Université de Montréal

Modélisation des bi-grappes et sélection des variables pour des données de grande dimension: application aux données d'expression génétique

par
Thierry Chekouo Tekougang

Département de Mathématiques et de Statistique
Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures
en vue de l'obtention du grade de Philosophiæ Doctor (Ph.D.)
en Statistique

Août, 2012

Université de Montréal
Faculté des études supérieures

Cette thèse intitulée:

Modélisation des bi-grappes et sélection des variables pour des données de grande dimension: application aux données d'expression génétique

présentée par:

Thierry Chekouo Tekougang

a été évaluée par un jury composé des personnes suivantes:

Jean-François Angers,	président-rapporteur
Alejandro Murua,	directeur de recherche
Mylène Bédard,	membre du jury
David Stephens,	examinateur externe
Nicolas Lartillot,	représentant du doyen de la FES

Thèse acceptée le: 23 Novembre 2012

RÉSUMÉ

Le regroupement des données est une méthode classique pour analyser les matrices d'expression génétiques. Lorsque le regroupement est appliqué sur les lignes (gènes), chaque colonne (conditions expérimentales) appartient à toutes les grappes obtenues. Cependant, il est souvent observé que des sous-groupes de gènes sont seulement co-régulés (i.e. avec les expressions similaires) sous un sous-groupe de conditions. Ainsi, les techniques de bi-regroupement ont été proposées pour révéler ces sous-matrices des gènes et conditions. Un bi-regroupement est donc un regroupement simultané des lignes et des colonnes d'une matrice de données. La plupart des algorithmes de bi-regroupement proposés dans la littérature n'ont pas de fondement statistique. Cependant, il est intéressant de porter une attention sur les modèles sous-jacents à ces algorithmes et de développer des modèles statistiques permettant d'obtenir des bi-grappes significatives. Dans cette thèse, nous faisons une revue de littérature sur les algorithmes qui semblent être les plus populaires. Nous groupons ces algorithmes en fonction du type d'homogénéité dans la bi-grappe et du type d'imbrication que l'on peut rencontrer. Nous mettons en lumière les modèles statistiques qui peuvent justifier ces algorithmes. Il s'avère que certaines techniques peuvent être justifiées dans un contexte bayésien. Nous développons une extension du modèle à carreaux (*plaid*) de bi-regroupement dans un cadre bayésien et nous proposons une mesure de la complexité du bi-regroupement. Le critère d'information de déviance (DIC) est utilisé pour choisir le nombre de bi-grappes. Les études sur les données d'expression génétiques et les données simulées ont produit des résultats satisfaisants.

À notre connaissance, les algorithmes de bi-regroupement supposent que les gènes et les conditions expérimentales sont des entités indépendantes. Ces algorithmes n'incorporent pas de l'information biologique *a priori* que l'on peut avoir sur les gènes et les conditions. Nous introduisons un nouveau modèle bayésien à carreaux pour les données d'expression génétique qui intègre les connaissances biologiques et prend en compte l'interaction par paires entre les gènes et entre les conditions à travers un champ de Gibbs. La dépendance entre ces entités est faite à partir des graphes relationnels, l'un pour les gènes et l'autre pour les conditions. Le graphe des gènes et celui des conditions sont construits par les k-voisins les plus proches et permet de définir la distribution *a priori* des étiquettes comme des modèles auto-logistiques. Les similarités des gènes se calculent en utilisant l'ontologie des gènes (GO). L'estimation est faite par une procédure hybride qui mixe les MCMC avec une variante de l'algorithme de Wang-Landau. Les expériences sur les données simulées et réelles montrent la performance de notre approche.

Il est à noter qu'il peut exister plusieurs variables de bruit dans les données à micropuces, c'est-à-dire des variables qui ne sont pas capables de discriminer les groupes. Ces variables peuvent masquer la vraie structure du regroupement. Nous proposons un modèle inspiré de celui à carreaux qui, simultanément retrouve la vraie structure de regroupement et identifie les variables discriminantes. Ce problème est traité en utilisant un vecteur latent binaire, donc l'estimation est obtenue via l'algorithme EM de Monte Carlo. L'importance échantillonnale est utilisée pour réduire le coût computationnel de l'échantillonnage Monte Carlo à chaque étape de l'algorithme EM. Nous proposons un nouveau modèle pour résoudre le problème. Il suppose une superposition additive des grappes, c'est-à-dire qu'une observation peut être expliquée par plus d'une seule grappe. Les exemples numériques démontrent l'utilité de nos méthodes en terme de sélection de variables et de regroupement.

Mots clés : groupement, critère d'information de déviance, expression génétique, ontologie des gènes, algorithme de Wang-Landau, modèle auto-logistique, sélection des variables, le modèle à carreaux, algorithme EM de Monte Carlo, l'importance échantillonnale.

ABSTRACT

Clustering is a classical method to analyse gene expression data. When applied to the rows (e.g. genes), each column belongs to all clusters. However, it is often observed that the genes of a subset of genes are co-regulated and co-expressed in a subset of conditions, but behave almost independently under other conditions. For these reasons, biclustering techniques have been proposed to look for sub-matrices of a data matrix. Biclustering is a simultaneous clustering of rows and columns of a data matrix. Most of the biclustering algorithms proposed in the literature have no statistical foundation. It is interesting to pay attention to the underlying models of these algorithms and develop statistical models to obtain significant biclusters. In this thesis, we review some biclustering algorithms that seem to be most popular. We group these algorithms in accordance to the type of homogeneity in the bicluster and the type of overlapping that may be encountered. We shed light on statistical models that can justify these algorithms. It turns out that some techniques can be justified in a Bayesian framework. We develop an extension of the biclustering plaid model in a Bayesian framework and we propose a measure of complexity for biclustering. The deviance information criterion (DIC) is used to select the number of biclusters. Studies on gene expression data and simulated data give satisfactory results.

To our knowledge, the biclustering algorithms assume that genes and experimental conditions are independent entities. These algorithms do not incorporate prior biological information that could be available on genes and conditions. We introduce a new Bayesian plaid model for gene expression data which integrates biological knowledge and takes into account the pairwise interactions between genes and between conditions via a Gibbs field. Dependence between these entities is made from relational graphs, one for genes and another for conditions. The graph of the genes and conditions is constructed by the *k*-nearest neighbors and allows to define a priori distribution of labels as auto-logistic models. The similarities of genes are calculated using gene ontology (GO). To estimate the parameters, we adopt a hybrid procedure that mixes MCMC with a variant of the Wang-Landau algorithm. Experiments on simulated and real data show the performance of our approach.

It should be noted that there may be several variables of noise in microarray data. These variables may mask the true structure of the clustering. Inspired by the plaid model, we propose a model that simultaneously finds the true clustering structure and identifies discriminating variables. We propose a new model to solve the problem. It assumes that an observation can be explained by more than one cluster. This problem is addressed by using a binary latent vector, so the estimation is obtained via the Monte Carlo EM algorithm. Importance Sampling is used to reduce the computational cost of the Monte Carlo sampling at each step of the EM algorithm. Numerical examples demonstrate the usefulness of these methods in terms of variable selection and clustering.

Keywords: Clustering, deviance information criterion, gene expression, gene ontology, Wang-Landau algorithm, auto-logistic models, variable selection, plaid model, Monte Carlo EM algorithm, Importance Sampling.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES

AIC	Critère d'information de Akaike
ANOVA	Analyse de la variance
ARN	Acide ribonucléique
ARNm	Acide ribonucléique messager
BIC	Critère d'information bayésien
Cov	Covariance
DIC	Critère d'information de déviance
DNA	Acide désoxyribonucléique
EM	Algorithme d'espérance-maximisation
GO	Ontologie des gènes
ICM	Algorithme itératif du mode conditionnel
IS	Importance échantillonnale
MCMC	Monte Carlo par chaines de Markov
MCEM	Monte Carlo espérance-maximisation

Je dédie ce travail à

mon épouse ***Eugenie*** et
mes enfants ***Océane, Albiol et Elisabeth***.

REMERCIEMENTS

J'aimerais exprimer ma profonde gratitude à mon directeur de thèse M. Alejandro Murua, pour sa disponibilité, son soutien et l'apprentissage continu qu'il a su me dispenser durant ces années. Je remercie aussi M. Christian Léger pour m'avoir incité à venir au Département de mathématiques et statistique. Merci également aux professeurs et membres du département qui ont su m'offrir un cadre tout à fait propice aux études et aussi pour leur disponibilité. Je ne saurais terminer sans remercier le CRM, l'ISM, le Département de mathématiques et statistique, la Faculté des études supérieures et post-doctorales pour leur soutien financier. Enfin, je remercie mes parents, mes frères et soeurs, ma belle famille, mes amis pour leur soutien moral et inconditionnel.

INTRODUCTION

Les données d'expression génétique obtenues par les technologies micro-puces d'ADN sont une forme de données génomiques à haut débit. Elles fournissent des mesures relatives de niveaux d'ARNm (Acide ribonucléique messager) pour des milliers de gènes dans un échantillon biologique (Lee et al. [14]). Typiquement, ces données contiennent un grand nombre (jusqu'à plusieurs dizaines de milliers) de gènes, et un nombre d'échantillons (individus) relativement faible. Ces mesures sont obtenues en immobilisant les gènes sur des spots disposés dans une grille (« array ») sur un support qui est typiquement une lame de verre, une plaquette de quartz, ou une membrane de nylon. À partir d'un échantillon d'intérêt, par exemple une biopsie tumorale, l'ARNm est extrait, marqué et hybridé à la grille. La mesure de la quantité de marques sur chaque spot donne une valeur d'intensité qui devrait être corrélée à l'abondance du transcrit correspondant d'ARN dans l'échantillon (Huber et al. [10]). La connaissance et l'analyse des données d'expression génétique peuvent s'avérer utile dans le diagnostic médical, le traitement et la conception de médicaments. Ces données à micro-puces peuvent être vues comme une matrice de données où les lignes et les colonnes représentent respectivement les gènes et les conditions ou échantillons expérimentaux (par exemple : patients, tissus, périodes de temps). Chaque cellule de la matrice est un nombre réel et représente le niveau d'expression d'un gène sous une condition expérimentale.

Une méthode standard pour analyser les données d'expression génétique est le *regroupement* (*clustering* en anglais) (Kerr et al. [12]) qui peut se faire soit sur les gènes, soit sur les conditions expérimentales. Les techniques de regroupement (*k*-moyennes : Hartigan et Wong [8], regroupement hiérarchique : Ward [19], modèle basé sur le regroupement : Fraley et Raftery [5]) ont prouvé leur utilité pour comprendre la fonction des gènes, la régulation des gènes, les processus cellulaires et les sous-types de cellules. Les gènes co-exprimés (avec les expressions similaires) peuvent être groupés ensemble et sont susceptible d'être impliqués dans le même processus cellulaire (Jiang et al. [11]).

Dans un regroupement de gènes, toutes les conditions expérimentales (échantillons) sont partagés par toutes les autres grappes. Il en est de même dans un regroupement de conditions où celles-ci sont groupées en utilisant tous les gènes. De plus, les grappes obtenues sont exclusives et exhaustives puisqu'elles forment une partition des gènes ou des conditions. Cependant, il est bien connu en biologie moléculaire qu'un processus cellulaire qui contient un petit sous-ensemble de gènes peut être actif seulement dans un sous-ensemble de conditions. En outre, un seul gène peut participer à plusieurs chemins (« pathways ») qui peuvent ou ne peuvent pas être co-actifs dans toutes les conditions. Ainsi, un gène peut donc participer dans plusieurs grappes ou dans rien du tout.

Le *bi-regroupement* tente de surmonter ces limites de regroupement. La notion de bi-regroupement (ou biclustering en anglais, aussi connu comme co-clustering ou two-way clustering) réfère au regroupement simultané de lignes et de colonnes d'une matrice de données. Chaque grappe obtenue de ce bi-regroupement sera appeléen *bi-grappe* (*bi-cluster* en anglais) . C'est donc une sous-matrice de la matrice des données dont les lignes exhibent un comportement similaire à travers les colonnes et vice versa. Un regroupement quant à lui ne peut s'appliquer que sur les lignes, ou sur les colonnes. Un regroupement fournit un modèle global tandis que le bi-regroupement donne un modèle local. Les lignes ou les colonnes peuvent appartenir à plusieurs bi-grappes. La bi-grappe peut être alors *imbriquée*. La détection des bi-grappes imbriquées fournit une meilleure représentation de la réalité biologique. Le bi-regroupement n'a pas seulement des applications en bioinformatique, il a aussi des applications importantes en marketing (Dolnicar et al. [4]), et dans l'exploration de texte (text-mining, Busygin et al. [2]). Jusqu'à récemment, le bi-regroupement n'avait pas reçu beaucoup d'attention dans la communauté statistique. Très peu de modèles de bi-regroupement ont été proposés dans la littérature.

L'objectif de cette thèse est de trouver des nouvelles méthodes qui permettent de sélectionner des bi-grappes dans une matrice de donnée. Nous présentons une revue des algorithmes de bi-regroupement qui sont classés en fonction du type d'imbrication et du type d'homogénéité. Nous illuminons les modèles statistiques sous-jacents à certains algorithmes populaires. Nous présentons également deux nouveaux modèles de bi-regroupement probabilistes qui sont des extensions du modèle à carreaux (ou *plaid*) de Lazzeroni et Owen [13]. Le premier modèle est un modèle à carreaux pénalisé. Il est bayésien et contient un paramètre relié à la distribution *a priori* des étiquettes d'appartenance des lignes et des colonnes. Ce paramètre contrôle le niveau d'imbrication entre les bi-grappes et permet de lier les deux algorithmes de bi-regroupement les plus populaires (l'algorithme de Cheng et Church [3] et celui de Lazzeroni et Owen [13]). Les méthodes d'échantillonnage de Gibbs [6] et de Metropolis-Hasting [9] nous permettent d'estimer les bi-grappes. Le second modèle est aussi bayésien et tient compte de l'information *a priori* sur les gènes et les conditions de la matrice d'expression génétique. Cette information est incorporée à travers un graphe relationnel par les modèles auto-logistiques Besag [1]. L'algorithme de Wang-Landau [18], combiné avec les méthodes de Monte Carlo par chaines de Markov, est utilisé pour estimer les paramètres. L'utilisation de l'algorithme de Wang-Landau est utile pour contourner l'indisponibilité de la constante de normalisation des distributions *a priori* des étiquettes.

Lorsqu'on regroupe les échantillons ou les conditions expérimentales, le but est de trouver les structures de phénotype des conditions qui sont généralement liées à certaines maladies ou à des effets des médicaments. Il a été démontré (Golub et al. [7]) que les phénotypes d'échantillons peuvent être discriminés à travers seulement un petit nombre de gènes qui ont des niveaux d'expression fortement corrélés avec les classes. Ces gènes sont donc informatifs. Les autres gènes sont non informatifs (ou bruits) car ils sont considérés non pertinents pour expliquer le regroupement en classes. Il est donc souvent nécessaire en pratique de sélectionner les gènes significatifs capables de révéler la vraie structure du regroupement dans les échantillons. Inspiré du modèle à carreaux, nous présentons un nouveau modèle capable de sélectionner les variables dans un contexte de regroupement des données. Ce modèle est relié à ce qui est appelé dans la littérature le modèle de mélange multiplicatif pour le regroupement avec imbrication (Qiang et Banerjee [16]). Il est différent et plus général que les modèles considérés dans la littérature (Pan et Shen [15] et Tadesse et al. [17]) car il permet non seulement l'imbrication entre les grappes, mais aussi, dans chaque grappe, les lignes et les colonnes se comportent de façon similaire (comme dans une bi-grappe à valeurs cohérentes sur les lignes et les colonnes). De plus, l'utilisation de la variable latente de sélection de variable dans un algorithme EM de Monte Carlo (Wei et Tanner [20]) semble être nouveau dans ce contexte.

Le premier chapitre de cette thèse introduit le concept de bi-regroupement et présente certains algorithmes en fonction de l'homogénéité recherchée dans les bi-grappes, et le type d'imbrication que l'on peut rencontrer. C'est un chapitre introductif à la thèse qui permet de comprendre le bi-regroupement et les approches utilisées dans la littérature. Il constituera le premier article de cette thèse intitulé : « A Survey of Practical Biclustering Methods For Gene Expression Data ». L'autre auteur de cet article est mon superviseur, M. Alejandro Murua, professeur à l'Université de Montréal. La contribution de l'étudiant dans cet article repose sur la dérivation des formules, la co-écriture du manuscrit et la revue de la littérature sur les algorithmes de bi-regroupement. Le second chapitre décrit le modèle à carreaux pénalisé et les algorithmes reliés. Il est écrit sous la forme d'un article intitulé : « The Penalized Plaid model and Related Algorithms ». Cet article a été soumis à « Journal of Applied Statistics ». L'autre auteur de cet article est également mon superviseur. Dans ce papier, l'étudiant a partiellement émis des idées sur la construction du modèle, partiellement dérivé les formules, partiellement conçu les expériences sur des données simulées et réelles, implémenté et exécuté les algorithmes, puis co-écrit le manuscrit. Le troisième chapitre utilise des champs de Gibbs afin d'introduire de l'information *a priori* dans le modèle à carreaux. Il est écrit sous forme d'article : « The Gibbs-Plaid Biclustering Model ». Les autres auteurs de cet article sont : Alejandro Murua, professeur à l'Université de Montréal, et Wolfgang Raffelsberger de l'Institut de la génétique et de la biologie moléculaire et cellulaire (IGBMC) de l'Université de Strasbourg, France. L'étudiant a partiellement émis des idées sur la construction du modèle, partiellement dérivé les formules, proposé l'algorithme de Wang-landau pour estimer les paramètres, partiellement conçu les expériences sur des données simulées et réelles, implémenté et exécuté les algorithmes, puis a co-écrit le manuscrit. Le quatrième et dernier chapitre présente un modèle de sélection de variable dans un contexte de regroupement de données. Il est rédigé sous forme d'article intitulé : « Variable Selection With The Plaid Mixture Model For Clustering ». L'autre auteur est mon superviseur Alejandro Murua. L'étudiant a partiellement émis des idées sur la construction du modèle, dérivé les formules, conçu l'algorithme EM de Monte Carlo, partiellement conçu les expériences sur des données simulées et réelles, implémenté et exécuté les algorithmes puis a co-écrit le manuscrit.

BIBLIOGRAPHIE

[1] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236, 1974.

[2] S. Busygin, O. Prokopyev et P. M. Pardalos. Biclustering in data mining. *Computers & Operations Research*, 35(9):2964 – 2987, 2008.

[3] Y. Cheng et G.M. Church. Biclustering of expression data. *Int. Conf. Intelligent Systems for Molecular Biology*, 12:61–86, 2000.

[4] S. Dolnicar, S. Kaiser, K. Lazarevski et F. Leisch. Biclustering. *Journal of Travel Research*, 51(1):41–49, 2012.

[5] C. Fraley et A. E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97:611–631, 2000.

[6] S. Geman et D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[7] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield et E. S. Lander. Molecular classification of cancer : class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

[8] J. A. Hartigan et M. A. Wong. Algorithm as 136 : A *k*-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[9] W.K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[10] W. Huber, A. V. Heydebreck et Vingron M. Analysis of microarray gene expression data. Dans *in 'Handbook of Statistical Genetics', 2nd edn*. Wiley, 2003.

[11] D. Jiang, C. Tang et A. Zhang. Cluster analysis for gene expression data : A survey. *IEEE Trans. on Knowl. and Data Eng.*, 16(11):1370–1386, 2004. ISSN 1041-4347.

[12] G. Kerr, H. J. Ruskin, M. Crane et P. Doolan. Techniques for clustering gene expression data. *Comput. Biol. Med.*, 38(3):283–293, mars 2008. ISSN 0010-4825.

[13] L. Lazzeroni et A. Owen. Plaid models for gene expression data. *Statistica Sinica*, 12:61–86, 2002.

[14] H. K. Lee, A. K. Hsu, J. Sajdak, J. Qin et P. Pavlidis. Coexpression analysis of human genes across many microarray data sets. *Genome Research*, 14:1085–1094, 2004.

[15] W. Pan et X. Shen. Penalized model-based clustering with application to variable selection. *Journal of Machine Learning Research*, 8:1145–1164, 2007. ISSN 1532-4435.

[16] F. Qiang et A. Banerjee. Multiplicative mixture models for overlapping clustering. Dans *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*, pages 791 –796, dec. 2008.

[17] M. G. Tadesse, N. Sha et M. Vannucci. Bayesian variable selection in clustering high-dimensional data. *Journal of the American Statistical Association*, 100:602–617, 2005.

[18] F. Wang et D. P. Landau. Efficient, multiple-range random walk algorithm to calculate the density of states. *Physical Review Letters*, 86:2050–2053, Mar 2001.

[19] J. H. Ward. Hierarchical groupings to optimize an objective function. *Journal American Statistical Association*, 58:234–244, 1963.

[20] G. C. G. Wei et M. A. Tanner. A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association*, 85(411):699–704, 1990.

CHAPITRE 1

A SURVEY OF PRACTICAL BICLUSTERING METHODS FOR GENE EXPRESSION DATA

1.1 Introduction and notation

With the recent advances in DNA microarray technology and genome sequencing, it has become possible to measure at once gene expression levels of many thousands of genes within a number of different experimental samples or conditions (e.g. different patients, different tissues, or different time points). Data collected with this technology are named *gene expression data*. They may be of great value in medical diagnosis, treatment, and drug design (Wu et al. [37]). Some researchers even claim that the future of medicine lies in this new type of technology. Gene expression data (or microarray data) can be viewed as a data matrix where rows and columns represent genes and experimental conditions respectively. Each matrix entry or cell is a real number, and represents the expression level (profile) of a gene under an experimental condition.

Clustering techniques can be used to group either the genes under all the different experimental conditions or the experimental conditions based on the expressions of all the genes in the data matrix. However, a cellular process may be active only in a subset of conditions and a single gene may participate in multiple cellular processes (Sara and Oliveira [29]). It is therefore highly desirable to move beyond the clustering paradigm, and to develop approaches capable of discovering local patterns (submatrix) in microarray data (Ben-Dor et al. [4]).

The data will be represented by a $p \times q$ matrix $\mathbf{Y} = (y_{ij})$. In the case of gene expression data, y_{ij} represents the expression level of the gene i under the experimental condition j, $i = 1, \ldots, p$, $j = 1, \ldots, q$. Table 1.1 illustrates a gene expression matrix.

	Condition 1	...	Condition j	...	Condition q
Gene 1	y_{11}	...	y_{1j}	...	y_{1q}
Gene...	...	...	...	...	...
Gene i	y_{i1}	...	y_{ij}	...	y_{iq}
Gene...	...	...	...	...	...
Gene p	y_{p1}	...	y_{pj}	...	y_{pq}

Table 1.1: Gene expression matrix

Consider K submatrices (or clusters) of the data matrix $\mathbf{Y}$. Let $\rho_{ik} = 1$ if row i belongs to the submatrix (or cluster) k, and let it be zero otherwise, $k = 1,...,K$. Similarly, let $\kappa_{jk} = 1$ if column j belongs to submatrix k. and let it be zero otherwise. We will denote by $I_k = \{i, \rho_{ik} = 1\}$ the set of rows in k, and by $J_k = \{j, \kappa_{jk} = 1\}$, the set of columns in k. Their sizes (cardinalities) will be denoted by r_k and c_k, respectively. Let $\bar{y}_{\cdot jk} = \sum_{i\in I_k} y_{ij}/r_k$ be the mean of column j in submatrix k, $\bar{y}_{i\cdot k} = \sum_{j\in J_k} y_{ij}/c_k$, the mean of row i in submatrix k, and $\bar{y}_k = \sum_{(i,j)\in k} y_{ij}/r_k c_k$, the overall mean of cells in submatrix k.

Clusters versus biclusters

A cluster k of rows is defined as a subset of rows that exhibit a similar behavior across all the columns. Thus, a cluster is a $r_k \times q$ submatrix of the data matrix $\mathbf{Y}$. Note that one has in this case $r_k = \sum_{i=1}^{p} \rho_{ik}$. A clustering of rows satisfies the following conditions

$$\sum_{k=1}^{K} \rho_{ik} = 1 \text{ and } \sum_{k=1}^{K} \kappa_{jk} = K \text{ for all } i,\ j, \tag{1.1}$$

since each row must belong to only one cluster, and each cluster must contain all the columns (see Figure 1.1). Similarly, a cluster k of columns is defined as a subset of columns that exhibit a similar behavior across all rows. A column cluster is then a $p \times c_k$ submatrix of the data matrix $\mathbf{Y}$. In this case, one can write $c_k = \sum_{j=1}^{q} \kappa_{jk}$. A clustering of columns satisfies the following conditions

$$\sum_{k=1}^{K} \rho_{ik} = K \text{ and } \sum_{k=1}^{K} \kappa_{jk} = 1 \text{ for all } i,\ j, \tag{1.2}$$

since each cluster must contain all rows, and each column must belong to only one cluster (see Figure 1.2). However, a bicluster k is a subset of rows that exhibit similar behavior across a subset of columns, and conversely (see Figure 1.3). A bicluster is a $r_k \times c_k$ submatrix of the data matrix $\mathbf{Y}$. It satisfies

$$0 \leq \sum_{k=1}^{K} \rho_{ik} \leq K \text{ and } 0 \leq \sum_{k=1}^{K} \kappa_{jk} \leq K \text{ for all } i,\ j, \tag{1.3}$$

since each row (or column) may belong to several biclusters (see Figure 1.3).

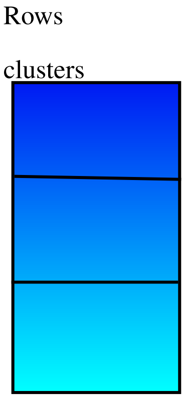


Figure 1.1: Clustering of rows

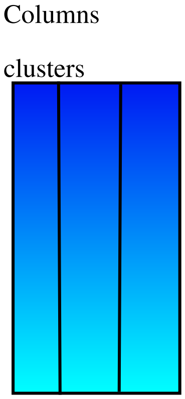


Figure 1.2: Clustering of columns

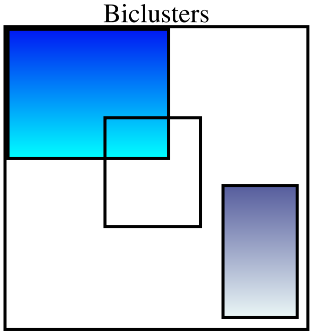


Figure 1.3: Biclustering of the matrix data

The problem of biclustering consists of finding a possibly overlapping partition of blocks (biclusters) of the data matrix. The main unknown parameters of biclustering are, as in the case of clustering, the number of biclusters K, and the row and column membership labels $(\rho,\kappa) = \{(\rho_{ik},\kappa_{jk})\}$, $i = 1,\ldots p$, $j = 1,\ldots,q$, $k = 1,\ldots K$. Note that contrary to clustering, biclustering involves two sets of unknown labels. As in any clustering model, each bicluster k must satisfy a predetermined specific characteristic of homogeneity. Sara and Oliveira [29] gave a somewhat thorough review of popular biclustering techniques. They analyzed and classified a large number of existing approaches according to the type of homogeneity defining biclusters. They identified four types of homogeneity: biclusters with constant values, biclusters with constant values on rows or columns, biclusters with coherent values, and biclusters with coherent evolution. In Section 1.2, we give the definitions and some examples of each type of homogeneity.

Another key difference between clustering and biclustering is the concept of *overlapping* between the clusters (or biclusters). We say that a bicluster k_1 overlaps with another bicluster k_2 if these two biclusters share some rows or some columns of the data matrix. From this definition, we can find in the literature three types of overlapping. The first one is the *row or column overlapping*. Only the rows (or the columns) can belong to more than one bicluster. The second one is the *Row-column overlapping*. Both rows and columns may belong to several biclusters, but a cell in the matrix cannot belong to more than one bicluster. The third and last type of overlapping is the *cell overlapping* which is more general than the others. A cell (a specific row and column) may belong to several biclusters. In Section 1.3 we survey some of the biclustering models and algorithms that have been developed for gene expression analysis for each type of overlapping. Our list of algorithms is not exhaustive, but it rather focuses on what we believe are the more practical methods.

1.2 Types of biclusters

In this section we follow closely the exposition of Sara and Oliveira [29].

1.2.1 Biclusters with constant values

A bicluster with constant values is a submatrix whose cells share a common value. In the case of gene expression data, constant biclusters are subsets of genes with similar expression values within a subset of conditions. A *perfect* constant bicluster verifies $y_{ij} = \mu_k$ for all $(i, j) \in k$. The values y_{ij} found in a constant bicluster can be written as: $y_{ij} = \mu_k + \varepsilon_{ij}$ where ε_{ij} is a noise associated to y_{ij}. Table 1.2 gives an example of this type of bicluster.

12	12	12	12
12	12	12	12
12	12	12	12

Table 1.2: Bicluster with constant values

Hartigan [16] seems to be the first to have applied a clustering method to simultaneously cluster rows and columns. He introduced a partition-based algorithm called *direct clustering* that allows the division of the data in submatrices (biclusters). The quality of a bicluster was evaluated by the sum of squared errors

$$\sum_{(i,j)\in k} (y_{ij} - \bar{y}_k)^2. \tag{1.4}$$

Hartigan's algorithm stops when the data matrix is partitioned into the desired number of biclusters, say K. The quality of the partition is evaluated by the total sum of squared errors

$$SSQ = \sum_{k=1}^{K} \sum_{(i,j)\in k} (y_{ij} - \bar{y}_k)^2.$$

Tibshirani et al. [33] and Cho et al. [10] have also used (1.4) as a measure of biclustering quality to find constant biclusters.

1.2.2 Biclusters with constant values on rows or columns

This type of bicluster exhibits coherent values either on the columns or the rows. The biclusters in Tables 1.3 and 1.4 are examples of perfect biclusters with constant rows and columns, respectively.

12	12	12	12
14	14	14	14
9	9	9	9

Table 1.3: Constant values on rows

12	7	10	11
12	7	10	11
12	7	10	11

Table 1.4: Constant values on columns

A perfect bicluster with constant values on rows is a submatrix k where all the values in the bicluster can be obtained using one of the following expressions:

$$y_{ij} = \mu_k + \alpha_{ik} \qquad \text{or} \qquad y_{ij} = \mu_k \times \alpha_{ik}, \tag{1.5}$$

where μ_k is the typical value in the bicluster, α_{ik} is the adjustment (additive or multiplicative) for row i. A perfect bicluster with constant values on columns is defined similarly. A direct approach to identify this type of bicluster is to first do a normalization on the rows or columns using the mean sample of the rows and of the columns, respectively, and then, apply a method to find biclusters with constant values. Sheng et al. [31] and Segal et al. [30] introduced a probabilistic model to find biclusters with constant values on columns (see Sections 1.3.1 and 1.3.3 for more details).

1.2.3 Biclusters with coherent values

A bicluster with coherent values both on rows and columns is an improvement over the types considered previously. A perfect bicluster k is defined as a subset of rows and a subset of columns verifying for all $(i, j) \in k$:

$$y_{ij} = \mu_k + \alpha_{ik} + \beta_{jk} \qquad \text{or} \qquad y_{ij} = \mu_k \times \alpha_{ik} \times \beta_{jk}, \tag{1.6}$$

where μ_k is the typical value of the bicluster, α_{ik} is the adjustment for row i, and β_{jk} is the adjustment for column j. This type of homogeneity is very common in the literature and many authors (Cheng and Church [9], Lazzeroni and Owen [20], Gu and Liu [14], Zhang [39], Turner et al. [34], Cho et al. [10], Chekouo and Murua [7], Hochreiter et al. [17], Lee et al. [21]) have used it. Note that when an additive model is assumed in a bicluster k, the residual of y_{ij} is $r_{ijk} = y_{ij} - \bar{y}_{i\cdot k} - \bar{y}_{\cdot jk} + \bar{y}_k$ and $r_{ijk} = 0$ if and only if $y_{ij} = \mu_k + \alpha_{ik} + \beta_{jk}$. The particular cases of $\alpha_{ik} = 0$ (or $\alpha_{ik} = 1$) and $\beta_{jk} = 0$ (or $\beta_{jk} = 1$) in the model given by expression (1.6) give the biclusters with constant values on columns and with constant values on rows, respectively. Tables 1.5 and 1.6 illustrate examples of an additive and a multiplicative model, respectively.

12	13	16	11
14	15	18	13
9	10	13	8

Table 1.5: Additive coherent values

12	24	6	18
10	20	5	15
1	2	0.5	1.5

Table 1.6: Multiplicative coherent values

1.2.4 Biclusters with coherent evolution

Ben-Dor et al. [4] have defined a bicluster as a submatrix preserving an order (OPSM). A submatrix preserves an order if there exists a permutation of its columns so that the sequence of values in each row is strictly increasing. An example of this type of bicluster is shown in Table 1.7. In the case of gene expression data, these biclusters correspond to subsets of genes and conditions such that the expression levels of all the genes have a same linear order across the conditions. Ben-Dor et al. [4] defined a complete model of OPSM as being a couple (T,π) where T is a set of s columns and $\pi = (t_1,...,t_s)$ is a linear order on T. In this model, a row i is said to support (T,π) if $\{y_{it_1}, y_{it_2}, \ldots, y_{it_s}\}$ is an increasing sequence. Their algorithm look for a complete maximal set in terms of rows.

12	8	10	9
15	11	14	13
32	7	20	10

Table 1.7: Bicluster with coherent evolution

1.3 Types of overlapping

1.3.1 Row or column overlapping methods

In this section, we will review some methods which look for biclusters with overlapping only between rows, or only between columns. In terms of labels, this type of biclustering can be characterized by $\sum_{k=1}^{K} \rho_{ik} \geq 1$ for row overlapping, or $\sum_{k=1}^{K} \kappa_{jk} \geq 1$

for column overlapping. Figures 1.4 and 1.5 illustrate the row overlapping and the column overlapping models, respectively.

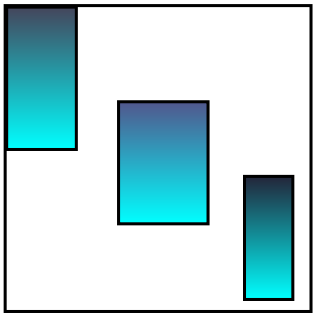


Figure 1.4: Row overlapping model

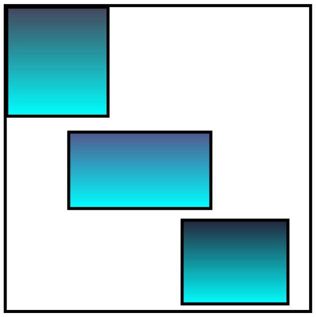


Figure 1.5: Column overlapping model

Tang et al. [32] applied an unsupervised approach for gene expression data analysis called Interrelated Two-Way Clustering (ITWC) to find biclusters with possible row (gene) overlapping. Their goal was to find important gene patterns, and at the same time perform cluster discovery on the experimental conditions. This is equivalent to variable selection (selection of genes) in the context of conditions clustering. There are five steps within each iteration of ITWC. The first step consists of clustering the rows of the matrix into two clusters G_1 and G_2 using k-means. In the second step, based on each gene group $G_i, i = 1,2$, the columns (conditions) are clustered into two clusters $S_{i,1}$ and $S_{i,2}$. The third step combines these clusters to form four groups of columns $C_1 = S_{1,1} \cap S_{2,1}$, $C_2 = S_{1,1} \cap S_{2,2}$, $C_3 = S_{1,2} \cap S_{2,1}$ and $C_4 = S_{1,2} \cap S_{2,2}$. A pair of groups (C_s, C_t) is said to be a heterogeneous pair if the groups do not share columns, i.e., for all $u \in C_s$, $v \in C_t$, if $u \in S_{i,j_1}$ and $v \in S_{i,j_2}$, then $j_1 \neq j_2$. The fourth step of ITWC consists of finding heterogeneous pairs (C_s, C_t), $s,t = 1,\dots,4$. The vector-cosine similarity between two vectors $u = (u_1, ..., u_q)$ and $v = (v_1, ..., v_q)$ is given by:

$$cos(u,v) = \frac{\sum_{j=1}^{q} u_j v_j}{\sqrt{\sum_{j=1}^{q} u_j^2}\sqrt{\sum_{j=1}^{q} v_j^2}}.$$

The fifth step sorts the rows in descending order according to the sum of vector-cosine similarities between each row and the two occupancy patterns associated with each heterogeneous pair. The two occupancy patterns for a heterogeneous pair (C_s, C_t) are obtained by setting all components corresponding to columns in C_r to one, and setting all remaining columns to zero, $r \in \{s,t\}$. The number of rows are reduced by keeping only the first $1/3$ of the sorted rows from each heterogeneous pair. Leave-one-out cross-validation is used to evaluate the prediction performance of the partition so obtained. These five steps are repeated using the selected rows until a predetermined stopping criterion is satisfied. For example, until the occupancy ratio between columns in the heterogeneous groups and all conditions, $(|C_s| + |C_t|)/q$, is maximized.

Gu and Liu [14] develop a Bayesian approach to find biclusters assuming that the only possible overlapping is between the experimental conditions (columns). The priors of the labels ρ and κ are set to respect this restriction. Their model, which is based on the plaid model, may be written as follows

$$y_{ij} = \sum_{k=1}^{K} (\mu_k + \alpha_{ik} + \beta_{jk} + \varepsilon_{ijk})\rho_{ik}\kappa_{jk} + (1 - \sum_{k=1}^{K} \rho_{ik}\kappa_{jk})e_{ij}, \tag{1.7}$$

where ε_{ijk} is the noise term for cluster k, and e_{ij} models the data points that do not belong to any cluster. From this expression, Gui and Liu derive the marginal distribution of $\mathbf{Y}$ given the labels. Inference is based on Markov chain Monte Carlo (MCMC) sampling. The number of biclusters is selected according to the Bayesian information criterion (BIC).

Sheng et al. [31] also develop a biclustering method where the overlapping is only allowed between the columns. Their method proposes a Bayesian framework and works on discrete data. In this model, the columns belonging to any determined bicluster follow independent multinomial distributions. Thus, these biclusters have constant values on columns. Sheng et al. use a Gibbs sampling to sample from the gene/column membership labels. In order to find several biclusters, the authors choose to mask the genes selected in previous biclusters, so as to run again the same algorithm on the remaining data. By doing this, only the columns can be selected more than once.

1.3.2 Row-column overlapping methods

These methods aim at finding biclusters where rows or columns (but not a cell) may belong to more than one bicluster. They may be characterized as satisfying $\sum_{k=1}^{K} \rho_{ik} \kappa_{jk} \leq 1$. They may also satisfy $\sum_{k=1}^{K} \rho_{ik} \geq 1$ or $\sum_{k=1}^{K} \kappa_{jk} \geq 1$, $i = 1, \ldots, p$, $j = 1, \ldots, q$. This type of biclustering may be further characterized by the type of pattern or structure found in the data matrix which may be checkerboard-like or not.

