High-dimensional data and graph clustering with discrete latent variable models

Nicolas Jouvin

Université Paris 1 Panthéon-Sorbonne

École Doctorale Sciences Mathématiques de Paris Centre (ED 386)

Laboratoire : Statistiques, Analyse et Modélisation Multidisciplinaire, EA 4543

Classification non-supervisée de données de grande dimension et de graphes à l'aide de modèles à variables latentes discrètes

Par Nicolas Jouvin

Thèse de doctorat de Mathématiques Appliquées

Dirigée par
Charles Bouveyron et Pierre Latouche

Soutenue le 11 décembre 2020 devant un jury composé de :

Pierre ALQUIER	RIKEN AIP	Rapporteur
Charles BOUVEYRON	Université Côte d'Azur – INRIA	Codirecteur
Julien CHIQUET	INRAE	Examinateur
Pierre LATOUCHE	Université de Paris	Directeur
Alain LIVARTOSWKI	Institut Curie	Codirecteur
Andrea RAU	INRAE	Rapportrice
Tabea REBAFKA	Sorbonne Université	Examinatrice
Madalina OLTEANU	Université Paris Dauphine-PSL	Examinatrice

Après un avis favorable de : Pierre ALQUIER (RIKEN AIP)
Andrea RAU (INRAE)

Résumé

L'accumulation de données de natures extrêmement différentes, et impliquant un nombre de variables toujours plus élevé, constitue l'un des défis centraux de l'analyse de données contemporaine. Dans ce contexte, la classification non-supervisée, ou *clustering*, propose un ensemble de méthodes permettant de regrouper des objets similaires afin de gagner en connaissance sur leurs relations sous-jacentes. L'objet de cette thèse est le clustering de trois types de données spécifiques : les données multivariées continues et de comptage, ainsi que les réseaux.

Après une courte présentation de ces dernières, détaillant leurs spécificités ainsi que leurs possibles applications, nous présentons les principales approches en clustering, ainsi que les difficultés liées à l'analyse de données multivariées en grande dimension. Nous motivons ensuite l'approche probabiliste utilisée dans cette thèse, et proposons une revue détaillée du clustering à l'aide de modèles à variables latentes discrètes. En particulier, nous nous concentrons sur le cas des données de grande dimension, avec les modèles de mélanges finis s'appuyant sur la réduction de la dimension.

La suite de cette thèse développe ses trois contributions originales. Dans un premier temps, nous introduisons un nouvel algorithme pour le clustering de données de comptage en grande dimension. Ce dernier affiche de bonnes performances comparé aux approches similaires, notamment avec un faible nombre d'échantillons. Nous présentons également une application en clustering de rapports médicaux d'anatomopathologie, en collaboration avec l'Institut Curie. Ensuite, nous proposons un nouveau modèle de mélange Gaussien pour les données de grande dimension, ainsi qu'un algorithme de clustering basé sur une version non-supervisée de l'analyse factorielle discriminante. Les résultats sur données simulées et réelles montrent un réel avantage de notre méthode comparé à l'état de l'art, et des perspectives de généralisation sont proposées. Enfin, nous proposons un algorithme de clustering hiérarchique utilisant une version exacte de la vraisemblance classifiante intégrée, aussi appelée ICL exacte. Cette dernière contribution est double, consistant d'abord en un algorithme génétique permettant d'améliorer les heuristiques existantes, basées sur la maximisation d'un critère ICL exacte par recherche locale et gloutonne. Nous introduisons ensuite une nouvelle approximation asymptotique de ce critère, ainsi qu'une heuristique de clustering hiérarchique permettant de fusionner les clusters obtenus par le premier algorithme. Nous montrons comment cette approche est générique, applicable à tout modèle à variables latentes discrètes pour lequel l'ICL exacte est calculable. Ce dernier est dérivé en détail pour un ensemble de modèles standards, en particulier les modèles à blocs latents pour l'analyse de réseaux. Les résultats sur données simulées montrent une claire supériorité par rapport aux approches similaires, et l'algorithme hiérarchique se montre particulièrement utile pour l'analyse de données réelles.

Mots-Clefs : Classification non-supervisée, Inférence variationnelle, Réduction de la dimension, Sélection de modèle, Données de comptages, Données continues, Réseaux.

Abstract

Modern statistical analysis encounters a wide variety of data sets, rapidly growing both in size and dimensionality, with a need to efficiently summarize and represent them. To that end, clustering consists in grouping objects together, forming meaningful clusters giving insights regarding the underlying structure of the data. This thesis focuses on three particular types of data: multivariate continuous and count data, and also networks.

After a brief presentation of these data and their specific applications, we review popular approaches to clustering, as well as modern challenges related to high-dimensionality. Several arguments are given supporting model-based approaches, grounded on a probabilistic formulation, which is the preferred framework for this thesis. Then, a detailed introduction on discrete latent variable models for clustering is given, with a focus on finite mixtures integrating dimension reduction.

The three original contributions of this thesis come in the remaining chapters. First, a new algorithm for the clustering of high-dimensional count data is described, showing a real advantage over competing approaches, especially in low sample size settings. A medical application is detailed, with the clustering of anatomopathological text reports from Institut Curie hospital. Then, a new Gaussian mixture model for high-dimensional continuous data is presented, along with a clustering algorithm relying on unsupervised linear discriminant analysis. The latter compares favorably to state-of-the-art approaches on simulated and real-data benchmarks, and potential extensions are discussed. We finish by proposing a two-fold methodology for hierarchical clustering, based on an exact version of the integrated classification likelihood (ICL). The first part consists in improving existing greedy heuristics, using a carefully designed genetic algorithm to reduce sensitivity to local maxima of the exact ICL. Then, we consider a new asymptotic approximation of the latter giving rise to a hierarchical strategy, merging the clusters obtained from the first algorithm. We show how this approach is generically applicable in any discrete latent variable model for which exact ICL are tractable, and we detail derivations for standard ones. Simulations and real-datasets applications demonstrate the interest of this methodology, in particular for statistical network analysis.

Keywords: Clustering, Variational inference, Dimension reduction, Model selection, Count data, Continuous data, Network analysis

Remerciements

Pierre et Charles, je tiens à vous remercier tous les deux pour la qualité de votre encadrement durant cette collaboration qui dure depuis le stage maintenant. La thèse est une aventure aussi humaine que scientifique, et vous m'avez fait profiter de vos qualités sur ces deux plans. Ce manuscrit doit beaucoup à votre disponibilité, vos conseils, vos idées, votre patience et votre confiance durant ces trois années.

Je suis extrêmement reconnaissant envers Andrea Rau et Pierre Alquier pour leur lecture attentive du manuscrit ainsi que leurs rapports qui ont contribué à l'améliorer. Merci également à Julien Chiquet, Tabea Rebafka et Madalina Olteanu d'avoir accepté d'assister à cette soutenance. Madalina, merci pour ces années passées au SAMM et le partage de ton compte Zoom si précieux lors de ce premier confinement. Alain, merci à toi et à la direction des datas de l'Institut Curie de m'avoir accueilli parmi vous. J'en profite également pour dire un grand merci à Guillaume Bataillon pour sa disponibilité et son enthousiasme sans faille lors de notre collaboration. Finalement, je remercie Etienne Côme pour son investissement dans mon stage de master et dans la collaboration qui a suivi.

Au terme de ce parcours universitaire de huit ans, j'ai une pensée pour mes professeur·e·s. Plutôt allergique aux mathématiques, j'ai pu rencontrer des enseignantes et enseignants formidables à l'université, dont certain·e·s sont même devenu·e·s des collègues. En parlant de collègues, je suis chanceux d'avoir pu effectuer ma thèse dans un environnement aussi bienveillant et chaleureux que le SAMM, et je remercie celles et ceux qui y ont contribué. Merci à Xavier, Clément, Raphaël pour les pétanques ou apéros décompression qui ont jalonné cette thèse. Merci à Cynthia, Antoine, Kamila, Clara, Valentin, Marie, Laurent, Madalina, Jean-Marc, Florence, Denis, Alice, Julien, Fabrice pour les nombreuses discussions et débats interminables lors des déjeuners et pauses thé. J'espère sincèrement que vous pourrez manger autre part que dans un Elior dans un futur non-confiné... Merci également à Marco de m'avoir passé le flambeau de thésard avec encouragements et beaucoup d'articles à lire ! Mohammed et Bruno, je n'oublie pas non plus ces fameuses 15 minutes que vous me devez depuis trois ans, et je compte bien les réclamer pour un café le jour où je remettrais les pieds au SAMM. Merci aux doctorantes et doctorants du MAP5 qui avaient toujours un bureau à offrir lorsque je débarquais à l'improviste. Merci, enfin, à Margot, presque jumelle de thèse, pour son soutien dans les derniers moments.

Pour ce qui est de la famille, on pourrait se passer de mots, mais ça va quand même mieux en le disant. Merci d'abord à mes parents de m'avoir transmis une curiosité et un capital culturel si utiles dans mes études. Merci à ma sœur Léa d'avoir été là, malgré la distance, pour écouter mes états d'âme. Merci à mon frère Martin pour son flegme à toute épreuve, et ce qu'il convient d'appeler les meilleurs khatchapouris cachanais. Maman, je te dédicace tous les *s* manquants à la 3ème personne de ce manuscrit, je sais que tu as fait de ton mieux, mais je crains que mon cas ne soit trop grave. Papa, merci pour les bons petits repas de rédaction, tes multiples relectures et les commandes Git de derrière les fagots. Merci Marie-

Hélène pour l'accueil à Gif en début et fin de tunnel et tes encouragements. Et bien sûr, mention spéciale à la famille élargie : Ali, Monireh, Bamdad, merci pour votre soutien sans faille, j'aurais enfin une réponse à la question de savoir comment avance ma thèse ! Merci aussi à Martine, on aurait toutes et tous aimé assister à ce moment en présence du grand absent parti trop tôt.

Last but not least, les amies, les amis. Comment dire... Je vais exploser les quotas si je vous cite toutes et tous, mais vous savez que vous êtes dans mon cœur. Les innombrables soirées, festivals et vacances que j'ai pu passer avec vous sont parmi mes souvenirs les plus précieux. Des bancs de l'école Carnot aux bancs de la fac, du Dour Festival au camping de Saou, des Alpes aux côtes bretonnes, de la ZAD de Bretignolles aux teufs de Normandie en passant par le parc Raspail, que d'incroyables moments passés avec une troupe de *jaloux saboteurs* qui ne cesse de se croiser et de s'agrandir d'année en année. Et bien sûr, il y a Bélise. Peu de colocations pourront se targuer d'avoir survécu à une thèse et deux confinements. Belico c'est d'abord un mot de passe beaucoup trop long et compliqué, mais surtout une superbe tranche de vie aux côtés de deux personnes formidables. Vous allez me manquer. A vous toutes et tous, j'aurais aimé vous dire qu'on allait fêter ça comme jamais, mais les apéros Discord c'est *so* premier confinement... Je n'ai aucun doute sur le fait qu'on trouvera une occasion tôt ou tard, après tout, on n'a jamais eu besoin d'une thèse pour prétexter une fête !

Pour finir, je suis heureux d'avoir participé, aux côtés de personnes formidables, à toutes ces mobilisations dans et en dehors de l'université. Encore une fois, la liste est trop longue pour être écrite ici mais merci à toutes et tous pour ces moments : grêver, chanter, danser ou arpenter les pâvés de l'Ouest parisien à vos côtés sera toujours un plaisir ! Si la coutume voudrait que l'on s'en tienne aux remerciements, la période particulière que nous traversons toutes et tous, à l'université comme ailleurs, m'invite à y déroger. J'adresse donc mes non-remerciements les plus sincères à Frédérique Vidal, encore ministre de l'enseignement supérieur et de la recherche au moment où j'écris ces lignes, pour Parcoursup, le mal nommé «Bienvenue en France» et la loi de programmation de la recherche. Pour cette dernière, que l'on considère son contenu initial ou bien les amendements scélérats du Sénat, on balance entre coupure de la réalité des jeunes chercheuses et chercheurs précaires, et mépris total. D'une manière plus générale, le gouvernement d'Emmanuel Macron aura fait preuve d'un mépris et d'une sauvagerie inouïe à l'égard de sa population, dont les images, peut-être bientôt interdites, resteront gravées dans nos mémoires. «Le vieux monde se meurt, le nouveau monde tarde à apparaître et dans ce clair obscur surgissent les monstres» est une citation de Gramsci qui me parle particulièrement; mes pensées sont avec celles et ceux qui refusent l'indifférence et mènent la bataille des imaginaires en les mettant en pratique.

Contents

Essential notations

Variables

n	Number of observations.
p	Dimension or number of variables.
$\boldsymbol{Y}$	Observed data in $\mathbb{R}^{n\times p}$, $\mathbb{N}^{n\times p}$ or $\mathbb{N}^{n\times n}$.
i	Observation index in $\{1,\dots,n\}$.
j	Variable index in $\{1,\dots,p\}$.
K	Number of clusters.
$\boldsymbol{z}$	Unobserved discrete latent variable in $\{0,1\}^K$.
d	Dimension of the latent space.
$\boldsymbol{x}$	Unobserved continuous latent variable in $\mathbb{R}^d$.

Distributions

$\mathcal{S}_p^{++}$	Set of $p\times p$ symmetric positive definite matrices.
Δ_K	Unit simplex of dimension $K-1$: $\Delta_K := \{w \in \mathbb{R}_+^K \,:\, \sum_{k=1}^K w_k = 1\}$.
$\mathcal{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^p$ and covariance $\boldsymbol{\Sigma} \in \mathcal{S}_p$.
$\mathcal{M}_K(L, \boldsymbol{\pi})$	Multinomial distribution with L repetitions, K issues and probability vector $\boldsymbol{\pi} \in \Delta_K$.
$p(\boldsymbol{y} \mid \boldsymbol{\theta})$	Density of distribution p with parameter $\boldsymbol{\theta}$, evaluated at $\boldsymbol{y}$.

Operators

$\mathrm{H}(q)$	Entropy of the distribution q over a space $\mathcal{X}$: $-\int_{\mathcal{X}} q(\boldsymbol{x}) \log q(\boldsymbol{x}) \,\mathrm{d}\boldsymbol{x}$.
$\mathrm{KL}(q \,\|\, p)$	Kullback-Leibler divergence between two probability distributions over the same space $\mathcal{X}$: $\int_{\mathcal{X}} \log \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} q(\boldsymbol{x}) \,\mathrm{d}\boldsymbol{x}$.
$\mathrm{Tr}\left[\boldsymbol{A}\right]$	Trace of the square matrix $\boldsymbol{A}$: $\mathrm{Tr}\left[\boldsymbol{A}\right] = \sum_i A_{ii}$.
$\mathbb{E}_{\boldsymbol{\eta}}\left[f(\boldsymbol{\eta})\right]$	Expectation of $f(\boldsymbol{\eta})$ under the distribution of $\boldsymbol{\eta}$.

1

Introduction

Epistemologically, some dates the practice of grouping objects together back to the beginning of language, or to the ancient Greeks like Plato or Aristotle (Bouveyron et al. 2019, p. 2). Modern approaches seem to be somewhat more practical, motivated by the paradigmatic shift of data collection characterizing the last decades. Indeed, with datasets rapidly increasing in volume and dimensionality, comes a need to summarize them in order to gain insight of the relationships between objects or individuals. Clustering addresses this problem in a mathematical fashion, seeking to group n individuals into K distinct classes, or *clusters*. Often based on a notion of distance or on an underlying statistical model, the nature of the data at hand will greatly influence both the modelization and the results.

1.1 The data zoo

This thesis focuses on three different types of data. On the one hand, in the context of multivariate continuous or count data, the observations are often thought of as n individuals described by p variables. On the other hand, relational data such as networks encode

relationships between n objects. In this scenario, the observations are about the connections between objects, rather than their geometry in p-dimensional spaces. Each data type has its own specificity and range of clustering applications.

1.1.1 Continuous data

Ubiquitous in statistics, multivariate continuous data constitute one of the most popular data representations. The observation of n individuals in dimension p is often summarized in a so-called data matrix, with n rows and p columns containing real-valued entries. Usages are extensive in statistics, from Pearson's work (Pearson 1901), to DNA microarray (Draghici 2016, chap. 3) and large-scale image databases that appeared in the last decades (Deng et al. 2009). In this context, clustering is useful in several ways. First, discovering and classifying new groups of biological entities such as species, cell types or genes is a fundamental task in biological taxonomy (Hajibabaei et al. 2007). Moreover, tumor classification and grading is an essential tool of anatomic pathologist for both diagnostic and prognosis, especially in widespread diseases such as breast cancer (Lakhani 2012). Modern approaches use the combination of knowledge expertise as well as unsupervised discovery methods such as clustering to refine and improve classifications. Computer vision, studying digital images described by pixels, is another important example of continuous data representation. Supervised tasks have been the subject of a lot of attention, especially image classification where an algorithm is trained on labeled data to distinguish between pre-defined classes of images. Indeed, the last decades have known a rapid progress of modern algorithms, leading to their increased use in everyday technologies, not without ethical concerns (Chamayou 2013). Clustering techniques have been used in this context as a form of image quantization (Nowak et al. 2006; Jégou et al. 2010), combining features extracted via state-of-the-art methods such as convolutional neural networks (Krizhevsky et al. 2012), or local descriptors (Lowe 1999; Bay et al. 2008), into a single image representation fed to a classifier (see Figure 1.1). Moreover, the task of image segmentation, looking for groups of spatially coherent areas in an image, such as cancerous cell tissues in medical imagery, is inherently linked to clustering (Coleman and Andrews 1979). Thus, both the modeling of the data as well as the interpretability of the resulting clusters appear to be primarily important in a number of applications. This thesis builds on statistical approaches for clustering, explicitly encoding assumptions on the underlying model of the data.

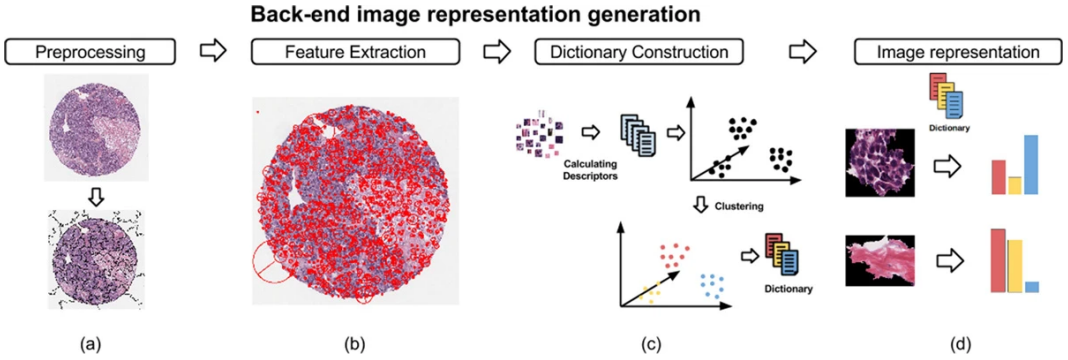


Figure 1.1: Workflow of an image classification pipeline: an illustration with histopathological images. Clustering is useful in the third step to aggregate different image features into a single representation. This figure was reproduced from Ding et al. (2015, Figure 2) (with permission).

1.1.2 Count data

Count data is used in many scientific fields in the form of frequency counts for instance as the occurrences of p distinct words in a *bag-of-words* model for text analysis (Aggarwal and Zhai 2012a), or as *read* counts obtained by the thriving use of next generation sequencing in genomics (Anders and Huber 2010). In ecology, a lot of studies also focus on abundance count data, representing ecosystems as the number of occurrences of p species (Fordyce et al. 2011). In this context, the data matrix $\boldsymbol{Y}$ contains n observations in dimension p, only its entries are now positive integers. With the increase in volume and dimensionality of these datasets, there is an interest in summarizing them with the help of new statistical tools, looking for groups of co-expressed genes or meaningful partitions of documents in text corpora. However, this type of data has a peculiar geometry and the computations of similarity, which is central in many continuous clustering algorithms, needs to be handled carefully. To that end, specific similarity functions comparing histograms have been proposed, such as Csiszar-Rényi divergence (Van Erven and Harremos 2014). Moreover, when applied to count data, most of the standard statistical hypothesis acceptable for continuous data, *e.g.* Gaussianity, fall apart. On the one hand, transformations of the raw data have been proposed to meet the normality assumptions, such as log-transforms in biology and ecology (Zwiener et al. 2014; St-Pierre et al. 2018), or the well-known term frequency-inverse document frequency in text analysis (Ramos 2003). While it is not the purpose of this thesis to discuss whether these modifications are statistically well grounded, we point out the work of Osborne (2005) and O'hara and Kotze (2010), who emphasized that caution should be taken when using such transformations. On the other hand, statistical models for count data, relying on probabilistic assumptions about the generative process of raw observations, have recently received an increasing amount of attention and developments (Fruhwirth-Schnatter et al. 2019, chap. 9).

Document-term matrix					
Documents \ Terms	lesions	ductal	...	lobular	metaplasia
"Cancerous lesions (...) ductal carcinoma"	1	1	...	0	0
"Cancerous lesions (...) lobular carcinoma"	1	0	...	1	0
"Benign lesions (...) metaplasia"	1	0	...	0	1

Figure 1.2: A classical text representation in bag-of-words models: each document is represented as the vector of its word counts. This artificial example relates to the clustering of anatomopathological reports described in Chapter 3.

1.1.3 Network data

Networks are extensively used in a wide range of scientific applications, modeling the interactions between a set of objects. The best known examples stem from social sciences (Palla et al. 2007), with a long line of work highlighted and nourished by the rapid expansions of digital social networks. Other popular uses include, but do not limit to, biology and

bioinformatics, modeling interactions between proteins (Barabasi and Oltvai 2004) or gene transcription regulation networks (Shen-Orr et al. 2002), modelization of Internet networks (Liu et al. 2006), or even analysis of co-authorship networks in scientific fields, with a famous example among mathematicians (Goffman 1969).

A graph $\mathcal{G}$ is traditionally defined as the collection $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{1, \dots, n\}$ a set of vertices and $\mathcal{E}$ a set of edges connecting them. This representation may be summed up in an $n \times n$ adjacency matrix $\boldsymbol{Y}$ with binary entries y_{ij} encoding the presence or absence of an edge between nodes i and j. Until now, we considered n observations in dimension p, however we emphasize that the case of graphs is different since, while there are n nodes, the observations are the n^2 edges and non-edges. Graph clustering consists in finding a partition of $\mathcal{V}$, and can be divided in two main methodologies. Community detection (Fortunato 2010) defines clusters as sets of nodes which tend to connect more with each other in an *associative* way, as represented in Figure 1.3. It is often dealt with model-free approaches such as the modularity score (Newman and Girvan 2004). Conversely, the second approach is about finding *disassortative* structures, where vertices predominantly connect with other classes, as in bipartite networks (Zhou et al. 2007). Our interest lies in probabilistic models for graph clustering, which can tackle both of these approaches.

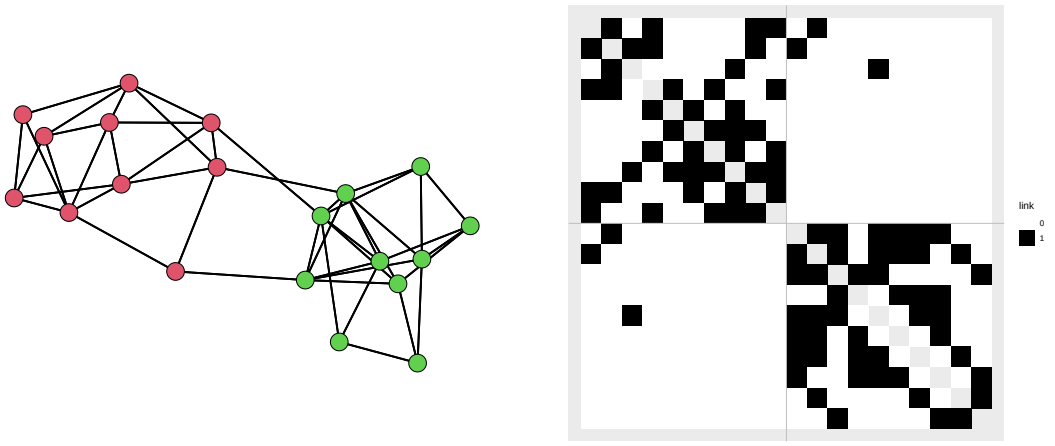


Figure 1.3: An undirected network with 20 nodes and community structure. The graph diagram (left) as well as its adjacency matrix (right) are displayed, and color indicates class membership. Real-world networks clustering applications are presented in Chapter 5.

1.2 Clustering

Formally, clustering searches for a partition $\mathcal{P}$ of the set $\{1, \dots, n\}$ into K distinct and non-empty subsets, maximizing some criteria supposed to adequately describe the underlying structure of the data. However, clustering inherently is a discrete, combinatorial optimization problem, as the number of distinct solutions is $S(n, K)$, the Stirling number of the second kind (Ronald L. Graham 1988, p. 244). Since a complete enumeration is not generally feasible, one has to rely on heuristics or relaxation of the problem. As for the design of criteria, two main approaches have been proposed.

1.2.1 Geometric approaches

On the one hand, similarity-based approaches define ad hoc criteria translating geometric assumptions on the desired clustering structure. Seeking to group similar objects together, they rely on a measure of affinity between observations. These so-called similarity functions are supposed to efficiently describe the desired cluster geometry.

Working with multivariate observations, one of the most popular instances of this approach is the k-means algorithm (MacQueen 1967). Geometrically, it seeks dense, well-separated point-clouds in the sense of the Euclidean norm, minimizing the sum of within-cluster distances. The problem being highly non-convex and combinatorial, the k-means algorithm consists in an iterative greedy heuristic exploring the space of partitions. Starting from an initial partition it cycles through the two following steps until convergence: first, the K barycenters of the current partition are computed, then the partition is updated by assigning each data point to the cluster of its nearest barycenter. Very popular, in part for its computational efficiency, the k-means has been generalized to other similarity functions such as the l_1 norm with the k-medians algorithm (Bradley et al. 1997), Bregman divergences (Banerjee et al. 2005), or kernel functions which implicitly non-linearly maps data to high-dimensional spaces before computing Euclidean distances (Dhillon et al. 2004).

