Les Cahiers du GERAD ISSN: 0711–2440

Forestogram: A visualization framework for hierarchical biclustering

M. Sajjad Ghaemi, V. Partovi Nia, B. Agard

G–2017–40

May 2017

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2017
– Bibliothèque et Archives Canada, 2017

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2017
– Library and Archives Canada, 2017

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Forestogram: A visualization framework for hierarchical biclustering

Mohammad Sajjad Ghaemi [a]
Vahid Partovi Nia [a]
Bruno Agard [b]

[a] *GERAD & Department of Mathematics and Industrial Engineering, Polytechnique Montréal (Québec) Canada, H3C 3A7*

[b] *CIRRELT & Department of Mathematics and Industrial Engineering, Polytechnique Montréal (Québec) Canada, H3C 3A7*

sajjad.ghaemi@gerad.ca
vahid.partovinia@polymtl.ca
bruno.agard@polymtl.ca

May 2017
Les Cahiers du GERAD
G–2017–40

Abstract: Many biological datasets such as microarrays, metabolomics, and proteomics involve observations (or subjects) in rows, and attributes (or genes, metabolites, proteins) in columns. Often simultaneous grouping of rows and columns, i.e. biclustering, is desired. Each bicluster consists of a group of observations highly correlated in a group of attributes. Despite great efforts on developing biclustering algorithms, a proper visualization seems to be lacking in the literature. A visualization tool helps practitioners to understand how biclusters evolve. Here we provide this tool using *forestogram*. Forestogram combines rows or columns iteratively towards constructing a forest over a collection of dendrograms with a common root. We develop a simple strategy for extracting natural biclusters by cutting the forest using a simple information criterion. The effectiveness of our technique is tested on simulated data, and on real data.

Keywords: Biclustering, dendrogram, hierarchical clustering, linkage

1 Introduction

Clustering, or data grouping is a challenging problem. Clustering in NP-hard, i.e. the number of different ways to group data grows exponentially with the sample size. Clustering algorithms can be categorized into two categories: hierarchical, and partitional. Hierarchical methods find the nested clusters recursively, while partitional approaches provide only a single grouping. Partitional algorithms require the number of clusters to be set a priori. Hierarchical approaches, on the contrary, starts from each item as a singleton and builds clusters until all data fall in a single cluster. A clustering algorithm that assumes a statistical model for clustering data, called model-based clustering (McLachlan et al., 2004). Practitioners often prefer hierarchical clustering, because of the visual guide produced through dendrogram. Clustering *linkage*, also known as *dissimilarity*, plays a central role in building the dendrogram.

Biclustering, also known as *coclustering*, and *joint clustering* is a general class of methods that aims to partition a data matrix. Unlike clustering that groups observations, *or* attributes, biclustering searches a grouping on observations *and* attributes at the same time. The advent of high-dimensional data calls for devising new algorithms to exploit the clusters more effectively.

Biclustering attracted researchers from various fields because of its modern applications (Zhang, 2010). Biclustering is used to cluster documents and words in text mining (Orzechowski and Boryczko, 2016), genes and experimental conditions in bioinformatics (Eren et al., 2013), tokens and contexts in natural language processing (Tu and Honavar, 2008), users and movies in recommender systems (Xu et al., 2012), etc. The first joint clustering method appeared in statistics literature in Hartigan (1972), but implemented after few decades (Cheng and Church, 2000). Like clustering, biclustering involves two clutures: i) statistical approach that assumes a probabilitic distribution (Sheng et al., 2003; Gan et al., 2008; van Uitert et al., 2008; Lazzeroni and Owen, 2002; Sheng et al., 2003; Gan et al., 2008); and (ii) the algorithmic approach that minimizes a dissimilarity (Hartigan, 1972; Hochreiter et al., 2010; Martella et al., 2008), for a comprehensive review see Busygin et al. (2008).

Most of the biclustering techniques are partitional and the number of blocks is the input of the algorithm. However, in a number of applications hierarchical approach is very common, because of two main advantages: i) having little assumption on data and number of groups ii) providing a visualization diagram through the dendrogram.

A simple hierarchical biclustering method is known as *heatmap*, and produces two independent dendrograms, one on rows and another on columns. This representation is loose due to the independent construction of row and column groupings. However, an interesting visualization tool for biclustering is proposed using convex reformulation of the biclustering problem in Chen et al. (2015), but it lacks the conventional dendrogram representation that practitioners are used to see. An agglomerative method using a complex Bayesian model is suggested in Fowler and Heard (2012). Smith et al. (2008) argues that complex models may lead to junk clusters if agglomerative method is used.

We propose i) a natural extension of biclustering method using common linkages, ii) produce forestogram, a conventional graphical tool that extends dendrogram, iii) benefit a simple hierarchical model to develop a criterion as a reference for cutting forestogram. It turns out that our criterion is the natural biclustering extension of the well-known information criteria, such as the AIC (Akaike, 1973) and BIC (Schwarz, 1978).

The paper is structured as follows. Section 2 describes our proposed methodology and forestogram. Section 3 studies the computational complexity of the forestogram construction. Section 4 compares forestogram with some common biclustering methods, and Section 5 shows the application of forestogram on the yeast galactose data.

2 Hierarchical biclustering

Hierarchical biclustering is a natural extension of hierarchical clustering for grid matrices. Section 2.1 generalizes common linkages for biclustering. Section 2.2 explains how to build the forestogram using the generalized

linkage. Section 2.3 develops an information criterion to provide a statistically meaningful suggestion for the forestogram cut, and Section 2.4 explores the relationship between biclustering and forestogram.

2.1 Bilinkage

Hierarchical biclustering algorithms require a dissimilarity measure to merge block of clusters and build nested groups. The dissimilarity measure is a positive semi-definite symmetric mapping of pair of groups, onto real numbers. Dissimilarity, however, may not satisfy the triangle inequality unlike the distance. The common linkages include single linkage or nearest neighbors, complete linkage or farthest neighbors, average linkage, centroid linkage, median linkage, and Ward's linkage, see (Sørensen, 1948; Sokal, 1958; Eisen et al., 1998; Murtagh and Legendre, 2014) for more details.

The linkage is defined using a distance, often the Euclidean distance, but may be defined on metrics such as Manhattan, Chebyshev, or Mahalanobis distance.