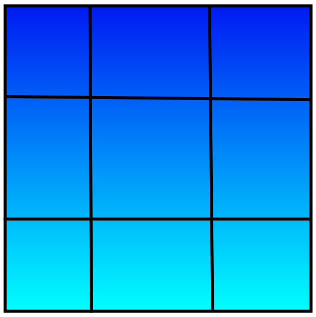


Figure 1.6: Checkerboard structure

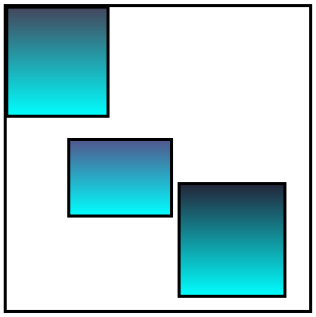


Figure 1.7: Non-checkerboard structure

1.3.2.1 Checkerboard structure

A particular type of row-column overlapping is given by assuming a checkerboard structure in the data matrix. Models requiring this structure allow for the existence of $K = ML$ non-exclusive biclusters, where each row belongs to exactly M biclusters, and each column belongs exactly to L biclusters. Figure 1.6 shows an example of this structure for $M = L = 3$.

Cho et al. [10] have developed an algorithm that simultaneously discovers clusters of rows and columns while monotonically decreasing the corresponding sum of squared residuals. Their optimization problems consists of minimizing the total sum of squared residuals given by

$$\sum_{k=1}^{K} r_k c_k H_k,$$

with $H_k = \sum_{(i,j)\in k} (y_{ij} - \bar{y}_{i\cdot k} - \bar{y}_{\cdot jk} + \bar{y}_k)^2 / r_k c_k$. In what follows, we show that this optimization is equivalent to the application of the hard EM algorithm on an inherently associated statistical model. Given the set of parameters $\theta = (\mu, \alpha, \beta, \sigma^2)$, the underlying model of Cho et al. [10] could be written as

$$P(\mathbf{Y}|\theta, \rho, \kappa) \propto \exp(-\frac{1}{2\sigma^2}\sum_{i,j}(y_{ij} - \sum_{m,l}(\mu_{ml} + \alpha_{iml} + \beta_{jml})\rho_{im}\kappa_{jl})^2),$$

where $\sum_{m=1}^{M} \rho_{im} = \sum_{l=1}^{L} \kappa_{jl} = 1$. The membership labels ρ_{im} and κ_{jl} are the membership labels associated with the clustering of the rows and the columns, respectively. This model assumes the same variance distribution in all K biclusters. If we further assume a uniform distribution as the prior distribution on the labels, i.e.,

$$\begin{aligned} p(\rho_{im} = 1) &= p(\rho_{im} = 1, \rho_{im'} = 0, m' \neq m) = \frac{1}{M} \quad \text{for all } i,\, m \\ p(\kappa_{jl} = 1) &= p(\kappa_{jl} = 1, \kappa_{jl'} = 0, l' \neq l) = \frac{1}{L} \quad \text{for all } j,\, l \end{aligned}$$

then, applying the hard EM algorithm on the complete distribution $P(y, \rho, \kappa|\theta)$ under the usual constraints of identifiability on α and β, yield the following parameter estimates at each EM iteration

$$\hat{\mu}_{ml} = \bar{y}_k, \qquad \hat{\alpha}_{iml} = \bar{y}_{i\cdot k} - \bar{y}_k, \qquad \hat{\beta}_{jml} = \bar{y}_{\cdot jk} - \bar{y}_k, \qquad \text{and } \hat{\sigma}^2 = \frac{1}{pq}\sum_{k=1}^{K} r_k c_k H_k.$$

For the labels, given these estimators, we have

$$\begin{aligned} p_{im} &= p(\rho_{im} = 1, \rho_{im'} = 0, m' \neq m|y, \Theta_{-\rho}) \\ &\propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{j,l}\kappa_{jl}(y_{ij} - (\mu_{ml} + \alpha_{iml} + \beta_{jml}))^2\right\}. \end{aligned}$$

Since $\sum_m p_{im} = 1$, then, maximizing over $\rho_i = (\rho_{im}, m = 1, \ldots M)$ assigns y_i to the cluster m with the highest probability, i.e.,

$$\begin{aligned} \rho_{im} = 1 \text{ if and only if } m &= \arg\max_m p_{im} \\ &= \arg\min_m \sum_l \sum_{j \in J_l, k=(m,l)} (y_{ij} - \bar{y}_{i \cdot k} - \bar{y}_{\cdot jk} + \bar{y}_k)^2. \quad (1.8) \end{aligned}$$

Similarly,

$$\kappa_{jl} = 1 \text{ if and only if } l = \arg\min_l \sum_m \sum_{i \in I_m, k=(m,l)} (y_{ij} - \bar{y}_{i \cdot k} - \bar{y}_{\cdot jk} + \bar{y}_k)^2. \quad (1.9)$$

Relations (1.8) and (1.9) are exactly the same relations that Cho et al. [10] have used to update the labels without explicitly writing a model for the data.

Another work which assumes the checkerboard structure is that of Govaert and Nadif [13]. They refer to their biclustering model as *block clustering*. In contrast to Cho et al. [10], Govaert and Nadif assume that the prior probabilities on the membership labels are also parameters of interest to be estimated. They also used a hard EM algorithm, the Classification EM (CEM) algorithm, to simultaneously cluster the rows and the columns. Their block mixture model is given by

$$P(y|\theta) = \sum_{\rho,\kappa} \prod_{i,m} p_m^{\rho_{im}} \prod_{j,l} q_l^{\kappa_{jl}} \prod_{i,j} \phi_{ij}(y_{ij}; \mu_{lm}),$$

where $\phi_{ij}(y_{ij}; \mu_{lm})$ is a probability density parametrized by μ_{lm}. The parameters p_l and q_m are the probabilities that a row and a column belong to the l-th and m-th component, respectively. The parameter θ in this model is the vector $(p_1, ..., p_M, q_1, ..., q_L, \mu_{11}, ..., \mu_{LM})$.

Kluger et al. [19] introduce a biclustering technique called *spectral biclustering*. It uses a singular value decomposition to identify bicluster structures in the data. This method assumes that the expression matrix $\mathbf{Y}$ has a checkerboard-like structure. By applying the singular value decomposition on $\mathbf{Y}$, one finds the eigenvectors of $\mathbf{Y}\mathbf{Y}^T$ and $\mathbf{Y}^T\mathbf{Y}$. Note that if v is an eigenvector of $\mathbf{Y}\mathbf{Y}^T$, then $\mathbf{Y}^T v$ is an eigenvector of $\mathbf{Y}^T\mathbf{Y}$. For any eigenvector pair $(v, \mathbf{Y}^T v)$, we check whether each of the eigenvectors can be approximated using a piecewise constant vector. This operation allows them to determine whether the data have a checkerboard pattern. For that, the authors use a one-dimensional k-means algorithm to test this fit. Kluger et al.'s assume a multiplicative model, that is, the expression level of a specific gene i in a condition j can be expressed as a product of three independent factors. The first factor is called the *hidden base expression level* E_{ij} (i.e., μ). The entries of E within each block are constant. The second factor represents the genes' expression tendencies across different conditions (α). The last factor represents the role of particular conditions over the genes' expression tendencies (β). The goal is to find the underlying block structure of E. For that purpose, the rows and the columns are first normalized. Let R and C denote the diagonal matrices $R = \mathrm{diag}(\mathbf{Y}1_p)$ where $1_p = (1,..,1) \in \mathbf{R}^p$, and $C = \mathrm{diag}(1_p^T\mathbf{Y})$. The block structure of E is now reflected in the stepwise structure of pairs of eigenvectors with the same eigenvalues of the normalized matrices $M = R^{-1}\mathbf{Y}C^{-1}\mathbf{Y}^T$ and M^T. Theses two eigenvalue problems can be solved through a standard singular value decomposition of $R^{-1/2}\mathbf{Y}C^{-1/2}$.

1.3.2.2 Non-checkerboard structure

Figure 1.7 shows an example of this structure with $K = 3$. The algorithm of Cheng and Church [9], one of the most popular biclustering algorithms, falls in this category. Cheng and Church seem to be the first authors to have introduced the term biclustering in the literature. In their algorithm, the mean squared residual H_k plays a crucial role as a measure of coherence of the rows and columns in a bicluster. Let $\delta > 0$. A sub-matrix k is said to be a δ-bicluster if $H_k < \delta$. Cheng and Church's algorithm aims at finding large and maximal biclusters with scores below a certain predetermined small threshold δ. Cheng and Church suggest a greedy heuristic search so as to rapidly converge to a locally maximal δ-bicluster. A single row or column deletion step iteratively removes the row or column that gives the maximum decrease in H_k. A multiple row or column deletion step follows the same idea, but this time it removes multiple rows or columns in a single iteration. A row or column addition step adds to a given bicluster rows and columns that do not increase the actual score of the bicluster. The general algorithm is composed of a row or column deletion followed by a row or column addition in each iteration. The biclusters are found one at a time. Once a bicluster is found, its rows and columns are masked with uniform random numbers. The process is repeated until K biclusters are found. The masking procedure renders the overlapping between the biclusters unlikely. Cheng and Church justify their algorithm based on the two following assertions.

Assertion 1. *The set of rows that can be completely or partially removed with the net effect of decreasing the score of a bicluster k is:*

$$R_1 = \left\{ i \in I_k, \frac{1}{c_k} \sum_{j \in J_k} (y_{ij} - \bar{y}_{i \cdot k} - \bar{y}_{\cdot jk} + \bar{y}_k)^2 > H_k \right\}. \tag{1.10}$$

Assertion 2. *The set of rows that can be completely or partially added with the net effect of decreasing the score of a bicluster k is:*

$$R_2 = \left\{ i \notin I_k, \frac{1}{c_k} \sum_{j \in J_k} (y_{ij} - \bar{y}_{i \cdot k} - \bar{y}_{\cdot jk} + \bar{y}_k)^2 < H_k \right\}. \tag{1.11}$$

Chekouo and Murua [7] have attempted to mimic Cheng and Church' algorithm within a Bayesian framework. They define the underlying model in Cheng and Church's algorithm as a model similar to the one given by the expression (1.7) of Gu and Liu [14]. However, contrary to (1.7), Chekouo and Murua's model assumes the possibility of having row or column overlapping in the same biclustering. In fact, the labels satisfy $\sum_{k=1}^{K} \rho_{ik} \kappa_{jk} \leq 1$ for all i, j. The prior on the labels is a double-exponential-like distribution with a large inverse scale (penalty) parameter (see Section 1.3.3 below for more details). Chekouo and Murua were successful in showing that both assertions 1 and 2 may be derived as updating proposal movements in a Metropolis-Hastings procedure. Consequently, using these assertions in a MCMC sampler will lead to estimates of the posterior labels.

1.3.3 Cell overlapping methods

In this section, we present biclustering methods that allow general overlapping between biclusters, i.e., where each cell (i, j) of the data matrix may belong to more than one bicluster. These methods may be characterized by the condition $\sum_{k=1}^{K} \rho_{ik} \kappa_{jk} > 1$.

The biclusters in this biclustering are arbitrarily positioned in the matrix. Figure 1.3 shows an example of this type of biclustering.

1.3.3.1 Additive models

One of the most popular models that takes into account this structure is the *plaid model* of Lazzeroni and Owen [20] which is defined by

$$y_{ij} \sim Normal\left(\mu_0 + \sum_{k=1}^{K} (\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk}, \sigma^2\right). \tag{1.12}$$

The general biclustering problem is now formulated as finding parameter values so that the resulting matrix would fit the original data as much as possible. Formally, the problem consists of minimizing

$$\sum_{i=1}^{p}\sum_{j=1}^{q}(y_{ij} - \sum_{k=0}^{K}(\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk})^2 \tag{1.13}$$

under the constraints: $\sum_{i=1}^{p} \alpha_{ik}\rho_{ik} = \sum_{j=1}^{q} \beta_{jk}\kappa_j = 0$ for all k. A layer is a bicluster in the sense of Cheng and Church [9]. A plaid is an ensemble of additive layers. Lazzeroni and Owen propose to minimize (1.13) by using an iterative heuristic algorithm. New layers are added to the model one at a time. To simplify the description of their method, suppose that we already know $K-1$ layers, and that we seek to uncover the K-th layer. Let $\mu_{ijk} = \mu_k + \alpha_{ik} + \beta_{jk}$ and $Z_{ij}^{K} = y_{ij} - \mu_{ij0} - \sum_{k=1}^{K-1} \mu_{ijk}\rho_{ik}\kappa_{jk}$ the residual from the first $K-1$ layers. We need to minimize

$$Q = \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{q}(Z_{ij}^{K} - \mu_{ijK}\rho_{iK}\kappa_{jK})^2, \tag{1.14}$$

subject to the above identifiability constraints on α_{iK} and β_{jK}.

The proposed method to solve (1.14) is again iterative. A relaxation in the parameters is introduced to simplify the optimization problem. The binary latent variables ρ_{ik}, κ_{jk} are replaced by continues ones. The constraints on α_{iK} and β_{jK} are replaced by the soft constraints: $\sum_{i=1}^{p} \alpha_{iK}\rho_{iK}^2 = \sum_{j=1}^{q} \beta_{jK}\kappa_{jK}^2 = 0$.

The hard-EM estimators given in the previous sections are similar to those of Lazzeroni and Owen with no relaxation. With the relaxed parameters, the updates $\bar{\rho}_{iK}$ and $\bar{\kappa}_{jK}$ are given by

$$\begin{aligned}\bar{\rho}_{ik} &= \frac{\sum_j \mu_{ijK}\kappa_{jk}Z_{ij}^K}{\sum_j \mu_{ijK}^2\kappa_{jK}^2}, & (1.15)\\ \bar{\kappa}_{jk} &= \frac{\sum_i \mu_{ijK}\rho_{ik}Z_{ij}^K}{\sum_i \mu_{ijK}^2\rho_{iK}^2}. & (1.16)\end{aligned}$$

The importance of layer k is measured by $\sigma_k^2 = \sum_{i=1}^{p}\sum_{j=1}^{q} \rho_{ik}\kappa_{jk}\mu_{ijk}^2$. A layer is accepted if its importance is significantly larger than what would be found in noise Z_{ij}. For a set of K layers, the algorithm allows to re-estimate all of the μ_{ijk}, by cycling through $k = 1,..,K$ several times. These *backfitting* cycles only conduct a partial re-optimization, since the updating is done only on all the μ_{ijk} parameters, but not on the labels ρ and κ parameters, which are kept as known values after the last layer has been found.

The most successful starting values have been found using a singular value decomposition on Z. The ρ and κ vectors are initialized as the eigenvectors associated with the largest singular values. This choice was motivated by the updating equations for ρ and κ (equations (1.15) and (1.16)) when $\mu_{ijk} = 1$.

Segal et al. [30] also assumed an additive model. This is given by

$$y_{ij} \sim^{i.i.d} Normal\left(\mu_0 + \sum_{k=1}^{K} (\mu_k + \beta_{jk})\rho_{ik}, \sigma_j^2\right). \qquad (1.17)$$

Note that each column belongs to all the biclusters as in clustering. However, this model allows for the overlapping of layers. Contrary to the work of Lazzeroni and Owen [20], the model of Segal et al. [30] does not consider row effects α_{ik}, and the variances may be column-dependent. It is easily shown (see Chekouo and Murua [8]) that this model is similar to the multiplicative mixture model for overlapping clustering (Qiang and Banerjee [24]). The authors referred to this model's biclusters as *processes.* The prior distributions for β_{jk} and ρ_{ik} are assumed to be independent uniforms (over some appropriately bounded range) and binomials, respectively. All the parameters are estimated using a hard EM algorithm as follows:

1. Initialize the labels ρ using a classical method of clustering (k-means for example).
2. (hard E-Step) Repeat (a) and (b) until convergence:
    (a) Find the β_{jk} that maximizes its full conditional distribution.
    (b) Find the ρ_{ik} that maximizes its full conditional distribution.
3. Estimate the parameters $q_{ik} = P(\rho_{ik} = 1)$ (M-step).

Turner et al. [34] propose an improved biclustering of microarray data using the plaid model. They also propose a different algorithm for fitting the plaid model. Their approach uses binary least squares (i.e., hard EM algorithm) to update the cluster membership parameters. This somewhat simplifies the updating of the other parameters. The *backfitting* is done as in the case of Lazzeroni and Owen [20].

Zhang [39] proposes a hierarchical Bayesian version of the plaid model. He provides an empirical Bayes algorithm for sampling the posteriors in two steps. In the first step, he estimates the membership labels by maximizing their marginal posteriors. During the second step, he directly calculates the Bayesian estimates for the other parameters given the values of the membership labels. To improve the overall estimation, he runs *backfitting* to update the parameters $\Theta_k = (\mu_k, \alpha_{ik}, \beta_{jk}, \rho_{ik}, \kappa_{jk}, \sigma^2)$ given Θ_t, $t \neq k$. The backfitting is performed after having done a greedy search for the K layers, as in the case of Lazzeroni and Owen [20].

Chekouo and Murua [7] generalize the plaid model also within a Bayesian framework. They introduce a modified extended version of the Bayesian plaid model. The authors refer to this model as the *penalized plaid model.* It aims at controlling the amount of bicluster overlapping by considering a penalization on the amount of overlapping. This is carried out by the introduction of a so-called penalty parameter which will be denoted by λ. The model fully accounts for a general overlapping structure, as opposed to just a one dimensional (row or column) overlapping as in the model of Gu and Liu [14]. The parameters are determined by an MCMC sampler all at once as opposed to the sequential greedy search algorithm of Zhang [39]. The model also takes into account the problem of identifiability of the row and column effects. As in ANOVA, it assumes that the sum of these effects vanishes within each bicluster. In addition, the penalized plaid model may be seen as a continuous extension of the non-overlapping model of Cheng and Church [9] to the plaid model. Formally, given all the parameters,

$$y_{ij} \sim_{i.i.d} Normal\left(\mu_0\gamma_{ij} + \sum_{k=1}^{K}(\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk}, \sigma^2(\rho_i, \kappa_j)\right), \tag{1.18}$$

where $\gamma_{ij} = \prod_k(1-\rho_{ik}\kappa_{jk})$ is the label associated with the zero-bicluster, i.e., a cluster containing some observations which are not well explained by the main biclusters; and μ_0 is the mean of the zero-bicluster. Note that when $\sum_{k=1}^{K}\rho_{ik}\kappa_{jk} \leq 1$ and $\sigma^2(\rho_i, \kappa_j) = \sigma_k^2$ depends of bicluster k, the model becomes the underlying model of Cheng and Church. But when $\sum_{k=1}^{K}\rho_{ik}\kappa_{jk}$ is allowed to become larger than 1, and $\sigma^2(\rho_i, \kappa_j) = \sigma^2$ is constant, the model becomes the plaid model introduced by Lazzeroni and Owen.

The prior distribution on the labels is defined by a discrete double-exponential with scale parameter λ

$$\pi((\rho,\kappa)|\lambda) \propto \exp\left\{-\lambda\sum_{i,j}\left|1-\gamma_{ij}-\sum_{k=1}^{K}\rho_{ik}\kappa_{jk}\right|\right\}.$$

The scale $\lambda \geq 0$ may be viewed as a penalty parameter that controls the amount of biclustering overlapping. If $\lambda = 0$, the labels are a priori uniformly distributed (e.g., as in the original plaid model). There is no constraint on the structure of the overlapping (i.e., there is cell overlapping). When $\lambda \to \infty$, the model becomes the row-column overlapping model of Cheng and Church. Chekouo and Murua show via a simulation study that the logarithm of the posterior mean of λ decreases nearly linearly with the number of biclusters and the amount of overlapping in them. They argue that the penalty parameter λ may serve as a measure of complexity of the data. In order to choose an appropriate number of biclusters, they suggest using a modified version of the deviance information criterion (DIC). The modified DIC is based on the conditional distribution given the membership labels, and on the maximum a posteriori (MAP) estimators of the parameters. We note that this work appears to be one of the first to have properly addressed the model selection (i.e., the choice of number of biclusters) problem within the context of biclustering. Their work also demonstrates that the use of Bayesian computational techniques such as the Gibbs sampler and Metropolis-Hastings algorithm to estimate the biclustering yield far better results than hard-EM or Iterated Conditional Modes (ICM), and ad hoc heuristic techniques.

1.3.3.2 Informative priors

In another paper, Chekouo and Murua [6] propose a model that takes into account prior information on genes and conditions through pairwise interactions. Their model is a Gaussian plaid model for biclustering combined with a discrete Gibbs field or autologistic distribution (Besag [5], Winkler [36]) that conveys the prior information. The Gibbs field prior is a model for dedicated relational graphs, one for the genes and another for the conditions, whose nodes correspond to genes (or conditions) and edges to gene (or condition) similarities. Each relational graph is provided with a neighborhood structure. The notation $i \sim i'$ will denote that nodes i and i' in the graph are connected with a graph edge, i.e., the relation $i \sim i'$ is satisfied if and only if i and i' are neighbors. Each edge is assigned a weight. For the gene graph, the weights are given by

$$B_{ii'}(T^{\rho}, \sigma_{\rho}) = \frac{1}{T^{\rho}} \exp\left(-\frac{1}{2\sigma_{\rho}^{2}} d^{\rho}(i,i')^{2} \right).$$

where T^{ρ} and σ_{ρ} are the temperature and kernel bandwidth parameters of the graph, respectively. The “distances” $d^{\rho}(i,i')$ are induced by genes similarities based on the entropy information (Resnik [26]) extracted from GO (Gene Ontology) annotations (Ashburner et al. [1]). The prior for the gene labels ρ_k is given by the binary Gibbs random field

$$p(\rho_k|a,T^\rho,\sigma_\rho^2) \propto h_{\rho,k}(\rho_k,T^\rho) \doteq \exp\left\{\sum_{i=1}^{p} a_i\rho_{ik} + \sum_{i\sim i'} B_{ii'}(T^\rho,\sigma_\rho^2)\,\mathbf{1}_{\{\rho_{ik}=\rho_{i'k}\}}\right\}$$

where $a = \{a_i\}_{i=1}^{p}$ is a set of hyper-parameters controlling the amount of membership ($\rho_{ik} = 1$) in the biclusters, and for every relation A, $\mathbf{1}_A$ denotes the indicator function associated with the set of elements satisfying relation A. The Gibbs prior favors biclusters formed by similar genes in the sense of the distances or similarities built in the relational graph. The incorporation of a complex data prior in the model poses many computational challenges. Chekouo and Murua develop a special version of the Wang-Landau algorithm (Atchadé and Liu [2], Wang and Landau [35]) to bypass the intractability of the normalizing constants in the prior distributions.

1.3.3.3 Multiplicative models

Hochreiter et al. [17] propose a novel generative approach for biclustering called FABIA (Factor Analysis for Bicluster Acquisition). Their model is based on a multiplicative model inspired by factor analysis with K factors. It takes into account the linear dependencies between gene expressions and conditions. Formally, the matrix data $\mathbf{Y}$ is written as a sum of multiplicative biclustering models

$$\mathbf{Y} = \sum_{k=1}^{K} \alpha_k\beta_k^T + \varepsilon,$$

where $\varepsilon = (\varepsilon_{ij})_{i=1,j=1}^{p,q}$ is the additive noise, $\alpha_k \in \mathbf{R}^p$ and $\beta_k \in \mathbf{R}^q$. The vector α_k corresponds to a column vector that contains zeros for genes not participating in the bicluster. The vector β_k is a vector of factors that contains zeros for conditions not included in the bicluster. Hochreiter et al. [17] try to find a biclustering by estimating sparse vectors α_k and β_k. Hence, the problem is to identify the zero components of these vectors. Hochreiter et al. also embed their model into a Bayesian model. The prior distributions on α_k and β_k are set to component-wise independent Laplace distributions. The variational EM algorithm for sparse factors introduced by Girolami [12] is used to estimate the parameters. The members of the k-th bicluster are obtained by thresholding the components of α_k and β_k.

An alternative method to find the biclusters using a multiplicative model is the Sparse Singular Value Decomposition (SSVD) approach of Lee et al. [21]. The singular value decomposition (SVD) of y can be written as

$$\mathbf{Y} = \alpha\mu\beta^T = \sum_{k=1}^{r} \mu_k \alpha_k \beta_k^T$$

where r is the rank of $\mathbf{Y}$, $\alpha = (\alpha_1|\cdots|\alpha_r)$ is a matrix of orthonormal left singular vectors, $\beta = (\beta_1|\cdots|\beta_r)$ is a matrix of orthonormal right singular vectors, and $\mu = \text{diag}(\mu_1,...,\mu_r)$ is a diagonal matrix with positive singular values $\mu_1 \geq ... \geq \mu_r$. SVD decomposes $\mathbf{Y}$ into a sum of rank-one matrices $\mu_k\alpha_k\beta_k^T$ referred to as SVD *layers* by the authors. Taking the first $K \leq r$ rank-one matrices, one obtains the following approximation of $\mathbf{Y}$:

$$\mathbf{Y} \approx \mathbf{Y}(K) = \sum_{k=1}^{K} \mu_k \alpha_k \beta_k^T.$$

Note that $\mathbf{Y}(K) = \arg\max_{y^\star \in \mathscr{A}_K} \parallel y - y^\star \parallel^2$, where $\mathscr{A}_K$ is the set of all the $p \times q$ matrices of rank K. The biclustering problem of Lee et al. [21] is similar to that of Hochreiter et al. [17]. It consists of seeking a low-rank matrix approximation to $\mathbf{Y}$ under the assumption that the vectors α_k and β_k are sparse (i.e., they contain many zeros). In order to obtain sparsity, contrary to the Bayesian approach of Hochreiter et al. [17], Lee et al. [21] simply impose sparsity penalties on both α and β. Thus, for a given k, bicluster k (or SVD *layer k*) consists of the genes (rows) associated with the non-zero components of α_k, and the conditions (columns) associated with the non-zero components of β_k.

1.4 Choosing the number of biclusters

In the literature, there are mainly three ways to estimate the number of biclusters K. The first and easier way to choose K is to fix it a priori. The biclustering algorithms are run until the K biclusters have been identified. Many authors such as Cheng and Church

[9], Turner et al. [34] and Lee et al. [21] use this strategy to sequentially discover one bicluster at a time. The second way to estimate K is to perform a greedy search to find the biclusters. The maximum number K_{max} of biclusters allowed in the search is fixed a priori. A stopping rule for the algorithm must also be specified. Lazzeroni and Owen (2002) determined the biclusters sequentially, i.e., one at a time. Their stopping rule is based on what they refer to as a measure of the importance of a bicluster. For bicluster k, this is defined by $\sigma_k^2 = \sum_{i=1}^{n} \sum_{j=1}^{p} \rho_{ik} \kappa_{jk} (\mu_{ijk})^2$. A new bicluster k is accepted if its importance is significantly larger than what is expected from noise. Consider again the residual matrix at stage k, given by $Z^{(k)} = (Z_{ij})$ where each $Z_{ij} = y_{ij} - \sum_{k'=1}^{k-1} \mu_{ijk'} \rho_{ik'} \kappa_{jk'}$. In order to evaluate σ_k^2 on noise, Lazzeroni and Owen consider estimating its distribution by resampling techniques. They randomly permute every row and every column of $Z^{(k)}$ a pre-defined number of times, say R, and obtain the matrices $Z_r^{(k)}$ for $r = 1, ..., R$. The next bicluster k is then estimated from each of these random matrices. Associated with every bicluster k so found from $Z_r^{(k)}$ there is an importance $\sigma_{k,r}^2$. If $\sigma_k^2 > \max_r \sigma_{k,r}^2$ and $k < K_{max}$, then the new bicluster k is added to the model; otherwise the algorithm is stopped and only $k-1$ biclusters are reported.

Zhang (2010) also find the biclusters one at the time. He estimates the bicluster memberships by maximizing their marginal posteriors. As Lazzeroni and Owen, Zhang also uses a gready search to estimate the biclustering. But he notes that a permutation of rows and columns of the matrix $\mathbf{Y}$ may no longer have a similar correlation structure to that of the original $\mathbf{Y}$. So he proposes a different stopping rule. The algorithm is run until K_{max} biclusters have been found. He computes $\sigma_{min}^2 = \min_k(\sigma_k^2)$, and considers σ_{min}^2 as an initial estimate of the background noise level. Then for a pre-determined constant $t_b > 0$, he computes the average σ_b^2 of all σ_k^2s verifying $\sigma_k^2 < t_b \sigma_{min}^2$. The background noise level is then updated to σ_b^2. Zhang uses the ratio σ_k^2 / σ_b^2 to test the hypothesis that bicluster k is just noise. The distribution of this ratio under the null hypothesis, and for large pq, is approximated by a χ_1^2 distribution. Finally, bicluster k is kept if and only if $\sigma_k^2 / \sigma_b^2 > t_c$, where t_c is the critical value of the test.

The third and last way to choose the number of biclusters K is by model selection. This is based on well-known criteria such as the Bayesian information criterion (BIC), the Akaike information criterion (AIC), or the deviance information criterion (DIC). In general, these criteria are used when the algorithms estimate all biclusters at the same time, as opposed to sequentially. Gu and Liu [14] use *BIC* to select the number of biclusters. Chekouo and Murua [7] note that BIC does not seem very suitable for the biclustering problem, and instead they propose using AIC or DIC to select an appropriate number of biclusters.

1.5 Comparison and validation of the biclustering algorithms

Several comparisons of biclustering methods have been proposed in the literature. In clustering, there are mainly two categories of indices to validate and compare the clustering results: internal and external indices. We can also find these categories within the context of biclustering (Santamaria et al., 2007). Internal indices are based only on the information intrinsic to the biclustering model, and not on exogenous prior information on the data. Among these indices, we have the well-known Bayesian information criterion (BIC), the Akaike information criterion (AIC), and also the Calinski-Harabasz index, and the Davies-Bouldin index, just to mention a few. The reader is referred to Rendón et al. [25] for a review on internal indices. In general, internal indices vary from one biclustering model to another, and consequently, often they are not suitable to make comparisons between models or algorithms. This is why we only focus on external indices in this section. Basically, there are two types of external indices: biological and non-biologica indices.

1.5.1 Biological external indices

We can use biological knowledge to validate the estimated biclusters. Many authors such as Lazzeroni and Owen [20], Zhang [39], Saez et al. [27], Prelić et al. [23], Eren et al. [11] and Chekouo and Murua [7] use the enrichment of the genes in their biclusters to validate their results, and also to compare their results to those yielded by other alternative algorithms. Usually, gene annotations from GO (Gene Ontology) or KEGG (Kyoto Encyclopedia of Genes and Genomes) are used to compute enrichment.

Gene Ontology [1] provides a controlled vocabulary (GO terms and annotations) to describe gene products characteristics and properties. Each gene is characterized by some GO terms, which in turn, are characterized by three biological functions referred to as molecular function, cellular components, and biological processes. In practice, we get all the GO terms for any of the genes in a given bicluster and determine if each term is relevant (i.e., over-represented). The common statistic used to test this hypothesis is based on the Fisher exact test. Assume that a population made of $M+N$ genes contains exactly M genes annotated to a term, say $GO1$. Suppose that a given bicluster contains k genes, r of which are annotated to the term $GO1$. In order to find out if the $GO1$ term is over-represented in the bicluster, one can perform a Fisher exact test. Its p-value is calculated as the probability that a random bicluster of (the same) size k contains at least r genes annotated to $GO1$. The hypergeometric distribution is used to compute this p-value:

$$p\text{-value} = \sum_{x=r}^{k} \frac{\binom{M}{r}\binom{N}{k-r}}{\binom{N+M}{k}}. \tag{1.19}$$

The R package *GOstat* [3] uses a χ^2 test to approximate this p-value when the expected value for any count is above five.

Another popular biological knowledge database is the KEGG database [18]. This provides the biological pathways where each gene belongs. The same procedure as in GO is also used to test if a KEGG pathway in a given bicluster is over-represented. The R Bioconductor package *clusterProfiler* [38] can be used to detect the over-represented genes.

1.5.2 Non-biological external indices

Non-biological external indices are very popular to validate and compare among biclusterings. In clustering, there are many known external indices [15] such as the Rand Index, the Adjusted Rand Index, the Jaccard index, the Folkes and Mallows index, and the Huberts statistic. These indices assume that the data are divided in a partition, that is, the data can be decomposed in disjoint sets (e.g., the clusters). This is not the case in biclustering, since biclusters are allowed to overlap. Consequently, Turner et al. [34]

adapted the F_1 index, a common measure used in text-mining, to measure the similarity between biclusterings. We note that the F_1 index has previously been suggested by Murua et al. [22] within the context of clustering as well. They show that within the context of clustering, the F_1 index behaves similarly to the Adjusted Rand Index. Let B be a bicluster, r_B be the number of genes in B, c_B be the number of conditions in B and $n_B = r_B c_B$ be the number of elements in B. Suppose that we wish to compare a target bicluster A and a known bicluster B. Consider the following two measures of similarity between A and B:

$$\begin{aligned} \text{recall} &= \frac{(r_{A\cap B})(c_{A\cap B})}{n_B}, \\ \text{precision} &= \frac{(r_{A\cap B})(c_{A\cap B})}{n_A}. \end{aligned}$$

Recall measures the proportion of elements in B that belong to A and precision measures the proportion of elements in A captured in B. Turner et al. [34] refer to precision as *specificity* and to recall as *sensitivity*. The F_1 measure is defined as

$$F_1(A,B) = 2(r_{A\cap B}) \times (c_{A\cap B})/(n_A + n_B).$$

When several biclusters are to be compared, one may use an F_1-average based index (Chekouo and Murua [7], Prelić et al. [23], Santamaría et al. [28]). Let $M_1 = \{A_1, \ldots, A_k\}$ be the set of estimated (target) biclusters, and $M_2 = \{B_1, \ldots, B_\ell\}$, the set of true (known) biclusters. The similarity of the estimate M_1 to the true biclustering M_2 can be measured by

$$S(M_1,M_2) = \frac{1}{k}\sum_{i=1}^{k} \max_j F_1(A_i,B_j) \text{ or } S(M_2,M_1) = \frac{1}{\ell}\sum_{i=1}^{\ell} \max_j F_1(A_j,B_i).$$

Note that $S(M_1,M_2) \leq 1$, and it is equal to 1 if $M_1 = M_2$.

1.6 Conclusion

In this article, we reviewed some practical biclustering algorithms according to the type of the homogeneity in the biclusters and the type of overlapping in the biclustering.

It turns out that most of the algorithms try to find biclusters with coherent values on the rows and columns, and biclusterings with cell-overlapping. The number of biclusters can be fixed, determined by a stopping rule, or by a model selection criterion. Also, the F_1-measure seems suitable to evaluate and compare the biclustering algorithms. The GO and KEGG biological knowledge databases may be used to give a biological interpretation of the biclustering results in terms of gene enrichment analysis.

BIBLIOGRAPHIE

[1] Ashburner, M., C. Ball, J. Blake, D. Bolsteing, H. Butler, J. Cherry, A. Davis, K. Dolinski, S. Dwight, J. Eppig, M. Harris, D. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. Matese, J. Richardson, M. Ringwald, G. Rubin, and G. Sherlock (2000). Geneontology : tool for the unification of biology the gene ontology consortium. *Nature Genetics 25*, 25–29.

[2] Atchadé, Y. F. and J. S. Liu (2010). The Wang-Landau algorithm in general state spaces : Applications and convergence analysis. *Statistica Sinica 20*(1), 1–26.

[3] Beissbarth, T. and T. P. Speed (2004). GOstat : find statistically overrepresented gene ontologies within a group of genes. *Bioinformatics 20*, 1464–1465.

[4] Ben-Dor, A., B. Chor, R. M. Karp, and Z. Yakhini (2003). Discovering local structure in gene expression data : The order-preserving submatrix problem. *Journal of Computational Biology 10*(3/4), 373–384.

[5] Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological) 36*(2), 192–236.

[6] Chekouo, T. and A. Murua (2012a). The Gibbs-plaid biclustering model. Technical Report, Université de Montréal.

[7] Chekouo, T. and A. Murua (2012b). The penalized biclustering model and related algorithms. Submitted for publication.

[8] Chekouo, T. and A. Murua (2012c). Variable selection with the plaid mixture model for clustering. Technical Report, Université de Montréal.

[9] Cheng, Y. and G. Church (2000). Biclustering of expression data. *Int. Conf. Intelligent Systems for Molecular Biology 12*, 61–86.

[10] Cho, H., I. S. Dhillon, Y. Guan, and S. Sra (2004). Minimum sum-squared residue co-clustering of gene expression data. In *SDM*.

[11] Eren, K., M. Deveci, O. Küçüktunç, and U. V. Çatalyürek (2012). A comparative analysis of biclustering algorithms for gene expression data. *Briefings in Bioinformatics*, 1–14.

[12] Girolami, M. (2001). A variational method for learning sparse and overcomplete representations. *Neural Comput. 13*(11), 2517–2532.

[13] Govaert, G. and M. Nadif (2003). Clustering with block mixture models. *Pattern Recognition 36*(2), 463 – 473.

[14] Gu, J. and S. Liu (2008). Bayesian biclustering of gene expression data. *The Int. Conf. on Bioinformatics & Computational Biology, BMC Genomics 9*(1), 113–120.

[15] Halkidi, M., Y. Batistakis, and M. Vazirgiannis (2001, December). On clustering validation techniques. *J. Intell. Inf. Syst. 17*(2-3), 107–145.

[16] Hartigan, J. A. (1972). Direct clustering of a data matrix. *Journal of the American Statistical Association 67*(337), 123–129.

[17] Hochreiter, S., U. Bodenhofer, M. Heusel, A. Mayr, A. Mitterecker, A. Kasim, T. Khamiakova, S. Van Sanden, D. Lin, W. Talloen, L. Bijnens, H. W. H. Gohlmann, Z. Shkedy, and D.-A. Clevert (2010). FABIA : Factor analysis for bicluster acquisition. *Bioinformatics 26*(12), 1520–1527.

[18] Kanehisa, M. and S. Goto (2000). KEGG : Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research 28*, 27–30.

[19] Kluger, Y., R. Basri, J. T. Chang, and M. Gerstein (2003). Spectral biclustering of microarray cancer data : Co-clustering genes and conditions. *Genome Research 13*, 703–716.

[20] Lazzeroni, L. and A. Owen (2002). Plaid models for gene expression data. *Statistica Sinica 12*, 61–86.

[21] Lee, M., H. Shen, J. Huang, and J. Marron (2010). Biclustering via sparse singular value decomposition. *Biometrics 66*(4), 1087–1095.

[22] Murua, A., W. Stuetzle, J. Tantrum, and S. Sieberts (2008). Model based document classification and clustering. *International Journal of Tomography and Statistics 8*(W08), 1–25.

[23] Prelić, A., S. Bleuler, P. Zimmermann, A. Wille, P. Bühlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler (2006, May). A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics 22*(9), 1122–1129.

[24] Qiang, F. and A. Banerjee (2008). Multiplicative mixture models for overlapping clustering. In *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*, pp. 791 –796.

[25] Rendón, E., I. M. Abundez, C. Gutierrez, S. D. Zagal, A. Arizmendi, E. M. Quiroz, and H. E. Arzate (2011). A comparison of internal and external cluster validation indexes. In *Proceedings of the 2011 American conference on applied mathematics and the 5th WSEAS international conference on Computer engineering and applications*, pp. 158–163.

[26] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *In Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 448–453.

[27] Saez, P. C., R. P. Marqui, F. Tirado, J. Carazo, and A. P. Montano (2006). Biclustering of gene expression data by non-smooth non-negative matrix factorization. *BMC Bioinformatics 7*(1), 78+.