Other popular clustering methods construct a weighted *affinity* graph using some similarity function between observations, expressing local connectivity between objects. Then, the problem of clustering may be defined as finding sets of densely connected nodes, *cutting* the graph into K groups. Different criteria have been proposed, usually related to a generalized eigenvalue problem involving the graph Laplacian matrix, hence the *spectral* clustering terminology (Ng et al. 2002). Based on local similarity, these methods take advantage of the manifold structure in the data, and are extensively used to uncover clusters with arbitrary shapes, or in the case of non-linearly separable clusters. Connections have also been made with weighted versions of kernel k-means (Dhillon et al. 2004).

Similarity-based methods assign each object to a unique cluster, in a so-called *hard* clustering fashion. This approach may contain some drawbacks, especially regarding the quantification of uncertainty. For instance, Figure 1.4b shows an example where assigning data points lying midway between two clusters may not be relevant and could influence the results more than desired.

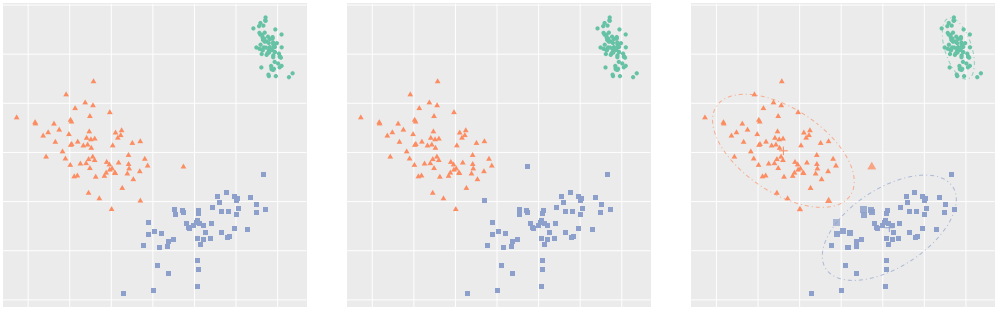
1.2.2 The rise of probabilistic methods

On the other hand, motivated by the development of statistical analysis, model-based approaches soon became popular as they offer a principled approach to clustering. Grounded on statistical theory, they posit a probabilistic model supposed to have generated the data conditionally on the partition. Then, the latter is considered as an unobserved, discrete random variable, and clustering is cast as an inference problem, seeking the partition that best fits the observed data according to the underlying statistical model.

The most popular example involves finite mixture models (McLachlan and Peel 2004), where a cluster is characterized by some parametric distribution, called mixture component. Then, objects belonging to the same cluster are supposed to be independent and identically distributed (*i.i.d*) from the corresponding component. The resulting distribution of the observations may be written as a convex combination over components, hence the *mixture* terminology. Parameter estimation is commonly done by means of an Expectation-Maximization algorithm (McLachlan and Krishnan 2007), and, the partition being treated as a random variable, probabilities of membership can be computed for each observation.

This mixture modeling approach is very general and applies to a wide range of data, including continuous and discrete multivariate data but also longitudinal, ordinal and functional data. Moreover, it is the building block for many other model-based approaches, such as block modeling for co-clustering and graph clustering (Govaert and Nadif 2003; Daudin et al. 2008). In this context, model-based approaches constitute an appealing framework to deal with the variety of data at hand in this thesis.

The statistical framework offers several advantages to model-based clustering over similarity-based methods. First, it allows to quantify uncertainty about the results using a probabilistic interpretation of the partition. In this *soft* clustering, an object belongs to every cluster with a certain probability of membership, as illustrated in Figure 1.4c. Moreover, model selection may then be used as a natural way to deal with the problem of choosing the number of clusters K, while other methods would rely on ad hoc heuristics. Finally, many of the similarity-based criteria may be recovered as particular instances of model-based approaches, the latter offering more flexibility and interpretability in the modelization. For instance, the k-means objective corresponds to the likelihood in an homoscedastic and isotropic Gaussian mixture model with equal proportions.



(a) True clustering **(b)** k-means **(c)** Gaussian mixture model

Figure 1.4: A toy example with 3 ellipsoid clusters of different shapes highlighting the interest of model-based clustering. (a) shows the true clustering where color and shape indicate clustering membership. (b) The k-means makes spherical assumptions on the shape of clusters and does not exactly recover the true partition because it is influenced by points lying in between clusters. (c) The Gaussian mixture model fit 2-dimensional Gaussian ellipses of different shapes and orientations, and assigns a membership probability to every point. Data points lying close to an ellipse center have a highly peaked membership towards the corresponding component, while those lying in the frontier of two clusters have balanced membership probabilities, thus more uncertainty, which is represented by a bigger point size. The hard clustering is obtained by taking the most probable cluster assignment. Chapter 4 introduces a new constrained Gaussian mixture model for the clustering of high-dimensional continuous data.

1.2.3 A detour by hierarchical clustering

The partitional framework described above supposes to work with a fixed number of clusters K. An alternative approach is proposed in hierarchical clustering. Inspired by the biological taxonomy problem of grouping species together in a nested fashion, from finer to coarser, the works of Sneath (1957) and Sokal and Michener (1958) paved the way for the so-called *agglomerative* clustering methods. Starting from the trivial partition with $K = n$, clusters are iteratively merged together to obtain a coarser partition until $K = 1$ and no more fusions are possible. Again, the decision of a fusion at each stage is based on some criterion

to be optimized, defining a specific cluster geometry. Traditional criteria are similarity-based such as the minimum sum-of-square, akin to k-means objective (Ward Jr 1963), and the complete and single linkage. A comprehensive summary of similarity-based hierarchical methods with corresponding references is given by Everitt et al. (2011, Table 4.1), along with a qualitative description of the induced cluster geometry. Model-based criteria have also been proposed to select the best fusion, such as the classification likelihood for mixture models (Banfield and Raftery 1993; Fraley 1998), or integrated likelihoods in a Bayesian setting (Heller and Ghahramani 2005). A mirror approach to agglomerative hierarchical clustering is the divisive methodology, where clusters are iteratively split using fast heuristics like k-means (see Everitt et al. 2011, sec. 4.2).

All these methods have in common to produce a set, or path, of nested partitions, meaning that a coarser partition results from the fusion of clusters at a lower level of the hierarchy. A common representation of the latter is via a tree-like diagram, called a dendrogram, which efficiently sums up the fusions made at each step. Akin to phylogenetic trees, the nodes of a dendrogram represent clusters while its branches represent fusions with height defining the order: the furthest from the leaves, the coarser the partition. This dendrogram can then be investigated and pruned at some stage by the user, avoiding the choice of a fixed K. A popular way to select a relevant number of clusters is via the *elbow rule*. This heuristic searches for the level K for which the marginal gain of one fusion starts to remarkably diminish.

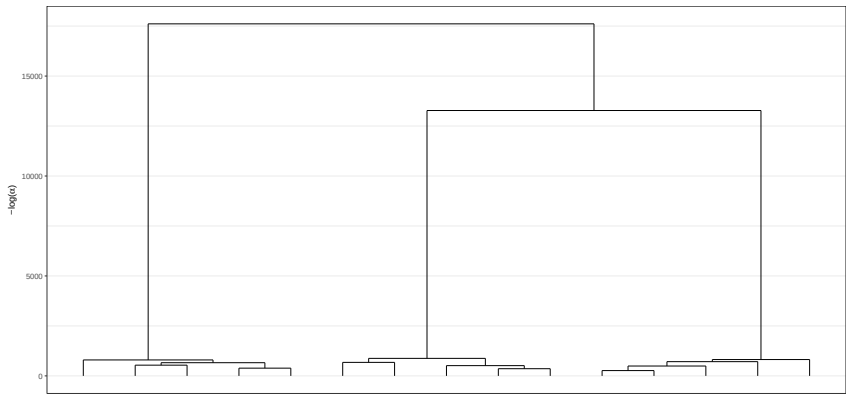


Figure 1.5: Dendrogram extracted by the hierarchical algorithm of Chapter 4 (see motivating example of Figure 5.1). The leaves are iteratively merged, the smaller height representing the first fusions. The y-axis quantifies the marginal cost of each fusion and the elbow heuristics would cut the tree at 3 clusters, which corresponds to the true hierarchical structure in this simulation.

1.2.4 What is a good clustering ?

Despite a somewhat clear-cut mathematical description, clustering may be considered as an ill-posed problem in its essence. Indeed, if the goal is to optimize some criterion to unveil relevant structure, the very definition of relevant does not necessarily possess a robust mathematical formulation and heavily depends on the context and practicalities. Therefore, the notion of a *true* or *good* clustering is always related to the modeling choice of the practitioner, which itself arises from subjective goals that needs to be explicit before analysis. In this context, model-based approaches are particularly attractive as they explicitly encode

these hypotheses and objectives in the generative process of the data (Fruhwirth-Schnatter et al. 2019, section 8.1). These somewhat epistemological concerns extend to any unsupervised learning method, such as unsupervised dimension reduction or variable selection, when there is no observed ground truth to guide the objectives. Thus, this thesis is placed within the model-based framework and proposes different models and algorithms for partitional (Chapters 3 and 4) and hierarchical clustering (Chapter 5), applied to data of different nature.

1.3 The challenges of high-dimensional clustering

Modern data collection has shifted from the design of experiments measuring few, precisely designed observables, to the acquisition of an ever-growing number of variables (Mattei et al. 2016, chap. 1). Datasets with a number of observations n comparable to, or even below the dimension p, is now commonly encountered in daily data analysis for a variety of fields, such as biology (Clarke et al. 2008), image processing (Deng et al. 2009) or text mining (Aggarwal and Zhai 2012b).

The term *curse of dimensionality* was first coined by Bellman (1957) in the preface of his book, and is nowadays used to designate a range of counterintuitive phenomena occurring in high-dimensional spaces. A popular example is the so-called *empty space* phenomenon (Scott and Thompson 1983), denoting the fact that the number of points needed to cover the unit hypercube $[0;1]^p$ with an evenly spaced grid grows exponentially with the dimension. Thus, unless we make additional assumptions or dispose of an impractical number of points n, we can assume that neighborhoods of high-dimensional spaces are essentially empty. As clustering, in its mathematical formulation, involves the computation of distances or statistical estimation, this has consequences for both similarity-based and model-based approaches. For the former, Steinbach et al. (2004) explain the difficulties of relying on the notion of distance or similarity in this scenario, along with a review of methods addressing this issue. As for the latter, Giraud (2014, p. 5) discusses in detail the peculiar geometry of high-dimensional spaces, giving a detailed example where classical statistical quantities such as averages have no hope to be robust with reasonable sample sizes. Moreover, other statistical phenomena fall under this curse, related to the size of the parameter space which is often linked to the dimension. For instance, the popular Gaussian mixture model for continuous data clustering, which is introduced in Chapter 2, involves covariance matrices growing as the square of the dimension, which quickly becomes over-parameterized without further constraints. On a side note, the difficulties when sampling from high-dimensional multimodal posteriors are well known in Bayesian inference, with an active line of research on *adaptive* Markov-Chain Monte-Carlo algorithms (Durmus 2016).

Several ways have been proposed to circumvent these issues. Among them, variable selection (Guyon and Elisseeff 2003) will seek to find a subset of $d < p$ relevant variables. In the context of clustering, Witten and Tibshirani (2010) introduced a similarity-based framework relying on sparsity, with lasso-like l_1 penalty (Tibshirani 1996). Model-based approaches to feature selection, which is cast as a model selection problem, were also proposed in the framework of mixture models (Raftery and Dean 2006; Maugis et al. 2009).

An alternative way to tackle the curse of dimensionality is through dimension reduction, where the data is assumed to lie in low-dimensional subspaces. The most popular instance is arguably principal component analysis (PCA, Jolliffe 2002), which finds the optimal linear mapping between the latent and observed subspaces in the sense of squared reconstruction error. In particular, a fruitful part of the literature stems from probabilistic formulations of

the latter relying on a Gaussian model (Tipping and Bishop 1999a), allowing generalizations to other types of distributions (Chiquet et al. 2018). Non-linear embeddings have also been proposed such as kernel versions of PCA or manifold learning (Verleysen 2007). Moreover, we emphasize that variable selection and dimension reduction are not mutually exclusive and may be combined (Bouveyron and Brunet-Saumard 2014; Mattei et al. 2016).

Chapters 3 and 4 of this thesis build on probabilistic models integrating clustering and linear dimension reduction to deal with multivariate count and continuous data.

1.4 Application to medical data and organization of the thesis

This thesis was done in collaboration with the Institut Curie hospital, and part of this work stems from fruitful discussions with medical practitioners. A tight collaboration was made with the anatomopathology service, and in particular with Dr. Guillaume Bataillon who put tremendous amounts of expertise into analyzing clustering results. Although these works do not yet lead to direct and concrete applications inside the hospital, there has been a lot of research and practical work done. Notably, significant advances were made in the comprehension of information present inside sometimes more than 15 years-old data, before the so-called *big data* era and careful acquisition routines were implemented. In particular, although being rather underrepresented in this final manuscript, the image analysis part has been the subject of a lot of efforts, with all the, sometimes difficult, aspects of *real* data analysis.

Having briefly described clustering along with accompanying modern challenges in this first chapter, Chapter 2 introduces model-based approaches along with variational inference which is the preferred statistical tool in this thesis. Discrete latent variable models are proposed as the main framework of this thesis, encompassing probabilistic models for the clustering of multivariate observations and networks. Then follows a detailed discussion of geometric and probabilistic linear dimension reduction methods for multivariate continuous and discrete data, after which statistical approaches integrating both clustering and linear dimension reduction are reviewed.

Original contributions of this thesis are then detailed in the next chapters. Chapter 3 addresses the problem of high-dimensional count data clustering, incorporating dimension reduction with the mixture of multinomial PCA distribution. A new clustering algorithm is proposed, leveraging on particular properties of the classification likelihood, and thorough numerical simulations assess the interest of our method in high-dimensional settings compared to state-of-the-art approaches. An application to the clustering of medical reports of Institut Curie is also presented. In Chapter 4, we discuss the notion of unsupervised linear discriminant analysis, injecting clustering information into the search of an optimal subspace. We introduce a Bayesian extension of the discriminative latent mixture model of Bouveyron and Brunet (2012a) and derive a new clustering algorithm for high-dimensional continuous data. Performances and stability of the proposed methodology are compared with state-of-the-art subspace clustering algorithms on several high-dimensional settings, showing a real interest for our method. Then, Chapter 5 focuses on hierarchical clustering in the general framework of discrete latent variable models, with two contributions. First, a genetic algorithm is proposed to improve greedy partitional clustering heuristics maximizing an exact integrated classification likelihood criterion. Then, a new approximation of the latter is introduced in order to derive a complete hierarchical agglomerative heuristic. Eventually, these two contributions working with similar objectives may be used together, the first one as an initialization for the second. Simulated and real-data experiments show

the superiority of the genetic algorithm over related approaches, and the interest of the hierarchical heuristic is illustrated for both visualization and real-data analysis. Finally, Chapter 6 is dedicated to a quick overview of these contributions as well as several leads for ongoing and future works.

2

Model-based clustering and dimension reduction

This chapter introduces the main models and existing methods that this thesis relies on. After a general discussion on the EM algorithm for latent variable models (LVMs) in Section 2.1, we detail its application to the main frameworks of both model-based clustering and dimension reduction in Sections 2.2 and 2.3 respectively. Then, we introduce popular models integrating both methods in an unified probabilistic framework. Finally, we review the main criteria used for model selection in a clustering context in Section 2.5.

2.1 Inference in latent variable models

This section introduces the principal inference methods used in this thesis. It is purposely general, and we will give detailed instantiations of the algorithms described hereafter throughout the rest of this chapter.

2.1.1 Latent variable models

When modeling complex phenomena, one often specifies the data generation process which led to the observation of $\boldsymbol{Y}$. Such an approach is called *generative*, where the observation of $\boldsymbol{Y}$ is viewed as the marginal consequence of a broader and complex scenario, and often implies hidden, unobserved, latent variables $\boldsymbol{\eta}$. In this scenario:

$$p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) = \int p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta}) \, \mathrm{d}\boldsymbol{\eta}, \tag{2.1}$$

where $\boldsymbol{\vartheta}$ denote a set of *parameters* to be estimated, *e.g.* by maximum likelihood. The left-hand side represents the *observed*-data likelihood, whereas the integrand in the right-hand side represents the *complete*-data likelihood. The rationale behind this approach can be twofold:

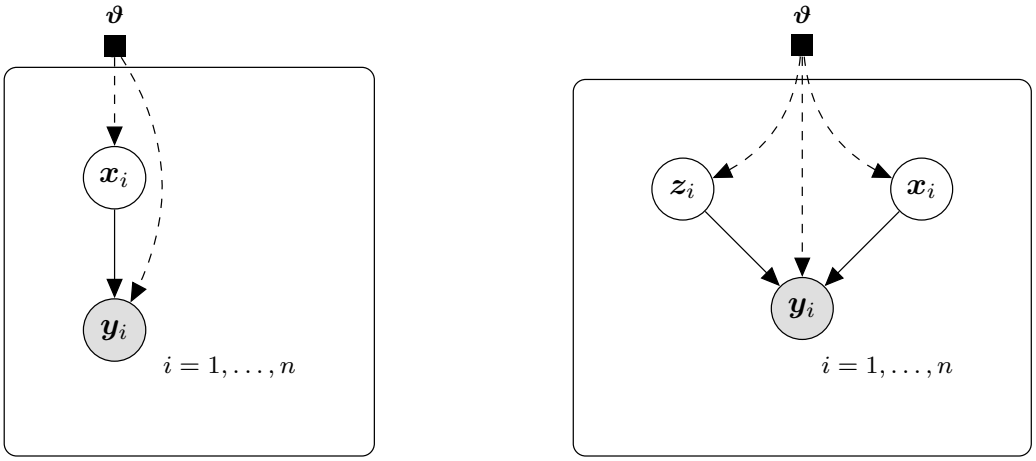
- the underlying physical phenomena at hand really motivates the introduction of η, which is a non-observable variable we want to *estimate* or *control* for. This is particularly the case in social and behavioural science, which has made an extensive use of latent variable models over the last decades (Everett 2013),
- the marginal distribution $p(\boldsymbol{Y} \mid \boldsymbol{\vartheta})$ may be intractable, in which case the introduction of $\boldsymbol{\eta}$ may conveniently lead to tractable inference procedure, as studied in Paquet (2008) and illustrated in Section 2.3.2.b.

Let us emphasize that this approach lies in-between the Bayesian and frequentist paradigms, since a fully Bayesian modelization would treat $\boldsymbol{\vartheta}$ as an additional latent variable, putting a prior on $(\boldsymbol{\eta}, \boldsymbol{\vartheta})$. In the following, we use semicolon notations to separate between random variables and fixed parameters.

This thesis focuses on model-based clustering and probabilistic dimension reduction. Therefore, the latent variables can be:

- discrete, denoted as $\boldsymbol{z} \in \{0, 1\}^K$, *e.g.* a clustering of the observed data as detailed in Section 2.2,
- continuous, denoted as $\boldsymbol{x} \in \mathbb{R}^d$, *e.g.* a lower dimensional representation of the observed data as encountered in Section 2.3,
- a combination of both, $\boldsymbol{\eta} = (\boldsymbol{X}, \boldsymbol{Z})$ as explained in Section 2.4.

As for the observed variable $\boldsymbol{Y}$, it can be as diverse as described in Chapter 1: continuous vectors in $\mathbb{R}^{n \times p}$, discrete data in $\mathbb{N}^{n \times p}$, or an $n \times n$ adjacency matrix representing a graph. The graphical model representation is often used for such models (Bishop 2006), specifying the conditional dependencies between the observed and the latent variables. Shaded nodes represent observed variables. Arrows represent conditional dependence of the in-node with respect to the out-node, while the absence of arrow expresses independence. The plate notation is used to represent repeating sub-graphs in the whole graphical model. Figure 2.1 shows the graphical representations of some latent variable models encountered in this thesis.



(a) Probabilistic PCA (PPCA)

(b) Mixture of PPCA

Figure 2.1: Examples of standard LVMs for model-based clustering and dimension reduction. Model (a) is introduced in Section 2.3.2.a on page 33 and Model (b) is introduced in Section 2.4.1 on page 37.

2.1.2 The expectation-maximization algorithm

In a frequentist approach, one seeks to estimate $\boldsymbol{\vartheta}$ following the maximum likelihood principle, which is solving the general problem:

$$\boldsymbol{\vartheta}^\star = \arg\max_{\boldsymbol{\vartheta}} \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}). \tag{MLE}$$

However, there can be several reasons that makes this particular problem difficult to solve directly. This is particularly the case in latent variable models, where the complete-data likelihood typically provides a much simpler expression than the observed one. In this specific context, Dempster et al. (1977) introduced a general algorithm to tackle the problem of finding the MLE. In an optimization point-of-view, we will see that it is basically a coordinate ascent on a lower bound of the observed-data log-likelihood, but first let us describe the intuition behind it. Suppose that the knowledge of $\boldsymbol{\eta}$ simplifies the problem of estimation, that is we can define a surrogate problem $\arg\max_{\boldsymbol{\vartheta}} f(\boldsymbol{Y}, \boldsymbol{\eta}; \boldsymbol{\vartheta})$ which is simpler to solve than the MLE. Naturally, $\boldsymbol{\eta}$ is not observed, however we observe it through $\boldsymbol{Y}$, hence we may try to *estimate* it. The natural distribution over $\boldsymbol{\eta}$ for such a task is the posterior: $p(\boldsymbol{\eta} \mid \boldsymbol{Y}; \boldsymbol{\vartheta})$. However, computing this distribution requires the knowledge of $\boldsymbol{\vartheta}$, which we seek to estimate in the first place. This chicken-and-egg situation suggests for an iterative scheme where one finds the most probable $\boldsymbol{\eta}$ given the observation of $\boldsymbol{Y}$ and a set of fixed parameters $\boldsymbol{\vartheta}$, and then updates $\boldsymbol{\vartheta}$ by solving the simpler surrogate problem.

Formally, let q be any probability distribution over $\boldsymbol{\eta}$, with $q(\boldsymbol{\eta}) > 0$. The objective in

problem (MLE) can be bounded below by:

$$\begin{aligned}
\log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) &= \log \int p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta}) \, \mathrm{d}\boldsymbol{\eta}, \\
&= \log \int p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta}) \frac{q(\boldsymbol{\eta})}{q(\boldsymbol{\eta})} \, \mathrm{d}\boldsymbol{\eta}, \\
&= \log \mathbb{E}_{\boldsymbol{\eta} \sim q} \left[\frac{p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta})}{q(\boldsymbol{\eta})} \right], \\
&\geq \mathbb{E}_{\boldsymbol{\eta} \sim q} \left[\log \frac{p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta})}{q(\boldsymbol{\eta})} \right], \\
&= \mathbb{E}_{\boldsymbol{\eta} \sim q} \left[\log p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta}) \right] + \mathrm{H}(q) \quad \text{with: } \mathrm{H}(q) = -\mathbb{E}_{\boldsymbol{\eta} \sim q} \left[\log q(\boldsymbol{\eta}) \right], \\
&:= \mathcal{J}(q; \boldsymbol{\vartheta}),
\end{aligned}$$

with Jensen's inequality applied to the log in the fourth line. This function $\mathcal{J}$ depends on $\boldsymbol{\vartheta}$ as well as the additional distribution q, and is often referred to as the evidence lower bound (ELBO) in a Bayesian context. The key property of $\mathcal{J}$ is the following proposition, that quantifies its gap to the observed-data likelihood in term of Kullback-Leibler divergence to the posterior.

Proposition 2.1. *For any pair* $(q, \boldsymbol{\vartheta})$*,*

$$\log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) - \mathcal{J}(q; \boldsymbol{\vartheta}) = \mathrm{KL}(q \,\|\, p(\cdot \mid \boldsymbol{Y}; \boldsymbol{\vartheta})) \geq 0. \tag{2.2}$$

Thanks to the non-negativity of the Kullback-Leibler divergence, Proposition 2.1 is consistent with $\mathcal{J}$ being a lower bound to the observed-data log-likelihood. However, its main interest lies in the fact that maximizing $\mathcal{J}$ w.r.t q amounts to minimize the KL between q and the posterior. This is the essence of the Expectation-Maximization algorithm proposed by Dempster et al. (1977), which, starting from an initial value of the parameters $\boldsymbol{\vartheta}^{(0)}$, consists in an iterative scheme decomposed into two steps at iteration $(t+1)$:

$$q^{(t+1)} = \arg\max_{q} \mathcal{J}(q; \boldsymbol{\vartheta}^{(t)}) \quad = \arg\min_{q} \mathrm{KL}(q \,\|\, p(\cdot \mid \boldsymbol{Y}; \boldsymbol{\vartheta}^{(t)}), \tag{E-step}$$

$$\boldsymbol{\vartheta}^{(t+1)} = \arg\max_{\boldsymbol{\vartheta}} \mathcal{J}(q^{(t+1)}; \boldsymbol{\vartheta}) = \arg\max_{\boldsymbol{\vartheta}} \mathbb{E}_{\boldsymbol{\eta} \sim q^{(t+1)}} \left[\log p(\boldsymbol{Y}, \boldsymbol{\eta} \mid \boldsymbol{\vartheta}) \right]. \tag{M-step}$$

The E-step gets its name from the fact that it allows to compute the expectation in $\mathcal{J}$, which is in turn maximized with respect to the parameters in the M-step. For a given $\boldsymbol{\vartheta}$, the optimal $q^\star$ in the E-step is the posterior $q^\star(\boldsymbol{\eta}) = p(\boldsymbol{\eta} \mid \boldsymbol{Y}; \boldsymbol{\vartheta})$, which makes the bound tight: $\mathcal{J}(q^{(t+1)}; \boldsymbol{\vartheta}^{(t)}) = \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}^{(t)})$. As for the M-step, the optimization program to solve is supposed to be easier than (MLE) and the updates depend on the generative model at hand. Assuming that both the E and M steps can be solved efficiently, this algorithm is quite universal for latent variable models and the following proposition states a useful convergence result.