We suppose grid biclusters, and use I to index row clusters, and J to index column clusters. The first step in build the hierarchical biclustering is to generalize the linkage to a *bilinkage* to measure the dissimilarity between matrix blocks. Any marge, however, cannot be visualized by a nested tree. Therefore, a convenient bilinkage must be defined over a pair of biclusters, using row and column directions. Suppose $\mathcal{C}_1$ and $\mathcal{C}_2$ are disjoint rectangular biclusters

$$\text{Bilinkage}(\mathcal{C}_1, \mathcal{C}_2) = \min_{I \neq I', J \neq J'} \left\{ D(\mathcal{C}_{I1}^{\text{row}}, \mathcal{C}_{I'2}^{\text{row}}), D(\mathcal{C}_{1J}^{\text{col}}, \mathcal{C}_{2J'}^{\text{col}}) \right\} \tag{1}$$

where $\mathcal{C}_{I1}^{\text{row}}$ is the Ith row-cluster of bicluster $\mathcal{C}_1$, $\mathcal{C}_{1J}^{\text{col}}$ is the Jth column-cluster of bicluster $\mathcal{C}_1$, and D is a clustering linkage. Table 1 gives the definition of the commonly used linkages. The minimum in (1) is taken once over a pair of row-clusters, and once over a pair of column-clusters. This minimum defines the direction of the merge, a row marge, or a column merge. We suppose that data are standardized, so that row and column blocks are comparable.

Table 1: A list of common linkages for hierarchical clustering, defined using the Euclidean distance, where $\bar{\mathbf{y}}$ denotes the mean $\tilde{\mathbf{y}}$ denotes the median.

Linkage	Definition
Single	$\min_{\mathbf{y}_i \in \mathcal{C}_1, \mathbf{y}_j \in \mathcal{C}_2} \|\mathbf{y}_i - \mathbf{y}_j\|$
Complete	$\max_{\mathbf{y}_i \in \mathcal{C}_1, \mathbf{y}_j \in \mathcal{C}_2} \|\mathbf{y}_i - \mathbf{y}_j\|$
Average	$\frac{1}{\lvert\mathcal{C}_1\rvert\lvert\mathcal{C}_2\rvert} \sum_{\mathbf{y}_i \in \mathcal{C}_1} \sum_{\mathbf{y}_j \in \mathcal{C}_2} \|\mathbf{y}_i - \mathbf{y}_j\|$
Ward	$\frac{\lvert\mathcal{C}_1\rvert\lvert\mathcal{C}_2\rvert}{\lvert\mathcal{C}_1\rvert+\lvert\mathcal{C}_2\rvert} \|\bar{\mathbf{y}}(\mathcal{C}_1) - \bar{\mathbf{y}}(\mathcal{C}_2)\|$
Centroid	$\|\bar{\mathbf{y}}(\mathcal{C}_1) - \bar{\mathbf{y}}(\mathcal{C}_2)\|$
Median	$\|\tilde{\mathbf{y}}(\mathcal{C}_1) - \tilde{\mathbf{y}}(\mathcal{C}_2)\|$

2.2 Forestogram

Forestogram is a collection of binary trees that consists of multiple hierarchical dendrograms. Construction of the forestogram is bottom-up, such that a pair of row-wise or column-wise clusters is combined together at each level by starting from singleton clusters.

Forestogram merges a block of rows or a block of columns in each step, depending on the direction that minimizes the bilinkage (1). After each merge, the dissimilarity measure is recomputed to identify the next merge direction. This approach, gives a new block of data on the forestogram. A grouping is extracted if the forestogram is cut at a certain height, see Figure 1.

Forestogram has a number of interesting advantages to interpret the block-clusters of data as follows. Each cluster reflects the order of rows and columns that shares a similar pattern. The merge path gives a

visual guide on the evolution of the biclusters. Forestogram gives a visual guide on the interaction between row and column groupings. A row dendrogram and a column dendrgoram can be extracted by projecting the forest over rows and columns, see Figure 2. The last property is attractive for practitioners who are used to heatmap graphics.

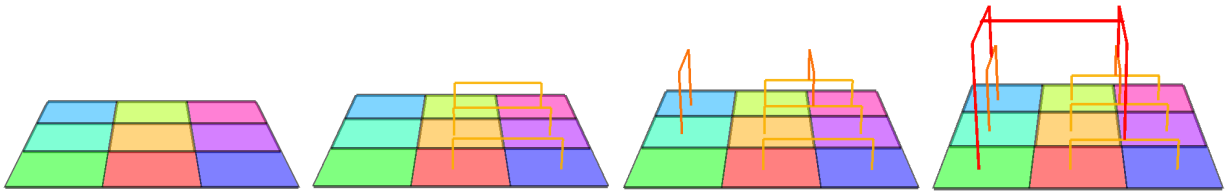

Figure 1: Forestogram building steps on a hypothetical 3×3 matrix. Left to right: the data matrix, merging a pair of columns, merging a pair of rows, and the completed forestogram.

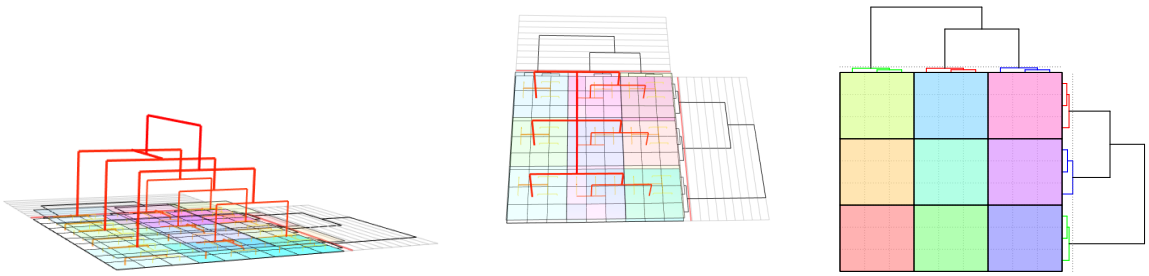

Figure 2: A hypothetical 9×9 matrix clustered into three row blocks and 3 column blocks after cutting the forestogram by a plane. Forestogram projection on rows and on columns provides two marginal dendrograms. Forestogram side view (left panel), above view (middle panel), projection of the forestogram on rows and columns resembeling a heamap graphics (right panel); the dotted horizontal and vertical lines is the projection of the cutting plane.

