	modèle de mélange linéaire
MMSE	estimateur qui minimise l'erreur quadratique moyenne (*minimum mean square error*)

MSE	erreur quadratique moyenne (*mean square error*)
NMF	factorisation en matrices non-négatives (*non-negative matrix factorization*) [LS00]
d.d.p.	densité de probabilité
PMD	décomposition matricielle pénalisée (*penalized matrix decomposition*) [WTH09]
PSRF	*potential scale reduction factor*
RJ-MCMC	méthodes de Monte Carlo par chaînes de Markov à sauts réversibles (*reversible jump MCMC*) [Gre95]
SAD	angle spectral (*spectral angle distance*)
SNR	rapport signal sur bruit (*signal-to-noise ratio*)
SVD	décomposition en valeurs singulières (*singular value decomposition*)
tBLU	démélange linéaire bayésien temporel (*temporal Bayesian linear unmixing*)
uBLU	démélange linéaire bayésien non-supervisé (*unsupervised Bayesian linear unmixing*)
VCA	*vertex component analysis* [NBD05]

Notations standard

$\in$	appartient à
$\mathbb{R}$	ensemble des réels
$\mathbb{R}^+$	ensemble des réels positifs
$\mathbb{R}^n$	ensemble des vecteurs de dimension n à valeurs rélles
$\mathbb{R}^{(n\times p)}$	ensemble des matrices de dimension $n \times p$ à valeurs rélles
$\propto$	proportionnel à
$<<$	très inférieur à
$>>$	très supérieur à

$\mathbf{1}_{\mathbb{E}}(\cdot)$	fonction indicatrice définie sur l'ensemble $\mathbb{E}$: $\mathbf{I}_{\mathbb{E}}(x) = \begin{cases} 1 & \text{si } x \in \mathbb{E} \\ 0 & \text{sinon} \end{cases}$
$\delta(\cdot)$	fonction dirac
$\arccos(\cdot)$	fonction cosinus inverse
$\Gamma(\cdot)$	fonction gamma
$\mathrm{B}(a, b)$	fonction beta : $\mathrm{B}(a, b) = \Gamma(a)\Gamma(b)/\Gamma(a+b)$

Notations matricielles

x	scalaire
$\mathbf{x}$	vecteur
$\mathbf{X}$	matrice
$\mathbf{x}_i$	$i^{\text{ème}}$ colonne de la matrice $\mathbf{X}$
$\mathbf{x}_{i:j}$	sous-vecteur $[x_i, x_{i+1}, \ldots, x_{j-1}, x_j]$ de $\mathbf{x}$
$\mathbf{X}_{\setminus i}$	matrice $\mathbf{X}$ privée de sa $i^{\text{ème}}$ colonne
$.^T$	transposé
$.^{-1}$	inverse
$\lvert\mathbf{X}\rvert$	déterminant de la matrice $\mathbf{X}$
$\mathbf{I}_n$	matrice identité de taille $n \times n$
$\mathbf{1}_n$	vecteur de "1" de taille n : $\mathbf{1}_n = [1, \ldots, 1]^T \in \mathbb{R}^n$
$\mathbf{0}_n$	vecteur nul de taille n : $\mathbf{0}_n = [0, \ldots, 0]^T \in \mathbb{R}^n$
$\lVert\cdot\rVert^2$	norme l_2 standard d'un vecteur : $\lVert\mathbf{x}\rVert^2 = \mathbf{x}^T\mathbf{x}$
$\lVert\cdot\rVert_0$	norme l_0 correspondant au nombre d'éléments non-nuls d'un vecteur

Notations relatives à la modélisation

r	indice de la signature génétique ou facteur
R	nombre de signatures génétiques ou facteurs
$R_{\max}$	nombre maximal de signatures génétiques ou facteurs présents dans le mélange
g	indice du niveau d'expression du gène
G	nombre de niveaux d'expression de gènes
k	indice de l'état
K	nombre d'états
s	indice du sujet
S	nombre de sujets
t	indice temporel
T	nombre d'instants temps
i	indice de l'échantillon
N	nombre d'échantillons

Algorithmes d'estimation statistique

$\widehat{x}$	estimation de x
P [A]	probabilité de l'événement A
$\sim$	distribué suivant
$\mathrm{N_{mc}}$	longueur totale de la chaîne de Markov
$\mathrm{N_{bi}}$	longueur de la période de chauffe (*burn-in*) de la chaîne de Markov
$\mathrm{N_r}$	nombre d'itérations de la chaîne de Markov construite pour réaliser les estimations ($\mathrm{N_r} = \mathrm{N_{mc}} - \mathrm{N_{bi}}$)
$x^{(\ell)}$	ℓ-ème échantillon du processus de Markov $\left(x^{(\ell)}\right)_{\ell=1,\ldots,\mathrm{N_{mc}}}$

Distributions de probabilité usuelles

$\mathcal{U}_{\mathbb{E}}$	loi uniforme sur l'ensemble $\mathbb{E}$
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	loi normale de vecteur moyenne $\boldsymbol{\mu}$ et de matrice de covariance $\boldsymbol{\Sigma}$
$\phi(\cdot\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	fonction densité de probabilité de la loi normale $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
$\Phi(\cdot\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	fonction de répartition de la loi normale $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
$\mathcal{N}_{\mathbb{E}}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	loi normale tronquée à l'ensemble $\mathbb{E}$, de vecteur moyenne $\boldsymbol{\mu}$ et de matrice de covariance $\boldsymbol{\Sigma}$ (voir la définition dans [Rob95])
$\phi_{\mathbb{E}}(\cdot\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	fonction densité de probabilité de la loi normale tronquée $\mathcal{N}_{\mathbb{E}}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
$\Phi_{\mathbb{E}}(\cdot\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	fonction de répartition de la loi normale tronquée $\mathcal{N}_{\mathbb{E}}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
$\mathcal{B}(a, b)$	loi beta de paramètres a et b
$\mathcal{G}(a, b)$	loi gamma de paramètres a et b (voir [RC99, p. 581] pour la définition des paramètres a et b)
$\mathcal{IG}(a, b)$	loi inverse-gamma de paramètres a et b (voir [RC99, p. 582] pour la définition des paramètres a et b)
$\mathcal{D}_n(\boldsymbol{\alpha})$	loi de Dirichlet de vecteur paramètre $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]$

Table des matières

Table des figures

Liste des tableaux

Liste des algorithmes

Introduction

Contexte et problématique de la thèse

Le sujet de cette thèse a été d'étudier et de développer des méthodes de démélange bayésiennes pour l'analyse de signaux génétiques.

Depuis le début des années 1990, la génomique a connu un intérêt scientifique grandissant, notamment depuis la publication complète des cartes du génome humain en 2001 et 2004 (Projet Génome Humain). Ces données sont difficiles à analyser et à interpréter telles quelles. Il a donc été nécessaire, pour les équipes médicales, de développer des programmes informatiques afin de répondre aux immenses besoins de traitement de ces données. Nous étudions donc ces signaux, souvent de très grande taille, délivrés par des puces ADN. Ces signaux permettent de connaître à un instant donné le niveau d'expression de dizaines de milliers de gènes dans un tissu considéré, pour un individu, sous des conditions particulières (phénotype, traitement, ...). Le but de cette recherche est d'identifier des séquences temporelles caractéristiques d'une pathologie.

Les solutions développées dans cette thèse consistent en la décomposition de ces signaux en facteurs élémentaires (ou signatures génétiques) selon un modèle bayésien de mélange linéaire, permettant une estimation conjointe de ces facteurs et de leur niveau d'expression (proportion de chacun des gènes étudiés dans chaque facteur). Les paramètres inconnus (facteurs et proportions) sont alors munis de lois *a priori* appropriées au modèle et aux contraintes considérés. Ces lois *a priori* peuvent également dépendre d'autres paramètres, ou hyperparamètres. La complexité de la loi *a posteriori* résultante nous a incité à utiliser des méthodes de Monte Carlo par Chaînes de Markov (méthodes MCMC). Des algorithmes de Gibbs hybrides ont ainsi été développés afin de générer des échantillons distribués

asymptotiquement suivant la loi *a posteriori* d'intérêt. Ces échantillons sont en fin utilisés pour calculer des estimateurs bayésiens standards, comme l'estimateur du maximum *a posteriori* (MAP) ou l'estimateur qui minimise l'erreur quadratique moyenne appelé estimateur MMSE (pour *minimum mean square error*) des paramètres inconnus.

Les modèles proposés ont été développés en collaboration avec le Prof. Alfred O. Hero (Université du Michigan, Etats-Unis). Ils ont pu être testés sur des signaux réels et expertisés, collectés lors d'une récente étude épidémiologique (2008) menée sur des patients volontaires.

Organisation du manuscrit

Ce manuscrit de thèse est organisé en quatre chapitres de part plus ou moins égales.

Chapitre 1 : Ce chapitre servira d'introduction sur la génomique et la problématique étudiée. Il présentera brièvement les caractéristiques des signaux génétiques étudiés ainsi que le modèle de mélange linéaire classiquement employé pour l'étude de tels signaux. Ce chapitre fera également un état de l'art sur les méthodes de décomposition factorielle adaptées et/ou développées pour l'analyse génétique.

Les trois chapitres suivants proposeront des algorithmes pour la résolution du modèle de mélange linéaire sous certaines contraintes.

Chapitre 2 : Ce chapitre présente le modèle hiérarchique bayésien sous contraintes, ainsi qu'un algorithme totalement non-supervisé pour l'estimation des différents paramètres inconnus. La particularité du modèle proposé réside dans les contraintes imposées sur les paramètres : contraintes de positivité sur tous les paramètres (facteurs et proportions) et contrainte d'additivité sur les proportions. Ces contraintes permettent notamment une meilleure interprétation des résultats obtenus sur données réelles et l'unicité de la décomposition factorielle. L'algorithme développé pour résoudre ce problème de factorisation matricielle sous contrainte permet également d'estimer le nombre de facteurs, nombre souvent fixé dans de nombreuses analyses génétiques, en utilisant un processus de naissance et de mort sur ce nombre de facteurs [BDTH10, BDT+13]. Cet algorithme a été comparé avec d'autres méthodes existantes de décomposition factorielle.

Les résultats obtenus sur données synthétiques et réelles ont montré les performances du modèle contraint et de l'algorithme proposé pour l'analyse de données génétiques.

Chapitre 3 : Dans ce chapitre, nous avons voulu prendre en considération la parcimonie des coefficients du mélange (proportions) de chaque signature génétique dans une grande bibliothèque de facteurs tout en considérant le modèle bayésien contraint. Pour cela, nous avons choisi une loi Bernoulli-gaussienne tronquée comme loi *a priori* pour ces coefficients [BDTH11a, BDTH11b]. Cette loi permet de satisfaire la contrainte de parcimonie sur les facteurs, mais également de satisfaire les contraintes physiques de positivité et d'additivité de ces coefficients énoncées précédemment. La difficulté de cette méthode réside en l'estimation de la bibliothèque de toutes les signatures génétiques tout en sachant que certaines signatures ne seront pas dans le mélange. Des simulations conduites sur des données synthétiques et réelles, en comparaison avec d'autres méthodes d'analyse factorielle non paramétriques, ont montré les performances du modèle et de l'algorithme proposés.

Chapitre 4 : Le modèle bayésien développé dans le chapitre 2 est étendu dans ce chapitre afin de prendre en compte les dépendances temporelles des échantillons. Pour cela, le modèle initial a été couplé avec un modèle de Markov caché (HMM pour *hidden Markov model*) afin d'associer à chaque échantillon une étiquette renseignant sur l'état de celui-ci : 1) avant inoculation d'une pathologie, 2) état post-inoculation asymptomatique, 3) état pré-symptomatique ou 4) état post-symptomatique [BDTH12]. L'algorithme proposé permet alors de faire une classification temporelle des échantillons, en plus de l'estimation des facteurs et des proportions. Ce modèle a également été validé sur données synthétiques et réelles.

Chapitre 1

L'analyse de données d'expression des gènes

Sommaire

L'objectif de ce premier chapitre est de poser les bases nécessaires à la compréhension des travaux de recherche réalisés dans le cadre de cette thèse, et plus particulièrement à la nature et aux caractéristiques des données étudiées : des données temporelles d'expression des gènes. Les premières questions qui peuvent venir à l'esprit sont : Comment sont obtenues de telles données génétiques ? Quelles informations contiennent-elles ? Quelle modélisation choisir pour les étudier ? Ce chapitre tentera de répondre brièvement à toutes ces questions afin d'en savoir plus sur l'analyse de données d'expression des gènes.

Ce chapitre débute par un résumé des grandes étapes biologiques depuis le début de la génétique moderne (science qui étudie l'hérédité et les gènes), jusqu'à l'avènement de la génomique (science qui étudie le fonctionnement d'un organisme à l'échelle du génome, sans être limitée à l'étude d'un seul gène) (paragraphe 1.1). Le paragraphe 1.2 présente les puces à ADN, biotechnologies récentes permettant d'analyser le niveau d'expression des gènes dans une cellule, un tissu, un organe ou un organisme complexe, à un instant donné, et dans des conditions particulières. Les données d'expression des gènes issues de ces puces à ADN peuvent être modélisées suivant un modèle de mélange linéaire présenté au

paragraphe 1.3. Un tel modèle a d'ores et déjà été étudié dans la littérature. Le paragraphe 1.4 fait un état de l'art des méthodes de décomposition factorielle adaptées et/ou développées pour l'étude de données génétiques. Le problème de l'inférence du nombre de facteurs y est également abordé. Enfin, le paragraphe 1.5 présente plus en détails les jeux de données sur lesquels les algorithmes, développés tout au long de cette thèse et détaillés dans les chapitres suivants, ont pu être testés.

1.1 De la génétique à la génomique

1.1.1 Bref historique

La génétique moderne débute avec les travaux de Mendel en 1865, qui fut le premier à établir les lois de l'hérédité. En 1866, il publie ses résultats et énonce notamment les lois de transmission de certains caractères héréditaires. Les réactions de la communauté scientifique à ces travaux seront quasi-inexistantes à cette époque. Ce n'est qu'à partir de 1900 que les lois de Mendel seront redécouvertes. La théorie chromosomique de l'hérédité sera proposée par Sutton en 1902, puis confirmée par les travaux de Morgan sur la drosophile en 1911. Les chromosomes sont donc les supports des gènes. Morgan et Sturtevant publient en 1913 les premières cartes génétiques montrant l'ordre et la succession des gènes le long du chromosome.

Jusqu'alors rien n'est établi sur la nature des gènes ou sur leur mode d'action. En étudiant l'alcaptonurie, maladie génétique humaine héréditaire, Garrod établit en 1902 la première relation entre un gène et une enzyme, relation approfondie par Beadle et Tatum en 1941. Avery élucide la nature biochimique du matériel génétique en 1944, notamment grâce à la découverte de la transformation génétique des bactéries par Griffith en 1928. Il s'agit de l'ADN (Acide DésoxyriboNucléique).

La découverte de la structure de l'ADN en double hélice par Crick et Watson en 1953 et le décryptage du code génétique dans les années 1960 opèrent un virage en biologie moléculaire. Savoir que l'information génétique est contenue dans l'ADN ouvre les portes de nombreuses recherches en génétique. L'essor du génie génétique, ensemble des techniques de la biologie moléculaire pour la traduction de l'information génétique et de ses mécanismes de régulation, étend la génétique à la génomique. La figure 1.1 issue de [CGGG03] présente un résumé des grandes dates de la génétique à la génomique.

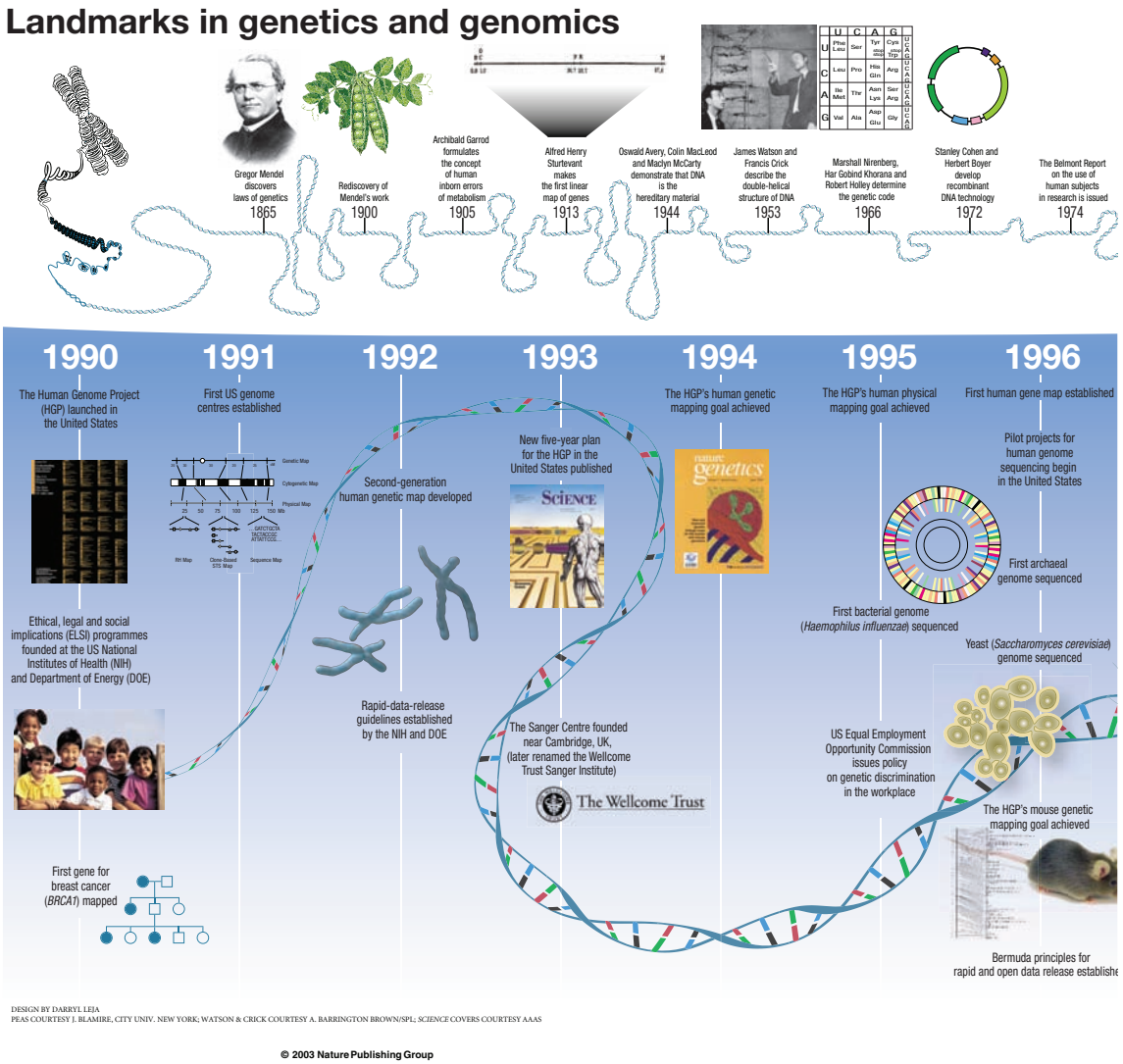


FIGURE 1.1 – Dates importantes en génétique et génomique [CGGG03].

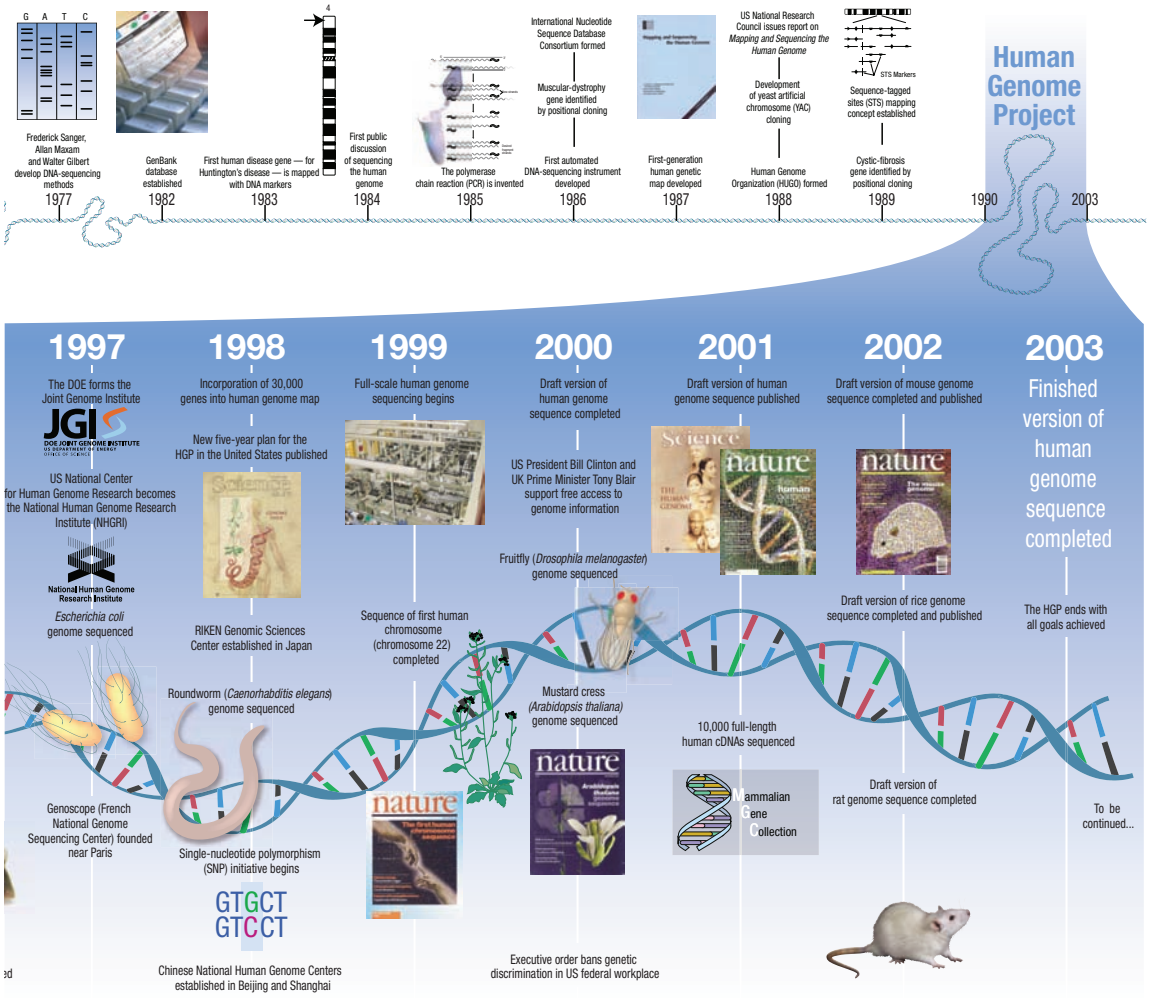


FIGURE 1.1 – Dates importantes en génétique et génomique [CGGG03].

1.1.2 Séquençage des génomes

Le séquençage est le procédé utilisé permettant de déterminer l'ordre (la séquence) des acides nucléiques constitutifs d'un brin d'ADN.

Premiers séquenceurs

En 1977, deux techniques de séquençage font leur apparition : la méthode enzymatique de Sanger et l'approche chimique de Gilbert et Maxam. Les connaissances acquises sur les enzymes amènent à privilégier la première méthode par rapport à la seconde jugée trop toxique. Les méthodes de séquençage vont se développer notamment avec la mise au point de la technique d'amplification génétique ou PCR (*Polymerase Chain Reaction*, pour amplification en chaîne par polymérisation) en 1985. Cette technique permet l'amplification sélective d'une séquence nucléique, et devient immédiatement un outil puissant et indispensable au séquençage des génomes. De là naît l'idée, en 1985 à l'*Imperial Cancer Research* (ICR) de Londres, de décrypter les trois milliards de paires de bases du génome humain afin de comprendre, dépister, prévenir les maladies génétiques et tenter de les soigner. Le premier séquenceur automatique est commercialisé en 1986.

Séquençage du génome humain

En 1988, afin de coordonner les efforts de cartographie et de séquençage de tous les pays au niveau mondial, l'organisation internationale des scientifiques impliquée dans le projet du génome humain (HUGO [1], pour *Human Genome Organization*) est créée. Le Projet Génome Humain ou *Human Genome Project* (HGP [2]) débute en 1990 pour une durée prévue de 15 ans. Ce projet international est coordonné par le *Department of Energy* (DOE [3]) américain et le *National Institutes of Health* (NIH [4]) et a pour but de décrypter et analyser le génome humain, mais aussi celui de nombreux autres organismes modèles afin de comprendre les fonctions des gènes.

La première séquence complète de génome publiée est celle de la bactérie *Haemophilus influenzae*,

1. http://www.hugo-international.org/
2. http://ornl.gov/sci/techresources/Human_Genome/home.shtml
3. http://www.energy.gov/engine/content.do
4. http://www.nih.gov

découverte en 1995 par l'équipe de Venter au TIGR (*The Institute for Genome Research*), grâce à une nouvelle technique de séquençage, appelée méthode *"shotgun"*. Suivent les séquençages des génomes de la levure *Saccharomyces cerevisiae* (1997), du ver nématode *Caenorhabditis elegans* (1998), de la drosophile *Drosophila melanogaster* (2000) et de la plante *Arabidopsis thaliana* (2000).

En 1998, Venter fonde une compagnie privée, Celera Genomics®, avec pour objectif de séquencer le génome humain en trois ans. Les premières ébauches du génome humain sont publiées en 2001, simultanément par Celera Genomics® et par le HGP. La séquence complète et précise à $99,99\%$ du génome humain est publiée en avril 2003 et est aujourd'hui librement accessible [SWG+04].

Conséquences et enjeu

L'achèvement du séquençage du génome humain a permis de donner une estimation du nombre de gènes de notre génome : entre 20 000 et 25 000 gènes. Il marque ainsi le début d'un long travail d'analyse de ces données et ouvre de nouveaux horizons de recherches en génomique. Le prochain enjeu est l'annotation des génomes, i.e., traiter l'information brute contenue dans les séquences dans le but de prédire le contenu en gènes, la position, l'organisation ou la fonction des gènes.

1.2 Puces à ADN

Définition des puces à ADN

Dès leur apparition, les puces à ADN, ou *"microarrays"* en anglais, se sont révélées être un outil privilégié pour l'étude des séquences de gènes, des mutations, ... En effet, les puces à ADN, également appelées biopuces, permettent de mesurer simultanément et quantitativement l'expression de milliers de gènes dans un type cellulaire donné, à un instant précis et dans une condition pathologique et/ou physiologique particulière, par rapport à un échantillon de référence. Elles offrent aujourd'hui des perspectives d'applications, notamment dans les domaines du pronostic, du diagnostique médical et de l'orientation thérapeutique dans le cas de pathologies diverses.

1.2.1 Principe des puces à ADN

Les puces à ADN consistent en un support solide sur lequel des milliers de fragments d'ADN, ou oligonucléotides, sont immobilisés selon une disposition ordonnée. Chacun des fragments d'ADN est représenté par un point sur le support, également appelé "spot", et sert de sonde pour fixer de façon très spécifique les fragments de gènes complémentaires (ou cibles), présents dans les échantillons à tester. La mise en contact des deux fragments d'ADN complémentaires (sondes et cibles) permet de reconstituer la double hélice d'ADN. Ce phénomène repose sur le principe d'hybridation spécifique entre deux séquences d'ADN complémentaires et remonte aux observations de Southern en 1975.

1.2.2 Technologies des puces à ADN

On distingue différents types de puces à ADN selon le support de la puce, le type de séquences utilisées comme sondes, le système de détection et de marquage des cibles, le mode de fabrication des puces, ... Le tableau 1.1 présente trois exemples de puces à ADN.

Les premières puces à ADN ont été conçues sur des membranes poreuses de nylon (appelées *"macroarrays"* en anglais). Puis, vers la fin des années 1990, les puces à ADN ont été mises au point sur des lames de verre d'un format compris entre un et quelques dizaines de cm^2. Aujourd'hui, les progrès de la robotique et l'utilisation de support solide ont permis de miniaturiser les puces. Elles comportent ainsi une très haute densité de spots (plusieurs dizaines de milliers de sondes par cm^2), susceptibles de couvrir l'intégralité du génome d'un organisme sur une simple lame de microscope.

Les types de séquences utilisés comme sondes sont : soit des fragments d'ADN complémentaires (puces à ADNc), soit des fragments oligonucléiques de synthèse (puces à oligonucléotides, aujourd'hui majoritaires sur le marché).

Les puces peuvent être produites par microdéposition des sondes (on parle de puces "spottées") ou par synthèse *in situ*. Les deux méthodes de synthèse *in situ* les plus courantes sont la photolithographie (puces *Genechips*® de la société Affymetrix [5], leader sur le marché) et la méthode à jet d'encre (puces de la société Agilent Technologies [6] par exemple).

5. http://www.affymetrix.com/index.affx
6. http://www.genomics.agilent.com/

TABLE 1.1 – Exemples de puces à ADN (d'après [Meu05]).

Support / Format	Les *"macroarrays"*	Les *"microarrays"*	Les *"Genechips®"*
Support / Format	Membrane de nylon	Verre ou silice avec revêtement chimique Lame de microscope	Verre avec revêtement chimique Cassette spécifique
Taille	12 cm × 8 cm	5,4 cm × 0,9 cm	1,28 cm × 1,28 cm
Fabrication	Microdépostion	Microdéposition	Synthèse *in situ*
Densité (par cm^2)	Quelques centaines de spots	Entre 1 000 et 10 000 spots	Jusqu'à 300 000 spots
Type de sondes	ADNc	ADNc ou oligonucléotides	Oligonucléotides
Marquage / Détection	Radioactif (^{33}P)	Fluorescent (double : cyanine 3 et 5)	Fluorescent (simple : biotine)

1.2.3 Représentation des données des puces à ADN

Pour permettre l'hybridation de la puce à ADN avec un échantillon biologique, l'échantillon à étudier doit être marqué par un radioélément ou par une molécule fluorescente, afin de détecter et quantifier ensuite l'ensemble des cibles qu'il contient.

Acquisition des images

Les résultats de l'hybridation sont acquis à l'aide de scanners de haute précision, adaptés aux marqueurs utilisés. Pour les technologies utilisant le marquage fluorescent (cas le plus fréquent), le scanner est un microscope confocal dont le pouvoir résolutif est compris entre 5 et 10 μm. En fonction du type de marquage choisi (simple ou double, i.e., par un ou deux marqueurs fluorescents), le microscope est couplé à un ou deux lasers excitant spécifiquement un fluorochrome. Le signal émis est amplifié par un photomultiplicateur et transformé en image. On obtient alors une image par fluorochrome. Chaque pixel de l'image correspond à une mesure de fluorescence, proportionnelle à l'intensité d'hybridation gène/sonde, donc à l'expression du gène ciblé. Par convention, pour les stratégies à un fluorochrome (cas des puces Affymetrix par exemple), l'intensité d'hybridation est représentée par un dégradé de bleu. Dans le cas des puces avec double marquage, l'intensité d'hybridation est représentée par un dégradé de rouge ou de vert, si l'un ou l'autre des fluorochromes prédomine, et de jaune si les deux fluorochromes sont de même intensité.

La figure 1.2 illustre le principe des puces à ADNc dans le cas d'un marquage fluorescent par deux fluorochromes : par exemple, cyanine verte (Cy3) et cyanine rouge (Cy5).

Pré-traitements des données

Les images ainsi générées comportent autant de lignes que de sondes, et donc de gènes étudiés, et autant de colonnes que d'échantillons à analyser. Notons G le nombre de gènes à étudier et N le nombre d'échantillons à analyser. A ce stade, les données sont donc représentées sous la forme d'une matrice $\mathbf{Y}$ de N colonnes, correspondant aux N échantillons, et G lignes, correspondant aux G niveaux d'expression des G gènes. En général, le nombre N d'échantillons est bien plus petit que le nombre G de gènes étudiés. Par exemple, dans le cadre de puces Affymetrix HU133 pour échantillons humains,

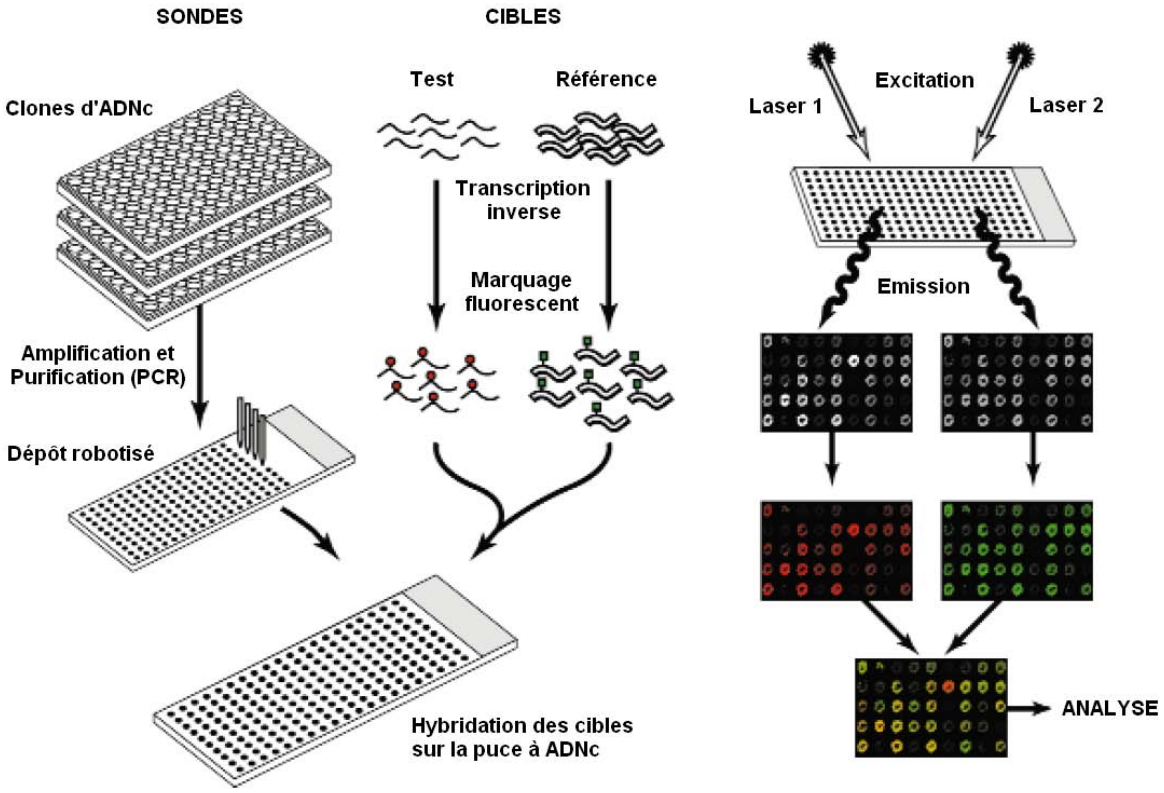


FIGURE 1.2 – Principe des puces à ADNc (d'après [DBC+99]).

le nombre G de gènes est compris entre 10 000 et 20 000, alors que nous n'analysons qu'une centaine d'échantillons. Il faut donc faire face au problème "*large p, small n*" [Wes03], avec ici $N << G$.

Dans le cas des données que nous étudierons (présentées plus en détail dans le paragraphe 1.5), chaque échantillon $\mathbf{y}_i$, pour $i = 1, \ldots, N$, de la matrice $\mathbf{Y}$ correspond à un prélèvement sanguin effectué sur un individu s ($s = 1, \ldots, S$) à un instant donné t ($t = 1, \ldots, T$). Ainsi, le nombre N total d'échantillons observés correspond à des prélèvements effectués sur S individus différents à T instants :

$$N = S \times T.$$

Les colonnes $\{\mathbf{y}_i\}_{i=1,\ldots,N}$ de la matrice des observations $\mathbf{Y}$ peuvent donc être ré-organisées de sorte que les T premières colonnes correspondent aux échantillons prélevés sur le sujet #1 aux T instants,

les T colonnes suivantes correspondent aux échantillons d'un autre sujet, par exemple le sujet #2, aux mêmes instants, etc. Cette ré-organisation des échantillons est représentée sur la figure 1.3.

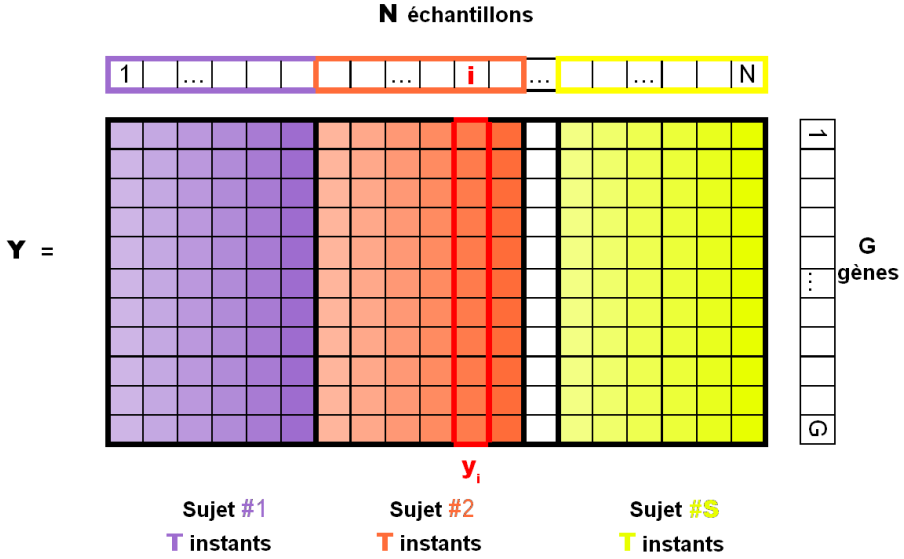


FIGURE 1.3 – Représentation de la matrice des observations $\mathbf{Y}$.

Avant d'être analysés, les résultats sont généralement "normalisés" afin de gommer les différences entre les diverses puces liées aux biais de marquage, d'hybridation ou aux variations du bruit de fond, ... Ils peuvent également être filtrés pour éliminer les résultats les moins fiables.

Organisation et diffusion des données

Les puces à ADN génèrent une masse considérable de données difficiles à gérer. Une standardisation des informations à sauvegarder a donc été indispensable pour un meilleur partage des connaissances. Dans ce but, la société savante *Microarray Gene Expression Data Society* (MGED[7]) a proposé le format MIAME (pour *"minimal information about microarray experiment"*) comportant les informations nécessaires et minimales à enregistrer pour décrire explicitement les données d'une expérience de puces à ADN. Le format MIAME est aujourd'hui la référence pour diffuser les données de puces à ADN sur des banques de données publiques telles que *Gene Expression Omnibus* (GEO[8]) ou *ArrayExpress*[9].

7. http://www.mged.org
8. http://www.ncbi.nlm.nih.gov/geo/
9. http://www.ebi.ac.uk/arrayexpress/

1.3 Modélisation des données génétiques

Un problème important rencontré en génomique est l'identification de séquences temporelles dans l'expression des différents gènes qui sont caractéristiques d'une pathologie. Pour identifier ces signatures biologiques, nous avons décidé de faire une analogie avec le problème de démélange spectral linéaire rencontré en imagerie hyperspectrale [BDPD+12]. Le but alors recherché est de décomposer les données d'expression des gènes en marqueurs élémentaires selon un modèle de mélange linéaire.

1.3.1 Le modèle de mélange linéaire (MML)

Le modèle de mélange linéaire (MML) se trouve particulièrement adapté à l'analyse de données génétiques. Il a été introduit par West [Wes03] pour l'analyse factorielle de données sous le paradigme "G grand, N petit", dans le but de l'appliquer sur des données d'expression des gènes [CCL+08].

Le modèle MML permet de décomposer chaque échantillon $\mathbf{y}_i$ $(i = 1, \ldots, N)$ (qui correspond à une colonne de la matrice des observations $\mathbf{Y}$) en une somme de R signatures génétiques élémentaires $\{\mathbf{m}_r\}_{r=1,\ldots,R}$, également appelées *facteurs* :

$$\mathbf{y}_i = \sum_{r=1}^{R} \mathbf{m}_r a_{r,i} + \mathbf{n}_i \tag{1.1}$$

où :

- $\mathbf{m}_r = [m_{1,r}, \ldots, m_{G,r}]^T$ désigne la $r^{\text{ème}}$ signature génétique, ou *facteur*,
- $a_{r,i}$ est la proportion, ou *score*, du $r^{\text{ème}}$ facteur dans le $i^{\text{ème}}$ échantillon,
- $\mathbf{n}_i$ correspond à l'erreur résiduelle dûe à la représentation MML.

En considérant les N échantillons observés, le modèle MML se réécrit sous la forme matricielle suivante :

$$\mathbf{Y} = \mathbf{MA} + \mathbf{N} \tag{1.2}$$

avec :

$$\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N] \in \mathbb{R}^{G \times N}, \qquad \mathbf{M} = [\mathbf{m}_1, \ldots, \mathbf{m}_R] \in \mathbb{R}^{G \times R},$$
$$\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N] \in \mathbb{R}^{R \times N}, \qquad \mathbf{N} = [\mathbf{n}_1, \ldots, \mathbf{n}_N] \in \mathbb{R}^{G \times N}.$$

Les matrices $\mathbf{M}$ et $\mathbf{A}$ représentent respectivement la matrice des signatures génétiques (facteurs) et celle des coefficients du mélange (scores). Les éléments $m_{g,r}$ de la matrice des facteurs $\mathbf{M}$ sont appelés *"loadings"* ($g = 1, \ldots, G$ et $r = 1, \ldots, R$).

Ajout de contraintes sur le modèle MML

Pour chaque échantillon i ($i = 1, \ldots, N$), en considérant les coefficients $\{a_{r,i}\}_{r=1,\ldots,R}$ comme des proportions (ou concentrations) de chaque facteur dans l'échantillon étudié, le vecteur $\mathbf{a}_i = [a_{1,i}, \ldots, a_{R,i}]^T$ (vecteur des scores du $i^{\text{ème}}$ échantillon) est soumis à des contraintes de non-négativité et de somme-à-un, contraintes qui permettent d'améliorer l'interprétabilité des résultats. De plus, les données issues des puces à ADN étant positives, il est tout à fait légitime d'imposer une contrainte supplémentaire de non-négativité sur les signatures génétiques $\{\mathbf{m}_r\}_{r=1,\ldots,R}$. Ces contraintes de positivité et de somme-à-un s'écrivent de la manière suivante :

$$\begin{aligned}
m_{g,r} \geq 0, \quad & \forall g = 1, \ldots, G, \quad \forall r = 1, \ldots, R \quad && \text{(non-négativité des signatures)}, \\
a_{r,i} \geq 0, \quad & \forall r = 1, \ldots, R, \quad \forall i = 1, \ldots, N && \text{(non-négativité des scores)}, \\
\sum_{r=1}^{R} a_{r,i} = 1, \quad & \forall i = 1, \ldots, N && \text{(somme-à-un des scores)}.
\end{aligned} \tag{1.3}$$

Remarquons que de telles contraintes ont déjà été imposées par Dobigeon *et al.*, notamment dans [DMC+09], pour le démélange linéaire d'images hyperspectrales. L'analogie entre le démélange de données génétiques et le démélange spectral d'images hyperspectrales est présentée dans le paragraphe suivant 1.3.2.

En revanche, de telles contraintes n'ont pas, à notre connaissance, été imposées dans le contexte de l'analyse génétique, alors qu'elles sont tout à fait justifiables et permettent notamment une meilleure interprétation des paramètres inconnus. L'apport de chacune de ces contraintes pour le démélange de données génétiques sera plus particulièrement étudié dans le chapitre 2 (paragraphe 2.6.3).

Comme dans de nombreuses méthodes d'analyse factorielle bayésiennes, le vecteur $\mathbf{n}_i = [n_{1,i}, \ldots, n_{G,i}]^T$ est une suite de variables aléatoires que l'on supposera indépendantes et identiquement distribuées (i.i.d.) suivant une loi normale centrée et de matrice de covariance $\mathbf{\Sigma} = \sigma^2 \mathbf{I}_G$:

$$\mathbf{n}_i | \sigma^2 \sim \mathcal{N}\left(\mathbf{0}_G, \sigma^2 \mathbf{I}_G\right) \tag{1.4}$$

où $\mathbf{I}_G$ est la matrice identité de dimension $G \times G$ et $\mathbf{0}_G$ est le vecteur de $\mathbb{R}^G$ constitué de G zéros.