Proposition 2.2. *The EM algorithm generates a sequence* $\{\boldsymbol{\vartheta}^{(t)}\}_t$ *inducing a monotonically increasing log-likelihood:*

$$\forall t, \quad \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}^{(t+1)}) \geq \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}^{(t)}) \tag{2.3}$$

Proof. We have:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}^{(t+1)}) \underbrace{\geq}_{\text{Prop. 2.1}} \mathcal{J}(q^{(t+1)}; \boldsymbol{\vartheta}^{(t+1)}) \underbrace{\geq}_{\text{M-step}} \mathcal{J}(q^{(t+1)}; \boldsymbol{\vartheta}^{(t)}) \underbrace{=}_{\text{E-step}} \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}^{(t)}).$$

□

Remark 1. At this level of generality, this result is only about the convergence of the log-likelihood to some local maxima or saddle point. However, nothing can be said about the convergence of the parameters $\{\boldsymbol{\vartheta}^{(t)}\}_t$ without further assumptions. This may be considered quite weak regarding the inference task, which is about finding the MLE. Fortunately, Wu (1983) fixed some proofs and claims originally made in Dempster et al. (1977), and gave quite mild sufficient conditions for the convergence of the sequence $\{\boldsymbol{\vartheta}^{(t)}\}_t$ to stationary points or local maxima of the likelihood. These conditions are verified in a large class of statistical models, such as exponential families. Finding general guarantees is still an open problem (Balakrishnan et al. 2017). In practice, most implementations of EM stop when a user-defined number of iterations is reached, or when the absolute difference between two successive values of $\mathcal{J}$ are below a certain user-defined threshold. Moreover, despite the probabilistic context of latent variable models, the EM algorithm is completely deterministic, making it sensible to poor initialization. It is advised to test different starting values $\boldsymbol{\vartheta}^{(0)}$, keeping the one achieving the greatest likelihood.

Remark 2. Looking at the proof of Proposition 2.2, one can see that $\mathcal{J}$ only needs to be greater in $\boldsymbol{\vartheta}^{(t+1)}$ than in $\boldsymbol{\vartheta}^{(t)}$, but not necessarily fully maximized. This allows for the use of any numerical optimization (Nocedal and Wright 2006) scheme in place of the traditional M-step, when the latter cannot be solved exactly. In this case, the algorithm is called a generalized EM (GEM) and both algorithms are actually particular instances of a larger class called minorization-maximization algorithms (MM, Lange 2016).

Note that we present the detailed computations of the EM for Gaussian mixture models in Section 2.2.1.c. In the following section, we detail how the EM framework can be used when the E-step is not tractable.

2.1.3 Mean-field approximations: a variational EM algorithm

Implicitly, the EM algorithm assumes that the posterior distribution $p(\boldsymbol{\eta} \mid \boldsymbol{Y}; \boldsymbol{\vartheta})$ is tractable. Yet, this assumption is not verified for a large class of statistical models. For instance, a classical setting is when the observed-data likelihood of Equation (2.1), *i.e.* the normalization term of the posterior in Bayes theorem, is intractable. This is the case for the stochastic block model of Section 2.2.2.a which requires an exponentially growing number of term to compute, or the latent Dirichlet allocation introduced in Section 2.3.2.b involving an intractable integral over continuous latent variables.

Nevertheless, the ELBO and the EM philosophy can still be useful in this context. The main idea is to replace the intractable posterior by some variational approximation (Jaakkola and Jordan 2000; Wainwright and Jordan 2008). Doing so amounts to posit a family of distributions $\mathcal{Q}$, and to restrict the KL minimization problem in the E-step to this family:

$$\text{VE-step: } q^{(t+1)} = \arg\max_{q \in \mathcal{Q}} \mathcal{J}(q; \boldsymbol{\vartheta}^{(t)}) = \arg\min_{q \in \mathcal{Q}} \mathrm{KL}(q \,\|\, p(\cdot \mid \boldsymbol{Y}; \boldsymbol{\vartheta})). \tag{VE-step}$$

In this section, we focus on one of the most popular approximation for the posterior, the

so-called *mean-field* approximation, which assumes:

$$\mathcal{Q} := \{q \ : \ q(\boldsymbol{\eta}) = \prod_h q_h(\boldsymbol{\eta}_h)\}. \tag{2.4}$$

This hypothesis may seem quite reductive at first, but as Blei et al. (2017, p. 8) points out:

> *We emphasize that the variational family is not a model of the observed data [...]. Instead, it is the ELBO, and the corresponding KL minimization problem, that connects the fitted variational density to the data and model.*

Moreover, no functional form is assumed on the individual factor distributions q_h. The main advantage of such approximation is that the optimization problem in the VE-step leads to the coordinate ascent variational inference algorithm (CAVI, Bishop 2006; Blei et al. 2017), which optimizes $\mathcal{J}$ with respect to q_h while considering the other latent distributions fixed. The general form of the updates is given in the following:

Proposition 2.3 (CAVI)**.** *Denoting by $\boldsymbol{\eta}_{-h}$ the vector $\boldsymbol{\eta}$ without its coordinate $\boldsymbol{\eta}_h$, and considering $\boldsymbol{\vartheta}$ and $q^\star_{-h}$ known and fixed, the ELBO is maximized in q_h by:*

$$q^\star_h(\boldsymbol{\eta}_h) \propto \exp\left\{\mathbb{E}_{-h}\left[\log p(\boldsymbol{\eta}_h, \boldsymbol{\eta}_{-h}, \boldsymbol{Y} \mid \boldsymbol{\vartheta})\right]\right\}, \tag{2.5}$$

where the expectation is taken over $\boldsymbol{\eta}_{-h} \sim q^\star_{-h}$.

Proof. Using the law of iterated expectations, one can write the ELBO as a function of q_h,

$$\mathcal{J}(q_h) = \mathbb{E}_h\left[\mathbb{E}_{-h}\left[\log p(\boldsymbol{\eta}_h, \boldsymbol{\eta}_{-h}, \boldsymbol{Y} \mid \boldsymbol{\vartheta})\right] - \log q_h\right] + \text{const}\,,$$

where we put constant terms w.r.t q_h in the constant and used the fact that $\mathrm{H}(q) = -\,\mathbb{E}_h\left[\log q_h\right] + \mathrm{H}(q^\star_{-h})$ for the mean-field family. Now, the right hand side can be written as the negative KL divergence between q_h and the distribution defined in Equation (2.5), plus a constant. Hence, maximizing the left-hand side w.r.t. q_h is equivalent to minimize this KL, which is done by setting $q^\star_h$ as in Equation (2.5). □

Thus, the CAVI acts like a fixed-point algorithm, each update maximizing $\mathcal{J}(q)$ sequentially. While the form of the updates in Proposition 2.3 may seem a bit convoluted, the rest of this chapter illustrates how it can be used rather easily with concrete examples. Indeed, even though no assumption is made about q_h, the optimum often happens to take the form of a known parametric distribution, with so-called *variational parameters* depending on $q^\star_{-h}$, $\boldsymbol{\vartheta}$ and $\boldsymbol{Y}$. The normalizing constant can then be inferred easily from the specific parametric family of distribution at hand. Notably, this is the case in all the models considered in this thesis. In practical implementations, the ascent algorithm stops when the marginal gain of a full pass over all q_h is below a certain threshold, or when a maximum number of iterations is reached.

Remark 3. The objective in the M-step remains unchanged, except that the conditional expectation is taken with respect to an approximation of the prior, hence the use of VEM terminology. The resulting sequence $\{\boldsymbol{\vartheta}^{(t)}\}$ is no longer guaranteed to increase the likelihood at each step, since the bound in not tight anymore in the VE-step. However, the ELBO sequence is still monotonically increasing and may be used as a surrogate for the intractable likelihood. Recently, the concept of *tempered* posteriors was used to derive concentration rates of variational Bayes approximations in latent variables models (Yang et al. 2020).

Variational methods have known successful developments over the last two decades, with extensions and applications from black box inference (Ranganath et al. 2014) to deep generative models such as variational autoencoders (Kingma and Welling 2019). Moreover, while out of the scope of this thesis, we emphasize that other exact or approximate inference approaches exist. In a Bayesian context, the well-known Markov chain Monte Carlo methods (MCMC, Robert and Casella 2013) aim at sampling from the true posterior. Their interest lie in the fact that they often come with asymptotic theoretical guarantees about convergence. While variational methods do not enjoy such properties, they tend to be faster than MCMC, efficiently scaling up to large data sets. In the case of intractable likelihoods, another interesting growing line of work is the approximate Bayesian computation methods, which aim at sampling from an approximate posterior distribution, building on refined versions of a rejection algorithm (see *e.g.* Marin et al. 2012)

2.2 Model-based clustering with discrete latent variable models

As explained in Chapter 1, when one seeks to perform clustering, an implicit assumption is made about the existence of an unobserved partition of the data. Latent variable models are perfectly adequate for this scenario, and provide a principled approach to do model-based clustering, where each individual i is assigned to a discrete latent variable $\boldsymbol{z}_i$ representing its cluster assignment. Canonically, these latent variables are represented as binary vectors $\boldsymbol{z}_i \in \{0,1\}^K$ where $\boldsymbol{z}_{ik} = 1$ if individual i belongs to cluster k. The unknown partition being fully characterized by the discrete latent variables, clustering is then cast as an inference problem over the posterior of $\boldsymbol{Z}$. Now, discrete latent variable models (DLVMs) assume that observations provided in $\boldsymbol{Y}$ are drawn from a two-step process: first, all the cluster indicator vectors are sampled independently from a multinomial $\mathcal{M}_K(1, \boldsymbol{\pi})$. Then, the observations $\boldsymbol{Y}$ are generated, conditionally on $\boldsymbol{Z}$, with some conditional independence assumption on $\boldsymbol{Y} \mid \boldsymbol{Z}$:

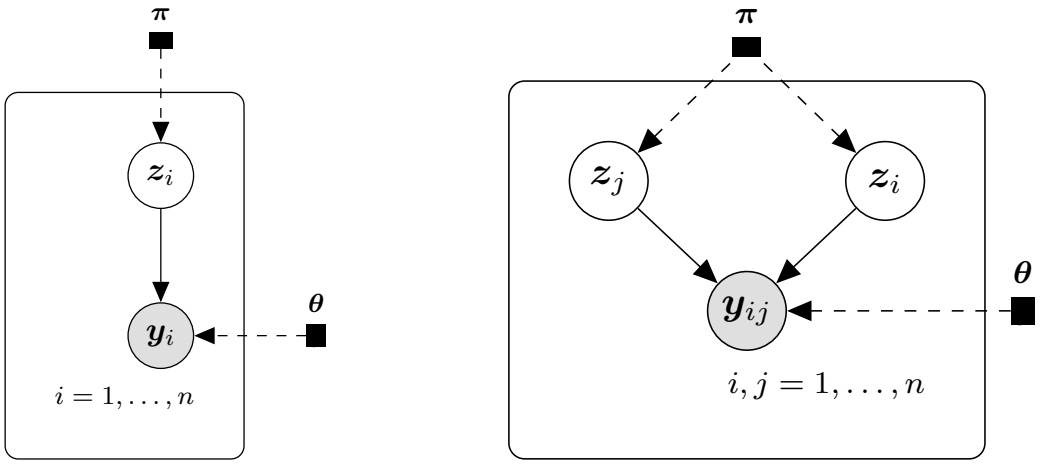
$$p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\vartheta}) = \prod_{\boldsymbol{z} \in \boldsymbol{Z}} p(\boldsymbol{z} \mid \boldsymbol{\pi}) \underbrace{\prod_{\boldsymbol{y} \in \boldsymbol{Y}} p(\boldsymbol{y} \mid \boldsymbol{Z}; \boldsymbol{\theta})}_{\text{factorized}}. \tag{2.6}$$

In this context, the complete-data likelihood above is often called the *classification* likelihood (Biernacki and Govaert 1997). With a slight abuse of notations compared to the previous section, we distinguish two types of parameters: $\boldsymbol{\theta}$ which is used to parameterize the conditional distribution of $\boldsymbol{Y} \mid \boldsymbol{Z}$, while $\boldsymbol{\pi}$ denotes the parameter of $p(\boldsymbol{Z} \mid \boldsymbol{\pi})$, which is a product of multinomials. This definition, relying on conditional independence given an unobserved random variable, closely relates on the Bayesian notion of *exchangeability* and De Finetti's theorem (Diaconis 1988; Kallenberg 2006). In the following, we detail how this specific framework encompasses finite mixture models, but also extends to more sophisticated structural dependencies such as latent block models, represented as graphical models in Figure 2.2.

2.2.1 Finite mixture models

2.2.1.a Definition

Finite mixture models (FMMs, McLachlan and Peel 2004) are the cornerstone of probabilistic clustering methods, and the most popular instance of DLVMs. Consider a family of K



(a) FMM **(b)** SBM

Figure 2.2: Graphical model representations of the main discrete latent variable models studied in this thesis: (a) Finite mixture models, (b) stochastic block models.

parametric densities $p(\cdot \mid \boldsymbol{\theta}_k)$ and positive weights $\boldsymbol{\pi} \in \Delta_K$. Then, finite mixture models assume that each observation in $\boldsymbol{Y} = \{\boldsymbol{y}_i\}$ is drawn *i.i.d.* from the convex combination:

$$p(\boldsymbol{y}_i \mid \boldsymbol{\pi}, \boldsymbol{\vartheta}) = \sum_{k=1}^{K} \pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k), \tag{FMM}$$

The left-hand side satisfies the definition of a density by construction, and the *mixture* terminology comes from its weighted average form. The generality of this definition allows to capture a large class of distributions, as well as enabling to encode real-world phenomenon such as multi-modality (see Figure 2.3). Naturally, specifying the type of distributions is a modeling choice depending on the data at hand. Gaussian mixture models constitute one of the most popular instance of FMMs when dealing with multivariate continuous data. In this context, the evidence writes as:

$$p(\boldsymbol{y}_i \mid \boldsymbol{\pi}, \boldsymbol{\vartheta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{m}_k, \boldsymbol{S}_k), \tag{GMM}$$

where $\mathcal{N}_p(\boldsymbol{y} \mid \boldsymbol{m}_k, \boldsymbol{S}_k)$ denotes the multivariate-Gaussian density with parameters $\boldsymbol{\theta}_k = (\boldsymbol{m}_k, \boldsymbol{S}_k) \in \mathbb{R}^p \times \mathcal{S}_p^{++}$.

For maximum likelihood inference to be a well-posed problem, and for the sake of consistent statistical analysis, we often ask for identifiability of the model (Casella and Berger 2002). That is, the observed-data likelihood needs to be an injective map with respect to $\boldsymbol{\vartheta}$.

$$p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) = p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}') \implies \boldsymbol{\vartheta} = \boldsymbol{\vartheta}'.$$

A crucial fact about FMMs is that parameters can only be identifiable (if they are) up to a permutations of the cluster indices. Indeed, take σ to be any permutation of $\{1, \ldots, K\}$, then it is clear from the definition of FMMs that $p(\boldsymbol{y} \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = p(\boldsymbol{y} \mid \sigma(\boldsymbol{\pi}), \sigma(\boldsymbol{\theta}))$. In the clustering literature, this is known as the *label switching* problem (Celeux 1998). Instinctively, this

lack of identifiability is not an issue since it stems from simple symmetries at the level of the distribution, however it may imply some computational and convergence issues, especially in Bayesian estimation with MCMC which can suffer from the $K!$ modes (Robert et al. 2010, p. 129).

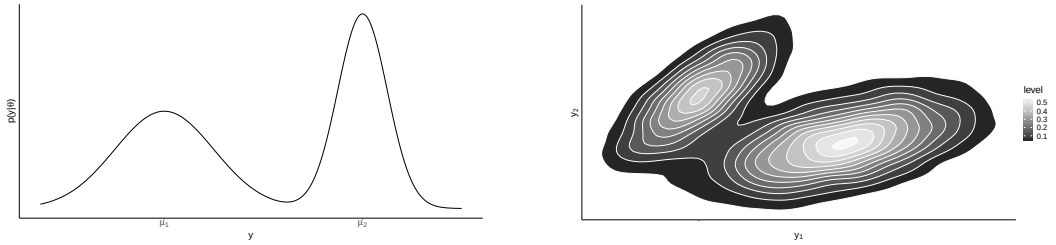


Figure 2.3: An exemple of a 2-components gaussian mixture model in 1-D (left) and in 2-D (right).

2.2.1.b Discrete latent variable models formulation

An equivalent formulation of mixture models is to assume that an observation comes from one of K different sub-populations, or clusters, modeled by $p(\cdot \mid \boldsymbol{\theta}_k)$, with probability π_k. In this scenario, one can view the likelihood of FMMs as the marginal outcome of the following generative process:

1. $\boldsymbol{z}_i \sim \mathcal{M}_K(1, \boldsymbol{\pi})$,
2. $\boldsymbol{y}_i \mid \{z_{ik} = 1\} \sim p(\cdot \mid \boldsymbol{\theta}_k)$,

where z_i represents the cluster assignment of $\boldsymbol{y}_i$. Then, the complete likelihood of $(\boldsymbol{y}_i, \boldsymbol{z}_i)$ writes as:

$$p(\boldsymbol{y}_i, \boldsymbol{z}_i \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = p(\boldsymbol{y}_i \mid \boldsymbol{z}_i; \boldsymbol{\theta}) p(\boldsymbol{z}_i \mid \boldsymbol{\pi}) = \prod_{k=1}^{K} \left[\pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k)\right]^{z_{ik}}. \tag{2.7}$$

One recovers the observed-data likelihood of FMMs when summing over the K possible values of $\boldsymbol{z}_i$. This second formulation draws a clear link to DLVMs. Indeed, writing the two distributions at the level of the whole dataset leaves:

$$p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k), \tag{2.8}$$

$$p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = \prod_{i=1}^{n} \prod_{k=1}^{K} \left[\pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k)\right]^{z_{ik}}. \tag{2.9}$$

This complete-data, or *classification* likelihood is then recognized as a particular form of Equation (2.6). While both formulations (2.8) and (2.9) assume marginal independence on the rows of $\boldsymbol{Y}$, the second one add a conditional independence assumption given $\boldsymbol{Z}$. Moreover, the tractable likelihood of FMMs along with the one-to-one correspondence between $\boldsymbol{z}_i$ and $\boldsymbol{y}_i$ gives an analytically tractable posterior distribution that factorizes. Indeed, using Bayes theorem, with the likelihood being tractable and factorized over observations, one

gets:

$$
\begin{aligned}
p(\boldsymbol{Z} \mid \boldsymbol{Y}; \boldsymbol{\pi}, \boldsymbol{\theta}) &= \frac{p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\pi}, \boldsymbol{\theta})}{p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta})}, \\
&= \frac{\prod_{i=1}^{n} p(\boldsymbol{y}_i \mid \boldsymbol{z}_i; \boldsymbol{\theta}) p(\boldsymbol{z}_i \mid \boldsymbol{\pi})}{\prod_{i=1}^{n} p(\boldsymbol{y}_i \mid \boldsymbol{\pi}, \boldsymbol{\theta})}, \\
&= \prod_{i=1}^{n} \frac{1}{\sum_{l=1}^{K} \pi_l p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_l)} \prod_{k=1}^{K} \left[\pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k)\right]^{z_{ik}}, \\
&= \prod_{i=1}^{n} \prod_{k=1}^{K} \left[\frac{\pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k}{\sum_{l=1}^{K} \pi_l p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_l)}\right]^{z_{ik}}.
\end{aligned}
$$

We then recognize the parametric form of a multinomial distribution $\mathcal{M}_K(\boldsymbol{z}_i \mid 1, \boldsymbol{\tau}_i)$, with

$$
\forall i, \forall k, \quad \tau_{ik} = \frac{\pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k)}{\sum_{l=1}^{K} \pi_l p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_l)}. \tag{2.10}
$$

After inference, a partition $\hat{\boldsymbol{Z}}$ can be inferred via a simple *maximum a posteriori* (MAP) estimate, that is $\hat{\boldsymbol{z}}_{ik} = 1$ with $k = \arg\max_l \tau_{il}$.

2.2.1.c The EM algorithm for mixture of Gaussians

In the particular case of Gaussian components, the MLE problem involves the following objective function:

$$
\log p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{m}_k, \boldsymbol{S}_k) \right)
$$

Although some approaches rely on numerical optimization, such as first or second order methods, inference for this model mostly uses the latent variable formulation as a FMM along with the EM algorithm. Indeed, the posterior for the E-step is tractable, thus one only needs to specify the updates wr.t. $(\boldsymbol{\pi}, \boldsymbol{\theta})$ in the M-step. In the latter, the lower bound is simply the expected logarithm of the complete-data likelihood, since the entropy term does not depend on the parameters:

$$
\begin{aligned}
\mathcal{J}(\boldsymbol{\pi}, \boldsymbol{\theta}) &= \mathbb{E}\left[\sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \left[\log(\pi_k) + \log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{m}_k, \boldsymbol{S}_k)\right] \right] + \text{const}, \\
&= -\frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_{ik} \left[\log(\pi_k) + \log |\boldsymbol{S}_k| + (\boldsymbol{y}_i - \boldsymbol{m}_k)^\top \boldsymbol{S}_k^{-1} (\boldsymbol{y}_i - \boldsymbol{m}_k) \right] + \text{const}.
\end{aligned}
$$

Where we used $\mathbb{E}[z_{ik}] = \tau_{ik}$, under the posterior. Finally, the constrained optimization problem can be solved at iteration (t) using classical first order conditions on the Lagrangian,

leaving:

$$\forall k,\ \pi_k^{(t)} = \frac{\sum_{i=1}^n \tau_{ik}^{(t)}}{n},\ \boldsymbol{m}_k^{(t)} = \frac{\sum_{i=1}^n \tau_{ik}^{(t)} \boldsymbol{y}_i}{\sum_{i=1}^n \tau_{ik}^{(t)}},\ \boldsymbol{S}_k^{(t)} = \frac{\sum_{i=1}^n \tau_{ik}^{(t)} (\boldsymbol{y}_i - \boldsymbol{m}_k^{(t)})(\boldsymbol{y}_i - \boldsymbol{m}_k^{(t)})^\top}{\sum_{i=1}^n \tau_{ik}^{(t)}}. \tag{2.11}$$

The detailed derivations can be found *e.g.* in McLachlan and Peel (2004). The simplicity of these updates outlines the computational interest of introducing latent variables. Indeed, the expected logarithm of the complete-data likelihood leaves a far simpler expression to differentiate than the logarithm of Equation (2.8). This is particularly the case with mixtures of exponential families.

The convergence of the likelihood to some saddle point is guaranteed by Proposition 2.2, however the convergence of the parameters is still an open problem even in simple settings* (Jin et al. 2016). Still, the EM is widely used in practice and, in this thesis, we adopt a clustering point-of-view with a focus on recovering the true partition $\boldsymbol{Z}$, rather than a density estimation one where the aim is to recover the true parameter $\boldsymbol{\vartheta}^\star$.

2.2.1.d Two variants around the EM: CEM and SEM

In the specific case of mixture models, several works have proposed variations of the EM algorithm. Celeux and Govaert (1992) introduced the classification EM (CEM) algorithm. The latter gets its name from the change of the objective function, which is now the classification likelihood in Equation (2.9). It introduces a supplementary C-step between the E and M-steps, where a partition $\hat{\boldsymbol{Z}}^{(t)}$ is computed through its current MAP estimate given by $\boldsymbol{\tau}^{(t)}$. Then, the conditional expectation in the M-step is replaced by the classification log-likelihood $\log p(\boldsymbol{Y}, \hat{\boldsymbol{Z}}^{(t)} \mid \boldsymbol{\pi}, \boldsymbol{\theta})$. Computationally, it simply amounts to replace the $\tau_{ik}^{(t)}$ by $\hat{\boldsymbol{z}}_{ik}^{(t)}$ in the M-step updates above, and is given as pseudo-code in Algorithm 1. It is a somewhat *hard*-clustering estimate of the conditional expectation, putting all the weight to the maximum of the posterior, as opposed to the *soft*-clustering version of the regular EM, weighting the contribution of each data point to each cluster estimates via its posterior membership. In practice, this version converges much faster than a regular EM, which comes at the cost of an inconsistent and biased estimation of the parameters.

Another modification was introduced in Celeux (1985) to deal with the sensibility of EM to its initialization $\boldsymbol{\vartheta}^{(0)}$. Indeed, the deterministic aspect of EM makes it more likely to converge to poor local maxima of the likelihood. Akin to CEM, the stochastic EM (SEM) is a stochastic version of the latter, introducing randomness in the procedure by drawing a partition $\hat{\boldsymbol{Z}}^{(t)}$ at each step, sampling from the posterior computed in the E-step. In this scenario, the likelihood is not guaranteed to monotonically increase and the sequence $\{\boldsymbol{\vartheta}^{(t)}\}_t$ is a Markov chain. While point-wise convergence is no longer achievable, the authors state a result about the convergence of its distribution under general assumptions (we refer to Celeux et al. (1995) for a general discussion about stochastic versions of EM).

*In fact, one can design simple settings where maximum likelihood estimation is inconsistent (see *e.g.* Alquier and Ridgway 2020, Section 7.8 for a 2-components unidimensional GMM exemple)

Algorithm 1: Pseudo code of the EM and CEM algorithms for finite mixture models. The E-step and C-step are common and the M-step differ.