2.3 Number of biclusters

Estimating the number of biclusters through cutting the forestogram at a certain height, is equivalent to finding a tangible gap on the height of the forestogram for a natural grouping. We propose to cut the forest when biclusters have the tendency of concentration about a center.

Assume a grid bicluster $\mathcal{C} = \mathcal{C}^{\text{row}} \times \mathcal{C}^{\text{col}}$ and therefore $\mathbf{Y}_{n\times p} \mid \mathcal{C}$ is clustered into several row and column clusters. Obviously, the total number of bicluster is $|\mathcal{C}| = |\mathcal{C}^{\text{row}}||\mathcal{C}^{\text{col}}|$. Index biclusters using $\mathbf{Y}_{IJ} = [y_{IiJj}]$, where the I denotes the row cluster and J denotes the column cluster, $I =, 1 \ldots, |\mathcal{C}^{\text{row}}|$, $J = 1, \ldots, |\mathcal{C}^{\text{col}}|$, and i and j index the rows and columns of $\mathbf{Y}_{IJ}$, $i = 1, \ldots, n_I$, and $j = 1, \ldots, p_J$, respectively. Note that n_I is the number of rows in cluster I, and p_J is the number of columns if cluster J, of course

$$n = \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} n_I, \quad p = \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} p_J.$$

In hierarchical clustering using a linkage, closer data have the tendency to merge. So, it is reasonable to cut the agglomerative tree using some concentration measure. Assume the average of data is subtracted so the data are centered around zero . The following statistical model looks meaningful to express the concentration of bicluster IJ around a center

$$\begin{aligned} y_{IiJj} \mid \theta_{IJ} &\sim \mathcal{N}(\theta_{IJ}, \sigma^2), \\ \theta_{IJ} &\sim \mathcal{N}(0, \phi\sigma^2), \end{aligned} \tag{2}$$

where σ^2 is the common within variance, and ϕ is the between-variance to within-variance ratio. We propose to cut the forestogram where this Gaussian model fits appropriately. It turns out that (2) yields a simple and interesting cutting strategy.

Define the within cluster variance

$$s_{IJ}^2 = \frac{1}{n_I p_J} \sum_{i=1}^{n_I} \sum_{j=1}^{p_J} (y_{IiJj} - \bar{y}_{IJ})^2.$$

and the pooled variance

$$s^2 = \frac{1}{np} \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} n_I p_J s_{IJ}^2.$$

The optimal number of clusters using (2) is found by minimizing the forest information criterion (FORIC). FORIC is a sort of penalized variance

$$np(1 + \log 2\pi s^2) + \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} \log(n_I p_J \phi + 1). \tag{3}$$

We suggest to fix $\phi = 1$ and estimate the pooled variance s^2 in each level of the forestogram tree. The following theorem shows how FORIC is derived.

Theorem 1 *if biclusters are generated from* (2)

$$-2 \log f(\mathbf{Y}) = \frac{1}{\sigma^2} \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} n_I p_J s_{IJ}^2 + \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} \log(n_I p_J \phi + 1). \tag{4}$$

Note that (4) is exact, but AIC and BIC are asymptotic approximations. Using the asymptotic argument similar to BIC, one may derive an extended version of FORIC,

$$-2 \log f(\mathbf{Y}) \approx -2 \log f(\mathbf{Y} \mid \hat{\boldsymbol{\theta}}) + \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} \log(n_I p_J \phi + 1). \tag{5}$$

FORIC is the adaptation of the AIC (Akaike, 1973) and BIC (Schwarz, 1978) for biclustering. Suppose biclusters are balanced and each bicluster contains $n_I = n_0$ rows, $p_J = p_0$ columns. The extended version (5) coincides with the AIC if $\phi = \frac{e^2-1}{n_0 p_0}$ and coincides with the BIC if $\phi = \frac{n_0 p_0 - 1}{n_0 p_0}$.

2.4 Separable biclusters

Hierarchical clustering algorithms are prone to converge to a sub-optimal grouping, due to their intrinsic greedy behavior. Here we show that a separable bicluster will appear always on the forestogram tree. This property holds for all linkages. Before defining a separable bicluster, we need to define the *diameter* and the *margin* concepts.

Take the submatrix $\check{\mathbf{Y}} \subset \mathbf{Y}_{n \times p}$. Denote the row extension and column extension of $\check{\mathbf{Y}}$ using $\check{\mathbf{Y}}^{\text{row}}$ and $\check{\mathbf{Y}}^{\text{col}}$ respectively, such that $\check{\mathbf{Y}}^{\text{row}} \cap \check{\mathbf{Y}}^{\text{col}} = \check{\mathbf{Y}}$, see Figure 3. Let $\mathbf{y}_i$ be the ith row of $\mathbf{Y}$ and $\mathbf{y}_j$ be the jth column of $\mathbf{Y}$. Likewise, let $\check{\mathbf{y}}_i^{\text{row}}$ is the ith row of $\check{\mathbf{Y}}^{\text{row}}$ and $\check{\mathbf{y}}_j^{\text{col}}$ is the jth column of $\check{\mathbf{Y}}^{\text{col}}$. The row margin of $\check{\mathbf{Y}}$ measures the pessimistic row-wise distance of $\check{\mathbf{Y}}^{\text{row}}$ from $\mathbf{Y}$, similarly the column margin measures the column-wise distance of $\check{\mathbf{Y}}^{\text{col}}$ from $\mathbf{Y}$

$$\mathfrak{M}^{\text{row}} = \min_{i \neq i'} \|\check{\mathbf{y}}_i^{\text{row}} - \mathbf{y}_{i'}\|^2,$$
$$\mathfrak{M}^{\text{col}} = \min_{j \neq j'} \|\check{\mathbf{y}}_j^{\text{col}} - \mathbf{y}_{j'}\|^2.$$

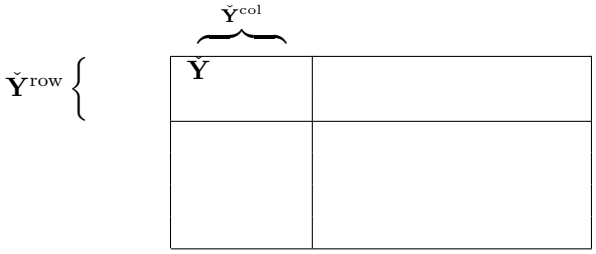

Figure 3: Visual illustration of submatrix $\check{\mathbf{Y}} \subset \mathbf{Y}$, extended on rows $\check{\mathbf{Y}}^{\text{row}}$, and on columns $\check{\mathbf{Y}}^{\text{col}}$.