[28] Santamaría, R., L. Quintales, and R. Therón (2007). Methods to bicluster validation and comparison in microarray data. In *Proceedings of the 8th international conference on Intelligent data engineering and automated learning*, pp. 780–789.

[29] Sara, C. M. and A. L. Oliveira (2004). Biclustering algorithms for biological data analysis : A survey. *IEEE Transactions on computational biology and bioinformatics 1*(1), 24–45.

[30] Segal, E., A. Battle, and D. Koller (2003). Decomposing gene expression into cellular processes. In *Pacific Symposium on Biocomputing*, pp. 89–100.

[31] Sheng, Q., Y. Moreau, and B. D. Moor (2003). Biclustering microarray data by Gibbs sampling. *Bioinformatics 19*, 196–205.

[32] Tang, C., L. Zhang, A. Zhang, and M. Ramanathan (2001, nov). Interrelated two-way clustering : an unsupervised approach for gene expression data analysis. In *Bioinformatics and Bioengineering Conference, 2001. Proceedings of the IEEE 2nd International Symposium on*, pp. 41–48.

[33] Tibshirani, R., T. Hastie, M. Eisen, D. Ross, D. Botstein, and P. Brown (1999). Clustering methods for the analysis of dna microarray data. Technical report, Dept. of Health Research and Policy, Dept. of Genetics, and Dept. of Biochemestry, Stanford Univ.

[34] Turner, H., T. Bailey, and W. Krzanowski (2004). Improved biclustering of microarray data demonstrated through systematic performance tests. *Computational Statistics & Data Analysis 48*, 235–254.

[35] Wang, F. and D. P. Landau (2001). Efficient, multiple-range random walk algorithm to calculate the density of states. *Physical Review Letters 86*, 2050–2053.

[36] Winkler, G. (2003). *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*, Volume 27. Springer.

[37] Wu, F.-X., W.-J. Zhang, and A. J. Kusalik (2004). Modeling gene expression from microarray expression data with state-space equations. In *Pacific Symposium on Biocomputing'04*, pp. 581–592.

[38] Yu, G., L.-G. G. Wang, Y. Han, and Q.-Y. Y. He (2012, May). clusterprofiler : an R package for comparing biological themes among gene clusters. *Omics : a journal of integrative biology 16*(5), 284–287.

[39] Zhang, J. (2010). A Bayesian model for biclustering with applications. *Journal of the Royal Statistical Society : Series C (Applied Statistics) 59*(4), 635–656.

CHAPITRE 2

THE PENALIZED BICLUSTERING MODEL AND RELATED ALGORITHMS

Abstract

Biclustering is the simultaneous clustering of two related dimensions, for example, of individuals and features, or genes and experimental conditions. Very few statistical models for biclustering have been proposed in the literature. Instead, most of the research has focused on algorithms to find biclusters. The models underlying them have not received much attention. Hence, very little is known about the adequacy and limitations of the models and the efficiency of the algorithms. In this work we shed light on associated statistical models behind the algorithms. This allows us to generalize most of the known popular biclustering techniques, and to justify, and many times improve on, the algorithms used to find the biclusters. It turns out that most of the known techniques have a hidden Bayesian flavour. Therefore, we adopt a Bayesian framework to model biclustering. We propose a measure of biclustering complexity (number of biclusters and overlapping) through a penalized plaid model, and present a suitable version of the DIC criterion to choose the number of biclusters, a problem that has not been adequately addressed yet. Our ideas are motivated by the analysis of gene expression data.

Key words: Clustering, deviance information criterion, gene expression, mixture, model selection, plaid model.

2.1 Introduction

The term biclustering seems to have been first introduced by Cheng and Church [9]. It refers to the simultaneous clustering of the individuals of a population and the features defining the individuals. In general, this is better understood if one thinks of a data matrix where the individuals correspond to the rows and the features to the columns. Biclustering is clustering done simultaneously in two dimensions (rows and columns). A bicluster is a sub-matrix of the data matrix where the rows exhibit a similar pattern across the columns, and the columns exhibit a similar pattern across the rows. Biclustering techniques have important applications in Bioinformatics (Tanay et al. [32]), Marketing (Dolnicar et al. [12]), and text mining (Busygin et al. [6]), to mention a few fields. For example, in Bioinformatics, they are usually applied to gene expression data. These are matrices of gene expression levels (rows) obtained under different experimental conditions (columns). Clustering methods such as hierarchical clustering (Sokal and Michener [30]) or k-means (Ward [34]) will group genes (or conditions) into subsets that convey biological significance. A gene can belong to only one cluster. All genes in a cluster must present similar co-regulation patterns across all conditions. However, a gene (or a condition) may participate in multiple biological pathways that could or not be co-active under all conditions. That is why biclustering is very relevant in the context of gene expression data. Genes in a bicluster could belong to other biclusters and could be co-regulated only in a subset of conditions.

Let $Y = (y_{ij})$ be the data matrix, with p cases (the rows) and q conditions (the columns). In practice, in bioinformatics applications, we think of the cases as genes, and of the conditions as different experimental conditions imposed on the genes. The biclustering literature have distinguished two main types of biclusters: additive or multiplicative ones (Sara and Oliveira [27]). In mathematical terms, the additive biclusters are given by $E(y_{ij}|\text{cell } (i,j) \in k) = \mu_k + \alpha_{ik} + \beta_{jk}$, where k denotes the bicluster, $E(\cdot|\cdot \in k)$ is the conditional expectation given that the cell is in bicluster k, μ_k is the overall mean of the objects in the bicluster, α_{ik} is the effect of the i-th case on bicluster k, and β_{jk} denotes the effect of the j-th condition on bicluster k. In this case, for identifiability purposes, we need to impose the constraints $\sum_{i\in k} \alpha_{ik} = \sum_{j\in k} \beta_{jk} = 0$. The multiplicative bicluster (Hochreiter et al. [18], Kluger et al. [22]) is given by $E(y_{ij}|\text{cell } (i,j) \in k) = \mu_k \alpha_{ik} \beta_{jk}$, where the meaning of the parameters is as in the additive model, but now the effects are multiplicative. We can impose the constraints $\prod_{i\in k} \alpha_{ik} = \prod_{j\in k} \beta_{jk} = 1$, or $\sum_{i\in k} \alpha_{ik} = |i \in k|$, and $\sum_{j\in k} \beta_{jk} = |j \in k|$. Here and in what follows, for any discrete set A, $|A|$ denotes the number of elements of A. The first constraints apparently make the additive and multiplicative models equivalent (the additive model is apparently the logarithm of the multiplicative one). However, this is only true if the errors in the multiplicative model are also multiplicative. But, most of the multiplicative models reported in the literature assume additive errors. In this work, we will restrict our attention to additive models because they are the most used in applications.

We note that most known biclustering techniques only give explicit assumptions on the mean of the observations. The implicit assumptions of the distribution of the observations need to be derived from the algorithms used to find the biclusters. However, it is easy to see that in all of them the errors may be assumed independent identically distributed Gaussian random variables. The variance is either common to all biclusters or bicluster dependent. Also note that the observed mean is modeled conditionally on knowing the bicluster membership. Let $\rho_{ik} = 1$ iff the i-th case is in bicluster k, and let it be zero otherwise. Similarly, let $\kappa_{jk} = 1$ iff the j-th condition is in bicluster k, and let it be zero otherwise. The bicluster problem consists of estimating the number of biclusters K and the labels $(\rho,\kappa) = \{(\rho_{ik},\kappa_{jk})\}$ for $i = 1,\dots p$, $j = 1,\dots,q$, and $k = 1,\dots K\}$. We will also use the notation $\rho_i = (\rho_{ik})$ and $\kappa_j = (\kappa_{jk})$. Most of the techniques suggested in the literature assume that K is known. This is usually a sufficiently large number (Cheng and Church [9], Lazzeroni and Owen [23], Turner et al. [33], Zhang [35]). However, many algorithms are sequential in the sense that they uncover one bicluster at a time. In this case, K is determined sequentially according to some stopping criterion. The sequential search for K is somewhat preferred since it apparently helps to discover large biclusters.

Biclustering has not received much attention from the statistical community. Many of the algorithms proposed in the literature have no probabilistic or statistical foundation (for a good survey on the topic see for example Sara and Oliveira [27]). Hence, little is known about the adequacy and limitations of the models and the efficiency of the algorithms. Very few models for biclustering have been proposed. One of the most popular is the plaid model of Lazzeroni and Owen [23]. In this model, the expectation of y_{ij} given the overall biclustering membership is written as a sum of layers, plaids, or biclusters (see equation (2.1)). The parameters are estimated by least squares. In order to facilitate their estimation, the labels are relaxed and assumed continuous. Based on the same model, Turner, Bailey and Krzanowsk [33] proposed a constrained estimation of the labels by binary least squares. Gu and Liu [16] generalized the plaid model by introducing bicluster dependent variances in a Bayesian framework. However, their model constraints the overlapping structure of the biclusters to one dimension. That is, the overlapping consists of either only rows or only columns. Caldas and Kaski [7] also extended the plaid model within a Bayesian framework. A drawback of the estimation algorithms presented in Gu and Liu [16] and Caldas and Kaski [7] is that their Gibbs samplers require either the inversion of potentially (depending on the dimension of the data) high-dimensional matrices and/or the computation of products of potentially high-dimensional matrices. These will require huge computational costs and may be intractable in practice when faced with high-dimensional data sets. Zhang [35] tried to overcome these limitations also within a Bayesian framework, by estimating the number of biclusters K and the bicluster parameters sequentially, that is, one bicluster at the time, using an ICM-type (Iterated Conditional Modes) algorithm (Besag [4], Lindley and Smith [24]).

In this work we shed light on the actual statistical models behind the algorithms. This allows us to generalize most of the known popular biclustering techniques, and to justify, and many times improve on, the algorithms used to find the biclusters. It turns out that most of the known techniques have a hidden Bayesian flavour. Therefore, we proposed a Bayesian framework to model biclusters. We show that algorithms such as the one of Lazzeroni and Owen [23] and Cheng and Church [9] can be justified as applications of ICM (Besag [4], Lindley and Smith [24]), or can be embedded into a Metropolis-Hastings paradigm. We introduce a modified extended version of the Bayesian plaid model, *the penalized plaid model,* that aims at controlling the amount of bicluster overlapping in the fitting. Our model fully accounts for a general overlapping structure as opposed to just one dimensional overlapping as in the model of Gu and Liu [16]. The parameters are determined all at once by a dedicated MCMC as opposed to the sequential algorithm of Zhang [35]. Our model also takes into account the problem of identifiability of the row and column effects. Inspired by the ANOVA model, it assumes that the sum of these effects vanishes within each bicluster. In addition, the penalized plaid model may be seen as a continuous extension of the non-overlapping model of Cheng and Church [9] to the plaid model. We show that the penalty parameter of our model may serve as a measure of complexity of the data: the more biclusters and the more overlapping, the smaller the penalty parameter. We also show that biclustering may be seen as a mixture model. However, we show that techniques such as the EM algorithm (Dempster et al. [11]) that are commonly used for mixture models may not be appropriate. We think that this might be the reason why hard-EM and ICM-like algorithms have been preferred in the literature. Through a simulation study, we show that Bayesian computational techniques such as the Gibbs sampler (Geman and Geman [15]) or Metropolis-Hastings (Hastings [17]) yield far better results. We also introduce a criterion to choose the number of biclusters, a problem that has not yet been addressed properly in the literature. Our criterion is a modified version of the Deviance Information Criterion (DIC) [31] based on the marginal (or conditional) distribution over the labels, and on Maximum A Posteriori (MAP) estimates of the model parameters.

The paper is organized as follows. Biclustering as a mixture model is treated in Section 2.2. The EM, hard-EM and ICM algorithms are described in Section 2.3. In Section 2.4, we describe a Bayesian framework for biclustering and introduce our penalized plaid model. Our experiments through simulations are shown in Section 3.4. An application of our penalized plaid model methodology to elucidate the biclustering structure of the gene expression data associated with the yeast cell cycle data (Eisen et al. [13]) is described in Section 2.6. We end up our exposition with some conclusions in Section 3.6.

2.2 Biclusters are mixtures

Using the labels, the additive model becomes

$$E(y_{ij}|\rho,\kappa) = \sum_k \rho_{ik}\kappa_{jk}(\mu_k + \alpha_{ik} + \beta_{jk}). \tag{2.1}$$

Let z_{ijk} be independent Gaussian random variables with corresponding means $\mu_k + \alpha_{ik} + \beta_{jk}$ and variances σ_k^2. It is useful to also consider the special constant mean bicluster, the 0-bicluster component, $z_{ij0} \sim \mathbb{N}(\mu_0, \sigma_0^2)$. This is used to model data cells that do not belong to any other bicluster. The 0-bicluster is a special cluster in the model in that we always assume that the event $\{\rho_{i0}\kappa_{j0} = 1\}$ occurs if and only if the event $\{\sum_{k=1}^K \rho_{ik}\kappa_{jk} = 0\}$ occurs. This event is better expressed mathematically by the indicator function $\gamma_{ij} = \prod_{k=1}^K (1 - \rho_{ik}\kappa_{jk})$. We may like to write

$$y_{ij} = \sum_{k=0}^{K} \rho_{ik}\kappa_{jk}\, z_{ijk}. \tag{2.2}$$

Several models can be derived from this simple expression. Note that model (2.2) corresponds to a Gaussian mixture model for the marginals of y_{ij}. Indeed, it is straightforward to see that these marginals are Gaussian-mixtures with 2^K components

$$\frac{1}{\sigma_0}\phi\left(\frac{y_{ij} - \mu_0}{\sigma_0}\right) p(\gamma_{ij} = 1) + \sum_{\substack{\rho_i, \kappa_j \\ \gamma_{ij}=0}} \frac{1}{\sigma(\rho_i, \kappa_j)}\phi\left(\frac{y_{ij} - \mu(\rho_i, \kappa_j)}{\sigma(\rho_i, \kappa_j)}\right) p(\rho_i, \kappa_j), \tag{2.3}$$

where $\sigma^2(\rho_i, \kappa_j) = \sum_{k=1}^K \rho_{ik}\kappa_{jk}\,\sigma_k^2$, and $\mu(\rho_i, \kappa_j) = \sum_{k=1}^K \rho_{ik}\kappa_{jk}\,(\mu_k + \alpha_{ik} + \beta_{jk})$. That is, each possible combination of layers (plaids) forms a bicluster. We will refer to the components of the mixture generated by only one layer as biclusters. The other components will be referred to as combination biclusters. In practice, most of the combination biclusters are empty. Hence, the actual number of components is much smaller than 2^K when K is large. Model (2.2) is not realistic since the variance increases with the number of biclusters that contribute to the response. Intuitively, the variance should be kept constant. This is what the plaid model assumes (Lazzeroni and Owen [23]), i.e. $\sigma(\rho_i, \kappa_j) = \sigma$ for all (ρ_i, κ_j). The plaid model is a regression model. As such, it cannot be expressed as (2.2). However, the marginal of y_{ij} is still a Gaussian-mixture with 2^K components. Also note that the original plaid model does not explicitly include the 0-bicluster component. However, it is implicitly given by a normal distribution with mean μ_0 and variance σ^2 equal to the regression variance. That is, the mixture model given by equation (2.3) is still valid with $\sigma_0 = \sigma(\rho_i, \kappa_j) = \sigma$, and $\mu_k = \mu_0 + \mu_k'$, where the $\{\mu_k'\}$ are the deviations from the overall mean μ_0. If we suppose that every data cell y_{ij} in the data matrix may belong to only one bicluster, then the labels must satisfy the constraints $\sum_{k=0}^{K}\rho_{ik}\kappa_{jk}=1$, for all (i,j). It is easy to see that this case corresponds to a $(K+1)$-component Gaussian mixture model for the marginal of y_{ij}

$$p(y_{ij})=\sum_{k=0}^{K}\frac{1}{\sigma_k}\phi\left(\frac{y_{ij}-\mu_k-\alpha_{ik}-\beta_{jk}}{\sigma_k}\right)p(\rho_{ik}\kappa_{jk}=1),$$

with $\alpha_{i0}=\beta_{j0}=0$ for all (i,j). This corresponds to the model introduced by Cheng and Church (2000). In this model there is no overlapping between layers, i.e. each cell belongs to only one bicluster, though a line or a column may belong to more than one bicluster. We note that the notion of overlapping is not always the same in the literature. Some authors say that there is overlapping if a line or column belongs to more than one bicluster (Gu and Liu [16], Lazzeroni and Owen [23], Turner et al. [33]).

In what follows, we will work with the general mixture model given by (2.3) where $\sigma^2(\rho_i,\kappa_j)$ will just denote the variance associated to y_{ij} given the pair (ρ_i,κ_j) (that is, we will no longer suppose that model (2.2) holds). This model is a special case of the so-called mixture of experts model in the machine learning literature (Jordan and Jacobs [19]). However, note that in the usual mixture of experts model $p(\rho_i,\kappa_j)$ is modeled by π_{ijk}. In general, the biclustering parametrization of the labels will yield a more parsimonious model, unless these probabilities are modeled as function of the observed variables (e.g. through logistic regression). This is difficult to do in the biclustering framework since usually there is no further information on the cell data, that is, there is no covariable information. To our knowledge, this type of model has not been proposed in the biclustering literature or at least, it has not been widely applied. Instead, the Lazzeroni and Owen plaid model has become more popular. Recall that this model assumes that $\sigma_{ij}^2=\sigma^2$ independently of the cell (i,j). In this model a data cell can be seen as having contributions from several biclusters. Many researchers think of this property as if the data cell were a member of several biclusters. However, as we can see from the mixture formulation of the model, this is not true. Membership in several biclusters is better thought of as a data cell with high probabilities of being in more than one bicluster. A data cell with contributions from several biclusters in the plaid model corresponds to a cell lying in a combination bicluster of the mixture.

2.3 Parameter estimation

Since biclustering corresponds to finite mixture modeling, a straightforward application of the EM algorithm (Dempster et al. [11]) appears as a good procedure to find estimates of the parameters and bicluster labels. This is the case for at least the Cheng and Church model where only $K+1$ sets of parameters need to be estimated. The general bicluster mixture model involves a number of parameters that increases exponentially with the number of layers K. It is perhaps for this reason that the literature does not favor EM as a valuable alternative. However, we will see here that instead hard-EM has been preferred. We remark that there is no mention of hard-EM in the biclustering literature. The community appears not to have realized yet that this is the algorithm most researchers are using to fit the models. Let θ denote the set of all parameters in the model. We will write $\mu(\rho_i, \kappa_j)$ as $\mu(\rho_i, \kappa_j, \theta)$ to make explicit the dependency of this quantity on θ. In the EM algorithm, the expectation step corresponds to the computation of

$$\begin{aligned} Q(\theta|\bar{\theta}) = -\frac{1}{2}\sum_{\rho,\kappa}\sum_{i,j} p(\rho,\kappa|\{y_{ij}\},\bar{\theta})\Bigg\{\log\sigma^2(\rho_i,\kappa_j) \\ + \frac{1}{\sigma^2(\rho_i,\kappa_j)}\Big(y_{ij}-\mu(\rho_i,\kappa_j,\theta)\Big)^2\Bigg\} + \sum_{\rho,\kappa}\log p(\rho,\kappa)\; p(\rho,\kappa|\{y_{ij}\},\bar{\theta}), \end{aligned}$$

where $\bar{\theta}$ is the current estimate of the parameters. The goal of computing this quantity is the M-step, that is, the maximization of $Q(\theta|\bar{\theta})$. Note, however, that this quantity is intractable for large K. A possible solution is to approximate the sum over all possible values of the labels (ρ,κ) by the largest term in the sum. But this term will depend on the value of θ. Another popular solution, the so-called hard-EM algorithm, is to replace the sum by the term associated to the largest weight $p(\hat{\bar{\rho}},\hat{\bar{\kappa}}|\{y_{ij}\},\bar{\theta})$. Once the labels have been fixed, the M-step corresponds to maximize over θ

$$\bar{Q}(\theta|\bar{\theta}) = -\frac{1}{2}\sum_{i,j}\Bigg\{\log\sigma^2(\hat{\bar{\rho}}_i,\hat{\bar{\kappa}}_j) + \frac{1}{\sigma^2(\hat{\bar{\rho}}_i,\hat{\bar{\kappa}}_j)}\Big(y_{ij}-\mu(\hat{\bar{\rho}}_i,\hat{\bar{\kappa}}_j,\theta)\Big)^2\Bigg\}.$$

For the plaid model, this reduces to

$$\bar{Q}(\theta|\bar{\theta}) = -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i,j}\left(y_{ij} - \mu(\hat{\bar{\rho}}_i, \hat{\bar{\kappa}}_j, \theta)\right)^2.$$

Note that the hard-EM algorithm in this setup coincides with a simple version of the iterated conditional modes (ICM) algorithm (Besag [4], Lindley and Smith [24]). That is, for the current value of the parameters $\bar{\theta}$, the labels are estimated by the mode of the full conditional distribution of the labels $p(\rho,\kappa|\{y_{ij}\},\bar{\theta})$. And once the labels have been chosen, the parameters are updated by the mode of the "full conditional" of θ, which is proportional to the likelihood. We remark that a full version of the ICM algorithm would maximize over θ by performing a series of consecutive maximizations of the full conditionals of the lower-dimensional parameters that make up θ. Therefore, a truly hard-EM iteration would be achieved by performing an ICM within ICM in order to maximize over θ. An alternative solution for the plaid model, no longer equivalent to ICM is to maximize

$$\bar{Q}(\theta|\bar{\theta}) = -\frac{n}{2}\log\sigma^2 - p(\hat{\bar{\rho}},\hat{\bar{\kappa}}|\{y_{ij}\},\bar{\theta}) \times \frac{1}{2\sigma^2}\sum_{i,j}\left(y_{ij} - \mu(\hat{\bar{\rho}}_i, \hat{\bar{\kappa}}_j, \theta)\right)^2,$$

which arises from noting that first term in the definition of $Q(\theta|\bar{\theta})$, is $-0.5n\log(\sigma^2)$.

Since hard-EM is a form of ICM with improper priors and uniform priors on the parameters, one should wonder if the use of proper priors would improve the results. Also, the use of priors allows the use of the Gibbs sampler and more generally, Metropolis-Hastings techniques, to obtain not only point estimates of the parameters but also their posterior distributions. These are discussed in Section 2.4.

2.3.1 The EM updating equations

We suppose that the bicluster and combination bicluster probabilities $p(\rho_i,\kappa_j)$ do not depend on the individual observations. Hence, they are constants depending only on the combination bicluster. We denote them as π_k, $k = 0,1,\ldots,2^K-1$. Most observations will fall in a single bicluster (as opposed to a combination bicluster). Hence, in general, the number of combination biclusters (components) is much smaller than 2^K. The expectation step can only be carried over if K is not large. In some problems, K may be moderate, so that a solution through EM may be possible. Also, the EM algorithm is still a very sensible procedure for the important case where each observation is supposed to be in a single bicluster. In this case, the bicluster mixture consists of only K components. We consider here the general case. The formulae are easily adapted to the K-component mixture case.

Note that $p(\rho,\kappa|\{y_{ij}\},\bar{\theta}) = \prod_{i,j} p(\rho_i,\kappa_j|y_{ij},\bar{\theta})$. It is straightforward to verify that

$$p(\rho_i,\kappa_j|y_{ij},\bar{\theta}) = \frac{\frac{1}{\sigma(\rho_i,\kappa_j)}\phi\left((y_{ij}-\mu(\rho_i,\kappa_j,\bar{\theta}))/\sigma(\rho_i,\kappa_j)\right)\pi_{cb(k)}}{\sum_{\rho_i',\kappa_j'}\frac{1}{\sigma(\rho_i',\kappa_j')}\phi\left((y_{ij}-\mu(\rho_i',\kappa_j',\bar{\theta}))/\sigma(\rho_i',\kappa_j')\right)\pi_{cb(k')}},$$

where $cb(k)$ and $cb(k')$ denote the combination biclusters associated to (ρ_i,κ_j) and (ρ_i',κ_j'), respectively. The maximizer of $Q(\theta|\bar{\theta})$ for the plaid model is obtained by taking the derivatives with respect to θ. This yields:

$$\begin{array}{rcl}
\mu_k & = & \frac{1}{\sum_{i,j}E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})}\sum_{i,j}\left\{E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})y_{ij}-\sum_{k'\neq k}E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}\rho_{ik'}\kappa_{jk'})(\mu_{k'}+\alpha_{ik'}+\beta_{jk'})\right\} \\
\alpha_{ik} & = & \frac{1}{\sum_{j}E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})}\sum_{j}\left\{E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})y_{ij}-\sum_{k'\neq k}E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}\rho_{ik'}\kappa_{jk'})(\mu_{k'}+\alpha_{ik'}+\beta_{jk'})\right\}-\mu_k \\
\beta_{jk} & = & \frac{1}{\sum_{i}E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})}\sum_{i}\left\{E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})y_{ij}-\sum_{k'\neq k}E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}\rho_{ik'}\kappa_{jk'})(\mu_{k'}+\alpha_{ik'}+\beta_{jk'})\right\}-\mu_k \\
\pi_{cb(k)} & \propto & \sum_{i,j}p(\rho_i=\rho_{cb(k)},\kappa_j=\kappa_{cb(k)}|y_{ij},\bar{\theta}),
\end{array}$$

where $(\rho_{cb(k)}, \kappa_{cb(k)})$ denotes the corresponding k-th combination bicluster, and

$$\begin{aligned}\sigma^2 &= \frac{1}{qp}\sum_{i,j} E_{\bar{\theta}}\left(y_{ij} - \sum_k \rho_{ik}\kappa_{jk}(\mu_k + \alpha_{ik} + \beta_{jk})\right)^2 \\ &= \frac{1}{qp}\sum_{i,j}\Bigg\{ y_{ij}^2 - 2\sum_k E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) y_{ij}(\mu_k + \alpha_{ik} + \beta_{jk}) \\ &\qquad + \sum_{k,k'} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}\rho_{ik'}\kappa_{jk'})(\mu_k + \alpha_{ik} + \beta_{jk})(\mu_{k'} + \alpha_{ik'} + \beta_{jk'})\Bigg\}.\end{aligned}$$

Note that the updating equations are recursive. The parameters can be estimated by using a Gauss-Seidel relaxation scheme over $k = 1, \ldots, K$. For example, let the superscript "$(t+1)$" denote the coefficients recently updated, and the superscript "(t)", the coefficients not yet updated. Then in order to solve the system, say for α_{ik}'s, we iterate for k within the EM iterations

$$\begin{aligned}\alpha_{ik}^{(t+1)} = \frac{1}{\sum_j E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})}\sum_j \Bigg\{ & E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) y_{ij} - \sum_{k'<k} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}\rho_{ik'}\kappa_{jk'})(\mu_{k'}^{(t+1)} + \alpha_{ik'}^{(t+1)} \\ & + \beta_{jk'}^{(t+1)}) - \sum_{k'>k} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}\rho_{ik'}\kappa_{jk'})(\mu_{k'}^{(t)} + \alpha_{ik'}^{(t)} + \beta_{jk'}^{(t)})\Bigg\} - \mu_k^{(t+1)}.\end{aligned}$$

Also note that the expectation is intractable if the number of biclusters K is large. For example, one needs to compute $E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) = \sum_{\substack{(\rho_i, \kappa_j) \\ \rho_{ik}\kappa_{jk}=1}} p(\rho_i, \kappa_j | y_{ij}, \bar{\theta})$, which is a sum involving 2^K terms. In the non-overlapping bicluster model, the sum reduces to one term $E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) = p(\rho_i\kappa_j = 1, \prod_{k' \neq k}(1 - \rho_{ik'}\kappa_{jk'}) = 1 \,|\, y_{ij}, \bar{\theta})$. In this latter case, the updating equations simplify to

$$\begin{aligned}
\mu_k &= \frac{1}{\sum_{i,j} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})} \sum_{i,j} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) y_{ij} \\
\alpha_{ik} &= \frac{1}{\sum_{j} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})} \sum_{j} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) y_{ij} - \mu_k \\
\beta_{jk} &= \frac{1}{\sum_{i} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})} \sum_{i} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) y_{ij} - \mu_k \\
\sigma_k^2 &= \frac{1}{\sum_{i,j} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk})} \sum_{i,j} E_{\bar{\theta}}(\rho_{ik}\kappa_{jk}) \left(y_{ij} - \mu_k - \alpha_{ik} - \beta_{jk}\right)^2 \\
\pi_{cb(k)} &\propto \sum_{i,j} p(\rho_{ik}\kappa_{jk} = 1 | y_{ij}, \bar{\theta}).
\end{aligned}$$

Unfortunately, despite the simple updating formulas, it does not seem possible to get reliable estimates of ρ_{ik} and κ_{jk} from the EM algorithms for clusters described here. Note that for any $j = 1, \ldots, q$

$$P(\rho_{ik} = 1|y) = P(\rho_{ik} = 1, \kappa_{jk} = 1|y) + P(\rho_{ik} = 1, \kappa_{jk} = 0|y) \geq P(\rho_{ik}\kappa_{jk} = 1|y). \quad (2.4)$$

Thus, if there is a single column j such that $p(\rho_{ik}\kappa_{jk} = 1|y)$ is large, then, the probability that the row i belongs to bicluster k will be high as well. To illustrate this point, we used an artificial data set with two non-overlapping biclusters. The bottom left panel of Figure 2.2 shows the two biclusters. Figure 2.1 shows the estimated probabilities of bicluster membership. The left panel image shows the EM estimated posterior membership probabilities $P(\rho_{i1} = 1, \kappa_{j1} = 1|y)$, whilst the right panel image shows the ones for $P(\rho_{i2} = 1, \kappa_{j2} = 1|y)$. Observe, for example, that in the left panel image (estimated bicluster 1) there are many spots of large (black) probabilities around the black rectangle (bicluster 1), surrounded by very small (white) probabilities. Therefore, by (2.4), almost all the rows belong to the bicluster 1. The same can be concluded from the right panel image; that is, almost all the rows belong to the bicluster 2. Unfortunately, this property makes biclustering different from clustering and renders biclustering as a mixture model an impractical solution. Perhaps, it is exactly to avoid this problem that the literature has favored a hard-EM type solution.

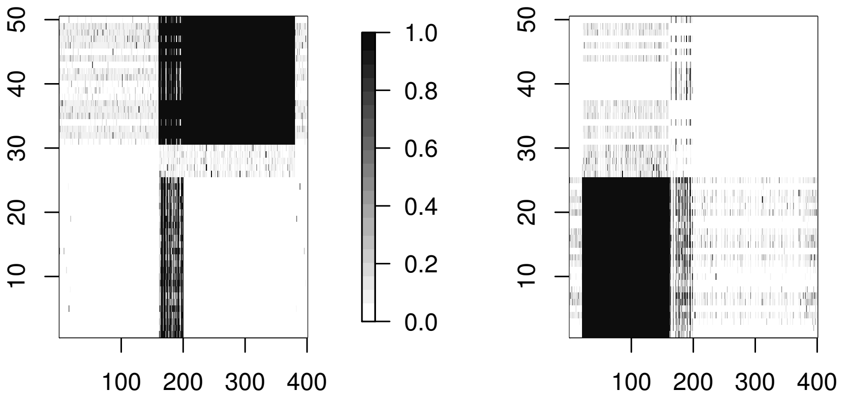


Figure 2.1: Estimated membership probabilities for the two biclusters

2.4 A Bayesian biclustering framework

In this section we consider a fully Bayesian model for biclustering. This would make ICM a more adequate approach than in the non-Bayesian setup. The Bayesian framework will also allow us to employ the Gibbs sampler and more complex techniques derived from the Metropolis-Hastings algorithm. We note that ICM may be seen as a deterministic greedy Gibbs, where instead of generating stochastic samples, one simply proposes the mode of the conditional distribution.

Given the bicluster labels (ρ, κ), we define $I_k = \{i : \sum_j \rho_{ik}\kappa_{jk} > 0\}$ as the set of rows making up the bicluster k; and $J_k = \{j : \sum_i \rho_{ik}\kappa_{jk} > 0\}$ as the set of columns in bicluster k, $k = 1, \ldots, K$. The bicluster k is given by $B_k = I_k \times J_k$. The number of elements in the bicluster k will be denoted as n_k. The number of rows and columns in this bicluster will be denoted by r_k and c_k, respectively. Note that $n_k = r_k \times c_k$.

We suppose that given the bicluster labels, the prior of the row effects $\{\alpha_{ik}\}$ is a multivariate normal distribution with mean zero and variance-covariance matrix given by

$$\mathrm{Cov}(\alpha_{ik},\alpha_{i'k}) = \begin{cases} (1-1/r_k)\sigma_\alpha^2, & \text{if } i'=i\in I_k \\ -\sigma_\alpha^2/r_k, & \text{otherwise.} \end{cases}$$

Similarly, we suppose that the prior for $\{\beta_{jk}\}|(\rho,\kappa)$ follows a multivariate normal distribution with mean zero and variance-covariance matrix given by

$$\mathrm{Cov}(\beta_{jk},\beta_{j'k}) = \begin{cases} (1-1/c_k)\sigma_\beta^2, & \text{if } j'=j\in J_k \\ -\sigma_\beta^2/c_k, & \text{otherwise.} \end{cases}$$

These prior distributions ensure that the row and column effects add up to zero on each bicluster (Kaufmann and Sain [21]). The distributions are degenerate since the variance-covariance matrices are singular. We will use this fact later on when deriving the full conditional distribution of these parameters.

The priors for the means $\mu_0, \{\mu_k\}$ are given by independent zero-mean normal distributions with variances $\sigma_{\mu_0}^2$, and σ_μ^2, respectively. The prior of the bicluster variances σ_k^2 are independent inverse-$\chi^2(s^2,\nu)$ with scaling parameter s^2 and ν degrees of freedom. The prior for the variance σ_0^2 associated to the zero-bicluster is also an inverse-$\chi^2(s_0^2,\nu_0)$.

In general, the bicluster may be overlapping (producing combination biclusters). The amount of overlapping may be controlled by imposing a prior that restricts it. Hence, we suppose that the prior for the bicluster labels is proportional to $\prod_{ij}\exp\{-\lambda\,|1-\gamma_{ij}-\sum_{k=1}^K \rho_{ik}\kappa_{jk}|\}$. The parameter $\lambda \geq 0$ controls the amount of biclustering overlapping. The larger λ, the less overlapping. We will refer to the plaid model with this prior as the *penalized plaid model.* Note that for very large values of λ the model is a non-overlapping bicluster model. If $K=1$ or $\lambda=0$, the prior on the labels becomes a uniform prior.

2.4.1 The full conditionals

The Gibbs sampler as well as ICM relies on the knowledge of the full conditionals of the parameters. In this section we spell them out.

Note that the likelihood may be written as

$$\exp\{-\frac{1}{2}\sum_{i,j}(1-\gamma_{ij})\Big\{\Big(\frac{y_{ij}-\sum_k \rho_{ik}\kappa_{jk}(\mu_k+\alpha_{ik}+\beta_{jk})}{\sigma(\rho_i,\kappa_j)}\Big)^2+\log\sigma(\rho_i,\kappa_j)^2\Big\}$$
$$-\frac{1}{2}\sum_{i,j}\gamma_{ij}\Big((\frac{y_{ij}-\mu_0}{\sigma_0})^2+\log\sigma_0^2\Big)\}.$$

Let k, i.e., the bicluster k, be fixed. Define the variables $z_{ijk} = y_{ij} - \sum_{k'\neq k}\rho_{ik'}\kappa_{jk'}(\mu_{k'} - \alpha_{ik'} - \beta_{jk'})$, $\alpha_k = (\alpha_{ik})_{i\in I_k} \in \mathbb{R}^{r_k}$, $\beta_k = (\beta_{jk})_{j\in J_k} \in \mathbb{R}^{c_k}$, and the matrices $R_k = \mathrm{diag}(\sum_{j\in J_k}\sigma^{-2}(\rho_i,\kappa_j))$, and $C_k = \mathrm{diag}(\sum_{i\in I_k}\sigma^{-2}(\rho_i,\kappa_j))$.

2.4.1.1 The row and column effects

Let $\mathbf{1}_m$ denote the vector of all 1's in $\mathbb{R}^m$. Since the variance of α_k is given by $\sigma_\alpha^2 V_k = \sigma_\alpha^2(I_{r_k} - \frac{1}{r_k}\mathbf{1}_{r_k}\mathbf{1}'_{r_k})$, we may write $\alpha_k = V_k a_k$ for a random vector $a_k \sim N(0,\sigma_\alpha^2 I_{r_k})$. It is easy to verify that the full conditional of a_k is a multivariate normal with mean $\mu_{a,k}$ and variance $\Sigma_{a,k}$ given by

$$\mu_{a,k} = (V_k R_k V_k + \sigma_\alpha^{-2} I_{r_k})^{-1} V_k z_{\alpha,k},$$
$$\Sigma_{a,k} = (V_k R_k V_k + \sigma_\alpha^{-2} I_{r_k})^{-1},$$

where $z_{\alpha,k} = (\sum_{j\in J_k}(z_{ij} - \mu_k - \beta_{jk})/\sigma^2(\rho_i,\kappa_j))_{i\in I_k}$. Similarly, let $U_k = (I_{c_k} - \frac{1}{c_k}\mathbf{1}_{c_k}\mathbf{1}'_{c_k})$. We may write $\beta_k = U_k b_k$ for a random vector $b_k \sim N(0,\sigma_\beta^2 I_{c_k})$. It is easy to verify that the full conditional of b_k is a multivariate normal with mean $\mu_{b,k}$ and variance $\Sigma_{b,k}$ given by

$$\mu_{b,k} = (U_k C_k U_k + \sigma_\beta^{-2} I_{c_k})^{-1} U_k z_{\beta,k},$$
$$\Sigma_{b,k} = (U_k C_k U_k + \sigma_\beta^{-2} I_{c_k})^{-1},$$

where $z_{\beta,k} = (\sum_{i\in I_k}(z_{ij} - \mu_k - \alpha_{ik})/\sigma^2(\rho_i,\kappa_j))_{j\in J_k}$.

For the plaid model $\sigma(\rho_i,\kappa_j) = \sigma$, and for the model of Cheng and Church [9], $\sigma(\rho_i,\kappa_j) = \sigma_k$. In both cases the variance is constant on each bicluster. Therefore, for these models, $R_k = \sigma_k^{-2} c_k I_{r_k}$ and $C_k = \sigma_k^{-2} r_k I_{c_k}$. Hence the conditional means and variances for a_k and b_k become

$$\begin{aligned}
\mu_{a,k} &= \left(\frac{c_k}{\sigma_k^2} + \frac{1}{\sigma_\alpha^2}\right)^{-1} \frac{c_k}{\sigma_k^2}(\bar{z}_{i\cdot k} - \bar{z}_k)_{i\in I_k},\\
\Sigma_{a,k} &= \left(\frac{c_k}{\sigma_k^2} + \frac{1}{\sigma_\alpha^2}\right)^{-1} \left(I_{r_k} + \frac{c_k}{r_k}\frac{\sigma_\alpha^2}{\sigma_k^2}\mathbf{1}_{r_k}\mathbf{1}_{r_k}'\right),\\
\mu_{b,k} &= \left(\frac{r_k}{\sigma_k^2} + \frac{1}{\sigma_\beta^2}\right)^{-1} \frac{r_k}{\sigma_k^2}(\bar{z}_{\cdot jk} - \bar{z}_k)_{j\in J_k},\\
\Sigma_{b,k} &= \left(\frac{r_k}{\sigma_k^2} + \frac{1}{\sigma_\alpha^2}\right)^{-1} \left(I_{c_k} + \frac{r_k}{c_k}\frac{\sigma_\beta^2}{\sigma_k^2}\mathbf{1}_{c_k}\mathbf{1}_{c_k}'\right),
\end{aligned}$$

where $\bar{z}_k$ denotes the mean of the values of z_{ijk} in the bicluster k, and $\bar{z}_{i\cdot k} = \sum_{j\in J_k} z_{ijk}/c_k$, and $\bar{z}_{\cdot jk} = \sum_{i\in I_k} z_{ijk}/r_k$.