Data: $\boldsymbol{Y}$
Result: Parameter $\hat{\boldsymbol{\vartheta}}$ and clustering $\boldsymbol{Z}$
Input: K, a tolerance ϵ and a maximum of iterations T

`// Initialization`
Initialize $(\boldsymbol{\pi}, \boldsymbol{\theta})$
$L^{(new)} \leftarrow p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta})$

`// Optimization`
for $t \leftarrow 1$ **to** T **do**
  $L^{(old)} \leftarrow L^{(new)}$
  `// E-step`
  Update $\boldsymbol{\tau}_i, \forall i$ with Equation (2.10)
  `// Optional C-step before M-step`
  Set $\tau_{ik} = 1$ for $k = \arg\max_l \tau_{il}$ and 0 otherwise
  `// M-step`
  Update $(\boldsymbol{\pi}, \boldsymbol{\theta})$, *e.g.* with Equation (2.11) for GMM
  `// Compute the likelihood`
  $L^{(new)} \leftarrow p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta})$
  if $|(L^{(new)} - L^{(old)}) \,/\, L^{(new)}| < \epsilon$ **then Break**;
end
Return $\hat{\boldsymbol{\vartheta}} \leftarrow (\boldsymbol{\pi}, \boldsymbol{\theta})$ *and* $\boldsymbol{Z}$ *such that* $z_{ik} = 1$ *for* $k = \arg\max_l \tau_{il}$

2.2.2 Latent block modeling

2.2.2.a The stochastic block model and the problem of graph-clustering

The stochastic block model (SBM, Wang and Wong 1987; Nowicki and Snijders 2001), is a statistical model for random graphs. As discussed in Chapter 1, a network or a graph is commonly observed through its adjacency matrix $\boldsymbol{Y}$ which is an $n \times n$ matrix with entries y_{ij} encoding the presence or absence of an edge. In the context of graph clustering, the individuals we seek to cluster are the vertices, while the observations are the edges $\{y_{ij}\}_{i,j \in \mathcal{I}}$. For the sake of simplicity, we only consider directed graphs without self loops, hence restricting $\mathcal{I} = \{(i,j),\ 1 \leq i \neq j \leq n\}$, but note that generalizations of the model are possible to deal with undirected graphs and self-loops (*e.g.* Peixoto 2012). The DLVM formulation of the SBM is as follows: first, each vertex is assigned to a discrete variable $\boldsymbol{z}_i$ independently. Then, the model assumes that the edge distribution between two vertices i and j only depends on their cluster assignments:

$$\begin{aligned} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) &= \prod_{i=1}^{n} \mathcal{M}_K(\boldsymbol{z}_i \mid 1, \boldsymbol{\pi}), \\ p(\boldsymbol{Y} \mid \boldsymbol{Z}; \boldsymbol{\theta}) &= \prod_{i \neq j}^{n} p(y_{ij} \mid \boldsymbol{z}_i, \boldsymbol{z}_j; \boldsymbol{\theta}) = \prod_{i \neq j}^{n} \prod_{k,l=1}^{K} p(y_{ij} \mid \boldsymbol{\theta}_{kl})^{z_{ik} z_{jl}}. \end{aligned} \tag{SBM}$$

The *block* terminology stems from the assumption that edges with extremities in the same pair of clusters (k, l) are i.i.d., hence forming homogeneous block in the adjacency matrix $\boldsymbol{Y}$. As in finite mixture models, the specification of the block distributions is a modeling choice depending on the data at hand. While the original model deals with binary graphs, edges being parameterized by Bernoulli distribution $\mathcal{B}(\theta_{kl})$, its general formulation allows for extensions dealing with weighted edges (Mariadassou et al. 2010) or overlapping partitions (Latouche et al. 2011). Recent works have also tackled the problem of dynamic networks evolving in time (Zreik et al. 2016), and textual edges which can be found in social, e-mail or scientific co-authorship networks (Bouveyron et al. 2018).

However, the SBM does not exactly fit the definition of finite mixture models, since the cluster latent variables are no longer independent a posteriori. Indeed, the intricate dependencies between the cluster latent variables prevent the posterior to be factorized over the nodes. This is directly connected to the fact that the partition is over the n nodes, while the observations consists in the $n^2 > n$ edges, thus the one-to-one correspondence between $\boldsymbol{z}$ and $\boldsymbol{y}$ is no longer applicable. Matias and Robin (2014) discuss this fact using the moralized graphical model (Bishop 2006, p. 392) of SBM in Figure 2.2b, which reveals the posterior dependencies that do not arise in finite mixture models. Another way of seeing the difficulty is that the observed data likelihood:

$$p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{\boldsymbol{z}_1, \dots, \boldsymbol{z}_n} \prod_{i \neq j}^{n} p(y_{ij}, \boldsymbol{z}_i, \boldsymbol{z}_j \mid \boldsymbol{\pi}, \boldsymbol{\theta}),$$

now involves K^n terms and does not simplify as a product over the observations like it does in FMMs. This exponentially growing number of terms implies that the marginalization is rapidly not a reasonable option, even in the case of a small sample setting. For this reason, several approximate inference procedure have been proposed for the estimation of SBM.

2.2.2.b A variational EM algorithm for the binary SBM

When dealing with binary edges $y_{ij} \in \{0, 1\}$, the distribution in block (k, l) is parameterized by a Bernoulli distribution:

$$p(\boldsymbol{Y} \mid \boldsymbol{Z}; \boldsymbol{\theta}) = \prod_{i \neq j}^{n} \prod_{k,l=1}^{K} \mathcal{B}(y_{ij} \mid \theta_{kl})^{z_{ik} z_{jl}}.$$

In this situation, the whole set of parameters $\boldsymbol{\theta} = \{\theta_{kl}\}_{k,l}$ may be viewed as a $K \times K$ matrix containing the probabilities of connection between two blocks k and l: the diagonal terms $(\theta_{kk})_{k=1,\dots,K}$ controlling the intra-cluster connectivity, while the off-diagonal terms control the inter-cluster propension to connect. Thus, the SBM model provides a flexible framework to analyze and recover traditional *assortative* community networks formed of well-separated clusters, as well as more *disassortative* structures.

Since the posterior distribution over the latent variable is not tractable, one can still resort to a variational approximation of the latter. Daudin et al. (2008) proposed a variational EM algorithm with q constrained to be in the mean-field family:

$$q(\boldsymbol{Z}) = \prod_{i=1}^{n} q_i(\boldsymbol{z}_i).$$

Then, the CAVI algorithm is employed to solve the VE-step by optimizing q_i sequentially.

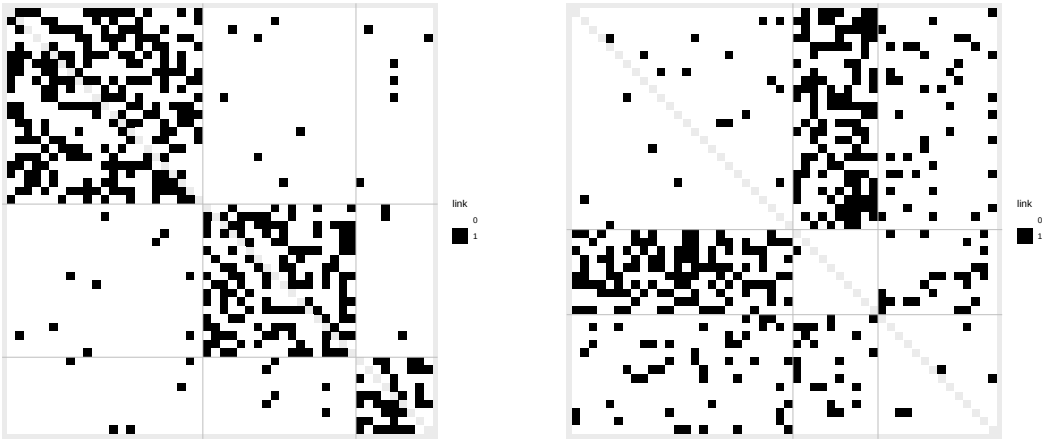


Figure 2.4: Adjacency matrices of simulated directed binary SBM with $K = 3$ clusters and proportion parameters $(0.5, 0.25, 0.25)$ with (Left) a strong affiliation structure: $\theta_{kk} = 0.4$, $\theta_{kl} = 0.025, \forall k \neq l$ and (Right) a strong disassortative structure: $\theta_{kk} = 0.025$, $\theta_{kl} = 0.15, \forall k \neq l$ and $\theta_{12} = \theta_{21} = 0.5$.

As discussed in Section 2.1.3, the optimal $q_i^\star$ updates of Proposition 2.3 happens to take a known parametric form here, as a multinomial with parameter $\boldsymbol{\tau}_i$ (Daudin et al. 2008, Proposition 5):

$$\forall i,\ q_i^\star = \mathcal{M}_K(1, \boldsymbol{\tau}_i) \text{ with: } \forall k, \tau_{ik} \propto \pi_k \prod_{j \neq i}^{n} \prod_{l=1}^{K} \mathcal{B}(y_{ij} \mid \theta_{kl})^{\tau_{jl}}. \tag{2.12}$$

Concerning the M-step, the expected complete-data log-likelihood under q is given by:

$$\mathcal{J}(\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_{ik} \log(\pi_k) + \prod_{i \neq j}^{n} \prod_{k,l=1}^{K} \tau_{ik}\tau_{jl} \left[y_{ij} \log(\theta_{kl}) + (1 - y_{ij}) \log(1 - \theta_{kl}) \right].$$

Including the constraints $\boldsymbol{\pi} \in \Delta_K$ and $\theta_{kl} \in [0, 1]$, the latter is maximized in π_k and θ_{kl}, by:

$$\hat{\pi}_k^{(t)} = \frac{\sum_{i=1}^{n} \tau_{ik}^{(t)}}{n}, \qquad \hat{\theta}_{kl}^{(t)} = \frac{\sum_{i \neq j}^{n} \tau_{ik}^{(t)} \tau_{jl}^{(t)} y_{ij}}{\sum_{i \neq j}^{n} \tau_{ik}^{(t)} \tau_{jl}^{(t)}}. \tag{2.13}$$

Celisse et al. (2012) and Bickel et al. (2013) provided theoretical guarantees about identifiability of the SBM parameters as well as convergence of the variational estimates $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\pi}}$ to the (MLE) with this VEM algorithm under mild assumptions.

2.2.2.c Co-clustering with the LBM

The problem of co-clustering is to simultaneously partition the rows and columns of a $n \times p$ data matrix $\boldsymbol{Y}$. The row and column partitions are respectively denoted as $\boldsymbol{Z_r} = \{\boldsymbol{z}_i\}_i$ with K_r clusters, and $\boldsymbol{Z_c} = \{\boldsymbol{z}_j\}_j$, with K_c clusters. The latent block model (LBM, Govaert and Nadif 2013) denotes a general class of generative models for co-clustering which assume that observations $\{y_{ij}\}$ are conditionally independent given the bipartition. Denoting the latter

as $\boldsymbol{Z} = (\boldsymbol{Z_r}, \boldsymbol{Z_c})$, it is formulated in the DLVM framework as:

$$\begin{aligned} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) &= p(\boldsymbol{Z_r} \mid \boldsymbol{\pi_r}) p(\boldsymbol{Z_c} \mid \boldsymbol{\pi_c}) = \prod_{i=1}^{n} \mathcal{M}_{K_r}(\boldsymbol{z}_i \mid 1, \boldsymbol{\pi_r}) \prod_{j=1}^{p} \mathcal{M}_{K_c}(\boldsymbol{z}_j \mid 1, \boldsymbol{\pi_c}), \\ p(\boldsymbol{Y} \mid \boldsymbol{Z}; \boldsymbol{\theta}) &= \prod_{i=1}^{n} \prod_{j=1}^{p} p(y_{ij} \mid \boldsymbol{z}_i, \boldsymbol{z}_j; \boldsymbol{\theta}) = \prod_{i=1}^{n} \prod_{j=1}^{p} \prod_{k=1}^{K_r} \prod_{l=1}^{K_c} p(y_{ij} \mid \boldsymbol{\theta}_{kl})^{z_{ik} z_{jl}}. \end{aligned} \tag{LBM}$$

with $\boldsymbol{\pi} = (\boldsymbol{\pi_r}, \boldsymbol{\pi_c}) \in \Delta_{K_r} \times \Delta_{K_c}$. This model shares a deep connection with the SBM since the latter may be viewed as a constrained case of the former, with $n = p$ and $\boldsymbol{Z_r} = \boldsymbol{Z_c}$. In particular, it has been widely used in the analysis of contingency table (Govaert and Nadif 2010), continuous (Lomet 2012), functional (Bouveyron et al. 2017), ordinal (Corneli et al. 2020) and mixed-type data (Selosse et al. 2020).

The problem of co-clustering then becomes that of inferring the bipartition $\boldsymbol{Z}$. However, as in the SBM, the posterior does not factorize over the observation (Keribin et al. 2015) and the marginalization:

$$p(\boldsymbol{Y} \mid \boldsymbol{\theta}) = \sum_{\boldsymbol{Z_r}} \sum_{\boldsymbol{Z_c}} p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\theta}),$$

now involves ${K_r}^n \times {K_c}^p$ terms. Once again, one has to resort to approximate inference procedures for the estimation of the LBM. A variational EM was described in Govaert and Nadif (2008) in the case of contingency table, where the VE-step and M-step closely resembles the one for SBM. On the theoretical side, Mariadassou and Matias (2015) extend previous work about identifiability and convergence for LBM and SBM in the binary and weighted case. Note that recent literature around LBM empirically advocates for a mixture of MCMC and EM, the SEM-Gibbs, to avoid poor local maxima of the variational bound (Keribin et al. 2015).

The cases of graph clustering and co-clustering with latent block models are discussed in Chapter 5 of this Thesis. A hierarchical algorithm is proposed, relying on the integrated classification likelihood which we introduce in Section 2.5.2.

2.3 Matrix factorization and probabilistic dimension reduction

A universal empirical observation from Pearson's work (Pearson 1901) and Hotteling children's test results (Hotelling 1933), to the rise of the so-called *Big Data* era with humongous databases (Deng et al. 2009; Wang et al. 2018) is that the observed data often lie in a low-dimensional subspace of dimension $d < p$. This can be exploited for several objectives. For instance, it can be used to remove irrelevant or redundant information in the original data in order to gain more insight on the data structure. It is particularly useful in high-dimensional scenarios when it is hard to interpret the contribution of each variable. Moreover, it can also allow to tackle learning in these scenarios, accelerating and improving inference procedure using the low-dimensional space and reducing the number of parameters to learn. Finally, when $d \leq 3$, visualization of the projected data in their subspace proves to be very useful in understanding the underlying structure of the data. Some methods focus on one or more of these goals, and we refer to Yu (2006, chap. 4) for a comprehensive review of modern dimension reduction algorithms.

In the following, we review some classical geometrical and statistical approaches to di-

mension reduction, highlighting the links between them. We focus on the specific case of matrix factorization and linear probabilistic projection methods, which can be summarized in the diagram of Figure 2.5. They turn out to be all expressed in the following global framework:

$$(\boldsymbol{U}^\star, \boldsymbol{X}^\star) = \underset{(\boldsymbol{U},\boldsymbol{X})\in\mathcal{H}}{\arg\min}\, L(\boldsymbol{Y}, \boldsymbol{U}, \boldsymbol{X}), \tag{2.14}$$

where $\boldsymbol{X} \in \mathbb{R}^{n\times d}$ denotes the row-stacked matrix of low dimensional representations $\{\boldsymbol{x}_i\}$, and the columns of $\boldsymbol{U} \in \mathbb{R}^{p\times d}$ span the optimal subspace with respect to the optimization problem at hand. The function L corresponds to a *loss-function* in the machine learning terminology, which can be model-free or the negative observed-data likelihood of a given probabilistic model.

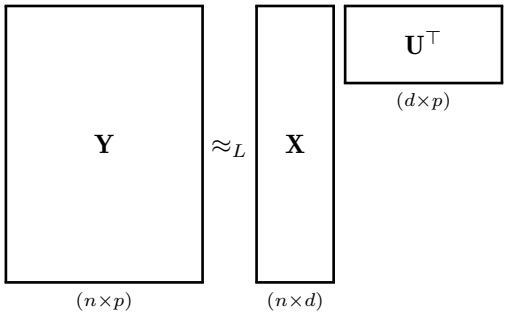


Figure 2.5: Matrix diagram for matrix factorization. The $\approx_L$ is used in a broad sense, specifying the dependency of the approximation to L. This is also known as dictionary learning in signal processing (Mairal et al. 2010).

2.3.1 Geometric approaches

2.3.1.a Principal component analysis

The work of Hotelling (1933) on children's test results led to the formulation of principal components analysis (PCA) which have become a fundamental tool in data analysis, ranging from psychology (Mulaik 2009), to micro-array data analysis (Ringnér 2008) and deep-learning (Chan et al. 2015). Even though it may feel that there are as many ways to introduce PCA as there are scientific disciplines using it, there are actually two main views of the latter. On the one hand, the statistical view searches for d linearly uncorrelated pseudo-variables from p the original ones, grouping highly linearly-correlated variables together. Considering that the data matrix $\boldsymbol{Y}$ is centered, every d-dimensional linear subspace maybe characterized by $\mathcal{S} = \{\boldsymbol{U}\boldsymbol{x}, \boldsymbol{x} \in \mathbb{R}^d\}$, where the columns of $\boldsymbol{U}$ form an orthonormal basis of $\mathcal{S}$, *i.e.* $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}_d$. Denoting $\boldsymbol{P}_\mathcal{S} = \boldsymbol{U}\boldsymbol{U}^\top$, the orthogonal projection matrix on $\mathcal{S}$, PCA searches to maximize the total variance, or inertia, of the projected point cloud:

$$\underset{\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_d}{\arg\max} \left\{ \|\boldsymbol{Y}\boldsymbol{P}_\mathcal{S}\|_F^2 := \mathrm{Tr}\left[\boldsymbol{P}_\mathcal{S}^\top\boldsymbol{Y}^\top\boldsymbol{Y}\boldsymbol{P}_\mathcal{S}\right] = \mathrm{Tr}\left[\boldsymbol{U}^\top\boldsymbol{S}_\boldsymbol{Y}\boldsymbol{U}\right] \right\}. \tag{PCA}$$

where $\boldsymbol{S}_\boldsymbol{Y} = \boldsymbol{Y}^\top\boldsymbol{Y} \,/\, n$ is the sample covariance matrix. This trace maximization problem is easily solved by a first order condition on the Lagrangian (Ghojogh et al. 2019), and the

optimal solution corresponds to the subspace spanned by the d eigenvectors corresponding to the d largest eigenvalues of the sample covariance matrix $\boldsymbol{S}_{\boldsymbol{Y}}$. This result is sometimes referred to as a consequence of the Courant-Fisher characterization (Kokiopoulou et al. 2011, p. 3). Thus, the matrix $\boldsymbol{U}$ contains the leading d eigenvectors which are often called the principal components, or *loadings*, in the PCA literature[†], while the low dimensional representations of each data points $\boldsymbol{X} = \boldsymbol{Y}\boldsymbol{U}$ are called the *scores*.

On the other hand, a geometrical interpretation of PCA searches for $\mathcal{S}$ minimizing the squared reconstruction error of $\boldsymbol{Y}\boldsymbol{P}_{\mathcal{S}}$ (Hastie et al. 2009, p. 535). Thanks to the property of orthogonal projections and Pythagoras' theorem, the total variance $\|\boldsymbol{Y}\|_F^2$ may be decomposed as the squared reconstruction error plus the variance of the projected points. The matrix factorization aspect of PCA described in Equation (2.14) clearly appear in this second formulation, looking for two low-rank matrices such that their product approximates the data matrix through the Frobenius norm:

$$\underset{\boldsymbol{X},\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_d}{\arg\min} \; L(\boldsymbol{Y},\boldsymbol{U},\boldsymbol{X}) := \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{U}^\top\|_F^2. \tag{2.15}$$

This formulation uses the Eckart-Young-Mirsky theorem (Eckart and Young 1936; Mirsky 1960) which states that the optimal $\boldsymbol{U}$ is given by the d leading right singular vectors of $\boldsymbol{Y}$, which corresponds to the same eigenvectors of $\boldsymbol{S}_{\boldsymbol{Y}}$. Moreover, the optimal $\boldsymbol{X}$ is given by $\boldsymbol{X} = \boldsymbol{Y}\boldsymbol{U}$, hence the strict equivalence between these two formulations. Several modifications of the PCA objectives were proposed to recover sparse solutions in $\boldsymbol{U}$, reducing the number of contributing original variables in the loadings when p is large. It was done either in the first formulation by setting some variable contributions to 0 (d'Aspremont et al. 2005), and in the second formulation using lasso type l_1-penalty (Zou et al. 2006). Another line of work has exploited the link between the singular value decomposition of the scatter matrix $\boldsymbol{Y}^\top\boldsymbol{Y}$ and the Gram matrix $\boldsymbol{Y}\boldsymbol{Y}^\top$ to replace the second one by a kernel matrix in order to unveil non-linear correlations (Mika et al. 1999).

2.3.1.b Non negative matrix factorization

Also known as positive matrix factorization (Paatero and Tapper 1994), non-negative matrix factorization (NMF) methods may be seen as a constrained PCA where the loadings and scores are constrained to be positives. Popularized by Lee and Seung (1999), which empirically showed their ability to decompose images in meaningful "parts", it quickly became an active field of research, in part due to the simplicity of its algorithms. In Lee and Seung (2001), the authors defined NMF as solving the particular optimization program:

$$(\boldsymbol{U}^\star, \boldsymbol{X}^\star) = \underset{\boldsymbol{U}\geq 0, \boldsymbol{X}\geq 0}{\arg\min} \; L(\boldsymbol{Y},\boldsymbol{U},\boldsymbol{X}), \tag{NMF}$$

[†]Depending on the scientific field, the *loadings* matrix $\boldsymbol{W}$ may also be introduced as the matrix $\boldsymbol{U}$ right multiplied, or scaled, by the diagonal matrix $\boldsymbol{L}^{1/2}$ containing the square-root of the corresponding eigenvalues: $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{L}^{1/2}$. This matrix contains the variance in its columns since $\|\boldsymbol{w}_{\cdot,h}\|^2 = \lambda_h$. The pPCA model in Section 2.3.2.a uses these definitions, note that this is just a matter of convention about how to distribute the variance between scores and principal components, the linear subspace defined by $\boldsymbol{U}$ is unchanged.

with two loss functions:

$$L_F(\boldsymbol{Y}, \boldsymbol{U}, \boldsymbol{X}) = \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{U}^\top\|_F^2 = \sum_{i=1}^n \sum_{j=1}^p \left(y_{ij} - \boldsymbol{u}_j^\top \boldsymbol{x}_i\right)^2,$$
$$L_{\mathrm{KL}}(\boldsymbol{Y}, \boldsymbol{U}, \boldsymbol{X}) = D(\boldsymbol{Y} \| \boldsymbol{X}\boldsymbol{U}^\top) = \sum_{i=1}^n \sum_{j=1}^p \left(y_{ij} \log \frac{y_{ij}}{\boldsymbol{u}_j^\top \boldsymbol{x}_i} + \boldsymbol{u}_j^\top \boldsymbol{x}_i - y_{ij}\right).$$

The first one is akin to PCA, the difference being the positivity constraint. The second may be seen as an un-normalized Kullback-Leibler divergence, *i.e.* when $\sum_{ij} y_{ij} = \sum_{ij} (\boldsymbol{u}_j^\top \boldsymbol{x}_i) = 1$ defines two discrete probabilistic distributions, it amounts to the KL between them. These functions are convex in $\boldsymbol{U}$ or $\boldsymbol{X}$ but non-jointly convex in $(\boldsymbol{U}, \boldsymbol{X})$. Instead of relying on gradient based methods, Lee and Seung (2001) proposed iterative updates that happens to be simple element wise multiplications:

$$x_{ih} \leftarrow x_{ih} \frac{(\boldsymbol{Y}\boldsymbol{U})_{ih}}{(\boldsymbol{X}\boldsymbol{U}^\top\boldsymbol{U})_{ih}} \qquad u_{jh} \leftarrow u_{jh} \frac{(\boldsymbol{Y}^\top\boldsymbol{X})_{jh}}{(\boldsymbol{U}\boldsymbol{X}^\top\boldsymbol{X})_{jh}}, \tag{NMF-F}$$
$$x_{ih} \leftarrow x_{ih} \frac{\sum_j u_{jh} y_{ij} \,/\, (\boldsymbol{U}\boldsymbol{X}^\top)_{jh}}{\sum_{j'} u_{j'h}} \qquad u_{jh} \leftarrow u_{jh} \frac{\sum_i x_{ih} y_{ij} \,/\, (\boldsymbol{U}\boldsymbol{X}^\top)_{jh}}{\sum_{i'} x_{i'h}}. \tag{NMF-KL}$$

Monotonic convergence of the objective under these updates can be proven by using similar techniques to those in Proposition 2.2. In fact, the proposed algorithms are MM algorithms (Lange 2016). Also note that, in both cases, if $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{U}^\top$, the multiplicative coefficients are equal to 1, thus showing that perfect reconstruction is a fixed point of the algorithm. In practice these updates are known to converge very fast, but Gonzalez and Zhang (2005) empirically showed that stationary points of the algorithms do not always correspond to local minima. Moreover, Donoho and Stodden (2004) showed that this decomposition is not unique, nuancing the universal aspect of $\boldsymbol{U}$ and moderating the "learning parts of objects" side of the story.

The objective have been used extensively to perform dimension reduction on count data. Indeed, the NMF constraints arise naturally in this context due to the positive nature of discrete data. Moreover, empirical arguments claim that NMF can be useful for clustering since it is supposed to naturally learn sparse representations. However, the discussion above shows that it is primarily designed for dimension reduction.

Link with probabilistic Latent Semantic indexing In the case of L_{KL}, Gaussier and Goutte (2005) showed that the NMF objective is equivalent to maximum likelihood estimation in a statistical model for count data: the probabilistic latent semantic indexing (pLSI, Hofmann 1999). The latter is a probabilistic formulation of a well-known method in text analysis called latent semantic analysis (LSA, Deerwester et al. 1990), which basically consists in performing PCA on $\boldsymbol{Y} \in \mathbb{N}^{n \times p}$. However, Ding et al. (2008) showed that even though the objective functions are the same for NMF and pLSI, the algorithms differ and do not yield the same solutions. In Section 2.3.2.b, we describe the latent Dirichlet allocation (Blei et al. 2003), a fully generative model for pLSI, drawing link with a probabilistic formulation of PCA.

As a side note, the recent use of neural networks in text analysis have known a tremendous success, especially *neural words embedding* which consists in finding low dimensional continuous representations of words. Recently, Levy and Goldberg (2014) showed the connections between a popular architecture, Word2Vec of Mikolov et al. (2013), and matrix factorization.

2.3.2 Latent variable models for dimension reduction

A key assumption in probabilistic models for dimension reduction is that each observation $\boldsymbol{y}_i$ can be linked to a latent random variable $\boldsymbol{x}_i$ lying in a subspace of dimension $d < p$. We focus on the popular case, akin to generalized linear models (Nelder and Wedderburn 1972), where this link is a combination of a linear transformation $\boldsymbol{U}$ on the latent space, and a possibly non-linear probabilistic *emission function* parameterized by this transformation (Bartholomew et al. 2011; Chiquet et al. 2018):

$$\begin{aligned} \boldsymbol{x}_i &\sim \mathcal{F}, \\ \boldsymbol{y}_i \mid \boldsymbol{x}_i &\sim p(\cdot \mid \boldsymbol{U}\boldsymbol{x}_i), \end{aligned}$$

where $\mathcal{F}$ is a distribution on a space $\mathbb{E} \subset \mathbb{R}^d$. In this context, linear latent variable models for dimension reduction may be viewed as a form of matrix factorization on the parameters, as well as in the sense of Equation (2.14), the negative likelihood acting as the loss function.