Le problème de démélange linéaire considéré ici consiste donc à estimer conjointement les matrices des signatures génétiques $\mathbf{M}$ et des scores $\mathbf{A}$, directement à partir de la matrice des observations $\mathbf{Y}$, en respectant les contraintes de non-négativité et de somme-à-un pré-citées (1.3). Formellement, il s'agit donc de résoudre un problème de séparation aveugle de sources ou de décomposition factorielle, sous les contraintes (1.3). La modélisation bayésienne se trouve être particulièrement appropriée pour ce genre de problèmes. En effet, les contraintes imposées peuvent guider naturellement le choix des lois *a priori* des paramètres inconnus (matrice des signatures, matrices des scores et variance du bruit). L'estimation des paramètres inconnus est alors réalisée à partir de leurs lois *a posteriori*. La complexité de ces lois impose le recours à des méthodes de simulation appropriées comme les méthodes de Monte Carlo par chaînes de Markov (MCMC) [GRS96].

1.3.2 Analogie entre l'imagerie hyperspectrale et l'analyse génétique

L'approche de démélange bayésien sous contraintes, adoptée dans cette thèse et appliquée aux données génétiques, s'appuie sur une étude récente [DMC+09] visant à résoudre le problème de démélange spectral rencontré en imagerie hyperspectrale. En effet, dans le cas des images hyperspectrales, une scène est observée dans plusieurs centaines de longueurs d'ondes étroites et contigües (et non plus trois ou quatre longueurs d'onde comme c'est le cas pour les images multispectrales). Un vecteur de mesures, formant un spectre, est alors associé à chaque pixel de l'image hyperspectrale.

Le problème de démélange spectral consiste alors à décomposer le spectre d'un pixel observé en un mélange de spectres caractéristiques de composants purs, comme de la végétation, de l'eau ou du sable par exemple, également appelés *signatures spectrales* (ou "*endmembers*") et à estimer la *proportion* (ou coefficient d'*abondance*) de chacun de ces matériaux purs dans le pixel étudié [KM02].

Pour modéliser ce problème, on considère une image hyperspectrale $\mathbf{Y}$ constituée de P pixels, acquise dans L longueurs d'ondes (ou bandes spectrales). Selon le modèle de mélange linéaire (MML), le spectre du $p^{\text{ème}}$ pixel $\mathbf{y}_p$ s'écrit alors comme la combinaison linéaire de R signatures spectrales $\{\mathbf{m}_r\}_{r=1,\ldots,R}$, à laquelle s'ajoute un bruit $\mathbf{n}_p$ gaussien constitué de variables aléatoires indépendantes et identiquement distribuées :

$$\mathbf{y}_p = \sum_{r=1}^{R} \mathbf{m}_r a_{r,p} + \mathbf{n}_p \qquad \text{pour} \qquad p = 1, \ldots, P,$$

où :

- $\mathbf{m}_r$ correspond au spectre du $r^{\text{ème}}$ matériau présent dans l'image,
- $a_{r,p}$ est la proportion (abondance) du $r^{\text{ème}}$ matériau dans le $p^{\text{ème}}$ pixel étudié.

En raison de considérations physiques, les mêmes contraintes de positivité (sur les spectres et abondances) et de somme-à-un (pour les abondances) s'appliquent aux images hyperspectrales.

En identifiant les pixels de l'image hyperspectrale avec les échantillons prélevés issus des puces à ADN et les longueurs d'ondes avec les indices génétiques, cette approche de démélange spectral peut être étendue aux données d'expression des gènes. La table 1.2 résume cette analogie entre analyse génétique et imagerie hyperspectrale.

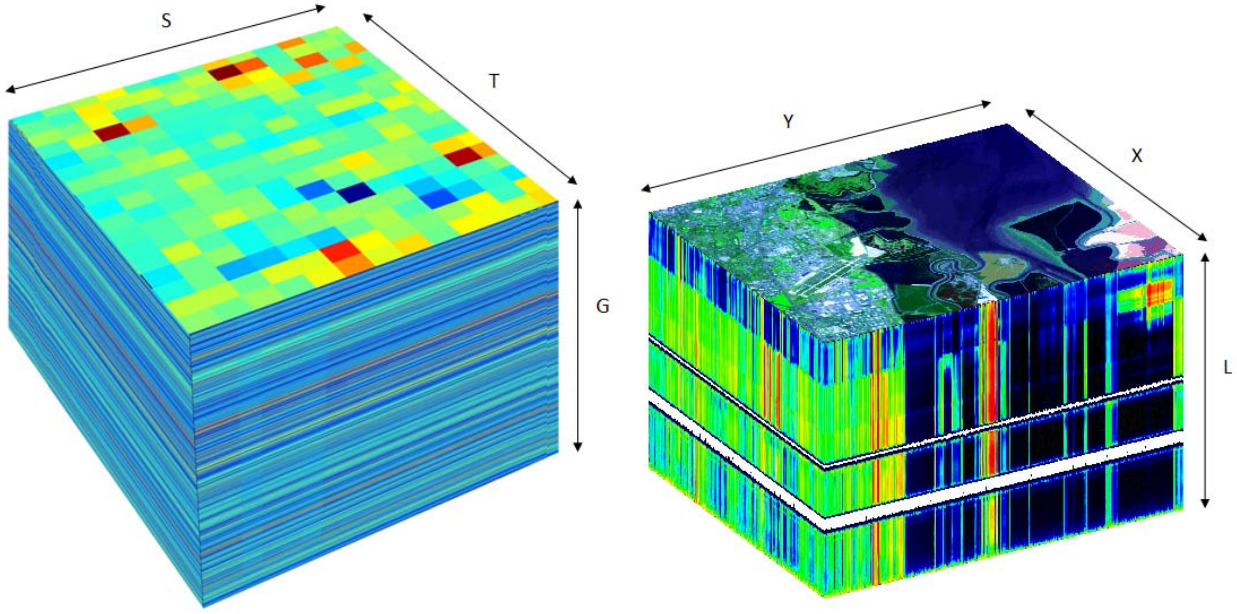
TABLE 1.2 – Analogie entre l'analyse génétique et l'imagerie hyperspectrale.

Analyse génétique	Imagerie hyperspectrale
$\mathbf{Y} \in \mathbb{R}^{G \times N}$	$\mathbf{Y} \in \mathbb{R}^{L \times P}$
G données d'expression des G gènes	L longueurs d'ondes
N échantillons d'observation	P pixels d'observation
$\mathbf{y}_i$: $i^{\text{ème}}$ échantillon ou prélèvement	$\mathbf{y}_p$: $p^{\text{ème}}$ pixel de l'image
$\mathbf{a}_i$: vecteur des *scores* de $\mathbf{y}_i$	$\mathbf{a}_p$: vecteur des *abondances* de $\mathbf{y}_p$
$\mathbf{m}_r$: $r^{\text{ème}}$ signature génétique, ou *facteur*	$\mathbf{m}_r$: $r^{\text{ème}}$ signature spectrale, ou *endmember*

On représente souvent les données hyperspectrales sous la forme d'un cube avec deux dimensions spatiales et une dimension spectrale (cf. figure 1.4b, image AVIRIS [10]). La face supérieure du cube est

10. http://aviris.jpl.nasa.gov/data/image_cube.html

alors la scène spatiale, une composition colorée de trois bandes spectrales. Toutes les scènes pour les différentes longueurs d'onde sont ensuite empilées pour former le cube. De même, on peut représenter les données génétiques sous la forme d'un cube de données. La figure 1.4a représente les données grippales H3N2 (détaillées dans le paragraphe 1.5.2) : la face du dessus correspond à l'expression moyenne des gènes des échantillons des S sujets au cours du temps (T instants temps).



(a) Cube de données génétiques (grippe H3N2)

(b) Cube hyperspectral (Moffet Field, Californie, Etats-Unis)

FIGURE 1.4 – Exemples de cubes de données : génétiques (a) et hyperspectrales (b).

1.4 Etat de l'art des méthodes d'analyse factorielle pour des données génétiques

Le démélange linéaire bayésien, traditionnellement utilisé pour l'analyse d'images hyperspectrales, est l'une des nombreuses méthodes possibles de décomposition factorielle qui pourraient être appliquées à l'analyse des données d'expression des gènes. Cette section présente différentes approches pour ajuster le modèle (1.2) aux données, dont l'analyse en composantes principales [YR01, NBD05], les

méthodes de factorisation de matrices par moindres carrés [LS00, FYHL07], les modèles de mélanges finis [MBP02, BM11], les approches bayésiennes [MKG+02, CCL+08, FDF+10], ...

1.4.1 Méthodes d'analyse factorielle classiques

Parmi les méthodes non-bayésiennes permettant de résoudre l'équation matricielle $\mathbf{Y} \approx \mathbf{MA}$, ou de manière équivalente $\mathbf{Y}^T \approx \mathbf{A}^T\mathbf{M}^T$, nous pouvons citer :

- l'**analyse en composantes principales (ACP)** [Jol86, YR01] : l'ACP de $\mathbf{Y}^T$ recherche les axes $\mathbf{m}_r$, plus communément appelés composantes principales, de plus grandes variances, sous contraintes d'orthogonalité ; l'ACP est aujourd'hui plus largement utilisée comme méthode de réduction de dimensionalité,
- l'**analyse en composantes indépendantes (ACI)** [HKO01, KVG+08] : l'ACI de $\mathbf{Y}^T$ permet de résoudre le problème de séparation de sources, en maximisant l'indépendance statistique entre les sources $\mathbf{m}_r$,
- la **factorisation en matrices non-négatives (NMF pour *"non-negative matrix factorization"*)** [LS00, FYHL07] : la NMF impose que tous les éléments de $\mathbf{M}$ et $\mathbf{A}$ soient non-négatifs (positifs ou nuls),
- la **décomposition en matrices pénalisées (PMD pour *"penalized matrix decomposition"*)** [WTH09] : la PMD permet d'approximer la matrice $\mathbf{Y}$ par une matrice de rang R, en utilisant des fonctions de pénalité,
- l'**algorithme GB-GMF (pour *"gradient-based algorithm for general matrix factorization"*)** de Nikulin *et al.* [NHN+11] : un algorithme rapide pour les factorisations matricielles de données de grande dimension, basé sur une descente de gradient stochastique, et sans contraintes sur les matrices $\mathbf{M}$ et $\mathbf{A}$.

Remarquons que l'ensemble de ces méthodes ont été appliquées aux données génétiques, même si elles n'avaient pas été dévéloppées initialement pour de telles données.

Les méthodes ACP, ACI et GB-GMF ne prennent pas en compte la non-négativité des facteurs et des scores. L'algorithme NMF a l'avantage d'imposer cette contrainte de non-négativité sur les éléments des matrices, ce qui permet une meilleure interprétation des facteurs et des scores. En

revanche, NMF ne prend pas en compte la contrainte de somme-à-un sur les colonnes de la matrice des scores.

L'ACP et l'ACI imposent respectivement l'orthogonalité ou l'indépendance des sources $\mathbf{m}_r$. Or, les processus biologiques sont souvent non-indépendants. L'algorithme GB-GMF, qui n'impose pas de telles contraintes, est donc plus flexible pour modéliser, par exemple, les comportements biologiques dans lesquels les signatures génétiques se chevaucheraient et seraient donc non-indépendantes.

1.4.2 Méthodes d'analyse factorielle bayésiennes

De nombreuses méthodes bayésiennes ont aussi été développées pour résoudre ce problème de décomposition factorielle des données génétiques. Nous pouvons citer :

- la **décomposition bayésienne (DB)** de Ochs *et al.* [OSAMB99, MKG+02] : méthode initialement développée pour la décomposition spectrale d'images par résonance magnétique nucléaire (IRM), puis appliquée aux données d'expression des gènes,
- le **modèle de régression bayésienne (noté BFRM)** de Carvalho *et al.* [CCL+08] : modèle généralisé développé pour les données génétiques dans le cas où $N << G$, pouvant être appliqué en fixant ou non le nombre de facteurs,
- la **méthode CoGAPS (pour *"coordinated gene activity in patterns sets"*)** [FDF+10] : disponible en open-source sous R [11] (dans la plateforme d'outils Bioconductor [12]),
- l'**analyse factorielle non-paramétrique bayésienne (notée NPBFA pour *"non-parametric Bayesian factor analysis"*)** de Chen *et al.* [CCP+10] : approche parcimonieuse et non-paramétrique, avec l'utilisation de processus Beta pour estimer le nombre de facteurs R.

En 2009, Kossenkov et Ochs [KO09] ont montré, sur des données génétiques de *Saccaromyces cerevisiae* (levure), que les méthodes bayésiennes de décomposition factorielle (dont DB [MKG+02] et BFRM [CCL+08]) étaient plus adaptées que les méthodes traditionnelles (comme la NMF [FYHL07]) pour déterminer et retrouver des signatures génétiques en relation avec les processus biologiques et les phénotypes. Ceci s'explique notamment par le fait que, dans le cas de l'analyse génétique, les facteurs

11. http://www.r-project.org/
12. http://www.bioconductor.org/

correspondent à des réponses géniques plus ou moins corrélées à des stimuli. En d'autres termes, les processus biologiques proviennent de l'activité coordonnée d'un ensemble de gènes. Il est donc plus intéressant d'inférer les processus biologiques, et donc inférer les facteurs, à partir de méthodes basées sur des ensembles de gènes, que sur des gènes isolés les uns des autres.

La plupart des méthodes présentées dans les deux paragraphes précédents nécessitent de spécifier le nombre de facteurs R de la décomposition, nombre dont dépend les dimensions des matrices $\mathbf{M}$ (nombre de colonnes) et $\mathbf{A}$ (nombre de lignes) à déterminer. Le paragraphe suivant traite de ce problème.

1.4.3 Comment inférer le nombre de facteurs R de la décomposition ?

La détermination du nombre de facteurs R présents dans les données est un problème inhérent à toute méthode d'analyse factorielle.

Dans de nombreuses approches non-bayésiennes (ACP, ACI, NMF, GB-GMF, ...), ce nombre R est fixé. Il est alors nécessaire d'exécuter les algorithmes en testant différentes valeurs de R, puis parmi les valeurs testées de trouver la meilleure reconstruction, ou alors d'utiliser un algorithme d'analyse statistique en pré-traitement pour déterminer le nombre de facteurs R. D'autres approches non-bayésiennes, comme l'ACP parcimonieuse [ZHT04] et la PMD [WTH09] déduisent R en surveillant l'erreur de reconstruction en fonction du nombre d'itérations.

Dans le cas des algorithmes bayésiens, l'approche la plus largement utilisée [BGM03] pour déterminer le nombre R de facteurs (et donc pour la sélection de modèles) est le critère d'information bayésien (ou BIC pour "*Bayesian information criterion*" en anglais) [Sch78]. Plus récemment, les approches bayésiennes non-paramétriques ont suscité un grand intérêt pour inférer R à partir des données observées $\mathbf{Y}$ [KG07]. Citons notamment le développement de méthodes basées sur les processus de Dirichlet, les processus de type "Indian buffet" (IBP pour "*Indian buffet process*" en anglais) [GG05] ou les processus Beta [PC09, CCP+10].

Il est donc intéressant de développer des méthodes qui permettraient d'estimer le nombre R de facteurs conjointement à l'estimation des matrices $\mathbf{M}$ et $\mathbf{A}$ de la décomposition. C'est ce que nous proposerons dans les chapitres suivants.

1.5 Détails sur les données étudiées

Les méthodes bayésiennes proposées dans ce manuscrit, et présentées dans les chapitres suivants, ont toutes été testées sur des données synthétiques génétiques, mais également validées sur des données réelles d'expression génique. Toutes les données réelles étudiées sont disponibles publiquement sur le site *Gene Expression Omnibus* (GEO [13]). Ce site est une base de données publique qui archive et distribue librement des données d'expression de gènes de grande taille soumises par la communauté scientifique [EDL02]. Il est développé et entretenu par le NCBI (*National Center for Biotechnology Information*). En plus du stockage de données, ce site propose un ensemble d'interfaces web et d'applications pour aider les utilisateurs à la recherche et à l'analyse des données stockées.

1.5.1 Données de boissons

Ces données de boissons correspondent à une étude, proposée par Baty *et al.* et détaillée dans [BFW+06]. Cette étude a pour but d'évaluer l'influence de la consommation de boissons (alcoolisées ou non) au cours du temps dans l'expression des gènes dans le sang. Les données sont disponibles sur le site GEO, sous le numéro GSE3846.

Six volontaires en bonne santé ont participé à cette étude : au cours de 4 jours indépendants, ils ont bu 4 boissons différentes (alcool, jus de raisin, eau et vin rouge). Des échantillons de sang ont été prélevés à cinq instants : à $t = 0$, 1, 2, 4 et 12 heures après l'ingestion de la boisson. Sur les 120 échantillons prélevés (6 sujets $\times$ 4 expériences $\times$ 5 instants = 120), seuls $N = 108$ ont pu être inclus dans l'analyse (les autres étant de trop mauvaise qualité). Chaque échantillon comporte l'expression de $G = 22\,283$ gènes. Les données ont été normalisées à l'aide de l'algorithme RMA (pour *"robust multi-array average"*) [IHC+03].

Sur la figure 1.5, les échantillons sont regroupés par boisson ingérée et ré-organisés de manière à mieux visualiser l'influence de chaque boisson au cours du temps pour chaque sujet. Les échantillons noirs correspondent aux données non exploitables.

13. http://www.ncbi.nlm.nih.gov/geo/

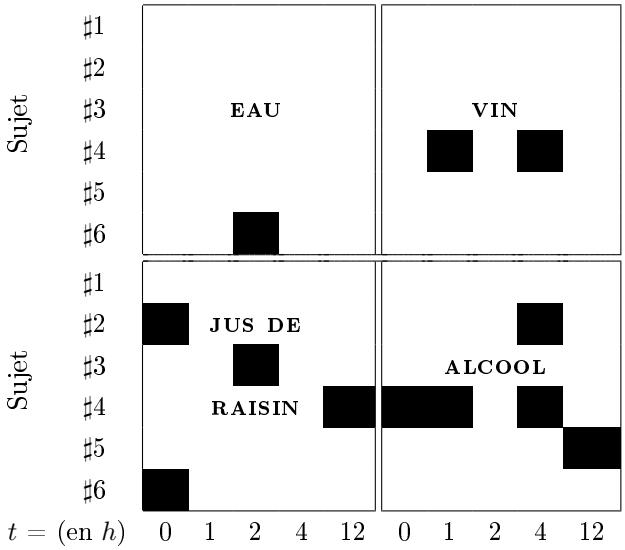


FIGURE 1.5 – Réorganisation des échantillons des données de boissons [BFW+06] pour mieux visualiser l'influence des boissons sur les gènes.

1.5.2 Données grippales H3N2

Ces données réelles correspondent à des données d'expression de gènes d'une récente étude sur la grippe A/H3N2/Wisconsin. Elles sont disponibles sous GEO, sous le numéro GSE30550, et brièvement décrites dans ce paragraphe. Pour plus de détails, se reporter aux articles de Zaas *et al.* [ZCV+09] et Huang *et al.* [HZR+11].

Les données H3N2 contiennent les niveaux d'expression des gènes de $N = 267$ puces Affymetrix, recueillies sur $S = 17$ volontaires sains ayant été expérimentalement infectés par des souches de la grippe A/H3N2.

Plus précisément, l'étude consiste à inoculer par voie intranasale une dose de 10^6 $TCID_{50}$ par mL (TCID : *"tissue culture infective dose"*) de souches de la grippe A/Wisconsin/67/2005 (H3N2) fabriquées et transformées selon les bonnes pratiques de fabrication (BPF) par Baxter BioScience. Des prélèvements sanguins sur puces Affymetrix U133a ont été effectués à $T = 16$ instants différents : un état correspondant à l'état initial (24 heures avant l'inoculation du virus), puis toutes les 8 heures pendant 120 heures, et enfin toutes les 24 heures pendant encore deux jours. Chaque échantillon est

composé de $G = 12\,023$ valeurs d'expression des gènes, normalisées à l'aide de l'algorithme RMA [HZR+11].

Parallèlement aux échantillons sanguins prélevés, le clinicien demandait régulièrement à chaque sujet son ressenti sur ses symptômes cliniques : nez bouché, gorge irritée, maux de tête, toux, ... Chaque réponse pour chaque symptôme est côtée sur une échelle de 0 à 3 pour "pas de symptôme", "symptôme à peine perceptible", "symptôme gênant mais n'empêchant pas la pratique des activités quotidiennes", "symptôme gênant rendant difficile la réalisation des activités quotidiennes". Le score total obtenu, dit "score de Jackson" [GFGV58], permet de déterminer quels sujets sont devenus symptomatiques et quels autres ne le sont pas devenus. Les résultats obtenus sont représentés sur la figure 1.6. Sur cette figure, les sujets ont également été ré-organisés de manière à ce que les huit premiers sujets correspondent aux patients asymptomatiques (n'ayant pas développé de symptômes cliniques significants), les neuf suivant (labelisés Z01, Z05, Z07, Z08, Z10; Z12, Z13 et Z15) correspondent aux patients symptomatiques. Ces résultats seront uniquement utilisés comme vérité terrain pour quantifier les performances des algorithmes développés, ils ne seront pas utilisés en tant que connaissance *a priori*.

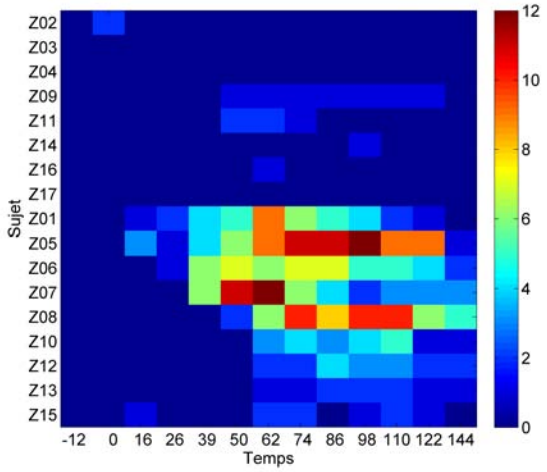


FIGURE 1.6 – Evaluation des symptômes cliniques pour chaque sujet (ligne), à différents instants temps (colonne), inoculé par la grippe H3N2 : score de Jackson [GFGV58].

Il est important de noter que la grippe est une infection virale respiratoire. Le délai moyen entre l'infection et la libération du virus grippal varie de 18 à 72 heures. Il est néanmoins possible de détecter le virus grippal dans le mucus respiratoire 24 heures avant le début des symptômes cliniques. La concentration du virus grippal atteint un pic de 10^3 à 10^7 TCID_{50} par mL, puis reste à un niveau élevé pendant 24 à 48 heures, et enfin décroit rapidement. Le virus n'est plus détectable après 5 à 10 jours en moyenne [NBAS07].

1.5.3 Données grippales H1N1

Ce dernier jeu de données est similaire au précédent : $S = 24$ patients volontaires et sains ont été infectés par des souches de grippe A/H1N1. Des prélèvements sanguins sur puces Affymetrix U133a ont été effectués à $T = 16$ instants différents. Chaque échantillon contient les données d'expression de $G = 12\ 023$ gènes, normalisées à l'aide de l'algorithme RMA [HZR+11]. Au total, ce jeu de données H1N1 comporte $N = 378$ échantillons.

Chapitre 2

Démélange bayésien non-supervisé pour l'analyse génétique

Sommaire

2.1 Introduction

Parmi les méthodes existantes de décomposition factorielle de la forme $\mathbf{Y} \approx \mathbf{MA}$, toutes ne permettent pas l'estimation du nombre de facteurs R de la décomposition (voir le paragraphe 1.4.3). D'autre part, les méthodes existantes ne prennent pas forcément en compte les contraintes liées à la physique des données étudiées : contraintes de non-négativité des facteurs et des scores et contrainte de somme-à-un des scores, définies précédemment dans (1.3). Le modèle de démélange linéaire bayésien proposé dans ce chapitre, nommé uBLU (pour *"unsupervised Bayesian linear unmixing"*), répond à ces deux problématiques.

Ce modèle uBLU s'appuie sur le modèle de démélange bayésien linéaire semi-supervisé développé par Dobigeon *et al.* pour l'imagerie hyperspectrale [DMC+09] et appliqué aux données génétiques dans [HZR+11] (nommé BLU pour *"Bayesian linear unmixing"* dans ce dernier). En effet, le paragraphe 1.3.2 a montré qu'il est tout à fait possible de faire une analogie entre les données génétiques et les

images hyperspectrales. La méthode BLU a l'avantage d'imposer les contraintes de non-négativité et de somme-à-un sur les paramètres inconnus à estimer. En revanche, BLU nécessite la détermination du nombre de facteurs R présents dans le mélange.

Le modèle uBLU proposé pour l'analyse de données génétiques étend le modèle bayésien hiérarchique précédent à une version entièrement non-supervisée qui permet l'estimation du nombre de facteurs R présents dans les échantillons observés, conjointement à l'estimation des matrices des facteurs $\mathbf{M}$ et des scores $\mathbf{A}$. Remarquons que les paramètres du mélange (dimensions des matrices $\mathbf{M}$ et $\mathbf{A}$) dépendent de ce nombre R de facteurs. Ainsi à chaque valeur possible de R correspond un modèle. L'estimation de ce nombre R revient donc à un problème de sélection de modèle (voir [Che98] pour plus de détails sur la sélection de modèle dans un cadre bayésien).

Pour résoudre ce problème de sélection de modèle, nous adoptons un processus de naissance et de mort, conjoint à une méthode de Monte Carlo par chaînes de Markov (MCMC). L'algorithme ainsi proposé permet des mouvements entre les modèles de dimensions différentes, afin de pouvoir estimer le paramètre R, comme le ferait un algorithme à sauts réversibles (RJ-MCMC pour *reversible-jump MCMC*). L'utilisation de méthodes RJ-MCMC [Gre95] a déjà fait ses preuves dans de nombreuses applications du traitement du signal et des images, comme par exemple en segmentation [PADF02], pour les signaux audio [DGI06], en analyse spectrale [AD99] ou encore en imagerie hyperspectrale [DTC08]. En revanche, cette approche a été peu développée pour les données génétiques (citons principalement [LW03] puis [GD08]) du fait des coûts calculatoires qu'elle impliquait sur de telles données. Ici, ce problème a pu être contourné par l'utilisation d'une méthode de réduction de dimensionalité, similaire à celle proposée dans [DMC+09], permettant ainsi de travailler dans l'espace projeté des facteurs de dimension $R-1$, au lieu de travailler dans l'espace des facteurs de dimension G ($G >> R$).

Organisation du chapitre

Ce chapitre est organisé comme suit. Le paragraphe 2.2 présente les différentes lois *a priori* choisies pour construire le modèle bayésien proposé pour l'analyse génétique. Un intérêt sera plus particulièrement apporté à la prise en compte des contraintes de non-négativité et de somme-à-un (1.3) dans le choix des lois *a priori*. Le paragraphe 2.3 étudie un échantillonneur de Gibbs hybride, combiné à

un processus de naissance et de mort pour la sélection de modèle (mise à jour du nombre de facteurs R), qui est utilisé pour générer des échantillons distribués suivant la loi *a posteriori* des paramètres inconnus du modèle uBLU. Les problèmes de diagnostic de convergence des méthodes MCMC sont traités au paragraphe 2.4. Enfin, des résultats de simulations associés à différents jeux de donnés synthétiques sont présentés au paragraphe 2.5 et à des données génétiques réelles au paragraphe 2.6. L'algorithme sera notamment comparé avec d'autres méthodes de décomposition factorielle présentées au paragraphe 1.4, telles que l'analyse en composantes principales (ACP) [YR01], la factorisation en matrices non-négatives (NMF) [FYHL07], le modèle BFRM [CCL+08] et l'algorithme GB-GMF [NHN+11].

2.2 Modèle bayésien hiérarchique

Cette partie présente le modèle uBLU utilisé pour l'analyse génétique. Ce modèle bayésien hiérarchique non-supervisé est basé sur la vraisemblance des observations et sur la définition de lois *a priori* adéquates, permettant de respecter les contraintes imposées sur les paramètres, notamment les vecteurs des scores $\{\mathbf{a}_i\}_{i=1,\dots,N}$ et les facteurs $\{\mathbf{m}_r\}_{r=1,\dots,R}$, et d'estimer le nombre R de facteurs du mélange.

2.2.1 Fonction de vraisemblance

Le modèle de mélange linéaire défini par l'équation (1.1) ainsi que les propriétés statistiques du vecteur de bruit (gaussien i.i.d.) $\mathbf{n}_i$ (1.4) permettent d'écrire, pour chaque échantillon $\mathbf{y}_i$ observé $(i = 1, \dots, N)$, la vraisemblance suivante :

$$\mathbf{y}_i|\mathbf{M}, \mathbf{a}_i, R, \sigma^2 \sim \mathcal{N}\left(\mathbf{M}\mathbf{a}_i, \sigma^2\mathbf{I}_G\right) \tag{2.1}$$

où $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ désigne la loi gaussienne multivariée de vecteur moyenne $\boldsymbol{\mu}$ et de matrice de covariance $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}_G$. Ainsi, en supposant que les vecteurs bruit $\{\mathbf{n}_i\}_{i=1,\dots,N}$ sont tous indépendants, la fonction de vraisemblance des observations $\mathbf{Y}$ s'écrit :

$$f\left(\mathbf{Y}|\mathbf{M}, \mathbf{A}, R, \sigma^2\right) = \prod_{i=1}^{N} f\left(\mathbf{y}_i|\mathbf{M}, \mathbf{a}_i, R, \sigma^2\right) = \frac{1}{(2\pi\sigma^2)^{GN/2}} \exp\left[-\frac{\sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\|^2}{2\sigma^2}\right] \tag{2.2}$$

où $\|\cdot\|$ est la norme l_2 définie par : $\|\mathbf{x}\| = \sqrt{\mathbf{x}^T\mathbf{x}}$.

2.2.2 Lois *a priori* des paramètres et hyperparamètres

L'approche et les lois *a priori* choisies pour les paramètres et hyperparamètres du modèle uBLU sont définies ci-après. Une attention particulière sera portée sur les lois *a priori* choisies pour les facteurs et les scores, lois tronquées pour prendre en compte les contraintes, ainsi que sur la technique de réduction de dimensionalité, permettant de travailler dans un sous-espace de dimension réduite.

Loi *a priori* pour le nombre de facteurs

Nous avons choisi une loi discrète uniforme sur $[2, \ldots, R_{\max}]$ comme loi *a priori* pour le nombre de facteurs R à estimer :

$$\mathrm{P}[R = k] = \frac{1}{R_{\max} - 1}, \quad \text{pour} \quad R = 2, \ldots, R_{\max}. \tag{2.3}$$

On note $R_{\max}$ le nombre maximal de facteurs pouvant être présents dans les échantillons. Cette valeur est fixée judicieusement afin de ne pas considérer inutilement des modèles de grandes dimensions improbables. D'autre part, le choix d'une distribution uniforme non-informative permet de ne pas privilégier un modèle parmi les modèles possibles.

Réduction de dimensionalité

Une première étape de l'algorithme consiste à faire une réduction de dimensionalité afin de travailler dans un sous-espace réduit $\mathcal{V}_{R-1}$ de dimension adéquate $R-1$ sans perte d'information. En effet, les vecteurs observés $\{\mathbf{y}_i\}_{i=1,\ldots,N} \in \mathbb{R}^G$ respectant les contraintes de non-négativité et de somme-à-un définies précédemment (1.3) appartiennent à un polytope convexe de $\mathbb{R}^G$ dont les sommets sont les R signatures génétiques à estimer $\{\mathbf{m}_r\}_{r=1,\ldots,R}$. Les données cachées $\mathbf{Y} - \mathbf{N} = \mathbf{MA}$ peuvent ainsi être représentées dans un sous-espace $\mathcal{V}_{R-1}$ de dimension $R-1$, avec $R << G$. Ce sous-espace peut être estimé préalablement par une méthode de réduction de dimensionalité, comme l'analyse en composantes principales (ACP) [Jol86]. Notons que ce genre d'approche a été utilisé avec succès dans le contexte de l'imagerie hyperspectrale [DMC+09] et permet de réduire considérablement le nombre de degrés de liberté de l'espace contenant les paramètres à estimer, notamment pour la matrice des

signatures génétiques $\mathbf{M}$. Cela réduit donc également les coûts calculatoires impliqués par l'utilisation du processus de naissance et de mort pour estimer le nombre de facteurs R.

Ainsi, au lieu d'estimer directement les signatures génétiques $\mathbf{m}_r$ $(r = 1, \ldots, R_{\max})$, nous proposons d'estimer leurs projections $\mathbf{t}_r$ sur le sous-espace $\mathcal{V}_{R_{\max}-1}$ de dimension réduite. Les signatures génétiques $\mathbf{m}_r$ (contenues dans $\mathbf{M}$) et leurs projections $\mathbf{t}_r = [t_{1,r}, \ldots, t_{R_{\max}-1,r}]^T$ (contenues dans $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_{R_{\max}}] \in \mathbb{R}^{(R_{\max}-1)\times R_{\max}}$) sur les composantes principales pertinentes issues de l'ACP sont reliées par les relations suivantes :

$$\begin{array}{rcl} \mathbf{m}_r &=& \mathbf{U}\mathbf{t}_r + \bar{\mathbf{y}}, \\ \mathbf{t}_r &=& \mathbf{P}\left(\mathbf{m}_r - \bar{\mathbf{y}}\right), \end{array} \quad \text{ou} \quad \begin{array}{rcl} \mathbf{M} &=& \mathbf{U}\mathbf{T} + \bar{\mathbf{y}}\mathbf{1}_{R_{\max}}^T, \\ \mathbf{T} &=& \mathbf{P}\left(\mathbf{M} - \bar{\mathbf{y}}\mathbf{1}_{R_{\max}}^T\right), \end{array} \tag{2.4}$$

où $\mathbf{P}$ est la matrice de projection sur le sous-espace $\mathcal{V}_{R_{\max}-1}$ ($\mathbf{P}$ est de taille $R_{\max}-1 \times G$), $\mathbf{U} = \{u_{g,k}\}$ la pseudo-inverse de $\mathbf{P}$, $\bar{\mathbf{y}}$ est la moyenne empirique des échantillons $\bar{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{y}_i = [\bar{y}_1, \ldots, \bar{y}_G]^T \in \mathbb{R}^G$, et $\mathbf{1}_{R_{\max}} = [1, \ldots, 1]^T \in \mathbb{R}^{R_{\max}}$.

Lois *a priori* pour les signatures génétiques projetées

Les lois *a priori* choisies pour les signatures projetées $\{\mathbf{t}_r\}_{r=1,\ldots,R}$ doivent permettre de respecter les contraintes de positivité (1.3) des facteurs $\{\mathbf{m}_r\}_{r=1,\ldots,R}$. Une loi normale multivariée $\mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right)$, de vecteur moyenne $\mathbf{e}_r$ et de matrice de covariance $s_r^2\mathbf{I}_{R-1}$, tronquée à un sous-espace $\mathcal{T}_r$ (défini par (2.6)) est donc choisie comme loi *a priori* pour chaque projection de facteur $\mathbf{t}_r$ $(r = 1, \ldots, R)$:

$$\mathbf{t}_r|\mathbf{e}_r, s_r^2, R \sim \mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right). \tag{2.5}$$

La densité de probabilité (d.d.p.) $\Phi_{\mathcal{T}_r}(\cdot)$ de cette loi normale multivariée tronquée est définie par :

$$\Phi_{\mathcal{T}_r}\left(\mathbf{t}_r|\mathbf{e}_r, s_r^2, R\right) \propto \Phi\left(\mathbf{t}_r|\mathbf{e}_r, s_r^2, R\right)\mathbf{1}_{\mathcal{T}_r}\left(\mathbf{t}_r\right),$$

en notant $\Phi\left(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ la d.d.p. de la loi normale multivariée de moyenne $\boldsymbol{\mu}$ et de matrice de covariance $\boldsymbol{\Sigma}$, et où $\mathbf{1}_{\mathcal{T}_r}(\cdot)$ est la fonction indicatrice définie sur l'ensemble $\mathcal{T}_r$:

$$\mathbf{1}_{\mathcal{T}_r}(\mathbf{x}) = \begin{cases} 1, & \text{si } \mathbf{x} \in \mathcal{T}_r, \\ 0, & \text{sinon.} \end{cases}$$

La troncature sur l'ensemble $\mathcal{T}_r \in \mathcal{V}_{R-1}$ assure la positivité des coefficients des signatures (*loadings*) introduite dans (1.3), c'est-à-dire que $\mathcal{T}_r$ est défini par :

$$\{m_{g,r} \geq 0,\ \forall g = 1, \ldots, G\} \quad \Leftrightarrow \quad \{\mathbf{t}_r \in \mathcal{T}_r\} \tag{2.6}$$

ou encore par les inégalités suivantes :

$$\mathcal{T}_r = \left\{ \mathbf{t}_r \,|\, \sum_{k=1}^{R_{\max}-1} u_{g,k} t_{k,r} + \bar{y}_g \right\}. \tag{2.7}$$

Les vecteurs moyennes $\{\mathbf{e}_r\}_{r=1,\ldots,R}$ sont fixés comme étant les solutions d'un algorithme d'extraction de pôles de mélange dédié à l'imagerie hyperspectrale, par exemple N-FINDR [Win99] ou l'algorithme *vertex component analysis* (VCA) [NBD05]. Les variances s_r sont fixées à de grandes valeurs ($s_1^2 = \ldots = s_R^2 = 100$).

En supposant que les facteurs projetés sont tous *a priori* indépendants, la loi jointe *a priori* pour la matrice des facteurs projetés $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_R]$ s'écrit :

$$f\left(\mathbf{T}|\mathbf{E}, \mathbf{s}^2, R\right) \propto \prod_{r=1}^{R} \exp\left[-\frac{\|\mathbf{t}_r - \mathbf{e}_r\|^2}{2s_r^2}\right] \mathbf{1}_{\mathcal{T}_r}(\mathbf{t}_r) \tag{2.8}$$

où $\propto$ signifie "proportionnel à", $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_R]$ et $\mathbf{s}^2 = \left[s_1^2, \ldots, s_R^2\right]$.

Loi *a priori* pour les scores

Pour chaque échantillon $i = 1, \ldots, N$, la contrainte de somme-à-un (1.3) imposée sur le vecteur des scores $\mathbf{a}_i$ permet de réécrire ce vecteur de la manière suivante :

$$\mathbf{a}_i = \begin{bmatrix} \mathbf{a}_{1:R-1,i} \\ a_{R,i} \end{bmatrix} \quad \text{où} \quad \mathbf{a}_{1:R-1,i} = \begin{bmatrix} a_{1,i} \\ \vdots \\ a_{R-1,i} \end{bmatrix} \quad \text{et} \quad a_{R,i} = 1 - \sum_{r=1}^{R-1} a_{r,i}. \tag{2.9}$$

Notons ici que n'importe quel élément du vecteur des scores $\mathbf{a}_i$ peut être retiré de celui-ci et exprimé en fonction des autres : $a_{r,i} = 1 - \sum_{k \neq r} a_{k,i}$, $\forall r \in \{1, R\}$. Pour des raisons de simplicité de notations, nous avons choisi que ce soit le dernier $a_{R,i}$. Cependant, l'algorithme proposé dans le paragraphe suivant supprimera aléatoirement un des R coefficients du vecteur $\mathbf{a}_i$ à chaque itération.

Afin de satisfaire la contrainte de positivité, les N sous-vecteurs $\mathbf{a}_{1:R-1,i}$ doivent appartenir à l'espace $\mathcal{S}$ défini par :

$$\mathcal{S} = \{\mathbf{a}_{1:R-1,i} \mid \|\mathbf{a}_{1:R-1,i}\|_1 \leq 1 \,\text{et}\, \mathbf{a}_i \succeq \mathbf{0}\}, \tag{2.10}$$

où $\|\cdot\|_1$ est la norme l_1 ($\|\mathbf{a}_i\|_1 = \sum_{r=1}^{R} |a_{r,i}|$) et $\mathbf{a}_i \succeq \mathbf{0}$ représente l'ensemble des inégalités $\{a_{r,i} \geq 0\}_{r=1,\ldots,R}$.

En suivant l'approche décrite dans [DTC08], nous avons choisi de prendre comme loi *a priori* pour les sous-vecteurs $\mathbf{a}_{1:R-1,i}$ $(i = 1, \ldots, N)$ des lois uniformes sur l'ensemble $\mathcal{S}$:

$$f\left(\mathbf{a}_{1:R-1,i}|R\right) = \mathbf{1}_{\mathcal{S}}\left(\mathbf{a}_{1:R-1,i}\right). \tag{2.11}$$

Remarquons que le fait de choisir une loi uniforme sur $\mathcal{S}$ comme loi *a priori* pour le sous-vecteur $\mathbf{a}_{1:R-1,i}$ $(i = 1, \ldots, N)$ revient à choisir une loi de Dirichlet $\mathcal{D}_R\left(1, \ldots, 1\right)$ pour le vecteur global des scores $\mathbf{a}_i$. Cette caractéristique sera développée plus en détail dans le chapitre 4 lors de la prise en compte de la dépendance temporelle.

Loi *a priori* pour la variance du bruit

Une loi *a priori* conjuguée est choisie pour la variance des erreurs résiduelles σ^2, c'est-à-dire une loi inverse-gamma de paramètres $\nu/2$ et $\gamma/2$:

$$\sigma^2|\nu, \gamma \sim \mathcal{IG}\left(\frac{\nu}{2}, \frac{\gamma}{2}\right), \tag{2.12}$$

où $\mathcal{IG}\left(\alpha, \beta\right)$ désigne une loi inverse-gamma de paramètres α et β. L'hyperparamètre ν (paramètre de forme) sera fixé à $\nu = 2$ alors que l'hyperparamètre γ (paramètre d'échelle) sera ajustable (comme dans [PADF02] ou [DTC08]). La qualité de l'estimation dépend de la valeur de l'hyperparamètre γ. N'ayant aucune connaissance sur cet hyperparamètre, nous proposons une loi non-informative de Jeffrey comme loi *a priori* pour cet hyperparamètre γ :

$$f\left(\gamma\right) \propto \frac{1}{\gamma}\mathbf{1}_{\mathbb{R}^+}\left(\gamma\right).$$

Résumé des lois *a priori* du modèle uBLU

Dans les paragraphes précédents, nous avons défini des lois *a priori* pour chacun des paramètres inconnus du modèle uBLU étudié. Ces lois *a priori* ainsi que les contraintes associées aux paramètres sont résumées ci-dessous :

$$
\begin{aligned}
\mathrm{P}[R=k] &= \tfrac{1}{R_{\max}-1} \quad \text{pour} \quad R=2,\ldots,R_{\max} && (2.3)\\
\mathbf{m}_r &= \mathbf{U}\mathbf{t}_r + \bar{\mathbf{y}} && (2.4)\\
\mathbf{t}_r|\mathbf{e}_r, s_r^2, R &\sim \mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right) && (2.5) \text{ et } (2.6)\\
a_{R,i} &= 1-\textstyle\sum_{r=1}^{R-1} a_{r,i} && (2.9)\\
\mathbf{a}_{1:R-1,i}|R &\sim \mathcal{U}_{\mathcal{S}}\left(\mathbf{a}_{1:R-1,i}\right) && (2.10) \text{ et } (2.11)\\
\sigma^2|\nu,\gamma &\sim \mathcal{IG}\left(\tfrac{\nu}{2},\tfrac{\gamma}{2}\right) && (2.12)
\end{aligned}
$$

De plus, afin d'estimer l'hyperparamètre γ, nous avons choisi d'introduire un deuxième niveau d'inférence bayésienne conformément aux modèles bayésiens hiérarchiques. Cette structure hiérarchique est résumée dans le graphe acyclique orienté (ou DAG, pour *directed acyclic graph*) de la figure 2.1.

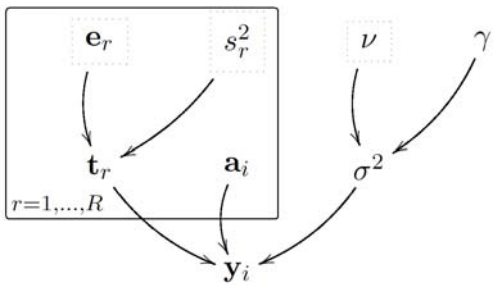


FIGURE 2.1 – DAG pour les lois *a priori* des paramètres et hyperparamètres du modèle bayésien uBLU (les paramètres fixés apparaissent dans les cases en pointillés).

2.2.3 Loi *a posteriori*

Notons $\boldsymbol{\Theta} = \{\mathbf{M}, \mathbf{A}, R, \sigma^2\}$ le vecteur des paramètres inconnus pour lesquels ont été définis précédemment des lois *a priori*. En supposant l'indépendance entre ces paramètres inconnus, on obtient :

$$f\left(\boldsymbol{\Theta}|\gamma\right) = f\left(R\right) f\left(\mathbf{T}|\mathbf{E},\mathbf{s}^2,R\right) f\left(\mathbf{A}|R\right) f\left(\sigma^2|\nu,\gamma\right) \tag{2.13}$$

où $f\left(R\right)$, $f\left(\mathbf{T}|\mathbf{E},\mathbf{s}^2,R\right)$, $f\left(\mathbf{A}|R\right)$ et $f\left(\sigma^2|\nu,\gamma\right)$ sont respectivement les lois *a priori* du nombre de signatures R, de la matrice des signatures projetées $\mathbf{T}$, de la matrice des scores $\mathbf{A}$ et de la variance des erreurs résiduelles σ^2 définies précédemment dans (2.3), (2.8), (2.11) et (2.12).