It can be easily shown that the full conditionals of the means μ_k, $k = 0, 1, \ldots, K$ are also normal distributions with means and variances given by

$$\begin{aligned}
\mu_{\mu,k} &= \left(\frac{1}{\sigma_\mu^2} + \sum_{(i,j)\in B_k} \frac{1}{\sigma^2(\rho_i, \kappa_j)}\right)^{-1} \sum_{(i,j)\in B_k} \frac{z_{ij} - \alpha_{ik} - \beta_{jk}}{\sigma^2(\rho_i, \kappa_j)},\\
\Sigma_{\mu,k} &= \left(\frac{1}{\sigma_\mu^2} + \sum_{(i,j)\in B_k} \frac{1}{\sigma^2(\rho_i, \kappa_j)}\right)^{-1}.
\end{aligned}$$

Again, for the plaid and Cheng and Church models, the means and variances simplify to

$$\mu_{\mu,k} = \left(\frac{1}{\sigma_\mu^2} + \frac{n_k}{\sigma_k^2}\right)^{-1} \frac{n_k}{\sigma_k^2}\bar{z}_k, \qquad \Sigma_{\mu,k} = \left(\frac{1}{\sigma_\mu^2} + \frac{n_k}{\sigma_k^2}\right)^{-1}.$$

Note that when σ_μ^2, σ_α^2, and σ_β^2 tend to infinity we obtain the hard-EM (or ICM) estimators.

2.4.1.2 The variances

The full conditionals of the variances σ_k^2 given by the model in equation (2.2) are also easily found. Although this model is not realistic (because it forces an increase in the variance with the number of biclusters), the equations will help us to find the full conditionals of the more practical models. Let $z_{ij} = y_{ij} - \sum_{k'} \rho_{ik'} \kappa_{jk'} (\mu_{k'} + \alpha_{ik'} + \beta_{jk'})$, and set $s^2_{ijk} = \sum_{k' \neq k} \rho_{ik'} \kappa_{jk'} \sigma^2_{k'}$. The full conditionals of the variances σ^2_k are proportional to

$$\exp\left\{ -\frac{1}{2} \left(\sum_{(i,j) \in B_k} \frac{z^2_{ij}}{\sigma^2_k + s^2_{ijk}} + \frac{\nu s^2}{\sigma^2_k} \right) - \frac{1}{2} \sum_{(i,j) \in B_k} \log(\sigma^2_k + s^2_{ijk}) - (\frac{\nu}{2} + 1) \log \sigma^2_k \right\}.$$

If we suppose that there is no overlapping among the biclusters, then $s^2_{ijk} = 0$. This corresponds to the Cheng and Church model [9]. The corresponding full conditional of σ^2_k is an inverse-χ^2 distribution with scale $(\nu s^2 + \sum_{(i,j) \in B_k} z^2_{ij})/(\nu + n_k)$, and $\nu + n_k$ degrees of freedom. If instead we suppose that $\sigma(\rho_i, \kappa_j) = \sigma$ independently of the cell (i, j) (i.e., $\sigma_k = \sigma$ and $s_{ijk} = 0$ for all $k = 0, 1, \ldots, K$), then we obtain the full conditional of σ^2 for the plaid model. This is also an inverse-χ^2 distribution, but this time with scale $(\nu s^2 + \sum_{i,j} z^2_{ij})/(\nu + pq)$, and $\nu + pq$ degrees of freedom.

2.4.1.3 The labels

To find the full conditional of the labels, say ρ_{ik}, we use the fact that

$$\begin{aligned} y_{ij} - \sum_{k'=1}^{K} \rho_{ik'} \kappa_{jk'} (\mu_{k'} + \alpha_{ik'} + \beta_{jk'}) &= z_{ijk} - \rho_{ik} \kappa_{jk} (\mu_k + \alpha_{ik} + \beta_{jk}) \\ &= \rho_{ik} \kappa_{jk} (z_{ijk} - \mu_k - \alpha_{ik} - \beta_{jk}) + (1 - \rho_{ik}) z_{ijk} + \rho_{ik} (1 - \kappa_{jk}) z_{ijk}. \end{aligned}$$

Note that $\gamma_{ij} = \prod_{k=1}^{K}(1-\rho_{ik}\kappa_{jk})$. For a given k, we will write $\gamma_{ijk} = \prod_{\substack{k'=1\\k'\neq k}}^{K}(1-\rho_{ik'}\kappa_{jk'})$. Then $\rho_{ik}\kappa_{jk}(1-\gamma_{ij}) = \rho_{ik}\kappa_{jk} - \gamma_{ijk}(1-\rho_{ik}\kappa_{jk})\rho_{ik}\kappa_{jk} = \rho_{ik}\kappa_{jk}$. We have

$$\begin{aligned}
&\sum_j (1-\gamma_{ij})\frac{(z_{ijk}-\rho_{ik}\kappa_{jk}(\mu_k+\alpha_{ik}+\beta_{jk}))^2}{\sigma^2(\rho_i,\kappa_j)}\\
&= \rho_{ik}\sum_{j\in J_k}\frac{(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2}{\sigma^2(\rho_i,\kappa_j)}\\
&\qquad + (1-\rho_{ik})\sum_j (1-\gamma_{ij})\frac{z_{ijk}^2}{\sigma^2(\rho_i,\kappa_j)} + \rho_{ik}\sum_{j\notin J_k}(1-\gamma_{ij})\frac{z_{ijk}^2}{\sigma^2(\rho_i,\kappa_j)}\\
&= \rho_{ik}\sum_{j\in J_k}\frac{(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2}{\sigma^2(\rho_i,\kappa_j)}\\
&\qquad + (1-\rho_{ik})\sum_{j\in J_k}(1-\gamma_{ijk})\frac{z_{ijk}^2}{\sigma^2(\rho_i,\kappa_j)} + \sum_{j\notin J_k}(1-\gamma_{ijk})\frac{z_{ijk}^2}{\sigma^2(\rho_i,\kappa_j)}.
\end{aligned}$$

As before, let θ denote the set of parameters of the model. Define

$$\begin{aligned}
A_{ik} &= \exp\left\{-\frac{1}{2}\sum_{j\in J_k}\frac{(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2}{\sigma^2(\rho_i,\kappa_j)}\right\}\left(\prod_{j\in J_k}\sigma^2(\rho_i,\kappa_j)\right)^{-1/2},\\
B_{ik} &= \exp\left\{-\frac{1}{2\sigma_0^2}\sum_{j\in J_k}\gamma_{ijk}(y_{ij}-\mu_0)^2\right\}\left(\sigma_0^2\right)^{-\sum_{j\in J_k}\gamma_{ijk}/2},\\
C_{ik} &= \exp\left\{-\frac{1}{2}\sum_{j\in J_k}(1-\gamma_{ijk})\left(\frac{z_{ijk}^2}{\sigma^2(\rho_i,\kappa_j)}+\log\sigma^2(\rho_i,\kappa_j)\right)\right\},\\
D_{ik,\rho_{ik}} &= \exp\left\{\frac{1}{2}\sum_{j\notin J_k}(1-\gamma_{ijk})\left(\frac{z_{ijk}^2}{\sigma^2(\rho_i,\kappa_j)}+\log\sigma^2(\rho_i,\kappa_j)\right)\right\}.
\end{aligned}$$

Also let $\rho_{(-ik)}$ denote the set of all row labels except ρ_{ik}. From the above equation it is straightforward to verify that the full conditionals of ρ_{ik} satisfy

$$p(\rho_{ik}|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta) \propto A_{ik}^{\rho_{ik}}B_{ik}^{1-\rho_{ik}}C_{ik}^{1-\rho_{ik}}D_{ik,\rho_{ik}}\pi(\rho_{ik}),$$

where

$$\pi(\rho_{ik}) = \exp\{-\lambda\sum_j(\sum_{\substack{k'=1\\k'\neq k}}^{K}\rho_{ik'}\kappa_{jk'}+\gamma_{ijk}+(1-\gamma_{ijk})\kappa_{jk}\rho_{ik}-1)\}.$$

In particular,

$$\frac{p(\rho_{ik}=1|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta)}{p(\rho_{ik}=0|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta)} = A_{ik}B_{ik}^{-1}C_{ik}^{-1}D_{ik,1}D_{ik,0}^{-1}\exp\{-\lambda\sum_j(1-\gamma_{ijk})\kappa_{jk}\}.$$

The term $D_{ik,\rho_{ik}}$ may be ignored for models whose variances do not depend on (i,j). In particular, for the plaid model, this ratio is

$$\exp\left\{-\frac{1}{2\sigma^2}\sum_{j\in J_k}\left((z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2+(1-\gamma_{ijk})z_{ijk}^2+\gamma_{ijk}(y_{ij}-\mu_0)^2\right)\right.$$
$$\left.-\lambda\sum_j(1-\gamma_{ijk})\kappa_{jk}\right\}.$$

The full conditional for κ_{jk}'s are found in a similar way by symmetry.

Next, consider the the non-overlapping bicluster model. Note that in this case the term C_{ik} may be conveniently written as

$$\exp\left\{-\frac{1}{2}\sum_{\substack{k'=1\\k'\neq k}}\rho_{ik'}\sum_{j\in J_k\cap J_{k'}}\frac{z_{ijk}}{\sigma_{k'}^2}\right\}\prod_{\substack{k'=1\\k'\neq k}}\sigma_{k'}^{-c_{i.k'k}},$$

where $c_{i.k'k}$ is the number of cells in the set $\{(i,j)\in B_{k'}: j\in J_k\}$. Also note that the term B_{ik} may be written as $\exp\left\{-\frac{\rho_{i0}}{2\sigma_0^2}\sum_{j\in J_k\cap J_0}z_{ij0}^2\right\}\sigma_0^{-c_{i.0k}}$, for $z_{ij0}=y_{ij}-\mu_0$, and where J_0 and $c_{i.0k}$ are defined as before but for the zero-bicluster. Therefore,

$$\frac{p(\rho_{ik}=1|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta)}{p(\rho_{ik}=0|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta)} =$$
$$\sigma_k^{-c_k}\prod_{\substack{k'=0\\k'\neq k}}\sigma_{k'}^{c_{i.k'k}}\exp\left\{-\frac{1}{2\sigma_k^2}\sum_{j\in J_k}(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2+\frac{1}{2}\sum_{\substack{k'=0\\k'\neq k}}\rho_{ik'}\sum_{j\in J_k\cap J_{k'}}\frac{z_{ijk}}{\sigma_{k'}^2}\right\}.$$

Let $E_{ik}=\prod_{\substack{k'=0\\k'\neq k}}\sigma_{k'}^{c_{i.k'k}}\exp\left\{\frac{1}{2}\sum_{\substack{k'=0\\k'\neq k}}\rho_{ik'}\sum_{j\in J_k\cap J_{k'}}\frac{z_{ijk}^2}{\sigma_{k'}^2}\right\}$. Performing the Gibbs sampler for the row labels in this model would favor inclusion, i.e. $\rho_{ik}=1$, if $A_{ik}E_{ik}>1$. That is, inclusion is favored when the likelihood associated with having the row i in bicluster k is larger than that associated with having row i elsewhere. Although this is very reasonable, it is not the only argument to include row i in bicluster k. Cheng and Church [9] thought of favoring inclusion when the average square error $c_k^{-1}\sum_{j\in J_k}(y_{ij}-\mu_k-\alpha_{ik}-\beta_{jk})^2$ is smaller than the current estimate of the bicluster variance σ_k^2. Also, a row is a candidate to be eliminated from bicluster k if the average square error is larger than the current estimate of the bicluster variance σ_k^2. We may incorporate these otherwise ad hoc ideas into a Metropolis-Hastings sampling procedure for sampling the labels as follows. Given the current values of the labels $\{\kappa_{jk}\}$, our proposal $q_{ik}(\rho'_{ik}|\{\kappa_{jk}\})$ proposed ρ'_{ik} with probability proportional to $\sigma_k^{-\rho'_{ik}c_k}\, e^{-\rho'_{ik}c_k/2}\, E_{ik}^{\rho'_{ik}}$. Note that an inclusion, i.e. $\rho'_{ik}=1$, may only be proposed if $\rho_{ik'}\kappa_{jk'}=0$ for all $1\le k'\ne k$, and $j\in J_k$. Hence, only the rows i for which all cells in $\{i\}\times J_k$ are in the zero-bicluster may be proposed for inclusion in the k-th bicluster. For these admissible rows, $E_{ik}=\sigma_0^{c_k}\exp\{\frac{1}{2\sigma_0^2}\sum_{j\in J_k}z_{ij0}^2\}$. Therefore, the proposal will favor inclusion if $\sigma_k^2/\sigma_0^2<\exp\{(s_{k0}^2-\sigma_0^2)/\sigma_0^2\}$, where $s_{k0}^2=\sum_{j\in J_k}z_{ij0}^2/c_k$. That is, inclusion is favored when the current "sample" estimate of the variance in the zero-bicluster of the columns in bicluster k, s_{k0}^2, is relatively large in comparison to the overall variance of the zero-bicluster, σ_0^2. The Metropolis-Hastings acceptance ratio is

$$\begin{aligned}\alpha(\rho'_{ik},\rho_{ik}) &= \min\left\{1,\ \frac{p(\rho'_{ik}|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta)q_{ik}(\rho_{ik}|\{\kappa_{jk}\})}{p(\rho_{ik}|\{y_{ij}\},(\rho_{(-ik)},\kappa),\theta)q_{ik}(\rho'_{ik}|\{\kappa_{jk}\})}\right\}\\ &= \min\left\{1,\exp\left(-\frac{\rho'_{ik}-\rho_{ik}}{2}\left[\sum_{j\in J_k}\frac{(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2}{\sigma_k^2}-c_k\right]\right)\right\}.\end{aligned}$$

Note that the proposal $\rho'_{ik}=\rho_{ik}$ is always accepted. Also always accepted are the proposals $\rho'_{ik}=1$, i.e., inclusion of row i in bicluster k, if

$$\frac{1}{c_k}\sum_{j\in J_k}(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2\le\sigma_k^2,$$

and $\rho'_{ik}=0$, i.e., removal of row i from bicluster k, if

$$\frac{1}{c_k}\sum_{j\in J_k}(z_{ijk}-\mu_k-\alpha_{ik}-\beta_{jk})^2\ge\sigma_k^2.$$

The procedure is similar for the plaid model: it suffices to set $\sigma_k=\sigma_0=\sigma$ in the above formulas.

Note that these moves do not consider moving or swapping rows between arbitrary bicluster. Only swaps between one fixed bicluster and the zero-bicluster are allowed. Also note, that these moves are sufficient to move around all biclusters. However, this procedure might be inefficient in the sense that it may take several moves to swap rows between biclusters. The root of this problem lies in the fact that most biclustering algorithms work with only one bicluster at the time. Once a bicluster and its parameters have been estimated, the labels already estimated are no longer touched. Instead, a search for a new bicluster starts. Once the search for new biclusters is finished, the labels may be re-estimated. However, most of the time this latter step is not done. We note that this type of procedure resembles ICM over the biclusters, i.e., only one bicluster is estimated at the time by letting the others fixed. To allow a move between biclusters in the non-overlapping model, we might proceed as follows. We have to propose two biclusters: k' the "targeted" bicluster, and k the "selected" bicluster. In order to be able to propose a move of row i from the targeted bicluster k' to the selected bicluster k, all columns in J_k need to be included in the k'-th bicluster, i.e. $J_k \cap J_{k'} = J_k$. This is the condition similar to the one already encountered when considering a move from the zero-bicluster to the k-th one. That is, we must have $E_{ik} = \sigma_{k'}^{c_k} \exp\{\frac{1}{2\sigma_{k'}^2} \sum_{j \in J_k} (y_{ij} - \mu_{k'} - \alpha_{ik'} - \beta_{jk'})^2\}$. Consider the same proposal as before. However, this time k' needs to be proposed as well. This may be done uniformly among the admissible biclusters. Note that in this case, the "exclusion" move corresponds to the reversal move, that is, of moving the i-th row in the k-th bicluster to the k'-th bicluster. This is possible since the k-th bicluster is admissible for the reversal move. The move will be favored if $\sigma_k^2/\sigma_{k'}^2 < \exp\{(s_{kk'}^2 - \sigma_{k'}^2)/\sigma_{k'}^2\}$, where $s_{kk'}^2 = \sum_{j \in J_k} (y_{ij} - \mu_{k'} - \alpha_{ik'} - \beta_{jk'})^2/c_k$. That is, the move is favored when the current "sample" estimate of the variance in the k'-th bicluster of the columns in bicluster k, $s_{kk'}^2$, is relatively large in comparison to the overall variance of the k'-th bicluster, $\sigma_{k'}^2$. Not surprisingly, the Metropolis-Hastings acceptance ratio and the general conclusions about when the moves are always accepted are exactly the same as before. This is due to the symmetry concerning all the biclusters in the formulas. We would like to stress that although these moves are always reasonable for the selected bicluster k, they are not necessarily good moves for the targeted bicluster k'. The best balance move may be obtained by performing a Gibbs sampler on the labels (see the comment above).

2.4.1.4 The penalty parameter

The penalty λ may be fixed a priori to a suitable value. However, in the absence of information about its value, one may choose to estimate it. In this latter case, λ is the penalty parameter of the model. We assume a gamma(a,b) prior for it. The full conditional of λ, $p(\lambda|(\rho,\kappa),\{y_{ij}\})$, is proportional to

$$\pi((\rho,\kappa)|\lambda)\pi(\lambda) \propto \prod_{ij}(Z_{ij}(\lambda))^{-1}\lambda^{a-1}\exp\left(-\lambda(b+|1-\gamma_{ij}-\sum_k \rho_{ik}\kappa_{jk}|)\right),$$

where

$$\begin{aligned} Z_{ij}(\lambda) &= \sum_k \sum_{\rho_{ik},\kappa_{jk}} \exp(-\lambda|1-\gamma_{ij}-\sum_k \rho_{ik}\kappa_{jk}|) \\ &= 1+\sum_{L=1}^{K}\binom{K}{L}\exp(-\lambda(L-1)) = 1+e^{\lambda}\left([1+e^{-\lambda}]^K-1\right). \end{aligned}$$

To generate λ in the MCMC sampling, we use a Metropolis-Hastings step. The proposal for λ' is again a gamma distribution

$$q(\lambda'|\lambda) \propto \lambda'^{a-1}\exp\left\{-\lambda'\left(b+\sum_{i,j}|1-\gamma_{ij}-\sum_k \rho_{ik}\kappa_{jk}|\right)\right\}.$$

Thus, the acceptance probability ratio in the Metropolis-Hastings step is given by

$$\min\left(1,\prod_{i,j}\left[(Z_{ij}(\lambda'))^{-1}Z_{ij}(\lambda)\right]\right) = \min\left(1,\exp(pq(\lambda-\lambda'))\frac{X(\lambda)}{X(\lambda')}\right),$$

where $X(\lambda) = \left\{\exp(-\lambda)+(1+\exp(-\lambda))^K-1\right\}^{pq}$ for all $\lambda > 0$.

In our simulations below, we note that λ may be seen as a measure of complexity of the data structure: its value decreases almost linearly in the log-scale with the number of biclusters and the amount of bicluster overlapping in the data.

2.4.2 Estimating the number of biclusters

The problem of estimating the number of biclusters is very rarely treated in the literature. For the most part, the number of biclusters is either fixed a priori, or estimated sequentially by some ad hoc stopping rule such as until no more biclusters of a minimum size are found. For example, the algorithm of Cheng and Church [9] fixes the number of biclusters K a priori. Then, it discovers one bicluster at a time. At each one of the K iterations, the algorithms starts with an initial bicluster that contains all rows and columns. The previously discovered biclusters are masked with uniform random numbers. The process is repeated until the K biclusters are found. A similar ad hoc criterion is applied in the algorithms of Lazzeroni and Owen [23], Turner et al. [33], and Zhang [35]. Therein, the number of maximum biclusters K is fixed a priori. The optimal number of bicluster is chosen according to a measure of relevance of the last bicluster kept. This measure compares a candidate bicluster with a pure noise bicluster. See the above references for further details. We proposed a modified *Deviance Information Criterion* (DIC) [31] suited for biclustering. As pointed out by Celeux et al. [8], in order to properly formulate the DIC criterion, a model needs a *focus* parameter. Unfortunately, this focus parameter is not obvious for mixture models. Let $\Theta = (\alpha, \beta, \mu, \sigma^2)$. Since in the clustering setup, the labels may be seen as latent variables, we suggest considering the marginals $E_{\rho,\kappa} p(y|\Theta, \rho, \kappa))$ instead of the full conditionals in the computation of DIC. This corresponds to choosing Θ as the focus parameter, i.e. our DIC, which we will refer to as DIC$_m$, is given by

$$DIC_m = -2E_\Theta \left[\log(E_{\rho,\kappa|y} p(y|\Theta, \rho, \kappa)) \,|\, y\right] + p_m(\tilde{\Theta}_m),$$

where $\tilde{\Theta}_m$ denotes the maximum a posteriori (MAP) estimator of Θ, and

$$p_m(\tilde{\Theta}_m) = -2E_\Theta \left[\log(E_{\rho,\kappa|y} p(y|\Theta, \rho, \kappa)) \,|\, y\right] + 2\log(E_{\rho,\kappa|y} p(y|\tilde{\Theta}_m, \rho, \kappa)),$$

is the so-called *effective dimension*. As suggested by Celeux et al. [8], we use the MAP estimator so as to have a positive effective dimension. This measure works well in our experiments. However, it is computationally expensive, since the labels' marginals have to be computed. An alternative measure to the DIC_m, is

$$DIC_c \quad = \quad -2E_{\Theta,\rho,\kappa}[\log p(y|\Theta,\rho,\kappa)|y] + p_c(\tilde{\Theta},\tilde{\rho},\tilde{\kappa}).$$

where $(\tilde{\Theta},\tilde{\rho},\tilde{\kappa})$ is the maximum a posteriori estimator of (Θ,ρ,κ), and

$$p_c(\tilde{\Theta},\tilde{\rho},\tilde{\kappa}) = -2E_{\Theta,\rho,\kappa}[\log p(y|\Theta,\rho,\kappa)|y] + 2\log p(y|\tilde{\Theta},\tilde{\rho},\tilde{\kappa}),$$

is the corresponding effective dimension. In our experiments, DIC$_c$ works as well as DIC$_m$, but its computation is much faster. This finding may be a bit surprising at first, since after all, in the clustering setup, the labels are not thought of as parameters but as latent variables. However, a closer look at the biclustering model reveals that the labels (ρ,κ) are not equivalent to the labels in the clustering setup. In fact, our analysis of biclustering as a mixture model in Section 2.3.1 clearly shows that it is the product $\rho_{ik}\kappa_{jk}$ that is equivalent to the cluster labels. The two sets of labels (ρ,κ) are more similar to parameters than latent variables. This is especially true for the overlapping bicluster model. Note that in the clustering setup, this latter model does not and cannot exist.

In our experiments we have also compared these two measures with more classical ones, such as AIC (Akaike [1]) and BIC (Schwarz [28]). In order to compute them, we have used the value of the parameters $(\hat{\Theta},\hat{\rho},\hat{\kappa})$ that maximizes the likelihood among the values generated by our MCMC sampler as the estimator of the maximum likelihood estimator.

2.4.3 Initial values

Finding the initial membership labels (ρ,κ) is a difficult task. Several procedures have been suggested in the literature. We have adopted a technique similar to that of

Turner et al. [33]. We run two independent k-means algorithms Ward [34] with $k = 2$: once for the rows and once for the columns. Using the Cartesian product of the resulting k-means row and column labels, we divide the data matrix into four groups. A single initial bicluster is chosen among these four groups according to a variance criterion explained a few lines below. The procedure is repeated as many times as the number of initial biclusters needed. A single initial bicluster is chosen after each application of the independent row and column k-means algorithms. The elements of the biclusters already chosen are *masked* by replacing their original values y_{ij} by random values. This is done so that in the next iteration a different group may be chosen. The masking procedure is not new. It has been used before by Sheng, Moreau and De Moor [29] to determinate multiple biclusters. The criterion to choose an initial bicluster among the four groups yielded by each iteration is the following. Suppose that the cells of each group follow a random effect additive ANOVA model. That is, on each group g , $y_{ij} = \mu_g + \alpha_{ig} + \beta_{jg} + \varepsilon_{ij}$, $g = 1,2,3,4$. The standard moment estimates of the variances are

$$\hat{\sigma}^2_{g\alpha} = \frac{1}{c_g}\Big(MSS_g(\alpha) - MSS_g(e)\Big),\ \hat{\sigma}^2_{g\beta} = \frac{1}{r_g}\Big(MSS_g(\beta) - MSS_g(e)\Big),\ \hat{\sigma}^2_{ge} = MSS_g(e), \tag{2.5}$$

where r_g is the number of rows in the g-th group, c_g is the number of columns, and $MSS_g(e), MSS_g(\alpha)$ and $MSS_g(\beta)$ are the corresponding mean sum of squares for error, rows and columns, respectively. We select as an initial bicluster the group g that maximizes $(\hat{\sigma}^2_{g\alpha} + \hat{\sigma}^2_{g\beta})/\hat{\sigma}^2_{ge}$. For each initial bicluster, the parameters μ_g, α_{ig} et β_{jg} are initialized as $\bar{y}_{..}$, $\bar{y}_{i.} - \bar{y}_{..}$, $\bar{y}_{.j} - \bar{y}_{..}$, respectively, where $y_{..}$, $\bar{y}_{i.}$, $\bar{y}_{.j}$ stand for the overall bicluster mean, the bicluster i-th row mean, and the bicluster j-th column mean, respectively. The parameter μ_0 is estimated as the arithmetic mean of the zero-bicluster. The variance σ^2 is initialized as the mean sum of squares of all the residuals.

2.4.4 Measuring MCMC convergence

As a stopping rule for exploring the support of the posterior distribution, we used the Kolmogorov-Smirnov (KS) test as suggested by Robert and Casella [25, Ch. 8, pp 370–

372]. We compare the "empirical" distribution of the log-likelihood from MCMC subsamples taken G samples apart. The gap G between the sub-samples is used to run the KS test with two (quasi-) independent populations. To monitor the MCMC convergence to the posterior distribution we plotted the KS test p-values against the number of iterations.

2.5 Experiments with artificial data

In this section we show the results of applying the Bayesian biclustering models to diverse simulated data sets. The goal is to study the behaviour of the models under two real complexities in the data: the number of biclusters and the amount of bicluster overlapping. We will see that augmenting the number of biclusters deteriorates the performance of all the algorithms studied. The same is true if the amount of overlapping between biclusters increases, though the performance of many of the algorithms is not affected as much as when the number of biclusters is increased.

The data were simulated for a number of biclusters $K \in \{1,2,4,6,8,10\}$. The biclusters were allowed to overlap. For each $K \geq 2$, we generated three scenarios of data with $p = 400$ rows and $q = 50$ columns. The first scenario corresponded to non-overlapping biclusters. These data were generated according to the non-overlapping model. The second and third scenarios allowed for a small and sizable amount of overlapping biclusters, respectively. These two scenarios were simulated according to the Bayesian plaid model described in this paper. Figure 2.2 shows some examples of our simulated data sets.

More specifically, for $K = 1$, the data were generated with $\mu_1 \sim N(4,0.05)$ and $\mu_0 \sim N(1,0.05)$. For $K = 2$, $\mu_k \sim N(2k,0.05)$, $k = 1,2$, and $\mu_0 \sim N(0,0.05)$. For $K = 4$ and $K = 6$, the data were generated with $\mu_k \sim N(-2(k+1),0.05)$, $k = 1,\ldots,K$. For $K = 8$ and $K = 10$, $\mu_k \sim N((4k^2+8)/(k+1),0.05)$, $k = 1,\ldots,K$. For $K \in \{4,8,10\}$, $\mu_0 \sim N(0,0.05)$, and for $K = 6$, $\mu_0 \sim N(1,0.05)$. The row effects α_{ik} were generated according to their prior distributions with means set to $\mu_{\alpha_{ik}} = \frac{2}{1+\exp(-i)} - \frac{1}{r_k}\sum_i \frac{2}{1+\exp(-i)}$, $k = 1,\ldots,K$. The column effects β_{jk} were generated similarly. For the non-overlapping bicluster data, we generated the variances as draws from the distributions $\sigma_0^2 \sim$ inverse-$\chi^2(0.01,3)$, and $\sigma_k^2 \sim$ inverse-$\chi^2(s_k^2,3)$, where $s_k^2 = k/100$, $k = 1,..,K$.

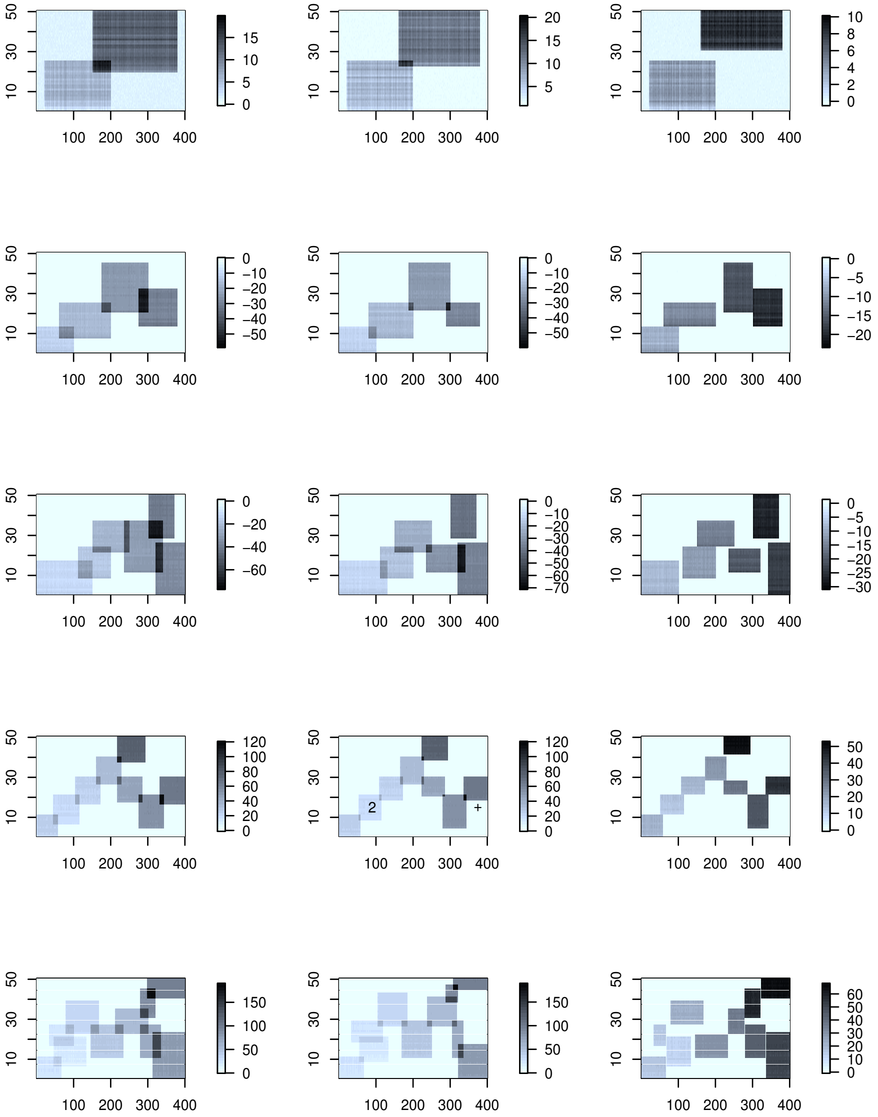


Figure 2.2: Examples of simulated data. The right column shows examples of non-overlapping biclusters (scenario 1). The middle and left columns show moderate (scenario 2) and heavy (scenario 3) overlapping of biclusters, respectively.

For the overlapping bicluster data, we draw the variance σ^2 from an inverse-$\chi^2(0.01,3)$. The hyper-parameters were set to the values: $\nu = \nu_0 = 1$, $s^2 = s_0^2 = 0.05$, and $\sigma_\alpha^2 = \sigma_\beta^2 = \sigma_\mu^2 = \sigma_{\mu_0}^2 = 0.5$. For the penalized plaid model, the prior of λ was a gamma distribution with shape $\alpha = 16$ and scale $\beta = 8$ (that is, the mean prior was set to 2, and the variance to 0.25).

2.5.1 Model comparison

In this section we compare several biclustering models and algorithms for a known biclustering comprising K biclusters. The case of unknown K is dealt with in the next section. The models compared were (A) the non-overlapping bicluster model; (B) the plaid model; and (C) our penalized plaid model with double exponential prior on the labels. The estimation methods compared were (I) the Gibbs sampler and (II) the Metropolis-Hastings algorithm. Furthermore, we compared these methods with (III) the Cheng and Church's biclustering algorithm, and (IV) the plaid model of Turner et al. [33]. We decide to use this latter implementation because it seems to perform better than the original Lazzeroni and Owen's method [23]. The Gibbs and Metropolis-Hastings samplers were run with $20,000$ burn in iterations. Only $2,000$ iterations were kept after the burn in to do the comparison analysis between the models.

2.5.1.0.1 The F1 measure. In order to compare the performance of the models plus algorithms, we used the so-called F1 measure (Allan et al. [2]). The F1 measure have been extensively used in the text mining literature and recently introduced to the biclustering literature (Santamaria et al. [26], Turner et al. [33]). It results from the harmonic mean of *precision* and *recall*. These indices are defined as follows. Let B be a bicluster, r_B be the number of genes in B, c_B be the number of conditions in B and $n_B = r_B c_B$ be the number of elements in B. Suppose that we wish to compare a target bicluster A and a known bicluster B. Then

$$\begin{aligned} \text{recall} &= \frac{(r_{A\cap B})(c_{A\cap B})}{n_B} \\ \text{precision} &= \frac{(r_{A\cap B})(c_{A\cap B})}{n_A}. \end{aligned}$$

Recall measures the proportion of elements in B that belong to A and precision measures the proportion of elements in A captured in B. Turner et al. [33] used these indices to measure quality of the biclusters but they mistakenly named precision by *specificity*. The F1 measure is defined as $F_1(A,B) = 2(r_{A\cap B}) \times (c_{A\cap B})/(n_A + n_B)$. When several biclusters are to be compared, we can use an F1-type average. Let $M_1 = \{A_1, \ldots, A_k\}$ be the set of estimated (target) biclusters, and $M_2 = \{B_1, \ldots, B_\ell\}$, the set of true (known) biclusters. We measure the similarity of the estimate M_1 to the true biclustering M_2 by $S(M_1,M_2) = \frac{1}{k}\sum_{i=1}^{k} \max_j F_1(A_i,B_j)$. Note that $S(M_1,M_2) \leq 1$, and it is equal to 1.0 if and only if $M_1 = M_2$.

A visual summary of the results can be seen in Figure 2.3. It is clear from these bar plots, that the fitting methods suggested by Cheng and Church [9] for the mixture model, and by Turner et al. [33] for the plaid model are not very competitive in comparison with the Bayesian models proposed in this paper. We carried out an analysis of variance of the square-root of the F1 measure (as suggested by the Box and Cox family of power transformations, Box and Cox [5]) for the three models (Penalized Plaid, Plaid and Mixture model), fitted using the two methods (Gibbs sampler or the Metropolis-Hastings sampler) presented in Section 2.4, for the three amounts of overlapping (No overlapping, Moderate overlapping and Heavy overlapping). The ANOVA revealed that a model with all second-order interactions and a third-order interaction between the amount of overlapping, the model and the number of biclusters fitted well the results. The Gibbs sampler performs better than the Metropolis-Hastings algorithm of Section 2.4 for the data sets with eight and ten biclusters. Furthermore, a multiple comparison analysis showed that the plaid and penalized plaid models performed similarly and were the best models. We note that the value of the penalty parameter λ depends a posteriori on the complexity of the data. Its logarithm decreases with the number of biclusters and the amount

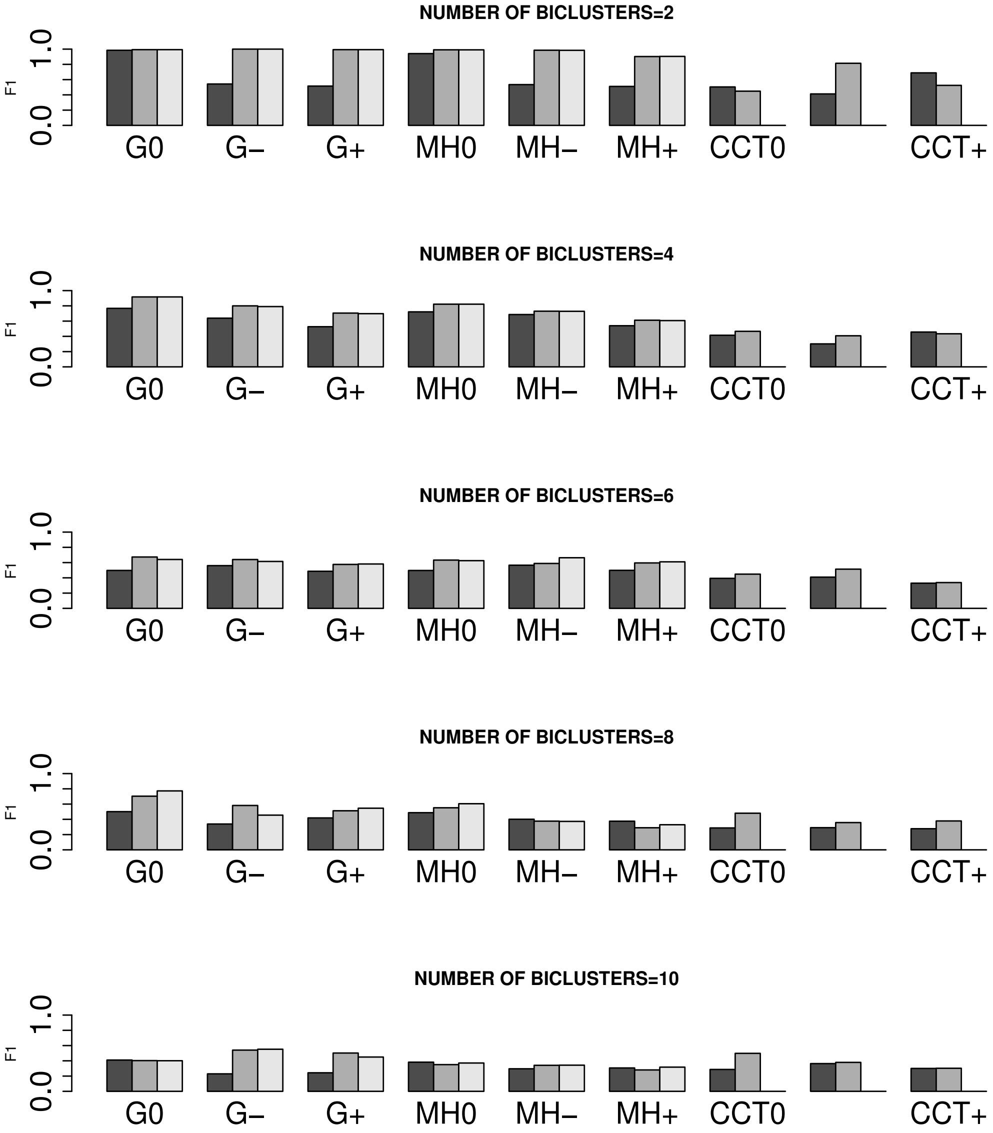


Figure 2.3: F1 means for the different models compared. The darker bars correspond to the Mixture model, the lighter bars, to the penalized plaid model, and the others to the plaid model. The letter "G" stands for the Gibbs sampler, the letters "MH" for the Metropolis-Hastings algorithm described in this paper, and the letters "CC" in the "CCT" triplet stands for the original method suggested by Cheng and Church to fit a mixture model (darker bars), and the letter "T" for the Turner et al.'s algorithm to fit the plaid model (light bars). The symbol "+" stands for *Heavy overlapping* in the biclusters; the symbol "-" stands for *Moderate overlapping*, and the "0" stands for *No overlapping*.

of overlapping (see Figure 2.4). As such the logarithm of λ may be used as a measure of data complexity. Note that the complexity is dominated by the number of biclusters when this is large relatively to the data size. The amount of overlapping does not appear to influence it in this case.