2.3.2.a Factor analysis and probabilistic PCA

A statistical model for principal component analysis was introduced simultaneously in Roweis (1998) and Tipping and Bishop (1999b), known as the probabilistic PCA (pPCA). The idea is to cast PCA as the following linear-Gaussian model:

$$\begin{aligned} \boldsymbol{x}_i &\sim \mathcal{N}_d(\boldsymbol{0}_d, \boldsymbol{I}_d), \\ \boldsymbol{y}_i &= \boldsymbol{m} + \boldsymbol{W}\boldsymbol{x}_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}_p(\boldsymbol{0}_p, \boldsymbol{\Psi}), \end{aligned} \tag{FA}$$

with no constraints on $\boldsymbol{W}$. Here, the emission function to the observed space is Gaussian $\boldsymbol{y}_i \mid \boldsymbol{x}_i \sim \mathcal{N}_p(\boldsymbol{W}\boldsymbol{x}_i + \boldsymbol{m}, \boldsymbol{\Psi})$, and the marginal distribution of an observation is also Gaussian $\boldsymbol{y}_i \sim \mathcal{N}_p(\boldsymbol{m}, \boldsymbol{S})$ with covariance $\boldsymbol{S} = \boldsymbol{W}\boldsymbol{W}^\top + \boldsymbol{\Psi}$. Hence, this model may be viewed as a constrained Gaussian model, with a covariance matrix factorized into a low rank product. When the noise variance $\boldsymbol{\Psi}$ is constrained to be diagonal, the coordinates of $\boldsymbol{y}_i = (y_{ij})_j$ become conditionally independent given the latent variable $\boldsymbol{x}_i$. Thus, the relevant correlations between variables are supposed to be captured in the latent space by $\boldsymbol{W}\boldsymbol{W}^\top$, while additive and uncorrelated errors are captured in $\boldsymbol{\Psi}$. This model is actually known as *factor analysis* (FA, Ghahramani and Hinton 1996), while the pPCA model makes the additional assumption of an isotropic noise covariance

$$\boldsymbol{\Psi} = \sigma^2 \boldsymbol{I}_p. \tag{pPCA}$$

Note that this factor analytic formulation distributes all the variance into the (scaled) loadings $\boldsymbol{W}$, while the scores are standardized Gaussian random variables. It contrasts from the chosen convention in Section 2.3.1.a, however an equivalent formulation with orthonormal loadings and variance in the latent space is also possible. Moreover, one might note that the matrix $\boldsymbol{W}$ is only identifiable up to a rotation of the latent space. Indeed, taking any $d \times d$ rotation matrix $\boldsymbol{R}$ such that $\boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{I}_d$, then setting $\tilde{\boldsymbol{W}} = \boldsymbol{W}\boldsymbol{R}$ leaves $\tilde{\boldsymbol{S}} = \tilde{\boldsymbol{W}}\tilde{\boldsymbol{W}}^\top + \sigma^2 \boldsymbol{I}_p = \boldsymbol{S}$.

Here, the model parameters are $\boldsymbol{\vartheta} = \{\boldsymbol{m}, \boldsymbol{W}, \sigma^2\}$ and can be estimated via maximum-likelihood. The posterior being tractable as a Gaussian distribution in this model, Roweis (1998) derived an EM algorithm, drawing link to standard PCA in the noiseless limit $\sigma^2 \to 0$. Detailed computations may be found in the article. Shortly, it gives an iterative algorithm to solve PCA: starting from a subspace $\boldsymbol{W}^{(0)}$, it alternates between projection in the current

subspace (E-step) and finding the subspace which minimizes the squared reconstruction error of the current projection (M-step). It is particularly interesting when $d << p$ since the particular form of the updates allows to avoid computing the sample covariance matrix $\boldsymbol{S}_{\boldsymbol{Y}}$, which is in $\mathcal{O}(np^2)$, achieving a complexity of $\mathcal{O}(npd)$ in the M-step. However, this procedure is iterative, while the covariance needs to be computed only once. In addition to the EM algorithm, Tipping and Bishop (1999b) gave a direct and analytical form of the MLE solutions in the pPCA model:

$$\begin{aligned} \boldsymbol{m}_{ML} &= \bar{y}, \\ \boldsymbol{W}_{ML} &= \boldsymbol{U}_{ML}(\boldsymbol{L}_{ML} - \sigma^2_{ML}\boldsymbol{I}_d)^{\frac{1}{2}}\boldsymbol{R}, \\ \sigma^2_{ML} &= \frac{1}{p-d}\sum_{h=d+1}^{p}\lambda_j, \end{aligned} \tag{2.16}$$

with $\boldsymbol{L}_{ML} = \text{diag}(\lambda_h)_{h=1,\dots,d}$ containing the top-d eigenvalues of $\boldsymbol{S}_{\boldsymbol{Y}}$, with the corresponding eigenvectors in $\boldsymbol{U}_{ML}$. The rotational ambiguity is characterized by $\boldsymbol{R}$, which could be set to any value, *e.g.* $\boldsymbol{R} = \boldsymbol{I}_d$. Another important result is that all stationary points of the likelihood have the form $\boldsymbol{W} = \boldsymbol{U}_d(\boldsymbol{L}_d - \sigma^2_d\boldsymbol{I}_d)^{\frac{1}{2}}\boldsymbol{R}$ where $(\boldsymbol{L}_d, \boldsymbol{U}_d)$ contains any combination of d eigenvalues of the sample covariance, with their corresponding eigenvectors. However, only the update in Equation (2.16) corresponds to a maximum of the likelihood[‡], the others being saddle points. This is a crucial point to have in mind when one resorts to the EM alternative rather than direct maximization since, as discussed in Section 2.1.2, there is a possibility to converge to one of these $p!/(p-d)! - d!$ stationary points.

Finally, the posterior expectation of $\boldsymbol{x}_i \mid \boldsymbol{y}_i$ at the optimum is an affine function of the observation:

$$\mathbb{E}[\boldsymbol{x}_i \mid \boldsymbol{y}_i] = (\boldsymbol{W}^\top\boldsymbol{W} + \sigma^2\boldsymbol{I}_d)^{-1}\boldsymbol{W}^\top(\boldsymbol{y}_i - \boldsymbol{m}) = \boldsymbol{R}^\top\boldsymbol{L}_{ML}^{-1}(\boldsymbol{L}_{ML} - \sigma^2_{ML}\boldsymbol{I}_d)^{1/2}\boldsymbol{U}_{ML}^\top(\boldsymbol{y}_i - \bar{y}).$$

This highlights the connection to standard PCA in the noiseless setting, when $\sigma^2_{ML} \to 0$, where the optimal subspace as well as the projections are the same.

This probabilistic formulation has allowed for a lot of extensions, from model selection criteria for d (Minka 2001; Bouveyron et al. 2011; Bouveyron et al. 2020b), to mixture modeling which is discussed in detail in Section 2.4. Sparse versions with parsimony inducing priors on $\boldsymbol{U}$ (Archambeau and Bach 2009), and globally sparse formulations performing variable selection and dimension reduction simultaneously (Mattei et al. 2016), have also been proposed.

2.3.2.b Latent Dirichlet Allocation or the Multinomial PCA

When dealing with discrete, count data, such as words in a document or read counts in a gene, the Gaussian hypothesis is no longer valid. In Buntine (2002), the author proposed a discrete analog of pPCA where the latent variables now represent a discrete probability distribution on $\{1, \dots, d\}$, *i.e.* $\boldsymbol{x}_i \in \Delta_d$. Dealing with multivariate count data, a natural distribution on the Simplex is the Dirichlet distribution:

$$\mathcal{D}_d(\boldsymbol{x}_i \mid \boldsymbol{\delta}) = \frac{1}{C(\boldsymbol{\delta})}\prod_{h=1}^{d}\boldsymbol{x}_{ih}^{\delta_h - 1}\mathbb{1}_{\Delta_d}(\boldsymbol{x}_i), \quad \text{with } \boldsymbol{\delta} = (\delta_1, \dots, \delta_d) \succcurlyeq 0.$$

[‡]Assuming there is no equality in the eigenvalues

Then, the probabilistic emission function to the observed space is assumed to be multinomial and the model writes as follow:

$$\begin{aligned} \boldsymbol{x}_i &\sim \mathcal{D}_d(\boldsymbol{\delta}), \\ \boldsymbol{y}_i \mid \boldsymbol{x}_i &\sim \mathcal{M}_p(c_i, \boldsymbol{U}\boldsymbol{x}_i), \end{aligned} \tag{MPCA}$$

where $c_i = \sum_{j=1}^{n} y_{ij}$ represents the total count of observation i. The columns of $\boldsymbol{U}$ contains d discrete probability distributions on $\{1, \dots, p\}$, called *topics* in the literature. The MPCA model may be thought of as a constrained multinomial model with a parameter $\boldsymbol{\vartheta}_i$ for each observation $\boldsymbol{y}_i$, where $\boldsymbol{\vartheta}_i = \boldsymbol{U}\boldsymbol{x}_i$ is factorized as a mixture of d global topics characterizing the whole data set, with document-dependent weights. Then, in an analogy to scores in PCA, an observation is represented by its mixture weights $\boldsymbol{x}_i$, regarded as coordinates in the latent topic space spanned by the columns of $\boldsymbol{U}$.

Here, $\boldsymbol{\delta}$ is viewed as an hyper-parameter and considered fixed for now, although estimation procedure exists. Thus, the parameter is simply $\boldsymbol{\vartheta} = \boldsymbol{U}$. Unfortunately, the observed-data likelihood is not tractable contrary to pPCA. Indeed, the integral

$$p(\boldsymbol{y}_i \mid \boldsymbol{U}) = \int_{\boldsymbol{x}_i} \frac{c_i!}{\prod_j y_{ij}!} \prod_{j=1}^{p} \left(\boldsymbol{u}_j^\top \boldsymbol{x}_i\right)^{y_{ij}} \mathcal{D}_d(\boldsymbol{x}_i \mid \boldsymbol{\delta}) \, \mathrm{d}\boldsymbol{x}_i$$

is analytically intractable, the Dirichlet-multinomial conjugacy being inapplicable due to the coupling between $\boldsymbol{x}_i$ and $\boldsymbol{U}$ (Dickey 1983). While the formulation of MPCA highlights its probabilistic dimension reduction aspect, the latter happens to be best known as the latent Dirichlet allocation (LDA, Blei et al. 2003). These two models emerged jointly and, even though their connection have been highlighted (*e.g.* Podosinnikova et al. 2015), the LDA formulation has been the most successful, forming the building block for many of the so-called *topic models* (Steyvers and Griffiths 2007). In fact, inference is mostly done in the LDA formulation with a variational EM algorithm which we detail here.

LINK WITH LATENT DIRICHLET ALLOCATION Blei et al. (2003) originally developed LDA as a generative model for text corpora, the observations $\boldsymbol{y}_i$ being documents represented as word counts with a vocabulary of size p. It uses an alternative representation of count data in this context, which models a document $\boldsymbol{y}_i$ as a *bag-of-words* $\boldsymbol{w}_i = \{\boldsymbol{w}_{il}, \, l = 1, \dots, c_i\}$, where token $\boldsymbol{w}_{il}$ is a binary vector of dimension p such that $w_{ilj} = 1$ if the l-th word in the document corresponds to the j-th word in the vocabulary. These representations are equivalent as long as the word/token order does not matter, since the count data representation $\boldsymbol{y}_i$ does not preserve order, and one can always form the latter from the bag-of-word as: $\boldsymbol{y}_i = \sum_{l=1}^{c_i} \boldsymbol{w}_{il}$. The LDA model introduces a supplementary discrete latent variable $\boldsymbol{t}_{il}$ at the level of a token which characterizes its topic assignment inside document i, the generative model is:

$$\begin{aligned} \boldsymbol{x}_i &\sim \mathcal{D}_d(\boldsymbol{\delta}), \\ \forall l &= 1, \dots, c_i, \\ &\quad \boldsymbol{t}_{il} \mid \boldsymbol{x}_i \sim \mathcal{M}_d(1, \boldsymbol{x}_i), \\ &\quad \boldsymbol{w}_{il} \mid t_{ilh} = 1 \sim \mathcal{M}_p(1, \boldsymbol{U}_{\cdot,h}), \end{aligned} \tag{LDA}$$

The equivalence between the two models appears when one marginalizes over the d possible values of $\boldsymbol{t}_{il}$ leaving the conditional distribution of a word in a document:

$$\boldsymbol{w}_{il} \mid \boldsymbol{x}_i \sim \mathcal{M}_p(1, \boldsymbol{U}\boldsymbol{x}_i),$$

which is the same as in model MPCA except for the number of repetitions. Moreover, it does not depend on the choice of the word l. Thus, all the words in the bag $\boldsymbol{w}_i$ are conditionally independent given $\boldsymbol{x}_i$, *and* identically distributed from the distribution above. Finally, we recover the conditional distribution of MPCA when creating $\boldsymbol{y}_i = \sum_l w_{il}$, the normalization constant of the multinomial being independent of the parameters, and simply accounting for the number of possible bag-of-words representations from $\boldsymbol{y}_i$. It follows that maximum likelihood inference in the two models are equivalent.

A VARIATIONAL EM ALGORITHM FOR LDA The likelihood of an observation $p(\boldsymbol{w}_i \mid \boldsymbol{U})$ is still intractable for the same reason as before, ruling out the possibility to compute the posterior $p(\boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{W}; \boldsymbol{U})$ exactly. Thus, Blei et al. (2003) proposed a variational EM relying on a mean-field approximation of the posterior:

$$q(\boldsymbol{X}, \boldsymbol{T}) = \prod_{i=1}^{n} q(\boldsymbol{x}_i) \prod_{i=1}^{n} \prod_{l=1}^{c_i} q(\boldsymbol{t}_{il}).$$

Considering $\boldsymbol{U}$ fixed in the VE-step, the CAVI updates are derived thank to Equation (2.5), dropping every fixed quantities inside the constants:

$$\begin{aligned}
\log q^\star(\boldsymbol{x}_i) &= \mathbb{E}_{\boldsymbol{T}, \boldsymbol{X}_{-i}} \left[\log p(\boldsymbol{W}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{U}) \right] + \text{const}\,, \\
&= \sum_{h=1}^{d} \left\{ \delta_h + \sum_{l=1}^{c_i} \mathbb{E}[t_{ilh}] - 1 \right\} \log(\boldsymbol{x}_{ih}) + \text{const}\ . \\
\log q^\star(\boldsymbol{t}_{il}) &= \mathbb{E}_{\boldsymbol{t}_{-(i,l)}, \boldsymbol{X}} \left[\log p(\boldsymbol{W}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{U}) \right] + \text{const}, \\
&= \sum_{h=1}^{d} \left\{ \mathbb{E}\left[\log(x_{ih}) \right] + \sum_{j=1}^{p} w_{ilj} \log(u_{jh}) \right\} t_{ilh} + \text{const}\ .
\end{aligned}$$

As already discussed in the previous sections, the optimal updates of CAVI often happen to have a known parametric form. This is the case here, where we recognize the functional form of a Dirichlet for $q^\star(\boldsymbol{x}_i)$ and a multinomial for $q^\star(\boldsymbol{t}_{il})$, *modulo* their normalizing constants. The quantity inside the brackets represents their respective parameters, hence:

$$\begin{aligned}
q^\star(\boldsymbol{x}_i) &= \mathcal{D}_d\left(\boldsymbol{x}_i \mid \boldsymbol{\gamma}_i\right) && \forall h, \gamma_{ih} = \delta_h + \textstyle\sum_{l=1}^{c_i} \mathbb{E}[t_{ilh}]. \\
q^\star(\boldsymbol{z}_{il}) &= \mathcal{M}_d\left(\boldsymbol{t}_{il} \mid 1, \boldsymbol{\phi}_{il}\right) && \forall h, \phi_{ilh} \propto \exp\left\{\mathbb{E}[\log(x_{ih})]\right\} \textstyle\prod_{j=1}^{p} u_{jh}^{w_{ilj}}.
\end{aligned} \tag{2.17}$$

Finally, the expectations involved are tractable here: $\mathbb{E}[t_{ilh}] = \phi_{ilh}$ and $\mathbb{E}[\log(x_{ih})] = \psi(\gamma_{ih}) - \psi(\sum_{h'} \gamma_{ih'})$, with $\psi(\cdot)$ the gamma function. As a side note, this is not the case in MPCA, where $\boldsymbol{T}$ is marginalized out and the expected complete-data log-likelihood involves expectations of the form $\mathbb{E}[\log(\boldsymbol{u}_j^\top \boldsymbol{x}_i)]$ that does not simplify. This explains why the LDA formulation is preferred for inference, and highlights the nature of $\boldsymbol{T}$ as a convenience latent variable.

The M-step is a constrained maximization problem over the elements of $\boldsymbol{U}$ with d con-

straints on the columns: $\forall h, \sum_j u_{jh} = 1$. Isolating terms of the expected complete-data log-likelihood *i.e.* the ELBO, depending on $\boldsymbol{U}$ and denoting constraints multipliers as λ_h, the Lagrangian can be written:

$$\begin{aligned}\mathcal{L}(\boldsymbol{U}, \lambda) &= \mathcal{J}(\boldsymbol{U}) + \sum_h \lambda_h \left(1 - \sum_j u_{jh}\right), \\ &= \sum_{i=1}^{n} \sum_{l=1}^{c_i} \sum_{h=1}^{d} \phi_{ilh} \sum_{j=1}^{p} w_{ilj} \log(u_{jh}) + \sum_h \lambda_h \left(1 - \sum_j u_{jh}\right).\end{aligned}$$

Setting its gradient w.r.t u_{jh} to 0 and using the constraints on columns leaves:

$$\forall (j,h), \quad u_{jh} \propto \sum_{i=1}^{n} \sum_{l=1}^{p} \phi_{ilh} w_{ilj}. \tag{2.18}$$

where the $\propto$ means that they are normalized such that $\forall h, \sum_j u_{jh} = 1$.

2.4 Integrating mixture modeling and dimension reduction

While the above models are very popular for both continuous and discrete data dimension reduction, they are not core-designed for clustering. Although it is common nowadays to perform dimension reduction before data analysis, it loses the principled approach of both model-based clustering and probabilistic dimension reduction. For continuous data, numerous analyses would perform PCA before fitting a clustering algorithm. In the same vein, there have been some empirical attempts to justify clustering with matrix factorization such as NMF (Xu et al. 2003) or latent Dirichlet allocation, but in practice many methods have been considered to post-process the topic proportions using standard clustering algorithms (Liu et al. 2016; Bui et al. 2017). On the other hand, model-based clustering for high-dimensional data suffers from a form of the so-called curse of dimensionality, mostly due to the exploding number of parameters to estimate when p is large. For this reason, a wealth of literature have been focusing on high-dimensional mixture modeling, building on the ideas of probabilistic dimension reduction. In the following, we develop some popular instances where these two philosophies have been integrated in a common finite mixture models framework: namely Gaussian and multinomial mixtures. We refer to Bouveyron et al. (2019, chap. 8) and McParland and Murphy (2019) for general and comprehensive surveys of this topic in Gaussian and non-Gaussian settings.

2.4.1 Parsimonious extensions to Gaussian mixture models

The number of free parameters of a Gaussian mixture model is

$$\gamma = \underbrace{K-1}_{\boldsymbol{\pi}} + \underbrace{Kp}_{\boldsymbol{m}} + \underbrace{K\frac{p(p+1)}{2}}_{\boldsymbol{S}} = \mathcal{O}(Kp^2). \tag{2.19}$$

For reasonable values of K and p in real world applications this number explodes, which is sometimes referred to as a form of *curse of dimensionality* (Bellman 1957, p. ix). For instance, when $K = 4$ and $p = 100$, it represents 20603 free parameters, which demands a huge counterpart observation-wise, or else leading to singular estimates for $\boldsymbol{S}_k$ and numerical issues. In addition, admitting that the data is in sufficient amount, the computation of the posterior probabilities $\tau_{ik} \propto \pi_k \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{m}_k, \boldsymbol{S}_k)$ from Equation (2.10) involves the evaluation

of a Gaussian density, and thus the inversion of $\boldsymbol{S}_k^{-1}$. This computational burden, inside an iterative algorithm like EM, is prohibitive.

2.4.1.a Constrained covariance structure

STANDARD CONSTRAINTS Most of the complexity in parameter estimation and computations for GMM comes from the covariance matrices $\boldsymbol{S}_k$. A classical approach is to constrain it to have a specific form. For instance, restricting it to be a diagonal matrix $\boldsymbol{S}_k = \operatorname{diag}(\sigma_{k1}^2, \dots, \sigma_{kp}^2)$ diminishes the complexity of free parameters from quadratic to linear with the dimension. Such approach is referred to as Diag-GMM in the literature. Going further, the covariance matrices may be restricted to be isotropic $\boldsymbol{S}_k = \sigma_k^2 \boldsymbol{I}_p$, inducing a spherical Gaussian ellipse. This constraint is sometimes called Sphe-GMM. Another way to reduce the model dimension is to impose homoscedasticity, *i.e.* the matrices are shared among clusters $\boldsymbol{S}_k = \boldsymbol{S}$. This approach is called Com-GMM and may be used in combinations with the constraints above. In the limit case imposing isotropic homoscedasticity $\boldsymbol{S}_k = \sigma^2 \boldsymbol{I}_p$ and equal group proportions $\pi_k = 1/K$, the CEM algorithm for GMM is similar to a K-means (Bishop 2006, p. 443).

SPECTRAL CONSTRAINTS Banfield and Raftery (1993) introduced an efficient framework for constraining covariance matrices, later generalized by Celeux and Govaert (1995), using an eigenvalue decomposition of $\boldsymbol{S}_k = \lambda_k \boldsymbol{D}_k \boldsymbol{\Delta}_k \boldsymbol{D}_k^\top$, with $\lambda_k = |\boldsymbol{S}_k|^{1/p}$, $\boldsymbol{D}_k$ is the matrix of eigenvectors and $\boldsymbol{\Delta}_k$ is a diagonal matrix with entries proportional to the eigenvalues. Several restrictions can be put on a combination of these quantities, recovering the traditional constraints and more, with a total of fourteen different sub-models implemented in the **mclust** R package (see *e.g.* Scrucca et al. 2016). For example, $\lambda_k = \sigma_k^2$, $\boldsymbol{D}_k = \boldsymbol{I}_p$, and $\boldsymbol{\Delta}_k = \boldsymbol{I}_p$ recovers the Sphe-GMM model. In addition, this flexible framework admit nice geometric interpretations of the constraints in term of orientations ($\boldsymbol{D}_k$), shape ($\boldsymbol{\Delta}_k$), and volume (λ_k) of the Gaussian density ellipsoids.

However, even though some sub-models may greatly reduce the number of parameters, the quantities at hand are $p \times p$ matrices, and there is no direct link to a latent factorization of the parameters into a product of low-rank matrices as discussed in the previous section.

2.4.1.b Subspace clustering models

Building on the generative model of factor analysis introduced in Section 2.3.2.a, a significant amount of work proposed to integrate mixture modeling and dimension reduction in a unified latent variable framework.

MIXTURE OF FACTOR ANALYZERS Ghahramani and Hinton (1996) proposed a straightforward extension of factor analysis models into mixtures of such. Aptly named mixture of factor analyzers (MFA), the generative model is:

$$\begin{aligned} \boldsymbol{x}_i &\sim \mathcal{N}_d(\mathbf{0}_d, \boldsymbol{I}_d), \\ \boldsymbol{z}_i &\sim \mathcal{M}_K(1, \boldsymbol{\pi}), \\ \boldsymbol{y}_i &= \boldsymbol{m}_k + \boldsymbol{W}_k \boldsymbol{x}_i + \epsilon_i, \quad \epsilon_i \mid \{z_{ik} = 1\} \sim \mathcal{N}_p(\mathbf{0}_p, \boldsymbol{\Psi}_k), \end{aligned} \tag{MFA}$$

with mixture proportions $\boldsymbol{\pi}$ and parameters $\boldsymbol{\theta} = \{\boldsymbol{W}_k, \boldsymbol{m}_k, \boldsymbol{\Psi}_k\}_k$. Tipping and Bishop (1999a) proposed the same model but with the additional restriction of pPCA $\boldsymbol{\Psi}_k = \sigma_k^2 \boldsymbol{I}_p$,

recovering a mixture of pPCA. Now, there are two types of latent variables, and integrating over $\boldsymbol{\eta}_i = (\boldsymbol{x}_i, \boldsymbol{z}_i)$ leaves the marginal distribution:

$$\boldsymbol{y}_i \sim \sum_k \pi_k \mathcal{N}_p(\boldsymbol{m}_k, \boldsymbol{W}_k \boldsymbol{W}_k^\top + \boldsymbol{\Psi}_k). \tag{2.20}$$

This underlines the interpretation of factor analysis as a particular Gaussian model with a constrained covariance matrix:

$$\boldsymbol{S}_k = \boldsymbol{W}_k \boldsymbol{W}_k^\top + \boldsymbol{\Psi}_k \tag{2.21}$$

Notice that the true number of free parameters in each $\boldsymbol{W}_k$ is $pd - \frac{d(d-1)}{2}$ because of its rotational indeterminacy (Bishop 2006, p. 577). Therefore, the complexity of estimation is directly linked to the dimension of the latent space:

$$\gamma = K - 1 + Kp + K\left(pd - \frac{d(d-1)}{2} + p\right).$$

In this model, one cannot derive analytical formulae for the (MLE), hence inference is done via an EM algorithm.