En multipliant la fonction de vraisemblance $f\left(\mathbf{Y}|\boldsymbol{\Theta}\right)$ (2.2) par la loi jointe $f\left(\boldsymbol{\Theta}|\gamma\right)$ (2.13), et en intégrant l'hyperparamètre γ, on peut écrire la loi *a posteriori* du vecteur $\boldsymbol{\Theta}$ des paramètres inconnus :

$$\begin{aligned} f\left(\boldsymbol{\Theta}|\mathbf{Y}\right) &= \int f\left(\boldsymbol{\Theta},\gamma|\mathbf{Y}\right) d\gamma \\ &\propto \int f\left(\mathbf{Y}|\boldsymbol{\Theta}\right) f\left(\boldsymbol{\Theta}|\gamma\right) f\left(\gamma\right) d\gamma \end{aligned}$$

ou encore :

$$\begin{aligned} f\left(\mathbf{T},\mathbf{A},R,\sigma^2|\mathbf{Y}\right) \propto & \prod_{r=1}^{R} \exp\left[-\frac{\|\mathbf{t}_r-\mathbf{e}_r\|^2}{2s_r^2}\right] \mathbf{1}_{\mathcal{T}_r}\left(\mathbf{t}_r\right) \\ & \times \prod_{i=1}^{N} \mathbf{1}_{\mathcal{S}}\left(\mathbf{a}_{1:R-1,i}\right) \\ & \times \frac{1}{R_{\max}-1} \\ & \times \prod_{i=1}^{N} \left(\frac{1}{\sigma^2}\right)^{\frac{G}{2}-1} \exp\left[-\frac{\|\mathbf{y}_i-\left(\mathbf{UT}+\bar{\mathbf{y}}\mathbf{1}_R^T\right)\mathbf{a}_i\|^2}{2\sigma^2}\right]. \end{aligned} \tag{2.14}$$

Les contraintes imposées sur les paramètres rendent la loi *a posteriori* $f\left(\boldsymbol{\Theta}|\mathbf{Y}\right)$ beaucoup trop complexe pour obtenir une expression simple des estimateurs bayésiens classiques, comme l'estimateur du maximum *a posteriori* (MAP) ou l'estimateur qui minimise l'erreur quadratique moyenne appelé estimateur MMSE (pour *minimum mean square error*). Dans de tels cas, il est usuel d'utiliser des méthodes de Monte Carlo par chaînes de Markov (MCMC) [GRS96] afin de générer des échantillons $\mathbf{M}^{(\ell)}$, $\mathbf{A}^{(\ell)}$, $R^{(\ell)}$ et $\sigma^{2(\ell)}$ asymptotiquement distribués selon la loi *a posteriori* (ℓ désigne l'indice de l'échantillon : $\ell = 1, 2, \ldots$).

Rappelons ici que les dimensions des matrices des signatures $\mathbf{M}$ et des scores $\mathbf{A}$ dépendent du nombre R inconnu de signatures. Ainsi, échantillonner suivant $f\left(\mathbf{M},\mathbf{A},R,\sigma^2|\mathbf{Y}\right)$ nécessite l'exploration d'espaces de dimensions différentes (modèles différents). Afin de résoudre ce problème de sélection de modèles, un processus de naissance et de mort est inclus dans le schéma classique MCMC, à la manière d'un RJ-MCMC (voir [Gre95] pour plus de détails sur les RJ-MCMC). Un algorithme de Gibbs est alors utilisé afin de déterminer les autres paramètres inconnus du modèle uBLU.

2.3 Algorithme de Gibbs hybride combiné à un processus de naissance et de mort

Cette partie présente les différentes étapes de l'algorithme de Gibbs hybride combiné à un processus de naissance et de mort permettant de générer des échantillons distribués asymptotiquement suivant la loi jointe *a posteriori* des paramètres inconnus $\boldsymbol{\Theta}$ (2.13) (voir aussi l'algorithme 2.1).

L'organigramme 2.2 représente schématiquement les grandes étapes de l'algorithme. A chaque itération de l'algorithme, le nombre de signatures R est d'abord mis à jour à l'aide d'un processus de naissance et de mort, qui sera détaillé dans le paragraphe 2.3.1. Puis, conditionnellement à ce nombre R, les autres paramètres inconnus sont mis à jour à l'aide d'un échantillonneur de Gibbs.

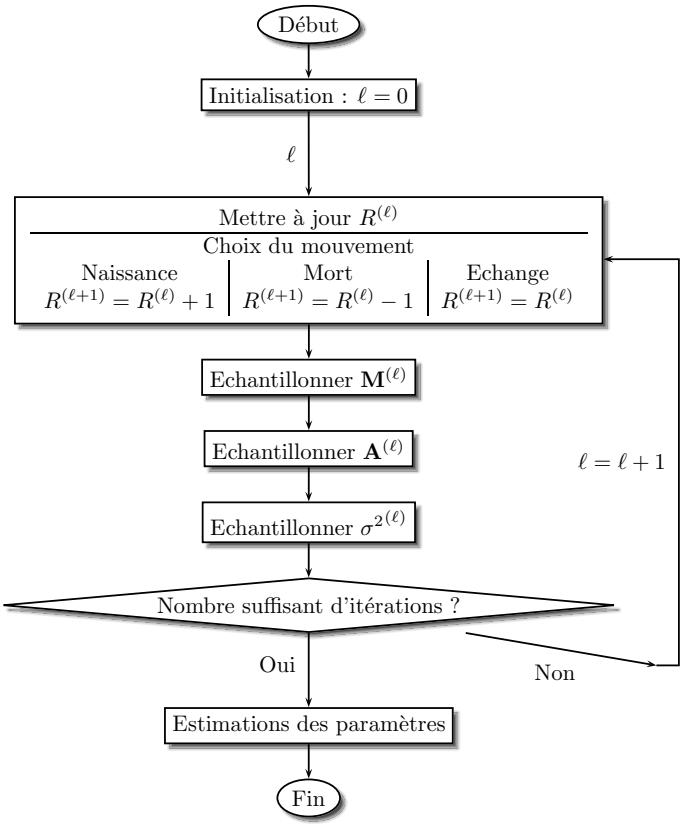


FIGURE 2.2 – Organigramme de l'algorithme bayésien uBLU pour le démélange de données génétiques.

ALGO. 2.1 – Echantillonneur de Gibbs hybride combiné à un processus de naissance et de mort, utilisé pour le démélange bayésien de données d'expression de gènes.

- Pré-traitements :
  - Calculer la moyenne empirique des échantillons $\bar{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{y}_i$ définie dans (2.4),
  - Déterminer la matrice de projection $\mathbf{P}$ (de taille $R_{\max} - 1 \times G$) à l'aide d'une technique de réduction de dimensionnalité (comme l'ACP [Jol86]),
  - Choisir les vecteurs moyennes $\{\mathbf{e}_r\}_{r=1,\ldots,R_{\max}}$ définis dans (2.5) comme les solutions d'un algorithme d'extraction de pôles de mélange (comme l'algorithme VCA [NBD05]).
- Initialisation ($\ell = 0$) :
  - Générer le nombre de facteurs $R^{(0)}$ à partir de sa loi *a priori* (2.3),
  - Générer la matrice des signatures projetées $\mathbf{T}^{(0)}$ selon la loi *a priori* (2.8),
  - Reconstruire la matrice des signatures $\mathbf{M}^{(0)}$ à partir de la matrice des signatures projetées $\mathbf{T}^{(0)}$ (2.4) : $\mathbf{M}^{(0)} = \mathbf{P}^{-1}\mathbf{T}^{(0)} + \bar{\mathbf{y}}\mathbf{1}_{R^{(0)}}^T$,
  - Pour $i = 1, \ldots, N$, générer les scores $\mathbf{a}_i^{(0)}$ à partir de (2.9) et (2.11),
  - Générer la variance du bruit $\sigma^{2(0)}$ selon la loi *a priori* (2.12).
  - Poser $\ell \leftarrow 1$.
- Itérations : Pour $\ell = 1, 2, \ldots, \mathrm{N_{mc}}$, faire :
  1. Mettre à jour le nombre de facteurs $R^{(\ell)}$:
     - Tirer $p \sim \mathcal{U}_{[0,1]}$ (p : probabilité de choisir un des trois mouvements possibles),
     - SI $p \leq n_{R^{(\ell-1)}}$, ALORS

       proposer un mouvement de *naissance* (voir l'algorithme 2.2),
     - SINON SI $n_{R^{(\ell-1)}} < p \leq n_{R^{(\ell-1)}} + m_{R^{(\ell-1)}}$, ALORS

       proposer un mouvement de *mort* (voir l'algorithme 2.3),
     - SINON

       proposer un mouvement d'*échange* (voir l'algorithme 2.4),
     - Tirer $u \sim \mathcal{U}_{[0,1]}$,
     - SI $u < \rho$ (ρ : probabilité d'acceptation du mouvement choisi, voir (2.15)), ALORS

       poser $\left(\mathbf{M}^{(\ell)}, \mathbf{A}^{(\ell)}, R^{(\ell)}\right) = (\mathbf{M}^\star, \mathbf{A}^\star, R^\star)$ (mouvement accepté),
     - SINON

       poser $\left(\mathbf{M}^{(\ell)}, \mathbf{A}^{(\ell)}, R^{(\ell)}\right) = \left(\mathbf{M}^{(\ell-1)}, \mathbf{A}^{(\ell-1)}, R^{(\ell-1)}\right)$ (mouvement rejeté),
  2. Echantillonner $\mathbf{T}^{(\ell)}$ selon la loi (2.16),
  3. Construire $\mathbf{M}^{(\ell)} = \mathbf{P}^{-1}\mathbf{T}^{(\ell)} + \bar{\mathbf{y}}\mathbf{1}_{R^{(\ell)}}^T$ (2.4),
  4. Echantillonner $\mathbf{A}^{(\ell)}$ selon la loi (2.17),
  5. Echantillonner $\sigma^{2(\ell)}$ selon la loi (2.18),
  6. Poser $\ell \leftarrow \ell + 1$.

2.3.1 Mise à jour du nombre R de facteurs

La mise à jour du nombre de facteurs R (et le changement de dimensions) est effectuée à l'aide d'un processus de naissance et de mort. Plus précisément, à chaque itération ℓ de l'algorithme, un mouvement de *naissance*, de *mort* ou d'*échange* est aléatoirement choisi avec les probabilités $n_{R^{(\ell)}}$, $m_{R^{(\ell)}}$ et $e_{R^{(\ell)}}$. Bien entendu, ces probabilités respectent les conditions suivantes :

- les probabilités de changement de mouvement somment à 1 : $n_{R^{(\ell)}} + m_{R^{(\ell)}} + e_{R^{(\ell)}} = 1$,
- il ne peut pas y avoir de mouvement de mort lorsque $R = 2$: $m_2 = 0$,
- il ne peut pas y avoir de mouvement de naissance lorsque $R = R_{\max}$: $m_{R_{\max}} = 0$,

et donc :

$$n_{R^{(\ell)}} = m_{R^{(\ell)}} = e_{R^{(\ell)}} = \tfrac{1}{3} \quad \text{pour} \quad R^{(\ell)} = 2, \ldots, R_{\max} - 1,$$
$$n_1 = m_{R_{\max}} = e_1 = e_{R_{\max}} = \tfrac{1}{2}.$$

Les mouvements de *naissance* et de *mort* consistent en l'augmentation ou la diminution du nombre de facteurs R de 1, alors que le mouvement d'*échange* ne change pas la dimension de R.

Considérons qu'à l'itération ℓ, un mouvement nous permettent de passer de l'état $\{\mathbf{M}^{(\ell)}, \mathbf{A}^{(\ell)}, R^{(\ell)}\}$ au nouvel état $\{\mathbf{M}^\star, \mathbf{A}^\star, R^\star\}$. Les trois types de mouvements sont définis de la même manière que ceux utilisés dans [DTC08] pour l'imagerie hyperspectrale à la différence près que dans [DTC08] les signatures étaient choisies parmi une bibliothèque de $R_{\max}$ spectres possibles (approche semi-supervisée) et qu'ici nous construisons également les signatures inconnues sans l'utilisation d'une bibliothèque (approche totalement non-supervisée). Les trois mouvements sont rappelés ici.

Mouvement de *naissance*

Lorsqu'un mouvement de *naissance* est proposé (avec la probabilité $n_{R^{(\ell)}}$), la dimension de l'espace de travail est augmentée : $R^\star = R^{(\ell)} + 1$. Une nouvelle signature $\mathbf{m}^\star$ est générée aléatoirement afin de construire une nouvelle matrice des facteurs $\mathbf{M}^\star = \left[\mathbf{M}^{(\ell)},\ \mathbf{m}^\star\right]$. Cette nouvelle signature est supposée être assez différente et séparée des autres déjà existantes. La matrice des scores $\mathbf{A}^{(\ell)}$ est également mise à jour : un nouveau coefficient est généré pour chaque vecteur $\mathbf{a}_i$ $(i = 1, \ldots, N)$ suivant une loi

Beta $\mathcal{B}\left(\frac{1}{N}, R^{(\ell)}\right)$, et la nouvelle matrice des scores $\mathbf{A}^\star$ est re-normalisée afin de respecter la contrainte de somme-à-un.

ALGO. 2.2 – Mouvement de *naissance*.

---

1. Poser $R^\star = R^{(\ell)} + 1$,
2. Construire une nouvelle signature projetée $\mathbf{t}^\star$ selon (2.5),
3. Vérifier que $\mathbf{t}^\star \neq \mathbf{t}_r^{(\ell)}$, $\forall r = 1, \ldots, R^{(\ell)}$,
4. Reconstruire la signature correspondante $\mathbf{m}^\star$ selon (2.4) : $\mathbf{m}^\star = \mathbf{P}^{-1}\mathbf{t}^\star + \bar{\mathbf{y}}$,
5. Ajouter $\mathbf{m}^\star$ à la matrice des signatures $\mathbf{M}^\star$,
   i.e. poser $\mathbf{M}^\star = \left[\mathbf{M}^{(\ell)},\ \mathbf{m}^\star\right]$,
6. Tirer $a^\star \sim \mathcal{B}\left(\frac{1}{N}, R^{(\ell)}\right)$,
7. Ajouter $a^\star$ à $\mathbf{a}_i^{(\ell)}$ (pour $i = 1, \ldots, N$) et re-normaliser les autres scores,
   i.e. poser $\mathbf{A}^\star = \left[\mathbf{A}^{(\ell)}\left(1 - a^\star\right),\ a^\star \mathbf{1}_N^T\right]$.

Mouvement de *mort*

Lorsqu'un mouvement de *mort* est proposé (avec la probabilité $m_{R^{(\ell)}}$), une des signatures de $\mathbf{M}^{(\ell)}$ est aléatoirement supprimée. Il en est de même pour les coefficients de $\mathbf{A}^{(\ell)}$ correspondants. Les scores restants formant la matrice $\mathbf{A}^\star$ sont re-normalisés pour sommer à 1.

ALGO. 2.3 – Mouvement de *mort*.

---

1. Poser $R^\star = R^{(\ell)} - 1$,
2. Tirer $k \sim \mathcal{U}_{\{1,\ldots,R^{(\ell)}\}}$,
3. Enlever $\mathbf{m}_k^{(\ell)}$ de la matrice des facteurs $\mathbf{M}^{(\ell)}$,
   i.e. poser $\mathbf{M}^\star = \left[\mathbf{m}_1^{(\ell)}, \ldots, \mathbf{m}_{k-1}^{(\ell)}, \mathbf{m}_{k+1}^{(\ell)}, \ldots, \mathbf{m}_{R^{(\ell)}}^{(\ell)}\right]$,
4. Tirer $k' \sim \mathcal{U}_{\{1,\ldots,k-1,k+1,\ldots,R^{(\ell)}\}}$,
5. Répartir les scores correspondants $\mathbf{a}_k^{(\ell)}$ sur les scores du $(k')^{\text{ème}}$ facteur : $\mathbf{a}_{k'}^{(\ell)} = \mathbf{a}_{k'}^{(\ell)} + \mathbf{a}_k^{(\ell)}$,
6. Enlever $\mathbf{a}_k^{(\ell)}$ de la matrice des scores $\mathbf{A}^{(\ell)}$,
   i.e. poser $\mathbf{A}^\star = \left[\mathbf{a}_1^{(\ell)}, \ldots, \mathbf{a}_{k-1}^{(\ell)}, \mathbf{a}_{k+1}^{(\ell)}, \ldots, \mathbf{a}_{R^{(\ell)}}^{(\ell)}\right]$.

Mouvement d'*échange*

Lorsqu'un mouvement d'*échange* est proposé (avec la probabilité $e_{R^{(\ell)}}$), une signature $\mathbf{m}^\star$ est aléatoirement choisie dans $\mathbf{M}^{(\ell)}$ et remplacée par une nouvelle signature aléatoirement générée. Si la nouvelle signature est trop proche d'une signature déjà existante, les scores correspondants sont proportionnellement distribués parmi ses plus proches voisins. Remarquons que l'utilisation d'un mouvement d'échange consiste en fait à créer une nouvelle signature (mouvement de *naissance*) et à en supprimer une autre (mouvement de *mort*) dans une seule et même étape.

ALGO. 2.4 – Mouvement d'*échange*.

---

1. Poser $R^\star = R^{(\ell)}$,
2. Tirer $k \sim \mathcal{U}_{\{1,\dots,R^{(\ell)}\}}$,
3. Construire une nouvelle signature projetée $\mathbf{t}^\star$ selon (2.5),
4. Vérifier que $\mathbf{t}^\star \neq \mathbf{t}_r^{(\ell)}$, $\forall r = 1, \dots, R^{(\ell)}$,
5. Reconstruire la signature correspondante $\mathbf{m}^\star$ selon (2.4) : $\mathbf{m}^\star = \mathbf{P}^{-1}\mathbf{t}^\star + \bar{\mathbf{y}}$,
6. Remplacer $\mathbf{m}_i^{(\ell)}$ dans la matrice des signatures $\mathbf{M}^\star$,
   i.e. poser $\mathbf{M}^\star = \left[\mathbf{m}_1^{(\ell)}, \dots, \mathbf{m}_{k-1}^{(\ell)}, \mathbf{m}^\star, \mathbf{m}_{k+1}^{(\ell)}, \dots, \mathbf{m}_{R^{(\ell)}}^{(\ell)}\right]$,
7. Poser $\mathbf{A}^\star = \mathbf{A}^{(\ell)}$.

Chacun de ces trois types de mouvements est ensuite accepté ou rejeté selon une probabilité d'acceptation/rejet empirique ρ, le rapport des fonctions de vraisemblance entre le nouvel état proposé et l'état actuel :

$$\rho = \min(1, A) \quad \text{avec}$$
$$A = \frac{f\left(\mathbf{Y}|\mathbf{M}^\star, \mathbf{A}^\star, R^\star, \sigma^{2(\ell)}\right)}{f\left(\mathbf{Y}|\mathbf{M}^{(\ell)}, \mathbf{A}^{(\ell)}, R^{(\ell)}, \sigma^{2(\ell)}\right)} = \exp\left[-\frac{\|\mathbf{Y}-\mathbf{M}^\star\mathbf{A}^\star\|^2 - \left\|\mathbf{Y}-\mathbf{M}^{(\ell)}\mathbf{A}^{(\ell)}\right\|^2}{2\sigma^{2(\ell)}}\right] \tag{2.15}$$

Après avoir fait cette procédure de naissance et de mort pour mettre à jour le nombre R de facteurs, et définir ainsi les dimensions de l'espace dans lequel on travaille, les facteurs $\mathbf{M}$, les scores $\mathbf{A}$ et la variance du bruit σ^2 sont générés, conditionnellement au nombre R, suivant les lois détaillées dans les paragraphes suivants.

2.3.2 Echantillonnage suivant $f\left(\mathbf{T}|\mathbf{A}, R, \sigma^2, \mathbf{Y}\right)$

Le théorème de Bayes nous permet d'écrire :

$$f\left(\mathbf{T}|\mathbf{A}, R, \sigma^2, \mathbf{Y}\right) \propto f\left(\mathbf{Y}|\mathbf{\Theta}\right) f\left(\mathbf{T}|\mathbf{E}, \mathbf{s}^2, R\right).$$

Notons $\mathbf{T}_{\backslash r}$ la matrice $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_{r-1}, \mathbf{t}_{r+1}, \ldots, \mathbf{t}_R]$ privée de sa $r^{\text{ème}}$ colonne. La loi conditionnelle de $\mathbf{t}_r$ est une loi gaussienne multivariée tronquée sur $\mathcal{T}_r$ défini dans (2.6) :

$$\mathbf{t}_r|\mathbf{T}_{\backslash r}, \mathbf{a}_r, R, \sigma^2, \mathbf{Y} \sim \mathcal{N}_{\mathcal{T}_r}\left(\boldsymbol{\tau}_r, \mathbf{\Gamma}_r\right), \tag{2.16}$$

où :

$$\begin{cases} \mathbf{\Gamma}_r &= \left[\sum_{i=1}^N a_{r,i}^2 \mathbf{P}\mathbf{\Sigma}^{-1}\mathbf{P}^T + \frac{1}{s_r^2}\mathbf{I}_R\right]^{-1}, \\ \boldsymbol{\tau}_r &= \mathbf{\Gamma}_r\left[\sum_{i=1}^N a_{r,i}\mathbf{P}\mathbf{\Sigma}^{-1}\boldsymbol{\epsilon}_{r,i} + \frac{1}{s_r^2}\mathbf{e}_r\right], \\ \boldsymbol{\epsilon}_{r,i} &= \mathbf{y}_i - a_{r,i}\bar{\mathbf{y}} - \sum_{j\neq r} a_{r,i}\mathbf{m}_j. \end{cases}$$

La principale difficulté pour générer des échantillons suivant cette loi vient de la troncature au sous-espace $\mathcal{T}_r$ permettant d'assurer la positivité des signatures. Pour résoudre ce problème, nous générons itérativement chaque composante $t_{k,r}$ $(k = 1, \ldots, R-1)$ du vecteur $\mathbf{t}_r$ conditionnellement aux autres composantes $\mathbf{t}_{\backslash k,r} = \{t_{j,r}\}_{j\neq k}$ suivant la loi doublement tronquée suivante [DMC+09] :

$$t_{k,r}|\mathbf{t}_{\backslash k,r}, \mathbf{T}_{\backslash r}, \mathbf{a}_{[1:R]\backslash k,i}, \sigma^2, R, \mathbf{Y} \sim \mathcal{N}_{\left[t_{k,r}^-, t_{k,r}^+\right]}\left(w_{k,r}, z_{k,r}^2\right),$$

avec :

$$\begin{cases} t_{k,r}^- = \max_{g\,;u_{g,k}>0} - \dfrac{\varepsilon_{g,k,r}}{u_{g,k}}, \\ t_{k,r}^+ = \max_{g\,;u_{g,k}<0} - \dfrac{\varepsilon_{g,k,r}}{u_{g,k}}, \\ \varepsilon_{g,k,r} = \bar{\mathbf{y}}_g + \displaystyle\sum_{j\neq k} u_{g,j}t_{j,r}, \end{cases}$$

et où $w_{k,r}$ et $z_{k,r}$ sont respectivement les moyennes et variances conditionnelles. La génération d'échantillons suivant une loi gaussienne doublement tronquée peut être facilement réalisée en suivant l'algorithme décrit dans [Rob95].

2.3.3 Echantillonnage suivant $f(\mathbf{A}|\mathbf{T}, \sigma^2, R, \mathbf{Y})$

Le théorème de Bayes nous donne, pour chaque échantillon i $(i = 1, \ldots, N)$:

$$f\left(\mathbf{a}_{1:R-1,i}|\mathbf{T}, \sigma^2, R, \mathbf{Y}\right) \propto f\left(\mathbf{Y}|\mathbf{\Theta}\right) f\left(\mathbf{a}_{1:R-1,i}\right).$$

Après quelques calculs, on en déduit que la loi conditionnelle *a posteriori* de chaque sous-vecteur des scores $\mathbf{a}_{1:R-1,i}$ est une loi normale multivariée tronquée sur le simplexe $\mathcal{S}$ défini précédemment (2.10) :

$$\mathbf{a}_{1:R-1,i}|\mathbf{T}, \sigma^2, R, \mathbf{y}_i \sim \mathcal{N}_{\mathcal{S}}\left(\mu_i, \Sigma_i\right), \tag{2.17}$$

avec :

$$\begin{cases} \Sigma_i = \left[\left(\mathbf{M}_{\backslash R} - \mathbf{m}_R \mathbf{1}_{R-1}^T\right)^T \mathbf{\Sigma}^{-1} \left(\mathbf{M}_{\backslash R} - \mathbf{m}_R \mathbf{1}_{R-1}^T\right)\right]^{-1}, \\ \mu_i = \Sigma_i \left[\left(\mathbf{M}_{\backslash R} - \mathbf{m}_R \mathbf{1}_{R-1}^T\right)^T \mathbf{\Sigma}^{-1} \left(\mathbf{y}_i - \mathbf{m}_R\right)\right], \end{cases}$$

et $\mathbf{1}_{R-1} = [1, \ldots, 1]^T \in \mathbb{R}^{R-1}$. La génération d'échantillons suivant une loi gaussienne multivariée tronquée peut s'effectuer en suivant une procédure d'acceptation/rejet comme celle décrite dans [Rob95].

2.3.4 Echantillonnage suivant $f(\sigma^2|\mathbf{T}, \mathbf{A}, R, \mathbf{Y})$

En utilisant la fonction de vraisemblance (2.2) et la loi *a priori* (2.12), on peut facilement montrer que la loi conditionnelle *a posteriori* $f\left(\sigma^2|\mathbf{T}, \mathbf{A}, \mathbf{Y}\right)$ de la variance du bruit σ^2 est la loi inverse-gamma suivante :

$$\sigma^2|\mathbf{T}, \mathbf{A}, \mathbf{Y} \sim \mathcal{IG}\left(\frac{GN}{2}, \frac{1}{2}\sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\|^2\right). \tag{2.18}$$

2.4 Contrôle de la convergence

Un problème inhérent à l'utilisation de méthodes MCMC est de pouvoir déterminer si les chaînes de Markov construites ont bien convergé. En effet, rappelons que l'échantillonneur de Gibbs défini dans la section précédente permet de générer des échantillons $\{\mathbf{\Theta}^{(\ell)}\}_{\ell=1,\ldots,\mathrm{N_{mc}}}$ qui sont asymptotiquement distribués suivant la loi jointe *a posteriori* $f\left(\mathbf{\Theta}|\mathbf{Y}\right)$. Les échantillons n'appartenant pas à la période de

chauffage (les premières itérations) sont ensuite utilisés pour estimer les paramètres inconnus. Dans un premier temps, il est important d'estimer le nombre de signatures afin de déterminer le modèle le plus adapté aux données. La stratégie utilisée dans cette thèse consiste à utiliser une estimation MAP du nombre de signatures :

$$\widehat{R} = \underset{k\in\{2,\ldots,R_{\max}\}}{\operatorname{argmax}} \mathrm{P}\left[R=k|\mathbf{Y}\right] \approx \underset{k\in\{2,\ldots,R_{\max}\}}{\operatorname{argmax}} \frac{\mathrm{N}_k}{\mathrm{N}_\mathrm{r}}, \tag{2.19}$$

où N_k est le nombre de valeurs de R générées parmi $R^{(\mathrm{N_{bi}}+1)},\ldots,R^{(\mathrm{N_{mc}})}$ satisfaisant $R^{(\ell)}=k$ et $\mathrm{N_r}=\mathrm{N_{mc}}-\mathrm{N_{bi}}$ (nombre d'échantillons réellement utilisés pour l'estimation). Dans une seconde étape, conditionnellement à $\widehat{R}$, nous faisons une estimation MAP jointe $\left(\widehat{\mathbf{M}},\widehat{\mathbf{A}}\right)$ des matrices des signatures et des scores :

$$\left(\widehat{\mathbf{M}},\widehat{\mathbf{A}}\right) = \underset{\ell=\mathrm{N_{bi}}+1,\ldots,\mathrm{N_{mc}}}{\operatorname{argmax}} f\left(\mathbf{M}^{(\ell)},\mathbf{A}^{(\ell)}|\mathbf{Y},R=\widehat{R}\right). \tag{2.20}$$

Vérifier la convergence d'une méthode d'échantillonnage se ramène à répondre à deux questions essentielles. La première est de savoir à partir de quel moment on peut affirmer que les échantillons générés $\{\mathbf{\Theta}^{(\ell)}\}$ sont bien distribués suivant la loi cible, ou en d'autres termes, de savoir quelle est la période de chauffage ($\mathrm{N_{bi}}$). La seconde est de savoir combien d'échantillons ($\mathrm{N_r}$) sont nécessaires pour obtenir une bonne estimation des différents paramètres. Ce paragraphe présente les critères que nous avons utilisés pour déterminer au mieux les valeurs de $\mathrm{N_{bi}}$ et $\mathrm{N_r}$.

2.4.1 Détermination de la période de chauffage ($\mathrm{N_{bi}}$)

Différentes mesures de convergence des méthodes MCMC peuvent être définies en construisant plusieurs chaînes de Markov en parallèle, initialisées de manière aléatoire [RR98]. Nous proposons ici d'utiliser un critère de variance inter- et intra-chaîne, initialement proposé par Gelman et Rubin dans [GR92] et généralisé dans [BG98]. Ce critère nécessite de construire C chaînes de Markov en parallèle de même longueur $\mathrm{N_{mc}}$ mais initialisées différemment. Plus récemment, Brooks et Giudici [BG00], puis Castelloe et Zimmerman [CZ02], ont étendu ces mêmes critères pour le cas des chaînes à sauts réversibles (RJ-MCMC) avec sélection de modèles, ce qui est plus adapté pour étudier la convergence de l'algorithme uBLU. Ce critère permet de détecter et prendre en compte :

- les variations entre les chaînes,
- les interactions entre les modèles et les chaînes (les modèles peuvent varier d'une chaîne à une autre),
- les différences qu'il peut y avoir entre le nombre de fois qu'un modèle est visité dans une chaîne ou dans une autre.

Ce diagnostic de convergence peut être étudié pour chaque paramètre inconnu. Notons θ l'un de ces paramètres et M le nombre maximal de modèles pouvant être sélectionnés pour chaque chaîne. Dans le cas de l'algorithme uBLU, le nombre M correspond au nombre maximal de signatures $R_{\max}-1$ qui peuvent être présentes dans le mélange. Les variances totale $\widehat{V}$, intra-chaîne W_{c}, intra-modèle W_{m} et intra-modèle et chaîne $W_{\mathrm{m}}W_{\mathrm{c}}$ sont définies pour le paramètre θ comme suit [1] :

$$\begin{array}{rcl}
\widehat{V}(\theta) & = & \frac{1}{C\mathrm{N_r}-1}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}\left(\theta_{cm}^{r}-\overline{\theta}_{\cdot\cdot}^{\cdot}\right)^2, \\
W_{\mathrm{c}}(\theta) & = & \frac{1}{C(\mathrm{N_r}-1)}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}\left(\theta_{cm}^{r}-\overline{\theta}_{c\cdot}^{\cdot}\right)^2, \\
W_{\mathrm{m}}(\theta) & = & \frac{1}{C\mathrm{N_r}-M}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}\left(\theta_{cm}^{r}-\overline{\theta}_{\cdot m}^{\cdot}\right)^2, \\
W_{\mathrm{m}}W_{\mathrm{c}}(\theta) & = & \frac{1}{C(\mathrm{N_r}-M)}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}\left(\theta_{cm}^{r}-\overline{\theta}_{cm}^{\cdot}\right)^2,
\end{array} \tag{2.21}$$

avec :

$$\left\{\begin{array}{rcl}
\overline{\theta}_{\cdot\cdot}^{\cdot} & = & \frac{1}{C\mathrm{N_r}}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}\theta_{cm}^{(r)}, \\
\overline{\theta}_{c\cdot}^{\cdot} & = & \frac{1}{\mathrm{N_r}}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}\theta_{cm}^{(r)}, \\
\overline{\theta}_{\cdot m}^{\cdot} & = & \frac{1}{R_{\cdot m}}\sum_{c=1}^{C}\sum_{r=1}^{R_{cm}}\theta_{cm}^{(r)}, \\
\overline{\theta}_{cm}^{\cdot} & = & \frac{1}{R_{cm}}\sum_{r=1}^{R_{cm}}\theta_{cm}^{(r)}, \\
R_{\cdot m} & = & \sum_{c=1}^{C}R_{cm},
\end{array}\right.$$

où R_{cm} est le nombre de fois où le $m^{\text{ème}}$ modèle est sélectionné dans la $c^{\text{ème}}$ chaîne et $\theta_{cm}^{(r)}$ est la $r^{\text{ème}}$ valeur du paramètre d'intérêt θ dans le $m^{\text{ème}}$ modèle et pour la $c^{\text{ème}}$ chaîne. La convergence des chaînes est alors mesurée par deux critères $PSRF_1$ et $PSRF_2$, ou potentiels d'échelle, qui se réfèrent au *potential scale reduction factor* défini dans [GCSR03] :

$$\begin{array}{rcl}
PSRF_1 & = & \frac{\widehat{V}(\theta)}{W_{\mathrm{c}}(\theta)}, \\
PSRF_2 & = & \frac{W_{\mathrm{m}}(\theta)}{W_{\mathrm{m}}W_{\mathrm{c}}(\theta)}.
\end{array} \tag{2.22}$$

1. Les indices m et c utilisés dans $W_{\mathrm{c}}(\cdot)$ et $W_{\mathrm{m}}(\cdot)$ font partie du noms des variables, et ne correspondent en aucun cas aux valeurs des indices de la partie de droite.

Pour justifier la bonne convergence de l'échantillonneur, et s'assurer ainsi qu'un nombre d'itérations de chauffe $\mathrm{N_{bi}}$ est suffisant pour obtenir des échantillons des paramètres inconnus de $\boldsymbol{\Theta}$ distribués suivant leur loi cible, il est recommandé que les valeurs des potentiels d'échelle $PSRF_1$ et $PSRF_2$ soient proches de 1 [CZ02].

2.4.2 Détermination du nombre d'itérations d'intérêt ($\mathrm{N_r}$)

La stratégie de convergence détaillée dans le paragraphe précédent permet de déterminer le nombre d'itérations $\mathrm{N_{bi}}$ de la période de chauffe. Une fois ce nombre fixé, il est nécessaire de déterminer le nombre total d'itérations $\mathrm{N_r}$ qui seront utiles pour obtenir une estimation correcte des différents paramètres de $\boldsymbol{\Theta}$ en utilisant les équations (2.19) et (2.20). Pour cela, nous utiliserons une approche *ad hoc* par visualisation graphique de l'erreur de reconstruction en fonction du nombre d'itérations ℓ :

$$\mathrm{RE}^{(\ell)} = \frac{1}{NG} \sum_{i=1}^{N} \left\| \mathbf{y}_i - \sum_{r=1}^{R^{(\ell)}} \mathbf{m}_r^{(\ell)} a_{r,i}^{(\ell)} \right\|^2 . \tag{2.23}$$

Le nombre d'itérations d'intérêt $\mathrm{N_r}$ est alors fixé à l'itération ℓ à partir de laquelle l'erreur de reconstruction $\mathrm{RE}^{(\ell)}$ semble avoir convergé ou est inférieure à un seuil donné.

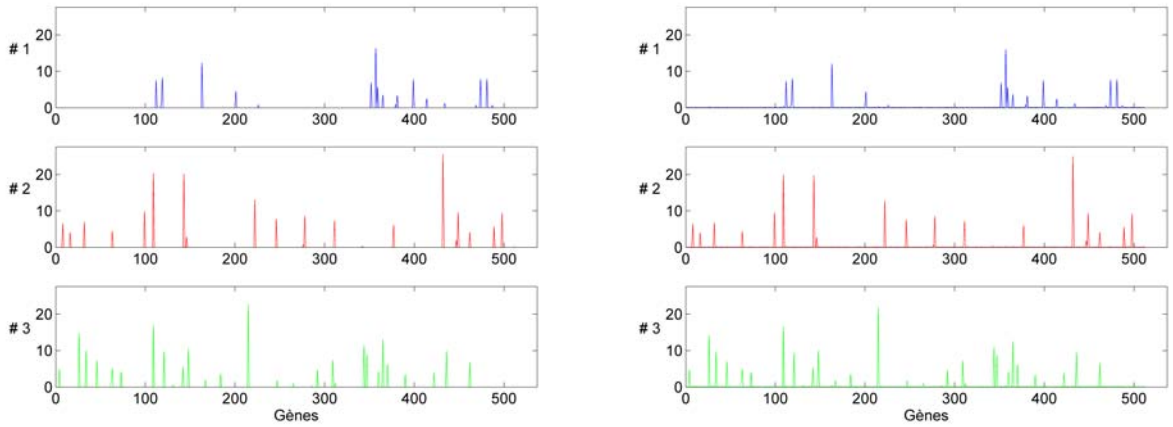
2.5 Résultats de simulations sur données synthétiques

Nous allons illustrer l'intérêt d'utiliser l'algorithme uBLU en l'appliquant dans un premier temps à des données synthétiques. Nous comparerons alors ces performances par rapport à d'autres algorithmes de décomposition factorielle.

Scénario de simulations

Les données synthétiques générées correspondent à l'expression de $G = 256$ gènes pour $N = 128$ échantillons. Chaque échantillon est composé d'exactement $R = 3$ signatures (positives), générées de manière à ce que seulement quelques gènes affectent chaque signature, selon le modèle de mélange linéaire (1.2). Les signatures sont représentées sur la figure 2.3a. Les coefficients de répartition (scores)

ont, quant à eux, été générés aléatoirement suivant une distribution de Dirichlet $\mathcal{D}(1,\ldots,1)$ afin de satisfaire aux contraintes de positivité et de somme-à-un. Chaque échantillon est corrompu par un bruit gaussien additif de rapport signal-à-bruit fixé à SNR = 20 dB, où $\text{SNR} = G^{-1}\sigma^{-2}\left\|\sum_{r=1}^{R}\mathbf{m}_r a_{r,i}\right\|^2$. Les vecteurs moyennes $\mathbf{e}_r$ $(r=1,\ldots,R)$ nécessaires pour évaluer la loi *a priori* des signatures (2.5) sont choisis comme les projections des signatures, identifiées préalablement par l'algorithme VCA [NBD05]. Les variances $\{s_r^2\}_{r=1,\ldots,R}$ et le paramètre de forme ν sont respectivement fixés à : $s_1^2 = \ldots = s_R^2 = 100$ et $\nu = 2$.



(a) Signatures synthétiques (jeu de données $\mathcal{J}_1$)

(b) Signatures estimées

FIGURE 2.3 – Signatures synthétiques et estimées par l'algorithme uBLU (jeu de données $\mathcal{J}_1$).

2.5.1 Diagnostic de la convergence

Nous proposons d'examiner la convergence de l'algorithme uBLU sur ces données synthétiques à l'aide de la variance du bruit σ^2. En effet, ce paramètre univarié permet un contrôle plus facile et rapide, et garde bien la même signification quelle que soit la valeur de R. Nous appliquons donc la procédure de diagnostic de convergence décrite dans le paragraphe 2.4 sur ce paramètre σ^2, en simulant $C = 20$ chaînes de Markov, initialisées aléatoirement, avec $\text{N}_{\text{mc}} = 50\,000$ itérations de Monte Carlo.

Les deux potentiels d'échelle ont été calculés suivant (2.22). Pour $\mathrm{N_{bi}} = 5\ 000$ itérations de chauffe, les valeurs obtenues pour $PSRF_1$ et $PSRF_2$ sont respectivement égales à 0,93 et 1,04 pour ce jeu de données synthétiques. Ces valeurs confirment la convergence de l'échantillonneur, puisqu'elles sont proches de 1 (valeur recommandée par [CZ02]).

Les figures 2.4 permettent de visualiser l'erreur de reconstruction, calculée selon (2.23), et le nombre de signatures R estimé en fonction de l'itération ℓ pour deux chaînes MCMC d'initialisations différentes. Ces figures montrent qu'un nombre d'itérations $\mathrm{N_{mc}}$ fixé à $\mathrm{N_{mc}} = 10\ 000$ est suffisant pour s'assurer d'une bonne estimation des paramètres inconnus selon les équations (2.19) et (2.20). En effet, à partir de $\mathrm{N_{mc}} = 10\ 000$ itérations, l'erreur de reconstruction reste stable. Ceci reste valable même si le modèle change de dimensions (valeur de R) après les 10 000 premières itérations. On peut également remarquer sur cette dernière figure les mouvements de naissance et de mort acceptés par l'algorithme uBLU. Ainsi, seules $\mathrm{N_r} = \mathrm{N_{mc}} - \mathrm{N_{bi}} = 5\ 000$ itérations d'intérêt seront utilisées pour les estimations MAP des paramètres inconnus $(\widehat{R}, \widehat{\mathbf{M}}, \widehat{\mathbf{A}})$.

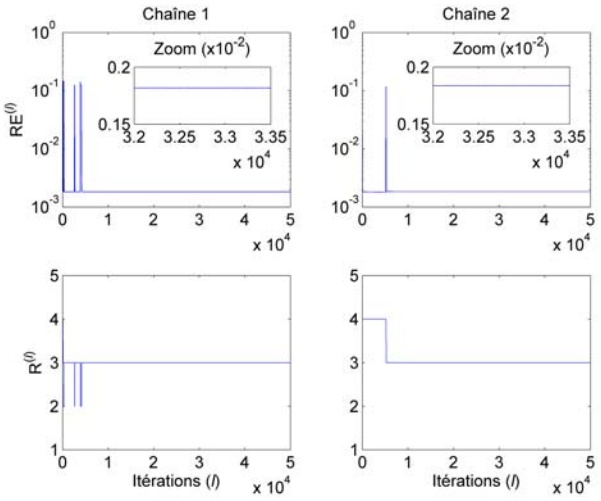


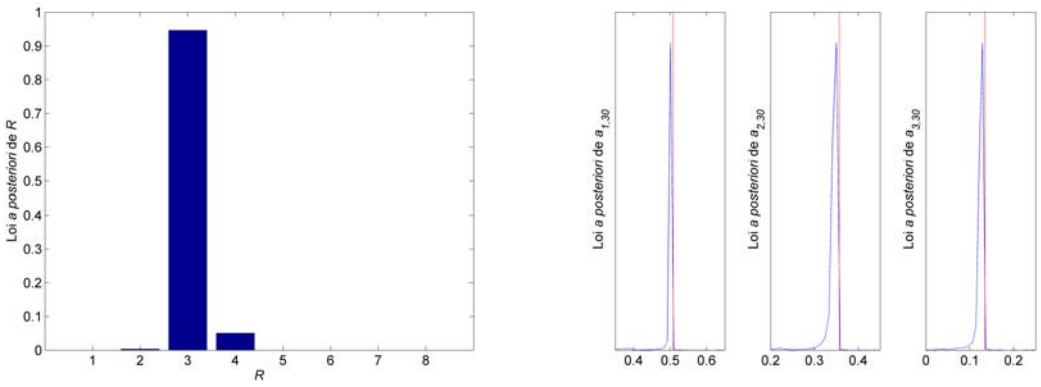
FIGURE 2.4 – Contrôle de la convergence de deux chaînes MCMC sur données synthétiques $\mathcal{J}_1$: en haut, erreur de reconstruction ($\mathrm{RE}^{(\ell)}$) calculée selon l'équation (2.23) en fonction de l'itération ℓ ; en bas, estimation du nombre de signatures $R^{(\ell)}$ à l'itération ℓ.

2.5.2 Implémentation de l'algorithme uBLU et résultats

Le paragraphe précédent a permis de déterminer des valeurs adéquates pour N_{bi} et N_{mc}. Les résultats de simulation présentés dans ce paragraphe ont donc été obtenus pour $N_{mc} = 20\ 000$ itérations de Monte Carlo, dont $N_{bi} = 5\ 000$ itérations de chauffe, en utilisant l'algorithme 2.1.

La première étape de l'analyse consiste en l'estimation de l'ordre du modèle, c'est-à-dire du nombre de signatures R présentes dans le mélange, et donc des dimensions des matrices $\mathbf{M}$ et $\mathbf{A}$. L'histogramme des échantillons $\{R^{(\ell)}\}_{\ell=N_{bi}+1,\ldots,N_{mc}}$ générés, représenté sur la figure 2.5a, est clairement en accord avec la valeur réelle $R = 3$ du nombre de signatures puisqu'il atteint son maximum pour cette valeur : l'estimation MAP de R est donc $\widehat{R} = 3$. Cet histogramme montre également que l'algorithme teste bien des espaces de dimensions différentes (correspondant à $R = 2$, $R = 3$ et $R = 4$).