Definition 1 *: margin of bicluster $\check{\mathbf{Y}}$ is the minimum of row and column margins*

$$\mathfrak{M}(\check{\mathbf{Y}}) = \min\left\{\mathfrak{M}^{\text{row}}, \mathfrak{M}^{\text{col}}\right\}.$$

Define the diameter of $\check{\mathbf{Y}}$ using row diameter and column diameter

$$\mathfrak{D}^{\text{row}} = \max_{i \neq i'} \|\check{\mathbf{y}}_i^{\text{row}} - \check{\mathbf{y}}_{i'}^{\text{row}}\|^2,$$
$$\mathfrak{D}^{\text{col}} = \max_{j \neq j'} \|\check{\mathbf{y}}_j^{\text{col}} - \check{\mathbf{y}}_{j'}^{\text{col}}\|^2,$$

Definition 2 *: diameter of bicluster $\check{\mathbf{Y}}$ is the maximum of row and column diameters*

$$\mathfrak{D} = \max\left\{\mathfrak{D}^{\text{row}}, \mathfrak{D}^{\text{col}}\right\}.$$

Sparability of a bicluster is defined by putting a condition on its margin and diameter.

Definition 3 *: bicluster $\check{\mathbf{Y}}$ is separable if $\mathfrak{M} > \mathfrak{D}$.*

In the following theorem we study the relationship between separability and forestogram.

Theorem 2 *separable submatrix $\check{\mathbf{Y}}$ always appear on the forestogram, regardless of the chosen linkage.*

See Appendix for the proof. Theorem 2 states that separable biclusters are kept intact during the hierarchical agglomeration. Such biclusters are recovered by cutting the forestogram at a specific level.

3 Computational complexity

A brute-force implementation of forestogram is of time complexity order $O(n^3 + p^3)$. This price is expensive for moderate matrices, and restricts the algorithm applicability on *omics* data. We provide computational tricks to improve the complexity of the algorithm.

Hierarchical clustering algorithms use a dissimilarity matrix to store the result of computation in an $n \times n$ matrix where n is the number of rows. The algorithm takes advantage of avoiding process of the pairwise dissimilarities repeatedly, by augmenting the stored data. One may prefer to compute the dissimilarities *on fly* to avoid storing the dissimilarity matrix. However, on-fly computation save the storage, with the price of increasing the computation. In the following we adapt the Lance-Williams technique (Lance and Williams, 1966) to hierarchical biclustering to accelerate the computations.

3.1 Lance-Williams speed-up

For each merge at each level of hierarchical clustering, a dissimilarity matrix for each pair of clusters is required. After each merge, the dissimilarity for newly merged clusters must be updated. Lance and Williams (1966) developed a concise formula to use the previous distance information, to update the dissimilarity matrix.

Suppose the merging cluster is denoted by $\mathcal{C}_1 \cup \mathcal{C}_2$, and $\mathcal{C}$ denotes another disjoint cluster in the same level of hierarchy

$$D(\mathcal{C}_1 \cup \mathcal{C}_2, \mathcal{C}) = \delta_1 D(\mathcal{C}_1, \mathcal{C}) + \delta_2 D(\mathcal{C}_2, \mathcal{C}) + \delta_3 D(\mathcal{C}_1, \mathcal{C}_2) + \delta_4 |D(\mathcal{C}_1, \mathcal{C}) - D(\mathcal{C}_2, \mathcal{C})|.$$

Table 2 gives more details about the coefficients $\delta_i, i = 1, \ldots, 4$.

Table 2: Lance-Williams coefficient merge updates for different linkages, if the Euclidean distance defines the linkage.

Linkage	δ_1	δ_2	δ_3	δ_4
Single	$\frac{1}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$
Complete	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$
Average	$\frac{1}{2}$	$\frac{1}{2}$	0	0
Centroid	$\frac{\|\|\mathcal{C}_1\|\|}{\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_2\|\|}$	$\frac{\|\|\mathcal{C}_2\|\|}{\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_2\|\|}$	$-\frac{\|\|\mathcal{C}_1\|\|\,\|\|\mathcal{C}_2\|\|}{(\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_2\|\|)^2}$	0
Ward	$\frac{\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_1\cup\mathcal{C}_2\|\|}{\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_2\|\|+\|\|\mathcal{C}_1\cup\mathcal{C}_2\|\|}$	$\frac{\|\|\mathcal{C}_2\|\|+\|\|\mathcal{C}_1\cup\mathcal{C}_2\|\|}{\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_2\|\|+\|\|\mathcal{C}_1\cup\mathcal{C}_2\|\|}$	$-\frac{\|\|\mathcal{C}_1\cup\mathcal{C}_2\|\|}{\|\|\mathcal{C}_1\|\|+\|\|\mathcal{C}_2\|\|+\|\|\mathcal{C}_1\cup\mathcal{C}_2\|\|}$	0
Median	$\frac{1}{2}$	$\frac{1}{2}$	$-\frac{1}{4}$	0

3.2 Time complexity

The implementation of hierarchical biclustering requires identifying the closest two clusters. The search algorithm looks up n times on the row dissimilarity matrix and p times on the column dissimilarity matrix. However, the dissimilarity matrix is shrunken after each merge, thus the overall computational complexity is $\sum_{i=1}^{n} i^2 + \sum_{j=1}^{p} j^2$ which is of order $\mathcal{O}(n^3 + p^3)$. A proper implementation of the Lance-Williams technique speeds up the algorithm to $O(n^2 + p^2)$.

3.3 Space complexity

Memory required to run the algorithm is important. If the data matrix $n \times p$ fits in the computer memory, the algorithm must reserve extra space for computation and storing the dissimilarity matrices. Hierarchical biclustering uses two dissimilarity matrices, and stores all pairwise dissimilarities for rows and columns. Therefore, early steps of the algorithm, all pairwise distances are computed and initiated in two different matrices, an $n \times n$ matrix for row dissimilarity and a $p \times p$ matrix for column dissimilarity. Using the Lance-William property, only a row group and a column group will be altered at each iteration of the algorithm. This implies $\mathcal{O}(n^2 + p^2)$ for the space.