MIXTURE OF COMMON FACTOR ANALYZERS The MFA model demonstrates correct performances while still suffering in high-dimensional cases. Moreover, the latent scores $\boldsymbol{x}_i$ are now supposed to live in different subspaces spanned by $\boldsymbol{W}_k$, hence losing the easy visualization property of the traditional factor analysis. In order to further reduce the complexity of estimation, Yoshida et al. (2004) and Baek and McLachlan (2008) proposed another formulation of MFA where the mixture is not placed in the observed but in the latent space, with a common loading matrix $\boldsymbol{U}$ shared among clusters:

$$\begin{aligned} \boldsymbol{z}_i &\sim \mathcal{M}_K(1, \boldsymbol{\pi}), \\ \boldsymbol{x}_i \mid \{z_{ik} = 1\} &\sim \mathcal{N}_d(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \\ \boldsymbol{y}_i &= \boldsymbol{U}\boldsymbol{x}_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}_p(\boldsymbol{0}_p, \boldsymbol{\Psi}). \end{aligned} \tag{MCFA}$$

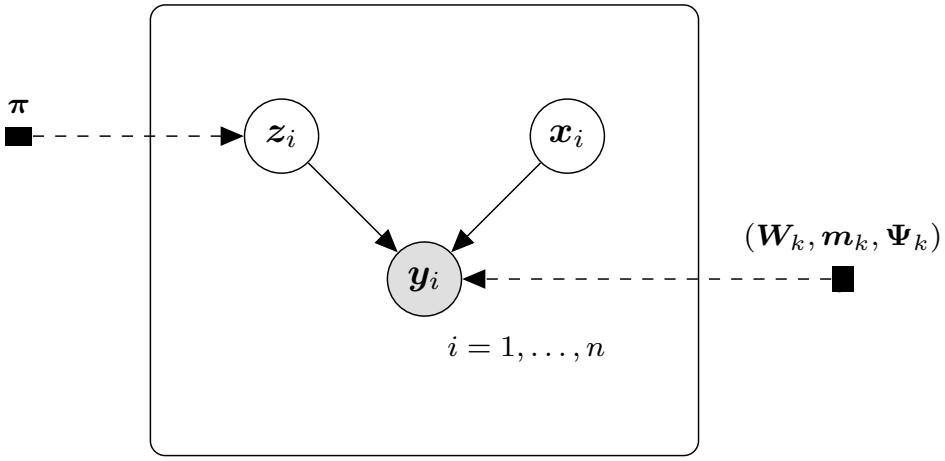
Here, the loading matrix is assumed to be column-orthonormal $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d$ such that all the variance is captured in the latent space by $\boldsymbol{\Sigma}_k$. Note that the noise covariance is common across clusters here. Yoshida et al. (2004) originally proposed the model with $\boldsymbol{\Psi} = \sigma^2 \boldsymbol{I}_p$ and $\boldsymbol{\Sigma}_k$ diagonal, and Baek and McLachlan (2008) generalized it, calling this specific case the MCUFSA for mixtures of common uncorrelated factors with spherical-error analyzers. In the general case, the marginal distribution of $\boldsymbol{y}_i$ is:

$$\boldsymbol{y}_i \sim \sum_k \pi_k \mathcal{N}_p(\boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{U}\boldsymbol{\Sigma}_k\boldsymbol{U}^\top + \boldsymbol{\Psi}), \tag{2.22}$$

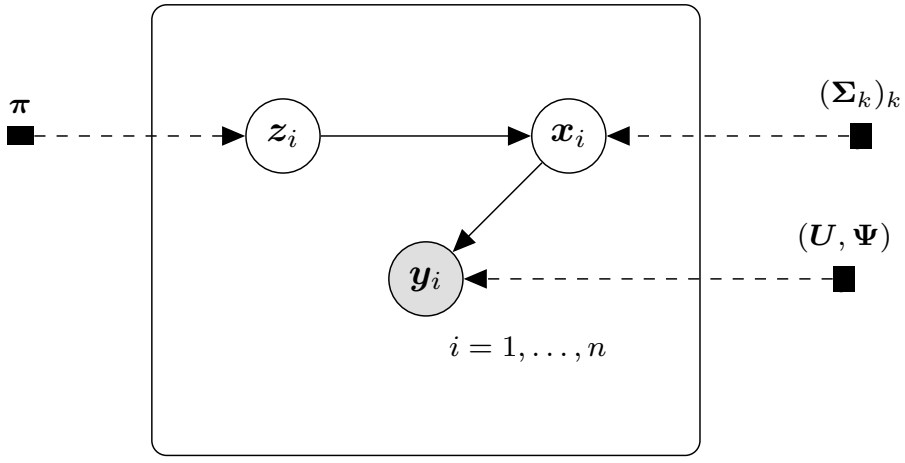
highlighting the fact that MCFA is a special case of MFA with constraints $\boldsymbol{m}_k = \boldsymbol{U}\boldsymbol{\mu}_k$ and $\boldsymbol{W}_k = \boldsymbol{U}\boldsymbol{\Sigma}_k^{1/2}\boldsymbol{R}$. The graphical models in Figure 2.6 emphasize their difference, but this slight modification has several implications: apart from reducing the number of parameter to

$$\gamma = K - 1 + Kd + K\frac{d(d+1)}{2} + pd - \frac{d(d+1)}{2} + p,$$

it makes the assumptions that the data being clustered live in the same subspace, whereas MFA assumes a different subspace per cluster. Thus, it can be seen at performing dimension reduction and then clustering in the latent space. This framework is convenient since we can further constrain the covariance $\boldsymbol{\Sigma}_k$ to have specific form as in the Diag-GMM and Sphe-GMM above. Moreover, this improves visualization capability of the method, since the clustering of the projected data may be apprehended in a common plot *e.g.* if $d = 2$. As for the MFA, inference is done with an EM algorithm working with $\eta = (\boldsymbol{x}_i, \boldsymbol{z}_i)_i$, and an implementation is available in the **EMMIXmfa** R package for both models (Rathnayake et al. 2019).



(a) MFA



(b) MCFA

Figure 2.6: Difference between the MFA and MCFA graphical models.

Montanari and Viroli (2010) proposed the same generative model except that the loading matrix is not constrained to be column-orthonormal, but rather the latent mixture parameters $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are constrained to respect $\mathbb{E}[\boldsymbol{x}_i] = \sum_k \pi_k \boldsymbol{\mu}_k = \boldsymbol{0}_d$ and $\mathbb{V}[\boldsymbol{x}_i] = \boldsymbol{I}_d$. Therefore, $(K-1)$ means and covariance matrices are sufficient to determine the latent mixture parameters. This model is called heteroscedastic factor mixture analysis (HFMA) and inference is

done with an EM algorithm. An implementation can be found in the **FactMixtAnalysis** R package (Viroli 2012).

PARSIMONIOUS GAUSSIAN MIXTURE MODELS McNicholas and Murphy (2008) proposed a generative framework called parsimonious Gaussian mixture models (PGMM) relying on the MFA decomposition $\boldsymbol{S}_k = \boldsymbol{W}_k\boldsymbol{W}_k^\top + \boldsymbol{\Psi}_k$. It consists in a family of 8 sub-models described by 3 letters indicating the presence (C) or absence (U) of constraints on homoscedastic loadings $\boldsymbol{W}$, homoscedastic noise covariance $\boldsymbol{\Psi}$, and isotropic noise covariance $\boldsymbol{\Psi}_k = \sigma_k^2\boldsymbol{I}_p$. The unconstrained model UUU is the standard MFA model, while the UUC model corresponds to the mixture of pPCA $\boldsymbol{S}_k = \boldsymbol{W}_k\boldsymbol{W}_k^\top + \sigma_k^2\boldsymbol{I}_p$. Although strongly connected, this framework does not exactly encompass the MCFA model since there is no distinction between the common loadings and the latent covariance matrices $\boldsymbol{\Sigma}_k$. In other words, the mixture is still at the level of observations, not at the level of scores. Thus, working with common loadings, the MCFA framework is more flexible as it allows to impose a common subspace while having different latent covariance matrices. An accelerated version of the EM, the AECM algorithm (Meng and Van Dyk 1997), is used and implemented in the **pgmm** R package (McNicholas et al. 2019).

MIXTURE OF HIGH-DIMENSIONAL GMM Finally, Bouveyron et al. (2007a) and Bouveyron et al. (2007b) extended the spectral constraints framework of Banfield and Raftery (1993): $\boldsymbol{S}_k = \boldsymbol{D}_k\boldsymbol{\Delta}_k\boldsymbol{D}_k^\top$. Allowing each cluster to have different *intrinsic* dimensions d_k, they posit a specific form for $\boldsymbol{\Delta}_k = \text{diag}(\alpha_{k1}, \ldots, \alpha_{kd_k}, \sigma_k^2, \ldots, \sigma_k^2)$. Spherical and homoscedasticity constraints may be put of the latent variance parameters (α_{kh}), on the orientations $\boldsymbol{D}_k$ and the intrinsic dimensions, leaving a family of 28 sub-models. An EM algorithm is used for inference and implemented in the **HDclassif** R package (Bergé et al. 2019), some models have closed form estimate for the M-step, while others require iterative maximization algorithm.

In Chapter 4, we build on the discriminative latent mixture of Bouveyron and Brunet (2012a), and propose a Bayesian mixture model for the clustering of high-dimensional data.

Model	Covariance structure	Free parameters γ	
Full	$\boldsymbol{S}_k$	$K-1+Kp+K\frac{p(p+1)}{2}$	20603
MFA	$\boldsymbol{W}_k\boldsymbol{W}_k^\top + \boldsymbol{\Psi}_k$	$K-1+Kp+Kd(p-\frac{(d-1)}{2})+Kp$	1603
VVE	$\lambda_k\boldsymbol{D}\boldsymbol{\Delta}_k\boldsymbol{D}^\top$	$K-1+Kp+K+(p+2K)\frac{(p-1)}{2}$	5753
CCU	$\boldsymbol{W}\boldsymbol{W}^\top + \boldsymbol{\Psi}$	$K-1+Kp+d(p-\frac{(d-1)}{2})+p$	800
MCFA	$\boldsymbol{U}\boldsymbol{\Sigma}_k\boldsymbol{U}^\top + \boldsymbol{\Psi}$	$K-1+Kd+K\frac{d(d+1)}{2}+d(p-\frac{d(d+1)}{2})+p$	415
HFMA	$\boldsymbol{W}\Omega_k\boldsymbol{W}^\top + \boldsymbol{\Psi}$	$(K-1)(1+d+\frac{d(d+1)}{2})+d(p-\frac{(d-1)}{2})+p$	427

Table 2.1: Non-exhaustive summary of global and local subspace clustering models, with their covariance structure along with their number of free parameters. The $\boldsymbol{U}$ matrix is column-orthonormal, the matrix $\boldsymbol{W}$ is only known up to a rotation, $\boldsymbol{\Psi}$ is diagonal, and the Ω_k matrices are constrained to respect $\sum_k \pi_k \mathbb{V}(\boldsymbol{x}_i \mid z_{ik} = 1) = \boldsymbol{I}_d$. The last column gives the number of free parameters for $p = 100$, $K = 4$ and $d = 3$.

2.4.2 Factorizing mixture parameters in discrete distributions

2.4.2.a Factorizing mixture of multinomial with the NMFEM algorithm

A popular FMM for the clustering of count data is the mixture of multinomial:

$$p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\theta}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \, \mathcal{M}_p(\boldsymbol{y}_i \mid c_i, \boldsymbol{\theta}_k), \tag{MoM}$$

with $\boldsymbol{\theta}_k \in \Delta_p$. However, when the dimension p is large, it requires the estimation of $K(p-1)$ parameters which can be a highly unstable, especially in skewed, zero-inflated real-world count data such as documents in a bag-of-words model, or RNA-seq data. In a document clustering context, Rigouste et al. (2007) proposed a detailed evaluation of the MoM model. Comparing an EM algorithm with a Gibbs sampler, they obtained comparable performances for both approaches, illustrating the difficulties of high-dimensional estimation.

As an alternative to this problem, Carel and Alquier (2017) proposed to factorize the mixture parameters $\boldsymbol{\theta}$ of the MoM model, which are positive and can be represented in a $K \times p$ matrix: $\boldsymbol{\theta} = [\boldsymbol{\theta}_1 \mid \ldots \mid \boldsymbol{\theta}_K]^\top$. The proposed factorization is:

$$\boldsymbol{\theta} = \boldsymbol{X}\boldsymbol{U}^\top \quad \text{with: } (\boldsymbol{X}, \boldsymbol{U}) \in \mathcal{H} := \{\boldsymbol{X} \in \mathbb{R}_+^{K \times d}, \boldsymbol{U} \in \mathbb{R}_+^{p \times d}, \textstyle\sum_{j=1}^{p} u_{j,h} = \sum_{h=1}^{d} x_{kh} = 1\}. \tag{2.23}$$

Thus, if $d < \min(K, p)$, the number of parameters greatly decreases. The limit case $d = K$ and $\boldsymbol{X} = I_K$ recovers the unconstrained model. The notations $\boldsymbol{X}$ and $\boldsymbol{U}$ were kept on purpose to underline the connection with Section 2.3. The difference being that the low-dimensional representation $\boldsymbol{x}_k$ is shared by all observations $\boldsymbol{y}_i$ belonging in the same cluster.

The E-step is common to any FMM and detailed in Equation (2.10) on page 24. For the M-step, there is no analytical solution for $\boldsymbol{X}$ and $\boldsymbol{U}$ under the constraints in $\mathcal{H}$. However, the expected complete-data log-likelihood can be rearranged, so that the maximization program in the $(t+1)$-th M-step is

$$\begin{aligned} \underset{(\boldsymbol{X},\boldsymbol{U})\in\mathcal{H}}{\arg\max}\, \mathcal{J}(\boldsymbol{X}, \boldsymbol{U}) &= \underset{(\boldsymbol{X},\boldsymbol{U})\in\mathcal{H}}{\arg\max} \sum_{k=1}^{K} \sum_{h=1}^{d} \left\{ \tilde{y}_{kj}^{(t)} \log\left(\boldsymbol{u}_j^\top \boldsymbol{x}_k\right) - \boldsymbol{u}_j^\top \boldsymbol{x}_k \right\} \quad \text{with: } \tilde{y}_{kj}^{(t)} = \textstyle\sum_i \tau_{ik}^{(t)} y_{ij}, \\ &= \underset{(\boldsymbol{X},\boldsymbol{U})\in\mathcal{H}}{\arg\min}\, L_{\mathrm{KL}}(\tilde{\boldsymbol{Y}}^{(t)}, \boldsymbol{U}\boldsymbol{X}). \end{aligned}$$

Thus, it can be solved via the multiplicative updates of Equation (NMF-KL), normalizing $\boldsymbol{X}^{(t+1)}$ and $\boldsymbol{U}^{(t+1)}$ at each pass of the updates to respect the simplex constraints. The result is an NMF-EM algorithm for mixture of multinomials, which is implemented in the eponymous **nmfem** R package (Carel 2017).

In Chapter 3, we introduce a Bayesian version of this model, putting a Dirichlet prior on $\boldsymbol{x}_k$. This model may be viewed as a mixture of MPCA and we derive useful properties of its classification likelihood, along with a variational version of the CEM algorithm for joint inference and clustering.

2.4.2.b Factorized mixtures of Poisson

The rise of next-generation sequencing technologies had a massive impact on genetics, especially with RNA-seq data which have known an important success this two last decades

(Wang et al. 2009). Distinct from the popular micro-array data, these methods provide an estimation of the expression level of a gene as positive, count data (reads count) instead of continuous ones (Auer and Doerge 2010). In this context, an observation i represents a gene, while the variable j represents a condition, *e.g.* a treatment. Grouping genes having the same expression profile in different conditions may be of interest for understanding the underlying biological processes in which they are involved. A particularity of these data is that each condition j can be repeated r_j times, as a *replicate* of the experiment. Then, the observations are $\{y_{ijl}\}_{ijl}$ where y_{ijl} denotes the differential gene expression of gene i, in replicate l of condition j. In term of the data matrix $\boldsymbol{Y}$, this amounts to the concatenation of p column blocks of size r_j, for a total of pr columns with $r = \sum_j r_j$.

Rau et al. (2011) proposed a model-based approach for RNA-seq data clustering, considering a Poisson mixture model with a local independence hypothesis that every replicates in every conditions are independent knowing their cluster assignment $\boldsymbol{z}_i$ (McCutcheon 1987). Thus, it corresponds to the following marginal distribution:

$$p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{\mu}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k p(\boldsymbol{y}_i \mid \boldsymbol{\mu}_{ki}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \prod_{j=1}^{p} \prod_{l=1}^{r_j} \mathcal{P}(y_{ijl} \mid \mu_{kijl}). \tag{2.24}$$

Naturally, with only npr observations for $Knpr$ parameters in the saturated case, the model needs some constraints to be identifiable. Rau et al. (2011) proposed the two following factorizations:

$$\mu_{kijl} = w_i \lambda_{kj}, \qquad \text{with: } \textstyle\sum_j r_j \lambda_{kj} = 1, \tag{PMM-I}$$

$$\mu_{kijl} = w_i s_{jl} \lambda_{kj}, \qquad \text{with: } \textstyle\sum_j \sum_l s_{jl} \lambda_{kj} = 1. \tag{PMM-II}$$

The first parametrization is independent of replicate l, while the second takes into account the variability of the means across replicates due to experimental conditions with s_{jl}. The latter is considered to be a fixed, known constant in the model, estimated on the data before inference. Then, an EM algorithm is derived, with $\hat{x}_i = c_i$ for both case, and $\lambda_{kj}^{(t+1)} \propto \sum_i \tau_{ik}^{(t)} \sum_l y_{ijl}$ where $\propto$ means that a normalization constant is set to respect the particular constraint of the corresponding factorization PMM-I or PMM-II.

Here, we can see that the factorization reduces the number of parameters from $Knpr$ in the saturated model (which would completely overfit the data with $\mu_{kijl} = y_{ijl}$), to Kp in λ_{kj}. Note that there is no continuous latent variable interpretation in w_i or $\boldsymbol{\lambda}$ here, however this model is deeply linked to mixture of multinomials. An implementation is given in the **HTSCluster** R package (Rau et al. 2015).

LINK WITH MIXTURE OF MULTINOMIALS A well-known fact is that the multinomial distribution may be written as the conditional distribution of a random vector $\boldsymbol{y}_i$ with Poisson independent coordinates, knowing its total count c:

$$\boldsymbol{y} \sim \textstyle\prod_{j=1}^{p} \mathcal{P}(y_j \mid \mu_j) \implies \boldsymbol{y} \mid c \sim \mathcal{M}_p(c, \boldsymbol{\theta}) \text{ with: } \boldsymbol{\theta} = \boldsymbol{\mu} \,/\, \textstyle\sum_j \mu_j$$

As noted in (Rau et al. 2015), using this equivalence, the PMM-I and PMM-II models may then be seen as a mixture of multinomials model:

$$\boldsymbol{y}_i \mid c_i \sim \mathcal{M}_{pr}(c_i, \boldsymbol{\theta}_k) \qquad \text{with: } \theta_{kjl} = s_{jl} \lambda_{kj}.$$

2.5 Model selection in model-based clustering

So far, the discussion on clustering considered the number of components K, and eventually the latent dimension d as a fixed hyper-parameter. However, in addition to its flexibility, the statistical framework provides a sound way to choose between different values of K as a model selection problem. Model selection is a transversal question across statistics, with a large variety of definitions ranging from information theoretic point-of-view (Hansen and Yu 2001), to penalized criteria and Bayesian approaches. A general discussion about these methods and their connections is beyond the scope of this thesis although we refer to Mattei (2017, chap. 2) for a nice overview of Bayesian model uncertainty. The following contains a modest discussion focusing on two specific penalized likelihood criteria, popular in model-based clustering and encountered in this thesis. We refer to Celeux et al. (2019) for a recent and thorough review about model selection in mixture models.

2.5.1 Bayesian Information Criterion and Laplace's approximation

The problem of choosing between M models $\mathcal{M}_1, \dots, \mathcal{M}_M$ is specially well-posed in the Bayesian paradigm. The core of Bayesian statistical modeling is the apprehension of uncertainty through prior distributions on parameters $p(\boldsymbol{\vartheta})$, now treated as latent variables. Then, a central notion in Bayesian model selection is the observed-data integrated likelihood:

$$p(\boldsymbol{Y} \mid \mathcal{M}_m) = \int_{\boldsymbol{\vartheta}_m} p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}_m, \mathcal{M}_m) p(\boldsymbol{\vartheta}_m \mid \mathcal{M}_m) \, \mathrm{d}\boldsymbol{\vartheta}_m, \tag{2.25}$$

also known as the evidence, or type-II likelihood (Berger 2013). Under a uniform prior over the models, we would like to pick the model $m^\star$ maximizing the evidence. The latter is generally an intractable integral over a space of dimension γ_m, and a numerous works are focused on finding good approximations, see *e.g.* Friel et al. (2017) and references therein. Informally, we stress out that integrating over parameters is a natural way of penalizing model complexity which is sometimes referred to as the Bayesian *Occam's razor* (MacKay 2002, chap. 28).

The Bayesian information criterion (BIC) is an asymptotic approximation of the integral above, based on a second order Taylor expansion of the logarithm of the integrand around its maximum $\hat{\boldsymbol{\vartheta}}$. In the following, the model $\mathcal{M}$ is supposed to be fixed, thus does not appear in the notations for the sake of simplicity. Define $g(\boldsymbol{\vartheta}) = \log p(\boldsymbol{Y}, \boldsymbol{\vartheta})$, and $\hat{\boldsymbol{\vartheta}}$ its maximum. Then, assuming that the latter exists, lies in the interior of the parameter space, and that g is twice differentiable, one has:

$$g(\boldsymbol{\vartheta}) = g(\hat{\boldsymbol{\vartheta}}) - \frac{1}{2}(\boldsymbol{\vartheta} - \hat{\boldsymbol{\vartheta}})^\top \boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}} (\boldsymbol{\vartheta} - \hat{\boldsymbol{\vartheta}}) + o(\|\boldsymbol{\vartheta} - \hat{\boldsymbol{\vartheta}}\|_2^2),$$

where $\boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}}$ is the negative hessian matrix of g at $\hat{\boldsymbol{\vartheta}}$. Then, assuming $\boldsymbol{\vartheta}$ is close to $\hat{\boldsymbol{\vartheta}}$ leaves a first approximation of the integrand in Equation (2.25), by exponentiating and neglecting the Taylor residuals:

$$p(\boldsymbol{Y}, \boldsymbol{\vartheta}) \approx p(\boldsymbol{Y}, \hat{\boldsymbol{\vartheta}}) \exp\left(-\frac{1}{2}(\boldsymbol{\vartheta} - \hat{\boldsymbol{\vartheta}})^\top \boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}} (\boldsymbol{\vartheta} - \hat{\boldsymbol{\vartheta}})\right). \tag{2.26}$$

Integration of this surrogate function over $\boldsymbol{\vartheta}$ is now tractable since the first term of the product is constant, and the second one corresponds to an un-normalized Gaussian density

with mean $\hat{\boldsymbol{\vartheta}}$ and covariance $\boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}}$ (positive definite since $\hat{\boldsymbol{\vartheta}}$ is the maximum). Thus, one has:

$$\int_{\boldsymbol{\vartheta}} \exp\left(-\frac{1}{2}(\boldsymbol{\vartheta}-\hat{\boldsymbol{\vartheta}})^\top \boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}}(\boldsymbol{\vartheta}-\hat{\boldsymbol{\vartheta}})\right) \mathrm{d}\boldsymbol{\vartheta} = (2\pi)^\gamma |\boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}}|^{-1/2},$$

with γ the dimension of the parameter space, and

$$\log p(\boldsymbol{Y}) \approx \log p(\boldsymbol{Y} \mid \hat{\boldsymbol{\vartheta}}) + \log p(\hat{\boldsymbol{\vartheta}}) + \frac{\gamma}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{A}_{\hat{\boldsymbol{\vartheta}}}|. \tag{2.27}$$

This method is called the Laplace's approximation for integral calculation (Tierney and Kadane 1986). The validity of the latter is discussed in Raftery (1995), who argues that in the large sample size, the function g is highly peaked around its maximum $\hat{\boldsymbol{\vartheta}}$, so that the higher-order terms quickly vanish in the Taylor expansion.

Then, several other approximations are made. First, $\hat{\boldsymbol{\vartheta}}$ is replaced by $\boldsymbol{\vartheta}^\star$, the MLE of Section 2.1, under the classical assumptions (regularity, identifiability, consistency) needed for the application of Bernstein-von Mises theorem (Vaart 2000, chap. 10.2). Second, assuming *i.i.d.* observations, the law of large number is used to approximate $\frac{1}{n}\boldsymbol{A}_{\boldsymbol{\vartheta}^\star}$ by $\boldsymbol{J}_{\boldsymbol{\vartheta}^\star}$, the Fisher information at $\boldsymbol{\vartheta}^\star$. Thus, $|\boldsymbol{A}_{\boldsymbol{\vartheta}^\star}| \approx n^\gamma |\boldsymbol{J}_{\boldsymbol{\vartheta}^\star}|$. A technical discussion on the assumptions and the cost of these approximations may be found in Lebarbier and Mary-Huard (2006). Then, the new approximation is:

$$\log p(\boldsymbol{Y}) \approx \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}^\star) - \frac{\gamma}{2}\log(n) + \underbrace{\log p(\boldsymbol{\vartheta}^\star) - \frac{\gamma}{2}\log(2\pi) - \log|\boldsymbol{J}_{\boldsymbol{\vartheta}^\star}|}_{\mathcal{O}_P(1)}. \tag{2.28}$$

Finally, the traditional BIC, originally proposed by Schwarz (1978), is obtained by dropping every $\mathcal{O}_P(1)$ terms above:

$$\log p(\boldsymbol{Y} \mid \mathcal{M}_m) \approx \log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}_m^\star) - \frac{\gamma_m}{2}\log(n). \tag{BIC}$$

In fact, discarding every $\mathcal{O}_P(1)$ terms basically amounts to ignore the prior. The popularity of the BIC is partly due to this quite appealing form, going from an integral involving a prior choice, to a maximized likelihood penalized by the dimension of the model. This rather frequentist way to perform Bayesian model selection has been underlined by several authors (see *e.g.* Robert and Rousseau 2016). In addition, an interesting discussion on the cost of dropping $\mathcal{O}_P(1)$ error terms can be found in Raftery (1995), who showed for Gaussian models that a data dependent prior may be set to reduce it to $\mathcal{O}_P(n^{-1/2})$. The BIC also features interesting asymptotic properties as demonstrated in Haughton (1988) which extended Schwarz's work and showed it is consistent, meaning that it is asymptotically able to capture the true model that generated the data if it is among $\{\mathcal{M}_m\}_m$. Still, note that several of the assumptions are not as mild as one might hope. In particular, the identifiability or regularity of the model are often not met in modern statistical models, which led to generalization of the criterion for these so-called *singular* models (Watanabe 2013).

Regarding identifiability or regularity, the specific case of mixture models do not avoid such pitfalls (Yamazaki and Watanabe 2003). However, specific results are available when the p.d.f. are bounded: Leroux (1992) proved that the BIC almost surely does not underestimate the number of clusters, later extended by Keribin (2000) which proved consistency. This criterion is very popular in the mixture model literature, allowing not only to com-

pare between different values of K, but also between constrained sub-models as discussed in Section 2.4 (McLachlan and Peel 2004). Empirical works have demonstrated its performance and usefulness on real data analysis (Roeder and Wasserman 1997; Dasgupta and Raftery 1998) and numerical simulations investigated its behavior in limit cases. Particularly, Celeux and Soromenho (1996) showed its propensity to underestimate the number of mixture components in low sample-sizes, while Biernacki et al. (2000) illustrated its tendency to over-estimate it when the true distribution is not a mixture.