La seconde étape de l'analyse consiste en l'estimation des autres paramètres inconnus du modèle ($\mathbf{M}$, $\mathbf{A}$ et σ^2) conditionnellement à $\widehat{R}$. Les signatures estimées par l'algorithme uBLU sont représentées sur la figure 2.3b. Les lois *a posteriori* des scores de l'échantillon ♯30 sont représentées sur la figure 2.5b. Ces distributions ont été obtenues par moyennage de $C = 10$ chaînes de Markov, c'est-à-dire 10 réalisations de bruit pour le même jeu de données $\mathcal{J}_1$. Les résultats obtenus concordent avec les valeurs réelles de ces coefficients représentées par la ligne rouge.



(a) Nombre de signatures R (b) Scores de l'échantillon ♯30

FIGURE 2.5 – Lois *a posteriori* du nombre de signatures R dans le mélange (a) et des scores $[a_{1,i}, a_{2,i}, a_{3,i}]^T$ conditionnellement au nombre de signatures estimé $\widehat{R} = 3$ (b).

2.5.3 Comparaisons avec d'autres algorithmes de décomposition factorielle

Afin d'étudier les performances de l'algorithme uBLU présenté au paragraphe 2.2, une comparaison avec d'autres algorithmes de décomposition factorielle est nécessaire. Nous proposons de comparer les méthodes suivantes, décrites dans le paragraphe 1.4 :

- l'analyse en composantes principales (ACP) [YR01],
- l'algorithme NMF [FYHL07],
- le modèle BFRM [CCL+08],
- l'algorithme GB-GMF [NHN+11].

Les résultats de simulations sont reportés dans le tableau 2.1 où nous avons évalué les critères suivants ($\widehat{x}$ désigne l'estimateur MAP du paramètre x, et $r = 1, \ldots, R$) :

- les erreurs quadratiques moyennes (MSEs pour *mean square errors*) des signatures $\mathbf{m}_r$:
$$\mathrm{MSE}_r^2 = \tfrac{1}{G} \left\| \widehat{\mathbf{m}}_r - \mathbf{m}_r \right\|^2,$$
- les erreurs quadratiques moyennes globales (GMSEs) des scores $\{\mathbf{a}_i\}_{i=1,\ldots,N}$:
$$\mathrm{GMSE}_r^2 = \tfrac{1}{N} \sum_{i=1}^{N} \left(\widehat{a}_{r,i} - a_{r,i} \right)^2,$$
- l'erreur de reconstruction (RE) :
$$\mathrm{RE} = \frac{1}{GN} \sum_{i=1}^{N} \left\| \mathbf{y}_i - \widehat{\mathbf{y}}_i \right\|^2, \tag{2.24}$$
- la distance angulaire (SAD pour *spectral angle distance*) entre les signatures réelles $\mathbf{m}_r$ et estimées $\widehat{\mathbf{m}}_r$:
$$\mathrm{SAD}_r = \arccos\left(\frac{\widehat{\mathbf{m}}_r^T \mathbf{m}_r}{\|\widehat{\mathbf{m}}_r\| \|\mathbf{m}_r\|} \right),$$
où $\arccos(\cdot)$ désigne la fonction cosinus inverse,
- la distance angulaire globale (GSAD) entre les échantillons observés $\mathbf{y}_i$ et leurs estimations $\widehat{\mathbf{y}}_i$:
$$\mathrm{GSAD} = \tfrac{1}{N} \sum_{i=1}^{N} \arccos\left(\frac{\widehat{\mathbf{y}}_i^T \mathbf{y}_i}{\|\widehat{\mathbf{y}}_i\| \|\mathbf{y}_i\|} \right),$$
- le temps de calcul (pour une implantation Matlab 7.8.0 (R2009a), sur un PC Intel® Core ™2 Duo cadencé à 3,0 GHz).

Remarquons que les méthodes ACP, NMF et GB-GMF ne permettent pas de faire l'estimation du nombre de facteurs R, elles ont donc été appliquées en fixant ce nombre à différentes valeurs : $R = 2$, 3 et 4. La méthode BFRM, quant à elle, a l'avantage de pouvoir être exécutée en fixant ou non

TABLE 2.1 – Résultats de simulation sur données synthétiques $\mathcal{J}_1$ et comparaison avec d'autres algorithmes.

		MSE_r^2 ($\times 10^{-2}$)			GMSE_r^2 ($\times 10^{-3}$)			SAD_r ($\times 10^{-1}$)			GSAD ($\times 10^{-2}$)	RE ($\times 10^{-2}$)	Temps de calcul (en s)
uBLU		**0,39**	0,60	0,54	**0,04**	0,04	**0,05**	**0,46**	**0,29**	0,28	3,39	**0,18**	$1{,}24.10^3$
ACP	$R=2$	N/A	6,04	0,97	N/A	2,00	0,30	N/A	3,49	1,49	20,38	9,12	0,03
	$R=3$	6,01	6,53	5,86	6,62	2,40	0,84	1,86	1,18	1,36	3,39	**0,18**	0,10
	$R=4$	6,02	6,53	8,03	23,82	11,70	6,37	1,86	1,18	1,36	**3,34**	**0,18**	0,11
NMF	$R=2$	N/A	61,12	9,78	N/A	2,00	0,28	N/A	3,50	1,50	20,38	9,12	0,71
	$R=3$	0,48	**0,45**	0,28	0,19	**0,01**	**0,05**	0,53	0,31	**0,26**	3,38	**0,18**	0,95
	$R=4$	87,78	**0,45**	**0,26**	26,56	0,23	18,04	6,14	0,31	**0,26**	3,36	**0,18**	0,96
BFRM	$R=2$	205,99	N/A	325,58	64,39	N/A	75,87	21,69	N/A	23,24	24,04	1,94	47,15
	$R=3$	212,30	681,42	137,22	76,09	142,72	76,22	10,68	15,18	5,33	24,23	1,84	53,60
	$R=4$	205,66	247,96	330,01	64,59	114,02	75,47	9,74	22,15	8,17	28,62	2,08	63,88
GB-GMF	$R=2$	267,42	N/A	67,14	226,58	N/A	41,33	12,48	N/A	27,43	37,35	9,16	$0{,}39.10^3$
	$R=3$	40,27	147,74	94,90	45,29	17,37	33,78	11,86	12,50	13,96	33,38	**0,18**	$0{,}56.10^3$
	$R=4$	195,89	101,34	68,69	57,58	3,10	27,72	8,84	26,80	27,32	29,23	**0,18**	$0{,}70.10^3$

ce nombre (mode statique à R fixé, ou, mode évolutif avec estimation de R). Lorsque nous fixons le nombre de facteurs à $R = 4$, nous ne considérons, pour le calcul des MSEs, GMSEs et SADs, que les 3 facteurs les plus proches "spectralement" des vrais facteurs, c'est-à-dire ceux permettant d'obtenir les plus petites valeurs de SADs.

Notons également à ce stade que le problème de décomposition d'une matrice $\mathbf{X}$ comme produit de deux matrices $\mathbf{M}$ et $\mathbf{A}$ (de la forme $\mathbf{X} = \mathbf{MA}$), sans ajout de contraintes, est un problème mal posé. En effet, si $\{\mathbf{M}, \mathbf{A}\}$ est un couple solutions de cette décomposition, alors $\{\mathbf{MB}, \mathbf{B}^T\mathbf{A}\}$ seront également des couples solutions, et ce quelle que soit la matrice unitaire $\mathbf{B}$. Or, l'algorithme présenté ici prend en compte explicitement des contraintes de positivité mais aussi et surtout d'additivité sur les scores (1.3). Cette contrainte d'additivité permet de gérer l'indétermination d'échelle. Ainsi il fournit une décomposition unique à une permutation de facteurs près seulement, alors que les autres méthodes nécessitent en plus une mise à l'échelle avant de calculer les différents critères proposés.

Le tableau 2.1 montre que les performances de démélange obtenues avec l'algorithme uBLU (erreur d'estimation et GSAD pour la comparaison entre les matrices $\mathbf{Y}$ observée et $\widehat{\mathbf{Y}}$ estimée) sont similaires avec celles obtenus par les méthodes NMF, PCA et GB-GMF. En revanche, si on s'intéresse à la reconstruction des facteurs (matrices $\mathbf{M}$ et $\widehat{\mathbf{M}}$ estimée) ou à celle des scores (matrices $\mathbf{A}$ et $\widehat{\mathbf{A}}$ estimée) individuellement (critères MSEs et SADs pour la comparaison des facteurs, GMSEs pour les scores), on remarque que les résultats obtenus avec la méthode uBLU sont bien meilleurs que ceux obtenus avec les algorithmes PCA et GB-GMF. Seul l'algorithme NMF a des performances similaires, mais NMF ne permet pas de retrouver le nombre R de facteurs de la décomposition. Ainsi, globalement, on montre que l'algorithme uBLU proposé pour l'analyse génétique a donc de meilleures performances que les autres algorithmes de décomposition factorielle testés dont le modèle BFRM dédié à l'analyse génétique, au prix d'un coût calculatoire plus élevé qu'avec les méthodes PCA et NMF.

2.5.4 Robustesse de l'algorithme à différents jeux de données

Le paragraphe précédent a montré les performances de l'algorithme uBLU sur un jeu de données de signatures piquées (nommé $\mathcal{J}_1$). Nous proposons dans ce paragraphe d'évaluer la robustesse de l'algorithme à d'autres jeux de données synthétiques ($\mathcal{J}_2$, $\mathcal{J}_3$ et $\mathcal{J}_4$). Ces données correspondent à

l'expression génique de $G = 512$ gènes pour les données $\mathcal{J}_3$ et $\mathcal{J}_4$, et $G = 12\,000$ gènes pour $\mathcal{J}_2$. Comme précédemment, chaque échantillon est composé de $R = 3$ signatures génétiques selon le modèle (1.1). Les signatures génétiques du deuxième jeu de données $\mathcal{J}_2$ ont été extraites à partir de données réelles, celles du troisième jeu $\mathcal{J}_3$ sont orthogonales mais pas nécessairement positives, alors que pour le dernier jeu $\mathcal{J}_4$ les signatures sont orthogonales et positives. Ces conditions sont résumées dans le tableau 2.2.

TABLE 2.2 – Erreur de reconstruction calculée pour chaque jeu de données $\mathcal{J}_1, \ldots, \mathcal{J}_4$ et comparaison avec d'autres algorithmes.

		Données $\mathcal{J}_1$	Données $\mathcal{J}_2$	Données $\mathcal{J}_3$	Données $\mathcal{J}_4$
Nature des signatures		piquées	réalistes	orthogonales	orthogonales et positives
		RE ($\times 10^{-2}$)	RE ($\times 10^{-2}$)	RE ($\times 10^{-4}$)	RE ($\times 10^{-4}$)
uBLU		**0,18**	0,64	3,11	4,49
ACP	$R = 2$	9,12	1,62	0,70	4,88
	$R = 3$	**0,18**	**0,63**	0,27	4,34
	$R = 4$	**0,18**	**0,63**	**0,16**	**4,30**
NMF	$R = 2$	9,12	1,65	0,70	4,89
	$R = 3$	**0,18**	1,55	0,29	4,36
	$R = 4$	**0,18**	0,69	**0,16**	4,33
BFRM	$R = 2$	1,94	0,65	0,47	19,34
	$R = 3$	1,84	0,75	0,49	25,00
	$R = 4$	2,08	0,86	0,41	13,48
GB-GMF	$R = 2$	9,16	5,47	2,50	8,48
	$R = 3$	**0,18**	1,62	2,44	8,48
	$R = 4$	**0,18**	1,50	2,45	8,29

Par souci de clarté, nous ne présentons dans le tableau 2.2 que les résultats obtenus pour l'erreur de reconstruction (RE calculée selon (2.24)). Les résultats obtenus pour les autres critères de comparaison (MSEs, GMSEs, SADs, GSAD et temps de calcul) sont présentés dans l'article [BDT+13]. Ces résultats montrent que le modèle uBLU proposé est assez robuste par rapport aux éventuelles propriétés que pourraient avoir les signatures, à condition de satisfaire à la positivité des signatures.

En effet, la positivité des facteurs est vérifiée dans les jeux de données $\mathcal{J}_1$, $\mathcal{J}_2$ et $\mathcal{J}_4$, mais pas pour le jeu de données $\mathcal{J}_3$. Ceci justifie que les résultats obtenus avec l'algorithme uBLU soient moins bons qu'avec les autres algorithmes pour les données $\mathcal{J}_3$ et deviennent bien meilleurs sur les données $\mathcal{J}_4$ dès que la positivité des facteurs est satisfaite.

2.6 Analyse de données génétiques réelles

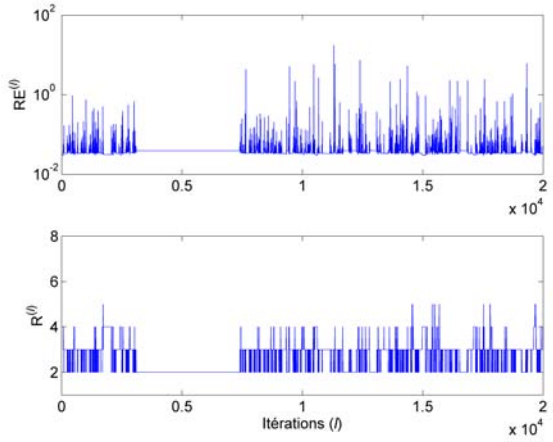
Cet algorithme a été proposé pour l'analyse génétique dans le but de regrouper les gènes pouvant être responsables d'une certaine pathologie entre eux. C'est ce que nous allons vérifier sur les trois jeux de données réelles à notre disposition : données de boissons (paragraphe 2.6.1), données de grippes H3N2 (paragraphes 2.6.2 à 2.6.4) et H1N1 (paragraphe 2.6.5).

2.6.1 Données de boissons

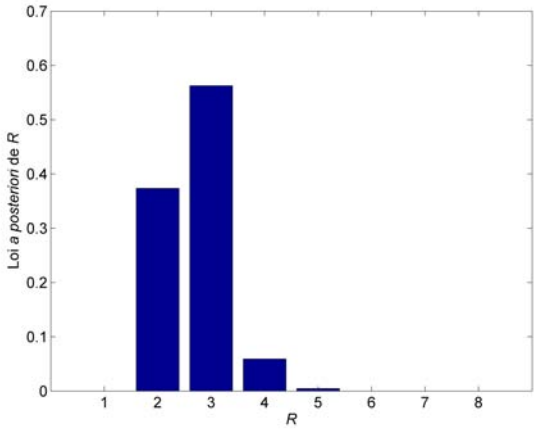
L'algorithme uBLU proposé a d'abord été évalué sur les données de boissons de Baty *et al.* [BFW+06] (brièvement décrites dans le paragraphe 1.5.1) avec $\mathrm{N_{mc}} = 20\,000$ itérations de Monte Carlo, dont $\mathrm{N_{bi}} = 5\,000$ itérations pour la période de chauffage.

Sur la figure 2.6a sont représentées les valeurs de l'erreur de reconstruction $\mathrm{RE}^{(\ell)}$, définie selon (2.23) (en haut) et le nombre de facteurs $R^{(\ell)}$ estimé (en bas) en fonction du nombre d'itérations ($\ell = 1, \ldots, \mathrm{N_{mc}}$). La figure 2.6b nous fournit l'estimateur MAP du nombre de signatures présentes dans ces données : $\widehat{R} = 3$. Ces deux figures montrent bien que l'algorithme uBLU teste des espaces de dimensions différentes ($R = 2$, 3, 4 ou 5).

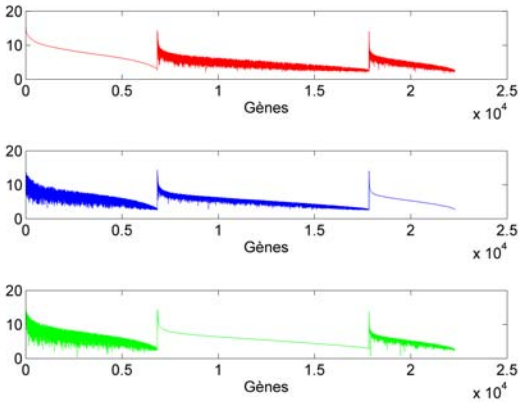
Les $\widehat{R} = 3$ signatures présentes dans ces données sont représentées sur la figure 2.6c où les indices des $G = 22\,283$ gènes ont été regroupés entre eux de manière à ce que, sur un intervalle de gènes, une seule signature soit dominante par rapport aux autres. Chaque signature est donc dominante pour un certain nombre de gènes donnés. Plus précisément, le $k^{\text{ème}}$ pic de la figure correspond au gène le plus dominant du $k^{\text{ème}}$ facteur et les gènes compris entre ce pic et le $(k+1)^{\text{ème}}$ sont aussi dominants pour ce facteur (par rapport aux autres facteurs) mais à des degrés moins importants. Par exemple, la première signature biologique, représentée en haut de la figure 2.6c, est dominante (par rapport aux autres signatures) pour les 6 836 premiers gènes.



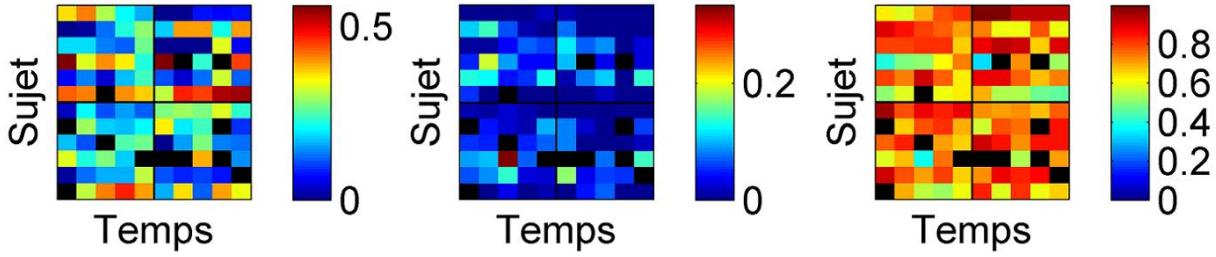
(a) Diagnostic de convergence (haut : erreur de reconstruction, bas : nombre de facteurs R estimé, en fonction du nombre d'itérations)



(b) Nombre de facteurs R estimé



(c) Facteurs estimés, rangés par dominance décroissante



(d) Scores estimés pour chacun des $\widehat{R} = 3$ facteurs

FIGURE 2.6 – Résultats de simulation de l'algorithme uBLU sur données de boissons [BFW+06].

Les scores pour chacune des signatures sont représentés sur la figure 2.6d sous forme d'images, selon l'organisation de la figure 1.5 : les lignes correspondent aux échantillons d'un même sujet au cours du temps successivement pour deux expériences (boissons ingérées) différentes. Nous pouvons alors remarquer que le facteur ♯1 (à gauche) est celui qui présente la plus grande variabilité des scores, par rapport aux deux autres facteurs qui ont des valeurs assez proches quel que soit l'échantillon. Ce facteur semble donc plus spécifique à l'expérience. Néanmoins il est assez difficile de conclure à partir de ces résultats quant à un lien entre facteur, gènes, sujet ou boissons.

2.6.2 Données H3N2

L'algorithme uBLU a été appliqué sur les données réelles H3N2 décrites dans le paragraphe 1.5.2 avec $N_{mc} = 10\,000$ itérations de Monte Carlo, dont $N_{bi} = 1\,000$ itérations pour la période de chauffe. La figure 2.7 montre l'erreur de reconstruction $RE^{(\ell)}$, calculée selon (2.23), en fonction de l'itération ℓ ($\ell = 1, \ldots$). Cette figure justifie la convergence de l'algorithme sur les données H3N2 avec les valeurs de N_{bi} et N_{mc} choisies.

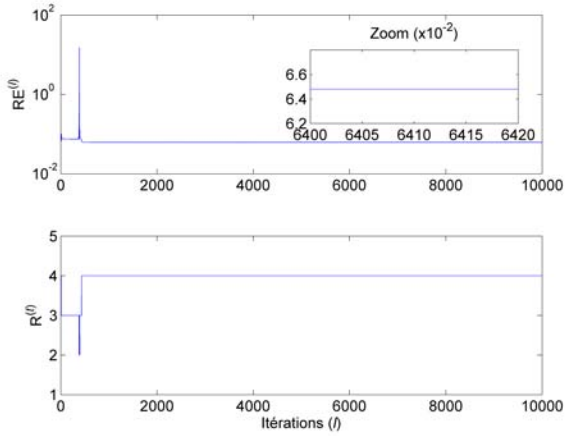
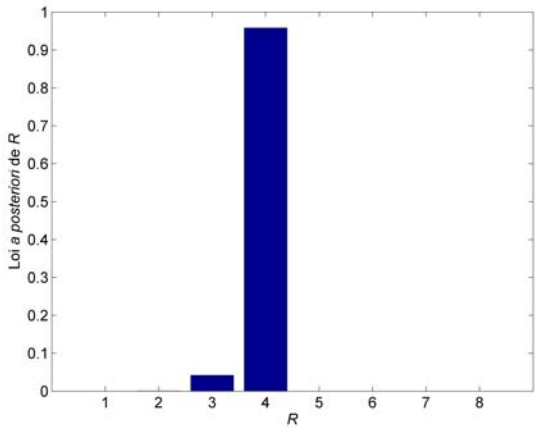
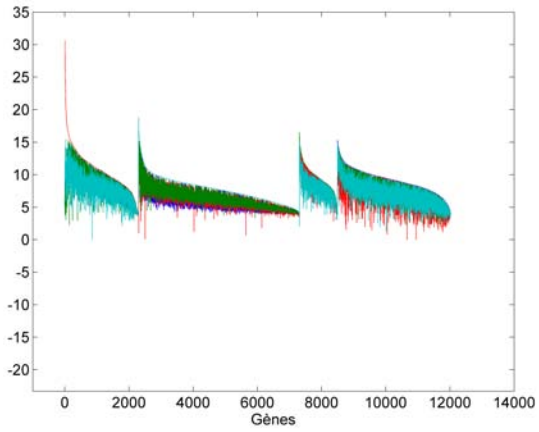


FIGURE 2.7 – Diagnostic de convergence de l'algorithme uBLU sur les données réelles H3N2 (haut : erreur de reconstruction, bas : nombre de facteurs R estimé, en fonction du nombre d'itérations).

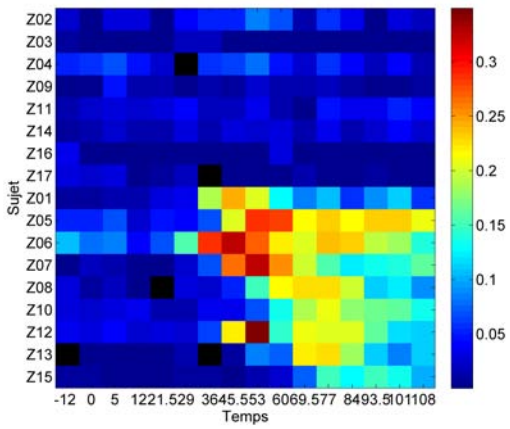
Les probabilités *a posteriori* du nombre de facteurs R sont représentées sur la figure 2.8a. On observe que plus de 90% des valeurs de R générées avec l'échantillonneur de Gibbs sont égales à 4 : l'estimation MAP du nombre de facteurs est donc $\widehat{R} = 4$.



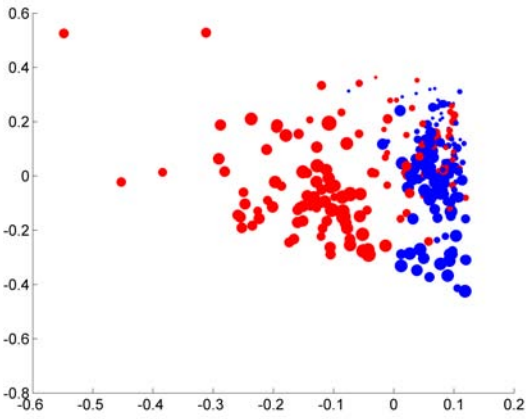
(a) Nombre de facteurs R estimé



(b) Facteurs et signatures biologiques estimés, rangés par dominance décroissante



(c) Scores du facteur inflammatoire



(d) Répartition des scores inflammatoires

FIGURE 2.8 – Résultats obtenus sur données réelles H3N2 avec l'algorithme uBLU.

Les $\widehat{R} = 4$ facteurs sont représentés sur la figure 2.8b, comme précédemment avec les données de boissons. Appliqué sur les données H3N2, le modèle uBLU identifie un facteur (le premier facteur, en rouge) particulièrement intéressant en raison de son loading maximum (égal à 30,66 et nettement supérieur aux autres) et de sa proportion de gènes dominants (2 297 gènes). De plus, il s'avère que plusieurs gènes de cette composante sont reconnaissables comme étant des gènes contrôlant les réponses immunitaire et inflammatoire à une infection virale respiratoire. Un facteur très semblable a été préalablement identifié par Zaas *et al.* [ZCV+09] et Huang *et al.* [HZR+11]. Ce dernier article [HZR+11] présente un algorithme de classification supervisée qui utilise les informations cliniques sur les symptômes pour former des clusters et établit notamment une liste de 44 gènes inflammatoires (SOM, cluster 3). Nous avons remarqué que l'ensemble des gènes identifiés par Huang *et al.* [HZR+11] comme étant inflammatoires sont retrouvés dans cette composante déterminé par l'algorithme uBLU.

Les scores correspondant à ce facteur, que nous nommons dans la suite *"facteur inflammatoire"*, sont représentés sur la figure 2.8c, où ils sont ré-organisés comme une image dont les lignes correspondent aux échantillons d'un sujet donné au cours du temps. Les cinq pixels noirs correspondent à des échantillons non-évalués. Sur cette figure, les sujets ont également été ré-organisés de manière à ce que les sujets présentant des symptômes soient associés aux 9 dernières lignes, et ceux restant asymptomatiques associés aux 8 premières lignes. Il apparaît sur cette figure qu'une segmentation en quatre états est possible : 1) état pré-inoculation, 2) état asymptomatique, 3) état pré-symptomatique, 4) état post-symptomatique. Nous pouvons notamment séparer clairement les échantillons après déclaration des symptômes (état 4) des autres échantillons. Nous reviendrons sur cette remarque dans le chapitre 4. Ces résultats sont en bonne concordance avec les symptômes cliniques recueillis au cours de l'étude (figure 1.6).

Le tableau 2.3 permet d'évaluer qualitativement les gènes présents dans cette composante inflammatoire déterminée par l'algorithme uBLU. Ce tableau liste les groupements de gènes (ou *pathways* en anglais) de la base de données PID (*pathway interaction database*[2]) [SAK+09], en les classant selon leur probabilité (p-valeur, ou *p-value* en anglais) qu'ils incluent des gènes de la composante inflammatoire de uBLU. Plus la valeur p est petite, plus les gènes de la composante inflammatoire

2. http://pid.nci.nih.gov

sont orientés vers un pathway donné. Le calcul des p-valeurs dépend notamment du nombre de gènes de la composante inflammatoire (ici, ne sont considérés que les 200 premiers gènes de cette composante inflammatoire), du nombre de gènes dans chaque pathway et du nombre total de gènes de la base de données PID. Ce tableau permet notamment de montrer que la plupart des gènes exprimés dans la composante inflammatoire sont des gènes codant des protéines ayant pour rôle de défendre l'organisme face à des agents pathogènes tels que des virus, des bactéries ou des cellules tumorales, comme les interférons (pathway IFN-gamma) ou les cytokines pro-inflammatoires (interleukines IL23, IL12, IL6).

TABLE 2.3 – Classement des pathways des gènes de la composante inflammatoire de uBLU sur les données H3N2.

Nom du pathway	Gènes	p-valeur
IFN-gamma pathway	CASP1, CEBPB, IL1B, IRF1, IRF9, PRKCD, SOCS1, STAT1, STAT3	$1{,}34.10^{-9}$
PDGFR-beta signaling pathway	DOCK4, EIF2AK2, FYN, HCK, LYN, PRKCD, SLA, SRC, STAT1, STAT3, STAT5A, STAT5B	$3{,}26.10^{-8}$
IL23-mediated signaling events	CCL2, CXCL1, CXCL9, IL1B, STAT1, STAT3, STAT5A	$2{,}18.10^{-7}$
Signaling events mediated by TCPTP	EIF2AK2, SRC, STAT1, STAT3, STAT5A, STAT5B, STAT6	$6{,}38.10^{-7}$
Signaling events mediated by PTP1B	FYN, HCK, LYN, SRC, STAT3, STAT5A, STAT5B	$2{,}40.10^{-6}$
GMCSF-mediated signaling events	CCL2, LYN, STAT1, STAT3, STAT5A, STAT5B	$3{,}70.10^{-6}$
IL12-mediated signaling events	HLA-A, IL1B, SOCS1, STAT1, STAT3, STAT5A, STAT6	$1{,}32.10^{-5}$
IL6-mediated signaling events	CEBPB, HCK, IRF1, PRKCD, STAT1, STAT3	$1{,}80.10^{-5}$

2.6.3 Apport des contraintes : résultats sur les données H3N2

Pour visualiser l'apport des contraintes imposées sur les facteurs (projetés) et les scores (1.3), l'algorithme uBLU a été modifié et appliqué sur le jeu de données H3N2 dans les cas suivants :

(a) sans aucune contrainte, ni sur les facteurs, ni sur les scores,

(b) avec seulement les contraintes de non-négativité sur les facteurs et les scores,

(c) avec la seule contrainte de somme-à-un imposée sur les scores,

(d) avec toutes les contraintes de non-négativité et de somme-à-un sur les facteurs et les scores, telles que proposées dans (1.3).

Les lois *a priori* et conditionnelles des facteurs projetés $\mathbf{t}_r$ et des scores $\mathbf{a}_i$ sont résumées dans le tableau 2.4.

TABLE 2.4 – Apport des contraintes : sous-espaces, lois *a priori* et loi conditionnelles des facteurs projetés et des scores.

(a) Aucune contrainte

	Facteurs projetés	Scores
Sous-espace		$\mathbb{R}^R$
Loi *a priori*	$\mathcal{N}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right)$	$\mathcal{U}_{\mathbb{R}^R}\left(\mathbf{a}_i\right)$
Loi conditionnelle	$\mathcal{N}(\boldsymbol{\tau}_r, \boldsymbol{\Gamma}_r)$	$\mathcal{N}_{\mathbb{R}^R}\left(\mu_i, \Sigma\right)$

(b) Seulement la positivité

	Facteurs projetés	Scores
Sous-espace	$\mathcal{T}_r$ (2.6)	$\mathcal{S}_+ = \{\mathbf{a}_i \vert \mathbf{a}_i \succeq 0\}$
Loi *a priori*	$\mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right)$ (2.5)	$\mathcal{U}_{\mathcal{S}_+}\left(\mathbf{a}_i\right)$
Loi conditionnelle	$\mathcal{N}_{\mathcal{T}_r}\left(\boldsymbol{\tau}_r, \boldsymbol{\Gamma}_r\right)$ (2.16)	$\mathcal{N}_{\mathcal{S}_+}\left(\mu_i, \Sigma\right)$

(c) Seulement la somme-à-un des scores

	Facteurs projetés	Scores
Sous-espace		$\mathcal{S}_1 = \{\mathbf{a}_i \vert \, \Vert\mathbf{a}_i\Vert_1 = 1\}$
Loi *a priori*	$\mathcal{N}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right)$	$\mathcal{U}_{\mathcal{S}_1}\left(\mathbf{a}_{1:R-1,i}\right)$
Loi conditionnelle	$\mathcal{N}(\boldsymbol{\tau}_r, \boldsymbol{\Gamma}_r)$	$\mathcal{N}_{\mathcal{S}_1}\left(\bar{\mathbf{a}}_{1:R-1,i}, \Sigma_{1:R-1,i}\right)$

(d) Toutes les contraintes (1.3)

	Facteurs projetés	Scores
Sous-espace	$\mathcal{T}_r$ (2.6)	$\mathcal{S}$ (2.10)
Loi *a priori*	$\mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right)$ (2.5)	$\mathcal{U}_{\mathcal{S}}\left(\mathbf{a}_{1:R-1,i}\right)$ (2.11)
Loi conditionnelle	$\mathcal{N}_{\mathcal{T}_r}\left(\boldsymbol{\tau}_r, \boldsymbol{\Gamma}_r\right)$ (2.16)	$\mathcal{N}_{\mathcal{S}}\left(\bar{\mathbf{a}}_{1:R-1,i}, \Sigma_{1:R-1,i}\right)$ (2.17)

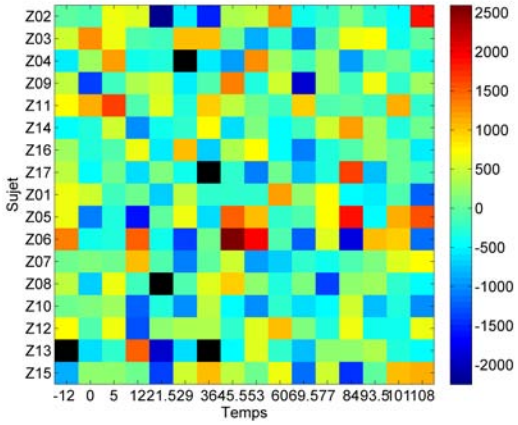
Les figures 2.9 représentent les cartes des scores du facteur inflammatoire. Précédemment nous avons vu que l'algorithme uBLU appliqué sur données H3N2 permettait de séparer clairement les échantillons en deux régions principales : les échantillons post-symptomatiques et les autres (figure 2.8c). En fait, cette segmentation ne devient apparente que lorsque la contrainte de somme-à-un est appliquée sur les scores. Notons que lorsque seule la contrainte de somme-à-un est imposée, la positivité n'est pas toujours garantie. Dans le cas présent, tous les scores du facteur inflammatoire sont positifs, mais ce n'est pas le cas pour les scores de certains autres facteurs, non représentés dans ce manuscrit.

Afin de quantifier ce résultat, nous proposons de calculer le critère de Fisher [DHS00, p. 119], en tant que mesure de contraste entre les échantillons post-symptomatiques et les autres échantillons :

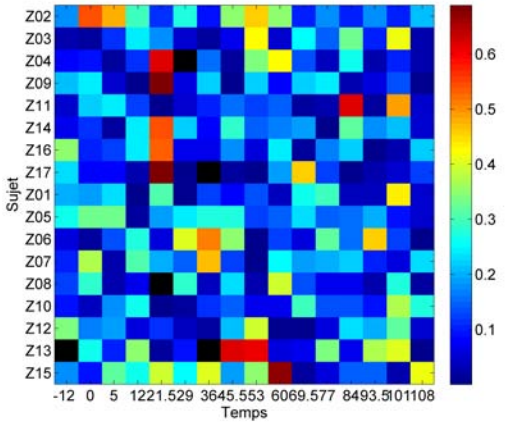
$$F = \frac{(\mu_{\text{pos}} - \mu_{\overline{\text{pos}}})^2}{N_{\text{pos}}\sigma^2{}_{\text{pos}} + (N - N_{\text{pos}})\sigma^2{}_{\overline{\text{pos}}}}, \tag{2.25}$$

où $\left(\mu_{\text{pos}}, \sigma^2{}_{\text{pos}}\right)$ sont les moyenne et variance empiriques des N_{pos} échantillons post-symptomatiques (état 4), et $\left(\mu_{\overline{\text{pos}}}, \sigma^2{}_{\overline{\text{pos}}}\right)$ les mêmes paramètres pour les autres échantillons. Le tableau 2.5 répertorie ces mesures pour chacun des quatre cas de contraintes étudiés. Ce tableau renseigne également le pourcentage des gènes inflammatoires de Huang *et al.* [HZR+11] retrouvés dans le facteur inflammatoire des quatre cas considérés, ainsi que la p-valeur des pathways suivants, indicateurs d'une infection virale : le pathway "IFN-gamma" (pathway comportant des gènes codant les interférons) et "IL23-mediated signaling events" (pathway de gènes codant l'interleukine IL-23).

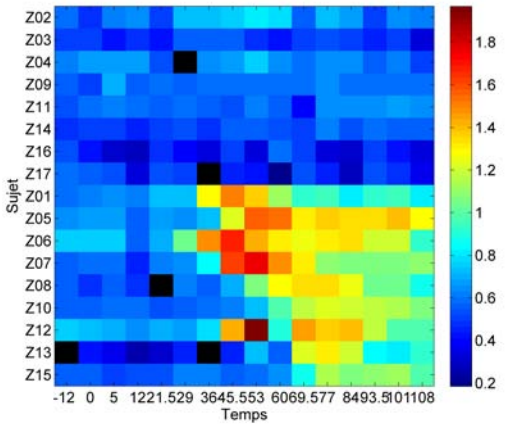
Les résultats obtenus montrent donc que les deux contraintes de non-négativité et de somme-à-un sont nécessaires pour : 1) discriminer les échantillons post-symptomatiques des autres échantillons (selon le critère de Fisher), et donc discriminer les individus malades des individus sains, 2) retrouver tous les gènes inflammatoires de Huang *et al.* [HZR+11] dans un seul facteur, le facteur inflammatoire.



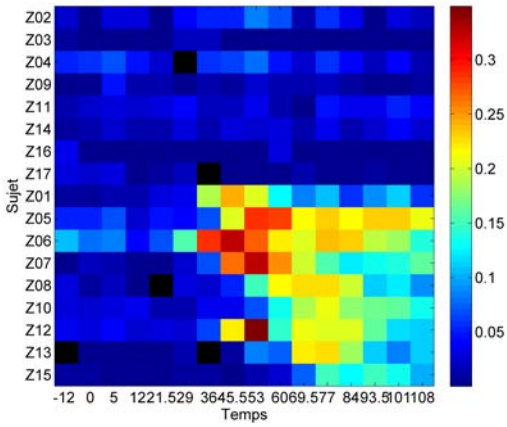
(a) Sans aucune contrainte imposée



(b) Avec seulement la contrainte de positivité sur les scores et les facteurs



(c) Avec seulement la contrainte de somme-à-un sur les scores



(d) Avec toutes les contraintes (1.3) imposées

FIGURE 2.9 – Contribution de chaque contrainte sur les scores du facteur inflammatoire.

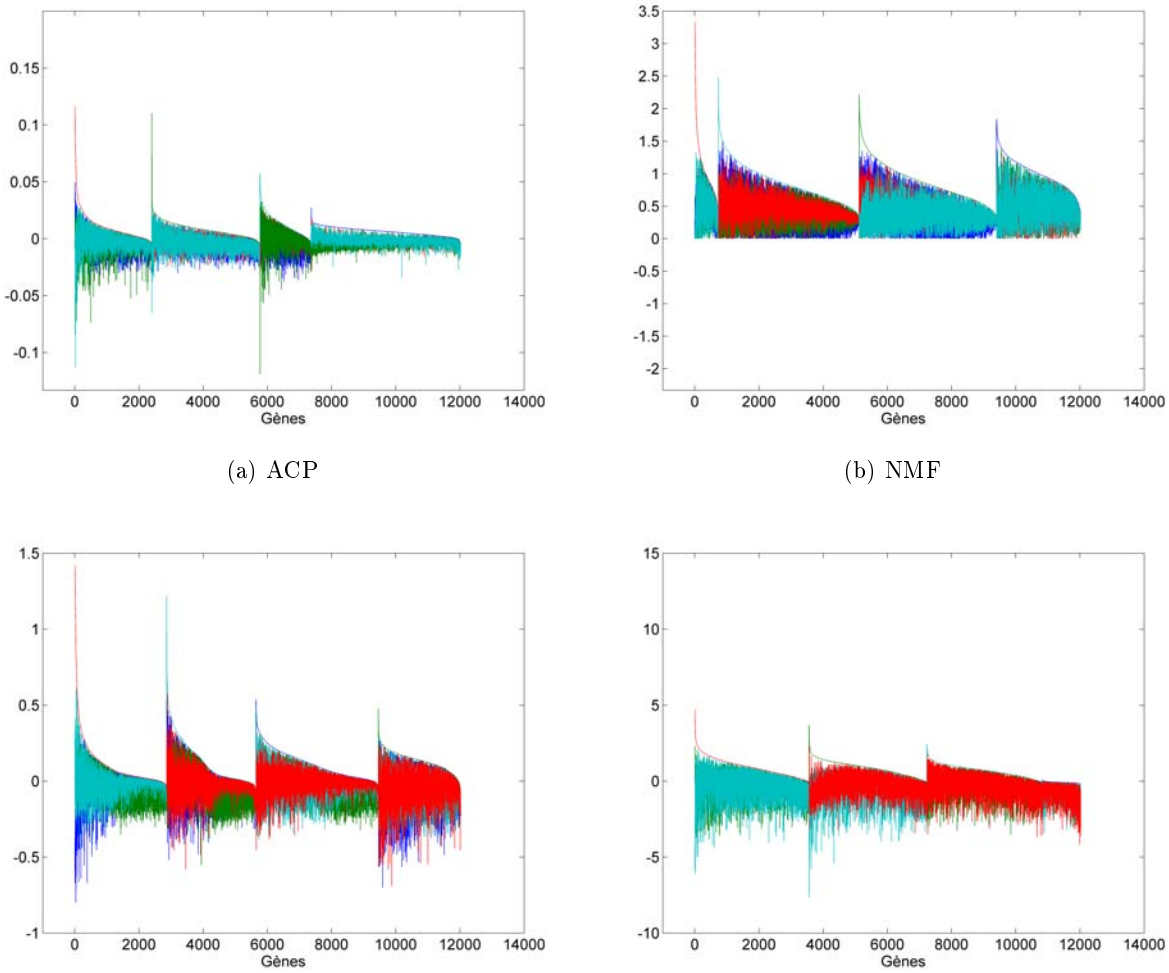
TABLE 2.5 – Contributions des contraintes.

	sans contraintes	positivité	somme-à-un	positivité et somme-à-un
Critère de Fisher (2.25)	$1{,}15.10^{-5}$	$3{,}31.10^{-8}$	$\mathbf{6{,}27.10^{-2}}$	$6{,}20.10^{-2}$
Gènes inflammatoires de [HZR+11]	88,64%	70,45%	61,36%	**100%**
p-valeur du pathway IFN-gamma	$6{,}00.10^{-2}$	$2{,}05.10^{-2}$	$2{,}17.10^{-1}$	$\mathbf{1{,}34.10^{-9}}$
p-valeur du pathway IL23	$2{,}60.10^{-1}$	$8{,}37.10^{-2}$	$2{,}28.10^{-2}$	$\mathbf{2{,}18.10^{-7}}$

2.6.4 Comparaison avec d'autres algorithmes

L'algorithme uBLU proposé a d'abord été comparé à sa version supervisée, c'est-à-dire en fixant le nombre de facteurs à $R = 4$. Les résultats obtenus, disponibles dans [HZR+11] et dans le chapitre 4, sont similaires à ceux obtenus de manière totalement non-supervisée, en ce qui concerne la découverte d'un facteur inflammatoire et la séparation des individus malades et sains. Puis nous l'avons comparé avec les quatre algorithmes de décomposition matricielle suivants : les méthodes ACP, NMF, BFRM et GB-GMF.

Les figures 2.10 représentent, pour chaque algorithme testé, les différentes signatures biologiques, organisées de façon à ce que le facteur inflammatoire soit celui le plus à gauche, en rouge. Rappelons que, sur ces figures, les gènes sont regroupés par facteur de manière à ce que pour chaque facteur une seule signature biologique soit dominante ; l'axe des abscisses (ordre des gènes) n'est donc pas le même pour toutes les figures. Nous pouvons alors remarquer que le facteur inflammatoire déterminé par chacun des algorithmes testés ne contient pas nécessairement le même nombre de gènes (voir tableau 2.6). Le diagramme de Venn, représenté dans la figure 2.11, permet de visualiser quels sont les algorithmes qui retrouvent le mieux les gènes inflammatoires de Huang *et al.* [HZR+11] (méthode supervisée SOM, cluster 3) dans leur facteur inflammatoire. On remarque sur cette figure que seul l'algorithme uBLU permet de retrouver tous les gènes de Huang *et al.* dans son facteur inflammatoire, ce qui se confirme par le pourcentage de gènes inflammatoires du tableau 2.6.



(a) ACP

(b) NMF

(c) BFRM

(d) GB-GMF

FIGURE 2.10 – Facteurs estimés rangés par dominance décroissante, données H3N2.