2.5.2 Choice of model

We apply the criteria described in Section 2.4.2 for selecting the number of biclusters. We only used the Gibbs sampler on the penalized plaid model to evaluate the criteria, since the above results have shown that this is one of the best combinations for biclustering estimation. Figure 2.5 shows the DIC_c, DIC_m, AIC and BIC as a function of the number of biclusters for five data sets with $K = 2, 4, 6, 8, 10$ and moderate overlapping. For each data set, each criterion, with the exception of BIC, reaches the minimum at the true number of biclusters, except for $K = 8$, where $K = 9$ is preferred. Also, we observed (not shown) that the F1-measure is maximized at the biclustering implied by the DIC_m's minimum. The data set corresponding to $K = 8$ is shown in the middle column of the fourth row in Figure 2.2. The nine-bicluster solution corresponds to all the biclusters seen in the image plus the block of the zero-bicluster marked with with a "+" in this image. The eight-bicluster solution combined the blocks marked with the "+" and a "2" in the image in one bicluster. It appears that our model preferred the nine-bicluster solution because the block marked with a "+" in the image is somewhat different to the background image (the zero-bicluster) but still much weaker than Bicluster 2. Further analysis reveals that the F1-measure between the true biclustering and the eight-bicluster solution is much smaller (0.63) than that associated with the nine-bicluster solution (0.85).

Note that the DIC_c and DIC_m, curves are almost indistinguishable. This indicates that the expected marginal likelihood is similar to the expected conditional likelihood in these examples. This is also seen in the gene expression application described in the next section. This is partly due to a relatively small effective dimension compared to the expected log-likelihood. Also, after thousands of iterations of the Gibbs sampler, the distribution of the bicluster labels becomes very asymmetric, strongly signalling the final bicluster memberships. As a consequence, the expected log-likelihood is very similar to

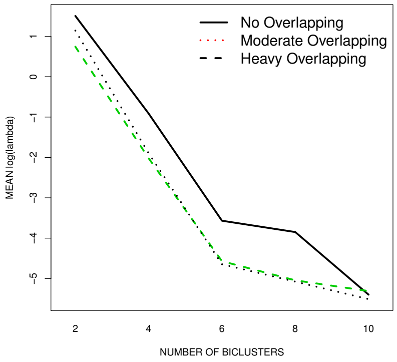


Figure 2.4: The profile means of the logarithm of the λ parameter in the penalized plaid model.

the log of the expected likelihood.

2.6 Applications to gene expression arrays

We have applied our penalized plaid model to elucidate the biclustering structure of the gene expression data associated with the yeast cell cycle data (Eisen et al. [13]). This data set was obtained for five experimental conditions: the diauxic shift, mitotic cell division cycle, sporulation, temperature shock, and reducing shock. The data is available at http://genome-www.stanford.edu/clustering/. It shows the fluctuation of the log-expression levels of 2467 genes over ten experimental series comprising 79 time-points. The columns are denoted by the following prefixes: alpha (columns 1-18), Elu (19-32), cdc (33-47), spo (48-53), spo5 (54-56), spo- (57-58), heat (59-64), dtt (65-68), cold (69-72) and diau (73-79). This data have been analyzed by several researchers: Chu et al. [10], Eisen et al. [13], Katsuhisa and Hiroyuki [20], Lazzeroni and Owen [23]. The original data contained some missing values (1,9% of the data), which we imputed as in Lazzeroni and Owen [23] by using the sum of the row and column means less the overall mean.

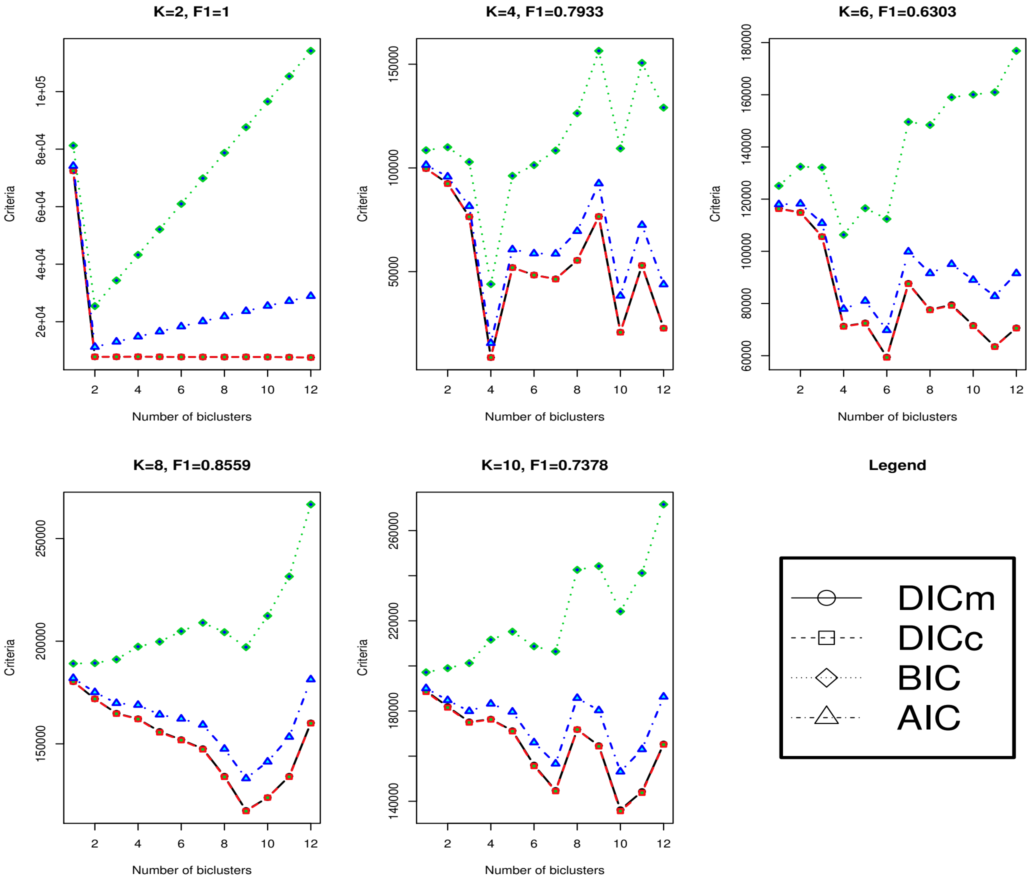


Figure 2.5: DIC$_m$, DIC$_c$, AIC and BIC for the penalized plaid model (p=400, q=50). The F1 measure refers to the F1 value between the true biclustering and that one associated to the biclustering that minimizes the DIC$_m$. The curves for DIC$_m$ and DIC$_c$ are overlying.

2.6.1 Biological interpretation of the biclusters

We use the Gene Ontology (GO) database (Ashburner et al. [3]) to investigate which terms are under or over-represented on each of the estimated bicluster and on each ontology. The GO project is a major bioinformatics initiative with the aim of standardizing the representation of gene attributes across species and databases. The GO project consists of three structured controlled vocabularies or *ontologies* that describe gene products in terms of their associated biological processes, cellular components, and molecular functions in a species-independent manner.

If a high fraction of the genes forming a bicluster are contained in a given ontology, then we expect, and we can say, that such a bicluster corresponds to known GO terms within that ontology. In general, this is difficult to assess, since the large number of genes makes it likely to find spurious but significant relations between GO terms and biclusters just by chance. The Bioconductor project (Falcon and Gentleman [14]) uses the Fisher exact test to measure such relations. In order to control the rate of false positive relations, we employed a Bonferroni correction by adjusting by the number of GO terms present in the data. Even after this adjustment, we found several relations that are significant within each estimated bicluster.

In what follows, as in the work of Lazzeroni and Owen (2002), we will say that gene i is up-regulated (respectively, down-regulated) within the k-th bicluster, if $\mu_k + \alpha_{ik}$ is positive (respectively, negative). We will say that the effect of the j-th condition is positive (respectively, negative) within the k-th bicluster if $\mu_k + \beta_{jk}$ is positive (respectively, negative).

2.6.2 Results

Figure 2.6 shows that the DIC$_m$ and DIC$_c$ criteria select thirteen biclusters for the yeast cell cycle data, while the AIC selects five biclusters. The AIC criterion differs from the DIC because this time the data size is large enough to make the penalty term much more relevant. Recall that the number of parameters in our model is proportional to the data size. The thirteen bicluster solution is more appealing to biologists. The five-bicluster solution aggregates too much of the data and hides interesting small biclusters. Therefore, we decided to continue the analysis of the biclustering chosen by the DIC$_c$. The number of genes and the number of conditions associated with each bicluster are displayed in Table 2.1. About 8% of the genes and 14% of conditions are in a single bicluster and only 3% of the genes are in the zero-bicluster. 86% of the genes and 80% of the conditions are found in overlapping biclusters. Biclusters 4 and 7 are the two largest, with 1388 genes and 1381 genes, respectively. Bicluster five is the smallest one. It is composed of 5 genes. We note that most of the genes are annotated by GO terms, that is, they have a known biological function within GO.

The smaller biclusters, in terms of the number of genes or experimental conditions included, are Biclusters 3, 5, 6, 9, 10, 11 and 13. In general, the smaller biclusters are the most interesting to analyze since their genes are more likely to share common functions. They are shown in Figure 2.7 along with the gene ($\mu_k + \alpha_{ik}$) and experimental condition ($\mu_k + \beta_{jk}$) effects.

Bicluster 3 has 908 genes, all down-regulated, and consists of five experimental conditions of sporulation. All the conditions act negatively. The main molecular functions that characterize this bicluster are associated with structural constituents of ribosomes and rRNA binding. The biological processes are associated with cytoplasmic translation, bio-synthetic and catabolic processes. The cellular components correspond to ribosomes and cytoplasm. This bicluster is similar to the third bicluster (layer 3) found in (Lazzeroni and Owen, 2002). Bicluster 5 contains only five genes. This bicluster is characterized by hydrolase activity (gene YLR286C), cell division (genes YNL327 and YNL066W) and cellular budding (genes YKL185W, YNL327W, and YNL066W). Bicluster 6 consists of 492 genes, six time-points of the sporulation and one time-point of the diauxic shift. In this bicluster, the genes are up-regulated and the experimental conditions present positive effects. The over-represented GO-term functions are related to the threonine-type endopeptidase activity, the DNA binding, the cell cycle, and the DNA replication processes. This bicluster is similar to the first bicluster (layer 1) found in Lazzeroni and Owen [23]. Bicluster 10 consists of 884 genes and 3 time-points of the sporulation The over-represented GO-term functions are related to catalytic and oxi-

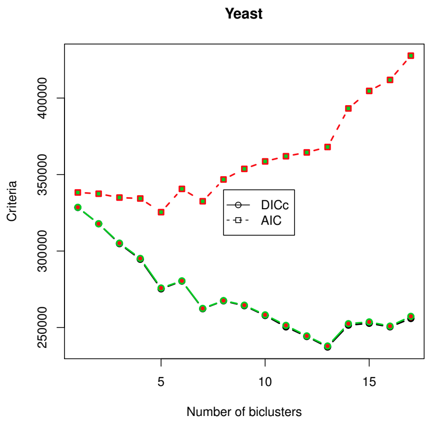


Figure 2.6: DIC$_c$ and AIC for the Yeast cell cycle data

Table 2.1: Summary of the results for the yeast cell cycle data

	Yeast data		
Bicluster	number of conditions	number of genes	annotated genes
0	5	71	71
1	41	415	414
2	53	488	487
3	5	908	907
4	28	1388	1383
5	36	5	5
6	7	492	491
7	31	1381	1378
8	22	680	679
9	8	711	711
10	3	884	882
11	4	789	788
12	10	953	948
13	9	1105	1103

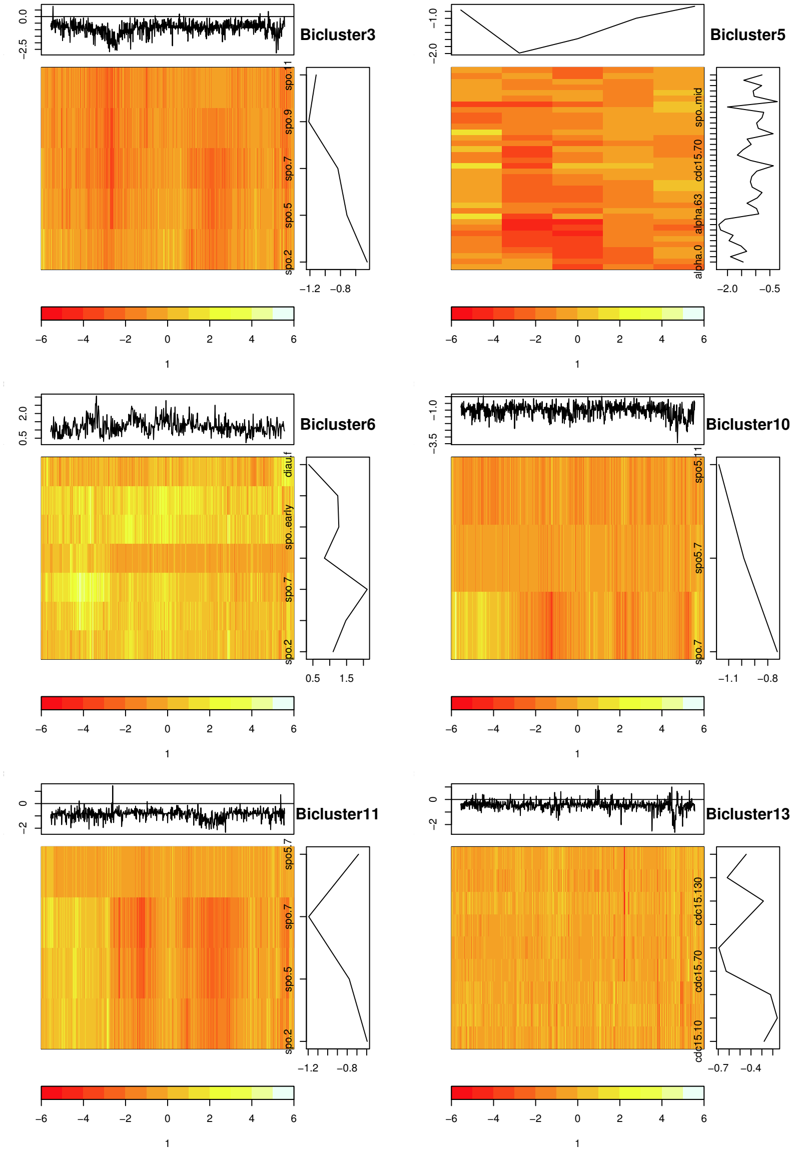


Figure 2.7: Biclusters 3, 5, 6, 10, 11 and 13 of the Yeast cell cycle data. The upper subplots correspond to the gene (column) effects, and the right subplots, to the experimental conditions (row) effects.

doreductase activities, and the catabolic process. Bicluster 11 consists of three times of the sporulation. Its associated GO-term significant functions are related to the structural constituent of ribosomes and rRNA binding, the cytoplasmic translation, the metabolic process, and the gene expression. All the genes were down-regulated in Biclusters 10 and 11. Moreover, the experimental conditions presented negative effects in these biclusters. Bicluster 13 consists of 1105 genes and nine cdc15 experimental conditions. These conditions from the mitotic cell cycle show negative effects. Almost all the genes were down-regulated (95%). The associated GO-term significant functions are related to proteolysis, the cellular protein catabolic process and the modification-dependent protein catabolic process.

The zero-bicluster has 71 genes and five time-points from the alpha-factor ("alpha.77"), the sporulation ("spo5.2"), the heat shock ("heat.10"), and the diauxic shift ("diau.a", "diau.b"). No gene is over-represented and no condition has an effect in this zero-bicluster.

Although most of the small biclusters consist primarily of experimental conditions from sporulation, these biclusters are very different. They play different biological roles in terms of cellular components, biological processes and molecular functions. Lazzeroni and Owen (2002) detected similar biclusters. However, the ones found with our model are also related to other functions that have not been reported before.

2.7 Conclusions

In this work, we introduced an extension of the plaid model, the penalized plaid model. This model incorporates a penalty parameter, λ, that controls (or measures) the amount of bicluster overlapping. Within this model, we found the original plaid model of Lazzeroni and Owen (2002) when λ is set to zero. At the other extreme ($\lambda \to +\infty$) we find a homogeneous-variance version of the non-overlapping model of Cheng and Church (2000). We have proposed both a Gibbs sampler and a Metropolis-Hastings algorithm to estimate the parameters. We note that our Metropolis-Hastings sampler applied to the mixture model with bicluster dependent variances justifies the optimality of the otherwise ad hoc algorithm proposed by Cheng and Church (2000).

We have shown that although the Biclustering problem may be studied as a mixture model, the commonly used (soft) EM-algorithm for mixtures does not seem appropriate. Instead, an ICM-like or hard-EM algorithm appears to be more suitable. In fact, we note that most of the underlying algorithms for biclustering reported in the literature may be justified using hard-EM or ICM. However, we have shown through our simulations that the results derived from our MCMC implementation of the models are better than the original Cheng and Church and plaid model algorithms.

We defined a DIC criterion that seems specially suitable for the biclustering problem. Our DIC for biclustering was inspired by the work of Celeux et al. [8] who noticed that the original DIC was not well-defined for mixture models. In our experiments, our marginal and conditional DIC criteria performed very well. Although we believe that, in principle, the marginal DIC should be preferred to the conditional DIC, we admit that this latter criteria is easier and faster to compute.

We applied our penalized plaid model to the yeast cell cycle data of Eisen et al. [13]. We found thirteen biclusters in the data as indicated by our conditional DIC model selection criterion. Among the biclusters, we obtained the main biclusters found in Lazzeronni and Owen (2002). We showed that these biclusters are very different as indicated by their diverse biological roles obtained using the GO ontology annotations (Falcon and Gentleman [14]).

Acknowledgments

This research was supported by NSERC grant 327689-06.

BIBLIOGRAPHIE

[1] Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control 19*(6), 716–723.

[2] Allan, J., J. Carbonell, G. Doddington, J. Yamron, and Y. Yang (1998). Topic detection and tracking pilot study : Final report. In *In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 194–218.

[3] Ashburner, M., C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock (2000). Gene ontology : tool for the unification of biology. *Nature Genetics 25*, 25–29.

[4] Besag, J. (1986). On the statistical analysis of dirty pictures. *J.R. Stat. 48*, 259–302.

[5] Box, G. E. P. and D. R. Cox (1964). An analysis of transformations. *J. Royal Stat. Soc. Series B 26*, 211–243.

[6] Busygin, S., O. Prokopyev, and P. M. Pardalos (2008). Biclustering in data mining. *Computers & Operations Research 35*(9), 2964 – 2987.

[7] Caldas, J. and S. Kaski (2008). Bayesian biclustering with the plaid model. In J. Príncipe, D. Erdogmus, and T. Adali (Eds.), *Proceedings of the* IEEE *International Workshop on Machine Learning for Signal Processing* XVIII, pp. 291–296. IEEE.

[8] Celeux, G., F. Forbes, C. Robert, and D. Titterington (2006). Deviance information criteria for missing data models. *Bayesian Analysis 1*(4), 651–674.

[9] Cheng, Y. and G. Church (2000). Biclustering of expression data. *Int. Conf. Intelligent Systems for Molecular Biology 12*, 61–86.

[10] Chu, S., J. DeRisi, M. Eisen, J. Mulholland, D. Botstein, P. Brown, and I. Herskowitz (1998). The transcriptional program of sporulation in budding yeast. *Science 282*, 699–705.

[11] Dempster, A., N. Laird, and D. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society 39*(1), 1–38.

[12] Dolnicar, S., S. Kaiser, K. Lazarevski, and F. Leisch (2012). Biclustering. *Journal of Travel Research 51*(1), 41–49.

[13] Eisen, M., P. Spellman, P. Brown, and D. Botstein (1998). Cluster analysis and display of genome-wide expression patterns. *Genetics 95*, 14863–14868.

[14] Falcon, S. and R. Gentleman (2007). Using GOstats to test gene lists for GO term association. *Bioinformatics 23*(2), 257–258.

[15] Geman, S. and D. Geman (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 6*(6), 721–741.

[16] Gu, J. and S. Liu (2008). Bayesian biclustering of gene expression data. *The Int. Conf. on Bioinformatics & Computational Biology, BMC Genomics 9*(1), 113–120.

[17] Hastings, W. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika 57*(1), 97–109.

[18] Hochreiter, S., U. Bodenhofer, M. Heusel, A. Mayr, A. Mitterecker, A. Kasim, T. Khamiakova, V. S. Sanden, D. Lin, W. Talloen, L. Bijnens, W. Hinrich, S. Z. Gohlmann, and D. Clevert (2010). Fabia : factor analysis for bicluster acquisition. *Bioinformatics 26*(12), 1520–1527.

[19] Jordan, M. I. and R. A. Jacobs (1994). Hierarchical mixtures of experts and the EM algorithm. *Neural Computation 6*(2), 181–214.

[20] Katsuhisa, H. and T. Hiroyuki (2001). Statistical estimation of cluster boundaries in gene expression profile data. *Bioinformatics 17*(12), 1143–1151.

[21] Kaufmann, C. and R. Sain (2010). Bayesian functional ANOVA modeling using Gaussian process prior distributions. *International Society for Bayesian Analysis 5*(1), 123–150.

[22] Kluger, Y., R. Basri, J. T. Chang, and M. Gerstein (2003). Spectral biclustering of microarray data : coclustering genes and conditions. *Genome Res. 13*, 703–716.

[23] Lazzeroni, L. and A. Owen (2002). Plaid models for gene expression data. *Statistica Sinica 12*, 61–86.

[24] Lindley, D. V. and A. F. M. Smith (1972). Bayes estimates for the linear model. *Journal of the Royal Statistical Society B 34*(1), 1–131+i–iii.

[25] Robert, C. P. and G. Casella (1999). *Monte Carlo Statistical Methods*, Chapter 8, pp. 370–372. Springer Verlag.

[26] Santamaria, R., L. Quintales, and R. Theron (2007). Methods to bicluster validation and comparison in microarray data. In *IDEAL 07 Proceedings of the 8th international conference on Intelligent data engineering and automated learning*, pp. 780–789.

[27] Sara, C. M. and A. L. Oliveira (2004). Biclustering algorithms for biological data analysis : A survey. *IEEE Transactions on computational biology and bioinformatics 1*, 24–45.

[28] Schwarz, G. E. (1978). Estimating the dimension of a model. *Annals of Statistics 6*(2), 461–464.

[29] Sheng, Q., Y. Moreau, and B. De Moor (2003). Biclustering microarray data by Gibbs sampling. *Bioinformatics 19*, ii196–ii205.

[30] Sokal, R. and C. Michener (1958). A statistical method for evaluating systematic relationships. *The University of Kansas Scientific Bulletin 38*, 1409–1438.

[31] Spiegelhalter, D. J., N. Best, B. P. Carlin, and A. van der Linde (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society, Series B 64*, 583–640.

[32] Tanay, A., R. Sharan, and R. Shamir (2005). Biclustering algorithms : A survey. In *In Handbook of Computational Molecular Biology Edited by : Aluru S. Chapman & Hall/CRC Computer and Information Science Series*.

[33] Turner, H., T. Bailey, and W. Krzanowski (2005). Improved biclustering of microarray data demonstrated through systematic performance tests. *Computational Statistics & Data Analysis 48*, 235–254.

[34] Ward, J. H. (1963). Hierarchical groupings to optimize an objective function. *J. American Statistical Association 58*, 234–244.

[35] Zhang, J. (2010). A Bayesian model for biclustering with applications. *J. Royal Statistical Society 59*, 635–656.

CHAPITRE 3

THE GIBBS-PLAID BICLUSTERING MODEL

Abstract

Within the context of gene expression, biclustering refers to the simultaneous clustering of genes and experimental conditions. Many biclustering algorithms have been proposed. Most of them consider genes or conditions as statistically independent entities. We propose a Bayesian plaid model for biclustering that accounts for the prior dependency between the genes or conditions through a random relational graph. The dependency information is modelled from biological knowledge gathered from Gene Ontologies such as GO. Our model assumes that the relational graph is governed by a Gibbs random field. We developed a stochastic algorithm partly based on the Wang-Landau flat-histogram algorithm in order to estimate the posterior distribution of the bicluster membership labels. We show some experiments with real and simulated data and compare the performance of our model with that of the most popular biclustering algorithms.

Key words: Clustering, relational graph, autologistic model, Wand-Landau algorithm, plaid model, gene expression, gene ontology.

3.1 Introduction

DNA microarray and other microarray technologies allow the measurement of the transcription level of a large number of genes within several diverse experimental conditions (or experimental samples) (Sara and Oliveira, 2004, Tanay et al, 2005). The experimental conditions may correspond to either different time points, different environmental samples, or different individuals or tissues. The resulting data from these technologies are usually referred to as *gene expression data*.

Basically, gene expression data may be seen as a data matrix with rows corresponding to genes, and columns to experimental conditions. Each cell of this matrix represents the expression level of a gene under a biological condition. The analysis of gene expression data usually implies the search for groups of co-regulated genes, that is, groups of genes that exhibit similar expression patterns. Or inversely, the analysis may consists of looking for samples or conditions (e.g. patients) with similar expression profiles. These may indicate the same attribute, such as a common type of particular disease (e.g. leukemia). To accomplish this task, exploratory data analysis such as *clustering* are required. A wide range of clustering algorithms have been proposed to analyze gene expression data. They are usually based on techniques such as hierarchical clustering (Sokal and Michener, 1958) and/or K-Means (Ward, 1963). However, there are at least two drawbacks with classical clustering in the context of gene expression data. The first one is the fact that within the context of clustering each gene must belong to one and only one cluster, even though a single gene may participate in multiple cellular processes. The second one is that the clustered genes must have similar expression patterns under all experimental conditions. However, a cellular process may be active only in a subset of conditions. Biclustering overcomes these drawbacks. It aims at discovering bidimensional clusters given by genes and conditions simultaneously. That is, a bicluster is a subset of genes and conditions of the original expression matrix for which the genes present similar patterns on the conditions and conversely, the conditions present similar patterns across the genes.

Good surveys of existing biclustering algorithms can be found in several papers, such as (Sara and Oliveira, 2004), (Tanay et al., 2005) and (Prelic et al., 2006). The Cheng and Church's algorithm ((Cheng and Church, 2000) and the plaid model (Lazzeroni and Owen, 2002) are two of the most popular biclustering methods. Cheng and Church (2000)) seem to be the first authors to propose the term biclustering for the analysis of microarray data. Their algorithm consists of a greedy iterative search aiming at minimizing the mean square residual error. Lazzeroni and Owen (2002) proposed the popular plaid model. They assumed that the expectation of each cell in the data matrix is form by the contribution (sum) of different layers or single biclusters. Recently, many authors (Gu and Liu, 2008; Caldas and Kaski, 2008; Zhang, 2010; Chekouo and Murua, 2012) have generalized the plaid model into a Bayesian framework.

It is apparent from our review of the literature, that prior information about genes or conditions, and pairwise interaction between them are not taken into account in most biclustering models. In this work, we propose a model that does take into account this information. We adopt a Gaussian plaid model as the model describing the biclustering structure of the data matrix. In addition, we incorporate prior information on the dependency between genes and between conditions through dedicated relational graphs, one for the genes and another for the conditions. These graphs are conveniently described by auto-logistic models (Besag, 1974, 2001; Winkler, 2003) for genes and conditions. These distributions are pairwise-interaction Gibbs Random Fields for dependent binary data. They can be interpreted as generalizations of the finite-lattice Ising model (Besag, 2001). The Ising model is a popular two-state discrete mathematical model for ferromagnetism in statistical mechanics. We will refer to this model as the *Gibbs-plaid* biclustering model. In our prior, the inter-dependencies between the genes, that is the edge weights in the graph, are elicited through the information contained in the GO (Gene Ontology) collection. The latter is a major bioinformatics initiative to unify the representation of gene and gene product attributes across all species (Ashburner et al, 2000). It provides an ontology of controlled vocabularies that describes gene products in terms of their associated biological processes, cellular components, and molecular functions in a species-independent manner.

Our prior is elicited from similarities obtained from the GO annotations. A k-nearest-neighbor graph over the genes is built from these similarities. A key parameter of the auto-logistic prior is the so-called temperature parameter T (due to its analogy with the physical process of tempering). The normalizing constant of this prior is, in general, unknown and intractable. Unfortunately, for computational purposes, this constant is needed in order to implement a stochastic algorithm aiming at estimating the posterior distribution of the genes bicluster memberships when T is unknown. Basically, this means that the usual MCMC Metropolis-Hastings procedure is too difficult to apply to our model. Instead, we adopt a hybrid procedure that mixes Metropolis-Hastings with a variant of the Wang-Landau algorithm (Wang and Landau, 2001; Atchade and Liu, 2010; Murua and Wicker, 2012). The convergence of the proposed algorithm to the posterior distribution of the bicluster membership is guaranteed by the work of Atchade and Liu (2010).

We note that some earlier attempts to incorporate gene dependency information have been made in the literature, but within the context of clustering (as opposed to biclustering) and variable selection. Vannucci and Stingo (2011) give a nice review. Stingo et al (2011) have proposed a Bayesian model which incorporates information on pathways and gene networks in the analysis of DNA microarray data. They assumed a Markov Random Field prior to capture the gene-gene interaction network. The neighborhood between the genes uses the pathway structure from the Kyoto Encyclopedia of Genes and Genomes (KEGG) database (Kanehisa and Goto, 2000). Hang et al (2009) and Vignes and Forbes (2011) have also used biological information to do a clustering analysis of gene expression data. Park, Hastie and Tibshirani (2007) also incorporated gene ontology GO annotations to predict survival time and time to metastases of breast cancer patients using gene expression data as predictors variables. There is also some work on clustering of gene expression data with a generalization of the Ising model, the Potts model (Murua, Stanberry and Stuetzle, 2008; Getz, Levine, Domany and Zhang, 2000). However, in these works the Potts model (Sokal, 1996) is used directly as a nonparametric model for clustering (Blatt, Shai and Domany, 1996), and not as a prior accounting for gene-gene interaction on another clustering model.

3.2 The Model

Let p be the number of genes, and q be the number of experimental conditions. Let Y_{ij} denote the expression level of gene i under condition j ($i = 1,\ldots,p$, $j = 1,\ldots,q$). Let K be the number of biclusters. For all i in the sets of genes, j in the set of conditions, and $k = 1,\ldots,K$, define the binary variables ρ_{ik} and κ_{jk} taking values in $\{0,1\}$, so that $\rho_{ik} = 1$ if and only if gene i belongs to bicluster k, and $\kappa_{jk} = 1$ if and only if condition j belongs to bicluster k. The symbols ρ_i and ρ will denote the K-dimensional vector of components $\{\rho_{ik}\}_{k=1}^{K}$ and the pK-dimensional vector comprising all the vectors ρ_i, $i = 1,\ldots,p$, respectively. The symbols κ_j and κ are similarly defined for the conditions.

3.2.1 The plaid model

The Gaussian plaid model states that $Y_{ij} = \mu_{ij}(\rho,\kappa,\Theta) + \varepsilon_{ij}$, where ε follows a Normal$(0,\sigma^2)$, and $\mu_{ij}(\rho,\kappa,\Theta) = \mu_0 + \sum_{i=1}^{K}(\mu_k + \alpha_{ik} + \beta_{jk})$, where $\alpha_k = \{\alpha_{ik}, i = 1,..,p\}$ and $\beta_k = \{\beta_{jk}, j = 1,..,q\}$ are the gene and condition effects associated to bicluster $k = 1,..,K$, measured as deviations from the bicluster mean $\mu_0 + \mu_k$. (μ_0 denotes the overall data mean). The symbol Θ denotes the ensemble of parameters of the model (μ_0,μ,α,β). We assume that the variables Y_{ij}'s given the labels (ρ,κ) and (σ^2,Θ) are independent. That is,

$$P(y|\rho,\kappa,\sigma^2,\Theta) = \prod_{i,j}\frac{1}{\sigma}\phi\left(\frac{y_{ij}-\mu_{ij}(\rho,\kappa,\Theta)}{\sigma}\right) \tag{3.1}$$

where ϕ stands for the standard normal density. Given the bicluster labels (ρ,κ), we define $I_k = \{i : \rho_{ik} = 1\}$ as the set of rows in the bicluster k, and $J_k = \{j : \kappa_{jk} = 1\}$ as the set of columns in bicluster k, $k = 1,\ldots,K$. The bicluster k is given by $B_k = I_k \times J_k$. Let be n_k the number of elements in the bicluster k. The number of rows and columns in this bicluster will be denoted by r_k and c_k, respectively. Note that $n_k = r_k \times c_k$. Let $\mathbf{1}_m$ denote the vector of all 1's in $\mathbb{R}^m$, and $\mathbf{I}_m$ stand for the identity matrix of dimension m. We further assume that given the bicluster labels, the prior of the gene effects $\{\alpha_{ik}\}$ is a multivariate normal distribution with mean zero and variance-covariance matrix given by $\sigma_\alpha^2 V_k = \sigma_\alpha^2(\mathbf{I}_{r_k} - \frac{1}{r_k}\mathbf{1}_{r_k}\mathbf{1}'_{r_k})$. As shown in (Chekouo and Murua, 2012), we may change the parametrization of the model to a proper multivariate normal vector $a_k \sim N(0,\sigma_\alpha^2 I_{r_k})$ so that $\alpha_k = V_k a_k$. Similarly, we suppose that the prior for $\{\beta_{jk}\}|(\rho,\kappa)$ follows a multivariate normal distribution with mean zero and variance-covariance matrix given by $\sigma_\beta^2 U_k = \sigma_\beta^2(\mathbf{I}_{c_k} - \frac{1}{c_k}\mathbf{1}_{c_k}\mathbf{1}'_{c_k})$. Note that these prior distributions verify the conditions of identifiability in the model, i.e., they ensure that the gene and condition effects add up to zero on each bicluster almost surely.

3.2.2 A prior for the bicluster membership

The gene labels ρ_{ik} as well as the condition labels κ_{jk} are usually assumed to be independent (Zhang, 2010; Gu and Liu, 2008). More realistically, in this work, we incorporate prior knowledge on the relation between genes and between conditions (if applicable) by means of relational graphs. For example, the gene relational graph is a k-nearest-neighbor graph whose nodes correspond to the set of genes, and whose edges correspond to the set of most-similar or "closer" genes. It is this notion of similarity that contains the relational information between genes. We define these similarties based on the GO annotations. The GO annotations, also known as GO terms, are organized in a directed acyclic graph (DAG) wherein children annotations inherit annotations from multiple parent terms. We adopt the minimum subsumer of (Resnik, 1999) as a means to build a notion of semantic similarity between any two GO annotations. This idea was first introduced by Lord et al. (2003). Let x denote a GO annotation, and let $P(x)$ be its empirical frequency within a collection of genes. The information content in x is defined as $IC(x) = -\log P(x)$. The information content of the minimum subsumer between two GO annotations x_1 and x_2 is based on the lowest common ancestor in the DAG

$$IC_a(x_1,x_2) = \max_{x \in A(x_1,x_2)} IC(x), \tag{3.2}$$

where $A(x_1,x_2)$ is the set consisting of all common ancestors of the annotations x_1 and x_2. These quantities are readily available using the R package GOSim (Frohlich et al, 2007). One of the simplest normalization of the Resnik's similarities is the Lin's pairwise similarity (Lin, 1998) given by

$$sim(x_1,x_2) = \frac{2IC_a(x_1,x_2)}{IC(x_1)+IC(x_2)} \tag{3.3}$$

Therefore, given two genes i and i', annotated with the sets of GO annotations X_i and $X_{i'}$, respectively, we define their similarity by

$$sim(i,i') = \max_{\substack{x\in X_i \\ x'\in X_{i'}}} sim(x,x'),$$

and their distance by $d^{\rho}(i,i') = 1 - sim(i,i')$. The gene relational graph is defined to have edge weights equal to

$$B_{ii'}(T^{\rho},\sigma_{\rho}) = \frac{1}{T^{\rho}} \exp\left(-\frac{1}{2\sigma_{\rho}^2} d^{\rho}(i,i')^2\right).$$

Here T^{ρ} and σ_{ρ} are the temperature and kernel bandwidth parameters of the graph, respectively. We assume that $B_{ii'}(T^{\rho},\sigma_{\rho}) = 0$ for pairs of genes not connected by an edge. These weights are larger, the more similar the genes are. We will use the notation $i \sim i'$ for nodes that are connected by an edge in the graph. The distribution of the gene labels in this graph is given by the binary Gibbs random field

$$p(\rho_k|a,T^{\rho},\sigma_{\rho}^2) \propto h_{\rho,k}(\rho_k,T^{\rho}) \doteq \exp\left\{\sum_{i=1}^{p} a_i\rho_{ik} + \sum_{i\sim i'} B_{ii'}(T^{\rho},\sigma_{\rho}^2)\,\mathbf{1}_{\{\rho_{ik}=\rho_{i'k}\}}\right\}$$

where $a = \{a_i\}_{i=1}^{p}$ are hyper-parameters controlling the amount of membership ($\rho_{ik} = 1$) in the bicluster, and for every relation A, $\mathbf{1}_A$ denotes the indicator function taking the value 1 if and only if the relation A is satisfied. This Gibbs field is actually a binary autologistic distribution on the labels (Besag, 1974, 2001; Winkler, 2003). This Gibbs prior favors biclusters formed by similar genes in the sense of the distances or similarities chosen to built the relational graph.