In the following, we investigate our efficient implementation on a synthetic matrix of data by fixing the number of columns to 10, and varying the number of rows. In a similar setting rows are fixed to 10 and the number of columns is varied. Figure 4 confirms the quadratic complexity in term of rows and columns.

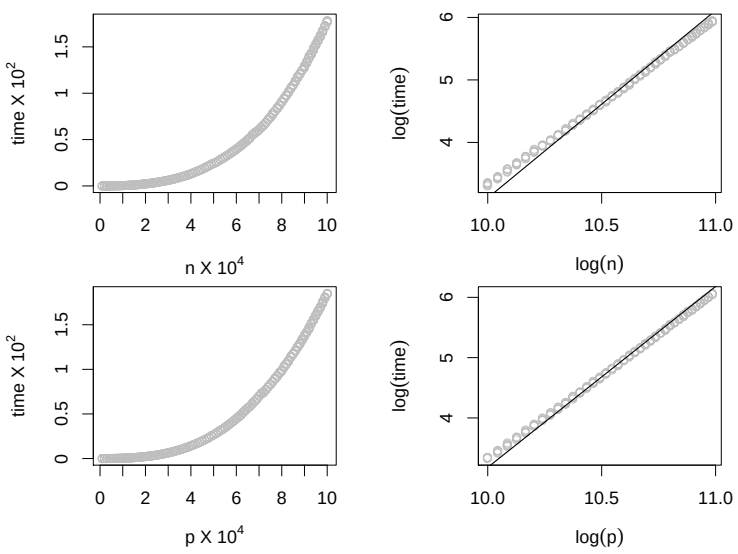

Figure 4: Time required to build the forestogram as the number of rows n increase (top panels), and as the number of columns p increase (bottom panels). The top right panel confirms that the algorithm is quadratic in n, the bottom right panel confirms that the algorithm is quadratic in p; the solid line is $y = \beta_0 + 2x$.

4 Simulation

In order to compare hierarchical method with other common biclustering techniques we generate a square matrix 30×30 a rectangular 150×30 matrix, divided in three clusters on rows and 3 clusters on columns. Both simulations include 10 columns in their column clusters.

In the square setup, each row cluster includes 10 rows see Figure 5, but in the rectangular simulation each row cluster simulation includes 50 rows. Each of the three biclusters is generated from uniform distribution of range 1 and varying mean $(-\Delta, 0, \Delta)$. The parameter $\Delta \in \{0.5, 1.0\}$ reflects the separability of biclusters. The larger the Δ is, the more separable biclusters are.

$\overbrace{\quad}^{10}$	$\overbrace{\quad}^{10}$	$\overbrace{\quad}^{10}$	
$-\Delta$	0	Δ	$\}10$
Δ	$-\Delta$	0	$\}10$
0	Δ	$-\Delta$	$\}10$

Figure 5: Symmetric simulation data consist of a matrix of size 30×30 with 9 biclusters. Each bicluster contains 100 data from uniform distribution with 10 rows in row cluster and 10 columns in column clusters. The parameter Δ controls the separability of biclusters.

The joint clustering of Lazzeroni and Owen (2002) and Cheng and Church (2000) are often used as standard biclustering methods. We found Cheng and Church (2000) performed poorly, so we report the *plaid* model of Lazzeroni and Owen (2002) only. The codes for Lazzeroni and Owen (2002) and Cheng and Church (2000) are available in the R package `biclust` R package (Kaiser et al., 2013). Our results are based on the the implementation of Lazzeroni and Owen (2002) developed by Turner et al. (2005).

There are few methods that combine biclustering with a visual guide. Practitioners often use the *heatmap* to produce a visualization of joint clusters. The `heatmap` produces a visualzation using independent row and column dendrograms. Convex biclustering (Chen et al., 2015) implmented in the R package `cvxclustr` is

a new technique with a visualization similar to dendrogram, produced by shrinking the mean of biclusters towards a common mean. Our method will be released as an R package in the near future.

We created three version of forestogram by varying the linkages. All linkages are defined using the Euclidean distance. The linkages include single, average, and Ward. Other linakges behavior was similar to the average linkage, and therefore not reported. A fully automatic version of forestogram is produced by cutting the forestogram by minimizing FORIC. The number of biclusters for all methods is set to 9. Note that, even after fixing the number of clusters the grouping may be different. Default parameters in the R package is used for the competing methods.

Table 3 summarizes the performance of all techniques using the adjusted Rand index of Hubert and Arabie (1985) implemented in the R package `mclust` (Fraley and Raftery, 1999). The adjusted Rand index is bounded from below by 0, and from above by 1. It gets the upper bound if the estimated biclustering matches the true clustering. We generated 100 replications of randomly generated data sets, and run different biclustering techniques. The average of the adjusted Rand index is reported. The maximum standard error is 0.1, so all reported digits are significant.

Table 3 confirms if separation parameter Δ increases the performance of all methods improve. Changing the matrix from square to rectangle increases the number of rows from 10 to 50. This change in data size, improves the clustering performance over column clusters, for all methods except for convex, and for heatmap single linkage.

It turns out the single linkage in heatmap implementation is an inefficient method, but the performance improves significantly after being implemented as a bilinkage. The automatic cut using FORIC on forestogram is the best for forests built using the Ward bilinkage. Plaid model appears to be the least favorable technique.

Table 3: The performance of different biclustering techniques using the average adjusted Rand index $\times 100$. The larger the adjusted Rand index is, the better the performance will be.

	Dimension	30×30				150×30			
	Separation	$\Delta = 0.5$		$\Delta = 1$		$\Delta = 0.5$		$\Delta = 1$	
	Side	row	col	row	col	row	col	row	col
Forestogram	Auto Single	55	55	55	55	56	100	56	100
	Auto Average	55	55	55	55	56	100	56	100
	Auto Ward	55	55	55	55	100	100	100	100
	Single	80	55	100	100	94	100	100	100
	Average	100	99	100	100	100	100	100	100
	Ward	100	99	100	100	100	100	100	100
Heatmap	Single	53	53	100	100	0	100	100	100
	Average	100	99	100	100	99	100	100	100
	Ward	100	99	100	100	100	100	100	100
Plaid	Bicluster	0	0	43	99	0	60	77	94
Convex	Bicluster	54	0	100	100	0	100	100	100

5 Application

The yeast galactose gene expression data (Ideker et al., 2001) investigates the influence of the **gal** gene family that allows cells to consume galactose, as a source of carbon. A perturbation is made in two different ways, related to a specific pathway component: i) eliminating one of the **gal** genes or ii) a wild-type for each subject regardless of galactose existence. We consider a sub-matrix of this data, widely analyzed by other researchers. The analysis of the entire data set 3935×20 is feasible thank to computational acceleration of the algorithm. For a similar analyses see Yeung et al. (2003); Yeung and Ruzzo (2001); Fowler and Heard (2012).