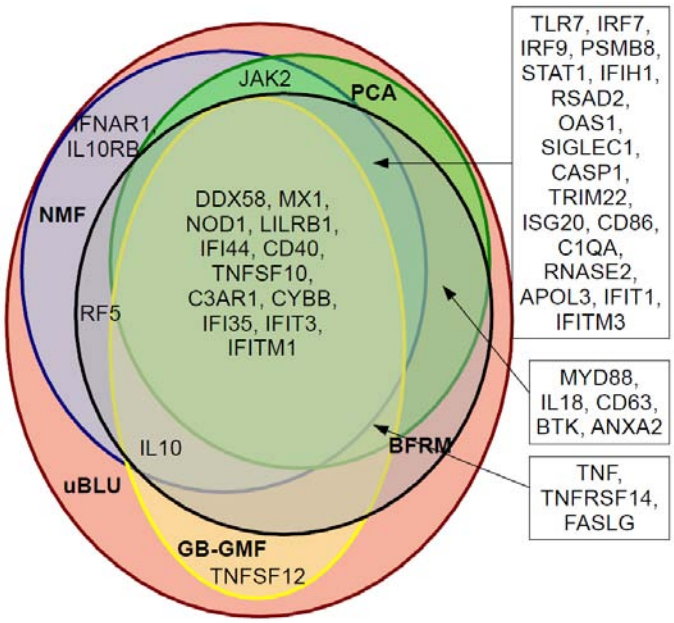
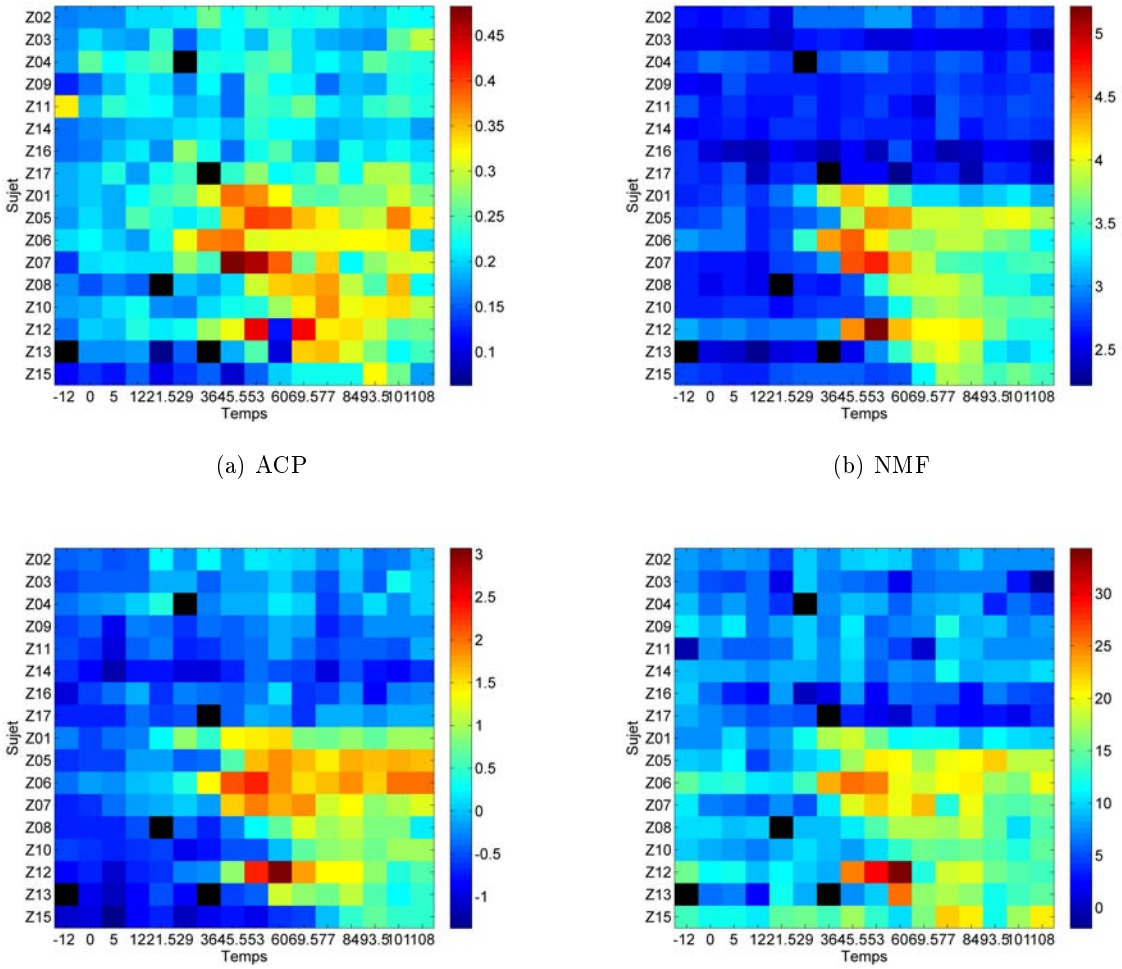


FIGURE 2.11 – Diagramme de Venn sur les gènes déterminés par Huang *et al.* comme étant les plus inflammatoires [HZR+11] (méthode supervisée SOM, cluster 3).

TABLE 2.6 – Résultats de comparaison entre uBLU et quatre autres algorithmes sur données H3N2.

	uBLU	ACP	NMF	BFRM	GB-GMF
Nombre de gènes inflammatoires	2 297	2 398	729	2 860	3 560
Loadings maximum	30,66 (17,78)	3,34 (2,48)	0,12 (0,11)	1,39 (1,22)	4,71 (3,69)
Erreur de reconstruction (2.24)	$\mathbf{6{,}48.10^{-2}}$	4,89	$7{,}31.10^{-2}$	4,82	$9{,}51.10^{-2}$
Critère de Fisher ($\times 10^{-2}$) (2.25)	**6,20**	2,03	6,17	4,68	2,30
Gènes inflammatoires de [HZR+11]	**100%**	88,64%	79,55%	90,91%	38,64%
p-valeur du pathway IFN-gamma	$\mathbf{1{,}34.10^{-9}}$	$2{,}77.10^{-3}$	$1{,}07.10^{-5}$	$6{,}59.10^{-2}$	N/A
p-valeur du pathway IL23	$\mathbf{2{,}18.10^{-7}}$	$1{,}06.10^{-5}$	$\mathbf{2{,}18.10^{-7}}$	$3{,}62.10^{-5}$	$1{,}31.10^{-2}$
Temps de calcul	$\approx 12\ h$	$\mathbf{1{,}5\ s}$	$116\ s$	$\approx 47\ min$	$\approx 10\ h$
Nombre d'itérations	10 000	N/A	5 000	10 000	500



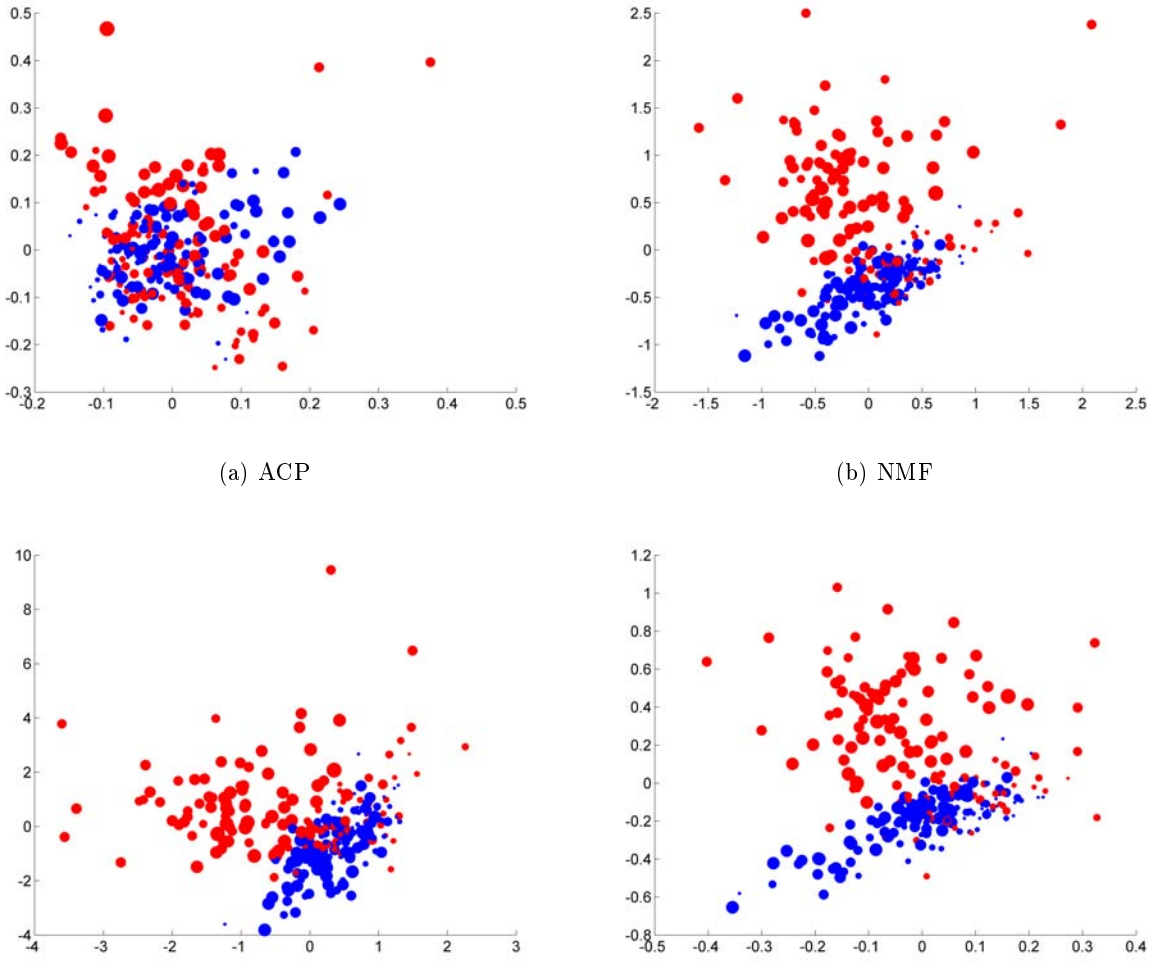
(a) ACP

(b) NMF

(c) BFRM

(d) GB-GMF

FIGURE 2.12 – Scores du facteur inflammatoire, données H3N2.



(a) ACP

(b) NMF

(c) BFRM

(d) GB-GMF

FIGURE 2.13 – Répartition des scores inflammatoires.

Les scores des facteurs inflammatoires sont représentés sur la figure 2.12. Parmi les cinq algorithmes testés, l'algorithme uBLU proposé (figure 2.8c) est celui qui permet de séparer, avec le plus grand contraste, les échantillons post-symptomatiques des autres échantillons. Ceci est confirmé par la mesure du critère de Fisher (2.25) entre ces deux régions (table 2.6).

Le tableau 2.6 présente également les valeurs maximales des loadings des groupements de gènes dominants (la valeur de gauche correspond au loading maximal, celle entre parenthèses au loading maximal du deuxième groupement de gènes), l'erreur de reconstruction calculée selon (2.24), les p-valeurs des pathways "IFN-gamma" et "IL23" en considérant les 200 premiers gènes des facteurs inflammatoires déterminés par chaque algorithme[3], le temps de calcul et le nombre d'itérations utilisées pour les résultats présentés.

Les figures 2.8d et 2.13 représentent les projections spatiales des vecteurs des scores de chaque échantillon, selon la méthode de positionnement multidimensionnel euclidien (MDS pour *"euclidian multidimensional scaling"*) [CC94]. Ces figures permettent donc de visualiser spatialement la séparation entre sujets malades (points rouges) et sujets sains (points bleus). Les résultats obtenus avec les algorithmes uBLU, NMF et BFRM montrent une nette séparation des sujets malades et sains, alors que la séparation est bien moins évidente avec l'ACP. Dans ces figures, remarquons que la taille de chaque point est proportionnelle à la durée depuis l'inoculation.

L'ensemble des résultats obtenus sur les données réelles H3N2 montrent que l'algorithme proposé uBLU donne de meilleures performances que les méthodes ACP, NMF, BFRM et GB-GMF concernant l'identification d'un facteur inflammatoire et la séparation entre individus sains et malades. L'algorithme GB-GMF est légèrement plus rapide que uBLU mais il ne permet pas de déterminer correctement le facteur inflammatoire de Zaas *et al.* [ZCV+09] et Huang *et al.* [HZR+11].

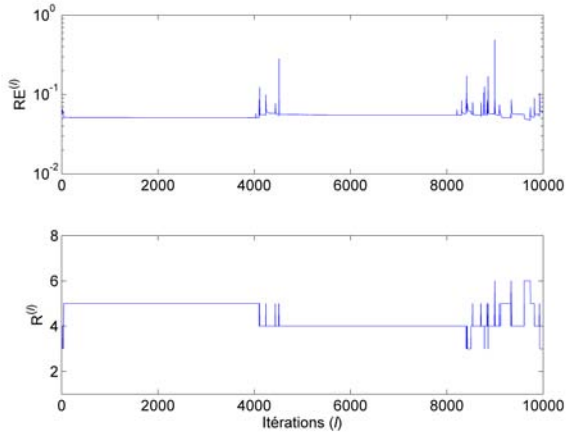
3. N/A signifie que les gènes du pathway ne sont pas représentés dans les 200 premiers gènes inflammatoires déterminés par l'algorithme étudié.

2.6.5 Résultats sur données H1N1

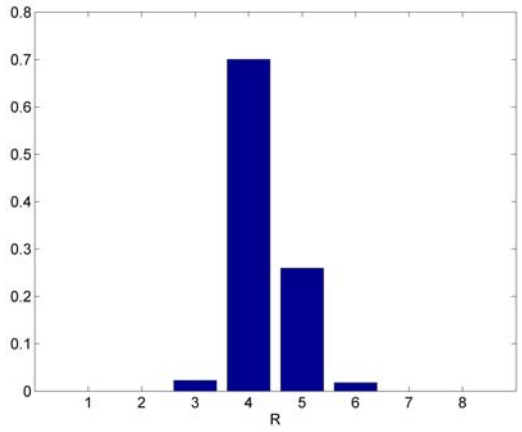
Enfin, l'algorithme uBLU a été appliqué sur les données H1N1 décrites dans le paragraphe 1.5.3, avec $\mathrm{N_{mc}} = 10\ 000$ itérations de Monte Carlo. La figure 2.14b montre que le nombre de facteurs estimés est $\widehat{R} = 4$ (environ 70% des échantillons générés $\{R^{(\ell)}\}_{\ell=\mathrm{N_{bi}}+1}^{\mathrm{N_{mc}}}$ valent 4). La figure 2.14a montre l'erreur de reconstruction (RE), calculée selon (2.24), en fonction du nombre d'itérations et confirme la bonne convergence de l'échantillonneur.

Comme pour les données H3N2, les figures 2.14c et 2.14d représentent respectivement les $\widehat{R} = 4$ signatures génétiques (avec la composante inflammatoire la plus à gauche), et les scores du facteur inflammatoire. Les résultats obtenus montrent que, sur les données H1N1, les scores du facteur inflammatoire peuvent être segmentés en plusieurs états : asymptomatique (échantillons des 12 premières lignes de la figure 2.14d, labélisés W01, W03, W04, W10, W11, W14, W16, W18, W19, W22, W23 et W24), pré-symptomatique (échantillons des 12 dernières lignes jusqu'à $t \approx 60$), post-symptomatique (autres échantillons). L'analyse NCI des pathways des gènes de la composante inflammatoire est disponible dans la table 2.7. On retrouve les mêmes groupements de gènes dominants que ceux obtenus dans le cadre de l'analyse des données de grippe H3N2, notamment les pathways IFN-γ et IL23 liés à une réaction du système immunitaire face à une infection virale par exemple.

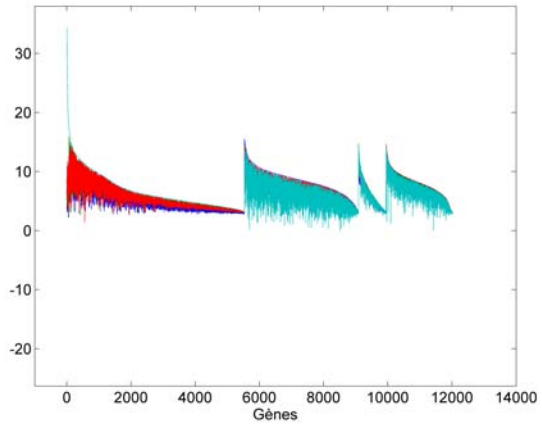
Enfin, dans le tableau 2.8, les résultats obtenus sur les données H1N1 sont comparés avec ceux obtenus avec les autres algorithmes de décomposition matricielle déjà étudiés : ACP, NMF, BFRM et GB-GMF. Sur ce jeu de données, l'algorithme uBLU reste également le plus performant : il offre une meilleure reconstruction des données, une meilleure séparabilité des sujets symptomatiques et asymptomatiques, et retrouve un grand nombre de gènes inflammatoires dans sa composante inflammatoire.



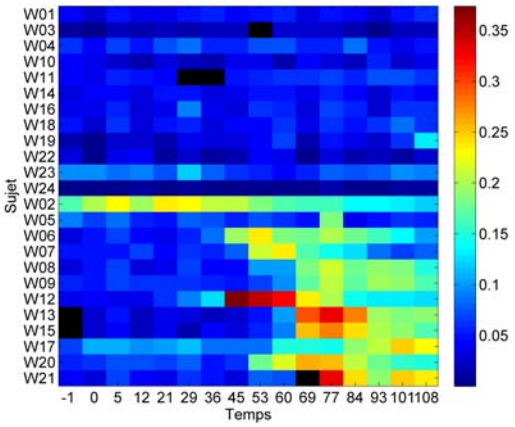
(a) Diagnostic de convergence (haut : erreur de reconstruction, bas : nombre de facteurs R estimé, en fonction du nombre d'itérations)



(b) Nombre de facteurs R estimés

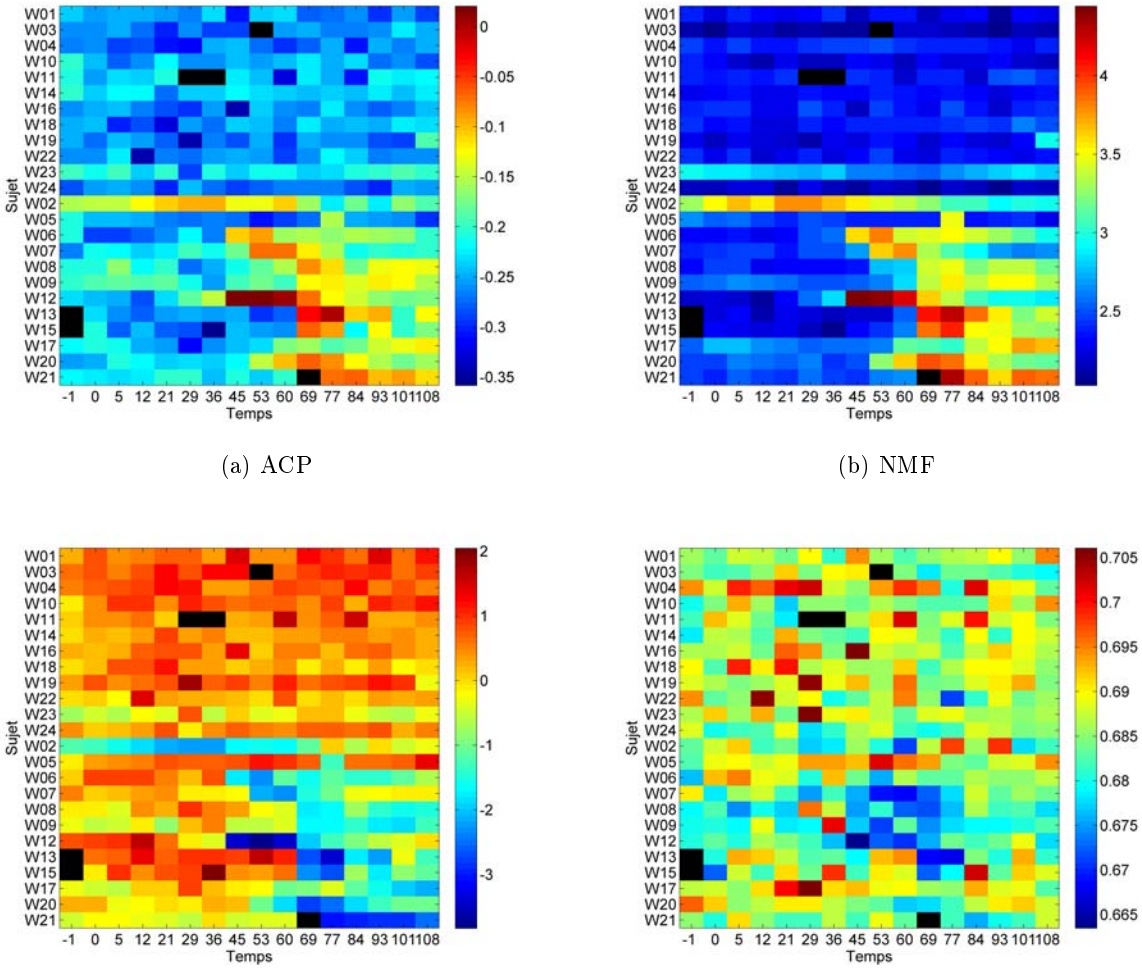


(c) Facteurs et signatures biologiques estimés, rangés par dominance décroissante



(d) Scores du facteur inflammatoire

FIGURE 2.14 – Résultats obtenus sur données réelles H1N1 avec l'algorithme uBLU.



(a) ACP

(b) NMF

(c) BFRM

(d) GB-GMF

FIGURE 2.15 – Scores du facteur inflammatoire, données H1N1.

TABLE 2.7 – Classement des pathways des gènes de la composante inflammatoire de uBLU sur les données H1N1.

Nom du pathway	Gènes	p-valeur
IFN-gamma pathway	CASP1, CEBPB, IL1B, IRF1, IRF9, SOCS1, STAT1	$6{,}38.10^{-7}$
Signaling events mediated by TCPTP	EIF2AK2, KPNB1, SRC, STAT1, STAT5A, STAT5B, STAT6	$6{,}38.10^{-7}$
IL12-mediated signaling events	GADD45B, HLA-A, IL1B, MAP2K6, SOCS1, STAT1, STAT5A, STAT6	$1{,}10.10^{-6}$
IL23-mediated signaling events	CCL2, CXCL1, CXCL9, IL1B, STAT1, STAT5A	$4{,}36.10^{-6}$
CXCR3-mediated signaling events	CXCL10, CXCL11, CXCL13, CXCL9, MAP2K6, SRC	$1{,}23.10^{-5}$
GMCSF-mediated signaling events	CCL2, LYN, STAT1, STAT5A, STAT5B	$6{,}24.10^{-5}$
IL1-mediated signaling events	CASP1, IL1B, IL1RN, MAP2K6, MYD88	$6{,}24.10^{-5}$
PDGFR-beta signaling pathway	DOCK4, EIF2AK2, FYN, LYN, SRC, STAT1, STAT5A, STAT5B	$1{,}38.10^{-4}$

TABLE 2.8 – Résultats de comparaison entre uBLU et quatre autres algorithmes sur données H1N1.

	uBLU	ACP	NMF	BFRM	GB-GMF
Nombre de gènes inflammatoires	5 538	3 038	513	4 724	1 528
Loadings maximum	34,47 (15,49)	0,24 (0,11)	3,32 (3,06)	0,715 (0,706)	8,93 (6,28)
Erreur de reconstruction (2.24)	$\mathbf{5{,}48.10^{-2}}$	5,10	$5{,}53.10^{-2}$	5,08	$9{,}72.10^{-2}$
Critère de Fisher ($\times 10^{-2}$) (2.25)	3,55	1,95	**3,74**	2,09	0,63
Gènes inflammatoires de [HZR+11]	**90,91%**	61,36%	68,18%	86,36%	29,55%
p-valeur du pathway IFN-gamma	$\mathbf{6{,}38.10^{-7}}$	N/A	$1{,}29.10^{-5}$	N/A	N/A
p-valeur du pathway IL23	$\mathbf{4{,}36.10^{-6}}$	$1{,}10.10^{-3}$	$5{,}27.10^{-6}$	N/A	$2{,}12.10^{-1}$
Temps de calcul	$\approx 18\ h$	**2,0 *s***	174 *s*	$\approx 1\ h$	$\approx 14\ h$
Nombre d'itérations	10 000	N/A	5 000	10 000	500

2.7 Conclusion

Ce chapitre a présenté une nouvelle approche bayésienne hiérarchique entièrement non-supervisée pour le démélange de données d'expression de gènes de grande dimension. Une propriété intéressante de l'algorithme proposé est qu'il impose des contraintes sur les paramètres à estimer : non-négativité des signatures génétiques et des scores, mais aussi somme-à-un des scores. Cela permet notamment de faire face au problème inhérent d'ambiguïté d'échelle existant lors de toute décomposition matricielle. Sur données réelles, ces contraintes conduisent à une meilleure discrimination entre les individus sains et les individus malades et permettent de regrouper les gènes inflammatoires dans une seule et unique composante, la composante inflammatoire.

Une procédure de réduction de dimensionalité est utilisée afin de réduire considérablement les dimensions de l'espace de travail, permettant ainsi de combiner un échantillonneur de Gibbs avec un processus de naissance et de mort, sans surcoût calculatoire prohibitif. L'algorithme ainsi proposé est donc totalement non-supervisé : il estime le nombre de signatures génétiques présentes dans le mélange directement à partir des échantillons, et ce, sans connaissance *a priori* sur la nature symptomatique ou non des échantillons.

Les résultats obtenus sur différents jeux de données synthétiques et réelles ont démontré les bonnes performances de l'algorithme uBLU. En effet, l'algorithme présenté permet notamment d'extraire, sur données réelles grippales, une composante inflammatoire avec un meilleur contraste entre les sujets symptomatiques et asymptomatiques, que celle obtenue avec d'autres algorithmes de décomposition factorielle.

Dans ce chapitre, les échantillons temporels d'un même sujet ont été considérés comme indépendants entre eux. Le chapitre 4 étendra le modèle présenté pour prendre en compte cette dépendance temporelle entre échantillons.

Contributions du chapitre

La première contribution de cette thèse est la considération de contraintes physiques liées aux données génétiques étudiées (non-négativité des facteurs et des scores, somme-à-un des scores). Cette

contribution est majeure et apparaît comme un leitmotiv dans les algorithmes proposés tout au long de cette thèse. Dans la littérature, certains algorithmes de décomposition factorielle appliqués à l'analyse génétique permettent la prise en compte de la non-négativité des facteurs et/ou des scores, mais aucun n'impose la somme-à-un des scores, qui permet pourtant une meilleure interprétation des paramètres et des résultats obtenus sur données réelles.

La deuxième contribution porte sur l'algorithme entièrement non-supervisé, associé au modèle bayésien introduit pour les mélanges linéaires de signatures génétiques. En effet, nous proposons d'estimer le nombre de facteurs dans le mélange grâce à l'utilisation d'un processus de naissance et de mort. Ce type de processus n'a guère été employé sur données génétiques du fait de la complexité calculatoire qu'il introduisait (recherche de solutions dans des espaces de dimensions différentes). Ceci a été ici contourné par une procédure de réduction de dimensionalité, permettant de travailler dans l'espace projeté des facteurs, espace de dimension réduite.

Chapitre 3

Prise en compte de la parcimonie : modèle Bernoulli-gaussien

Sommaire

3.1 Introduction

Dans le chapitre précédent, nous avons développé un algorithme bayésien de démélange permettant d'estimer conjointement les facteurs et les scores sous des contraintes de positivité et de somme-à-un. Le modèle précédent avait l'avantage d'être entièrement non-supervisé et permettait ainsi d'estimer le nombre de facteurs présents dans le mélange en utilisant un processus de naissance et de mort sur ce nombre de facteurs.

Ce chapitre présente une autre manière d'estimer ce nombre de facteurs en considérant que nous avons à notre disposition une plus grande bibliothèque de facteurs et que nous recherchons les facteurs qui sont réellement dans le mélange parmi tous les facteurs disponibles dans la bibliothèque. Ceci est donc une approche parcimonieuse pour l'estimation du nombre de facteurs. Cette notion de parcimonie sur le vecteur des scores vient ici comme une contrainte supplémentaire à satisfaire, en plus des contraintes physiques précédentes de positivité et de somme-à-un (1.3). Cela signifie également que seulement certains éléments du vecteur des scores seront non-nuls.

La stratégie développée dans ce chapitre repose sur un modèle bayésien hiérarchique basé sur le choix d'une loi *a priori* adéquate pour les vecteurs des scores : une loi tronquée Bernoulli-gaussienne. Cette loi est définie à l'aide d'un mélange d'une loi normale tronquée (afin de s'assurer de la positivité et de la somme-à-un des scores pour chaque échantillon) et d'une masse en zéro (pour forcer certains éléments à être nuls). Cette stratégie de mélange d'une masse en zéro et d'une loi exponentielle ou gaussienne a déjà été utilisée dans la littérature dans le but d'accentuer la parcimonie de la loi *a priori*. De telles lois ont été employées pour résoudre des problèmes de débruitage [JS04], pour la reconstruction d'images bruitées [TRH09] ou d'images issues d'un microscope à résonance magnétique [DHT09], et plus récemment pour l'acquisition comprimée (en anglais "*compressed sensing*") en imagerie ultrasonore [DBTK12].

En analyse génétique, les approches parcimonieuses développées sont le plus souvent des approches non-paramétriques basées sur les processus Beta [PC09] ou les processus du buffet indien [CCP+10]. En revanche, aucune de ces méthodes bayésiennes non-paramétriques ne permet de satisfaire aux contraintes de positivité et d'additivité liées à la physique du modèle (1.3).

Comme dans le chapitre précédent, chaque échantillon observé $\mathbf{y}_i$ $(i = 1, \ldots, N)$ se décompose suivant le modèle de mélange linéaire (1.1) en R facteurs. Cependant, dans ce chapitre, les R facteurs $\{\mathbf{m}_r\}_{r=1,\ldots,R}$ sont supposés appartenir à une bibliothèque $\mathbf{M} \in \mathbb{R}^{G \times R_{\max}}$ de $R_{\max}$ signatures possibles, avec $R_{\max} > R$. Les vecteurs des scores, de taille $R_{\max}$, comporteront donc R valeurs non-nulles sur $R_{\max}$. Le modèle de mélange linéaire parcimonieux s'écrit donc :

$$\mathbf{y}_i = \sum_{r=1}^{R_{\max}} \mathbf{m}_r a_{r,i} + \mathbf{n}_i, \quad \text{pour} \quad i = 1, \ldots, N, \tag{3.1}$$

ou matriciellement :

$$\mathbf{Y} = \mathbf{M}\mathbf{A} + \mathbf{N} \quad \text{avec} \quad \begin{cases} \mathbf{M} & \in & \mathbb{R}^{G \times R_{\max}}, \\ \mathbf{A} & \in & \mathbb{R}^{R_{\max} \times N}. \end{cases} \tag{3.2}$$

Dans un premier temps la bibliothèque des facteurs $\mathbf{M}$ sera fixée (approche semi-supervisée), puis nous estimerons également cette bibliothèque (approche non-supervisée). Dans les deux cas, le nombre de facteurs R pourra être estimé implicitement au travers de la matrice des scores ($\mathbf{A}$) et donc directement

à partir des données :

$$R = \#\{r \,|\, \mathbf{a}_{r,\cdot} \neq \mathbf{0}_N^T, r = 1, \ldots, R_{\max}\},$$

où $\mathbf{a}_{r,\cdot} \in \mathbb{R}^N$ correspond à la $r^{\text{ème}}$ ligne de la matrice des scores $\mathbf{A}$.

Organisation du chapitre

Le modèle bayésien hiérarchique développé dans ce chapitre est défini dans le paragraphe 3.2. Une loi Bernoulli-gaussienne tronquée est choisie comme loi *a priori* pour les vecteurs des scores. Cette loi va permettre de prendre en compte les contraintes de positivité, de somme-à-un ainsi que de parcimonie imposées sur les scores. La loi jointe *a posteriori* de ces paramètres et hyperparamètres est comme précédemment trop complexe pour en déduire facilement les expressions d'estimateurs bayésiens classiques (MAP et/ou MMSE). Ainsi, pour résoudre ce problème de complexité, nous proposons d'employer une méthode MCMC. Le paragraphe 3.3 étudie l'échantillonneur de Gibbs hybride qui est employé pour générer des échantillons distribués suivant la loi *a posteriori* des paramètres inconnus. Des résultats de simulations conduites sur des données synthétiques, avec et sans estimation de la bibliothèque des facteurs, sont présentés au paragraphe 3.4, avec notamment des comparaisons avec les méthodes NPBFA [CCP+10] et BFRM [CCL+08]. Le paragraphe 3.5 présentera les résultats obtenus sur des données génétiques réelles.

3.2 Modèle bayésien Bernoulli-gaussien

Le modèle bayésien hiérarchique du chapitre précédent est adapté ici pour la prise en compte de la contrainte de parcimonie sur le vecteur des scores. L'expression de la fonction de vraisemblance reste identique à celle définie dans l'équation (2.2). Une attention particulière est donc portée au choix de la loi *a priori* pour les scores respectant toutes les contraintes sus-citées (parcimonie, non-négativité et somme-à-un).

3.2.1 Lois *a priori* des paramètres et hyperparamètres

Ce paragraphe présente les lois *a priori* des paramètres inconnus et de leurs hyperparamètres associés qui seront utilisées pour ce modèle Bernoulli-gaussien (BeG).

Les lois *a priori* de la bibliothèque des facteurs $\mathbf{M}$, de la variance du bruit σ^2 et de l'hyperparamètre γ détaillées dans la section 2.2.2 sont conservées dans ce chapitre. Seules les lois *a priori* des scores et des hyperparamètres associés seront détaillées dans ce paragraphe.

Loi *a priori* des scores

Considérons tout d'abord une loi normale de moyenne nulle et de variance α^2, tronquée sur l'intervalle $]0, \mu^+]$ et notée $\mathcal{N}_{]0,\mu^+]}\left(0, \alpha^2\right)$. Sa densité de probabilité s'écrit [Rob95] :

$$\varphi_{]0,\mu^+]}(x) = \frac{C}{\sqrt{2\pi\alpha^2}} \exp\left(-\frac{x^2}{2\alpha^2}\right) \mathbf{1}_{]0,\mu^+]}(x)\,. \tag{3.3}$$

Dans (3.3), $C = \left[\Phi\left(\frac{\mu^+}{\alpha}\right) - \frac{1}{2}\right]^{-1}$ est une constante de normalisation où Φ est la fonction de répartition de la loi normale centrée réduite. La troncature sur l'intervalle $]0, \mu^+]$ permet clairement de satisfaire les contraintes de positivité et d'additivité des scores (1.3). La génération d'échantillons distribués suivant la loi normale tronquée (3.3) peut être effectuée avec une stratégie similaire à celle décrite dans [MBI05].

Pour chaque échantillon $i = 1, \ldots, N$, un nombre réduit de facteurs (noté $R < R_{\max}$) appartenant à la bibliothèque $\mathbf{M}$ participent au mélange (3.1). Ceci se traduit par de nombreux $(R_{\max} - R)$ coefficients $a_{r,i}$ égaux à 0. Nous proposons d'utiliser une loi *a priori* exploitant cette propriété de parcimonie des vecteurs $\{\mathbf{a}_i\}_{i=1,\ldots,N}$. En suivant l'approche décrite dans [DHT09], il semble intéressant d'utiliser une loi *a priori* définie par le mélange d'une masse à l'origine (fonction Dirac) et de la loi normale tronquée précédemment définie (3.3). Ainsi, si nous notons $\mathbf{a}_{1:r-1,i}$ $(r = 2, \ldots, R_{\max} - 1)$ le vecteur constitué des $r-1$ premiers éléments du vecteur $\mathbf{a}_i$, la loi *a priori* choisie pour les scores est la loi tronquée Bernoulli-gaussienne suivante :

$$\begin{aligned} a_{1,i} \quad &\sim (1 - w_i)\,\delta\left(a_{1,i}\right) + w_i\,\mathcal{N}_{]0,1]}(0, \alpha^2), \\ a_{r,i}|\mathbf{a}_{1:r-1,i} \quad &\sim (1 - w_i)\,\delta\left(a_{r,i}\right) + w_i\,\mathcal{N}_{]0,\mu^+_{r,i}]}(0, \alpha^2), \end{aligned} \tag{3.4}$$

où $\delta(\cdot)$ est une masse en zéro (Dirac) et w_i est un hyperparamètre inconnu renseignant sur la probabilité *a priori* d'avoir un coefficient non-nul. De plus, pour respecter la contrainte d'additivité, la loi normale associée aux termes non-nuls du mélange est tronquée à droite par $\mu^+_{r,i}$ $(r = 2, \ldots, R_{\max} - 1)$ et le dernier élément du vecteur des abondances est fixé à $a_{R_{\max},i}$, tels que définis ci-dessous :

$$\begin{aligned} \mu^+_{r,i} &= 1 - \textstyle\sum_{j=1}^{r-1} a_{j,i}, \\ a_{R_{\max},i} = \mu^+_{R_{\max},i} &\triangleq 1 - \textstyle\sum_{r=1}^{R_{\max}-1} a_{r,i}. \end{aligned} \tag{3.5}$$

Ainsi, la distribution *a priori* pour le vecteur des proportions $\mathbf{a}_i$ $(i = 1, \ldots, N)$ dont le dernier élément $a_{R_{\max},i}$ est fixé à $\mu^+_{R_{\max},i}$ s'écrit de manière récursive :

$$f(\mathbf{a}_i) = f(a_{1,i}) \left[\prod_{r=2}^{R_{\max}-1} f(a_{r,i} | \mathbf{a}_{1:r-1,i}) \right] \delta\left(a_{R_{\max},i} - \mu^+_{R_{\max},i} \right).$$

Notons $\mathcal{I}_{0,i} = \{r \,|\, a_{r,i} = 0\}$ et $\mathcal{I}_{1,i} = \{r \,|\, a_{r,i} \neq 0\} = \overline{\mathcal{I}_{0,i}}$, pour $i = 1, \ldots, N$. L'équation précédente se réécrit alors :

$$f(\mathbf{a}_i | w_i, \alpha^2) \propto \left[(1 - w_i)^{n_{0,i}} \prod_{r \in \mathcal{I}_{0,i}} \delta(a_{r,i}) \right] \left[\left(\frac{w_i}{\sqrt{2\pi\alpha^2}} \right)^{n_{1,i}} \prod_{r \in \mathcal{I}_{1,i}} \exp\left(-\frac{a^2_{r,i}}{2\alpha^2} \right) \mathbf{1}_{]0,\mu^+]}(a_{r,i}) \right], \tag{3.6}$$

où $n_{x,i} = \text{card}\{\mathcal{I}_{x,i}\}$ $(x = 0, 1)$. Remarquons que $n_{1,i} = \|\mathbf{a}_i\|_0$ où $\|\cdot\|_0$ est la norme l_0 : $\|\mathbf{a}_i\|_0 = \sharp\{r \,|\, a_{r,i} \neq 0\}$, et $n_{0,i} = R_{\max} - n_{1,i}$.

En supposant que les vecteurs des scores $\{\mathbf{a}_i\}_{i=1,\ldots,N}$ sont *a priori* indépendants, on obtient la loi jointe *a priori* suivante pour la matrice des scores $\mathbf{A}$ $(\mathbf{A} \in \mathbb{R}^{R_{\max} \times N})$:

$$f(\mathbf{A}) = \prod_{i=1}^{N} f\left(\mathbf{a}_i | w_i, \alpha^2\right).$$

Loi *a priori* de la proportion moyenne des scores non-nuls

Notons $\boldsymbol{w} = [w_1, \ldots, w_N]^T$. Une loi uniforme sur l'ensemble $[0, 1]$ est choisie comme loi *a priori* pour la proportion moyenne des scores non-nuls :

$$w_i \sim \mathcal{U}([0, 1]), \quad i = 1, \ldots, N. \tag{3.7}$$

En supposant que tous les hyperparamètres de ce modèle bayésien sont statistiquement indépendants, la loi du vecteur d'hyperparamètres est :

$$f(\boldsymbol{\Psi}) = f(\boldsymbol{w}) f(\gamma) \propto \frac{1}{\gamma} \prod_{i=1}^{N} \mathbf{1}_{[0,1]}(w_i)\, \mathbf{1}_{\mathbb{R}^+}(\gamma)\,. \tag{3.8}$$

Résumé des lois *a priori* du modèle BeG

La structure hiérarchique du modèle BeG proposé est représentée dans le DAG de la figure 3.1. Les lois *a priori* choisies pour les paramètres inconnus sont rappelées et résumées ci-après :

$$\begin{aligned}
\mathbf{m}_r &= \mathbf{U}\mathbf{t}_r + \bar{\mathbf{y}} && (2.4)\\
\mathbf{t}_r|\mathbf{e}_r, s_r^2 &\sim \mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2 \mathbf{I}_{R-1}\right) && (2.5)\\
a_{1,i} &\sim (1-w_i)\,\delta(a_{1,i}) + w_i\,\mathcal{N}_{]0,1]}(0,\alpha^2) && (3.4)\\
a_{r,i}|\mathbf{a}_{1:r-1,i} &\sim (1-w_i)\,\delta(a_{r,i}) + w_i\,\mathcal{N}_{]0,\mu_{r,i}^+]}(0,\alpha^2) && (3.4)\\
a_{R_{\max},i} &= 1 - \textstyle\sum_{r=1}^{R_{\max}-1} a_{r,i} && (3.5)\\
w_i &\sim \mathcal{U}([0,1]) && (3.7)\\
\sigma^2|\nu,\gamma &\sim \mathcal{IG}\left(\tfrac{\nu}{2}, \tfrac{\gamma}{2}\right) && (2.12)
\end{aligned}$$

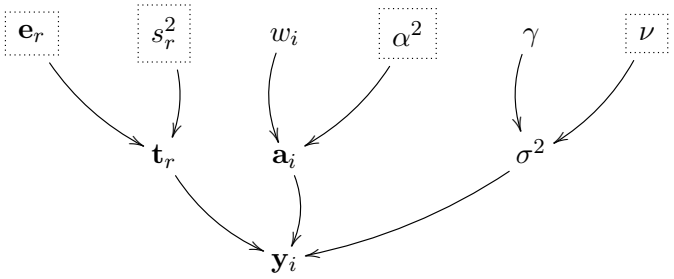


FIGURE 3.1 – DAG pour les lois *a priori* des paramètres et hyperparamètres du modèle bayésien Bernoulli-gaussien (BeG) (les paramètres fixés apparaissent dans les cases en pointillés).

3.2.2 Loi *a posteriori*

La loi *a posteriori* jointe des vecteurs des paramètres inconnus $\boldsymbol{\Theta} = \{\mathbf{T}, \mathbf{A}, \sigma^2\}$ et hyperparamètres $\boldsymbol{\Psi} = \{\boldsymbol{w}, \gamma\}$ s'écrit :

$$f(\boldsymbol{\Theta}, \boldsymbol{\Psi}|\mathbf{Y}) \propto f(\mathbf{Y}|\boldsymbol{\Theta})\, f(\boldsymbol{\Theta}|\boldsymbol{\Psi})\, f(\boldsymbol{\Psi}) \tag{3.9}$$

où $f(\mathbf{Y}|\mathbf{\Theta})$ et $f(\mathbf{\Psi})$ ont été respectivement définis dans (2.2) et (3.8). Sous l'hypothèse que les paramètres sont *a priori* indépendants, on obtient :

$$f(\mathbf{\Theta}|\mathbf{\Psi}) = f(\mathbf{T})\, f\left(\mathbf{A}|\boldsymbol{w}, \alpha^2\right) f\left(\sigma^2|\nu, \gamma\right). \tag{3.10}$$

Ainsi, en intégrant l'hyperparamètre $\boldsymbol{w}$ et la variance de l'erreur résiduelle σ^2, on arrive à :

$$f(\mathbf{a}_i|\mathbf{y}_i) \propto \int\int f\left(\mathbf{a}_i, \boldsymbol{w}, \sigma^2|\mathbf{y}_i\right) \mathrm{d}\boldsymbol{w}\, \mathrm{d}\sigma^2$$

et :

$$f(\mathbf{a}_i|\mathbf{y}_i) \propto \frac{\mathrm{B}(1+n_{0,i}; 1+n_{1,i})\,\Gamma\left(\frac{G}{2}\right)}{\|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\|^G} \exp\left(-\frac{\sum_{r\in\mathcal{I}_{1,i}} a_{r,i}^2}{2\alpha^2}\right) \tag{3.11}$$

où $\Gamma(\cdot)$ représente la fonction Gamma et $\mathrm{B}(a,b) = \Gamma(a)\Gamma(b)/\Gamma(a+b)$ la fonction Beta.

3.3 Echantillonneur de Gibbs hybride

Cette partie présente l'algorithme de Gibbs utilisé pour générer aléatoirement des échantillons asymptotiquement distribués suivant la loi *a posteriori* d'intérêt. Plus précisément, les lois conditionnelles de $f\left(\mathbf{T}, \mathbf{A}, \sigma^2, \boldsymbol{w}|\mathbf{Y}\right)$ sont décrites ci-dessous. Les différentes étapes de l'algorithme de Gibbs sont détaillées dans l'algorithme 3.1.