The conditions prior

A similar prior relational graph may be built for the conditions if a notion of similarity between the conditions may be defined. This is the case, for example, when the conditions correspond to similar measurements taken over a period of time, such as in gene expression evolution (i.e., time-course) profiles. In this latter case, the distance between conditions may incorporate a measure of smoothness of the time-course profile during consecutive measurements. Alternatively, a measure of correlation may be incorporated in the similarities, if a moving-average or specific ARMA process is assumed on the time-course profiles. These aspects of the modelling processes are better explained within the context of specific applications, such as the ones describe in Section 3.5. For the moment, assume that such a distance between conditions may be defined. We will denote the distance between two conditions j and j' by $d^{\kappa}(j,j')$. The condition relational graph is defined to have edge weights equal to

$$D_{jj'}(T^{\kappa},\sigma_{\kappa}) = \frac{1}{T^{\kappa}} \exp\left(-\frac{1}{2\sigma_{\kappa}^{2}} d^{\kappa}(j,j')^{2}\right).$$

As before, T^{κ} and σ_{κ} are the temperature and kernel bandwidth parameters of the graph, respectively. And we assume that $D_{jj'}(T^{\kappa},\sigma_{\kappa}) = 0$ for pairs of conditions not connected by an edge. The distribution of the condition labels in this graph is then given by the binary auto-logistic distribution

$$p(\kappa_k|c,T^{\kappa},\sigma_{\kappa}^{2}) \propto h_{\kappa,k}(\kappa_k,T^{\kappa}) \doteq \exp\left\{\sum_{j=1}^{q} c_j\kappa_{jk} + \sum_{j\sim j'} D_{jj'}(T^{\kappa},\sigma_{\kappa})\,\mathbf{1}_{\{\kappa_{jk}=\kappa_{j'k}\}}\right\}$$

where $c = \{c_j\}_{j=1}^{q}$ are hyper-parameters controlling the amount of condition membership ($\kappa_{jk} = 1$) in the bicluster. Note that in the absence of any prior information on the dependency between conditions, we may assume that all pairs of conditions (j,j') are far apart, and consequently, that $D_{jj'}(T^{\kappa},\sigma_{\kappa}) = 0$ for all pairs (j,j'). This leads to a prior where all the condition labels κ_{jk} are a priori independent.

3.3 Posterior Estimation

To estimate the posterior of the parameters, specially the one associated to the labels (ρ,κ), we use a hybrid stochastic algorithm. First of all an augmented model is considered in order to efficiently sample the labels through a block Gibbs sampling. This is the

Swendsen-Wang algorithm (Swendsen and Wang,). The algorithm is very-well known in the Physics and imaging literature. We briefly describe it below. The effects parameters and the variances are readily sampled using the usual Gibbs sampler. However, the temperature hyper-parameters associated to the label priors need extra consideration. In order to sample from their posterior, one needs to know the normalizing constant of the priors which unfortunately are intractable. To solve this impass, we adopt the Wang-Landau algorithm (Wang and Landau, 2001; Atchade and Liu, 2010). This is a technique that efficiently samples from a grid of finite temperature values by cleverly estimating the normalizing constant at each iteration. The algorithm travels efficiently over all the temperatures by penalizing each visit. The resulting algorithm is also referred to as a flat-histogram algorithm. Below, we explain a bit more how the technique is applied to our model.

3.3.1 Sampling the labels with known temperatures

Let the number of biclusters k be fixed. We will denote the residuals by $z_{ijk} = y_{ij} - \mu_0 - \sum_{k' \neq k}^{K} \theta_{ijk'} \rho_{ik'} \kappa_{jk'}$. The likelihood is given by

$$P(y|\rho,\kappa,\sigma^2,\Theta) \propto \frac{1}{\sigma^{np}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i,j}(z_{ijk} - \rho_{ik}\kappa_{jk}(\mu_k + \alpha_{ik} + \beta_{jk}))^2\right\}$$

$$= \frac{1}{\sigma^{np}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i,j}\rho_{ik}\kappa_{jk}(z_{ijk} - \mu_k - \alpha_{ik} - \beta_{jk})^2 - \frac{1}{2\sigma^2}\sum_{i,j}(1-\rho_{ik}\kappa_{jk})(z_{ijk})^2\right\}$$

Consequently, the full conditional probability of the genes labels is given by

$$P(\rho_k|y,\rho_{-k},\kappa_k,\sigma^2,\Theta,T^\rho) \propto \exp\left\{\sum_i A_{ik}\rho_{ik} + \sum_{i\sim i'} B_{ii'}(T^\rho,\sigma_\rho)\mathbf{1}_{\{\rho_{ik}=\rho_{i'k}\}}\right\},$$

where $A_{ik} = a_i - (1/2)\sigma^{-2}\sum_{j=1}^{q}\left\{\kappa_{jk}(z_{ijk} - \mu_k - \alpha_{ik} - \beta_{jk})^2 - \kappa_{jk}(z_{ijk})^2\right\}$, and $\rho_{-k} = \rho \setminus \rho_k$. To sample from this full conditional, we use the Swendsen-Wang algorithm (Swendsen and Wang, 1987). This algorithm samples the label in blocks by taking into account the data graph neighborhood system. It defines a set of the independent auxiliary

$0-1$ binary variables $R = \{R_{ii'} : i, i' = 1, \ldots, p\}$, called the bonds. The bonds are set to the value 1 with label dependent probabilities given by

$$p_{ii'} \doteq P(R_{ii'} = 1 | \rho_k) = (1 - \exp\{-B_{ii'}(T^{\rho}, \sigma_{\rho})\}) \mathbf{1}_{\{\rho_{ik} = \rho_{i'k}\}} \mathbf{1}_{\{i \sim i'\}}. \qquad (3.4)$$

The bond $R_{ii'}$ is said to be *frozen* if $R_{ii'} = 1$. Note that necessarily a frozen bond can occur only between neighboring points that share the same label. A set of data graph nodes is said to be connected if for every pair of nodes (i, i') in the set there is a path of frozen nodes in the set connecting i with i'. The Swendsen-Wang algorithm to sample the labels is the following:

1. Given the labels ρ_k, each bond $R_{ii'}$ is frozen independently of the others with probability $p_{ii'}$ if $i \sim i'$ and $\rho_{ik} = \rho_{i'k}$. Otherwise, the bond is set to zero.

2. Given the bond variables R, the graph is partitioned into its connected components. Each connected component C is randomly assigned a label. The assignment is done independently, with 1-to-0 log-odds proportional to $\sum_{i \in C} A_{ik}$. In the special case of the Ising model, and more generally, when $A_{ik} = 0$ for all i, the labels are chosen uniformly at random.

The conditions labels are sampled in a similar manner given the genes labels.

3.3.2 Sampling the labels with unknown temperatures

We assume that the temperatures T^{ρ} and T^{κ} take finitely many values. Let $\mathscr{T}_{\rho}$ and $\mathscr{T}_{\kappa}$ be the sets of m and n possible values for T^{ρ} and T^{κ}, respectively. We assume that a priori that (T^{ρ}, T^{κ}) is distributed uniformly on the grid of values $\mathscr{T}_{\rho} \times \mathscr{T}_{\kappa}$. We may write

$$p(\sigma^2, \Theta, \rho, \kappa, T^{\rho}, T^{\kappa} | y) \propto p(y | \sigma^2, \Theta, \rho, \kappa) \pi(\sigma^2, \Theta) \prod_{k=1}^{K} \left(\frac{h_{\rho,k}(\rho_k, T^{\rho})}{Z_{\rho}(T^{\rho})} \frac{h_{\kappa,k}(\kappa_k, T^{\kappa})}{Z_{\kappa}(T^{\kappa})} \right).$$

where $Z_{\rho}(T)$ and $Z_{\kappa}(T)$ denote the normalizing constants for $h_{\rho,k}(\rho_k, T)$ and $h_{\kappa,k}(\kappa_k, T^{\rho})$, respectively. In general, these constants cannot be easily evaluated and are intractable, except for the very simplest cases. MCMC techniques such as Metropolis-Hastings are of no use here, since the constants change with the value of T. Instead, in order to obtain samples from the posterior of the labels, we used a stochastic algorithm based on the Wang-Landau algorithm (Wang and Landau, 2001; Atchade and Liu, 2010). The sampling from this algorithm gives at the same time approximate samples from the posterior of the labels and the parameters (σ^2, Θ), and estimates of the posterior probability mass function of (T^ρ, T^κ). Atchade and Liu (2010) give a nice exposition of the algorithm and show its convergence. Murua and Wicker (2012) have successfully used a variant of the Wang-Landau algorithm to estimate the posterior of the temperature of the Potts model. Basically, the Wang-Landau algorithm considers the target joint distribution

$$\pi(\sigma^2, \Theta, \rho, \kappa, T^\rho, T^\kappa) \propto p(y|\sigma^2, \Theta, \rho, \kappa)\pi(\sigma^2, \Theta) \prod_{k=1}^{K} h_{\rho,k}(\rho_k, T^\rho) h_{\kappa,k}(\kappa_k, T^\kappa) / \phi(T^\rho, T^\kappa),$$

where

$$\phi(T^\rho, T^\kappa) = Z^{-1} \sum_{\rho,\kappa} \left\{ \left(\int p(y|\sigma^2, \Theta, \rho, \kappa)\pi(\sigma^2, \Theta)\, d\sigma^2 d\Theta \right) \prod_{k=1}^{K} h_{\rho,k}(\rho_k, T^\rho) h_{\kappa,k}(\kappa_k, T^\kappa) \right\},$$

and Z is the constant such that $\sum_{T^\rho \in \mathscr{T}_\rho, T^\kappa \in \mathscr{T}_\kappa} \phi(T^\rho, T^\kappa) = 1$. The algorithm samples from iterative stochastic approximations of this distribution (see the algorithm steps below), so that the marginal of the parameters and labels converges to the target marginal $\pi(\sigma^2, \Theta, \rho, \kappa) = p(\sigma^2, \Theta, \rho, \kappa|y)$ and the marginal of (T^ρ, T^κ) converges to $\pi(T^\rho, T^\kappa) \propto Z$, a uniform distribution on the grid $\mathscr{T}_\rho \times \mathscr{T}_\kappa$. The main idea of the stochastic approximation is to replace $\phi(T^\rho, T^\kappa)$ by an iterative estimate, say $\hat{\phi}(T^\rho, T^\kappa)$. Since $\pi(T^\rho, T^\kappa)$ is uniform, at convergence

$$\frac{\hat{\phi}(T^\rho, T^\kappa)}{\sum_{t^\rho \in \mathscr{T}_\rho, t^\kappa \in \mathscr{T}_\kappa} \hat{\phi}(t^\rho, t^\kappa)} \approx \phi(T^\rho, T^\kappa). \tag{3.5}$$

Therefore, the quantities given in the left-hand-side of equation (3.5) give an estimate of the posterior probability mass function of the temperatures (T^ρ, T^κ).

The Wang-Landau algorithm we have implemented depends on an updating proposal of the form $q(T^\rho, T^\kappa | T^{\rho,(t)}, T^{\kappa,(t)}) = q_\rho(T^\rho | T^{\rho,(t)}) q_\kappa(T^\kappa | T^{\kappa,(t)})$, with $q_\rho(t_1, t_2) =$

$q_\rho(t_m, t_{m-1}) = 1$ and $q_\rho(t_i, t_{i-1}) = q_\rho(t_i, t_{i+1}) = 1/2$ if $1 < i < m$, where we have written $\mathscr{T}_\rho = \{t_1 < t_2 < \cdots < t_m\}$. The proposal q_κ is similarly defined. This proposal corresponds to the proposal of Geyer and Thompson (1995) used within the context of simulated tempering. Atchade and Liu (2010) suggest a different proposal based on a multinomial distribution. However, this latter proposal involves considerably more computations. The algorithm proceeds as follows: Given $(\sigma^{2,(t)}, \Theta^{(t)}, \rho^{(t)}, \kappa^{(t)}, T^{\rho,(t)}, T^{\kappa,(t)})$ and $\hat{\phi}^{(t)} = \{\hat{\phi}(t^\rho, t^\kappa) : (t^\rho, t^\kappa) \in \mathscr{T}_\rho \times \mathscr{T}_\kappa\}$ at iteration t,

1. Sample T from the proposal distribution $q_\rho(\cdot|T^{\rho,(t)})$. Set $T^{\rho,(t+1)} = T$ with probability:

$$\min\Bigg(1, \frac{q_\rho(T|T^{\rho,(t)})}{q_\rho(T^{\rho,(t)}|T)} \frac{\hat{\phi}(T^{\rho,(t)}, T^{\kappa,(t)})}{\hat{\phi}(T, T^{\kappa,(t)})} \\ \times \exp\Bigg\{\sum_{k=1}^{K} \sum_{i \sim i'} B_{ii'}(T, \sigma_\rho^2) - B_{ii'}(T^{\rho,(t)}, \sigma_\rho^2) \mathbf{1}_{\{\rho_{ik}^{(t)} = \rho_{i'k}^{(t)}\}}\Bigg\}\Bigg),$$

otherwise set $T^{\rho,(t+1)} = T^{\rho,(t)}$.

2. Sample T from the proposal distribution $q_\kappa(\cdot|T^{\kappa,(t)})$. Set $T^{\kappa,(t+1)} = T$ with probability:

$$\min\Bigg(1, \frac{q_\kappa(T|T^{\kappa,(t)})}{q_\kappa(T^{\kappa,(t)}|T)} \frac{\hat{\phi}(T^{\rho,(t)}, T^{\kappa,(t)})}{\hat{\phi}(T^{\rho,(t)}, T)} \\ \times \exp\Bigg\{\sum_{k=1}^{K} \sum_{j \sim j'} D_{jj'}(T, \sigma_\kappa^2) - D_{jj'}(T^{\kappa,(t)}, \sigma_\kappa^2) \mathbf{1}_{\{\kappa_{jk}^{(t)} = \kappa_{j'k}^{(t)}\}}\Bigg\}\Bigg),$$

otherwise set $T^{\kappa,(t+1)} = T^{\kappa,(t)}$.

3. Update $\hat{\phi}^{(t+1)}$:

$$\log \hat{\phi}^{(t+1)}(t^\rho, t^\kappa) = \log \hat{\phi}^{(t)}(t^\rho, t^\kappa) + \gamma^{(t)} \left(\mathbf{1}_{\{(T^{\rho,(t+1)}, T^{\kappa,(t+1)}) = (t^\rho, t^\kappa)\}} - \frac{1}{mn}\right), \quad (3.6)$$

$(t^\rho, t^\kappa) \in \mathscr{T}_\rho \times \mathscr{T}_\kappa$.

4. Sample $\rho^{(t+1)}$ and $\kappa^{(t+1)}$ with the Swendsen-Wang algorithm.

5. Sample $(\sigma^{2,(t+1)}, \Theta^{(t+1)})$ using the usual Gibbs sampler.

In the step (3), we need how to choose $\gamma^{(t)}$, which is a random sequence of real numbers decreasing slowly to 0. We chose $\gamma^{(t)}$ according to the Wang-Landau approach. $\gamma^{(t)}$ is kept constant until the histogram of the temperatures is flat, that is, until $(T^{\rho,(t)}, T^{\kappa,(t)})$ has equiprobably visited all the values of the grid $\mathscr{T}_\rho \times \mathscr{T}_\kappa$. At the k^{th} recurrent time n_k such that $(T^{\rho,(t)}, T^{\kappa,(t)})$ is approximately uniformly distributed, we set $\gamma^{(n_k+1)} = \gamma^{(0)}/2^k$ where $\gamma^{(0)} = 1$. When γ_n becomes too small, γ_n is set to $0.0001/n^{0.7}$ as suggested by Atchade and Liu (2010).

In Step 5, the parameters (σ^2, Θ) are sampled with a Gibbs sampler. The full conditional posterior of the parameters (σ^2, Θ) is straightforward to derive (see for example (Chekouo and Murua, 2012)). Hence, it will not be spelled out here.

3.4 Experiments with Simulated Data

In this section, we show the results of a performance comparison between the Gibbs-plaid model and algorithm with the classical plaid Model of Turner et al. (2005), the algorithm of Cheng and Church (2000), and the Bayesian penalized plaid model of Chekouo and Murua (2012).

To build our simulated data we use the Yeast Cycle data of Cho et al. (1998). This data set shows the time-course fluctuation of the log-gene-expression levels of 6000 genes over seventeen time-points. The data have been analyzed by several researchers (Cho et al., 1998 ; 2002; Mewes et al., 1999 ;Tavazoie et al., 1999) and have been a classical example for testing clustering algorithms (Yeung et al., 2001). We use the five-phase subset of this data that consists of 384 genes whose expression levels peak at different time points that correspond to the five cell cycle phases (Cho et al., 1998). Of the 384 genes, there are 355 genes annotated with GO terms. Based on the Lin's pairwise similarities discussed in Section 3.2.2, we built a relational graph comprising the annotated genes. Since the Gibbs prior is of the form of the Potts model, we perform Potts model clustering [20, 21] in order to discover clusters of genes in the graph.

These were used to set the genes labels. Clusters that split at higher temperatures in the Potts model were use as candidates for overlapping biclusters. As with the real data, we consider seventeen simulated conditions. The relational condition graph was modelled inspired by the time-dependency in the Yeast Cycle data. This allowed us to consider biclusters formed by consecutive conditions. These are easier to visualize. The similarity between conditions was given by a set correlation of $\gamma = 0.8$ between consecutive conditions. The correlation distance between conditions was set to

$$d^{\kappa}(j,j') = \begin{cases} 2(1-\gamma^{|j-j'|}), & |j-j'| \leq 3, \\ 0 & \text{otherwise.} \end{cases}$$

The condition labels were sampled according to the Gibbs prior given by this graph. The expression levels of the bicluster cells were then generated as follows: μ_0 was generated from a Normal$(0, 0.05)$; μ_k was generated from a Normal$(2(k+1), 0.05)$, $k = 1, 2, ..., K$; the gene effects α_{ik} were generated according to their prior distribution with means set to $\mu_{\alpha_{ik}} = \frac{2}{1+\exp(-i)} - \frac{1}{r_k}\sum_i \frac{2}{1+\exp(-i)}$, $k = 1, \ldots, K$; the condition effects β_{jk} were generated similarly; and the variance σ^2 was generated from an Inverse-$\chi^2(3, 0.03)$. We created in this fashion data sets with number of biclusters $K = 2, 3, 4, 5, 6, 7, 8$. Figure 3.1 shows some examples of the simulated data for different values of K.

3.4.1 The F1-measure of performance

A measure of similarity between two sets of biclusters $M_1 = \{A_1, \ldots, A_k\}$ and $M_2 = \{B_1, \ldots, B_\ell\}$ is given by by the so-called F1-measure (Santamaria et al., 2007; Turner et al., 2005). The F1-measure is an average between *recall* and *precision*, two measures of retrieval quality introduced in the text-mining literature (Allan et al., 1998). Let A, B be two biclusters, r_A and r_B be the number of genes in A and B, c_A and c_B be the number of conditions in A and B, and $n_A = r_A c_A$ and $n_B = r_B c_B$ be the number of elements in A and B, respectively. Precision and recall are given by

$$\text{recall} = \frac{(r_{A\cap B})(c_{A\cap B})}{n_B}, \qquad \text{precision} = \frac{(r_{A\cap B})(c_{A\cap B})}{n_A}.$$

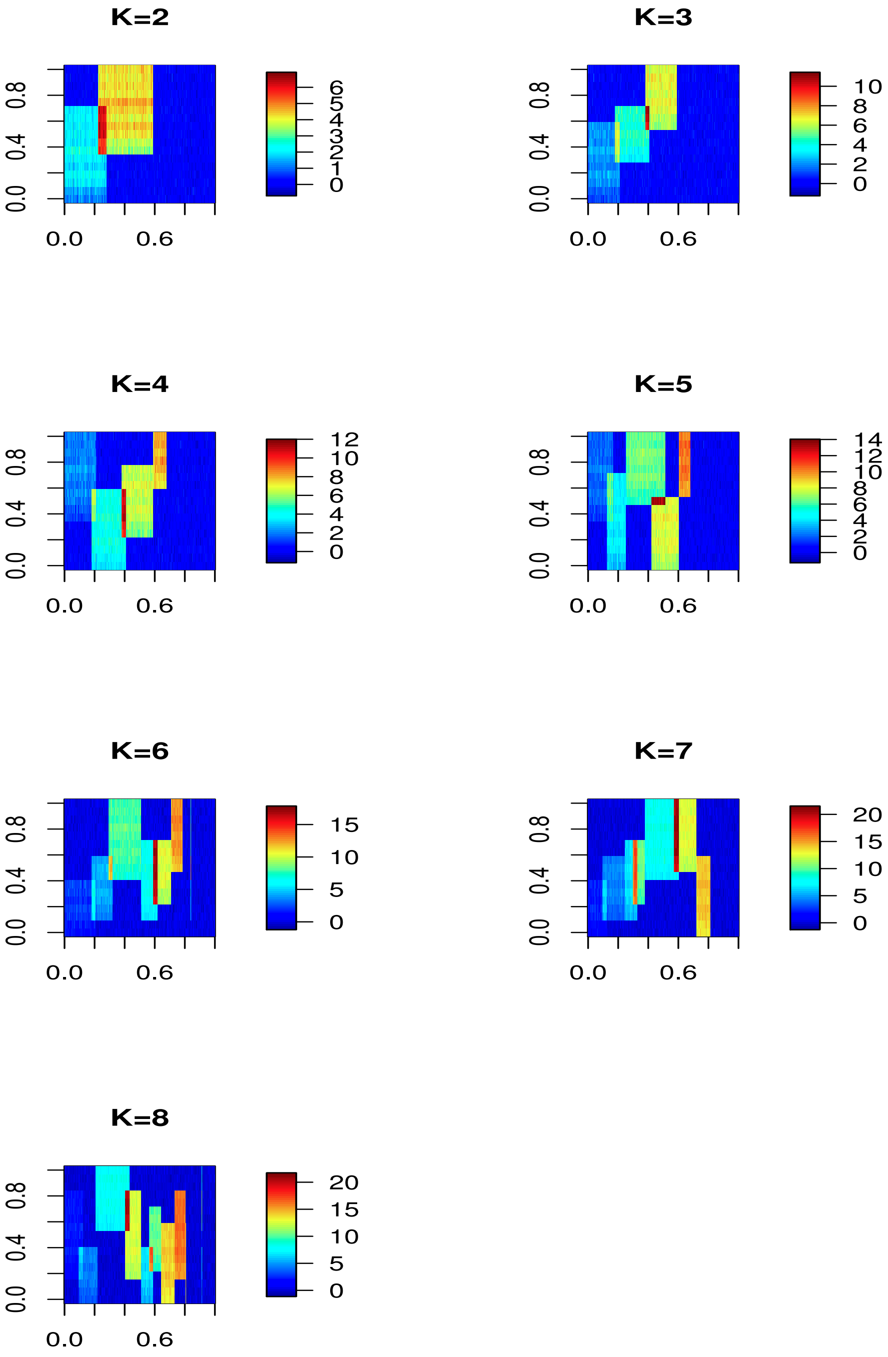


Figure 3.1: Examples of simulated data.

Recall is the proportion of elements in B that are in A. Precision is the proportion of elements in A that are also found in B. The F1-measure between A and B is given by $F_1(A,B) = 2(r_{A\cap B}) \times (c_{A\cap B})/(n_A + n_B)$. When several target biclusters M_1 are to be compared with known biclusters M_2, we use the F1-measure average: $F_1(M_1,M_2) = \frac{1}{k}\sum_{i=1}^{k} \max_j F_1(A_i,B_j)$.

3.4.2 Comparison of results

The Gibbs-plaid model was run with the stopping criterion suggested by Atchade and Liu (2010), but with a maximum number of iterations fixed at $500,000$. We used the last 2000 samples after the burn-in period to do the analysis. We compared our results with the ones obtained with two of the most popular algorithms for biclustering: the algorithm of Cheng and Church (2000), and the plaid model (Turner et al., 2005). The results are shown in Figure 3.2. Our results are far better that the others. We also compared with the Bayesian penalized plaid model of Chekouo and Murua (2012). In general, our model is the best performing one.

3.4.3 Choosing the number of biclusters

As in the work of (Chekouo and Murua, 2012), we used two model selection criteria to decide what is the appropriate number of biclusters for each data set. The criteria used were the Akaike information (AIC) (Akaike, 1974) and the conditional Deviance information (DIC$_c$) introduced in (Chekouo and Murua, 2012) is given by

$$DIC_c \quad = \quad -2E_{\sigma^2,\Theta,\rho,\kappa}\left[\log p(y|\sigma^2,\Theta,\rho,\kappa)|y\right] + p_c(\tilde{\sigma}^2,\tilde{\Theta},\tilde{\rho},\tilde{\kappa}).$$

where $(\tilde{\sigma}^2,\tilde{\Theta},\tilde{\rho},\tilde{\kappa})$ is the maximum a posteriori estimator of $(\sigma^2,\Theta,\rho,\kappa)$, and

$$p_c(\tilde{\sigma}^2,\tilde{\Theta},\tilde{\rho},\tilde{\kappa}) = -2E_{\sigma^2,\Theta,\rho,\kappa}\left[\log p(y|\sigma^2,\Theta,\rho,\kappa)|y\right] + 2\log p(y|\tilde{\sigma}^2,\tilde{\Theta},\tilde{\rho},\tilde{\kappa}),$$

is the corresponding effective dimension. Figure 3.3 shows the model selection results for some of the simulated data sets. We note that, in general, AIC and DIC$_c$ perform

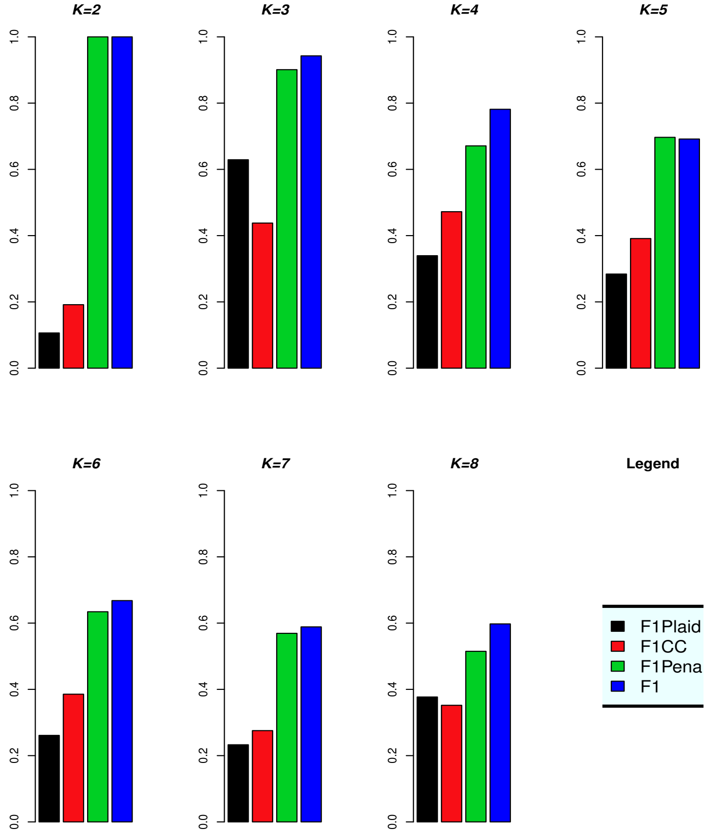


Figure 3.2: F1-measure means for the different models compared. The darker bars correspond to the classical plaid model, the red bars, to the algorithm of Cheng and Church, the green bars, to the Bayesian penalized plaid model, and the blue bars, to the Gibbs-plaid model.

very well. Most of the criteria reach a minimum at the true number of biclusters. In some cases the DIC$_c$ criterion reaches a minimum for a number of biclusters larger than the true number of biclusters. A closer look at the extra biclusters reveals that these are, in general, very small, containing only a couple of conditions or a handful of genes. A rule-of-thumb would be to select the biclustering model associated to the first minimum or near optimal value of DIC$_c$.

3.5 Applications

We applied our model to the Yeast Cycle data already described in Section 3.4. The data is displayed in Figure 3.4. The DIC$_c$ criterion chooses fourteen biclusters (see Figure 3.5). The size of the biclusters is shown in Table 3.1. Some biclusters are displayed in Figure 3.6. Most of the genes and the conditions (95%) were in more than one bicluster. Only one bicluster, Bicluster 7, contains down-regulated genes. Bicluster 7 also presents time-points with negative effects. This bicluster contains 100 genes and only the first two time-points. The significant genes in this bicluster are related to the nucleosome. All the others biclusters contain up-regulated genes and the time-points have positive effects. Bicluster 2 contains 58 genes and 8 time-points. The significant genes in this bicluster are related to the condensed chromosome kinetochore, the condensed nuclear chromosome and the centromeric region. The corresponding time-points are related to the S and G2 phases ([11]). It contains some genes which peak in both phases. Biclusters 1 and 3 are also characterized by functions related to cellular components MCM complex and cytoskeletal part, respectively. Bicluster 6 contains 118 genes that are up-regulated on the three last time-points. The main gene functions that characterize this bicluster are associated to the transporter activity and the substrate-specific transmembrane transporter activity (molecular functions); to the cellular ketone metabolic process, the oxidation-reduction process and to transport (cellular components); to the mitochondrial part (biological process). Bicluster 5 consists of 83 genes and 5 timepoints (1,9,10,11,17). Cellular respiration is the most significant biological process present in this bicluster. The main cellular component is the mitochondrial part. Finally, Bicluster

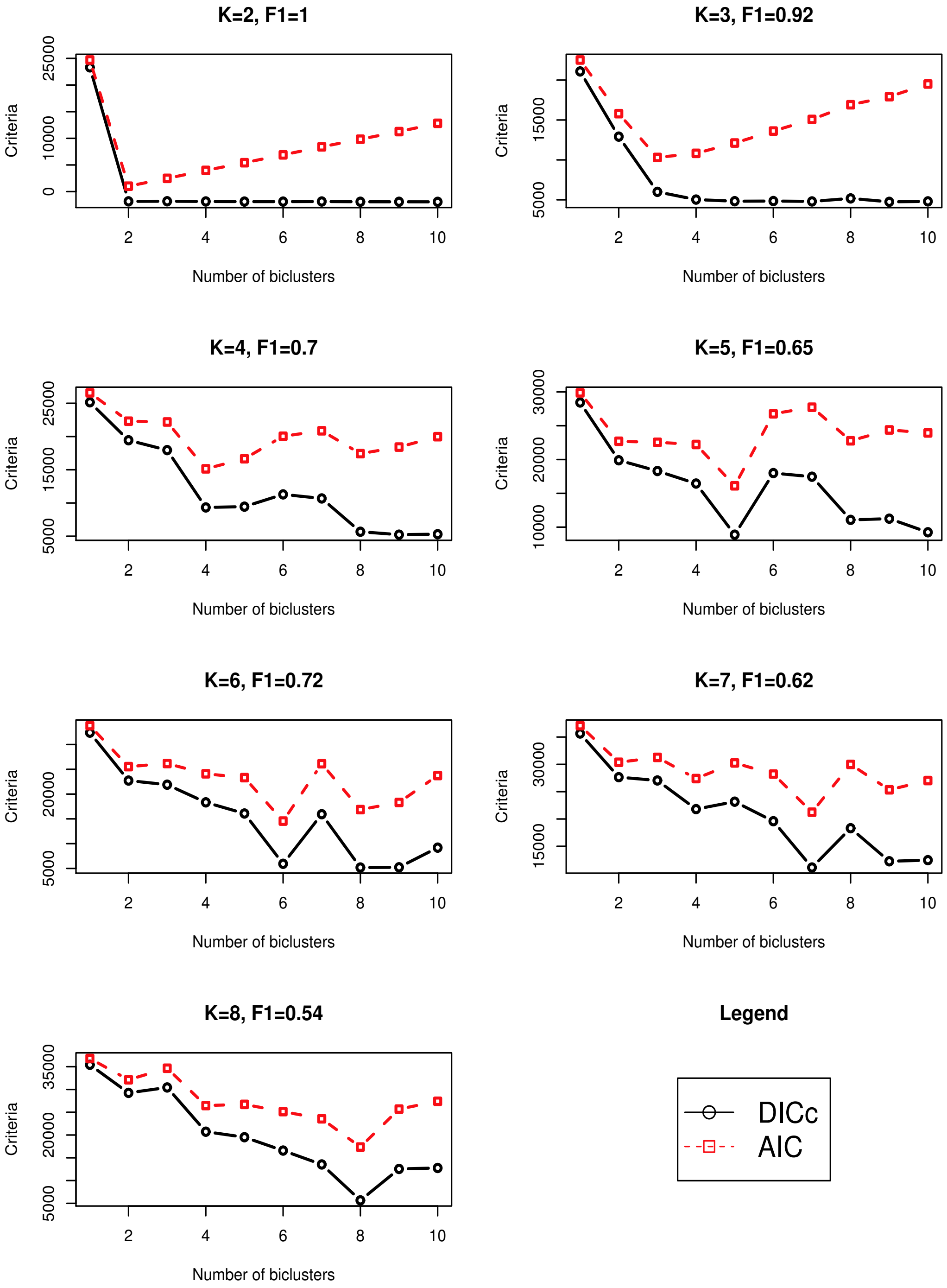


Figure 3.3: AIC and DIC$_c$ for the Gibbs-plaid model (p=355, q=17). The F1 measure refers to the F1 value between the true biclustering and that one associated to the biclustering that minimizes the DIC$_c$.

12 contains genes with functions related to the DNA repair and the cellular response to stress.

3.6 Conclusion

We proposed a model for biclustering that incorporates biological knowledge from Gene Ontology (GO) and experimental conditions (if available). This knowledge is used to specify prior distributions that account for the dependency structure between genes and between conditions. Our goal was to show that using prior information on the genes and the conditions helps improve the significance of the biclustering obtained from a biological point of view. We incorporated this prior information by efficiently modeling mutual interactions between genes (or conditions) with discrete Gibbs fields. The pairwise interaction between the genes is given by entropy similarities estimated from GO. These are embedded into a relational graph whose nodes correspond to genes, and edges to similarities. The graph is kept sparse by filtering out gene interactions (edges) coming from genes that do not share much common biological functionality as measured by GO. In some cases, the conditions may also be compared by building a notion of similarity between them, e.g., in the case of gene expression time courses. These similarities can also be represented by a corresponding relational graph. To our knowledge the introduction of Markov models and Gibbs fields in the context of biclustering seems new. However, this has already been attempted in the fields of clustering and regression.

In order to estimate the biclusters, we adopted a hybrid procedure that mixes Metropolis-Hastings with a variant of the Wang-Landau algorithm. To efficiently sample the labels through a block Gibbs sampling, we used an algorithm based on Swendsen and Wang algorithm. Preliminary results are very promising. Experiments on simulated data showed that our model is an improvement over other algorithms. They also showed that criteria based on our conditional DIC and AIC may be used to guide the choice of the number of biclusters.

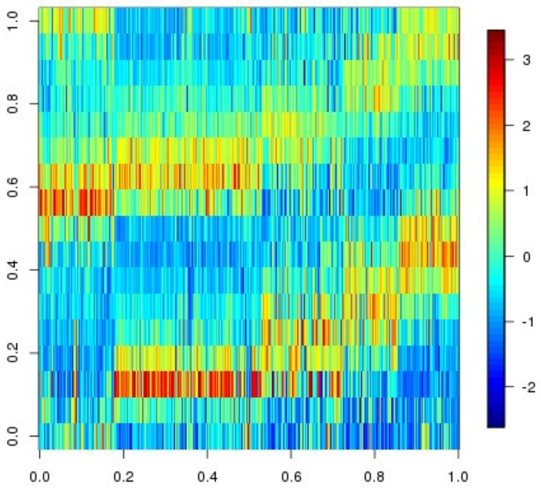


Figure 3.4: The gene expression levels of the Yeast Cycle data set.

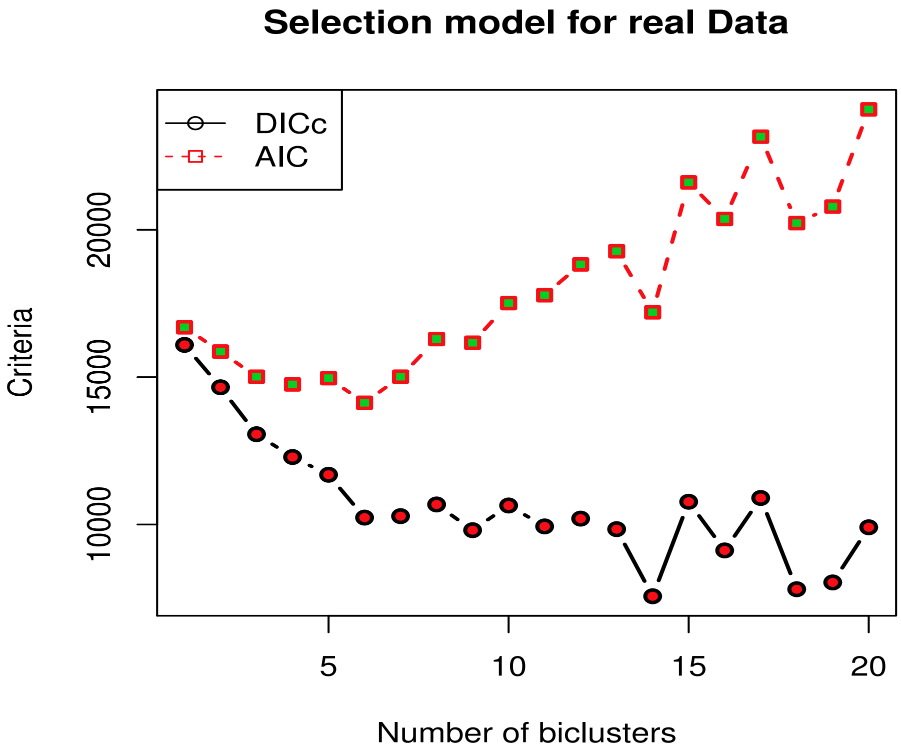


Figure 3.5: AIC and DIC$_c$ for the Gibbs-plaid model (p=355, q=17) applied to the Yeast Cycle data set.