Each value in this data matrix is an average of four replicates. We cluster $\log_{10}$ of data with no preprocessing. The data are available in the supplementary material of Ideker et al. (2001). Forestogram helps to recognize similar group of genes with the same reaction to genetic perturbation. Figure 6 (bottom panel) is

the two-dimensional projection the forestogram of Figure 6 (top panel). Presence of gene **gal** perturbation is indicated by + sign.

The **gal4** is the only gene that stays in the same cluster regardless of whether galactose present or not after perturbation. This means the presence or absence of galactose has no effect on **gal4**. A similar result is reported in Fowler and Heard (2012) but with a Bayesian biclustering model.

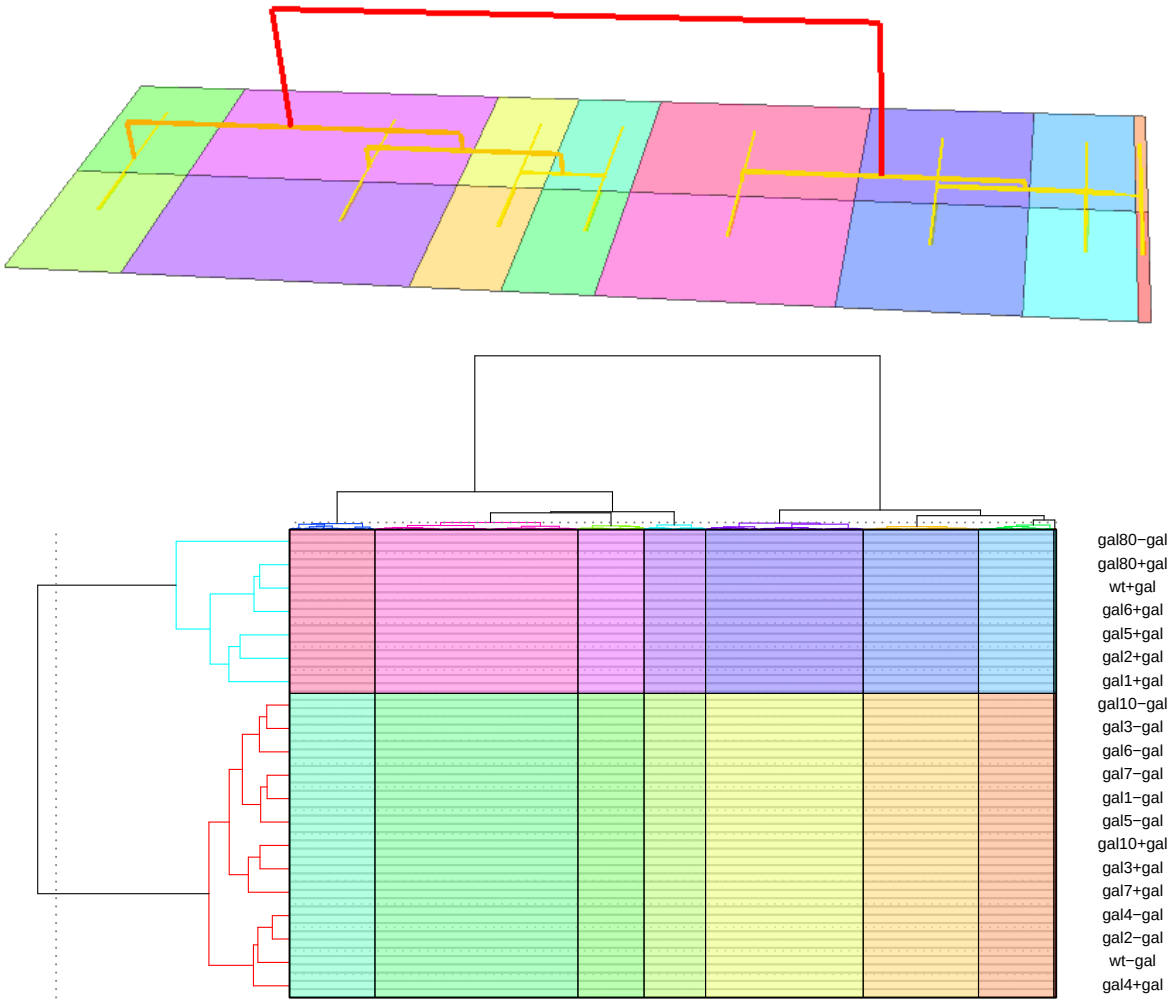

Figure 6: Top panel: forestogram produced using Ward bilinkage with automatic cut using FORIC. Buttom panel: two-dimensional projection of forestogram on rows and columns.

6 Proof of Theorem 1

Suppose the biclustering $\mathcal{C}$ is given. Following the analysis of variance notation, the Gaussian model (2) can be re-written in terms of a linear model, by putting the data matrix $\mathbf{Y}_{n\times p}$ in a long vector $\mathbf{y}_{np\times 1}$. The binary design matrix $\mathbf{X}_{np\times|\mathcal{C}|}$ consist of bicluster membership indicators, and $\boldsymbol{\theta}_{|\mathcal{C}|\times 1} = [\theta_{IJ}]$

$$\mathbf{y} \mid \boldsymbol{\theta} \sim \mathcal{MN}(\mathbf{X}\boldsymbol{\theta}, \boldsymbol{\Sigma}),$$

$$\boldsymbol{\theta} \sim \mathcal{MN}(\boldsymbol{\tau}, \boldsymbol{\Omega}),$$

where $\mathcal{MN}$ denotes the multivariate normal distribution. The conditional density is

$$f(\mathbf{y} \mid \boldsymbol{\theta}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left\{-\frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})\right\}$$

and the prior distribution on $\boldsymbol{\theta}$ is

$$f(\boldsymbol{\theta}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Omega}|}} \exp\left\{-\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\tau})^\top \boldsymbol{\Omega}^{-1}(\boldsymbol{\theta} - \boldsymbol{\tau})\right\}.$$