3.3.1 Echantillonnage suivant $f(\mathbf{A}|\boldsymbol{w}, \mathbf{T}, \sigma^2, \mathbf{Y})$

Après calculs, la loi conditionnelle des scores $a_{r,i}$ est une loi Bernoulli-gaussienne tronquée de paramètres $(\widetilde{w}_{r,i}, \mu_{r,i}, \eta_{r,i}^2, \mu_{r,i}^+)$, pour $r = 1, \ldots, R_{\max}$ et $i = 1, \ldots, N$:

$$a_{r,i}|w_i, \sigma^2, \mathbf{a}_{\setminus r,i}, \mathbf{y}_i \sim (1-\widetilde{w}_{r,i})\delta(a_{r,i}) + \widetilde{w}_{r,i}\mathcal{N}_{]0,\mu_{r,i}^+[}\left(\mu_{r,i}, \eta_{r,i}^2\right) \tag{3.12}$$

où $\mathbf{a}_{\setminus r,i}$ correspond au vecteur $\mathbf{a}_i$ privé de sa $r^{\text{ème}}$ composante et :

$$\begin{cases} \widetilde{w}_{r,i} &= \frac{u_{r,i}}{u_{r,i}+(1-w_i)}, \\ u_{r,i} &= w_i \frac{\eta_{r,i}}{\alpha} \exp\left(\frac{\mu_{r,i}^2}{2\eta_{r,i}^2}\right)\left[\Phi\left(\frac{\mu_{r,i}^+ - \mu_{r,i}}{\eta_{r,i}}\right) - \Phi\left(\frac{-\mu_{r,i}}{\eta_{r,i}}\right)\right], \\ \eta_{r,i}^2 &= \left(\frac{\|\mathbf{m}_r\|^2}{\sigma^2} + \frac{1}{\alpha^2}\right)^{-1}, \\ \mu_{r,i} &= \eta_{r,i}^2\left(\frac{\mathbf{m}_r^T \boldsymbol{\epsilon}_{\setminus r}}{\sigma^2}\right), \\ \boldsymbol{\epsilon}_{\setminus r} &= \mathbf{y}_i - \sum_{j=1, j\neq r}^{R_{\max}} \mathbf{m}_j a_{j,i}. \end{cases} \tag{3.13}$$

ALGO. 3.1 – Echantillonneur de Gibbs hybride pour le modèle Bernoulli-gaussien.

- Pré-traitements :
  - Calculer la moyenne empirique des échantillons $\bar{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{y}_i$ définie dans (2.4),
  - Déterminer la matrice de projection $\mathbf{P}$ (de taille $R_{\max} - 1 \times G$) à l'aide d'une technique de réduction de dimensionnalité (comme l'ACP [Jol86]),
  - Choisir les vecteurs moyennes $\{\mathbf{e}_r\}_{r=1,\ldots,R_{\max}}$ définis dans (2.5) comme les solutions d'un algorithme d'extraction de pôles de mélange (comme l'algorithme VCA [NBD05]).
- Initialisation ($\ell = 0$) :
  - Echantillonner la matrice des signatures projetées $\mathbf{T}^{(0)}$ selon la loi *a priori* (2.8),
  - Reconstruire la matrice des signatures $\mathbf{M}^{(0)}$ à partir de la matrice des signatures projetées $\mathbf{T}^{(0)}$: $\mathbf{M}^{(0)} = \mathbf{P}^{-1}\mathbf{T}^{(0)} + \bar{\mathbf{y}}\mathbf{1}_{R^{(0)}}^T$ (2.4),
  - Pour $r = 1, \ldots, R_{\max}$ et $i = 1, \ldots, N$, générer les scores $a_{r,i}^{(0)}$ suivant (3.4),
  - Générer la variance du bruit $\sigma^{2(0)}$ suivant la loi définie dans (2.12),
  - Poser $\ell \leftarrow 1$.
- Itérations : Pour $\ell = 1, 2, \ldots, \mathrm{N_{mc}}$, faire :
  1. Echantillonner l'hyperparamètre $\boldsymbol{w}^{(\ell)}$ suivant (3.14),
  2. Pour $r = 1, \ldots, R_{\max}$ et $i = 1, \ldots, N$, générer les scores $a_{r,i}^{(\ell)}$ suivant (3.12),
  3. Echantillonner $\mathbf{T}^{(\ell)}$ selon la loi (2.16),
  4. Construire $\mathbf{M}^{(\ell)} = \mathbf{P}^{-1}\mathbf{T}^{(\ell)} + \bar{\mathbf{y}}\mathbf{1}_{R^{(\ell)}}^T$ (2.4),
  5. Echantillonner $\sigma^{2(\ell)}$ selon la loi (2.18),
  6. Poser $\ell \leftarrow \ell + 1$.

Notons que cette loi *a posteriori* Bernoulli-gaussienne tronquée sur l'ensemble $]0, \mu_{r,i}^{+}[$ permet de respecter les contraintes d'additivité et de positivité (1.3).

3.3.2 Echantillonnage suivant $f(\boldsymbol{w}|\mathbf{A})$

La génération d'échantillons distribués suivant $f(\boldsymbol{w}|\mathbf{A})$ se fait selon la loi Beta suivante (pour $i = 1, \ldots, N$) :

$$w_i|\mathbf{a}_i \sim \mathcal{B}(1 + n_{1,i}, 1 + n_{0,i}), \tag{3.14}$$

avec $n_{1,i} = \sharp\{r|a_{r,i} \neq 0\}$ et $n_{0,i} = R_{\max} - n_{1,i}$.

3.4 Résultats de simulation sur données synthétiques

Afin d'évaluer les performances de l'algorithme BeG proposé, nous allons l'appliquer sur des données synthétiques et comparer ces résultats avec ceux obtenus par d'autres algorithmes de décomposition matricielle parcimonieux.

Scénario de simulation

Les données synthétiques générées sont constituées de $N = 128$ échantillons, chacun composé exactement de $R = 3$ facteurs, parmi une bibliothèque $\mathbf{M}$ de $R_{\max} = 9$ signatures possibles de $G = 256$ gènes. Par souci de simplicité et sans perte de généralité, nous fixons, pour tous les échantillons $i = 1, \ldots, N$: $R = 3$ et $\mathbf{m}_1, \ldots, \mathbf{m}_R$ sont les seules et les mêmes signatures présentes. Les scores ont été générés aléatoirement suivant une distribution de Dirichlet $\mathcal{D}(1, \ldots, 1)$ et les échantillons sont bruités avec un rapport signal-à-bruit fixé à $\mathrm{SNR}_i = 20$ dB ($\mathrm{SNR}_i = G^{-1}\sigma^{-2}\left\|\sum_{r=1}^{R}\mathbf{m}_r a_{r,i}\right\|^2$, $i = 1, \ldots, N$).

Plus précisément, nous avons utilisé le jeu de données $\mathcal{J}_1$ décrit dans le paragraphe 2.5 où chaque échantillon était composé de $R = 3$ facteurs piqués. Nous avons alors complété la bibliothèque $\mathbf{M}$ par $R_{\max} - R = 6$ autres facteurs de même nature et dont les scores correspondants pour chaque échantillon sont nuls.

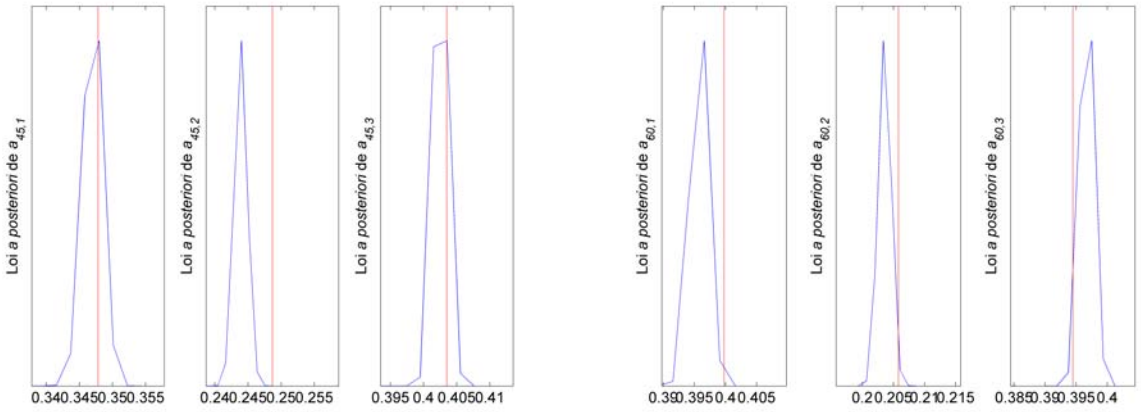
3.4.1 Estimation des scores

Dans un premier temps de l'analyse, les signatures sont supposées connues et appartenir à une bibliothèque $\mathbf{M}$ fixée. Nous nous intéresserons donc uniquement à l'estimation des scores, respectant les contraintes de positivité, de somme-à-un et de parcimonie.

Les échantillons générés des scores $\{\mathbf{a}_i^{(\ell)}\}_{\ell=1,\ldots,\mathrm{N_{mc}}}$ $(i = 1, \ldots, N)$ sont utilisés afin de déterminer l'estimateur MMSE du score $\mathbf{a}_i$ du $i^{\text{ème}}$ échantillon :

$$\widehat{\mathbf{a}}_{i\mathrm{MMSE}} \approx \frac{1}{\mathrm{N_{mc}} - \mathrm{N_{bi}}} \sum_{\ell=\mathrm{N_{bi}}+1}^{\mathrm{N_{mc}}} \mathbf{a}_i^{(\ell)}. \tag{3.15}$$

Les distributions *a posteriori* des scores non-nuls $(r = 1, \ldots, 3)$ des échantillons ♯45 et ♯60 sont représentées sur la figure 3.2. Ces distributions ont été obtenues par moyennage de $C = 10$ chaînes MCMC. Les résultats obtenus pour ces deux échantillons sont en accord avec les valeurs réelles tracées en rouge. Les valeurs obtenues pour les autres facteurs $(r = 4, \ldots, R_{\max})$, non représentées ici, sont bien nulles ou quasi-nulles.



(a) Scores non-nuls de l'échantillon ♯45

(b) Scores non-nuls de l'échantillon ♯60

FIGURE 3.2 – Distribution *a posteriori* des scores non-nuls $a_{i,j}$ $(j = \{r | a_{i,r} \neq 0\})$ de deux échantillons.

Nous comparons la méthode Bernoulli-gaussienne (BeG) développée dans ce chapitre avec :

- l'algorithme "*fully constrained least-squares*" (FCLS) détaillé dans [HC01] : algorithme itératif qui minimise au sens des moindres carrés et sous les contraintes de positivité et de somme-à-un le critère suivant : $J(\mathbf{a}_i) = \|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\|^2$, pour $i = 1, \ldots, N$,
- la méthode LASSO proposée par Themelis *et al.* [TRK10] : algorithme itératif qui consiste à minimiser le critère précédent $J(\mathbf{a}_i)$ pénalisé par un terme de norme l_1 favorisant les solutions parcimonieuses,
- l'analyse factorielle bayésienne non-paramétrique de Chen *et al.* (NPBFA) [CCP+10] : algorithme qui utilise un processus de buffet indien pour prendre en compte la parcimonie des scores.

Les signatures étant connues, la matrice $\mathbf{M}$ sera fixée pour la méthode proposée et la méthode NPBFA. Dans l'échantillonneur de Gibbs, cela consiste à enlever l'étape d'échantillonnage des signatures projetées (2.16).

Afin d'évaluer les performances de ces algorithmes, les erreurs quadratiques moyennes globales (GMSEs) ont été calculées à partir des estimateurs MMSE des scores $\widehat{\mathbf{a}}_{i\text{MMSE}}$ $(i = 1, \ldots, N)$ comme suit :

$$\text{GMSE}^2 = \frac{1}{N}\sum_{i=1}^{N} \|\widehat{\mathbf{a}}_{i\text{MMSE}} - \mathbf{a}_i\|^2 .$$

Les résultats obtenus sont présentés dans la figure 3.3, où les GMSEs sont représentées en fonction du rapport signal-à-bruit (SNR). Cette figure montre que la méthode BeG obtient des résultats meilleurs que la méthode LASSO [TRK10] et l'analyse non-paramétrique NPBFA [CCP+10], pour des SNR inférieurs à 25 dB. L'algorithme FCLS reste le plus performant et ce quel que soit le SNR. En revanche, contrairement à l'algorithme FCLS et la méthode LASSO, les méthodes NPBFA et BeG permettent d'estimer conjointement les facteurs et les scores, comme dans le paragraphe suivant.

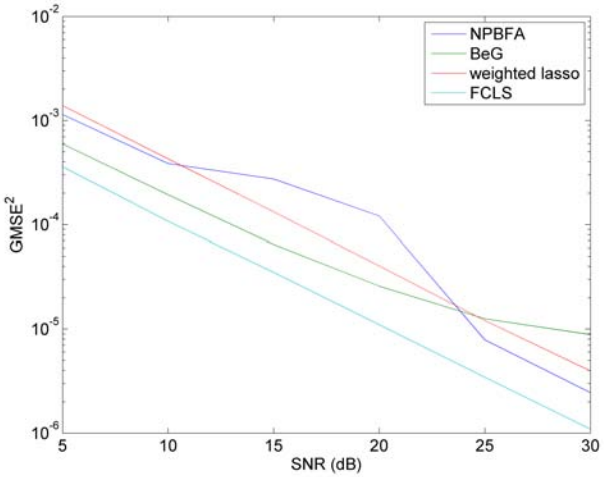


FIGURE 3.3 – Résultats de simulation de l'algorithme BeG sur données synthétiques avec bibliothèque fixée ($\mathrm{GMSE}^2 = f(\mathrm{SNR})$).

3.4.2 Estimation des facteurs et des scores

Dans ce paragraphe, la bibliothèque des facteurs $\mathbf{M}$ n'est plus connue et est donc estimée conjointement aux scores. Les vecteurs moyennes cachés $\mathbf{e}_r$ $(r = 1, \ldots, R_{\max})$ nécessaires pour évaluer la loi *a priori* des signatures génétiques (loi introduite dans le paragraphe 2.2.2) sont choisis comme les projections des signatures identifiées par une analyse VCA [NBD05].

Afin d'évaluer les performances de l'algorithme proposé (BeG), nous allons le comparer avec la méthode non-paramétrique bayésienne (NPBFA) [CCP+10], le modèle uBLU développé dans le chapitre 2, l'analyse ACP, l'algorithme NMF [LS00], le modèle BFRM de Carvalho *et al.* [CCL+08] et la décomposition GB-GMF [NHN+11]. Les résultats de simulations sont reportés dans le tableau 3.1 en utilisant les mêmes critères de comparaison que ceux définis dans le paragraphe 2.5.3 (MSEs, GMSEs, SADs, GSAD, erreur de reconstruction RE et temps de calcul). Les méthodes ACP, NMF et GB-GMF sont appliquées en fixant le nombre de facteurs à trouver à $R = 3$, alors que les autres méthodes autorisent une recherche parcimonieuse ou permettent l'estimation du nombre de facteurs. L'estimation MMSE du nombre de facteurs est : $\widehat{R}_{\mathrm{MMSE}} = 3$, en accord avec la valeur théorique.

TABLE 3.1 – Comparaison des performances d'estimation entre différents algorithmes et l'approche BeG proposée.

		BeG	NPBFA	uBLU	ACP	NMF	BFRM	GB-GMF
MSE^2 $(\times 10^{-1})$	Facteur 1	1,24	7,17	0,13	0,72	0,13	20,22	6,21
	Facteur 2	0,17	17,50	0,20	0,86	0,12	15,84	15,82
	Facteur 3	0,19	16,09	0,12	0,70	0,11	12,63	16,37
$GMSE^2$ $(\times 10^{-2})$	Facteur 1	5,91	7,52	0,01	1,75	0,01	9,98	6,04
	Facteur 2	7,53	5,96	0,02	2,69	0,01	9,02	2,89
	Facteur 3	8,40	7,77	0,01	0,92	0,01	7,80	2,36
SAD $(\times 10^{-1})$	Facteur 1	2,19	3,57	0,88	2,05	0,89	11,92	13,09
	Facteur 2	0,54	1,58	0,55	1,35	0,51	16,05	17,20
	Facteur 3	0,55	3,46	0,51	1,25	0,51	16,15	16,29
GSAD $(\times 10^{-2})$		6,79	25,84	6,73	14,78	6,71	118,17	6,74
RE $(\times 10^{-2})$		0,75	47,88	0,72	24,01	0,72	120,12	17,09
Temps de calcul (en s)		7624	9950	2892	0,1	2,3	55,19	639

Ces résultats illustrent la précision de la méthode BeG proposée. En outre, nous pouvons noter que cette approche BeG pour les données génétiques est plus performante que l'approche parcimonieuse non-paramétrique NPBFA et reste compétitive par rapport à l'approche précédente uBLU bien que la complexité calculatoire soit plus importante.

3.4.3 Contrôle de la convergence

Les résultats présentés dans les deux paragraphes précédents ont été obtenus pour $N_{mc} = 10\ 000$ itérations dont $N_{bi} = 3\ 000$ itérations dans la période de chauffage. Il est donc intéressant de regarder si ces valeurs sont suffisantes pour s'assurer de la bonne convergence de l'échantillonneur, comme nous l'avions vérifié pour l'algorithme uBLU.

Pour cela, nous utilisons le critère de variance intra/inter-chaîne défini par Gelman et Rubin [GR92]. Ce critère consiste tout d'abord à générer C chaînes de Markov en parallèle, de longueur

$\mathrm{N_r} = \mathrm{N_{mc}} - \mathrm{N_{bi}}$, mais avec différentes valeurs initiales. Les variances inter-chaîne B et intra-chaîne W pour ces C chaînes sont respectivement définies par :

$$\begin{cases} B = \dfrac{\mathrm{N_r}}{C-1}\displaystyle\sum_{c=1}^{C}\left(\overline{\theta}_c - \overline{\theta}\right)^2, \\ W = \dfrac{1}{C}\displaystyle\sum_{c=1}^{C}\dfrac{1}{\mathrm{N_r}-1}\sum_{\ell=1}^{\mathrm{N_r}}\left(\theta_c^{(\ell)} - \overline{\theta}_c\right)^2, \end{cases} \text{avec} \quad \begin{cases} \overline{\theta}_c = \dfrac{1}{\mathrm{N_r}}\displaystyle\sum_{\ell=1}^{\mathrm{N_r}}\theta_c^{(\ell)}, \\ \overline{\theta} = \dfrac{1}{C}\displaystyle\sum_{c=1}^{C}\overline{\theta}_c, \end{cases} \tag{3.16}$$

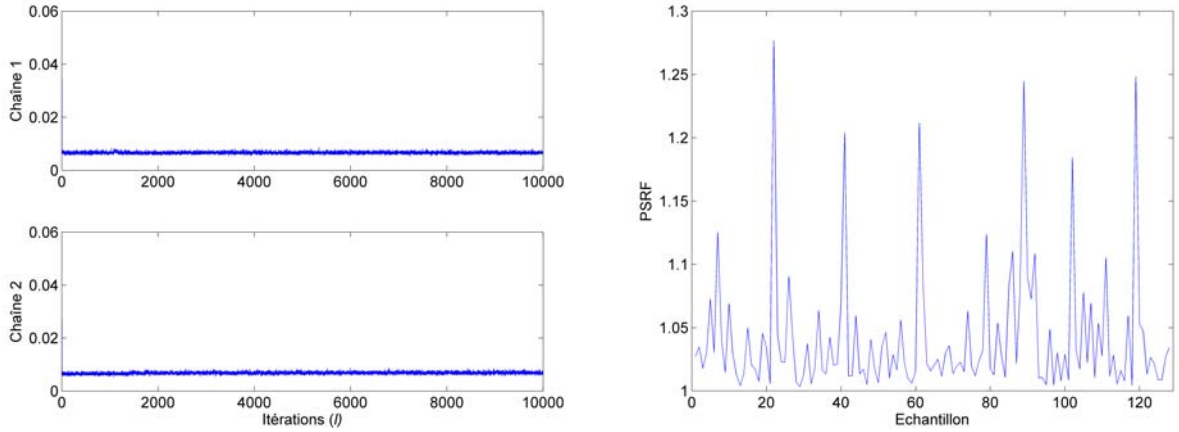
où θ est le paramètre d'intérêt ($\theta \in \boldsymbol{\Theta}$) et $\theta_c^{(\ell)}$ est le $\ell^{\text{ème}}$ échantillon de θ de la $c^{\text{ème}}$ chaîne MCMC. Puis la convergence des chaînes est mesurée à l'aide du potentiel d'échelle $PSRF$ [1] défini par [GCSR03] :

$$PSRF = \sqrt{\frac{1}{W}\left(\frac{\mathrm{N_r}-1}{\mathrm{N_r}}W + \frac{1}{\mathrm{N_r}}B\right)}. \tag{3.17}$$

Une valeur de $PSRF$ inférieure à 1,2 indique une bonne convergence de l'échantillonneur [GCSR03], i.e., un nombre d'itérations de chauffage égal à $\mathrm{N_{bi}}$ est suffisant pour obtenir des échantillons $\{\boldsymbol{\Theta}^{(\ell)}\}_{\ell=1,\ldots,\mathrm{N_r}}$ distribués suivant la loi cible $f(\boldsymbol{\Theta}, \boldsymbol{\Psi}|\mathbf{Y})$.

Comme au chapitre précédent, nous choisissons de suivre la convergence de l'échantillonneur avec la variance du bruit σ^2. Pour évaluer le potentiel d'échelle $PSRF$, nous construisons $C = 10$ chaînes MCMC de longueur $\mathrm{N_{mc}} = 10\,000$ itérations dont $\mathrm{N_{bi}} = 3\,000$ itérations de chauffe. La figure 3.4a montre que les deux chaînes MCMC représentées, pour le même échantillon ♯45, convergent clairement vers une valeur identique : $\sigma^2 \approx 0{,}7.10^{-2}$. Les valeurs obtenues pour les potentiels d'échelle $PSRF$, calculées selon (3.17) et représentées sur la figure 3.4b, sont bien inférieures à 1,2 pour une très grande majorité d'échantillons (moyenne de 1,04 sur tous les échantillons). Ceci confirme la convergence de l'échantillonneur de Gibbs.

1. Remarque : L'algorithme BeG ne fait pas de sélection de modèles, contrairement à l'algorithme uBLU. Le critère $PSRF$ est donc simplifié par rapport à celui défini dans le paragraphe 2.4.1.



(a) Exemples de deux chaînes MCMC relatives au paramètre σ^2 du $45^{\text{ème}}$ échantillon.

(b) Potentiels d'échelle $PSRF$ calculés selon (3.17) pour tous les échantillons.

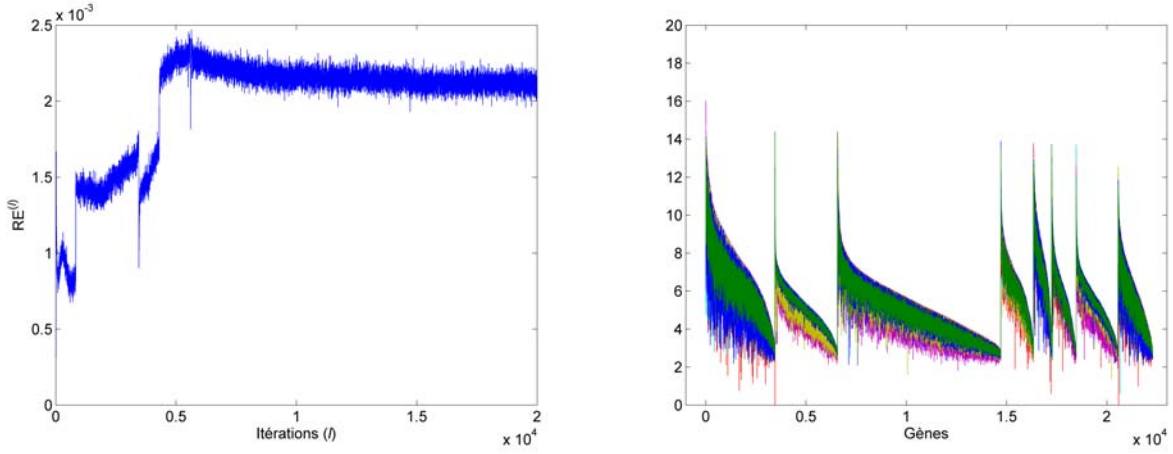
FIGURE 3.4 – Contrôle de la convergence de l'algorithme BeG sur données synthétiques.

3.5 Applications sur données réelles génétiques

Les résultats obtenus sur données synthétiques parcimonieuses ont montré l'efficacité de la méthode BeG proposée. Ce paragraphe applique la méthode BeG aux trois jeux de données réelles à notre disposition.

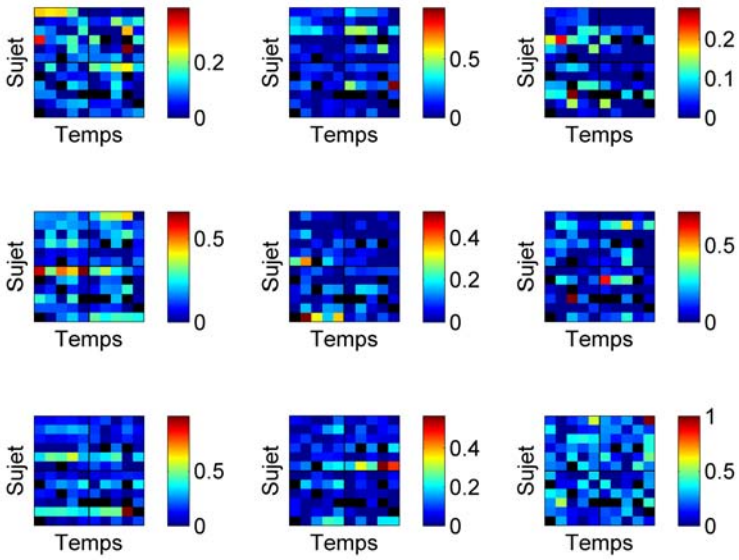
3.5.1 Données de boissons

Dans un premier temps, nous allons évaluer l'algorithme BeG sur les données de boissons de Baty *et al.* [BFW+06] en prennant $\mathrm{N_{mc}} = 20\ 000$ itérations de Monte Carlo, dont $\mathrm{N_{bi}} = 5\ 000$ itérations de chauffe. La figure 3.5a permet de visualiser l'erreur de reconstruction $\mathrm{RE}^{(\ell)}$ calculée à chaque itération ℓ ($\ell = 1, \ldots, \mathrm{N_{mc}}$) selon (2.23) et de s'assurer de la bonne convergence de l'échantillonneur de Gibbs sur ces données, pour les valeurs de $\mathrm{N_{mc}}$ et $\mathrm{N_{bi}}$ considérées.



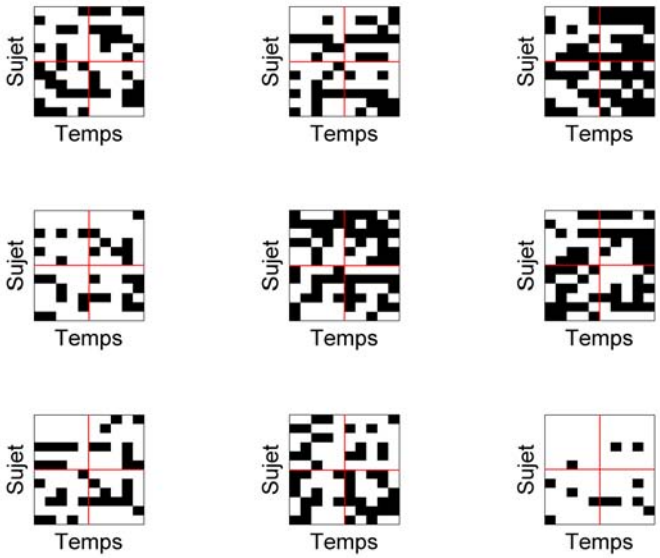
(a) Diagnostic de convergence : erreur de reconstruction $\mathrm{RE}^{(\ell)}$ en fonction du nombre d'itérations ℓ

(b) Facteurs estimés, rangés par dominance décroissante

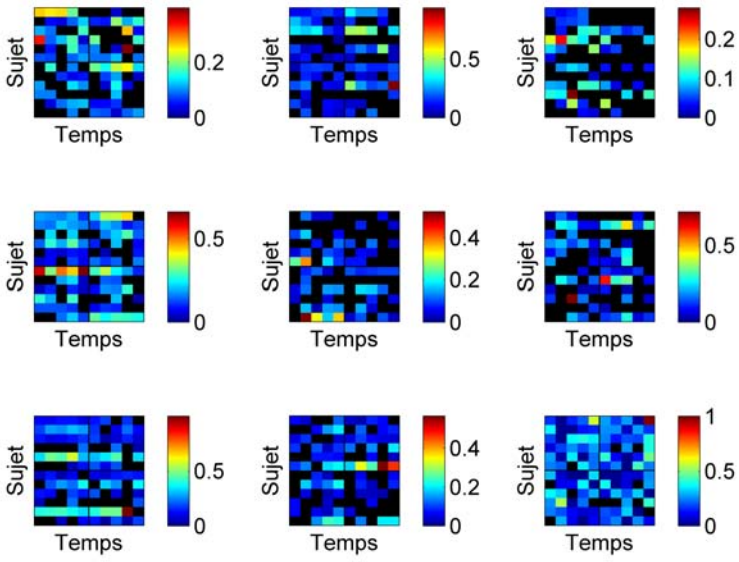


(c) Scores estimés pour les $R_{\max} = 9$ facteurs

FIGURE 3.5 – Résultats de simulation de l'algorithme BeG sur données de boissons [BFW+06].



(a) Matrice binaire de présence (cases blanches) / absence (cases noires) des facteurs dans les échantillons



(b) Reconstruction des scores (les cases noires correspondent aux scores nuls)

FIGURE 3.6 – Représentations des scores déterminés par l'algorithme BeG sur données de boissons [BFW+06].

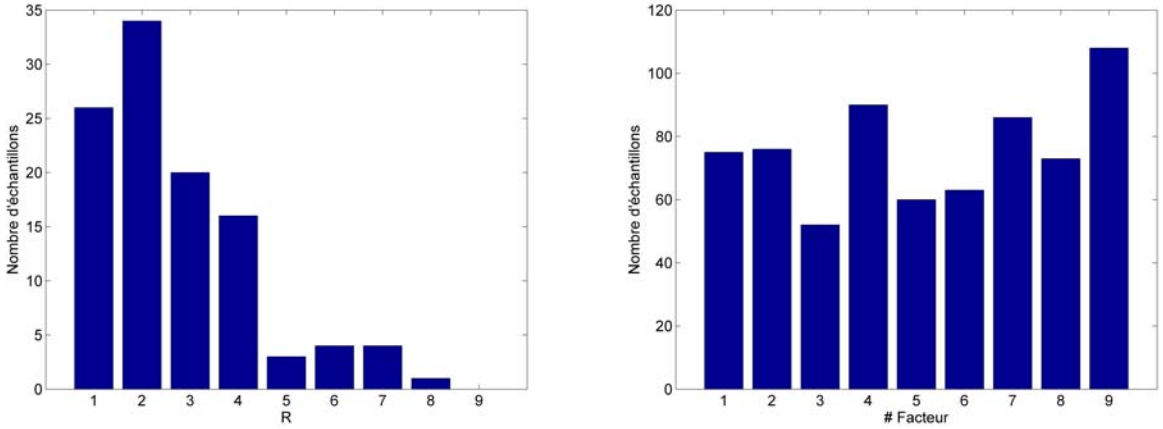
L'analyse BeG est appliquée sur ces données en fixant $R_{\max} = 9$ signatures présentes au maximum et représentées sur la figure 3.5b, avec la même ré-organisation des indices des gènes que celle utilisée au chapitre précédent. Fixer $R_{\max} = 9$ permet de réduire considérablement la dimensionalité du problème, en conservant tout de même une énergie cumulée suffisante. Les $R_{\max} = 9$ facteurs découverts sont dominants pour des groupes de gènes respectivement de taille [3 449, 3 119, 8 146, 1 626, 906, 1 222, 2 090, 1 725]. On remarque qu'il n'y a que 8 groupes de gènes, cela signifie qu'un des 9 facteurs n'est dominant pour aucun gène.

Les scores pour chacune des $R_{\max} = 9$ signatures, classées selon leur dominance (figure 3.5b), sont représentés sur la figure 3.5c sous forme d'images, selon l'organisation de la figure 1.5. Sur ces figures, les cases noires correspondent à des échantillons non exploités. Cependant, on peut remarquer sur ces figures que de nombreux scores sont égaux à 0. Les figures 3.6a et 3.6b permettent de mieux visualiser les scores nuls (cases noires). En particulier, les figures 3.6a permettent de visualiser pour chaque échantillon quels facteurs sont présents dans son mélange. La figure 3.7a est l'histogramme du nombre de facteurs R par échantillon. En moyenne, chaque échantillon est composé de $\widehat{R}_{\text{MMSE}} = 2{,}67$ facteurs. Enfin, la figure 3.7b compte le nombre d'échantillons contenant chaque facteur.

A partir de ces résultats (notamment les figures 3.6b), nous pouvons remarquer que les facteurs #1 et #4 semblent être associés au sujet #1, le facteur #7 au sujet #5, ... Cependant, comme avec l'algorithme précédent, il est toujours difficile de conclure sur un réel lien entre sujet, boisson et gènes.

3.5.2 Données de grippes H3N2

Dans un deuxième temps, nous avons appliqué l'algorithme BeG aux données de grippe H3N2 détaillées dans le paragraphe 1.5.2 avec $\text{N}_{\text{mc}} = 20\,000$ itérations de Monte Carlo, dont $\text{N}_{\text{bi}} = 5\,000$ itérations de chauffage, et $R_{\max} = 9$ signatures au maximum. La figure 3.8a justifie la convergence de l'échantillonneur de Gibbs pour ces valeurs de N_{mc} et N_{bi}.

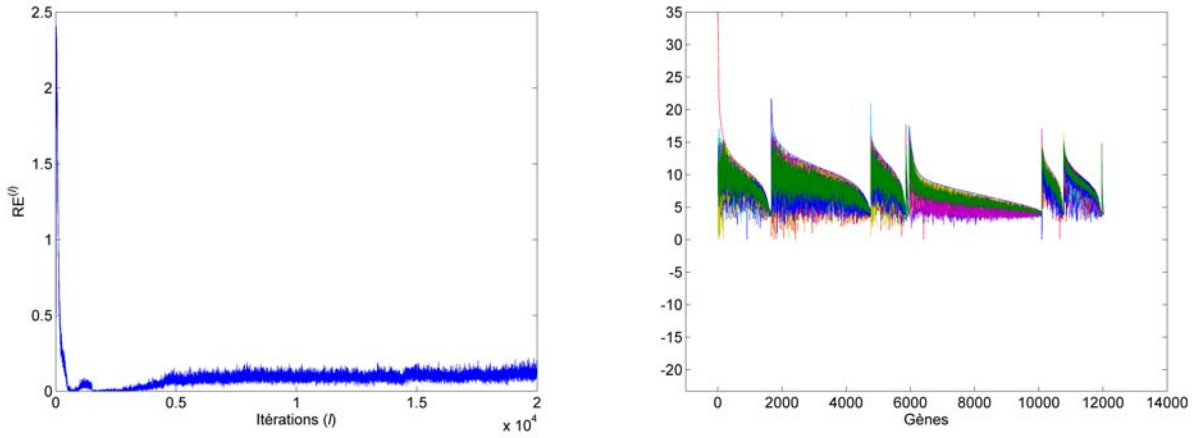


(a) Histogramme du nombre de facteurs par échantillon (b) Nombre d'échantillons pour chaque facteur

FIGURE 3.7 – Nombres de facteurs par échantillon et d'échantillon par facteur, sur données de boissons [BFW+06].

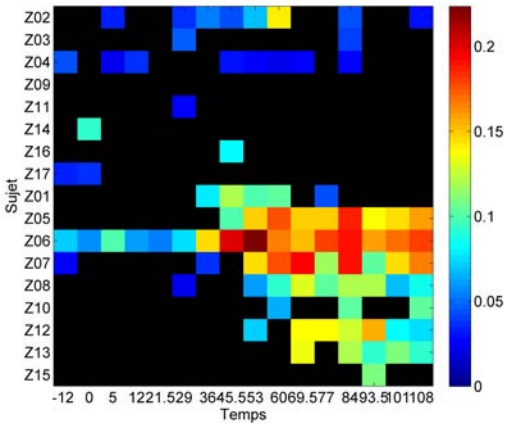
Les résultats obtenus sont présentés dans la figure 3.8 et les tables 3.2 et 3.3. Plus particulièrement, les $R_{\max} = 9$ signatures présentes dans les données H3N2 sont représentées dans la figure 3.8b, où les facteurs sont organisés par ordre de dominance décroissante. La figure 3.8c correspond à la carte des scores associés au facteur le plus dominant ♯1 (signature génétique en rouge dans la figure 3.8b). Les cases en noir correspondent soit aux 5 échantillons non analysés, soit à des échantillons qui ne contiennent pas le facteur étudié (scores nuls). De part sa structure, similaire à celle obtenue avec l'algorithme uBLU (figure 2.8c), ce facteur correspond donc au facteur inflammatoire.

L'analyse des pathways va permettre de quantifier la capacité des 1 668 gènes de cette composante inflammatoire déterminée par l'algorithme BeG à appartenir à certains groupements de gènes connus et à être liés à des processus biologiques intervenant suite à une infection. Les résultats, présentés dans le tableau 3.2, montrent que les quatre pathways les plus représentés dans cette composante inflammatoire (TCPTP, IL23, IL12 et IFN-γ) étaient également présents dans la composante inflammatoire déterminée par l'algorithme uBLU (cf. table 2.3) bien que cette composante contienne moins de gènes que celle déterminée par l'algorithme uBLU (1 668 gènes contre 2 297 gènes inflammatoires).



(a) Diagnostic de convergence : erreur de reconstruction (b) Facteurs estimés, rangés par dominance décroissante

$\mathrm{RE}^{(\ell)}$ en fonction du nombre d'itérations ℓ



(c) Reconstruction des scores du facteur inflammatoire

FIGURE 3.8 – Résultats de simulation de l'algorithme BeG sur données H3N2 [ZCV+09].

La méthode BeG apparaît donc comme une méthode plus sélective des gènes inflammatoires par rapport à la méthode uBLU, ceci est notamment dû à la contrainte de parcimonie imposée.

TABLE 3.2 – Classement des pathways des gènes de la composante inflammatoire de BeG sur les données H3N2.

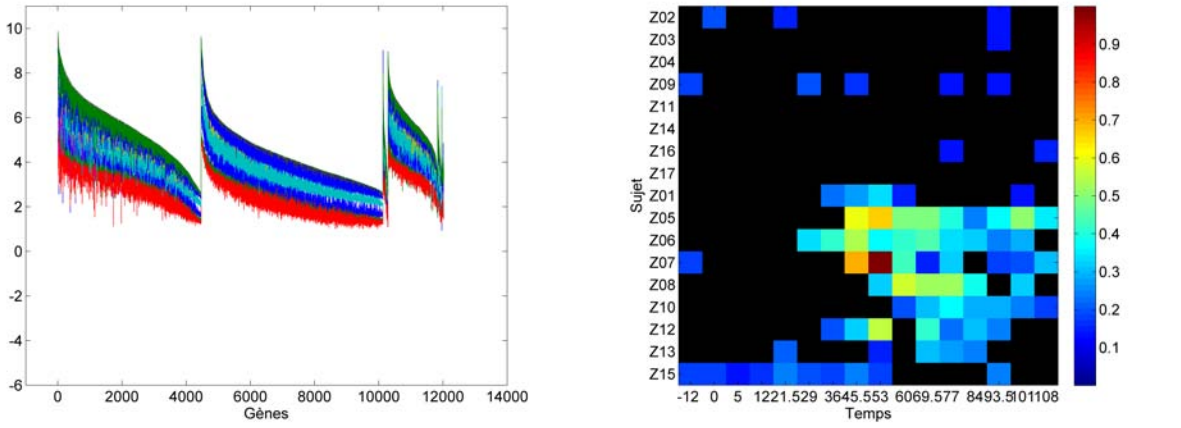
Nom du pathway	Gènes	p-valeur
Signaling events mediated by TCPTP	CSF1R, EIF2AK2, STAT1, STAT5A, STAT5B, STAT6	$4{,}20.10^{-6}$
IL23-mediated signaling events	CCL2, CXCL1, CXCL9, STAT1, STAT5A	$3{,}30.10^{-5}$
IL12-mediated signaling events	HLA-A, HLA-DRB1, SOCS1, STAT1, STAT5A, STAT6	$5{,}49.10^{-5}$
IFN-gamma pathway	CASP1, IRF1, IRF9, SOCS1, STAT1	$6{,}89.10^{-5}$
GMCSF-mediated signaling events	CCL2, STAT1, STAT5A, STAT5B	$4{,}59.10^{-4}$
CXCR3-mediated signaling events	CXCL10, CXCL11, CXCL13, CXCL9	$9{,}78.10^{-4}$
Glucocorticoid receptor regulatory network	CDKN1A, IRF1, STAT1, STAT5A, STAT5B	$1{,}41.10^{-3}$
IL2-mediated signaling events	SOCS1, STAT1, STAT5A, STAT5B	$2{,}21.10^{-3}$

L'algorithme BeG proposé est comparé avec la méthode non-paramétrique bayésienne (NPBFA) proposée par Chen *et al.* [CCP+10]. Les figures 3.9a et 3.9b représentent respectivement les signatures biologiques et les scores du facteur inflammatoire, déterminés par l'algorithme NPBFA. Enfin, la table 3.3 compare les différents critères énoncés précédemment (notamment le nombre de gènes inflammatoires, l'erreur de reconstruction, le critère de Fisher, et les p-valeurs des pathways IFN-γ et IL23). D'après ces résultats, on remarque que la composante inflammatoire déterminée par la méthode non-paramétrique contient plus de gènes, mais que cela ne permet pas de retrouver plus de gènes caractéristiques de réponse inflammatoire à une infection. La discrimination entre individus sains et malades se fait également moins bien avec la méthode NPBFA qu'avec la méthode proposée.

Sur cet ensemble de données réelles, l'algorithme BeG permet donc, comme l'algorithme uBLU, de discriminer les individus sains des individus malades et de mettre en avant un groupement de gènes codant des protéines faisant partie de la réponse inflammatoire à un agent infectieux.

TABLE 3.3 – Résultats obtenus avec les algorithmes BeG et NPBFA sur les données grippales H3N2.

	BeG	NPBFA
Nombre de gènes inflammatoires	1 668	4 467
Loadings maximum	38,36 (21,69)	9,89 (9,63)
Erreur de reconstruction (2.24)	$\mathbf{7{,}55.10^{-2}}$	13,07
Critère de Fisher (2.25)	$\mathbf{2{,}12.10^{-2}}$	$1{,}58.10^{-2}$
Gènes inflammatoires de [HZR+11]	**93,18%**	29,55%
p-valeur du pathway IFN-gamma	$6{,}89.10^{-5}$	$\mathbf{8{,}53.10^{-8}}$
p-valeur du pathway IL23	$\mathbf{3{,}30.10^{-5}}$	$1{,}45.10^{-3}$
Temps de calcul	$\approx 3{,}5\ j$	$\approx 19{,}5\ h$
Nombre d'itérations	20 000	20 000



(a) Facteurs estimés, rangés par dominance décroissante (b) Reconstruction des scores du facteur inflammatoire

FIGURE 3.9 – Comparaison sur données H3N2 [ZCV+09] avec la méthode NPBFA [CCP+10].

3.5.3 Données de grippes H1N1

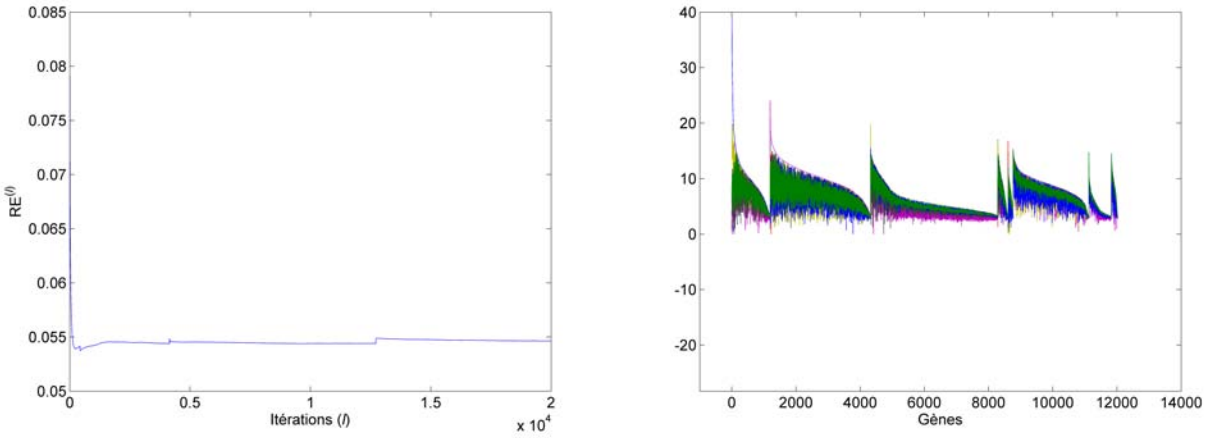
Pour finir, nous appliquons l'algorithme BeG proposé aux données réelles H1N1 détaillées dans le paragraphe 1.5.3, avec les mêmes nombres d'itérations de Monte Carlo et de chauffage que pour les données H3N2, et en fixant $R_{\max} = 9$. Les résultats obtenus sont présentés dans les figures 3.10 et les tableaux 3.4 et 3.5.