Finally, note that the *i.i.d* hypothesis is crucial to leverage the approximations above, and is no longer met in other DLVMs such as the SBM or the LBM. Indeed, the observations are no longer marginally independent due to the complex dependency structure of their graphical models as discussed in Section 2.2.2.

2.5.2 Integrated Classification Likelihood

Specific to the field of model-based clustering with discrete latent variable models, a line of work proposed to take into account the clustering information for model selection, which the BIC could not (Celeux and Soromenho 1996; Biernacki and Govaert 1997). The integrated classification likelihood (ICL) was first introduced by Biernacki et al. (2000) for GMM, although its formulation was extended to any DLVM as:

$$\log p(\boldsymbol{Y}, \boldsymbol{Z} \mid \mathcal{M}_m) = \log \int_{\boldsymbol{\vartheta}_m} p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\vartheta}_m, \mathcal{M}_m) p(\boldsymbol{\vartheta}_m \mid \mathcal{M}_m) \, \mathrm{d}\boldsymbol{\vartheta}_m, \tag{2.29}$$

Now, as discussed in Section 2.2, the global parameter may be separated in two components $\boldsymbol{\vartheta} = (\boldsymbol{\pi}, \boldsymbol{\theta})$ governing the distribution of $\boldsymbol{Z} \mid \boldsymbol{\pi}$ and $\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}$ respectively. Considering the model $\mathcal{M}_m$ fixed with K clusters for now, and assuming the prior over parameter is factorized

$$p(\boldsymbol{\vartheta}) = p(\boldsymbol{\pi}) p(\boldsymbol{\theta}), \tag{2.30}$$

a straightforward application of Fubini's theorem gives:

$$p(\boldsymbol{Y}, \boldsymbol{Z}) = \int_{\boldsymbol{\theta}} p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} \int_{\boldsymbol{\pi}} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) p(\boldsymbol{\pi}) \, \mathrm{d}\boldsymbol{\pi} = p(\boldsymbol{Y} \mid \boldsymbol{Z}) p(\boldsymbol{Z}).$$

In addition, DLVMs assume that $p(\boldsymbol{Z} \mid \boldsymbol{\pi})$ is a product of multinomials parameterized by $\boldsymbol{\pi}$. Thus, taking a conjugate symmetric Dirichlet prior $\boldsymbol{\pi} \sim \mathcal{D}_K(\alpha, \dots, \alpha)$ parameterized by $\alpha > 0$, one can leverage Dirichlet-multinomial conjugacy to obtain an exact integral for $p(\boldsymbol{Z})$, Equation (2.29) becoming:

$$\log p(\boldsymbol{Y}, \boldsymbol{Z}) = \log p(\boldsymbol{Y} \mid \boldsymbol{Z}) + \log \left\{ \frac{\Gamma(\alpha K) \prod_k \Gamma(\alpha + n_k)}{\Gamma(\alpha)^K \Gamma(n + \alpha K)} \right\}, \tag{2.31}$$

where $n_k = \sum_i z_{ik}$ and $\Gamma(\cdot)$ is the Gamma function. Detailed calculations may be found in Appendix C.1 on page 149. This second term is common to all DLVMs, whereas the first term depends on the generative model at hand. Then, the different approaches taken for computing the conditional evidence $\log p(\boldsymbol{Y} \mid \boldsymbol{Z})$ leads to distinct criteria.

LAPLACE'S APPROXIMATION OF THE CONDITIONAL DISTRIBUTION In their seminal paper, Biernacki et al. (2000) proposed a BIC-like approximation on $\log p(\boldsymbol{Y} \mid \boldsymbol{Z})$ as:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{Z}) = \max_{\boldsymbol{\theta}} \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) - \frac{\gamma(\boldsymbol{\theta})}{2} \log(n). \tag{2.32}$$

As noted in the original paper, the parameter $\hat{\boldsymbol{\theta}}$ maximizing this conditional likelihood does not necessarily coincide with the MLE. However, it is not hard to compute if one resorts to an EM algorithm, since it may be viewed as the estimate of M-step with $\boldsymbol{\tau}$ replaced by $\boldsymbol{Z}$. Still, the approximation $\hat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}^\star$ is often done in practice and justified when the cluster are well-separated (*i.e.* $\boldsymbol{\tau} \approx \boldsymbol{Z}$).

The authors proposed a second approximation for the exact term $\log p(\boldsymbol{Z})$, using the Stirling formula on the Γ functions of Equation (2.31), assuming that the n_k's are large enough with n. After having set the hyper-parameter to $\alpha = 1/2$, which corresponds to an uninformative Jeffreys' prior (Robert 2007, p.129), they discarded every $\mathcal{O}(1)$ terms in Stirling formula finally uncovering:

$$\log p(\boldsymbol{Z}) \approx \max_{\boldsymbol{\pi}} \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}) - \frac{K-1}{2} \log(n). \tag{2.33}$$

The final approximations is given by summing the two above, leaving:

$$\log p(\boldsymbol{Y}, \boldsymbol{Z}) = \max_{\boldsymbol{\theta}} \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) - \frac{\gamma(\boldsymbol{\theta})}{2} \log(n) + \max_{\boldsymbol{\pi}} \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}) - \frac{K-1}{2} \log(n).$$

Then, the link with BIC is explicit by regrouping the likelihoods and penalizations together:

$$\mathrm{ICL}_{BIC}(\boldsymbol{Z}, \mathcal{M}_m) = \max_{\boldsymbol{\vartheta}} \log p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\vartheta}, \mathcal{M}_m, K) - \frac{\gamma_m}{2} \log(n), \tag{2.34}$$

where $\gamma_m = \gamma(\boldsymbol{\theta}_m) + K_m - 1$. Here, we made explicit the dependency on both the clustering $\boldsymbol{Z}$ and the model $\mathcal{M}_m$. Traditionally, the ICL_{BIC} criterion uses $\hat{\boldsymbol{Z}}$, a MAP estimate on the (variational) posterior defined by $\boldsymbol{\tau}(\hat{\boldsymbol{\vartheta}})$ in Equations (2.10) and (2.12). Here $\hat{\boldsymbol{\vartheta}}$ can be any estimator of $\boldsymbol{\vartheta}$, *e.g.* the MLE or the MAP. Then $\hat{\boldsymbol{Z}}$ is plugged into Equation (2.34) for the calculations.

The ICL_{BIC} criterion is very popular in the model-based clustering literature, and has also been used in non-Gaussian FMMs such as the mixture of multinomials defined in Section 2.4.2 (Biernacki et al. 2010). Moreover, the Laplace approximation of Equation (2.32) is now valid for any DLVM due to the conditional independence assumption given $\boldsymbol{Z}$. Thus, its construction has been extended to SBM (Daudin et al. 2008) and LBM (Lomet et al. 2012) modulo a factorized Dirichlet prior over row and column partitions. In a clustering perspective, it has been emphasized that its performance is very stable for recovering the true number of clusters, even when the model is ill-specified with respect to the true density of the data (Biernacki et al. 2000). This is partly due to its bias towards well separated clusters, while the BIC does not take into account clustering. For instance, in Gaussian mixtures, the BIC would tend to fit several Gaussian components in a single non-Gaussian cluster, privileging the density estimation perspective. This observation has led some authors to plead for a distinction between the notions of mixture component and cluster (Baudry et al. 2010).

VARIATIONAL APPROXIMATIONS OF THE CONDITIONAL LIKELIHOOD The ICL_{BIC} requires the conditional likelihood $p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta})$ to be tractable. However, the latter may be intractable when dealing with an additional integral over some hidden latent variable:

$$p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) = \int_{\boldsymbol{X}} p(\boldsymbol{Y}, \boldsymbol{X} \mid \boldsymbol{Z}, \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{X}.$$

This is the case in some extensions of the SBM (Bouveyron et al. 2018) or in Bayesian treatments of some mixture model such as Chapter 3. However, as suggested in Lomet et al. (2012), if a VEM algorithm is used to maximize $\log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta})$, the lower bound at convergence may be used as a surrogate for the maximized likelihood

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}, \mathcal{M}_m) \approx \mathcal{J}_{\boldsymbol{Z}, \mathcal{M}_m}(q^\star, \boldsymbol{\theta}_m^\star), \tag{2.35}$$

and plugged into Equation (2.32).

NON-ASYMPTOTIC CRITERIA: EXACT ICL Another line of works has taken the side of computing the first term of Equation (2.31) exactly, not relying on Laplace approximations. While its expression depends on the model at hand, the principled way to achieve an exact formula is through conjugate priors on $p(\boldsymbol{\theta} \mid \boldsymbol{\beta})$ where $\boldsymbol{\beta}$ denotes a set of hyper-parameters for the priors. Exact expressions have been derived this way for GMM (Bertoletti et al. 2015), the MoM model (Biernacki et al. 2010), the SBM (Côme and Latouche 2015) and LBM (Wyse et al. 2017). The corresponding criterion is then exactly the logarithm of the integrated classification likelihood without approximations:

$$\mathrm{ICL}_{ex}(\boldsymbol{Z}, \mathcal{M}_m) = \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\beta}_m, \mathcal{M}_m) + \log p(\boldsymbol{Z} \mid \mathcal{M}_m, \alpha). \tag{2.36}$$

It is particularly useful when $\boldsymbol{\beta}$ can be set to a value defining an uninformative or a Jeffreys prior over $\boldsymbol{\theta}$, which is the case for multivariate discrete data where consensual non-informative priors exists.

Motivated by the recent use of selection model criterion as objective functions (Tessier et al. 2006; Silvestre et al. 2014), another line of works consider a direct maximization of ICL_{ex} with respect to $\boldsymbol{Z}$. In a clustering objective, such methods avoid parameter estimation and the use of EM-like algorithms, while performing clustering and model selection in the same task. Naturally, the discrete nature of the optimization over $\boldsymbol{Z}$ leads to rely on greedy heuristics, looking for cluster swaps or merges marginally increasing the ICL_{ex}. First proposed in Côme and Latouche (2015) and later extended to other DLVMs, they represent a new promising field of model-based clustering with good performances and scalable algorithms. In Chapter 5 we address the problem of sensitivity to initialization and introduce an evolutionary genetic algorithm that efficiently combines several solutions, therefore avoiding spurious local maxima of $\boldsymbol{Z} \mapsto \mathrm{ICL}_{ex}(\boldsymbol{Z})$. Moreover, we derive a new approximation of the ICL_{ex} when the Dirichlet parameter α goes to 0. Then, we show how the latter may be viewed as a regularization parameter in a hierarchical clustering algorithm, constructing a set of nested and ordered solutions according to the new criterion.

Organization of the remaining chapters

Now that we have reviewed state-of-the-art methods, the remaining chapters will focus on the contributions of the thesis. Chapters 3 and 4 build on Section 2.4, with a Bayesian treat-

ment of finite mixture models combining dimension reduction. Chapter 3 tackles the case of high-dimensional count data clustering, and can be linked to the NMFEM model. Chapter 4 proposes a subspace Gaussian mixture models which, while related to MCFA, imposes different assumptions on the subspace properties. Both chapters propose new clustering algorithms relying on variational approximations described in Section 2.1. Then, Chapter 5 is placed in the general framework of Section 2.2, and addresses the problem of clustering and hierarchical clustering in discrete latent variable models, via a direct maximization of the ICL_{ex}. Finally, we propose a brief summary of these contributions, and a discussion about future works in Chapter 6.

3

Greedy clustering of count data through a mixture of multinomial PCA

This chapter introduces a new model-based algorithm for count data clustering, integrating mixture modeling and dimension reduction and capable of handling high-dimensional

datasets. We first describe a Bayesian formulation of the NMFEM algorithm which we call mixture of multinomial PCA, also known as the probabilistic clustering-projection model in its LDA formulation. We propose a novel procedure for this model, where inference and clustering are jointly done by efficiently mixing a greedy classification variational expectation maximization algorithm, with a branch & bound like strategy on a variational lower bound. An integrated classification likelihood criterion is derived for model selection, and a thorough study with numerical experiments is proposed to assess both the performance and robustness of the method. Finally, we illustrate the qualitative interest of the latter in a real-world application, for the clustering of anatomopathological medical reports, in partnership with expert practitioners from the Institut Curie hospital.

3.1 Introduction

Count data is becoming more and more ubiquitous in a wide range of applications, with datasets growing both in size and in dimension. In this context, an increasing amount of work is dedicated to the construction of statistical models directly accounting for the discrete nature of the data. However, it is common to encounter scenarios where the number of variables p is large with respect to the number of observations n. This is particularly the case in text applications, where p represents the number of unique words in a corpus of documents than can be orders of magnitude higher than the number of observations.

3.1.1 Integrating clustering and dimension reduction for count data

In Chapters 1 and 2, we saw that statistical estimations in high-dimensional settings present some pitfalls. Although geometric and probabilistic dimension reduction methods exist, such as NMF (Lee and Seung 2001) or MPCA (Buntine 2002), there is a need for principled statistical approaches integrating model-based clustering. In particular, Section 2.4.2 discusses the case of mixture of multinomials distributions with the NMFEM (Carel and Alquier 2017) and the factorized Poisson mixture models (Rau et al. 2015).

Recently, Watanabe et al. (2010) proposed an extension of the mixture of pPCA to exponential family distributions, putting explicit constraints on their natural parameter. The proposed variational Bayes algorithm relies on iterative clustering-projection phase, where the objective function is a variational lower bound of the model evidence with an additional Laplace approximation step. Specifically relying on topic models, in Chapter 5 of her PhD thesis, Wallach (2008) proposed the cluster topic model (CTM), an extension of LDA, where the latent topic proportions are now drawn from a mixture of K Dirichlet distributions with different hyper-parameters. Inference is done with a Gibbs sampling algorithm. Chien et al. (2017) proposed a variational Bayes algorithm for inference in the same model, along with a supervised version for text classification. Xie and Xing (2013) extended this model in their multi-grain clustering topic model, modeling an observation as a mixture between a *global* and a second mixture of *local* models LDA with different topic matrices. The inference relies on a VEM algorithm. However, we point out that the model is highly parameterized due to the multiple local LDA model parameters, causing the model to suffer from over-parametrization in high-dimensional problems with few observations.

In this chapter, we rely on the probabilistic clustering-projection model (PCP, Yu et al. 2005), a Bayesian formulation of the NMFEM model, relying on MPCA as well as mixture models. In this model, given the latent topic proportions, the law of an observation is a mixture of MPCA with the topic space $\boldsymbol{U}$ shared across clusters, hence its alternative name:

the mixture of multinomial PCA (MMPCA). Yu et al. (2005) originally proposed a VBEM algorithm for maximum likelihood estimation, then performing clustering with a maximum a posteriori estimate on the posterior cluster membership probabilities.

3.1.2 Contributions and organization of the chapter

In this chapter, we aim at clustering count data in high-dimensional spaces. To this end, we introduce a greedy inference procedure for MMPCA, focusing on maximizing an integrated classification likelihood. The algorithm is a refined version of the classification VEM (C-VEM) of Bouveyron et al. (2018), in the spirit of the branch & bound algorithm, where clustering and inference are done simultaneously. This approach, based on topic modeling, allows to tackle high-dimensional problems, with a limited number of observations. A reference implementation of the proposed algorithm is available in the **MoMPCA** R Package (Jouvin 2020).

In Section 3.2, we present the model and discuss some useful properties of its classification likelihood, along with its link to related models. Section 3.3 details the greedy clustering algorithm and an ICL criterion for model selection. Then, a thorough study on numerical simulations is detailed in Section 3.4, comparing the performance of MMPCA with other state-of-the-art methods. Finally, Section 3.5 describes a qualitative analysis for the clustering of oncology medical reports, in partnership with two expert doctors, illustrating the capacity of the methodology to uncover useful information from count data.

3.2 The model

This section aims at describing the MMPCA model, let us first recall some notations. In the following, $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i=1,\dots,N}$ denotes the set of observations, where $\boldsymbol{y}_i \in \mathbb{N}^p$. The total count for observation i will be denoted as $c_i := \sum_j y_{ij}$. In text analysis, p denotes the *vocabulary* size when observations are documents represented in a *bag-of-words* model, and y_{ij} is the j-th word total count inside document $\boldsymbol{y}_i$. In RNA-seq data, y_{ij} represents the total count of reads inside gene $\boldsymbol{y}_i$ in the j-th biological sample. In ecology, it might denote the observed number of plants belonging to species j in a geographical site $\boldsymbol{y}_i$. For more details about abundance count data, we refer to Cunningham and Lindenmayer (2005).

3.2.1 Mixture of Multinomial PCA

Recall the multinomial PCA formulation described in Section 2.3.2.b:

$$\begin{aligned} \boldsymbol{x}_i &\sim \mathcal{D}_d(\boldsymbol{\delta}), \\ \boldsymbol{y}_i \mid \boldsymbol{x}_i &\sim \mathcal{M}_p(c_i, \boldsymbol{U}\boldsymbol{x}_i), \end{aligned} \tag{MPCA}$$

where the columns of $\boldsymbol{U}$ are called *topics* and defines a discrete distribution $\boldsymbol{u}_{\cdot,h} \in \Delta_p$. Although MPCA allows dimension reduction on discrete data, it is not designed for clustering *per se*. Yu et al. (2005) proposed to integrate these two aspects simultaneously, using both topic and mixture modeling, in the same probabilistic model that we call mixture of MPCA (MMPCA) afterwards. Suppose that there are K low-dimensional latent variables $\boldsymbol{x}_k$, representing each cluster topic proportions, drawn independently:

$$\forall k,\ \boldsymbol{x}_k \sim \mathcal{D}_d(\,\delta). \tag{3.1}$$

Then, conditionally to its group assignment $\boldsymbol{z}_i$ and the set $\boldsymbol{X} = \{\boldsymbol{x}_k\}$, each observation is assumed to follow an MPCA distribution with cluster specific topic proportions:

$$\begin{aligned} \boldsymbol{z}_i &\sim \mathcal{M}_d(1, \boldsymbol{\pi}), \\ \boldsymbol{y}_i \mid \{z_{ik} = 1\}, \boldsymbol{X} &\sim \mathcal{M}_p(c_i, \boldsymbol{U}\boldsymbol{x}_k). \end{aligned} \tag{MMPCA}$$

The generative model is detailed in Figure 3.1. One of the main differences with MPCA is that the individual latent variable $\boldsymbol{x}_i$ now becomes $\boldsymbol{x}_k$, at the cluster level. Note that $\boldsymbol{U}$ does not depend on the cluster assignment and is common across groups. Knowing $\boldsymbol{X}$, a distribution of interest is the conditional *classification* likelihood, which can be written at the observation level:

$$p(\boldsymbol{y}_i, \boldsymbol{z}_i \mid \boldsymbol{X}; \boldsymbol{U}, \boldsymbol{\pi}) = \prod_{k=1}^{K} \left[\pi_k \, \mathcal{M}_p(\boldsymbol{y}_i \mid c_i, \boldsymbol{U}\boldsymbol{x}_k)\right]^{z_{ik}}. \tag{3.2}$$

Then, marginalizing on $\boldsymbol{z}_i$ leads to the conditional marginal distribution of an observation:

$$p(\boldsymbol{y}_i \mid \boldsymbol{X}; \boldsymbol{U}, \boldsymbol{\pi}) = \sum_{k=1}^{K} \pi_k \, \mathcal{M}_p(\boldsymbol{y}_i \mid c_i, \boldsymbol{U}\boldsymbol{x}_k), \tag{3.3}$$

which corresponds to a mixture of MPCA distributions, hence the model name. In the next section, we detail another formulation of the model which will prove useful for inference.

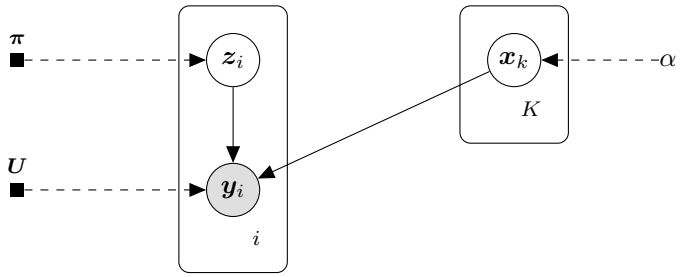


Figure 3.1: Graphical model representation of MMPCA.

3.2.2 Link with the NMFEM and latent Dirichlet allocation

Adopting a frequentist point of view considering $\boldsymbol{X}$ as a parameter for a moment, and looking at the distribution in Equation (3.3), one recognizes the NMFEM factorization of the mixture of multinomial in Equation (2.23). Thus the MMPCA model may be viewed as Bayesian formulation of NMFEM. However, the inference and optimization procedures differ as we will discuss in the next section. Carel and Alquier (2017) propose to focus on a marginal likelihood maximization through a regular EM algorithm. In this formulation, the E-step consists in computing the posterior distribution $p(\boldsymbol{Z} \mid \boldsymbol{Y}, \boldsymbol{X}; \boldsymbol{U}, \boldsymbol{\pi})$ which is available in closed form, while we rely on variational approximations. As for the M-step, the authors proposed to rely on the multiplicative updates of Lee and Seung (2001) in order to maximize the EM lower bound with respect to $\boldsymbol{X}$ and $\boldsymbol{U}$ iteratively. Clustering is done using a MAP estimate on the posterior of $\boldsymbol{Z}$ after convergence.

Moreover, as discussed in Section 2.3.2.b, MPCA admit an equivalent formulation, the

latent Dirichlet allocation (LDA, Blei et al. 2003) using the *bag-of-words* token representation $\boldsymbol{W} = \{\boldsymbol{w}_i\}_i$, with $\boldsymbol{y}_i = \sum_{l=1}^{c_i} \boldsymbol{w}_{il}$. It happens that MMPCA shares the same link, with a slight modification of the LDA generative model:

$$\begin{aligned} \boldsymbol{z}_i &\sim \mathcal{M}_K(1,\, \boldsymbol{\pi}), \\ \forall l \in \{1, \dots, c_i\}, \quad \boldsymbol{t}_{il} \mid \{z_{ik} = 1\}, \boldsymbol{x}_k &\sim \mathcal{M}_d(1,\, \boldsymbol{x}_k), \\ \boldsymbol{w}_{il} \mid \{t_{ilh} = 1\} &\sim \mathcal{M}_p(1,\, \boldsymbol{u}_{\cdot,h}). \end{aligned} \tag{PCP}$$

For any word $\boldsymbol{w}_{il}$, its topic assignment $\boldsymbol{t}_{il}$ can be marginalized out, leaving the distribution:

$$\boldsymbol{w}_{il} \mid \{z_{ik} = 1\}, \boldsymbol{x}_k \sim \mathcal{M}_p(1, \boldsymbol{U}\boldsymbol{x}_k). \tag{3.4}$$

Similarly to LDA, this distribution is independent of the choice of l, hence $(\boldsymbol{w}_{il})_l$ are conditionally independent knowing $\{z_{ik} = 1\}$ and $\boldsymbol{x}_k$, and identically distributed from (3.4). Furthermore, the correspondence with MMPCA appears clearly when marginalizing on $\boldsymbol{z}_i$:

$$p(\boldsymbol{w}_i \mid \boldsymbol{X}; \boldsymbol{U}, \boldsymbol{\pi}) = \sum_{k=1}^{K} \pi_k \prod_{l=1}^{c_i} \mathcal{M}_p(\boldsymbol{w}_{il} \mid 1, \boldsymbol{U}\boldsymbol{x}_k) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{p} (\boldsymbol{u}_j^\top \boldsymbol{x}_k)^{y_{ij}}. \tag{3.5}$$

Clearly, Equations (3.3) and (3.5) are equivalent, up to the multinomial coefficients which are independent of the parameters. Actually, this second model is the original formulation of the probabilistic clustering-projection model by Yu et al. (2005). For similar reason as in LDA, this formulation will be preferred for inference as the introduction of $\boldsymbol{T}$ allows deriving tractable variational lower bound. Hence, we will work with the LDA formulation throughout the rest of this chapter. Note that this implies a slight change of notation as $\boldsymbol{W}$ is now employed to design the whole set of observations. This is possible since the token and count representations are equivalent for this model.

3.2.3 Construction of the meta-observations

While the previous section discusses some useful properties of MMPCA at an observation level, another interesting feature of the latter arises when working with the whole set of observed variables. Indeed, conditionally on $\boldsymbol{X}$, observations belonging to the same cluster are independent and identically distributed from $\mathcal{M}_p(1, \boldsymbol{U}\boldsymbol{x}_k)$. This, along with the stability of the multinomial distribution under addition, suggests an aggregation scheme at the cluster level.

Proposition 3.1 (Proof in Appendix A.1 on page 134)**.** *Let $\boldsymbol{Z} = \{\boldsymbol{z}_1, \dots, \boldsymbol{z}_n\}$ be a set of discrete vectors in $\{0,1\}^K$ characterizing the clustering. Then,*

$$p(\boldsymbol{W}, \boldsymbol{X} \mid \boldsymbol{Z}; \boldsymbol{U}) = \prod_{k=1}^{K} \left[p(\boldsymbol{x}_k) \prod_{j=1}^{p} \left(\boldsymbol{u}_j^\top \boldsymbol{x}_k \right)^{\sum_{i=1}^{n} \sum_{l=1}^{c_i} z_{ik} w_{ilj}} \right]. \tag{3.6}$$

In the following, we define the aggregated counts of variable j in cluster k as $\tilde{\boldsymbol{W}}_{kj}(\boldsymbol{Z}) = \sum_{i=1}^{n} z_{ik} y_{ij} = \sum_{i=1}^{n} \sum_{l=1}^{c_i} z_{ik} w_{ilj}$. Then, knowing $\boldsymbol{Z}$, the p.d.f of Equation (3.6) is equivalent to that of a LDA model on K *meta-observations* $\tilde{\boldsymbol{W}}_k(\boldsymbol{Z}) = \{z_{ik}\boldsymbol{w}_{il}, i = 1, \dots, n, l = 1 \dots, c_i\}$. Therefore, with $\boldsymbol{Z}$ known and fixed, maximum likelihood inference is equivalent in our model and in a LDA model on the induced K meta-observations. Naturally, the construction of meta-observations depends on the clustering $\boldsymbol{Z}$. In the next section, we rely on this property

and propose a clustering algorithm, alternating between parameter inference in a model with $\boldsymbol{Z}$ fixed, and a clustering phase where $\boldsymbol{Z}$ is updated according to the current parameters.