The predictive distribution can be derived by integrating out $\boldsymbol{\theta}$ with respect to its prior

$$\begin{aligned} f(\mathbf{y}) &= \int f(\mathbf{y} \mid \boldsymbol{\theta}) f(\boldsymbol{\theta}) d\boldsymbol{\theta} \\ &= \int \exp\left\{\log f(\mathbf{y} \mid \boldsymbol{\theta})\right\} f(\boldsymbol{\theta}) d\boldsymbol{\theta} \\ &= \int f(\mathbf{y} \mid \hat{\boldsymbol{\theta}}) \exp\left\{\frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top \left(-\mathbf{X}^\top \boldsymbol{\Sigma}^{-1}\mathbf{X}\right)(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})\right\} f(\boldsymbol{\theta}) d\boldsymbol{\theta} \\ &= \frac{f(\mathbf{y} \mid \hat{\boldsymbol{\theta}})}{\sqrt{|2\pi\boldsymbol{\Omega}|}} \int \exp\left\{\frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top \left(-\mathbf{X}^\top \boldsymbol{\Sigma}^{-1}\mathbf{X}\right)(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})\right\} \exp\left\{-\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\tau})^\top \boldsymbol{\Omega}^{-1}(\boldsymbol{\theta} - \boldsymbol{\tau})\right\} d\boldsymbol{\theta} \end{aligned} \tag{6}$$

Take $\boldsymbol{\tau} = \hat{\boldsymbol{\theta}}$ and the predictive distribution simplifies to

$$\begin{aligned} f(\mathbf{y}) &= \frac{f(\mathbf{y} \mid \hat{\boldsymbol{\theta}})}{\sqrt{|2\pi\boldsymbol{\Omega}|}} \int \exp\left\{-\frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top \left(\mathbf{X}^\top \boldsymbol{\Sigma}^{-1}\mathbf{X} + \boldsymbol{\Omega}^{-1}\right)(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})\right\} d\boldsymbol{\theta} \\ &= \frac{f(\mathbf{y} \mid \hat{\boldsymbol{\theta}})}{\sqrt{|2\pi\boldsymbol{\Omega}|}} \sqrt{|2\pi \left(\mathbf{X}^\top \boldsymbol{\Sigma}^{-1}\mathbf{X} + \boldsymbol{\Omega}^{-1}\right)^{-1}|} \end{aligned} \tag{7}$$

Suppose $\mathbf{I}$ is the identity matrix and $\mathbf{J}$ is the Fisher information. In this case the Fisher information is a diagonal matrix with elements $n_I p_J$, $\mathbf{J} = \text{diag}\{n_I p_J\}$.

Model (2) implies $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$, and $\boldsymbol{\Omega} = \phi\mathbf{J}_1^{-1}$, where $\mathbf{J}_1 = \sigma^2\mathbf{I}$ is the Fisher information of a single observation. This setting simplifies the predictive distribution further and gives

$$\begin{aligned} f(\mathbf{y}) &= \frac{f(\mathbf{y}|\hat{\boldsymbol{\theta}})}{\sqrt{|2\pi\sigma^2\phi\mathbf{I}|}} \sqrt{\left|2\pi\text{diag}\left\{\frac{\sigma^2\phi}{n_I p_J \phi + 1}\right\}\right|} \\ &= \frac{f(\mathbf{y} \mid \hat{\boldsymbol{\theta}})}{\sqrt{\prod_{I=1}^{|\mathcal{C}^{\text{row}}|} \prod_{J=1}^{|\mathcal{C}^{\text{col}}|} (n_I p_J \phi + 1)}}. \end{aligned}$$

Thus

$$-2\log f(\mathbf{y}) = -2\log f(\mathbf{y}|\hat{\boldsymbol{\theta}}) + \sum_{I=1}^{|\mathcal{C}^{\text{row}}|} \sum_{J=1}^{|\mathcal{C}^{\text{col}}|} \log(n_I p_J \phi + 1). \tag{8}$$

But $f(\mathbf{y} \mid \boldsymbol{\theta})$ is a Gaussian likelihood so deriving (4) is straightforward. Substituting σ^2 with its empirical estimator s^2 simplifies (8) even further and gives (3).

7 Proof of Theorem 2

The proof is by contradiction. Here we only concentrate on rows i.e. supposing $\mathfrak{D} = \mathfrak{D}^{\text{row}}$ and $\mathfrak{M} = \mathfrak{M}^{\text{row}}$, and only focus on the complete bilinkage, but the argument is equally valid for other cases.

Suppose bicluster $\check{\mathbf{Y}} \subset \mathbf{Y}$ is separable, Figure 7 helps to follow the notation. From the separability we know $\mathfrak{M} > \mathfrak{D}$. Now assume $\mathbf{Y}_1 \subset \mathbf{Y}$ is merged with $\check{\mathbf{Y}}_1 \subset \check{\mathbf{Y}}$, at a certain step, before $(\check{\mathbf{Y}}_1, \check{\mathbf{Y}}_2)$ merge together, $\mathbf{Y}_1 \cap \check{\mathbf{Y}} = \emptyset$, and $\check{\mathbf{Y}}_1 \cup \check{\mathbf{Y}}_2 = \check{\mathbf{Y}}$. Suppose $\mathbf{y}_{1i}$ denotes the rows of $\mathbf{Y}_1$, $\check{\mathbf{y}}_{1i}$ denotes the rows of $\check{\mathbf{Y}}^{\text{row}}$, and $\check{\check{\mathbf{y}}}_{2i}$ denotes the rows of $\check{\check{\mathbf{Y}}}_2^{\text{row}}$, for some $\check{\check{\mathbf{Y}}}_2 \subset \check{\mathbf{Y}}_2$. By he definition of complete linkage, merging $\mathbf{Y}_1$ with $\check{\mathbf{Y}}_1$ means

$$\max_{i \neq i'} \|\mathbf{y}_{1i} - \check{\mathbf{y}}_{1i'}\| < \max_{i \neq i'} \|\check{\check{\mathbf{y}}}_{2i} - \check{\mathbf{y}}_{1i'}\|, \tag{9}$$

and by definition of diameter

$$\max_{i \neq i'} \|\check{\check{\mathbf{y}}}_{2i} - \check{\mathbf{y}}_{1i'}\| < \mathfrak{D}. \tag{10}$$