Appliqué aux données H1N1, l'algorithme BeG permet d'extraire une composante inflammatoire plus sélective que celle déterminée par la méthode uBLU (cf. figure 3.10b et nombre de gènes inflammatoires du tableau 3.4 : 1 193 gènes contre 5 538 avec l'algorithme uBLU), mais tout aussi efficace en terme de discrimination des individus sains et malades (cf. figure 3.10c et critère de Fisher dans la table 3.4) et représentative de gènes inflammatoires (cf. liste des pathways, table 3.5).

En revanche, sur ces données grippales, la méthode non-paramétrique NPBFA de Chen *et al.* [CCP+10] ne semble pas être en mesure de regrouper convenablement les gènes inflammatoires en une seule composante. Les résultats relatifs à l'application de cette méthode sont présentés dans la figure 3.11 et le tableau 3.4, et confirment cette observation.

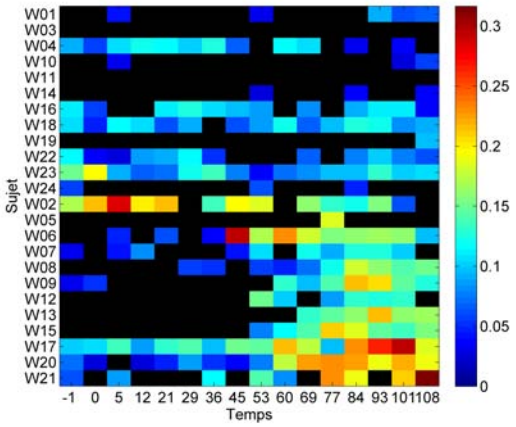
TABLE 3.4 – Résultats obtenus avec les algorithmes BeG et NPBFA sur les données grippales H1N1.

	BeG	NPBFA
Nombre de gènes inflammatoires	1 193	8 329
Loadings maximum	39,25 (24,03)	9,55 (9,16)
Erreur de reconstruction (2.24)	$\mathbf{5{,}46.10^{-2}}$	11,24
Critère de Fisher (2.25)	$\mathbf{1{,}04.10^{-2}}$	$4{,}6.10^{-3}$
Gènes inflammatoires de [HZR+11]	**100,00%**	93,18%
p-valeur du pathway IFN-gamma	$\mathbf{5{,}17.10^{-5}}$	$4{,}20.10^{-3}$
p-valeur du pathway IL23	$\mathbf{2{,}47.10^{-5}}$	$3{,}34.10^{-1}$
Temps de calcul	$\approx 4{,}5\ j$	$\approx 21{,}2\ h$
Nombre d'itérations	20 000	20 000



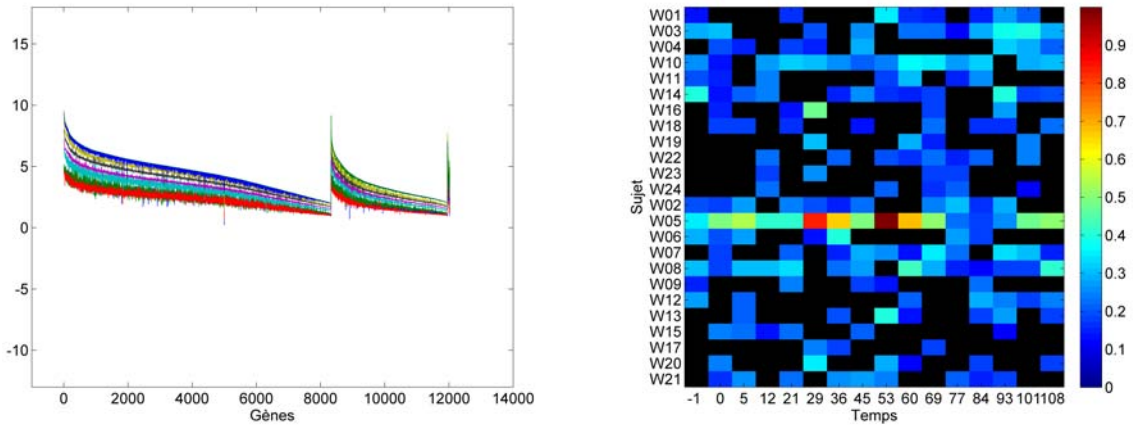
(a) Diagnostic de convergence : erreur de reconstruction (b) Facteurs estimés, rangés par dominance décroissante

$\mathrm{RE}^{(\ell)}$ en fonction du nombre d'itérations ℓ



(c) Reconstruction des scores du facteur inflammatoire

FIGURE 3.10 – Résultats de simulation de l'algorithme BeG sur données H1N1.



(a) Facteurs estimés, rangés par dominance décroissante (b) Reconstruction des scores du facteur inflammatoire

FIGURE 3.11 – Comparaison sur données H1N1 avec la méthode NPBFA [CCP+10].

TABLE 3.5 – Classement des pathways des gènes de la composante inflammatoire de BeG sur les données H1N1.

Nom du pathway	Gènes	p-valeur
Signaling events mediated by TCPTP	EIF2AK2, KPNB1, STAT1, STAT5A, STAT5B, STAT6	$2{,}95.10^{-6}$
IL12-mediated signaling events	GADD45B, HLA-A, HLA-DRB1, SOCS1, STAT1, STAT5A, STAT6	$3{,}01.10^{-6}$
IL23-mediated signaling events	CCL2, CXCL1, CXCL9, STAT1, STAT5A	$2{,}47.10^{-5}$
IFN-gamma pathway	CASP1, IRF1, IRF9, SOCS1, STAT1	$5{,}17.10^{-5}$
GMCSF-mediated signaling events	CCL2, STAT1, STAT5A, STAT5B	$3{,}66.10^{-4}$
CXCR3-mediated signaling events	CXCL10, CXCL11, CXCL13, CXCL9	$7{,}85.10^{-4}$
Glucocorticoid receptor regulatory network	CDKN1A, IRF1, STAT1, STAT5A, STAT5B	$1{,}09.10^{-3}$
IL2-mediated signaling events	SOCS1, STAT1, STAT5A, STAT5B	$1{,}78.10^{-3}$

3.6 Conclusion

Ce chapitre a présenté une approche parcimonieuse pour le démélange de données d'expressions des gènes sous contraintes physiques de positivité et de somme-à-un. Pour cela, une loi Bernoulli-gaussienne tronquée a été considérée comme loi *a priori* pour les scores afin de respecter à la fois les contraintes liées à la physique du modèle, mais aussi la contrainte supplémentaire de parcimonie. Un échantillonneur de Gibbs est proposé pour générer des échantillons distribués asymptotiquement suivant la loi *a posteriori* d'intérêt. Les échantillons ainsi générés sont ensuite utilisés pour estimer les paramètres inconnus : matrices des facteurs et des scores notamment.

Les résultats obtenus sur données synthétiques ont permis d'évaluer les performances de l'algorithme proposé en comparaison avec l'algorithme uBLU présenté au chapitre précédent. L'algorithme retrouve bien les facteurs présents dans le mélange de chaque échantillon et donne la proportion de chacun de ces facteurs. De plus, il force les scores des facteurs non présents à être nuls et permet ainsi de déterminer le nombre de facteurs présents pour chaque échantillon. Dans le cas non-supervisé, les performances de reconstruction sont légèrement moins bonnes que dans le cas où la bibliothèque de facteurs est fixée du fait de l'estimation de facteurs potentiellement absents du mélange.

Sur données réelles de grippes, l'algorithme BeG extrait une composante inflammatoire plus sélective que l'algorithme uBLU en terme de nombre de gènes inflammatoires mais tout aussi efficace quant au regroupement de gènes connus pour une action dans les processus biologiques de défense de l'organisme à un agent infectieux. La méthode BeG proposée permet également de discriminer les sujets sains des sujets malades, avec des performances similaires à la méthode uBLU.

Contributions du chapitre

La principale contribution de ce chapitre est la considération d'une approche parcimonieuse pour l'estimation du nombre de facteurs dans le mélange (parcimonie des scores), tout en travaillant sur le modèle sous contraintes. En effet, à notre connaissance, aucune méthodes ne permet d'estimer les scores, les facteurs et leur nombre sous des contraintes de positivité et de somme-à-un, tout en garantissant une solution parcimonieuse.

L'approche parcimonieuse proposée a également l'avantage de regrouper les gènes inflammatoires en un seul facteur et ce de manière plus sélective que l'approche non-paramétrique testée. Elle impose la mise à zéro des scores de facteurs non présents dans le mélange. Ceci est vrai plus particulièrement pour le facteur inflammatoire car la parcimonie rend la discrimination entre individus sains et individus malades plus évidente (les scores associés au facteur inflammatoire des sujets sains sont nuls tandis que ceux des sujets malades sont non-nuls).

Chapitre 4

Prise en compte de la dépendance temporelle : modèle de Markov caché

Sommaire

4.1 Introduction

Les résultats précédents obtenus sur des données d'expressions des gènes temporelles (données H3N2 et H1N1) et présentés dans les paragraphes 2.6.2 et 2.6.5 ont montré que les réponses moléculaires à un agent infectieux (virus grippaux dans notre cas) peuvent être classées en $K = 4$ états notés $\mathcal{E}_1, \ldots, \mathcal{E}_K$ et définis comme suit :

1. avant l'inoculation (état $\mathcal{E}_1$),
2. asymptomatique : après l'inoculation mais sans déclaration de symptômes (état $\mathcal{E}_2$),
3. pré-symptomatique : après inoculation, mais avant que de réels symptômes (fatigue, fièvre, toux, ...) apparaissent (état $\mathcal{E}_3$),
4. post-symptomatique : après déclaration des symptômes (état $\mathcal{E}_4$).

Ce dernier chapitre propose de prendre en compte cette dépendance temporelle entre échantillons d'un même individu. Pour cela, nous combinons le modèle de mélange linéaire proposé précédemment dans

le chapitre 2 à un modèle de Markov caché (HMM pour *hidden Markov model*). En effet, les modèles de Markov cachés sont des outils populaires et souvent utilisés pour l'analyse de données temporelles. Ils ont notamment été appliqués à la reconnaissance de la parole [Rab89] et de l'écriture manuscrite [KHB88], à l'analyse de séquences biologiques [DEKM98] et plus récemment à l'analyse de données d'expression des gènes dans d'autres contextes [SSS03, HWQZ11]. L'algorithme ainsi proposé permet donc de faire une classification des échantillons en plus de faire le démélange des données.

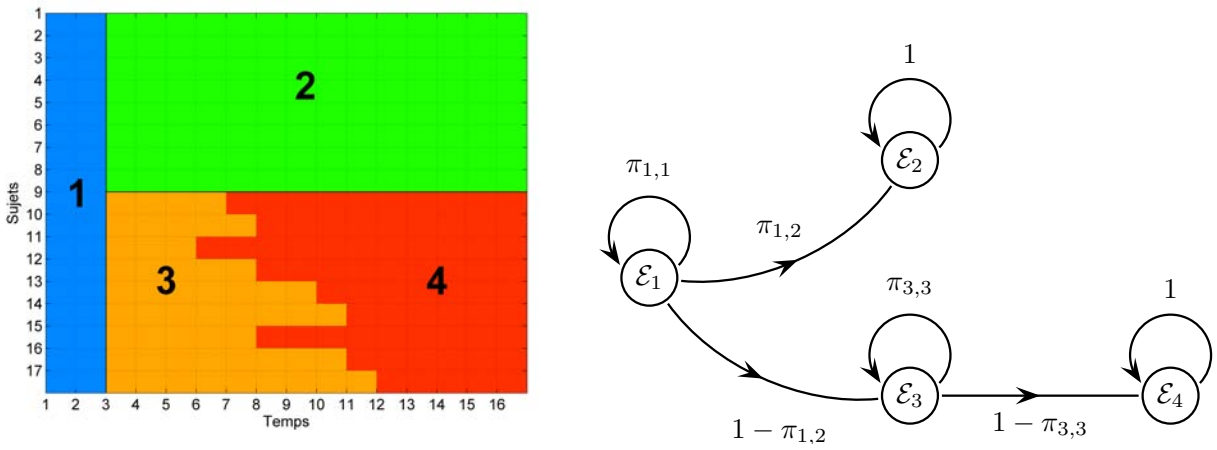
Organisation du chapitre

Ce chapitre est organisé comme suit. Le paragraphe 4.2 décrit le modèle de Markov caché introduit pour la prise en compte des dépendances temporelles entre échantillons. Le modèle bayésien étudié est présenté dans le paragraphe 4.3. L'échantillonneur de Gibbs utilisé pour générer des échantillons des paramètres inconnus selon la distribution *a posteriori* est décrit dans le paragraphe 4.4. Enfin, l'algorithme proposé est évalué sur données synthétiques (paragraphe 4.5) et appliqué sur les données réelles grippales H3N2 et H1N1 (paragraphe 4.6).

4.2 Définition du modèle de Markov caché utilisé

Rappelons que la matrice des observations $\mathbf{Y}$ est composée de N colonnes correspondant aux N échantillons collectés sur S sujets à T instants, de telle manière que $N = ST$ (voir la figure 1.3). Pour identifier l'état d'un individu donné s à un instant donné t, nous introduisons une variable discrète latente $z_{s,t}$ qui prend ses valeurs dans l'ensemble fini $\{1, \ldots, K\}$. Par conséquent, $z_{s,t} = k$ si et seulement si le $t^{\text{ème}}$ échantillon du $s^{\text{ème}}$ sujet est dans le $k^{\text{ème}}$ état $\mathcal{E}_k$. Notons $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_S]^T \in \mathbb{R}^{S \times T}$ la matrice des étiquettes renseignant sur l'état des échantillons, pour chaque sujet ($s = 1, \ldots, S$) à chaque instant ($t = 1, \ldots, T$). Le vecteur $\mathbf{z}_s = [z_{s,1}, \ldots, z_{s,T}]$ est le vecteur d'étiquettes des états du $s^{\text{ème}}$ sujet. Une vue schématique de ce processus de classification est représentée sur la figure 4.1a, où l'état d'un individu donné au cours du temps (respectivement à un instant donné sur les individus) apparaît dans les lignes (resp. les colonnes) de cette matrice de classification. Cette matrice des étiquettes a été obtenues par seuillage des valeurs des scores du facteur inflammatoire déterminé par l'algorithme uBLU sur les données H3N2.

Pour exploiter l'évolution temporelle des réponses moléculaires à un agent infectieux (grippe par exemple), ces K états sont modélisés en utilisant un modèle de Markov caché (HMM) affecté aux variables latentes de la matrice $\mathbf{Z}$ (voir [Rab89] pour plus de détails sur les HMMs). Le modèle à K états proposé est représenté sur la figure 4.1b. Il est associé à la structure temporelle représentée sur la figure 4.1a. Cette structure HMM est également en accord avec la modélisation épidémiologique SIR pour "susceptible – infectious – recovered" [CWB09].



(a) Matrice des étiquettes (b) Modèle de Markov caché

FIGURE 4.1 – Prise en compte de la dépendance temporelle par un modèle de Markov caché à $K = 4$ états $(\mathcal{E}_1, \ldots, \mathcal{E}_4)$.

À partir de ce graphe orienté étiqueté et en supposant que les probabilités de transitions d'un état à un autre sont indépendantes du sujet considéré, la matrice des probabilités de transitions des états $\mathbf{\Pi}$ peut s'écrire de la manière suivante :

$$\mathbf{\Pi} = \begin{bmatrix} \pi_{1,1} & \pi_{1,2} & \pi_{1,3} & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \pi_{3,3} & \pi_{3,4} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

avec comme vecteur des probabilités des états initiaux le vecteur $\boldsymbol{\pi}^{(\mathbf{0})}$ tel que :

$$\boldsymbol{\pi}^{(\mathbf{0})} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \tag{4.2}$$

et où $\pi_{k,k'} = \mathrm{P}[z_{s,t} = k' \,|\, z_{s,t-1} = k]$ pour $1 \leq k, k' \leq K$ et $t = 2, \ldots, T$. Remarquons que $\pi_{1,3} = 1 - \pi_{1,1} - \pi_{1,2}$ et $\pi_{3,4} = 1 - \pi_{3,3}$, c'est-à-dire que la somme des probabilités des transitions partant d'un état est égale à 1 : $\forall k$, $\sum_{k'} \pi_{k,k'} = 1$. Notons également que les probabilités de transition $\pi_{k,k'}$ peuvent servir à l'interprétation clinique. Par exemple, $\pi_{1,3}$ correspond au taux de contamination des individus, $\pi_{3,4}$ au taux de rétablissement. Enfin, la probabilité $\pi_{1,1}$ est considérée comme inconnue ici et donc estimée, mais elle pourrait être fixée puisqu'elle dépend du temps d'inoculation qui est connu.

4.3 Modèle temporel de démélange

Le modèle bayésien utilisé est basé sur la vraisemblance des observations (2.2) et sur la définition de lois *a priori* adéquates pour les paramètres inconnus $\boldsymbol{\Theta} = \{\mathbf{M}, \mathbf{A}, \mathbf{Z}, \sigma^2\}$ associés à ce modèle temporel de démélange, noté tBLU.

4.3.1 Lois *a priori* des paramètres et hyperparamètres

Les lois *a priori* des signatures $\mathbf{M}$, de la variance du bruit σ^2 détaillées dans la section 2.2.2 sont conservées dans ce chapitre. Nous nous intéresserons donc plus particulièrement aux lois *a priori* choisies pour la matrice des scores $\mathbf{A}$, la matrice des étiquettes $\mathbf{Z}$ et pour les hyperparamètres associés.

Loi *a priori* des scores

Comme cela a été montré dans [HZR+11], les réponses moléculaires des sujets asymptomatiques et symptomatiques diffèrent principalement dans les niveaux d'expression des facteurs, i.e., des scores. Par conséquent, les lois *a priori* des vecteurs des scores $\{\mathbf{a}_i\}_{i=1,\ldots,N}$ sont supposés être distinctes pour les scores associés à différents états $\mathcal{E}_1, \ldots, \mathcal{E}_K$. En outre, pour promouvoir l'interprétabilité des résultats, les scores $\{\mathbf{a}_i\}_{i=1,\ldots,N}$ doivent satisfaire les contraintes de non-négativité et de somme-à-un définies précédemment. Ainsi, une distribution de Dirichlet est choisie comme loi *a priori* pour les scores $\mathbf{a}_i$ $(i = 1, \ldots, N)$ conditionnellement à l'étiquette k attribuée au $i^{\text{ème}}$ échantillon [1] $\mathbf{y}_i$:

$$\mathbf{a}_i | z_i = k, \boldsymbol{\delta}_k \sim \mathcal{D}_R(\boldsymbol{\delta}_k) \tag{4.3}$$

1. Remarquons que par souci de clarté, les variables latentes z_i sont ici indexées par un seul indice. Il est évident qu'il y a une relation directe entre l'indice i et le couple d'indices (s, t) introduit dans le paragraphe 4.2.

où $\mathcal{D}_R(\boldsymbol{\delta}_k)$ est la loi de Dirichlet de paramètres $\boldsymbol{\delta}_k = [\delta_{1,k}, \ldots, \delta_{R,k}]^T$. En supposant que tous les vecteurs des scores $\{\mathbf{a}_i\}_{i=1,\ldots,N}$ sont *a priori* indépendants, la loi jointe *a priori* pour la matrice des scores $\mathbf{A}$ est :

$$f(\mathbf{A}|\mathbf{Z}, \Delta) = \prod_{k=1}^{K} \prod_{i \in \mathcal{C}_k} f(\mathbf{a}_i | z_i = k, \boldsymbol{\delta}_k)$$

avec $\Delta = [\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_K] \in \mathbb{R}^{R \times K}$. $\mathcal{C}_k = \{i = 1, \ldots, N | z_i = k\}$ représente le sous-ensemble des indices des échantillons dans le $k^{\text{ème}}$ état $\mathcal{E}_k$.

Loi *a priori* des étiquettes

Les probabilités *a priori* des variables latentes z_i $(i = 1, \ldots, N)$ sont données par la matrice des probabilités des états initiaux $\boldsymbol{\pi}^{(\mathbf{0})}$ et la matrice des probabilités des transitions $\boldsymbol{\Pi}$ définies précédemment dans (4.2) et (4.1). Certaines probabilités ($\pi_{1,1}$, $\pi_{1,2}$ et $\pi_{3,3}$) sont inconnues et seront estimées en utilisant un algorithme bayésien hiérarchique.

Sous l'hypothèse que tous les paramètres sont *a priori* indépendants entre eux, la loi jointe *a posteriori* du vecteur des paramètres inconnus $\boldsymbol{\Theta} = \{\mathbf{Z}, \mathbf{T}, \mathbf{A}, \sigma^2\}$ s'écrit :

$$f(\boldsymbol{\Theta}|\Delta, \gamma) = \mathrm{P}[\mathbf{Z}] f(\mathbf{T}) f(\mathbf{A}|\Delta) f\left(\sigma^2|\nu, \gamma\right). \tag{4.4}$$

Lois *a priori* des hyperparamètres

En raison de l'absence d'information *a priori* pour les hyperparamètres, nous avons choisi des lois non-informatives comme lois *a priori*. Plus précisément, une loi impropre uniforme sur $\mathbb{R}^+$ est choisie comme loi *a priori* pour les hyperparamètres Δ des scores :

$$f(\Delta) \propto \mathbf{1}_{\mathbb{R}_+^{RK}}(\Delta). \tag{4.5}$$

Notons $\boldsymbol{\pi}_1 = [\pi_{1,1}, \pi_{1,2}, \pi_{1,3}]$ et $\boldsymbol{\pi}_3 = [\pi_{3,3}, \pi_{3,4}]$ les sous-vecteurs des probabilités de transitions inconnues de la matrice $\boldsymbol{\Pi}$. En suivant l'approche de [DTS06], une loi de Dirichlet de paramètres $\boldsymbol{\alpha_i}$ $(i = 1, 3)$ est choisie comme loi *a priori* pour chaque sous-vecteur $\boldsymbol{\pi}_i$, i.e. :

$$\boldsymbol{\pi}_1|\boldsymbol{\alpha_1} \sim \mathcal{D}_3(\boldsymbol{\alpha_1}), \quad \boldsymbol{\pi}_3|\boldsymbol{\alpha_3} \sim \mathcal{D}_2(\boldsymbol{\alpha_3}). \tag{4.6}$$

Nous fixerons toutes les valeurs des paramètres des lois de Dirichlet $\{\boldsymbol{\alpha_i}\}_{i=1,3}$ égaux à 1. Ces lois reflètent bien le manque de connaissance sur ces hyperparamètres.

Sous l'hypothèse que tous les hyperparamètres de ce modèle bayésien hiérarchique sont *a priori* indépendants entre eux, la loi jointe *a posteriori* du vecteur des hyperparamètres $\boldsymbol{\Psi} = \{\Delta, \boldsymbol{\Pi}, \gamma\}$ s'écrit alors :

$$f(\boldsymbol{\Psi}) = f(\Delta)\, f(\boldsymbol{\Pi})\, f(\gamma)\,. \tag{4.7}$$

4.3.2 Résumé des lois *a priori* du modèle tBLU

Ce paragraphe permet de rappeler l'ensemble des lois *a priori* choisies pour les paramètres inconnus du modèle tBLU. Il présente également dans la figure 4.2 la structure hiérarchique du modèle sous forme de DAG.

$$\begin{array}{rcll}
\mathbf{m}_r & = & \mathbf{U}\mathbf{t}_r + \bar{\mathbf{y}} & (2.4) \\
\mathbf{t}_r|\mathbf{e}_r, s_r^2 & \sim & \mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2\mathbf{I}_{R-1}\right) & (2.5) \\
\mathbf{a}_i|z_i = k, \boldsymbol{\delta}_k & \sim & \mathcal{D}_R\left(\boldsymbol{\delta}_k\right) & (4.3) \\
\mathrm{P}[z_{s,t} = k' \,|\, z_{s,t-1} = k] & = & \pi_{k,k'} & (4.1), (4.2) \\
\Delta = [\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_K] & \sim & \mathcal{U}_{\mathbb{R}_+^{RK}}(\Delta) & (4.5) \\
\boldsymbol{\pi}_1|\boldsymbol{\alpha_1} & \sim & \mathcal{D}_3(\boldsymbol{\alpha_1}) & (4.6) \\
\boldsymbol{\pi}_3|\boldsymbol{\alpha_3} & \sim & \mathcal{D}_2(\boldsymbol{\alpha_3}) & \\
\sigma^2|\nu, \gamma & \sim & \mathcal{IG}\left(\frac{\nu}{2}, \frac{\gamma}{2}\right) & (2.12)
\end{array}$$

4.3.3 Loi *a posteriori*

La loi jointe *a posteriori* des vecteurs des paramètres $\boldsymbol{\Theta} = \{\mathbf{T}, \mathbf{A}, \mathbf{Z}, \sigma^2\}$ et hyperparamètres $\boldsymbol{\Psi} = \{\Delta, \boldsymbol{\Pi}, \gamma\}$ est définie de la manière suivante :

$$f(\boldsymbol{\Theta}, \boldsymbol{\Psi}|\mathbf{Y}) \propto f(\mathbf{Y}|\boldsymbol{\Theta})\, f(\boldsymbol{\Theta}|\boldsymbol{\Psi})\, f(\boldsymbol{\Psi}) \tag{4.8}$$

où $f(\mathbf{Y}|\boldsymbol{\Theta})$, $f(\boldsymbol{\Theta}|\boldsymbol{\Psi})$ et $f(\boldsymbol{\Psi})$ ont respectivement été définies dans (2.2), (4.4) et (4.7). Les contraintes imposées sur les données ainsi que la structure temporelle du modèle HMM rendent cette loi jointe *a*

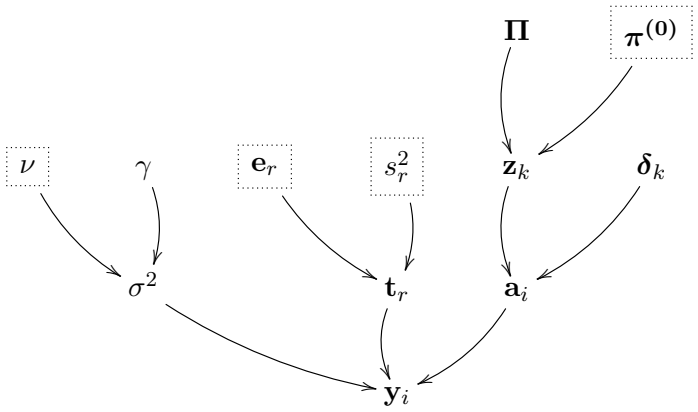


FIGURE 4.2 – Graphe acyclique orienté (DAG) pour les lois *a priori* des paramètres et hyperparamètres, dans le cas du modèle HMM temporel. (Les paramètres fixés apparaissent dans les cases en pointillés.)

posteriori $f(\boldsymbol{\Theta}, \boldsymbol{\Psi}|\mathbf{Y})$, définie dans (4.8), beaucoup trop complexe pour pouvoir obtenir des expressions analytiques des estimateurs bayésiens du vecteur des paramètres inconnus $\boldsymbol{\Theta}$. Pour pallier ce problème, nous faisons encore appel aux méthodes de Monte Carlo par chaînes de Markov (MCMC) [GRS96] pour générer des échantillons distribués selon (4.8) et calculer les estimateurs bayésiens, MMSE ou MAP, à partir de ces échantillons générés.

4.4 Echantillonneur de Metropolis-within-Gibbs

Cette section présente l'algorithme de Metropolis-within-Gibbs utilisé pour générer aléatoirement des échantillons asymptotiquement distribués suivant la loi *a posteriori* d'intérêt, définie dans (4.8). Le principe d'un échantillonneur de Metropolis-within-Gibbs est d'utiliser un mouvement de Metropolis pour chaque loi conditionnelle qui ne peut pas être échantillonnée directement. Plus précisément, les différentes étapes de l'algorithme proposé sont détaillées dans l'algorithme 4.1.

ALGO. 4.1 – Echantillonneur de Metropolis-within-Gibbs pour le modèle temporel tBLU.

- Pré-traitements :
  - Calculer la moyenne empirique des échantillons $\bar{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{y}_i$ définie dans (2.4),
  - Déterminer la matrice de projection $\mathbf{P}$ (de taille $R_{\max}-1 \times G$) à l'aide d'une technique de réduction de dimensionnalité (comme l'ACP [Jol86]),
  - Choisir les vecteurs moyennes $\{\mathbf{e}_r\}_{r=1,\ldots,R_{\max}}$ définis dans (2.5) comme les solutions d'un algorithme d'extraction de pôles de mélange (comme l'algorithme VCA [NBD05]).
- Initialisation ($\ell = 0$) :
  - Echantillonner la matrice des signatures projetées $\mathbf{T}^{(0)}$ selon la loi *a priori* (2.8),
  - Reconstruire la matrice des signatures $\mathbf{M}^{(0)}$ à partir de la matrice des signatures projetées $\mathbf{T}^{(0)}$: $\mathbf{M}^{(0)} = \mathbf{P}^{-1}\mathbf{T}^{(0)} + \bar{\mathbf{y}}\mathbf{1}_{R^{(0)}}^T$ (2.4),
  - Pour $i = 1, \ldots, N$,
    - Générer les scores $\mathbf{a}_i^{(0)}$ suivant (4.3),
    - Générer les étiquettes $\mathbf{z}_i^{(0)}$ en utilisant les matrices de transitions (4.1) et (4.2),
  - Générer la variance du bruit $\sigma^{2(0)}$ suivant la loi définie dans (2.12),
  - Poser $\ell \leftarrow 1$.
- Itérations : Pour $\ell = 1, 2, \ldots, \mathrm{N_{mc}}$, :
  1. Echantillonner les signatures projetées $\mathbf{T}^{(\ell)}$ selon la loi (2.16),
  2. Reconstruire les signatures $\mathbf{M}^{(\ell)} = \mathbf{P}^{-1}\mathbf{T}^{(\ell)} + \bar{\mathbf{y}}\mathbf{1}_{R^{(\ell)}}^T$ (2.4),
  3. Pour chaque échantillon $i = 1, \ldots, N$,
     - Echantillonner les scores $\mathbf{a}_i^{(\ell)}$ selon la loi (4.9),
     - Echantillonner les étiquettes $z_i^{(\ell)}$ selon la loi (4.10),
  4. Echantillonner la variance du bruit $\sigma^{2(\ell)}$ selon la loi (2.18),
  5. Pour $r = 1, \ldots, R$ et $k = 1, \ldots, K$, échantillonner les hyperparamètres $\delta_{r,k}^{(\ell)}$ selon (4.11),
  6. Echantillonner les probabilités de transitions $\boldsymbol{\pi}_i^{(\ell)}$ $(i = 1, 3)$ selon (4.12),
  7. Poser $\ell \leftarrow \ell + 1$.

4.4.1 Echantillonnage suivant $f\left(\mathbf{a}_i|\mathbf{M}, z_i = k, \sigma^2, \boldsymbol{\delta}_k, \mathbf{y}_i\right)$

La loi conditionnelle *a posteriori* du vecteur des scores du $i$ème échantillon $\mathbf{a}_i$ $(i = 1, \ldots, N)$ est :

$$f\left(\mathbf{a}_i|\mathbf{M}, z_i = k, \sigma^2, \boldsymbol{\delta}_k, \mathbf{y}_i\right) \propto \prod_{r=1}^{R} a_{r,i}^{\delta_{r,k}-1} \times \exp\left(-\frac{\|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\|^2}{2\sigma^2}\right) \mathbf{1}_{\mathcal{A}}\left(\mathbf{a}_i\right). \tag{4.9}$$

Malheureusement, il n'est pas facile de générer des échantillons aléatoires distribués suivant (4.9). Pour remédier à cette difficulté, nous proposons d'utiliser une étape de Metropolis-Hastings. La loi de proposition pour les $R-1$ premiers éléments du vecteur des scores, $\mathbf{a}_{1:R-1,i} = [a_{1,i}, \ldots, a_{R-1,i}]$, est une loi gaussienne multivariée tronquée sur le simplexe $\mathcal{S}$ défini dans (2.10). Enfin, la contrainte de somme-à-un permet d'exprimer le dernier score : $a_{R,i} = 1 - \sum_{r=1}^{R-1} a_{r,i}$.

4.4.2 Echantillonnage suivant $\mathbf{P}\left[z_{s,t} = k|z_{s,t-1}, \mathbf{a}_i, \boldsymbol{\delta}_k, \boldsymbol{\pi}^{(\mathbf{0})}, \boldsymbol{\Pi}\right]$

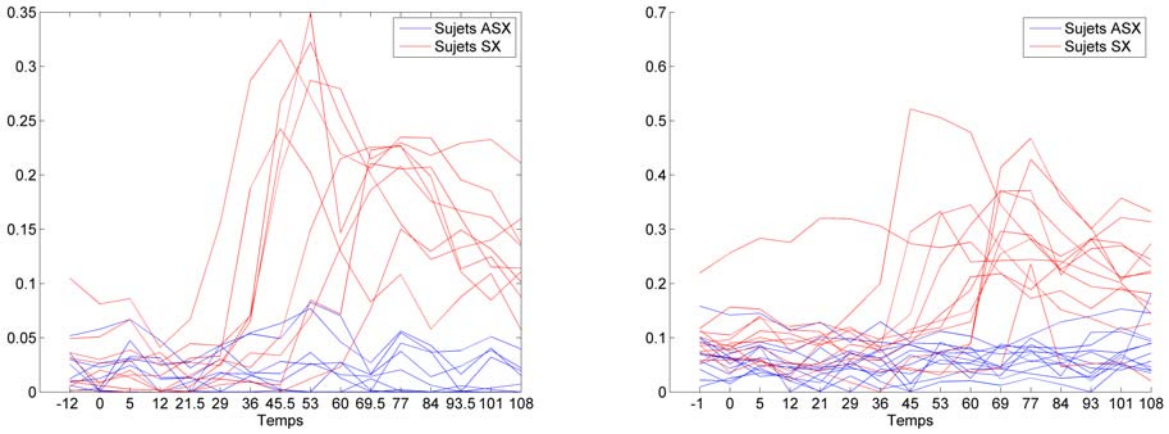
Pour chaque échantillon i $(i = 1, \ldots, N)$, correspondant plus particulièrement au $t^{\text{ème}}$ échantillon du sujet $\#s$, l'étiquette $z_{s,t}$ est une variable aléatoire discrète dont la loi conditionnelle est caractérisée par les probabilités suivantes :

$$\begin{aligned} \mathrm{P}\left[z_{s,t} = k|z_{s,t-1}, \mathbf{a}_i, \boldsymbol{\delta}_k, \boldsymbol{\pi}^{(\mathbf{0})}, \boldsymbol{\Pi}\right] &\propto \mathrm{P}[z_{s,t} = k|z_{s,t-1}] f\left(\mathbf{a}_i|z_{s,t-1} = k, \boldsymbol{\delta}_k\right) f\left(\mathbf{y}_i|\mathbf{a}_i, \sigma^2\right) \\ &\propto \mathrm{P}[z_{s,t} = k|z_{s,t-1}] \frac{\Gamma\left(\sum_{r=1}^{R} \delta_{r,k}\right)}{\prod_{r=1}^{R} \delta_{r,k}} \prod_{r=1}^{R} a_i^{\delta_{r,k}-1} \mathbf{1}_{\mathcal{A}}\left(\mathbf{a}_i\right). \end{aligned} \tag{4.10}$$

Les probabilités $\mathrm{P}[z_{s,t} = k|z_{s,t-1}]$ sont définies dans (4.1) et (4.2). Pour échantillonner suivant cette loi conditionnelle discrète, on génère des valeurs discrètes dans $\{1, \ldots, K\}$ suivant les probabilités (4.10) (après normalisation).

Malheureusement, nous devons faire face au problème de permutation d'étiquettes (ou "*label switching*" en anglais) qui peut se produire lors de l'attribution des étiquettes $\mathcal{E}_2$ et $\mathcal{E}_3$. Il s'agit d'un problème récurrent dû au manque d'identifiabilité dans le modèle de Markov proposé (voir [CMR05, p. 478] pour plus de détails sur le *label switching*). Pour résoudre ce problème, une approche courante consiste à imposer des contraintes. Étant donné que les fluctuations des scores associés à des échantillons de sujets symptomatiques devraient être supérieures à celles des sujets asymptomatiques, nous imposons que la variance des scores des sujets asymptomatiques soit inférieure à celle des sujets

symptomatiques. Cela se vérifie sur la figure 4.3 où les scores du facteur inflammatoire déterminés par l'algorithme uBLU sur les données réelles ont été représentés en fonction du temps pour chaque sujet. Les courbes des scores des sujets asymptomatiques (courbes en bleu) fluctuent beaucoup moins que celles des sujets symptomatiques (en rouge).



(a) Données H3N2 (b) Données H1N1

FIGURE 4.3 – Scores du facteur inflammatoire déterminé par l'algorithme uBLU pour chaque sujet.

4.4.3 Echantillonnage suivant $f\left(\delta_{r,k}|\mathbf{a}_r, \mathbf{Z}\right)$

Pour chaque facteur r $(r = 1, \ldots, R)$ et chaque état $\mathcal{E}_k$ $(k = 1, \ldots, K)$, les paramètres de Dirichlet $\delta_{r,k}$ sont générés selon :

$$\begin{aligned} f\left(\delta_{r,k}|\mathbf{a}_r, \mathbf{Z}\right) &\propto f\left(\delta_{r,k}\right) \prod_{i \in \mathcal{C}_k} f\left(\mathbf{a}_i | z_i = k, \boldsymbol{\delta}_k\right) \\ &\propto \prod_{i \in \mathcal{C}_k} \left[\frac{\Gamma(\sum_{r=1}^{R} \delta_{r,k})}{\Gamma(\delta_{r,k})} a_{r,i}^{\delta_{r,k}-1} \right] \mathbf{1}_{\mathbb{R}^+}\left(\delta_{r,k}\right). \end{aligned} \tag{4.11}$$

Une étape de Metropolis-Hastings est employée pour générer des échantillons distribués suivant (4.11). Plus précisément, les échantillons sont générés en utilisant une marche aléatoire avec une loi gaussienne $\mathcal{N}_{\mathbb{R}^+}\left(0, w^2\right)$ tronquée sur $\mathbb{R}^+$, de moyenne nulle et de variance w^2, comme loi de proposition. La

variance de cette loi w^2 est fixée de manière à ce que le taux d'acceptation soit compris entre 0,15 et 0,50, comme recommandé dans [Rob96, p. 55].

4.4.4 Echantillonnage suivant $f(\mathbf{\Pi}|\mathbf{Z})$

De simples calculs permettent de déterminer la loi conditionnelle des vecteurs des probabilités de transition inconnues $\boldsymbol{\pi}_i$ $(i = 1, 3)$. Il s'agit d'une loi de Dirichlet de paramètres $\boldsymbol{\alpha}_i + N_i$, où $N_1 = [n_{1,1}, n_{1,2}, n_{1,3}]$, $N_3 = [n_{3,3}, n_{3,4}]$, et $n_{i,j} = \#\{t \,|\, z_{s,t} = j \text{ et } z_{s,t-1} = i\}$:

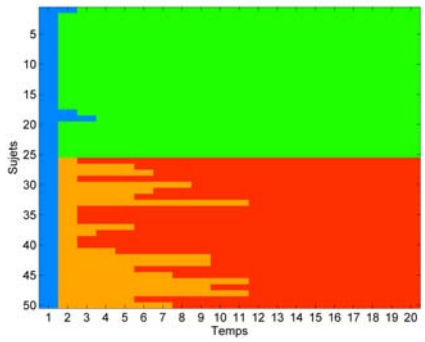
$$\boldsymbol{\pi}_i \sim \mathcal{D}(\boldsymbol{\alpha}_i + N_i) \tag{4.12}$$

Plus particulièrement, $n_{1,2}$ correspond au nombre de sujets asymptomatiques et $n_{1,3} = n_{3,4}$ est le nombre de sujets symptomatiques.

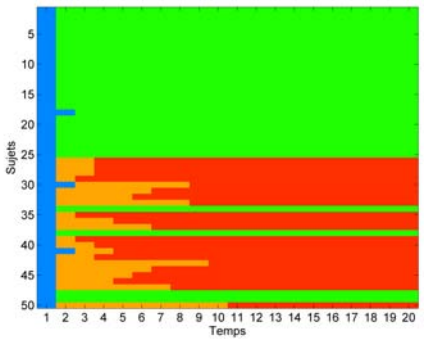
4.5 Résultats de simulation sur données synthétiques

Scénario de simulation

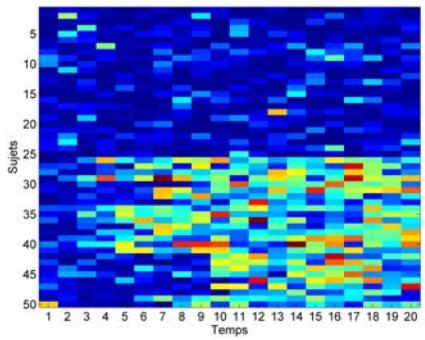
Les performances de l'algorithme proposé ont d'abord été évaluées à l'aide d'un jeu de données synthétiques, noté $\mathcal{J}_t$, de $N = 1\,000$ échantillons, plus précisément $S = 50$ sujets et $T = 20$ instants. Chaque échantillon est composé d'exactement $R = 4$ facteurs, et $G = 12\,000$ gènes. Afin de générer des signatures réalistes, les facteurs ont été extraits des résultats précédents obtenus sur données réelles temporelles avec l'algorithme uBLU, puis ils ont été mélangés selon le modèle MML (1.2). La carte des états, représentée sur la figure 4.4a, a été générée en suivant la chaîne de Markov à 4 états de la figure 4.1b avec $\boldsymbol{\pi}_1 = [0.1, 0.45, 0.45]$ et $\boldsymbol{\pi}_3 = [0.8, 0.2]$. Les paramètres de Dirichlet Δ ont été estimés à partir des résultats obtenus avec l'algorithme uBLU sur les données réelles H3N2, en utilisant la méthode décrite dans [Nar91]. Les échantillons sont bruités avec un rapport signal-à-bruit fixé à SNR = 20 dB. Les vecteurs moyennes $\mathbf{e}_r$ $(r = 1, \ldots, R)$ nécessaires pour évaluer la loi *a priori* des signatures génétiques sont choisis comme les projections des signatures identifiées par une analyse VCA [NBD05].



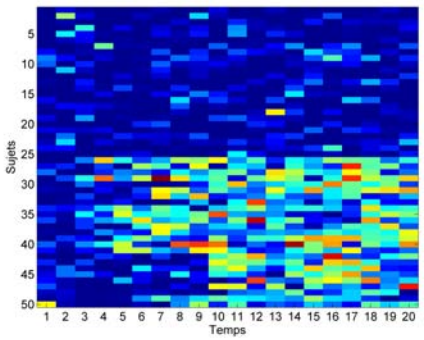
(a) Matrice des étiquettes actuelle $\mathbf{Z}$



(b) Estimation MAP marginale de la matrice des étiquettes $\widehat{\mathbf{Z}}$



(c) Scores du facteur inflammatoire $\mathbf{a}_r$



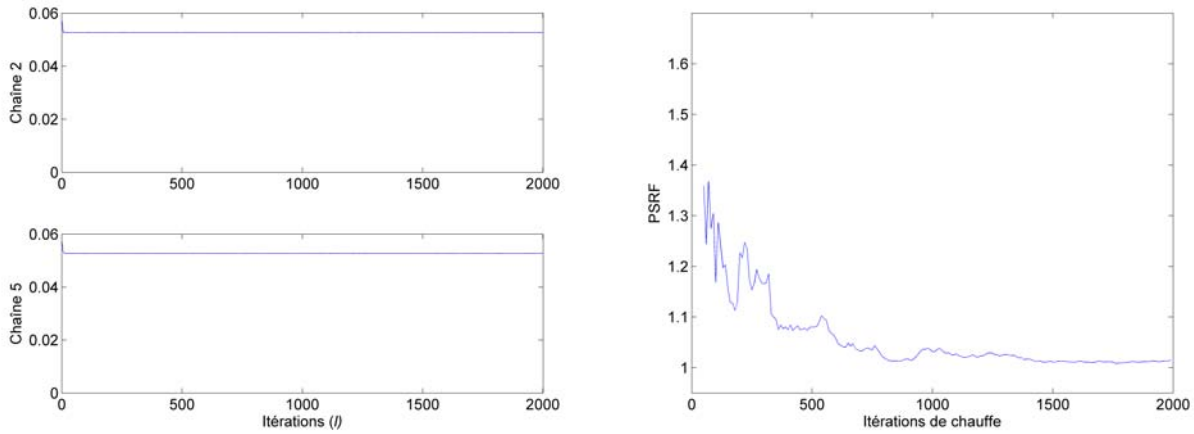
(d) Estimation MAP des scores du facteur inflammatoire $\widehat{\mathbf{a}}_r$

FIGURE 4.4 – Résultats de simulations de l'algorithme tBLU sur données synthétiques $\mathcal{J}_t$.