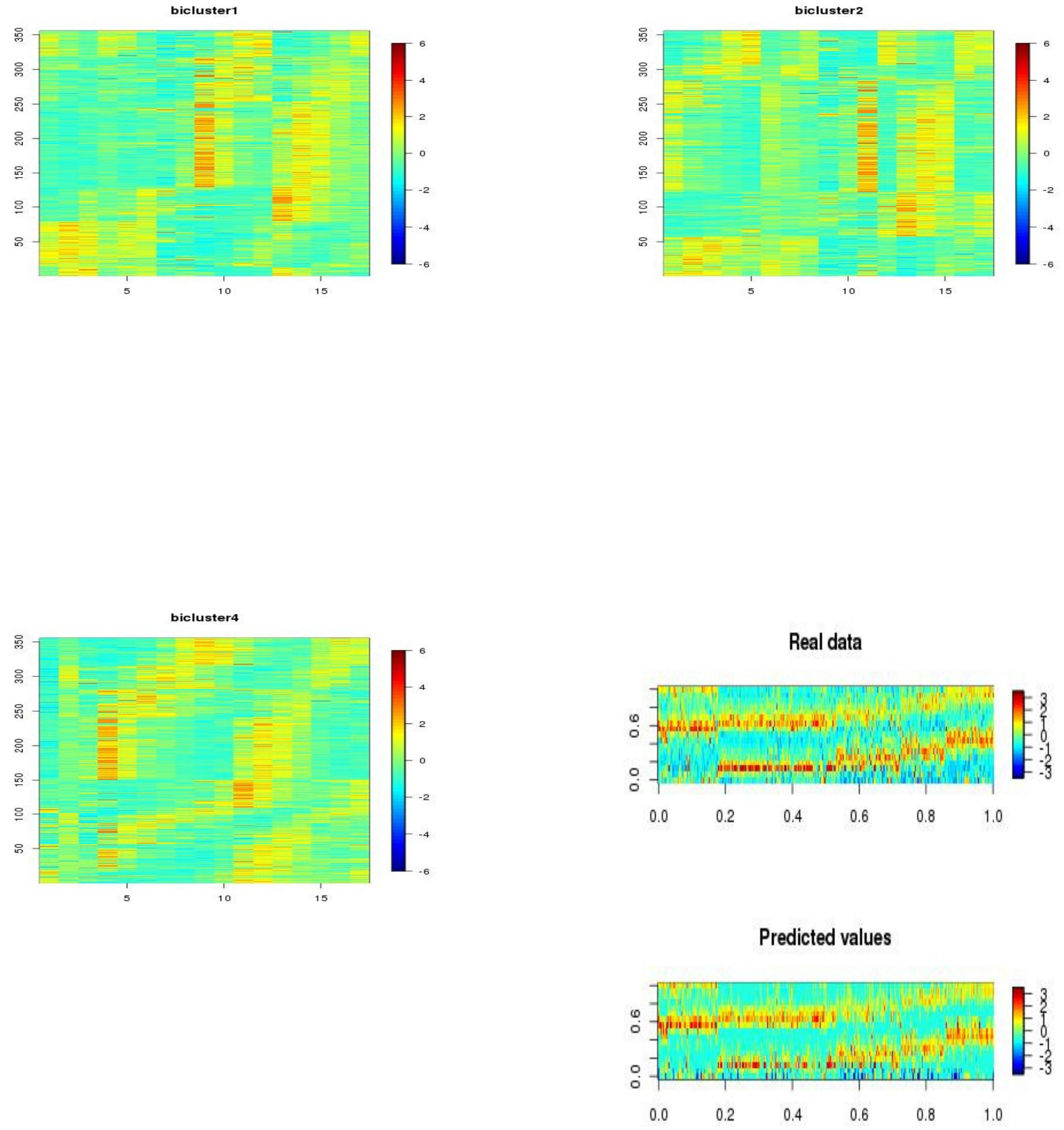


Figure 3.6: Yeast Cell Cycle Data. Biclusters 1, 2, and 4 seen at the bottom left corner of the first three images in contrast to the whole data matrix. The rightmost bottom panel corresponds to the original data and the fitted values predicted by the model.

Table 3.1: The size of the biclusters found in the Yeast Cycle data set

Bicluster	number of conditions	number of genes
1	6	79
2	8	58
3	5	147
4	2	109
5	5	83
6	3	118
7	2	100
8	2	30
9	3	216
10	2	26
11	2	179
12	3	41
13	3	149
14	3	105

BIBLIOGRAPHIE

[1] Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control 19*(6), 716–723.

[2] Allan, J., J. Carbonell, G. Doddington, J. Yamron, and Y. Yang (1998). Topic detection and tracking pilot study : Final report. In *In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 194–218.

[3] Ashburner, M., C. Ball, J. Blake, D. Bolsteing, H. Butler, J. Cherry, A. Davis, K. Dolinski, S. Dwight, J. Eppig, M. Harris, D. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. Matese, J. Richardson, M. Ringwald, G. Rubin, and G. Sherlock (2000). Geneontology : tool for the unification of biology the gene ontology consortium. *Nature Genetics 25*, 25–29.

[4] Atchadé, Y. F. and J. S. Liu (2010). The Wang-Landau algorithm in general state spaces : Applications and convergence analysis. *Statistica Sinica 20*(1), 1–26.

[5] Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological) 36*(2), 192–236.

[6] Besag, J. (2001). Markov chain Monte Carlo for statistical inference.

[7] Blatt, M., S. Wiseman, and E. Domany (1996, Apr). Superparamagnetic clustering of data. *Phys. Rev. Lett. 76*, 3251–3254.

[8] Caldas, J. and S. Kaski (2008). Bayesian biclustering with the plaid model. In J. Príncipe, D. Erdogmus, and T. Adali (Eds.), *Proceedings of the* IEEE *International Workshop on Machine Learning for Signal Processing* XVIII, pp. 291–296. IEEE.

[9] Chekouo, T. and A. Murua (2012). The penalized biclustering model and related algorithms. Submitted for publication.

[10] Cheng, Y. and G. Church (2000). Biclustering of expression data. *Int. Conf. Intelligent Systems for Molecular Biology 12*, 61–86.

[11] Cho, R. J., M. J. Campbell, E. A. Winzeler, L. Steinmetz, A. Conway, L. Wodicka, T. G. Wolfsberg, A. Gabrielian, D. Landsman, L. D. J., and R. W. Davis (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell 2*, 65–73.

[12] Frohlich, H., N. Speer, A. Poustka, and T. BeiSZbarth (2007). GOSim - an R-package for computation of information theoretic GO similarities between terms and gene products. *BMC Bioinformatics 8*(1), 166.

[13] Getz, G., E. Levine, E. Domany, and M. Q. Zhang (2000). Super-paramagnetic clustering of yeast gene expression profiles. *Physica A : Statistical Mechanics and its Applications 279*(1–4), 457 – 464.

[14] Geyer, C. J. and E. A. Thompson (1995). Annealing Markov chain Monte Carlo with applications to ancestral inference. *Journal of the American Statistical Association 90*(431), 909–920.

[15] Gu, J. and S. Liu (2008). Bayesian biclustering of gene expression data. *The Int. Conf. on Bioinformatics & Computational Biology, BMC Genomics 9*(1), 113–120.

[16] Hang, S., Z. You, and L. Y. Chun (2009). Incorporating biological knowledge into density-based clustering analysis of gene expression data. In *Proceedings of the 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery - Volume 05*, FSKD ’09, Washington, DC, USA, pp. 52–56. IEEE Computer Society.

[17] Kanehisa, M. and S. Goto (2000). KEGG : Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res. 28*, 27–30.

[18] Lazzeroni, L. and A. Owen (2002). Plaid models for gene expression data. *Statistica Sinica 12*, 61–86.

[19] Lord, P., R. Stevens, A. Brass, and C. Goble (2003). Semantic similarity measures as tools for exploring the gene ontology. *Pac. Symp. Biocomput. 8*, 601–612.

[20] Murua, A., L. Stanberry, and W. Stuetzle (2008). On Potts model clustering, kernel K-means and density estimation. *Journal of Computational and Graphical Statistics 17*(3), 629–658.

[21] Murua, A. and N. Wicker (2012). The conditional-Potts clustering model. Submitted for publication.

[22] Park, M. Y., T. Hastie, and R. Tibshirani (2007). Averaged gene expressions for regression. *Biostatistics 8*, 212–227.

[23] Prelic, A., S. Bleuler, P. Zimmermann, A. Wille, P. Buhlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler (2006). A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics 22*(9), 1122–1129.

[24] Resnik, P. (1999). Semantic similarity in a taxonomy : an informationbased measure and its application to problems of ambiguity in natural language. *J. Artif. Intelligence Res. 11*, 95–130.

[25] Santamaría, R., L. Quintales, and R. Therón (2007). Methods to bicluster validation and comparison in microarray data. In H. Yin, P. Tino, E. Corchado, W. Byrne, and X. Yao (Eds.), *Intelligent Data Engineering and Automated Learning - IDEAL 2007*, Volume 4881 of *Lecture Notes in Computer Science*, pp. 780–789. Springer Berlin / Heidelberg.

[26] Sara, C. M. and A. L. Oliveira (2004). Biclustering algorithms for biological data analysis : A survey. *IEEE Transactions on computational biology and bioinformatics 1*(1), 24–45.

[27] Sokal, A. D. (1996). Monte Carlo methods in statistical mechanics : Foundations and new algorithms. Lectures at the Cargese Summer School on “Functional Integration : Basics and Applications.

[28] Sokal, R. and C. Michener (1958). A statistical method for evaluating systematic relationships. *The University of Kansas Scientific Bulletin 38*(38), 1409–1438.

[29] Stingo, F. C., Y. A. Chen, M. G. Tadesse, and M. Vannucci (2011). Incorporating biological information into linear models : A Bayesian approach to the selection of pathways and genes. *Annals of Applied Statistics 5*(3), 1978–2002.

[30] Swendsen, R. H. and J.-S. Wang (1987, Jan). Nonuniversal critical dynamics in monte carlo simulations. *Phys. Rev. Lett. 58*, 86–88.

[31] Tanay, A., R. Sharan, and R. Shamir (2005). Biclustering algorithms : A survey. In *In Handbook of Computational Molecular Biology Edited by : Aluru S. Chapman & Hall/CRC Computer and Information Science Series*.

[32] Turner, H., T. Bailey, and W. Krzanowski (2005). Improved biclustering of microarray data demonstrated through systematic performance tests. *Computational Statistics & Data Analysis 48*, 235–254.

[33] Vannucci, M. and F. C. Stingo (2001). Bayesian models for variable selection that incorporate biological information. *Bayesian Statistics 9*, 659–678.

[34] Wang, F. and D. P. Landau (2001, Mar). Efficient, multiple-range random walk algorithm to calculate the density of states. *Phys. Rev. Lett. 86*, 2050–2053.

[35] Ward, J. H. (1963). Hierarchical groupings to optimize an objective function. *Journal of the American Statistical Association 58*, 234–244.

[36] Winkler, G. (2003). *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*, Volume 27. Springer.

[37] Zhang, J. (2010). A Bayesian model for biclustering with applications. *Journal of the Royal Statistical Society : Series C (Applied Statistics) 59*(4), 635–656.

CHAPITRE 4

VARIABLE SELECTION WITH THE PLAID MIXTURE MODEL FOR CLUSTERING

4.1 Introduction

Microarray data consist of many thousands of gene expression profiles but just about tens or hundreds of samples. These sorts of data are typical examples of high dimensional data for which the number of covariates (genes) is considerably larger than the sample size. Having so much information poses some problems on model selection. The decision on which data to keep or even look at becomes very relevant. For this reason, a classical way to start the analysis of high-dimensional data is with exploration analysis techniques such as clustering or biclustering. Both these techniques may be used for data compression and/or dimensionality reduction. However, in many situations clustering is the goal, for example to detect subtypes of a desease. In this case, having a sound and efficient methodology to perform variable selection is key to the advancement of the study of the desease. For example, in cancer research, only a few genes in the genome are supposed to contribute most to characterize cancer subtypes.

In the last few years, several authors have treated the problem of variable selection in the context of clustering. Tadesse et al. [24] formulated the clustering problem in a Bayesian context. In their model, the nondiscriminating variables follow a multivariate normal distribution, while the discriminating ones follow a multivariate normal mixture model with an unknown number of components. They use reversible jump Markov chain Monte Carlo techniques to define a sampler that moves between spaces of different dimensions. In their model, a binary exclusion/inclusion latent vector is introduced to indicate whether a variable is selected (i.e., discriminanting) or not. Another approach to do variable selection within a mixture model for clustering, described by Raftery and Dean [19], is the use of Bayes factors to do model selection. Raftery and Dean [19] propose a greedy search algorithm to find a local optimum in model space. They approximate the Bayes factors by the Bayes information criterion (BIC). Other authors (Kim et al. [13], Hoff [10]) have also introduced Bayesian variable selection methods through binary latent vectors to select the discriminating variables as in Tadesse et al. [24].

Another class of models for variable selection uses penalization methods for model-based clustering (Pan and Shen [18], Xie et al. [28], Wang and Zhu [26]). One of the most popular approaches among these latter methods is that of Pan and Shen [18] which is based on a penalized likelihood approach with an L_1 penalty term. Specifically, following Hoff [10], Pan and Shen [18] parametrize the cluster means, say μ_k, for each variable $j = 1, \ldots, q$, as $\mu_{jk} = \upsilon_j + \beta_{jk}$, where υ_j is the overall cluster-independent mean for variable j. They infer that if $\beta_{jk} = 0$ for all clusters k, then the variable j is non-informative for clustering (at least in terms of the mean). The model is fitted with an EM algorithm.

Here, we propose a novel method to select the variables in the context of clustering. This method is inspired by the plaid model (Lazzeroni and Owen [14]) in the context of biclustering. A biclustering is a simultaneous clustering of the observations (rows) and the variables (columns) of a data matrix. The biclusters obtained are submatrices where the rows exhibit a similar pattern across a subset of columns and vice-versa. The works of Madeira and Oliveira [16], Tanay et al. [25], and also Chekouo and Murua [5] give nice reviews on the topic. The key is to realize that when the same subset of columns is selected on each bicluster, then what we have really obtained is a clustering of the observations given by a selected subset of discriminating variables.

Let $\mathscr{Y} = \{y_1, y_2, \ldots, y_i, \ldots, y_p\} \subset \mathbf{R}^q$ be a random sample of p observations. Assume that the data has a structure consisting of K clusters. We introduce the latent variables $\rho = \{\rho_{ik}\}_{i=1,k=1}^{p,K}$ and $\kappa = \{\kappa_j\}_{j=1}^{q}$, so that

$$\rho_{ik} = \begin{cases} 1, & \text{iff the } i\text{-th row is in bicluster } k \\ 0, & \text{otherwise;} \end{cases}$$

$$\kappa_j = \begin{cases} 1, & \text{if the } j \text{ variable is discriminating,} \\ 0, & \text{otherwise.} \end{cases}$$

We will also use the notation $\rho_i = \{\rho_{ik}\}_{i=1}^{p}$. Our variable selection model for clustering is defined as follows. Given the overall memberships ρ, κ, the expectation of y_{ij} is written as a sum of layers or plaids

$$E(y_{ij}|\rho_i, \kappa_j) = \kappa_j \sum_{k=0}^{K} \rho_{ik}(\mu_k + \alpha_{ik} + \beta_{jk}) + (1 - \kappa_j)\upsilon_j,$$

where μ_k is the overall mean of the objects in cluster k, β_{jk} is the effect of the j-th variable in cluster k, and α_{ik} is the random effect in cluster k associated to the i-th observation. For identifiability purposes, we impose the constraints $\sum_{i\in k} \alpha_{ik} = \sum_{j\in k} \beta_{jk} = 0$, $k = 1,..,K$. Each plaid corresponds to a cluster. Note that the usual mixture model may be written as

$$E(y_{ij}|\rho_i, \kappa_j, k) = \kappa_j(\mu_k + \beta_{jk}).$$

Therefore, our model differs from other variable selection models based on mixtures in that:

(A) We consider the possibility that some observations are not well explained by the main clusters, but rather lie in what we called the zero-cluster ($k = 0$). These observations satisfy the constraints $\rho_{i0} = \prod_{k=1}^{K}(1 - \rho_{ik})$, $\alpha_{i0} = \beta_{j0} = 0$, for all $j = 1,\ldots,q$. Note that υ_j is the background or zero-cluster mean of variable j. The presence of this cluster may be justified by some observations in real life datasets. In clustering there is often a "ragbag" cluster for subjects that do not belong to any well defined cluster and are considered as noisy-individuals. Hence, it is desirable to consider a model which can leave a few points un-clustered if necessary.

(B) We incorporate observation random effects. Thus, as in the biclustering, the observations and the variables play a symmetrical role in each cluster. One particularity of our approach is that when $K = 1$, the problem of variable selection is equivalent to searching for a single bicluster (a submatrix) in the data matrix. Not only the random effects take into account the potential influence of single observations in the model, but they also introduce compound symmetry in the variance-covariance matrix associated to observations given the clusters. When this is not appropriate for a dataset at hand, then

one could either simply eliminate the random effects from the model, or consider them as fixed effects. For example, in the case of gene expression data, the effect of each gene (the observations) is of interest, so it makes sense to incorporate fixed gene effects in the model (as opposed to gene random effects) and to avoid imposing compound symmetry. In the present paper we work with the observation fixed effects assumption.
(C) The observations may be explained by more than a single cluster ($\sum_{k=1}^{K} \rho_{ik} \geq 1$). This yields an aggregate superposition of clusters which is different from a distributional overlapping of clusters as in the usual mixture model. For example, in clinical applications (Bhattacharya [3]), a patient may belong to more than one clinical group, i.e., a patient who complains of headache may have migraine symptoms and other causes of headache (such as nasal or sinus problems/desease). Many authors in the literature have worked on overlapping clustering (Fu and Banerjee [6], Fu and Banerjee [7] and Heller and Ghahramani [8]). Their models, which are motivated by the product-of-experts model (Hinton [9]), are often called *multiplicative mixture models*. We will show later on that our approach is related to the work of these authors.

Since our model, similarly to many of the other models for clustering in the literature, involves latent labels ρ, κ, we use a stochastic version of the EM algorithm based on the so-called Monte Carlo EM (MCEM) algorithm (Wei and Tanner [27]) to estimate the parameters. This is a modified EM algorithm where the expectation in the E-step is computed numerically through Monte Carlo simulations. We do the Monte Carlo sampling in each iteration of the MCEM algorithm with a Gibbs sampler. However, as suggested by Levine and Casella [15], we also use Importance Sampling to overcome the computational cost of the MC sampling at each step of the EM algorithm.

4.2 The plaid mixture model

In this section and throughout the paper we keep the notation introduced in the previous section. Inspired by the biclustering plaid model of Lazzeroni and Owen [14], we propose a general model for variable selection in the context of clustering. Our model comprises the clustering label parameters ρ, the variable selection parameters κ, the variance parameters $\Sigma = (\sigma^2, \{\sigma^2_{jk}\}_{j=1,k=0}^{p,K})$, and the mean parameters $\Psi = (\mu, \{\mu_k\}_{k=0}^{K}, \beta, \alpha)$, with $\alpha = \{\alpha_{ik}\}_{i=1,k=0}^{q,K}$ and $\beta = \{\beta_{jk}\}_{j=1,k=0}^{q,K}$. It is given by

$$y_{ij} = \kappa_j(\sum_{k=0}^{K}(\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik} + \eta_{ij}) + (1-\kappa_j)(\upsilon_j + \varepsilon_{ij}), \tag{4.1}$$

where the η_{ij}'s and ε_{ij}'s are assumed to follow independent zero-mean normal distributions. The variance of ε_{ij} is σ^2_{oj}. The variance of η_{ij} is the harmonic mean of the variances σ^2_{jk} –which could depend on the cluster k and the variable j– and is given by

$$\tau^2_{ij} = \sum_{k=0}^{K}\rho_{ik} / \sum_{k=0}^{K}(\rho_{ik}/\sigma^2_{jk}) = \left(\sum_{k=0}^{K}\rho_{ik}/r_i\sigma^2_{jk}\right)^{-1},$$

where $r_i = \sum_{k=0}^{K}\rho_{ik} \geq 1$ is the number of clusters that jointly explain observation y_i. This form of the variance allows us to cast our model as a multiplicative mixture model whose observation and component dependent variances are $r_i\sigma^2_{kj}$ (see equation (4.3) below).

Prior distribution.

The prior probability that variable j is selected is chosen to be the same for all $j = 1,\ldots,q$, and will be denoted by $\pi = P(\kappa_j = 1)$, any j. The prior probability that the i-th observation is explained by cluster k is denoted by $\pi_k = P(\rho_{ik} = 1)$ and it is supposed to be the same for all observations $i = 1,\ldots,p$. Let $\Pi = (\pi, \{\pi_k\}_{k=0}^{K})$ be the prior probability parameters. Furthermore, we assume that a priori the set of labels $(\{\rho_i\}_{i=1}^{p}, \{\kappa_j\}_{j=1}^{q})$ are mutually independent Bernoulli latent variables.

Likelihood.

In what follows, we will write μ_{ijk} for $\mu_k + \alpha_{ik} + \beta_{jk}$. Let $\theta = (\Sigma, \Psi, \Pi)$. The complete data likelihood is given by

$$
\begin{aligned}
L(\theta|\mathscr{Y}, \rho, \kappa) &= P(\mathscr{Y}|\rho, \kappa, \Sigma, \Psi) \prod_{i,k} {\pi_k}^{\rho_{ik}} (1-\pi_k)^{1-\rho_{ik}} \prod_j \pi^{\kappa_j} (1-\pi)^{1-\kappa_j} \\
&= \prod_{i,j} \left[\frac{1}{\tau_{ij}} \phi\left(\frac{y_{ij} - \sum_{k=0}^{K} \mu_{ijk}\rho_{ik}}{\tau_{ij}} \right) \right]^{\kappa_j} \left[\frac{1}{\sigma} \phi\left(\frac{y_{ij} - \upsilon_j}{\sigma} \right) \right]^{1-\kappa_j} \\
&\qquad \times \prod_{i,k} {\pi_k}^{\rho_{ik}} (1-\pi_k)^{1-\rho_{ik}} \prod_j \pi^{\kappa_j} (1-\pi)^{1-\kappa_j} \qquad (4.2)
\end{aligned}
$$

Let $\kappa^* = \{j : \kappa_j = 1, j = 1, \ldots, q\}$ be the set of the selected variables. One can show that the density of $\mathscr{Y}$ on the selected discriminating variables, that is $j \in \kappa^*$ is given by

$$
P(\mathscr{Y}|\rho, \kappa^*, \theta) = \prod_{i,j} \frac{1}{c_{ij}(\rho, \kappa^*, \theta)} \prod_{k=0}^{K} \left[\frac{1}{\sqrt{r_i}\sigma_{jk}} \phi\left(\frac{y_{ij} - \mu_{ijk} r_i \sigma_{jk}^2 / \tau_{ij}^2}{\sqrt{r_i}\sigma_{jk}} \right) \right]^{\rho_{ik}}, \qquad (4.3)
$$

where

$$
c_{ij}(\rho, \kappa^*, \theta) = \frac{\tau_{ij}\sqrt{2\pi}}{\prod_k (\sqrt{r_i}\sigma_{jk}\sqrt{2\pi})^{\rho_{ik}}} \exp\left\{ \frac{1}{2\tau_{ij}^2} \left(\sum_{k=0}^{K} \mu_{ijk}\rho_{ik} \right)^2 - \frac{1}{2\tau_{ij}^4} \sum_{k=0}^{K} r_i \mu_{ijk}^2 \rho_{ik} \sigma_{jk}^2 \right\}
$$

Equation (4.3) shows that our model is similar to the multiplicative mixture model for Overlapping Clustering described by Fu and Banerjee [6], Fu and Banerjee [7], and Heller and Ghahramani [8]. Within this model, the location and the scale parameters corresponding to cluster k are, respectively, $\mu_{ijk} r_i \sigma_{jk}^2 / \tau_{ij}^2$ and $r_i \sigma_{jk}^2$. Note that when there is no aggregate overlapping of clusters, i.e., $r_i = 1$ for all $i = 1, \ldots, p$, and ignoring the observation random effects, these parameters (μ_{ijk} and σ_{jk}^2) are, respectively, the mean and the variance of cluster k. Equation (4.3) is also related to the Product of Experts (PoE) of Hinton [9].

4.3 Estimation

The EM algorithm is particularly suitable for learning the parameters of our model (4.2) because the likelihood of the complete data $(\mathscr{Y}, \rho, \kappa)$ is much easier to calculate than the likelihood of the observed data $\mathscr{Y}$. More specifically, the EM algorithm starts with an initial guess $\theta^{(0)} = (\Sigma^{(0)}, \Psi^{(0)}, \Pi^{(0)})$ of the unknown parameters and iteratively aims at estimating the MLE $\theta^{\star}$. Each iteration consists of the expectation (E) step and the maximization (M) step.

4.3.1 The E-step

Given an estimate of θ at the current iteration t, say $\theta^{(t)}$, the conditional expectation of the complete data log-likelihood with respect to the density $P(\rho, \kappa | \mathscr{Y}, \theta)$ is computed in the E-step:

$$Q(\theta|\theta^{(t)}) = E\left(\log(P(\mathscr{Y}, \rho, \kappa|\theta))|\mathscr{Y}, \theta^{(t)}\right), \qquad t \geq 0. \tag{4.4}$$

Unfortunately, we cannot compute the exact expectation (4.4) since we do not have a tractable closed form expression for the joint conditional density $P(\rho, \kappa|\mathscr{Y}, \theta)$. However, since the full conditionals of ρ and κ are easily obtained, we propose to estimate $Q(\theta|\theta^{(t)})$ via a Monte Carlo EM (MCEM) algorithm [27]. The proposed estimator is given by

$$Q_m(\theta|\theta^{(t)}) = \frac{1}{m}\sum_{l=1}^{m} \log(P(\mathscr{Y}, \rho(l), \kappa(l)|\theta)), \tag{4.5}$$

where $\rho(l), \kappa(l)$, $l = 1, .., m$ are samples from the conditional joint distribution of the latent variables ρ, κ given the observed data $\mathscr{Y}$ and the current value of the parameters $\theta^{(t)}$. The estimator in (4.5) converges to the theoretical expectation in (4.4) by the law of large numbers. Below, we explain how to obtain the label samples via a Gibbs sampler.

4.3.2 The M-step

The M-step maximizes the sum (4.5) with respect to θ subject to the identifiability constraints $\sum_i \rho_{ik}\alpha_{ik} = \sum_j \kappa_j \beta_{jk} = 0$, for all i, j, k. To overcome the computational cost of performing MCMC sampling within the MCEM algorithm when m is large, Levine and Casella [15] propose to use instead importance sampling (Robert and Casella [21]). The algorithm is initialized by m samples, $\rho(l), \kappa(l)$, $l = 1,..,m$ from the joint distribution $P(\rho, \kappa|\mathscr{Y}, \theta^{(0)})$. At iteration t, we estimate $Q(\theta|\theta^{(t)})$ by importance sampling (IS):

$$Q_{IS,m}(\theta|\theta^{(t)}) = \frac{1}{\sum_{l=1}^{m} w_l^{(t)}} \sum_{l=1}^{m} w_l^{(t)} \log(P(\mathscr{Y}, \rho(l), \kappa(l)|\theta)) \tag{4.6}$$

where $w_l^{(t)} = P(\mathscr{Y}|\rho(l), \kappa(l), \theta^{(t)})/P(\mathscr{Y}|\rho(l), \kappa(l), \theta^{(0)})$. Thus, we do not need to obtain a new sample of m labels from $P(\rho, \kappa|\mathscr{Y}, \theta^{(t)})$ at each iteration t in order to estimate $Q(\theta|\theta^{(t)})$. The cost of obtaining a new sample of m labels at each iteration is higher than obtaining the IS weights. The weights are given by:

$$w_l^{(t)} = \prod_{i,j} w_l^{(t)}(i,j), \text{ with } w_l^{(t)}(i,j) = \frac{P(y_{ij}|\rho_i(l), \kappa_j(l), \theta^{(t)})}{P(y_{ij}|\rho_i(l), \kappa_j(l), \theta^{(0)})}. \tag{4.7}$$

4.3.3 The EM updating equations

Note that the identifiability constraints for the M-Step now become

$$\sum_i \rho_{ik}(l)\alpha_{ik} = \sum_j \kappa_j(l)\beta_{jk} = 0, \text{ for all } i, j, k, \text{ and } l = 1, \ldots, m. \tag{4.8}$$

If for a particular observation i, and cluster k one obtains $\rho_{ik}(l) = 0$ for all $l \in \{1, \ldots, m\}$, then, it is reasonable to set $\alpha_{ik} = 0$. This latter condition may be written as the identity $\alpha_{ik} \prod_l (1 - \rho_{ik}(l)) = 0$ for all i, k.

In order to simplify the estimation problem, we will assume that the variances σ_{jk}^2 do not depend on the cluster. Thus, $\sigma_{jk}^2 = \sigma_j^2$, for all $k = 0, \ldots, K$. Consequently, $\tau_{ij}^2 = \sigma_j^2$.

In what follows, for any function of the labels $f(\rho, \kappa)$, the Important Sampling weighted average $\sum_{l=1}^{m} w_l^{(t)} f(\rho(l), \kappa(l)) / \sum_{l=1}^{m} w_l^{(t)}$ will be denoted by

$E_{IS}(f(\rho,\kappa)|\mathscr{Y},\theta^{(t)})$. Consider the following quantities

$$\alpha'_{ik} = D_i \sum_j \frac{1}{2\sigma_j^2} \left\{ E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)})(y_{ij}-\beta_{jk}) - \sum_{k'\neq k} E_{IS}(\rho_{ik}\kappa_j\rho_{ik'}|\mathscr{Y},\theta^{(t)})\mu_{ijk'} \right\} - \mu_k$$

$$\beta'_{jk} = C_j \sum_i \left\{ E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)})y_{ij} - \sum_{k'\neq k} E_{IS}(\rho_{ik}\kappa_j\rho_{ik'}|\mathscr{Y},\theta^{(t)})\mu_{ijk'} \right\} - \mu_k.$$

and also

$$D_i^{-1} = \sum_j \frac{1}{2\sigma_j^2} E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)}), \qquad C_j^{-1} = \sum_i E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)}).$$

The maximization of (4.6) subject to the constraints in (4.8) yields the following EM updating equations:

$$\begin{aligned}
\alpha_{ik} &= \alpha'_{ik}(1-\prod_{l=1}^{m}[1-\rho_{ik}(l)]) - D_i \sum_{l=1}^{m} \lambda_{lk}^{(\alpha)} \rho_{ik}(l) \\
\beta_{jk} &= \beta'_{jk}(1-\prod_{l=1}^{m}[1-\kappa_j(l)]) - 2\sigma_j^2 C_j \sum_{l=1}^{m} \lambda_{lk}^{(\beta)} \kappa_j(l) \\
\mu_k &= [\sum_{i,j} E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)})/\sigma_j^2]^{-1} \\
&\qquad \times \sum_{i,j} \{E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)})(y_{ij}-\beta_{jk}) - \sum_{k'\neq k} E_{IS}(\rho_{ik}\kappa_j\rho_{ik'}|\mathscr{Y},\theta^{(t)})\mu_{ijk'}\}/\sigma_j^2 \\
\pi_k &= \sum_i E_{IS}(\rho_{ik}|\mathscr{Y},\theta^{(t)})/p \\
\upsilon_j &= \sum_i y_{ij}/p \\
\pi &= \sum_j E_{IS}(\kappa_j|\mathscr{Y},\theta^{(t)})/q
\end{aligned}$$

$$
\begin{aligned}
\sigma_j^2 &= [pE_{IS}(\kappa_j|\mathscr{Y},\theta^{(t)})]^{-1}\sum_i E_{IS}(\kappa_j[y_{ij}-\sum_k \rho_{ik}\{\mu_k+\alpha_{ik}+\beta_{jk}\}]^2\,|\mathscr{Y},\theta^{(t)}) \\
&= [pE_{IS}(\kappa_j|\mathscr{Y},\theta^{(t)})]^{-1} \\
&\qquad \times \sum_i \{E_{IS}(\kappa_j|\mathscr{Y},\theta^{(t)})y_{ij}^2 - 2\sum_k E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y},\theta^{(t)})\mu_{ijk}y_{ij}+ \\
&\qquad + \sum_{k,k'} E_{IS}(\rho_{ik}\kappa_j\rho_{ik'}|\mathscr{Y},\theta^{(t)})\mu_{ijk}\mu_{ijk'}\} \\
\sigma_{oj}^2 &= \sum_i (y_{ij}-\upsilon_j)^2/p.
\end{aligned}
$$

The Lagrange multipliers λ_{lk} involved in the updating equations for the variable and random effects verify the following equations for $l = 1, ..., m$:

$$
\lambda_{lk}^{(\alpha)}\sum_{i=1}^{p} D_i\rho_{ik}(l) = \sum_i \rho_{ik}(l)\alpha'_{ik} - \sum_{l'\neq l}\lambda_{l'k}^{(\alpha)}\sum_{i=1}^{p} D_i\rho_{ik}(l)\rho_{ik}(l')
$$

$$
\lambda_{lk}^{(\beta)}\sum_{j=1}^{q} 2\sigma_j^2 C_j\kappa_j(l) = \sum_j \kappa_j(l)\beta'_{jk} - \sum_{l'\neq l}\lambda_{l'k}^{(\beta)}\sum_{j=1}^{q} 2\sigma_j^2 C_j\kappa_j(l)\kappa_j(l')
$$

Note that in practice, one may have $\rho_{ik}(l) = \rho_{ik}(l')$ for all observations $i = 1, \ldots, p$ for a couple of samples l, l' (that is, the membership in cluster k is the same for these two samples), or $\kappa_j(l) = \kappa_j(l')$ for all variables j for a couple of samples l, l'. If this is the case, then we do not have necessarily m constraints for α_k (or β_k) , and we only need to consider the multiplier $\lambda_{lk} + \lambda_{l'k}$ instead of the two multipliers λ_{lk} and $\lambda_{l'k}$ separately.

4.3.4 Sampling the labels

As we previously mentioned, the joint density of the labels $P(\rho, \kappa|\mathscr{Y}, \theta^{(t)})$ is not known in closed form. Therefore we cannot perform the Monte Carlo sampling of the labels (ρ, κ) required to compute $Q_{IS,m}(\theta|\theta^{(t)})$. However, we can obtain a Markov chain Monte Carlo (MCMC) estimate of this quantity. This is carried out with a Gibbs sampler, since the full marginal conditionals of the labels are known. For $i \in \{1, \ldots, p\}$ and $k \in \{0, \ldots, K\}$, let $\rho_{i0}^{(k)} = \prod_{k'\neq k}(1-\rho_{ik'})$, and $\rho_{-ik} = \rho_k \setminus \{\rho_{ik}\}$. The labels ρ_i for each $i = 1, ..., p$ and κ_j for each $j = 1, .., q$ are generated independently according to the following density equations

$$\frac{P(\rho_{ik}=1|\mathscr{Y},\rho_{-ik},\kappa,\theta)}{P(\rho_{ik}=0|\mathscr{Y},\rho_{-ik},\kappa,\theta})=$$

$$\exp\left\{\sum_{j=1}^{q}\frac{\kappa_j}{2\sigma_j^2}(\mu_{ijk}-\mu_0\rho_{i0}^{(k)})(2y_{ij}-2\sum_{k'\neq k}\mu_{ijk'}\rho_{ik'}-\mu_0\rho_{i0}^{(k)}-\mu_{ijk})\right\}\frac{\pi_k}{1-\pi_k} \tag{4.9}$$

$$\frac{P(\kappa_j=1|\mathscr{Y},\rho,\theta)}{P(\kappa_j=0|\mathscr{Y},\rho,\theta)}=\frac{\sigma^p}{\sigma_j^p}\exp\left\{\frac{-1}{2\sigma_j^2}\sum_{i=1}^{p}(y_{ij}-\sum_k\mu_{ijk}\rho_{ik})^2+\frac{1}{2\sigma^2}\sum_i(y_{ij}-\upsilon_j)^2\right\}\frac{\pi}{1-\pi} \tag{4.10}$$

In the case of non aggregate overlapping clusters, that is, $r_i = 1$ for all i, the Gibbs sampler uses instead

$$P(\rho_{ik}=1|\kappa,\theta)=\frac{A_{ik}}{\sum_{k=0}^{K}A_{ik}}, \qquad \text{and} \qquad P(\kappa_j=1|\rho,\theta)=\frac{B_{j1}}{B_{j0}+B_{j1}},$$

where:

$$\begin{aligned}
A_{ik} &= \prod_j\left[\frac{1}{\sigma_{kj}}\phi\left(\frac{y_{ij}-\mu_k-\alpha_{ik}-\beta_{jk}}{\sigma_{kj}}\right)\right]^{\kappa_j}\pi_k,\\
B_{j1} &= \prod_{i,k}\left[\frac{1}{\sigma_{kj}}\phi\left(\frac{y_{ij}-\mu_k-\alpha_{ik}-\beta_{jk}}{\sigma_{kj}}\right)\right]^{\rho_{ik}}\pi, \qquad \text{and}\\
B_{j0} &= \prod_i\left[\frac{1}{\sigma}\phi\left(\frac{y_{ij}-\mu}{\sigma}\right)\right](1-\pi).
\end{aligned}$$

4.3.5 Increasing the IS size m

As pointed out in Robert and Casella [21], the importance sampling estimator (4.6) would be inaccurate if the initial parameter values $\theta^{(0)}$ were poor. In addition, the estimator would take a long time to converge. Hence, as suggested by Levine and Casella [15], we obtain MCMC samples from $P(\rho,\kappa|\mathscr{Y},\theta^{(t)})$ for the first few iterations. The choice of the MCMC sample size m is an issue within the MCEM algorithm, since we do not want to use a large m when $\theta^{(t)}$ is far from the true MLE $\hat{\theta}$. The trade-off between the computational cost and the accuracy of the estimator of $Q(\theta|\theta^{(t)})$ could be resolved by increasing the sample size m as $\theta^{(t)}$ approaches the true MLE during the progression of the EM algorithm. This is what Booth and Hobert [4] do within the context of generalized linear mixed models. In their procedure, the increase in m obeys a schedule induced by a simple confidence region test: at the $(t+1)th$ iteration of the MCEM, an approximate $100(1-\alpha)\%$ confidence ellipsoid for $\hat{\theta}^{(t+1)} = \arg\max_\theta Q(\theta|\theta^{(t)})$ is constructed using the central limit theorem (CLT); if the previous estimate of the parameter $\theta^{(t)}$ lies in this region, then the the procedure declares that "the EM-Step was *swamped* by Monte Carlo error" and the number of simulations, m, is increased. We note that this schedule is based on true Monte Carlo samples, whereas in our case, we use MCMC samples. Unfortunately, the dependency between the MCMC samples do not allow us to use directly the CLT to construct a confidence interval. However, we overcome this limitation by borrowing some ideas from [15, 22] to limit the effect of the correlation between successive samples. We choose a sequence u_r, $r = 1,...,N$ such that $u_r - 1 \sim \text{Poisson}(\nu_r)$ where, $\nu_r = \nu r^d$ for some $\nu \geq 0$ and $d > 0$. The sums $l_r = \sum_{j=1}^{r} u_r$ are used as the subsampling points, and N, the number of such subsamples taken from the m samples, is set to $\sup\{r : l_r \leq m\}$. Using these subsamples, one gets an estimator of $Q^{(1)}(\theta|\theta^{(t-1)}) = (\partial/\partial\theta)Q(\theta|\theta^{(t-1)})$ evaluated at $\theta^{(t)}$

$$Q^{(1)}_{IS,m}(\theta^{(t)}|\theta^{(t-1)}) = \sum_{r=1}^{N} w_{l_r}^{(t-1)} \frac{\partial}{\partial\theta} \log P(\rho(l_r), \kappa(l_r), \mathscr{Y}|\theta^{(t)}) / \sum_{r=1}^{N} w_{l_r}^{(t-1)}.$$

Having obtained $\theta^{(t+1)}$, following the procedure described by Levine and Casella [15], we construct a confidence interval (CI) for each of the components of the vector $Q^{(1)}(\theta|\theta^{(t)})$ by evaluating its mean and variance estimates

$$\begin{aligned}
\hat{\mu}_m(\theta) &= [\sum_{l=1}^{m} w_l^{(t)}]^{-1} \sum_{l=1}^{m} w_l^{(t)} \frac{\partial}{\partial\theta} \log P(\rho(l), \kappa(l), \mathscr{Y}|\theta), \\
\hat{v}_m(\theta) &= -\hat{\mu}_m \hat{\mu}_m^T + \\
&\quad [\sum_{l=1}^{m} w_l^{(t)}]^{-1} \sum_{l=1}^{m} w_l^{(t)} \left(\frac{\partial}{\partial\theta} \log P(\rho(l), \kappa(l), \mathscr{Y}|\theta) [\frac{\partial}{\partial\theta} \log P(\rho(l), \kappa(l), \mathscr{Y}|\theta)]^T \right),
\end{aligned}$$

at $\theta = \theta^{(t+1)}$ (here the superscript ^{T} stands for matrix transposition). The CI for the j-th component is given by $\hat{\mu}_{mj}(\theta^{(t+1)}) \pm z_\gamma \sqrt{\hat{v}_{m,jj}(\theta^{(t+1)})}$, where z_γ is the percentile of order γ of the standard Normal distribution (usually $\gamma = 0.95$). The importance sample size m is increased if any of the components of the vector $Q^{(1)}_{IS,m}(\theta^{(t)}|\theta^{(t-1)})$ lies in the corresponding CI.