From (9) and (10)

$$\max_{i \neq i'} \|\mathbf{y}_{1i} - \check{\mathbf{y}}_{1i'}\| < \mathfrak{D},$$

which turns out to be a contradiction, because by separability of $\check{\mathbf{Y}}$

$$\min_{i \neq i'} \|\mathbf{y}_{1i} - \check{\mathbf{y}}_{1i'}\| > \mathfrak{D}.$$

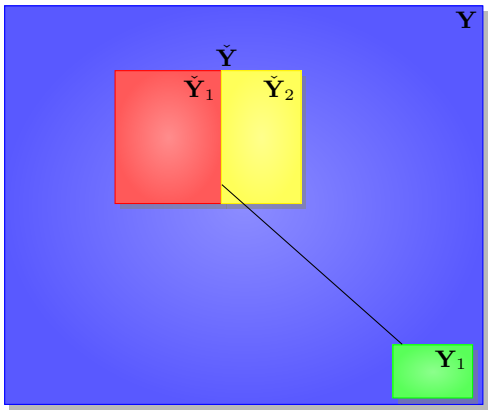

Figure 7: Notation for a separable bicluster $\check{\mathbf{Y}} \subset \mathbf{Y}$.

References

Akaike, H. (1973) Information theory and an extension of the maximum likelihood principle. In Second International Symposium on InformationTheory, eds B. N. Petrov and F. Csaki, 267–281.

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

Chen, G. K., Chi, E. C., Ranola, J. M. O. and Lange, K. (2015) Convex clustering: An attractive alternative to hierarchical clustering. PLoS Comput Biol 11(5), e1004228.

Cheng, Y. and Church, G. M. (2000) Biclustering of expression data. In International Conference on Intelligent Systems for Molecular Biology, 8, 93–103.

Eisen, M. B., Spellman, P. T., Brown, P. O. and Botstein, D. (1998) Cluster analysis and display of genome-wide expression patterns. Proceedings of the National Academy of Sciences 95(25), 14863–14868. Opne source C library for clustering.

Eren, K., Deveci, M., Küçüktunç, O. and Çatalyürek, Ü. V. (2013) A comparative analysis of biclustering algorithms for gene expression data. Briefings in Bioinformatics 14(3), 279–292.

Fowler, A. and Heard, N. A. (2012) On two-way bayesian agglomerative clustering of gene expression data. Statistical Analysis and Data Mining 5(5), 463–476.

Fraley, C. and Raftery, A. E. (1999) MCLUST: software for model-based cluster analysis. Journal of Classification 16, 297–306.

Gan, X., Liew, A. and Yan, H. (2008) Discovering biclusters in gene expression data based on high-dimensional linear geometries. BMC Bioinformatics 9, 209.

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

Hochreiter, S., Bodenhofer, U., Heusel, M., Mayr, A., Mitterecker, A., Kasim, A., Khamiakova, T., Sanden, S. V., Lin, D., Talloen, W., Bijnens, L., Ghlmann, H. W., Shkedy, Z. and Clevert, D. A. (2010) FABIA: factor analysis for bicluster acquisition. Bioinformatics 26, 1520–1527.

Hubert, L. and Arabie, P. (1985) Comparing partitions. Journal of classification 2(1), 193–218.

Ideker, T., Thorsson, V., Ranish, J. A., Christmas, R., Buhler, J., Eng, J. K., Bumgarner, R., Goodlett, D. R., Aebersold, R. and Hood, L. (2001) Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. Science 292(5518), 929–934.

Kaiser, S., Santamaria, R., Tatsiana, Khamiakova, Sill, M., Theron, R., Quintales, L. and Leisch., F. (2013) Biclust: BiCluster Algorithms. R package version 1.0.2.

Lance, G. N. and Williams, W. T. (1966) A general theory of classifactory sorting strategies, i. hierarchical systems. Computer Journal 9, 373–380.

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

Martella, F., Alfò, M. and Vichi, M. (2008) Biclustering of gene expression data by an extension of mixtures of factor analyzers. The International Journal of Biostatistics 4(1), 1–19.

McLachlan, G., Do, K.-A. and Ambroise, C. (2004) Finite Mixture Models. New York: Wiley.

Murtagh, F. and Legendre, P. (2014) Ward's hierarchical agglomerative clustering method: Which algorithms implement ward's criterion? Journal of Classification 31(3), 274–295.

Orzechowski, P. and Boryczko, K. (2016) Text mining with hybrid biclustering algorithms. In International Conference on Artificial Intelligence and Soft Computing, pp. 102–113.

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

Sheng, Q., Moreau, Y. and De Moor, B. (2003) Biclustering microarray data by Gibbs sampling. Bioinformatics 19, 196–205.

Smith, J., Anderson, P. and Liverani, S. (2008) Separation measures and the geometry of Bayes factor selection for classification. Journal of the Royal Statistical Society, Series B 70, 957–980.

Sokal, R. R. (1958) A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin 38, 1409–1438.

Sørensen, T. (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. Biologiske Skrifter 5, 1–34. First complete Linkage suggestion.

Tu, K. and Honavar, V. (2008) Unsupervised learning of probabilistic context-free grammar using iterative biclustering. In International Colloquium on Grammatical Inference, 224–237.

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

van Uitert, M., Meuleman, W. and Wessels, L. (2008) Biclustering sparse binary genomic data. Journal of Computational Biology 15, 1329–1345.

Xu, B., Bu, J., Chen, C. and Cai, D. (2012) An exploration of improving collaborative recommender systems via user-item subgroups. In Proceedings of the 21st International Conference on World Wide Web, 21–30.

Yeung, K., Medvedovic, M. and Bumgarner, R. (2003) Clustering gene-expression data with repeated measurements. Genome Biology 4, R34.

Yeung, K. and Ruzzo, W. (2001) Principal component analysis for clustering gene expression data. Bioinformatics 17, 763–774.

Zhang, J. (2010) A Bayesian model for biclustering with applications. Journal of the Royal Statistical Society, Series C 59, 635–656.