Le paragraphe suivant permet de déterminer des valeurs adéquates pour les nombres d'itérations de chauffage ($\mathrm{N_{bi}}$) et de Monte Carlo ($\mathrm{N_{mc}}$). Le paragraphe 4.5.2 présentera les résultats obtenus avec l'algorithme tBLU sur ce jeu de données synthétiques $\mathcal{J}_t$ et étudiera les performances de l'algorithme proposé en comparaison avec sa version uBLU non-temporelle.

4.5.1 Contrôle de la convergence

Afin de contrôler la convergence des chaînes de Monte Carlo générées par l'échantillonneur de Metropolis-within-Gibbs détaillé au paragraphe 4.4, nous avons simulé $C = 10$ chaînes de Monte Carlo en parallèle, de $\mathrm{N_{mc}} = 2\ 000$ itérations et d'initialisations différentes. Nous utilisons le critère de variance inter-intra chaîne de Gelman et Rubin [GR92] en suivant la variance du bruit σ^2. Deux des $C = 10$ chaînes MCMC sont représentées sur la figure 4.5a. Cette figure montre notamment que les deux chaînes convergent bien vers une valeur identique : $\sigma^2 \approx 5{,}27.10^{-2}$. Le potentiel d'échelle $PSRF$ est calculé selon l'équation (3.17) définie au paragraphe 3.4.3. Les valeurs obtenues pour le $PSRF$ sont représentées sur la figure 4.5b en fonction du nombre d'itérations dans la période de chauffage $\mathrm{N_{bi}}$, à $\mathrm{N_{mc}}$ fixé. Cette figure montre qu'à partir de $\mathrm{N_{bi}} = 350$ itérations, les échantillons générés sont bien distribués suivant la loi *a posteriori* d'intérêt puisqu'à partir de cette valeur les potentiels d'échelle restent inférieurs à 1,1. Pour les simulations, nous fixerons donc $\mathrm{N_{mc}} = 2\ 000$ itérations MCMC et $\mathrm{N_{bi}} = 350$ itérations de chauffe.



(a) Exemples de deux chaînes MCMC relatives au paramètre σ^2

(b) Potentiels d'échelle $PSRF$ calculés selon (3.17) en fonction du nombre d'itérations de chauffe $\mathrm{N_{bi}}$

FIGURE 4.5 – Contrôle de la convergence de l'algorithme tBLU sur données synthétiques.

4.5.2 Performances de simulation

L'algorithme a été exécuté avec les valeurs de N_{mc} et N_{bi} déterminées par l'analyse de la convergence : $N_{mc} = 2\ 000$ itérations de Monte Carlo dont $N_{bi} = 350$ itérations de burn-in. Les estimateurs MAP des paramètres inconnus ont été calculés à partir des échantillons générés par l'échantillonneur de Metropolis-within-Gibbs présenté dans la section 4.4.

L'estimateur MAP marginalisé $\widehat{\mathbf{Z}}$ de la matrice des états $\mathbf{Z}$ et les estimateurs MAP des scores correspondants au facteur inflammatoire sont respectivement représentés sur les figures 4.4b et 4.4d. Les résultats obtenus sont clairement en accord avec les valeurs réelles des figures 4.4a et 4.4c. Les estimations MMSE des probabilités de transitions inconnues sont : $\widehat{\boldsymbol{\pi}}_1 = [0.09, 0.52, 0.39]$ et $\widehat{\boldsymbol{\pi}}_3 = [0.81, 0.19]$. Deux types de classifications peuvent être considérés à partir de la matrice des états $\mathbf{Z}$:

- une classification par sujets : symptomatiques (SX) ou asymptomatiques (ASX),
- une classification par états : $\mathcal{E}_1, \ldots, \mathcal{E}_4$.

Les matrices de confusions, ou tables de contingence, représentées dans les tables 4.1 et 4.2 permettent de mesurer la qualité des deux types de classification à partir de la matrice des étiquettes estimée $\widehat{\mathbf{Z}}$ [Col01]. Les cellules sur la diagonale de cette matrice renseignent sur le nombre de sujets et d'échantillons correctement classés. La qualité de la classification est évaluée par le calcul du coefficient Kappa κ proposé par Cohen [Coh60]. Les résultats obtenus sont de 84,0% et 81,3% respectivement pour la classification par sujets ASX/SX et pour celle par états. Ces résultats attestent des bonnes performances de l'algorithme tBLU proposé.

Les résultats de simulations des MSEs, GMSEs, SADs, GSADs et erreur de reconstruction (critères définis au paragraphe 2.5.3) sont reportés dans le tableau 4.3 où le modèle temporel proposé est comparé avec le modèle linéaire Bayésien non-temporel du chapitre 2 en fixant le nombre de facteurs à $R = 4$ (modèle également détaillé dans l'article [HZR+11]). Ces résultats montrent que l'algorithme proposé a des performances identiques voire meilleures que sa version non-temporelle. De plus, il a l'avantage de fournir une classification des échantillons selon l'état du sujet à l'instant considéré, ce qui n'est pas possible avec sa version non-temporelle.

TABLE 4.1 – Matrice de confusion pour la classification des sujets : symptomatiques (SX) / asymptomatiques (ASX).

		Z réel		
		SX	ASX	Total
$\widehat{\mathbf{Z}}$ estimé	SX	**21**	0	21
	ASX	4	**25**	29
	Total	25	25	50

TABLE 4.2 – Matrice de confusion pour la classification par états : $\mathcal{E}_1, \ldots, \mathcal{E}_4$.

		Z réel				
		$\mathcal{E}_1$	$\mathcal{E}_2$	$\mathcal{E}_3$	$\mathcal{E}_4$	Total
$\widehat{\mathbf{Z}}$ estimé	$\mathcal{E}_1$	**51**	0	0	0	51
	$\mathcal{E}_2$	3	**471**	17	59	550
	$\mathcal{E}_3$	0	0	**71**	9	80
	$\mathcal{E}_4$	0	0	25	**292**	317
	Total	54	471	115	360	1000

TABLE 4.3 – Comparaison des performances d'estimation entre le modèle temporel proposé (tBLU) et sa version non-temporelle (uBLU) sur données synthétiques $\mathcal{J}_t$.

		tBLU	uBLU ($R = 4$)
MSE_r^2 ($\times 10^{-2}$)	Facteur 1	**0,25**	1,49
	Facteur 2	**103,39**	105,22
	Facteur 3	17,92	**14,41**
	Facteur 4	**93,09**	**93,09**
GMSE_r^2 ($\times 10^{-3}$)	Facteur 1	**17,47**	23,71
	Facteur 2	18,86	**16,58**
	Facteur 3	**6,68**	7,53
	Facteur 4	**5,37**	5,48
SAD ($\times 10^{-2}$)	Facteur 1	**0,62**	1,31
	Facteur 2	**12,66**	12,78
	Facteur 3	5,28	**4,81**
	Facteur 4	**12,42**	**12,42**
GSAD ($\times 10^{-3}$)		30,00	**29,40**
RE ($\times 10^{-2}$)		**5,27**	5,55
Temps de calcul		6 *h*	6 *h*

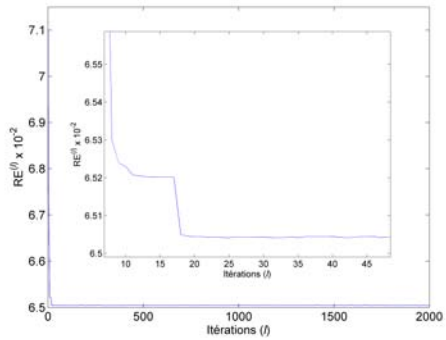
4.6 Applications sur données réelles

L'algorithme proposé dans ce chapitre a été plus particulièrement développé pour des applications de détection de symptômes de maladies sur des données d'expressions de gènes temporelles. Il est donc naturellement appliqué dans cette section aux données des grippes H3N2 (paragraphe 4.6.1) et H1N1 (paragraphe 4.6.2) et sera comparé à sa version non-temporelle présenté au chapitre 2 en fixant le nombre de facteurs.

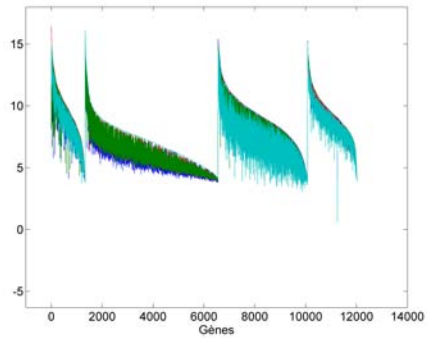
4.6.1 Données de grippes H3N2

Comme pour les autres algorithmes proposés dans ce manuscript, nous avons appliqué l'algorithme tBLU aux données de grippe H3N2 décrites au paragraphe 1.5.2, en fixant le nombre de facteurs à $R = 4$ (résultat obtenu avec l'algorithme uBLU sur ces mêmes données, cf. paragraphe 2.6.2) et avec $\mathrm{N_{mc}} = 2\,000$ itérations de Monte Carlo dont $\mathrm{N_{bi}} = 500$ itérations de chauffage. Du fait de la connaissance de l'instant d'inoculation, la probabilité $\pi_{1,1}$ a été fixée à $\pi_{1,1} = 2/T$. La convergence de l'algorithme pour ces valeurs de $\mathrm{N_{mc}}$ et $\mathrm{N_{bi}}$ est justifiée par la figure 4.6a.

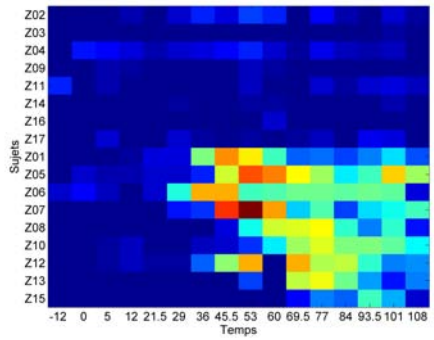
Comme dans les analyses précédentes (articles [ZCV+09, HZR+11, BDT+13] et paragraphes 2.6.2 et 3.5.2), l'algorithme temporel proposé permet d'identifier un facteur caractéristique, ou *facteur inflammatoire*, composé de 1 327 gènes, en rouge sur la figure 4.6b, et dont les scores correspondant sont représentés sur la figure 4.6c. La matrice des états estimée $\widehat{\mathbf{Z}}$ associée à cette composante inflammatoire est représentée sur la figure 4.6d. Les figures 4.6c et 4.6d montrent que l'algorithme proposé sépare clairement les sujets qui déclarent des symptômes (correspondant aux 9 dernières lignes) des sujets asymptomatiques (correspondant aux 8 premières lignes). Le sujet #15 (dernière ligne de l'image) est classé parmi les sujets asymptomatiques, ce qui semble cohérent avec les valeurs des scores de ce sujet (remarquons qu'il n'existe pas de réelle vérité terrain pour ces données). L'algorithme proposé peut également être utilisé pour estimer les probabilités de transitions inconnues : $\widehat{\boldsymbol{\pi}}_1 = [0.14, 0.45, 0.41]$ et $\widehat{\boldsymbol{\pi}}_3 = [0.84, 0.16]$ (estimations MMSE).



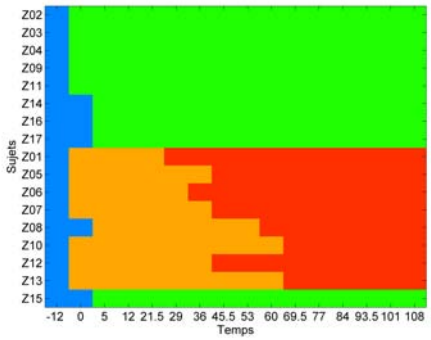
(a) Diagnostic de convergence



(b) Factors estimés, rangés par dominance décroissante



(c) Scores du facteur inflammatoire



(d) Matrice des étiquettes

FIGURE 4.6 – Résultats de simulation de l'algorithme tBLU, sur données H3N2 [ZCV+09].

La méthode temporelle tBLU proposée couplée avec un modèle HMM a été comparée avec sa version non-temporelle uBLU, présentée au chapitre 2, en fixant le nombre de facteurs à $R = 4$. Différents critères de comparaison ont été évalués pour les deux versions considérées dont le nombre de gènes contenus dans le facteur inflammatoire, les valeurs maximales des loadings des groupements de gènes (loading maximal du facteur inflammatoire et loading maximal du deuxième groupement de gènes entre parenthèses), l'erreur de reconstruction (RE) (2.24), le critère de Fisher [DHS00, p. 119] (2.25), ... Ils ont été reportés dans la table 4.4. Les résultats obtenus par l'algorithme temporel proposé sur ce jeu de données grippales sont très similaires avec ceux obtenus avec la version non-temporelle uBLU à R fixé (table 4.4) ou non (cas non-supervisé présenté au chapitre 2 et dont les résultats, présentés dans la table 2.6, sont rappelés en dernière colonne de la table 4.4) en termes d'erreur de reconstruction et concernant la mise en évidence d'un facteur de gènes inflammatoires et la distinction entre sujets malades et sujets sains (critère de Fisher). Notons que cette version temporelle est bien plus rapide que les autres algorithmes du fait que l'on fixe le nombre de facteurs R du mélange et que l'algorithme ne cherche donc pas des solutions dans des espaces différents. Le principal avantage de cette version temporelle est qu'elle permet une classification des sujets selon leur état.

TABLE 4.4 – Performances de l'algorithme tBLU sur les données réelles H3N2 et comparaison avec la version non-temporelle (uBLU avec $R = 4$ fixé).

	tBLU	uBLU ($R = 4$)	uBLU (table 2.6)
Nombre de gènes inflammatoires	1 327	1 400	2 297
Loadings maximum	16,47 (16,09)	16,47 (16,09)	30,66 (17,78)
Erreur de reconstruction (2.24)	$6{,}51.10^{-2}$	$6{,}59.10^{-2}$	$6{,}48.10^{-2}$
Critère de Fisher ($\times 10^{-2}$) (2.25)	5,30	5,21	6,20
Gènes inflammatoires de [HZR+11]	88,64%	88,64%	100%
p-valeur du pathway IFN-gamma	$7{,}76.10^{-4}$	$9{,}01.10^{-5}$	$1{,}34.10^{-9}$
p-valeur du pathway IL23	$4{,}70.10^{-2}$	$5{,}49.10^{-2}$	$2{,}18.10^{-7}$
Temps de calcul	6 035 s	6 125 s	≈ 12 h
Nombre d'itérations	2 000	2 000	10 000

4.6.2 Données de grippes H1N1

De même que précédemment, l'algorithme tBLU a été appliqué sur les données H1N1 en fixant le nombre de facteurs à $R = 4$ (résultat trouvé avec la méthode uBLU, cf. paragraphe 2.6.5) et avec $N_{mc} = 2\,000$ itérations de Monte Carlo dont $N_{bi} = 500$ itérations de chauffage. La convergence de l'algorithme pour ses valeurs de N_{mc} et N_{bi} est justifiée par la figure 4.7a.

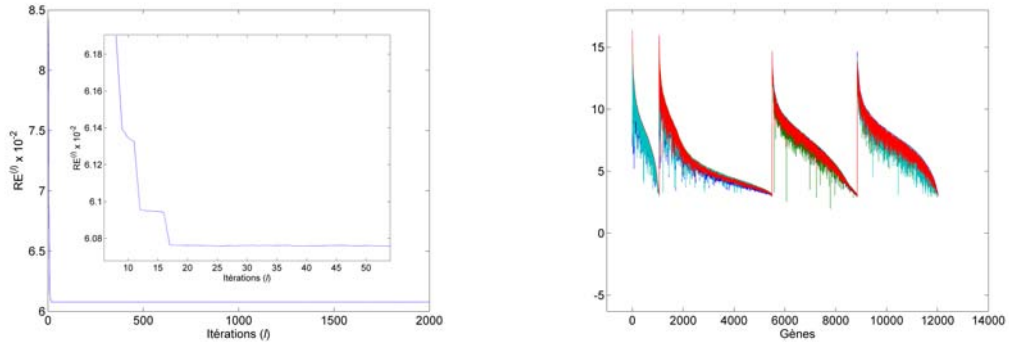
L'algorithme tBLU proposé permet ici encore d'identifier un facteur caractéristique inflammatoire (signature la plus à gauche en rouge sur la figure 4.7b et scores associés de la figure 4.7c) mais moins clairement que la version non-supervisée uBLU à R fixé (voir la figure 2.14c et les loadings maximum des deux premiers facteurs détaillés dans les tables 4.5). Il en est de même pour la distinction entre individus sains et malades : les mesures du critères de Fisher restent inférieures pour la version tBLU que pour la version uBLU à R fixé. Remarquons à ce stade que la représentation proposée avec les sujets asymptomatiques sur les lignes du haut de la figure 4.7c et les sujets symptomatiques en bas n'est pas une vérité terrain exacte.

TABLE 4.5 – Performances de l'algorithme tBLU sur les données réelles H1N1 et comparaison avec la version non-temporelle (uBLU avec $R = 4$ fixé).

	tBLU	uBLU ($R = 4$)	uBLU (table 2.8)
Nombre de gènes inflammatoires	1 054	5 027	5 538
Loadings maximum	16,40 (15,99)	15,70 (14,70)	34,47 (15,49)
Erreur de reconstruction (2.24)	$6{,}08.10^{-2}$	$5{,}92.10^{-2}$	$5{,}48.10^{-2}$
Critère de Fisher ($\times 10^{-2}$) (2.25)	2,98	1,71	3,55
Gènes inflammatoires de [HZR+11]	68,18%	63,64%	90,91%
p-valeur du pathway IFN-gamma	$5{,}17.10^{-5}$	$3{,}52.10^{-7}$	$6{,}38.10^{-7}$
p-valeur du pathway IL23	$4{,}54.10^{-2}$	$6{,}73.10^{-4}$	$4{,}36.10^{-6}$
Temps de calcul	8 551 s	7 205 s	≈ 18 h
Nombre d'itérations	2 000	2 000	10 000

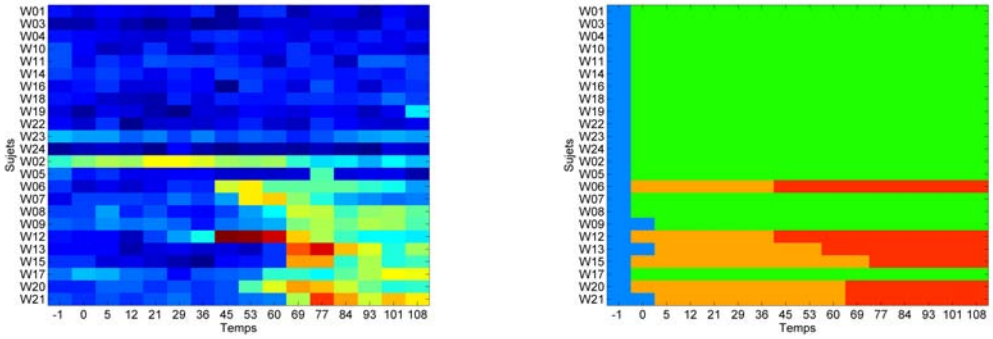
Ces résultats se font donc également ressentir sur la classification en états : estimation de la matrice des états $\widehat{\mathbf{Z}}$, représentée sur la figure 4.7d. En effet, l'algorithme semble mal classer les sujets $\#2, 5, 7, 8, 9$ et 17. Or, ces résultats restent tout de même cohérents avec les valeurs des scores associés visibles sur la figure 4.7c, qui n'ont que très peu de variabilité temporelle.

L'algorithme proposé peut également être utilisé pour estimer les probabilités de transitions inconnues : $\widehat{\boldsymbol{\pi}}_1 = [0.13, 0.74, 0.26]$ et $\widehat{\boldsymbol{\pi}}_3 = [0.86, 0.14]$ (estimations MMSE).



(a) Diagnostic de convergence

(b) Factors estimés, rangés par dominance décroissante



(c) Scores du facteur inflammatoire

(d) Matrice des étiquettes

FIGURE 4.7 – Résultats de simulation de l'algorithme tBLU, sur données H1N1.

4.7 Conclusion

Ce chapitre étend l'algorithme uBLU développé au chapitre 2 par la prise en compte des dépendances temporelles des échantillons d'un même individu. Pour cela, le modèle bayésien linéaire uBLU non-temporel est combiné à un modèle de Markov caché à 4 états dont la structure est en accord avec les modélisations épidémiologiques traditionnelles. Un échantillonneur de Gibbs hybride est proposé pour générer des échantillons distribués asymptotiquement suivant la loi jointe *a posteriori*. Les estimateurs MAP des paramètres et hyperparamètres du modèle peuvent alors être calculés en utilisant les échantillons générés.

Les résultats de simulations obtenus sur données temporelles synthétiques, en comparaison avec la version uBLU non-temporelle, ont démontré les bonnes performances de l'algorithme proposé en termes de démélange et de classification des échantillons en 4 états. Sur données grippales, la méthode présentée dans ce chapitre permet, comme sa version non-temporelle, d'extraire une composante inflammatoire et de discriminer les individus sains des individus malades. Le principal avantage de cette version temporelle est qu'elle propose une classification des échantillons selon l'état du sujet à l'instant considéré (pré-inoculation, asymptomatique, pré-symptomatique ou post-symptomatique), ce qui n'est pas possible avec sa version uBLU non-temporelle.

Remarquons néanmoins que l'algorithme proposé ici suppose la connaissance du nombre de facteurs dans le mélange. Il serait donc intéressant d'étendre cette version avec l'estimation de ce nombre conjointement aux autres paramètres. Une autre perspective à ce travail serait de prendre en compte dans le modèle HMM des états non-stationnaires, exploitant ainsi le fait que les probabilités de transitions des états ont tendance à augmenter au cours du temps.

Contributions du chapitre

La principale contribution de ce chapitre est la classification temporelle des échantillons en quatre états, et ce conjointement au démélange des données génétiques sous contraintes. Cette classification permet ainsi de discriminer bien plus explicitement les sujets malades des sujets sains, et d'indiquer le moment à partir duquel les sujets malades ont developpé de réels symptômes. A notre connaissance,

aucune des méthodes existantes appliquées sur données génétiques ne permet le démélange des données en considérant un *a priori* épidémiologique (structure du modèle de Markov caché).

Conclusions et perspectives

Les travaux de recherche réalisés dans le cadre de cette thèse ont consisté dans l'étude et le développement de modèles bayésiens hiérarchiques de démélange linéaire pour le traitement de données temporelles d'expression de gènes. Ces données, issues des biopuces à ADN, sont souvent de très grande taille et nécessitent la mise en place d'algorithmes de traitement de plus en plus complexes et performants, capables d'identifier des séquences temporelles dans l'expression des différents gènes caractéristiques d'une pathologie. Le modèle de mélange linéaire a largement été utilisé dans la littérature pour répondre à cette problématique [Wes03, BFW+06, CCL+08]. Il consiste en la décomposition des données en un certain nombre (connu ou non) de facteurs élémentaires (ou signatures génétiques) et en l'estimation conjointe de ces facteurs et de leur niveau d'expression (proportion de chacun des gènes étudiés dans chaque facteur, ou score).

Les algorithmes proposés dans cette thèse ont la particularité de prendre en compte les contraintes physiques relatives aux données, à savoir la positivité des facteurs et des proportions, mais aussi la contrainte de somme-à-un des proportions. L'ajout de telles contraintes au modèle de mélange linéaire permet de réduire considérablement l'espace des solutions et offre une meilleure interprétation des paramètres du modèle et des résultats obtenus.

Nous avons choisi d'évoluer dans un cadre bayésien en définissant des lois *a priori* appropriées au modèle et aux contraintes considérées pour chacun des paramètres et hyperparamètres à estimer. La complexité de la loi *a posteriori* résultante nous a incité à utiliser des méthodes de Monte Carlo par chaînes de Markov (méthodes MCMC). Ces méthodes permettent de générer des vecteurs distribués asymptotiquement suivant la loi *a posteriori* d'intérêt, qui sont ensuite utilisés pour calculer les estimateurs bayésiens standards (MAP ou MMSE) des paramètres inconnus.

Un problème inhérent au démélange est l'inférence du nombre de facteurs compris dans le mélange. Le premier algorithme proposé est un algorithme totalement non-supervisé permettant l'estimation conjointe des paramètres (facteurs et proportions) inconnus, mais aussi l'estimation de ce nombre de facteurs grâce à l'utilisation d'un processus de naissance et de mort. Le deuxième algorithme a mis en avant une approche parcimonieuse pour l'estimation de ce nombre, en considérant que seuls quelques facteurs appartenant à une grande bibliothèque de facteurs étaient réellement présents dans le mélange. L'utilisation d'une loi Bernoulli-gaussienne tronquée comme loi *a priori* pour les proportions répond à toutes les contraintes imposées. La difficulté principale de cette approche par parcimonie réside en l'estimation des facteurs de la bibliothèque sachant que certains de ces facteurs ne sont pas présents dans le mélange.

Dans les deux cas précédents, les résultats sur données réelles ont permis d'identifier un groupement de gènes impliqués dans la réponse immunitaire de patients infectés par le virus de la grippe, distinguant ainsi ceux qui tombent malades de ceux qui restent sains. La représentation des scores de ce facteur inflammatoire mettait en évidence une classification en 4 états : 1) avant inoculation d'une pathologie ou d'un traitement, puis 2) post-inoculation asymptomatique, 3) avant déclaration de symptômes (ou réaction au traitement), et enfin 4) après déclaration de symptômes. Le modèle bayésien de démélange linéaire sous contraintes a alors été couplé dans le dernier chapitre à un modèle de Markov caché (HMM) à 4 états permettant d'associer à chaque échantillon une étiquette renseignant sur l'état du sujet à l'instant considéré. Ainsi en plus de réaliser le démélange des données étudiées, ce dernier algorithme a l'avantage de proposer une classification des échantillons.

Les travaux menés dans le cadre de cette thèse ouvrent la voie à de nombreuses perspectives et améliorations. Il serait par exemple intéressant d'étudier d'autres manières d'estimer le nombre de facteurs présents dans le mélange, tout en considérant le modèle de mélange linéaire sous contraintes. Citons par exemple l'utilisation de méthodes basées sur les processus de Dirichlet (sticky HDP), les processus du buffet indien, ou encore une modélisation par mélange infini de gaussiennes.

Il conviendrait également d'améliorer le modèle HMM présenté au dernier chapitre afin d'estimer conjointement aux autres paramètres le nombre de facteurs, mais aussi pour prendre en compte le

fait que les probabilités des états ont tendance à augmenter au cours du temps. Pour cela, il serait donc intéressant de considérer des états non-stationnaires dans le modèle HMM.

Enfin, les algorithmes de Gibbs hybrides proposés tout au long de cette thèse se sont avérés très coûteux en temps de calcul. L'utilisation de méthodes bayésiennes variationnelles pourrait être une alternative intéressante. En effet, ces méthodes permettraient d'approcher les lois *a posteriori* complexes obtenues dans cette thèse par des produits de lois séparables pour chaque paramètre, et donc de proposer des algorithmes itératifs plus abordables en coût de calcul que les méthodes MCMC.

Liste des publications

Revues internationales avec comité de lecture

[BDT+13] C. Bazot, N. Dobigeon, J.-Y. Tourneret, A. K. Zaas, G. S. Ginsburg and A. O. Hero, "Unsupervised Bayesian linear unmixing of gene expression microarrays," *BMC Bioinformatics*, 2013, **14**:99.

Conférences internationales avec comité de lecture

[BDTH10] C. Bazot, N. Dobigeon, J.-Y. Tourneret and A. O. Hero, "Unsupervised Bayesian analysis of gene expression patterns", in *Rec. 44th IEEE Asilomar Conf. Signals, Systems and Computers (Asilomar)*, Pacific Grove, CA, Nov. 2010, pp. 364 – 368.

[BDTH11a] C. Bazot, N. Dobigeon, J.-Y. Tourneret and A. O. Hero, "Bernoulli-Gaussian model for gene expression analysis," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, May 2011, pp. 5996 – 5999.

[BDTH12] C. Bazot, N. Dobigeon, J.-Y. Tourneret and A. O. Hero, "Bayesian analysis of time-evolving gene expression data with hidden Markov model," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Bucharest, Romania, Sept. 2012, pp. 944 – 948.

Conférences nationales avec comité de lecture

[BDTH11b] C. Bazot, N. Dobigeon, J.-Y. Tourneret and A. O. Hero, "Modèle Bernoulli-gaussien pour l'analyse génétique," in *Actes du XXIIIième Colloque GRETSI*, Bordeaux, France, Sept. 2011, in French.

Bibliographie

[AD99] C. Andrieu and A. Doucet. Joint Bayesian model selection and estimation of noisy sinusoids via reversible jump MCMC. *Signal Processing, IEEE Transactions on*, 47(10):2667–2676, Oct. 1999.

[BDPD+12] J.M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Qian Du, P. Gader, and J. Chanussot. Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 5(2):354–379, 2012.

[BDT+13] C. Bazot, N. Dobigeon, J.-Y. Tourneret, A. K. Zaas, G. S. Ginsburg, and A. O. Hero. Unsupervised Bayesian linear unmixing of gene expression microarrays. *BMC Bioinformatics*, 14(1):99, 2013.

[BDTH10] C. Bazot, N. Dobigeon, J.-Y. Tourneret, and A. O. Hero. Unsupervised Bayesian analysis of gene expression patterns. In *Rec. 44th IEEE Asilomar Conf. Signals, Systems and Computers (Asilomar)*, pages 364–368, Pacific Grove, CA, Nov. 2010.

[BDTH11a] C. Bazot, N. Dobigeon, J.-Y. Tourneret, and A. O. Hero. Bernoulli-Gaussian model for gene expression analysis. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, pages 5996–5999, Prague, Czech Republic, May 2011.

[BDTH11b] C. Bazot, N. Dobigeon, J.-Y. Tourneret, and A. O. Hero. Modèle Bernoulli-gaussien pour l'analyse génétique. In *Actes du XXIIIième Colloque GRETSI*, Bordeaux, France, Sept. 2011. in French.

[BDTH12] C. Bazot, N. Dobigeon, J.-Y. Tourneret, and A. O. Hero. Bayesian analysis of time-evolving gene expression data with hidden Markov model. In *Proc. European Signal Processing Conf. (EUSIPCO)*, pages 944–948, Bucharest, Romania, Sept. 2012.

[BFW+06] F. Baty, M. Facompre, J. Wiegand, J. Schwager, and M. Brutsche. Analysis with respect to instrumental variables for the exploration of microarray data structures. *BMC Bioinformatics*, 7(1):422, 2006.

[BG98] S. P. Brooks and A. Gelman. General Methods for Monitoring Convergence of Iterative Simulations. *Journal of the American Statistical Association*, 7(7):434–455, 1998.

[BG00] S. P. Brooks and P. Giudici. Markov chain monte carlo convergence assessment via two-way analysis of variance. *Journal of Computational and Graphical Statistics*, 9(2):266–285, June 2000.

[BGM03] J. O. Berger, J. K. Ghosh, and N. Mukhopadhyay. Approximations and consistency of Bayes factors as model dimension grows. *Journal of Statistical Planning and Inference*, 112(1Ű2):241–258, 2003. Special issue II: Model Selection, Model Diagnostics, Empirical Bayes and Hierarchical Bayes.

[BM11] J. Baek and G. J. McLachlan. Mixtures of common t-factor analyzers for clustering high-dimensional microarray data. *Bioinformatics*, 27(9):1269–1276, 2011.

[CC94] T. F. Cox and M. A. A. Cox. *Multidimensional Scaling.* Chapman and Hall, 1994.

[CCL+08] C. Carvalho, J. T. Chang, J. Lucas, J. R. Nevins, Q. Wang, and M. West. High-dimensional sparse factor modelling: Applications in gene expression genomics. *JASA*, 103(484):1438–1456, 2008.

[CCP+10] B. Chen, M. Chen, J. Paisley, A. Zaas, C. Woods, G. S. Ginsburg, A. O. Hero, J. Lucas, D. Dunson, and L. Carin. Bayesian inference of the number of factors in gene-expression analysis: Application to human virus challenge studies. *BMC Bioinformatics*, 11(1):552, 2010.

[CGGG03] F. S. Collins, E. D. Green, A. E. Guttmacher, and M. S. Guyer. A vision for the future of genomics research. *Nature*, 422(6934):835–847, 2003.

[Che98] R. C. H. Cheng. Bayesian model selection when the number of components is unknown. In *Simulation Conference Proceedings, 1998. Winter*, volume 1, pages 653–659, Dec. 1998.

[CMR05] O. Cappé, E. Moulines, and T. Ryden. *Inference in Hidden Markov Models*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[Coh60] Jacob Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, Apr. 1960.

[Col01] C. Collet. *Précis de Télédétection, Volume 3 : Traitements Numériques d'Images de Télédétection*. Universités francophones. Presses de l'Universite du Quebec, 2001.

[CWB09] B. Coburn, B. Wagner, and S. Blower. Modeling influenza epidemics and pandemics: insights into the future of swine flu (H1N1). *BMC Medicine*, 7(1):30, 2009.

[CZ02] J. M. Castelloe and D. L. Zimmerman. Convergence assessment for reversible jump mcmc samplers. Technical report, 2002.

[DBC+99] D. J. Duggan, M. Bittner, Y. Chen, P. Meltzer, and J. M. Trent. Expression profiling using cDNA microarrays. *Nat Genet*, 21:10–14, 1999.

[DBTK12] N. Dobigeon, A. Basarab, J.-Y. Tourneret, and D. Kouamé. Regularized Bayesian compressed sensing in ultrasound imaging. In *Proc. European Signal Processing Conf. (EUSIPCO)*, pages 2600–2604, Bucharest, Romania, Sept. 2012.

[DEKM98] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, April 1998.

[DGI06] M. Davy, S. Godsill, and J. Idier. Bayesian analysis of polyphonic western tonal music. *Journal of the Acoustical Society of America*, 119(4):2498–2517, 2006.

[DHS00] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, New York, 2nd edition, 2000.

[DHT09] N. Dobigeon, A. O. Hero, and J.-Y. Tourneret. Hierarchical Bayesian sparse image reconstruction with application to MRFM. *IEEE Trans. Image Processing*, 18(9):2059–2070, Sept. 2009.

[DMC+09] N. Dobigeon, S. Moussaoui, M. Coulon, J.-Y. Tourneret, and A. O. Hero. Joint Bayesian endmember extraction and linear unmixing for hyperspectral imagery. *IEEE Trans. Signal Processing*, 57(11):4355–4368, Nov. 2009.

[DTC08] N. Dobigeon, J.-Y. Tourneret, and C.-I. Chang. Semi-supervised linear spectral unmixing using a hierarchical Bayesian model for hyperspectral imagery. *IEEE Trans. Signal Processing*, 56(7):2684–2695, July 2008.

[DTS06] N. Dobigeon, J.-Y. Tourneret, and J. D. Scargle. Joint segmentation of multivariate Poissonian time series applications to burst and transient source experiments. In *Proc. European Signal Processing Conf. (EUSIPCO)*, Florence, Italy, Sept. 2006.

[EDL02] R. Edgar, M. Domrachev, and A. E. Lash. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, 30(1):207–210, 2002.

[FDF+10] E. J. Fertig, J. Ding, A. V. Favorov, G. Parmigiani, and M. F. Ochs. CoGAPS: an R/C++ package to identify patterns and biological process activity in transcriptomic data. *Bioinformatics*, 26:2792–2793, Nov. 2010.

[FYHL07] P. Fogel, S. S. Young, D. M. Hawkins, and N. Ledirac. Inferential, robust non-negative matrix factorization analysis of microarray data. *Bioinformatics*, 23:44–49, Jan. 2007.

[GCSR03] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC, London, 2 edition, July 2003.

[GD08] J Ghosh and D. B. Dunson. "bayesian model selection in factor analytic models,". *Random Effect and Latent Variable Model Selection*, 2008.

[GFGV58] Jackson G., Dowling H. F., Spiesman I. G., and Boand A. V. Transmission of the common cold to volunteers under controlled conditions: I. the common cold as a clinical entity. *Archives of Internal Medicine*, 101(2):267–278, 1958.

[GG05] T. L. Griffiths and Z. Ghahramani. Infinite latent feature models and the indian buffet process. *Advances in Neural Information Processing Systems*, pages 475–482, 2005.

[GR92] A. Gelman and D.B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7:457–511, 1992.

[Gre95] P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[GRS96] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, London, 1996.

[HC01] D. C. Heinz and C.-I. Chang. Fully constrained least squares linear spectral mixture analysis method for material qualification in hyperspectral imagery. *IEEGRS*, 39:529–545, March 2001.

[HKO01] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. New York: John Wiley, 2001.

[HWQZ11] Q. Huang, L.-Y. Wu, J.-B. Qu, and X.-S. Zhang. Analyzing time-course gene expression data using profile-state hidden Markov model. In *Proc. IEEE Int. Conf. Systems Biology (ISB)*, pages 351–355, Sept. 2011.

[HZR+11] Y. Huang, A. K. Zaas, A. Rao, N. Dobigeon, P. J. Woolf, T. Veldman, N. C. Oien, M. T. McClain, J. B. Varkey, B. Nicholson, L. Carin, S. Kingsmore, C. W. Woods, G. S. Ginsburg, and A. O. Hero. Temporal dynamics of host molecular responses differentiate symptomatic and asymptomatic influenza A infection. *PLoS Genetics*, 8(7), Aug. 2011.

[IHC+03] R. A. Irizarry, B. Hobbs, F. Collin, Y. D. Beazer-Barclay, K. J. Antonellis, U. Scherf, and T. P. Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2):249–264, 2003.

[Jol86] I. T. Jolliffe. *Principal Component Analysis*. New York: Springer-Verlag, 1986.

[JS04] I. M. Johnstone and B. W. Silverman. Needles and straw in haystacks: empirical Bayes estimates of possibly sparse sequences. *The Annals of Statistics*, 32(4):1594–1649, 2004.

[KG07] D. Knowles and Z. Ghahramani. Infinite sparse factor analysis and infinite independent components analysis. *7th International Conference on Independent Component Analysis and Signal Separation*, pages 381–388, 2007.

[KHB88] A. Kundu, Y. He, and P. Bahl. Recognition of handwritten word: first and second order hidden Markov model based approach. In *Computer Vision and Pattern Recognition, 1988. Proceedings CVPR '88., Computer Society Conference on*, pages 457–462, June 1988.

[KM02] N. Keshava and J. F. Mustard. Spectral unmixing. *IEEE Signal Processing Magazine*, 19:44–57, Jan. 2002.

[KO09] A. V. Kossenkov and M. F. Ochs. Matrix factorization for recovery of biological processes from microarray data. *Methods Enzymol*, 467:59–77, 2009.

[KVG+08] W. Kong, C. R. Vanderburg, H. Gunshin, J. T. Rogers, and X. Huang. A review of independent component analysis application to microarray gene expression data. *BioTechniques*, 45(5):501–520, Nov. 2008.

[LS00] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Proceedings of Neural Information Processing Systems*, 2000.

[LW03] H. F. Lopes and M. West. Bayesian model assessment in factor analysis. *Statistica Sinica*, 14:41–67, 2003.

[MBI05] V. Mazet, D. Brie, and J. Idier. Simuler une distribution normale à support positif à partir de plusieurs lois candidates. In *Actes 20^e coll. GRETSI*, volume 2, pages 1077–1080, Sept. 2005.

[MBP02] G. J. McLachlan, R. W. Bean, and D. Peel. A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18(3):413–422, 2002.

[Meu05] N. Le Meur. *De l'acquisition des données de puces à ADN vers leur interprétation.* PhD thesis, Université de Nantes, 2005.

[MKG+02] T. D. Moloshok, R. R. Klevecz, J. D. Grant, F. J. Manion, W. F. Speier, and M. F. Ochs. Application of Bayesian decomposition for analysing microarray data. *Bioinformatics*, 18:566–575, Apr. 2002.

[Nar91] A. Narayanan. Algorithm AS 266: Maximum Likelihood Estimation of the Parameters of the Dirichlet Distribution. *Applied Statistics*, 40(2):365–374, 1991.

[NBAS07] É. Nicand, Y. Buisson, M. Aymard, and P. Saliou. *La grippe en face.* X. Montauban, Montrouge, France, 2007.".

[NBD05] J. M. Nascimento and J. M. Bioucas-Dias. Vertex component analysis: A fast algorithm to unmix hyperspectral data. *IEEE Trans. Geosci. and Remote Sensing*, 43(4):898–910, April 2005.

[NHN+11] V. Nikulin, T.-H. Huang, S.-K. Ng, S. Rathnayake, and G. J. McLachlan. A very fast algorithm for matrix factorization. *Statistics & Probability Letters*, 81(7):773–782, 2011.

[OSAMB99] M. F. Ochs, R. S. Stoyanova, F. Arias-Mendoza, and T. R. Brown. A new method for spectral decomposition using a bilinear Bayesian approach. *Journal of Magnetic Resonance*, 137(1):161–176, 1999.

[PADF02] E. Punskaya, C. Andrieu, A. Doucet, and W. Fitzgerald. Bayesian curve fitting using MCMC with applications to signal segmentation. *IEEE Trans. Signal Processing*, 50(3):747–758, March 2002.

[PC09] J. Paisley and L. Carin. Nonparametric factor analysis with beta process priors. In *Proc. 26th Annual Int. Conf. on Machine Learning*, pages 777–784. ACM, 2009.

[Rab89] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77(2):257–286, 1989.

[RC99] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, 1 edition, Aug. 1999.

[Rob95] C. P. Robert. Simulation of truncated normal variables. *Statistics and Computing*, 5:121–125, June 1995.

[Rob96] G. O. Roberts. Markov chain concepts related to sampling algorithms. In *Markov chain Monte Carlo in practice*, pages 45–57. Chapman & Hall, London, 1996.

[RR98] C. P. Robert and S. Richardson. Markov chain monte carlo methods. In Christian P. Robert, editor, *Discretization and MCMC Convergence Assessment*, volume 135, pages 1–25. Springer-Verlag, New York, 1998.

[SAK+09] C. F. Schaefer, K. Anthony, S. Krupa, J. Buchoff, M. Day, T. Hannay, and K. H. Buetow. PID: the Pathway Interaction Database. *Nucleic Acids Research*, 37(suppl 1):D674–D679, 2009.

[Sch78] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6, March 1978.

[SSS03] A. Schliep, A. Schönhuth, and C. Steinhoff. Using hidden Markov models to analyze gene expression time course data. *Bioinformatics*, 19(suppl 1):i255–i263, 2003.

[SWG+04] J. Schmutz, J. Wheelera, J. Grimwood, J. Yang, C. Caoile, E. Bajorek, S. Black, Y. M. Chan, M. Denys, J. Escobar, D. Flowers, D. Fotopulos, C. Garcia, M. Gomez, E. Gonzales, L. Haydu, F. Lopez, L. Ramirez, J. Retterer, A. Rodriguez, S. Rogers, A. Salazar, M. Tsai, and R. M. Myers. Quality assessment of the human genome sequence. *Nature*, 429:365–368, 2004.

[TRH09] M. Ting, R. Raich, and A. O. Hero. Sparse image reconstruction for molecular imaging. *IEEE Trans. Image Process*, 18(6):1215–1227, June 2009.

[TRK10] K. E. Themelis, A. A. Rontogiannis, and K. Koutroumbas. Semi-supervised hyperspectral unmixing via the weighted lasso. *Int. Conf. Acoust., Speech, and Signal Proc.*, 2010.

[Wes03] M. West. Bayesian Factor Regression Models in the "Large p, Small n" Paradigm. In *Bayesian Statistics*, pages 723–732. Oxford University Press, 2003.

[Win99] M. E. Winter. N-FINDR: an algorithm for fast autonomous spectral end-member determination in hyperspectral data. volume 3753, pages 266–275. SPIE, 1999.

[WTH09] D. M. Witten, R. Tibshirani, and T. Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534, July 2009.

[YR01] K. Y. Yeung and W. L. Ruzzo. Principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9):763–774, 2001.

[ZCV+09] A. K. Zaas, M. Chen, J. Varkey, T. Veldman, A. O. Hero, J. Lucas, Y. Huang, R. Turner, A. Gilbert, R. Lambkin-Williams, N. C. Øien, B. Nicholson, S. Kingsmore, L. Carin, C. W. Woods, and G. S. Ginsburg. Gene expression signatures diagnose influenza and other symptomatic respiratory viral infections in humans. *Cell host & microbe*, 6(3):207–17, Sept. 2009.

[ZHT04] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15:2006, 2004.