4.3.6 The algorithm

1. Initialize m, and $\theta^{(0)} = (\Sigma^{(0)}, \Psi^{(0)}, \Pi^{(0)})$.
   Set $t = 0$.

2. Generate m label samples $\rho(l)$, $\kappa(l)$, $l = 1,..,m$ using the Gibbs sampler according to equations (4.9) and (4.10).

3. Compute the importance weights $w_l(i, j)$ for all i, j using the equation (4.7).

4. **E-step:** Estimate $Q(\theta|\theta^{(t)})$ by:

$$E_{IS}(\kappa_j|\mathscr{Y}, \theta^{(t)}) = \sum_{l=1}^{m} w_l \kappa_j(l) / \sum_{l=1}^{m} w_l, \tag{4.11}$$

$$E_{IS}(\rho_{ik}|\mathscr{Y}, \theta^{(t)}) = \sum_{l=1}^{m} w_l \rho_{ik}(l) / \sum_{l=1}^{m} w_l, \tag{4.12}$$

$$E_{IS}(\rho_{ik}\kappa_j|\mathscr{Y}, \theta^{(t)}) = \sum_{l=1}^{m} w_l \rho_{ik}(l)\kappa_j(l) / \sum_{l=1}^{m} w_l. \tag{4.13}$$

5. **M-step:** Maximize $Q_m(\theta|\theta^{(t)})$ over θ to get $\theta^{(t+1)}$ through the EM updating equations given in Section 4.3.3.

6. **MC error** Perform the tests described in Section 4.3.5. If any of the tests in negative, i.e., if any of the components of the vector $Q^{(1)}_{IS,m}(\theta^{(t)}|\theta^{(t-1)})$ lies in the corresponding CI, then

   (a) Set $m_0 = m$.

   (b) Set $m = m_0 + \lfloor m_0/c \rfloor$ where $c = 3$ in our simulations.

   (c) Generate new labels $\rho(l), \kappa(l), l = m_0 + 1, ..., m$ via Gibbs sampler.

7. Set $t = t + 1$. Repeat steps 3 through 6 until convergence.

As mentionned earlier, if the initial value $\theta^{(0)}$ is poor, that is, if $P(\rho,\kappa|\mathscr{Y},\theta^{(0)})$ is far from $P(\rho,\kappa|\mathscr{Y},\theta^{*})$, then the algorithm will take a long time to converge. Thus, in our simulations, we include a burn-in period in Step 1 above, so that at each burn-in iteration we estimate $Q_m(\theta|\theta^{(t)})$ via MCMC instead of by IS. Therefore, our computations during the burn-in period behave like the MCEM algorithm described by McCulloch [17].

4.4 Model selection

We propose a modified BIC criterion (Schwarz, 1978) to perform model selection within our multiplicative plaid mixture model:

$$\text{BIC}_{plaid} = -2\log L(\hat{\theta}|\mathscr{Y}) + d_e \log(p)$$

where $L(\hat{\theta}|\mathscr{Y})$ is the likelihood of the incomplete data, $\hat{\theta}$ is the maximum likelihood estimator, and $d_e = d - s$ is the effective number of parameters, which is given by the difference between d, the total number of parameters, and s, the number of non-informative parameters, that is

$$\begin{aligned} s = {} & \text{Card}\{(i,k) : \alpha_{ik} = 0\} + \text{Card}\{(j,k) : \beta_{jk} = 0\} \\ & + \text{Card}\{j : \kappa_j = 0 \text{ (associated to } \sigma_j^2 \text{ for variables excluded from the model)}\} \\ & \qquad + 2\,\text{Card}\{j : \kappa_j = 1 \text{ (associated to } \upsilon_j = 0 \text{ and } \sigma_{oj}^2)\}, \end{aligned}$$

where Card stands for the cardinality of the set. This definition of BIC is inspired by that of Pan and Shen [18] for penalized model-based clustering with variable selection. We use it as a goodness-of-fit criterion to select an appropriate number of clusters K. The optimal K is the one that maximizes BIC_{plaid}. Note that our BIC_{plaid} is the analog of the usual BIC used in model-based clustering, since only those parameters actually used in the model are considered in the penalty term. The term $L(\hat{\theta}|\mathscr{Y})$ is intractable, since it involves the sum of all possible combinations of label values. So, in order to compute BIC_{plaid}, we use an estimate of $L(\hat{\theta}|\mathscr{Y})$ derived by importance sampling. This is given by $L_{IS}(\hat{\theta}|\mathscr{Y}) = \sum_{l=1}^{m} w_l P(\mathscr{Y}, \rho(l), \kappa(l)|\hat{\theta}) / \sum w_l$. Therefore, in our experiments we use

$$\mathrm{BIC}_{IS,plaid} = -2\log L_{IS}(\hat{\theta}|\mathscr{Y}) + d_e \log(p).$$

4.5 Comparison of methods by simulation

In this section, we illustrate the effectiveness of our method by conducting a simulation study with two different data scenarios. The first one is a mimicry of the synthetic data described by Pan and Shen [18] with $K = 1$ and no aggregate overlapping clusters. The second scenario concerns four synthetic data built with $K = 1,2,3,4$, respectively. For each $K \geq 2$, we simulate data with some aggregate overlapping clusters. We note that by definition for $K = 1$ there is not possible overlapping among the clusters. We apply to versions of our model to the simulated data. The first one assumes that there is some aggregate overlapping clusters The second version assumes there is no aggregate overlapping at all. We will refer to these two versions of our model as Plaid-Full and Plaid-Restricted. We compare the performance of our model with that of the LASSO-type L_1-penalization method of Pan and Shen [18], and the Gaussian model-based clustering greedy search (GS) method of Raftery and Dean [19]. We will refer to these methods as L_1-Penalty and GMBC-GS, respectively. The L_1-Penalty of Pan and Shen [18] penalizes the L_1-norm of the clusters means so as to obtain sparseness in the mean vectors. In this approach a zero component across all cluster means corresponds to a variable not being selected. The GMBC-GS variable selection of Raftery and Dean [19] models the non-discriminating variables as cluster-independent multivariate Normal variables. The algorithm uses Bayes factor estimates given by BIC to first filter out most models from the final search, and then to select the best model among those that pass the filter. We used the code published by Zhou [29], and the package *clustvarsel* to run the L_1-penalty and the GMBC-GS method, respectively. It is important to remember that our clustering model contains $K+1$ clusters including the zero-cluster. Thus, if another clustering method selects, say two clusters, then the corresponding K for comparison with our model is $K = 1$.

4.5.1 Simulated data

4.5.1.1 Scenario 1

In the first scenario, we closely followed the simulation done in Pan and Shen [18] so as to be able to compare our results with those given by the L_1-Penalty method. A two-cluster 1000-dimensional data set with a hundred observations is generated. Eight-five observations live in the first cluster; the remaining fifteen live in the second cluster. Only the first 150 variables are discriminating variables for clustering. More specifically, the first 150 variables were independent and identically distributed (iid) generated as

$$y_{ij} \sim I_{\{1\leq i\leq 85\}}N(0,1)+I_{\{86\leq i\leq 100\}}N(1.5,1),$$

whereas the remaining 850 variables were all iid $N(0,1)$. Since these data do not present fixed effects in the response, the first as well the second cluster may be considered as the zero-cluster of our multiplicative plaid mixture model.

4.5.1.2 Scenario 2

In this scenario, we simulated data with a more complicated clustering structure. The data was generated according to our model. We generated fifty 1000-dimensional observations. Only the first 20 dimensions were discriminating for clustering. More specifically, for each K, the first 20 variables are independently distributed $N(\sum_{k=0}^{K}(\mu_k+\alpha_{ik}+\beta_{jk})\rho_{ik},\sigma_j^2)$, whereas the other 980 variables are all iid $N(\upsilon_j,\sigma^2)$, $i=1,...,50$. Let $A_k=\{i:\rho_{ik}=1\}$, $k=0,\dots,K$. For each $i\in A_k$, define $R_k(i)=\text{Card}\{i'\in A_k : i'\leq i\}$ be the "rank" of i in A_k. The observation effects are generated as follows

$$\alpha_{ik}=\begin{cases} 0, & \text{if } \rho_{ik}=0 \\ 2/(1+\exp\{-[R_k(i)-1]\}) & \\ \qquad -\left(\sum_{i',\rho_{i'k}=1} 2/(1+\exp\{-[R_k(i')-1]\})\right)/(\text{Card}\{A_k\}), & \text{if } \rho_{ik}=1 \end{cases}$$

The column effects β_{jk} were generated similarly. We set $\upsilon_j = 2/(1+\exp\{-[R(j)-1]\})$ where $R(j) = j-20$ is the "rank" of the variable j among the 980 non-informative variables. The mean μ_0 of the zero-cluster is set to zero. For $K=1$, $\mu_K = 1$. For $K=2$, $\mu_k = 3k+1$, $k = 1,..,K$. For $K=3$, $\mu_1 = 7, \mu_2 = -4, \mu_3 = 4$. For $K=4$, $\mu_1 = 4, \mu_2 = 7, \mu_3 = -2, \mu_4 = -4$. All variances are draws from an Inverse-χ^2 distribution with 3 degrees of freedom and scale equals to 0.1.

4.5.2 Results

The algorithm to fit our model were run with $m = 60$. We included a burn-in period of twenty samples. We set a maximum of 100 iterations for finding the optimal parameters. In practice, our algorithm converged in much fewer iterations. To get good starting values for any given K, we ran the MCEM algorithm multiple times with random starting values. In order to initialize the labels, we randomly start several K-means algorithms. To find initial values for the cluster labels ρ, we run K-means with $K+1$ clusters, and find a "good" zero-cluster among them. To initialize the variable labels κ, we also run K-means, but this time on the variables. We set $K=2$ and consider separately each one of these two clusters as possible initial selected variables. For any given K, we perform multiple runs of this procedure. Our final result is the one associated with the optimal run, that is, the one yielding the highest log-likelihood for the given K.

For each scenario, we simulated ten data sets and recorded the number of times that each method detected the true number of clusters. We also recorded the number of discriminating variables excluded from the model (Z_1), and the number of non-informative variables excluded from the model (Z_2). In order to measure the quality of the clustering estimated by the methods, we compared the estimated clustering with the true clustering of the data through the so-called F_1-measure. This is defined as the harmonic average between recall and precision, which are two measures of retrieval quality introduced in the text-mining literature [1]. Let A, B be two clusters , and r_A and r_B be the number of observations in A and B be the number of elements in A and B, respectively. Recall and precision are given by

$$\text{recall} = \frac{\text{Card}(A \cap B)}{\text{Card}(B)}, \qquad \text{precision} = \frac{\text{Card}(A \cap B)}{\text{Card}(A)}.$$

So, recall is the proportion of elements in B that are in A, and precision is the proportion of elements in A that are also found in B. The F1-measure between A and B is given by $F_1(A,B) = 2\,\text{Card}(A \cap B))/(\text{Card}(A) + \text{Card}(B))$. When an estimated clustering $M_1 = \{A_1, \ldots, A_k\}$ is to be compared with the true clustering $M_2 = \{B_1, \ldots, B_\ell\}$, we use the F1-measure average: $F_1(M_1, M_2) = \frac{1}{k}\sum_{i=1}^{k} \max_j F_1(A_i, B_j)$. We note that the more common measure of clustering quality, the adjusted Rand index [11, 20], is not properly defined for overlapping clusters. For this reason, the F_1-measure seems to be preferred in the literature. We computed the F_1-measure associated to the clustering of observations (F_1), and the F_1-measure associated to the selected variables (F_1^v). Their corresponding standard deviations are also reported (in brackets). F_1^v may be interpreted as a global measure of quality of the variable selection obtained. It can be written as $F_1^v = 2(q_0 - Z_1)/(q_0 - Z_1 + q - Z2)$ where q_0 is the true number of informative variables. The results are shown in Tables 4.1 and 4.2.

Method	K=1				
	K = 1	F_1	Z_1	Z_2	F_1^v
Plaid-Full	10	1 (0)	2.4 (1.42)	849.6 (0.51)	0.99(0.005)
Plaid-Restricted	10	1 (0)	2.6 (1.26)	849.5 (0.70)	0.99(0.004)
L_1-Penalty	10	1 (0)	0.1 (0.31)	815.2 (13.58)	0.89(0.03)
GMBC-GS	8	0.72 (0.24)	142.6 (8.05)	818.8 (2.74)	0.07(0.08)

Table 4.1: Results for the Scenario 1. The column “K = 1” is the number of times (out of 10) that 1 was identified as the number of clusters. F_1 is the F1 measure evaluated between the true clustering and the estimated one by the corresponding method. Z_1 is the number of variables excluded from the model out of the 150 informative variables. Z_2 is the number of excluded variables from the model out of the 850 noise variables. The numbers in the parentheses are the corresponding standard deviations.

Table 4.1 shows the results for the first scenario. As we can see, the three methods Plaid-Full, Plaid-Restricted and L_1-Penalty always selected two clusters (i.e., $K = 1$) as the number of clusters for the ten data sets. Also, they detected the true structure of the clustering since their F_1 measure is exactly 1. In contrast, the GMBC-GS method of Raftery and Dean selected $K = 1$ for eight data sets and its clustering results are less good that the others (the F_1 is smaller). Our methods (Plaid-Full and Restricted) also performed better than the others two in terms of variable selection (the F_1^v is larger) but tended to keep slightly fewer (about 1.7% excluded) informative variables than the L_1-Penalty method. Curiously, this did not affect their good results probably because of the abundance of informative variables. Also, our methods show very little variability in the results as compared to those of the other methods. Note that GMBC-GS selected only approximately eight variables among the 150 informatives variables with very high variability in its results.

Table 4.2 shows the results for the second scenario. In terms of quality of clustering, all four methods perform similarly. One can also see that it becomes more difficult to estimate the right number of clusters when the true number of clusters increases. However, our methods get the right informative variables most of the time. In this sense it performs much better than the other two methods. We stress that obtaining the correct variables is specially important in many applications such as those involving gene expression data.

4.6 Application to gene expression data

4.6.1 The Colon tumor data

The first data set is an Affymetrix oligonucleotide array from 62 samples collected from colon-cancer patients (Alon et al. [2]). It contains 40 tumor biopsies and 22 normal biopsies from healthy parts of the colons of the same patients. 2000 out of around 6500 human genes were selected based on the highest minimal intensity across the samples.

We applied the Plaid-Full and Plaid-Restricted models to analyse this data. The BIC criterion selected $K = 3$ (plus the zero-cluster) as the number of clusters in the two methods. Figure 4.1 shows the corresponding BIC curves. The zero-clusters from the Plaid-Full and Plaid-Restricted models contain four and six conditions, respectively. The Plaid-Full model selected 700 discriminating genes. The Plaid-Restricted model selected 721 discriminating genes. The 700 genes selected by Plaid-Full and the 721 genes selected by Plaid-Restricted have 677 genes in common. The clustering results are displayed in Table 4.3. Most of the 40 patients with tumor belong to Clusters 1 and 2. The normal biopsies are not really well distinguished from the tumor ones in the clusterings. We also applied the L_1-Penalty model to this data. It selected all the 2000 genes as discriminating. The clustering results are also shown in Table 4.3. It is clear that none of these three methods was fully capable of clearly distinguishing the normal from the tumor biopsies. However, the images in the Figure 4.3 show a clear distinction between the four clusters found by the plaid clustering algorithms. In these images, only the discriminating genes are plotted (x-axis). The biopsies are sorted according to the clusterings given by the methods. Consequently, and in view of our results and those of the L_1-penalty model, it is posible that these particular biopsies cannot be well separated on the solely basis of their gene expression data.

4.6.2 The SRBCT data

The second application comprises microarray gene expression data coming from the small round blue cell tumors (SRBCTs) of childhood (Khan et al. [12]). These are divided in four groups: Burkitt lymphoma (BL), Ewing sarcoma (EWS), neuroblastoma (NB), and rhabdomyosarcoma (RMS). The data consist of 6567 genes, and have been divided into 63 training samples and 20 test samples. The training samples include 8 samples of BL, 23 samples of EWS, 12 samples of NB, and 20 of RMS. The test samples contained 6, 5, 6 and 3 samples of EWS, RMS, NB and BL, respectively. Each sample consists of gene expression levels associated with $q = 2308$ genes. We applied the Plaid-Full and Plaid-Restricted models to the training data. Figure 4.2 shows the BIC curves for the two plaid models. BIC selected $K = 3$ clusters (plus the zero-cluster) for the Plaid-Restricted model, and $K = 1$ (plus the zero-cluster) for the Plaid-Full model. Plaid-Restricted selected 39 genes, whilst Plaid-Full selected only 28 genes. Figure 4.4 shows the clusters found by the two models. We also applied the L_1-Penalty of Pan and Shen. Their method selected 2183 genes. Their clustering results are also summarized in Table 4.4. All three methods have difficulties separating the EWS and RMS groups. The results from Plaid-Restricted model with $K = 4$ (plus the zero-cluster) are a bit better than all the three others. These also can be seen in Table 4.4.

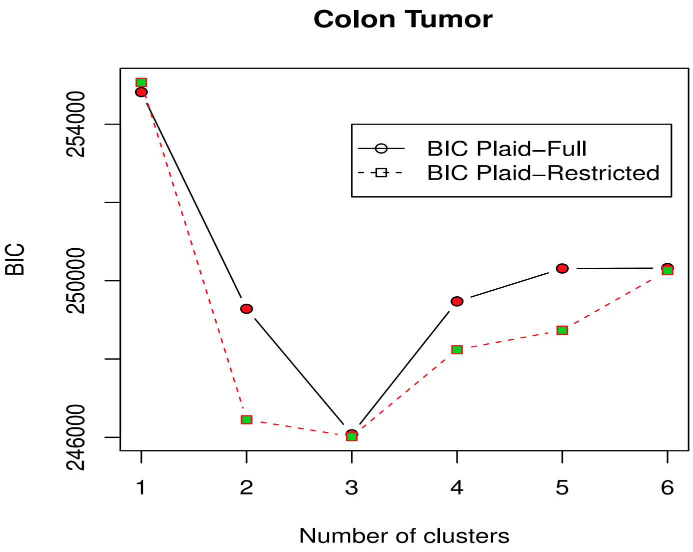


Figure 4.1: BIC for the Colon tumor data

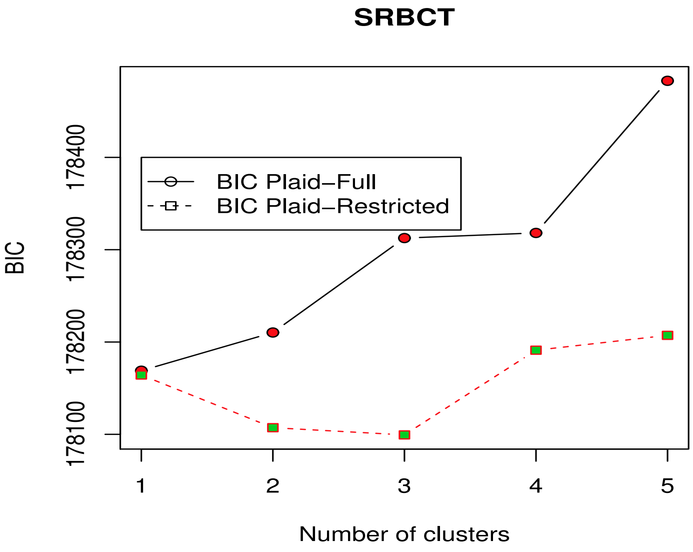
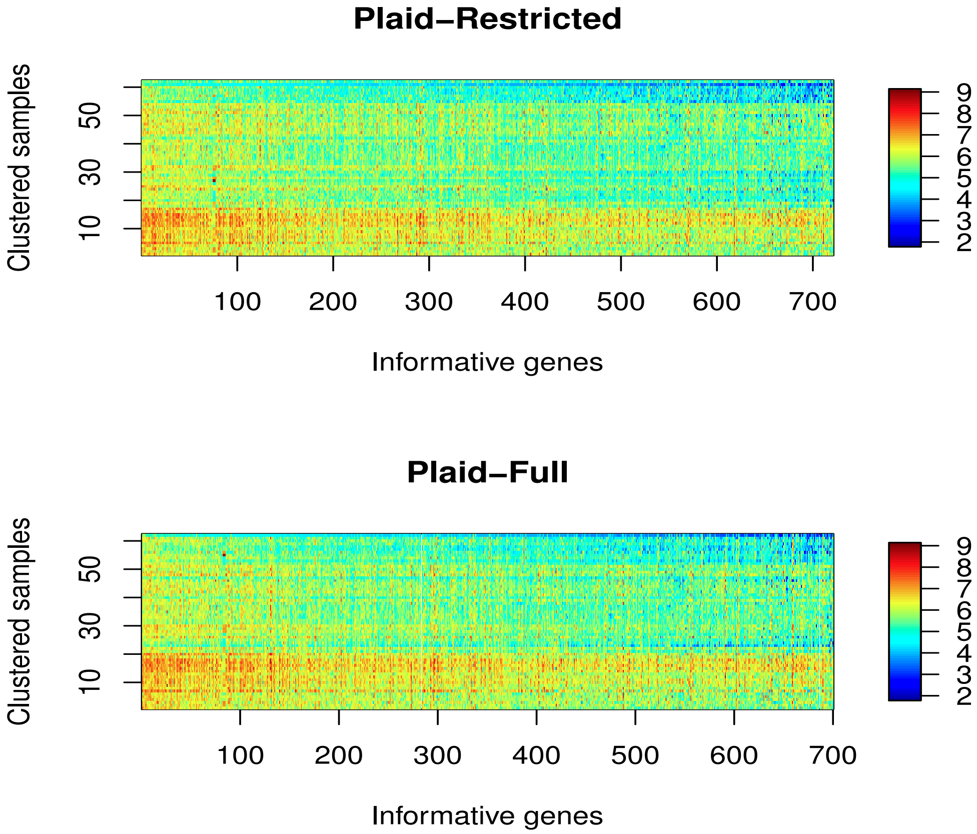


Figure 4.2: BIC for the SRBCT data


Plaid−Restricted

Plaid−Full

Figure 4.3: Clustering images for the Colon Tumor data

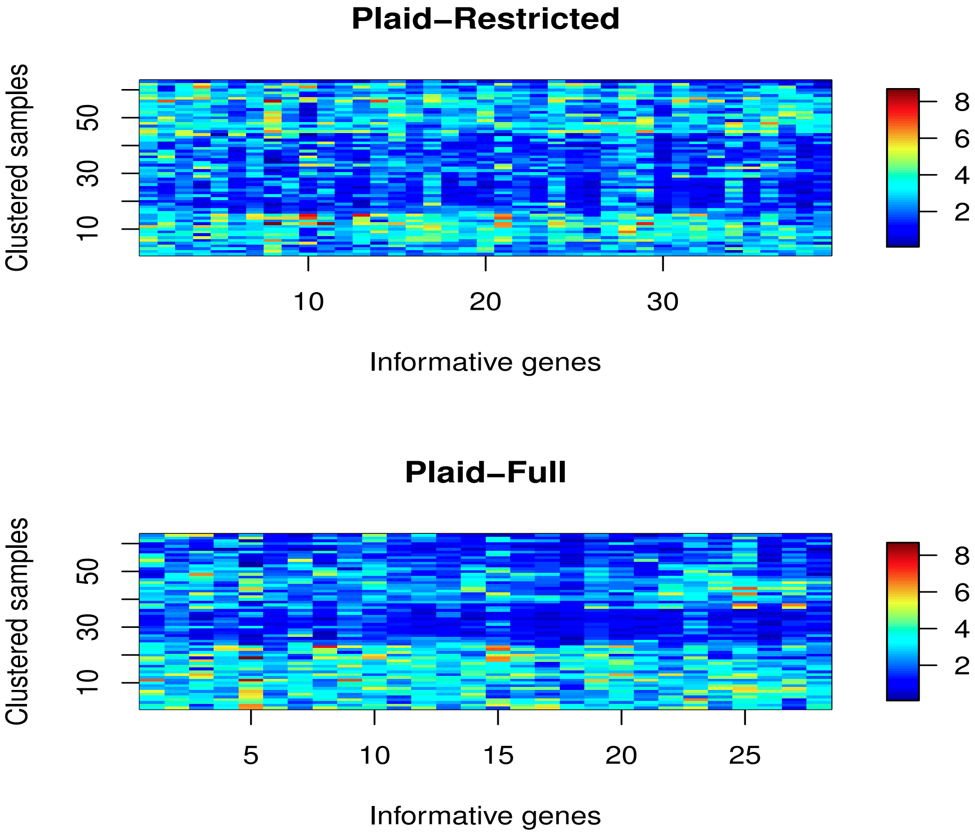


Figure 4.4: Clustering images for the SRBCT data

	K=1				
Method	K = 1	F_1	Z_1	Z_2	F_1^v
Plaid-Full	10	1 (0)	0 (0)	980 (0)	1 (0)
Plaid-Restricted	10	1 (0)	0 (0)	980 (0)	1 (0)
L_1-Penalty	5	0.85 (0.15)	0 (0)	978.8 (3.8)	0.97(0.07)
GMBC-GS	0	0.71 (0.02)	15.8 (1.54)	965.9 (7.01)	0.22(0.06)
	K=2				
Method	K = 2	F_1	Z_1	Z_2	F_1^v
Plaid-Full	8	0.84 (0.07)	0 (0)	980 (0)	1 (0)
Plaid-Restricted	6	0.77 (0.1)	0 (0)	980 (0)	1 (0)
L_1-Penalty	6	0.88 (0.08)	0 (0)	908.5 (125.1)	0.70(0.36)
GMBC-GS	2	0.81 (0.06)	17.3 (0.82)	961.6 (2.01)	0.13(0.04)
	K=3				
Method	K = 3	F_1	Z_1	Z_2	F_1^v
Plaid-Full	5	0.66 (0.06)	0 (0)	980 (0)	1 (0)
Plaid-Restricted	6	0.67 (0.06)	0 (0)	980 (0)	1 (0)
L_1-Penalty	3	0.69 (0.05)	0 (0)	930.0 (55.1)	0.59(0.3)
GMBC-GS	2	0.76 (0.1)	18.0 (0.66)	970.3 (9.20)	0.14(0.05)
	K=4				
Method	K = 4	F_1	Z_1	Z_2	F_1^v
Plaid-Full	4	0.80 (0.06)	0 (0)	980 (0)	1 (0)
Plaid-Restricted	3	0.85 (0.05)	0 (0)	980 (0)	1 (0)
L_1-Penalty	4	0.85 (0.07)	0 (0)	976.5 (3.6)	0.92(0.07)
GMBC-GS	0	0.76 (0.04)	18.0 (0.66)	977.6 (3.86)	0.17(0.06)

Table 4.2: Results for Scenario 2. The column “K= k” is the number of times (out of 10) that the right number of clusters was identified by each of the four methods: Plaid-Full, Plaid-Restricted, L_1-Penalty and GMBC-GS. There are $q = 1000$ variables. F_1 is the F1 measure evaluated between the true clustering and the estimated one by the corresponding method. Z_1 is the number of variables excluded from the model out of the first 20 informative variables. Z_2 is the number of excluded variables from the model out of the last 980 noise variables.

	Plaid-Restricted				Plaid-Full				L_1-Penalty					
	0	1	2	3	0	1	2	3	1	2	3	4	5	6
Normal	3	2	13	4	5	3	7	7	3	7	3	1	2	6
Tumor	1	11	24	4	1	11	24	4	7	9	8	6	1	9

Table 4.3: Clustering results for Colon tumor data

	Plaid-Restricted				Plaid-Restricted					Plaid-Full		L_1-Penalty					
	0	1	2	3	2	0	1	4	3	0	1	1	2	3	4	5	6
EWS	1	1	17	4	14	3	0	3	3	4	19	3	1	5	5	9	0
BL	3	0	0	5	0	0	8	0	0	4	4	4	4	0	0	0	0
NB	0	5	0	7	0	1	3	7	1	10	2	3	9	0	0	0	0
RMS	1	4	12	3	7	0	0	5	8	5	15	0	4	2	6	6	2

Table 4.4: Clustering results for SRBCT data

BIBLIOGRAPHIE

[1] Allan, J., J. Carbonell, G. Doddington, J. Yamron, and Y. Yang (1998). Topic detection and tracking pilot study : Final report. In *In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 194–218.

[2] Alon, U., N. Barkai, D. A. Notterman, K. Gishdagger, S. Ybarradagger, D. Mackdagger, and A. J. Levine (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the United States of America 96*(12), 6745–6750.

[3] Bhattacharya, A. K. (2005). Evaluation of headache. *Journal, Indian Academy of Clinical Medicine 6*(1), 17–22.

[4] Booth, J. G. and J. P. Hobert (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *Journal of the Royal Statistical Society. Series B (Statistical Methodology) 61*(1), pp. 265–285.

[5] Chekouo, T. and A. Murua (2012). The penalized biclustering model and related algorithms. Submitted for publication.

[6] Fu, Q. and A. Banerjee (2008). Multiplicative mixture models for overlapping clustering. In *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*, pp. 791 –796.

[7] Fu, Q. and A. Banerjee (2009). Bayesian overlapping subspace clustering. In *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, pp. 776–781.

[8] Heller, K. A. and Z. Ghahramani (2007). A nonparametric Bayesian approach to modeling overlapping clusters. *Journal of Machine Learning Research - Proceedings Track 2*, 187–194.

[9] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Comput. 14*(8), 1771–1800.

[10] Hoff, P. D. (2006). Model-based subspace clustering. *Bayesian Analysis 1*(2), 321–344.

[11] Hubert, L. and P. Arabie (1985). Comparing partitions. *Journal of Classification 2*, 193–218.

[12] Khan, J., J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat Med 7*(6), 673–679.

[13] Kim, S., M. G. Tadesse, and M. Vannucci (2006). Variable selection in clustering via Dirichlet process mixture models. *Biometrika 93*(4), 877–893.

[14] Lazzeroni, L. and A. Owen (2002). Plaid models for gene expression data. *Statistica Sinica 12*, 61–86.

[15] Levine, R. and G. Casella (2001). Implementations of the Monte Carlo EM algorithm. *Journal of Computational and Graphical Statistics 10*(10), 422–439.

[16] Madeira, S. C. and A. L. Oliveira (2004). Biclustering algorithms for biological data analysis : A survey. *IEEE/ACM Trans. Comput. Biol. Bioinformatics 1*(1), 24–45.

[17] McCulloch, C. E. (1997). Maximum likelihood algorithms for generalized linear mixed models. *Journal of the American Statistical Association 92*(437), 162–170.

[18] Pan, W. and X. Shen (2007). Penalized model-based clustering with application to variable selection. *J. Mach. Learn. Res. 8*, 1145–1164.

[19] Raftery, A. E. and N. Dean (2006). Variable selection for model-based clustering. *Journal of the American Statistical Association 101*, 168–178.

[20] Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association 66*, 846–850.

[21] Robert, C. and G. Casella (2004). *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer.

[22] Robert, C. P., T. Rydén, and D. Titterington (1999). Convergence controls for MCMC algorithms, with applications to hidden Markov chains. *Journal of Statistical Computation and Simulation 64*(4), 327–355.

[23] Schwarz, G. E. (1978). Estimating the dimension of a model. *Annals of Statistics 6*(2), 461–464.

[24] Tadesse, M. G., N. Sha, and M. Vannucci (2005). Bayesian variable selection in clustering high-dimensional data. *Journal of the American Statistical Association 100*, 602–617.

[25] Tanay, A., R. Sharan, and R. Shamir (2005). Biclustering algorithms : A survey. In *In Handbook of Computational Molecular Biology Edited by : Aluru S. Chapman and Hall/CRC Computer and Information Science Series*.

[26] Wang, S. and J. Zhu (2008). Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics 64*(2), 440–448.

[27] Wei, G. C. G. and M. A. Tanner (1990). A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association 85*(411), 699–704.

[28] Xie, B., W. Pan, and X. Shen (2008). Variable selection in penalized model-based clustering via regularization on grouped parameters. *Biometrics 64*(3), 921–930.

[29] Zhou, H. (2009). Manual for program of the algorithm of Pan, W. and Shen, X. (2007). Available online at `http://www.biostat.umn.edu/~weip/prog.html`.

CONCLUSION

Cette thèse résume les contributions sur l'utilisation des extensions du modèle de plaid pour effectuer un bi-regroupement et la sélection des variables dans une matrice d'expression génétique. Le bi-regroupement, aussi bien la sélection des variables constituent un défi en bio-informatique puisqu'il vise à trouver des régions locales d'intérêt dans les ensembles de données à haut débit.

Au chapitre 1, nous avons proposé une revue de la littérature sur certains algorithmes de bi-regroupement. Il ressort de cette revue que les méthodes de bi-regroupement pourraient être classées en fonction du type d'imbrication entre les bi-grappes. Les types d'imbrication populaire seraient des bi-regroupements où ses bi-grappes sont positionnées de façon arbitraire (possibilité de chevauchement des cellules de la matrice) et le type d'imbrication lignes-colonnes où soit les lignes, soit les colonnes peuvent s'imbriquer dans un bi-regroupement. De plus, les modèles de bi-grappes peuvent être multiplicatifs et additifs. De plus, le nombre de bi-grappes est soit fixé, soit déterminé par un critère d'arrêt ou par un critère de sélection de modèle. Tout au long de cette thèse, nous nous sommes intéressés au modèle additif avec la possibilité d'avoir les chevauchements entre les bi-grappes.

Au chapitre 2, nous avons introduit une extension du modèle de plaid ; le modèle de plaid pénalisé. Ce modèle incorpore un paramètre qui contrôle la quantité d'imbrication de cellules entre les bi-grappes. Si ce paramètre est nul, l'on obtient le modèle plaid original de Lazzeroni et Owen. Par contre, s'il est très large, l'on pourrait retrouver le modèle sous-jacent de Cheng et Church. A travers une étude de simulations, nous avons prouvé que les résultats issus des implémentations MCMC (Échantillonnage de Gibbs et de Metropolis-Hastings) des modèles sont meilleur que ceux de l'algorithme original de Cheng et Church et du modèle plaid. Nous avons défini un critère DIC de sélection de modèle qui semble approprié pour le problème de bi-regroupement. Le modèle de plaid pénalisé a été appliqué aux données de cycle cellulaire de levure de Eisen et al (1998). Les bi-grappes obtenues sont toutes différentes compte tenu de la diversité de leurs rôles biologiques obtenus à l'aide de l'ontologie des gènes (GO).

Le chapitre 3 propose un modèle de bi-regroupement qui incorpore la connaissance biologique des gènes et des conditions expérimentales. Cette connaissance aide à définir des distributions à priori sur les étiquettes d'appartenance qui tiennent compte de la structure de dépendance entre les gènes et entre les conditions expérimentales. L'introduction d'un champ de Markov dans le contexte de bi-regroupement est nouveau comparé au groupement ou aux modèles de régression. Pour estimer les paramètres, nous avons adopté une procédure basée sur une variante de l'algorithme de Wang-Landau qui nous aide à contourner l'intractabilité de la constante de normalisation des distributions a priori des étiquettes. Nos expériences sur des données simulées montrent que notre approche peut améliorer les autres algorithmes dans la littérature tels que celui de Cheng et Church, Lazzeroni et Owen. L'expérience sur les données réelles (cycle cellulaire de la levure) souligne d'autres caractéristiques intéressantes de notre modèle en terme d'enrichissement de gènes dans les bi-grappes.

Un modèle approprié pour la sélection des variables dans un contexte de groupement a été proposé au chapitre 5. Nous avons proposé deux méthodes pour retrouver la vraie structure des groupes. L'une qui tient compte de l'imbrication des gènes, et l'autre non. Ces méthodes groupent les "individus" et sélectionnent les variables informatives simultanément. Ce modèle inspiré de celui de plaid tient compte de l'imbrication des individus et chaque grappe obtenue est une bi-grappe à valeurs cohérente sur les lignes et sur les colonnes. Nous avons utilisé l'algorithme EM de Monte Carlo avec échantillonnage d'importance pour estimer les paramètres. Nos études expérimentales montrent que notre modèle donne de bons résultats en terme de sélection de variable et de regroupement.

La plupart des modèles proposés dans cette thèse supposent que le nombre de bi-grappes (ou grappes) K est fixé. Il est déterminé après par un critère de sélection de modèle. Nous pensons par la suite supposer K comme un paramètre du modèle. Dans le cas de groupement, plusieurs approches dans un contexte bayésien ont été proposées. L'exemple populaire est l'utilisation d'un processus de Dirichlet sur les probabilités a priori des étiquettes. La possibilité d'imbrication des bi-grappes rendrait la tâche moins facile pour définir un tel processus dans le cas du bi-regroupement. Une autre façon de considérer K comme un paramètre serait de supposer que K suit une distribution uniforme ou une distribution de poisson tronquée à une valeur arbitraire K_{max}.

Il faut aussi noter que nous avons supposé dans cette étude que nos données suivent des distributions normales alors qu'il est connu que les données d'expression génétiques peuvent avoir des queues très épaisses dans leur distribution. L'on pourrait donc penser à chercher les bi-grappes en supposant par exemple des distributions de Student. Ainsi, les paramètres estimés seront moins affectés par des valeurs aberrantes. L'on pourra qualifier cette approche de bi-regroupement bayésien robuste si nous travaillons dans un contexte bayésien.

Au chapitre 5 de cette thèse qui traite de la sélection de variables, les distributions de probabilités a priori sur les colonnes (gènes) sont supposées indépendantes. Il est possible d'améliorer cette hypothèse en considérant l'interaction qu'il pourrait y avoir entre les gènes à travers un graphe relationnel. Ce graphe pourrait être construit comme au chapitre 3 en utilisant l'annotation GO des gènes. L'on espère obtenir moins de gènes informatifs capable de discriminer la structure cachée de groupement des individus.