3.3 A greedy clustering algorithm for MMPCA

We focus in this chapter in maximizing the following integrated classification log-likelihood:

$$\log p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{U}, \boldsymbol{\pi}) = \log \sum_{\boldsymbol{T}} \int_{\mathcal{X}} p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{T} \mid \boldsymbol{U}, \boldsymbol{\pi}) \, \mathrm{d}\boldsymbol{X}, \tag{3.7}$$

with respect to the parameters $(\boldsymbol{U}, \boldsymbol{\pi})$ as well as $\boldsymbol{Z}$. Contrary to the standard missing data framework of Dempster et al. (1977), we emphasize that $\boldsymbol{Z}$ is not treated as a set of latent variables and the goal is not to approximate its posterior distribution. Conversely, $\boldsymbol{Z}$ is seen as a set of binary vectors to be estimated through a discrete optimization scheme. Related to Bouveyron et al. (2018), this approach is grounded on Proposition 3.1 which, conditionally to the knowledge of $\boldsymbol{Z}$, casts MMPCA as a LDA model with K meta-observations, for which there exist efficient optimization procedures.

In this section, we propose a classification variational EM (C-VEM) algorithm mixed with an enhanced greedy swapping strategy in order to perform inference and clustering simultaneously . First, we derive a variational bound of Equation (3.7), alongside a VEM algorithm for inference. Then, we detail the proposed clustering procedure for the maximization in $\boldsymbol{Z}$. Finally, a model selection criterion is derived from our model to estimate the number of clusters together with the number of topics, relying on the *integrated* classification likelihood (ICL) of Biernacki et al. (2000).

3.3.1 Classification evidence lower bound

As discussed above, Equation (3.7) decomposes as a sum of a LDA term on the K aggregated meta-observations, plus a clustering term as follows:

$$\log p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{U}, \boldsymbol{\pi}) = \log \sum_{\boldsymbol{T}} \int_{\mathcal{X}} p\left(\tilde{\boldsymbol{W}}(\boldsymbol{Z}), \boldsymbol{X}, \boldsymbol{T} \mid \boldsymbol{Z}; \boldsymbol{U}\right) \mathrm{d}\boldsymbol{X} \; + \; \log p\left(\boldsymbol{Z} \mid \boldsymbol{\pi}\right). \tag{3.8}$$

Here, $\tilde{\boldsymbol{W}}(\boldsymbol{Z})$ represents the collection of the K meta-observations $(\tilde{\boldsymbol{W}}_k(\boldsymbol{Z}))_k$. Unfortunately, a direct consequence of Proposition 3.1 is that neither the integral in Equation (3.8), nor the posterior distribution of latent variables $p(\boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{Z}, \boldsymbol{W}; \boldsymbol{U}, \boldsymbol{\pi})$ have any analytical form. To tackle this issue, we propose to resort to variational approximation. The derivations are the same as in Section 2.1.3, except that $\boldsymbol{Z}$ is treated as an observed variable for now, and not as a latent one. Introducing a distribution $q(\boldsymbol{T}, \boldsymbol{X})$ on the latent variables, the following identity is true, for any clustering $\boldsymbol{Z}$:

$$\log p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{\pi}, \boldsymbol{U}) = \mathcal{J}(q(\cdot); \, \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z}) \; + \; \mathrm{KL}(q(\cdot) \,\|\, p(\cdot \mid \boldsymbol{W}, \boldsymbol{Z}; \boldsymbol{\pi}, \boldsymbol{U})),$$

with

$$\mathcal{J}(q; \, \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z}) = \mathbb{E}_{(\boldsymbol{T}, \boldsymbol{X}) \sim q} \left[\log p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{\pi}, \boldsymbol{U})\right] + \mathrm{H}\left(q\right). \tag{3.9}$$

Equation (3.9) constitutes a lower bound of Equation (3.7), which is an analog of the evidence lower bound in the standard VEM framework. Furthermore, the posterior being intractable, we follow Blei et al. (2003), and assume that $q(\cdot)$ factorizes over the two sets of latent

variables, *i.e.*:

$$q(\boldsymbol{T}, \boldsymbol{X}) = \prod_i \prod_l q(\boldsymbol{t}_{il}) \prod_k q(\boldsymbol{x}_k).$$

3.3.2 Optimization

Considering $\boldsymbol{Z}$ fixed for now, the goal is to maximize $\mathcal{J}$, with respect to q and the parameters $(\boldsymbol{\pi}, \boldsymbol{U})$. We consider a VEM algorithm as described in Section 2.1.3, cycling over q and $(\boldsymbol{\pi}, \boldsymbol{U})$, while maintaining one fixed. To that end, the following proposition rewrite the objective as the sum of the LDA variational lower bound on the K meta-observations and a clustering term.

Proposition 3.2 (Proof in Appendix A.2 on page 134)**.**

$$\mathcal{J}(q;\, \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z}) = \mathcal{J}_{LDA}(q(\cdot);\, \boldsymbol{U}, \boldsymbol{Z}) \,+\, \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}),$$

where

$$\mathcal{J}_{LDA}(q;\, \boldsymbol{U}, \boldsymbol{Z}) = \mathbb{E}_{(\boldsymbol{T}, \boldsymbol{X}) \sim q} \left[\log p(\tilde{\boldsymbol{W}}(\boldsymbol{Z}), \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{Z}; \boldsymbol{U}) \right] + \mathrm{H}\left(q\right). \tag{3.10}$$

With such a decomposition, maximizing $\mathcal{J}$ with respect to $\boldsymbol{\pi}$ is direct, and most of the work lies in the maximization of $\mathcal{J}_{\mathrm{LDA}}$ with respect to $\boldsymbol{U}$ as well as q. The latter can efficiently be done by constructing the meta-observations $\tilde{\boldsymbol{W}}(\boldsymbol{Z})$ and using the VEM algorithm of Blei et al. (2003).

Although this algorithm has already been described in Section 2.3.2.b, the notations have slightly changed, hence we recall the updates for the sake of completeness. The following propositions detail the CAVI update for each individual distribution, *i.e.* VE-step obtained from the maximization of Equation (3.10).

Proposition 3.3 (Proof in Appendix A.3 on page 135)**.** *The VE-step update for $q(\boldsymbol{t}_{il})$ is given by:*

$$q(\boldsymbol{t}_{il}) = \mathcal{M}_d(\boldsymbol{t}_{il} \mid 1,\, \boldsymbol{\phi}_{il} = (\phi_{il1}, \ldots, \phi_{ild})),$$

with

$$\forall (i, l, h), \quad \phi_{ilh} \propto \left(\prod_{j=1}^{p} u_{jh}^{w_{ilj}} \right) \prod_{k=1}^{K} \exp\left\{ \psi(\gamma_{kh}) - \psi\left(\textstyle\sum_{h'=1}^{d} \gamma_{kh'} \right) \right\}^{z_{ik}}.$$

Proposition 3.4 (Proof in Appendix A.4 on page 136)**.** *The VE-step for $q(\boldsymbol{X})$ is*

$$q(\boldsymbol{X}) = \prod_{k=1}^{K} \mathcal{D}_d\left(\boldsymbol{x}_k \mid \boldsymbol{\gamma}_k = (\gamma_{k1}, \ldots, \gamma_{kd})\right),$$

with

$$\forall (k, h), \quad \gamma_{kh} = \delta_k + \sum_{i=1}^{n} z_{ik} \sum_{l=1}^{c_i} \phi_{ilh}.$$

A fixed point algorithm is used, alternating between updates of Propositions 3.3 and 3.4, until the bound converges. Regarding $(\boldsymbol{\pi}, \boldsymbol{U})$, they appear in separate terms of $\mathcal{J}$. The maximization with respect to $\boldsymbol{U}$ corresponds to the M-step maximizing Equation (3.10), whereas the optimal $\boldsymbol{\pi}$ is simply the standard mixture proportion estimate.

Proposition 3.5 (Proof in Appendix A.5 on page 136 and A.6)**.** *The M-step estimates of* $\boldsymbol{U}$ *and* $\boldsymbol{\pi}$ *respectively are:*

$$\forall (j,h), \quad u_{jh} \propto \sum_{i=1}^{n} \sum_{l=1}^{c_i} \phi_{ilh}\, w_{ilj},$$
$$\forall k, \quad \pi_k \propto \textstyle\sum_{k=1}^{K} z_{ik}.$$

We now detail a clustering algorithm for MMPCA to estimate $\boldsymbol{Z}$.

3.3.3 A clustering algorithm for MMPCA

Optimizing the lower bound $\mathcal{J}$ in $\boldsymbol{Z}$ is a combinatorial problem, involving searching over K^n possible partitions. Although it is not possible to find a global maximum within a reasonable time, several heuristics have been proposed to explore efficiently local maxima. Among them, greedy methods have received an extended amount of attention. Notably, Bouveyron et al. (2018) proposed a C-VEM algorithm for the clustering of nodes in networks. While applicable in this setting, a regular C-VEM algorithm converges to local maxima of the variational lower bound leading to poor clustering performances. Hence, we propose a refined version of the C-VEM algorithm inspired from the branch & bound methods. Considering an initial clustering solution $\boldsymbol{Z}$, the algorithm starts by the VEM of Section 3.3.2, with $\boldsymbol{Z}$ fixed, and then cycles randomly through the observations. For each $\boldsymbol{y}_i$, all possible cluster swaps are tested, modifying $\boldsymbol{z}_i$, and leaving other observations unchanged. For each swap, meta-observations are updated and the VEM algorithm above is used again to update the variational distributions and the parameters. Then, the swap inducing the greatest positive variation of $\mathcal{J}$ is validated, if any, and $(\boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{U}, q)$ are updated accordingly. Moving to the next observation, the algorithm repeats the procedure until no possible swaps increasing the bound may be found, or when a user-defined maximum number of iterations is reached. The whole procedure is described in Algorithm 2 as a pseudo-code. A key difference between the C-VEM algorithm of Bouveyron et al. (2018) is that parameters and variational distributions are updated for each swaps in the greedy procedure, instead of being held fixed. This strategy is close to a *branch & bound* procedure, the lower bound acting as the surrogate for the objective

$$\forall (\boldsymbol{Z}, q, \boldsymbol{\pi}, \boldsymbol{U}), \quad \log p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{\pi}, \boldsymbol{U}) \geq \mathcal{J}_{\mathrm{LDA}}(q; \boldsymbol{U}, \boldsymbol{Z}) + \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}),$$

the goal is to efficiently explore a part of the decision tree by temporarily validating a swap, constructing new meta-observations, and re-maximizing the bound with respect to the parameters. It can be done efficiently thanks to the fact that a given swap, from cluster l to cluster k, only affects meta-observations $\tilde{\boldsymbol{W}}_l$ and $\tilde{\boldsymbol{W}}_k$. Thus, the cost of each VE-step is considerably reduced since the only needed updates concern observations in these two clusters.

Both VEM and greedy procedures are only ensured to converge to local maxima of $\mathcal{J}$, and we recommend several restarts with different initial clustering solution $\boldsymbol{Z}$, selecting the run achieving the greatest value. We also found that $\boldsymbol{U}$ plays a crucial role in the optimization algorithm. Therefore, we recommend to estimate it with a regular LDA on the whole set of observation at the beginning, without aggregating it, and to use it as a starting value for $\boldsymbol{U}$. Regarding the initialization of $\boldsymbol{Z}$, we found that there is a negligible impact of using a refined initialization strategy instead of a random balanced one. The methodology is robust to the initialization strategy, which is due to the ability of the branch & bound approach to

Algorithm 2: Branch and Bound C-VEM algorithm

Data: $\boldsymbol{Y}$
Result: Clustering $\boldsymbol{Z}$
Input: Q, K, any initializations for $\boldsymbol{Z}$ and $\boldsymbol{U}$. Maximum number of epochs: E.

$q, \pi, \boldsymbol{U} \leftarrow \texttt{VEM}(\boldsymbol{Y}, \boldsymbol{Z})$
$f \leftarrow \mathcal{J}(q; \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z})$

for $t \leftarrow 1$ **to** E **do**
  $\boldsymbol{Z}^{(old)} \leftarrow \boldsymbol{Z}$
  for $i \leftarrow 1$ **to** n **do**
    Find l such that $z_{il} = 1$
    for $k \leftarrow 1$ **to** K **do**
      if $k \neq l$ **then**
        Set $z_{ik}^{(tmp)} = 1$ and
        `// extra VEM-step: difference with a regular C-VEM`
        $q^{(k)}, \boldsymbol{\pi}^{(k)}, \boldsymbol{U}^{(k)} \leftarrow \texttt{VEM}(\boldsymbol{Y}, \boldsymbol{Z}^{(tmp)})$
        $f^{(k)} \leftarrow \mathcal{J}(q^{(k)}; \boldsymbol{\pi}^{(k)}, \boldsymbol{U}^{(k)}, \boldsymbol{Z}^{(tmp)})$
        Compute: $\Delta_i(k) \leftarrow f^{(k)} - f$.
      else
        $\Delta_i(l) \leftarrow 0$
      end
    end
    $k^\star \leftarrow \arg\max_k \Delta_i(k)$
    if $k^\star \neq l$ *and Cluster* l *not empty after swap* **then**
      Set $z_{ik^\star} = 1$, and $f \leftarrow f^{(k^\star)}$
  end
  if $\boldsymbol{Z} == \boldsymbol{Z}^{(old)}$ **then Break**;
end

efficiently explore the space of partitions.

3.3.4 Model selection

So far, everything described above considered the number of clusters K and topics d given and fixed. Thus, we still need to handle the task of estimating the best pair (K, d), which can be viewed as a model selection problem discussed in Section 2.5. In Carel and Alquier (2017), the authors proposed to rely on the BIC approximation of the evidence, since the observed data likelihood is available in closed form. In a clustering context, working with a classification likelihood, we propose a ICL-like criterion for our model, designed to approximate the likelihood of Equation (3.7) integrated with respect to the parameters: $\log p(\boldsymbol{W}, \boldsymbol{Z})$. The proposition hereafter results from a Laplace approximation combined with a variational estimation of the maximum log-likelihood, alongside a Stirling formula on the marginal law of $\boldsymbol{Z}$.

Proposition 3.6 (Proof in Appendix A.7 on page 137)**.** *A ICL criterion for MMPCA can*

be derived

$$\begin{aligned} \mathrm{ICL}_{MMPCA}(K,d) &= \mathcal{J}^\star\left(q(\cdot);\boldsymbol{\pi},\boldsymbol{U},\boldsymbol{Z}\right) \\ &\quad -\frac{d(p-1)}{2}\log(K)-\frac{K-1}{2}\log(n), \end{aligned} \tag{3.11}$$

where $\mathcal{J}^\star$ is the lower bound evaluated after convergence of Algorithm 2.

3.3.5 Run time and complexity

We now detail the algorithmic complexity of one epoch of Algorithm 2, where $\boldsymbol{U}$ is initialized once at the beginning and fixed. For an arbitrary observation $\boldsymbol{y}_i$ belonging to cluster l, all possible $K-1$ swaps from cluster l to cluster k are tested, where each swap has the computational cost of two VE steps in LDA. Indeed, from an implementation point of view, the only meta-observations affected by the swap are $\tilde{\boldsymbol{W}}_l(\boldsymbol{Z})$ and $\tilde{\boldsymbol{W}}_k(\boldsymbol{Z})$. Hence, we just need to update these two meta-observations accordingly, and run the VE-step fixed point algorithm in order to update q_l and q_k. The latter is simply the cost of computing (ϕ_l,ϕ_k) and (γ_l,γ_k) which is $\mathcal{O}(pd)$. Indeed, (ϕ_l,ϕ_k) requires to compute $2dp$ coefficients, whereas (γ_l,γ_k) requires only $2K$. There is an alternation between these two steps until convergence of the evidence lower bound, but, in practice, the convergence is really fast and there is no need for more than a few iterations for each VE-step. In conclusion, it makes $\mathcal{O}(nKdp)$ operations for one epoch. In the experimental setting of Section 3.4.3, one run of Algorithm 2 takes between 2 and 3 min on a single CPU with a frequency of 2.3 GHz, and Figure 3.8 shows the computational time evolution according to n.

3.4 Numerical Experiments

A specific simulation scheme is detailed in the following, in order to evaluate the performance of Algorithm 2.

3.4.1 Experimental setting

Hereafter, unless stated explicitly otherwise, the number of observations is fixed to $N=400$, with total count $c_i=250$, $\forall i$. The matrix $\boldsymbol{U}$ is computed only once on the whole corpus with a mixed strategy of a Gibbs sampling estimate as a starting point for the VEM algorithm of Blei et al. (2003). The maximum number of epoch in Algorithm 2 is fixed to $E=7$, and $\boldsymbol{Z}$ initialized randomly.

We describe hereafter how we simulate data from an MMPCA model. We propose to use the following values for model parameters:

$$K^\star=6,\ d^\star=4,\ \boldsymbol{X}^\star=\begin{bmatrix} 0.50 & 0.17 & 0.17 & 0.17 \\ 0.17 & 0.50 & 0.17 & 0.17 \\ 0.17 & 0.17 & 0.50 & 0.17 \\ 0.17 & 0.17 & 0.17 & 0.50 \\ 0.33 & 0.17 & 0.33 & 0.17 \\ 0.17 & 0.33 & 0.17 & 0.33 \end{bmatrix}.$$

It corresponds to a setting where each of the first four clusters is peaked towards one of the four topics, whereas the last two clusters are more *mixed* across topics.

Topics are defined using the empirical distribution of words across four different articles from BBC news, talking about unrelated issues: the birth of Princess Charlotte, black holes in astrophysics, UK politics, and cancer diseases in medicine. The matrix $\boldsymbol{U}^\star$ is then simply computed as the row-normalized document-term matrix of those four messages, and exhibits a strong block structure, implying that each topic uses a different set of words. The vocabulary size is $p = 915$, which makes it a fairly high-dimensional problem.

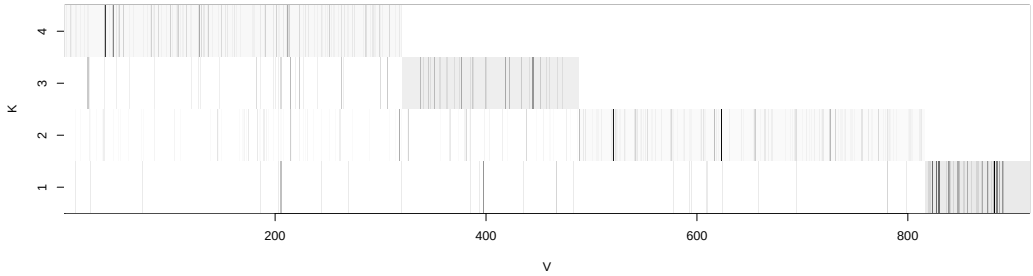


Figure 3.2: Visualization of the matrix $\boldsymbol{U}^\star$. Darker grey indicates stronger probabilities.

As we are dealing with a clustering task, a similarity metric, invariant to label switching, should be used to evaluate the quality of the recovered partition. Several choices are possible in the literature, here we chose the *Adjusted Rand Index* of Rand (1971) as it is a widely used and accepted metric in the clustering literature.

All experiments were run using the `R` programming language with the following methods comparison:

1. The non-negative matrix factorization algorithm proposed in Xu et al. (2003), denoted as `NMF`.
2. A clustering found by maximum a posteriori on the latent topic proportions of a LDA model. Inference is done with a VEM algorithm, with d fixed to $d^\star$.
3. A Gaussian mixture model (GMM) with $K^\star$ components in the latent space $\boldsymbol{X}$ of an LDA with $d^\star$ topics. This method will be called `GMM.LDA`.
4. A simple mixture of multinomial model for count data clustering, denoted as `MixMult`.
5. The `NMFEM` algorithm of Carel and Alquier (2017).
6. The factorized Poisson mixture model of Rau et al. (2011). This method is denoted as `HTSclust`, from the eponym package.

Our implementation of Algorithm 2 also relies on the `topicmodels` package of Hornik and Grün (2011)* for the VE-steps and lower bound computation detailed in Section 3.3.2

3.4.2 An introductory example

Figure 3.3 shows the joint evolution of the variational bounds and the adjusted rand index on a run of Algorithm 2. The random initialization gives an ARI close to 0, which is expected, then we observe a quick maximization of the bound on the first epoch, which

*Available on the CRAN

also corresponds to an amelioration of the ARI. After the first epoch, the bound growth is less pronounced, although swaps still happen at this stage. It tends to indicate that the marginal bound increase of a swap is decreasing. Furthermore, the passage from a good partition to the true one is done with an almost constant bound in the third epoch. Once the true partition is attained, no more swaps can maximize the bound. Hence, in this simple setting, the local maxima of the bound coincide with a maximum ARI. In the next section, we propose more complex simulations through the addition of a noise parameter.

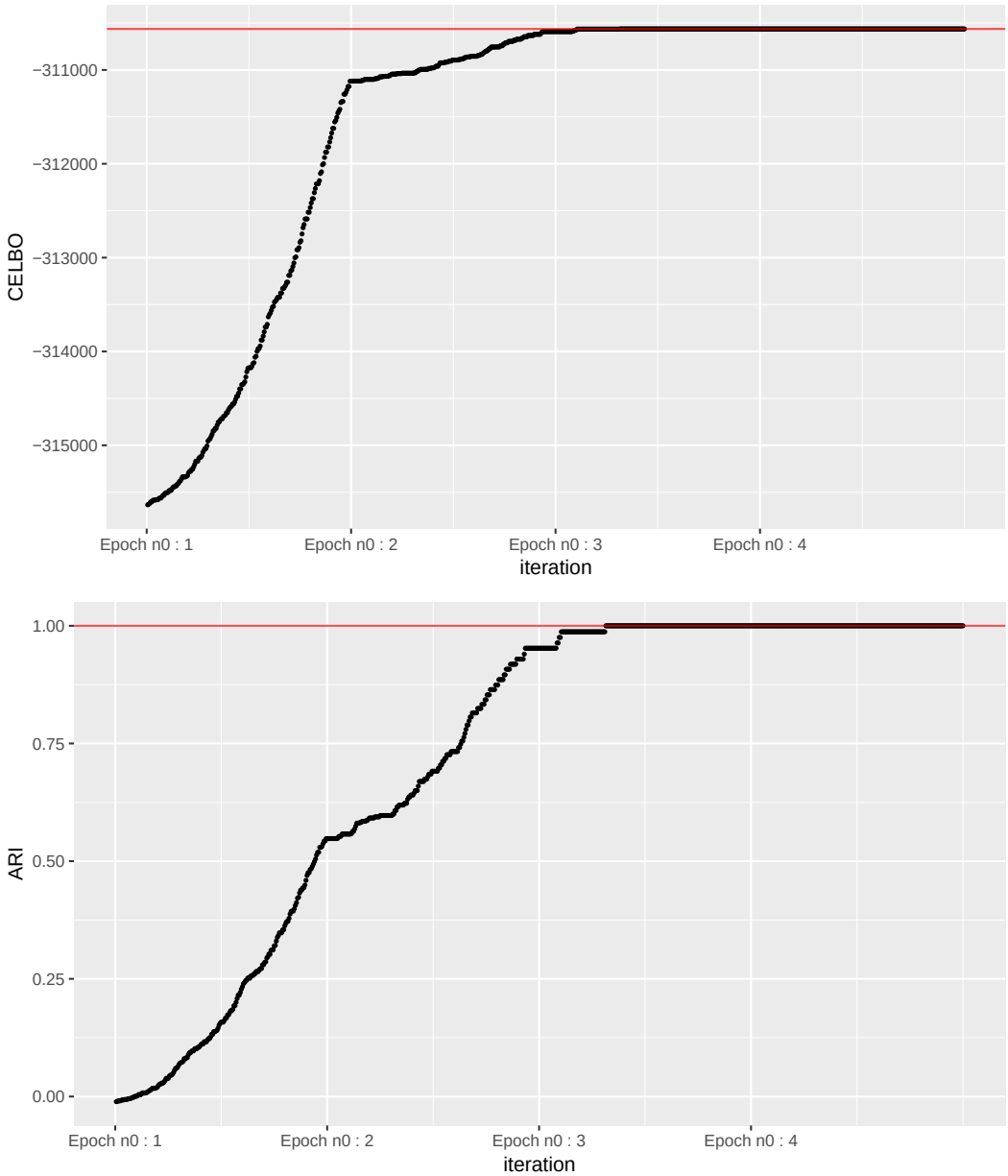


Figure 3.3: Lower bound (up) and ARI (down) evolution during a full run of the algorithm.

3.4.3 Robustness to noise

Leaving $\boldsymbol{U}^\star$ unchanged, hence controlling for its complexity, we propose to focus on $\boldsymbol{X}^\star$ to investigate the robustness of our method. Indeed, in order to complicate the simulation, we introduce noise in the observations by changing the distribution in the latent space. Indeed, fixing $\epsilon \in [0, 1]$ and modifying the generative process of the MMPCA model described in 3.2.2, we now draw:

$$\boldsymbol{t}_{il} \mid \{z_{ik} = 1\}, \boldsymbol{x}_k^\star \sim (1-\epsilon)\, \mathcal{M}_d(1, \boldsymbol{x}_k^\star) + \epsilon \mathcal{U}(\{1, \dots, d\})$$

Thus, $\epsilon = 0$ implies that each token in cluster k follows the standard MMPCA distribution $\mathcal{M}_d(1, \boldsymbol{x}_k^\star)$. When ϵ reaches 1, there is absolutely no cluster structure to be found and the groups are totally mixed since they all share the same common discrete distribution over topics $\mathcal{U}(\{1, \dots, d\})$.

Moreover, the strength of mixture modeling approaches is also to capture unbalanced cluster sizes. We propose to control group proportions via a parameter λ such that $\pi_k \propto \lambda^{K^\star - k}$. The case $\lambda = 1$ corresponds to balanced clusters, whereas $\lambda < 1$ put more emphasis on cluster 5 and 6, which may be considered as the *difficult* ones, considering that they are peaked towards two topics instead of only one.

Figure 3.4, 3.5 and 3.6 represent the mean ARI of each method with respect to the noise level, for $\lambda = 1$, 0.85 and 0.7 respectively. For every possible pair (λ, ϵ), means and standard errors are computed across 50 simulated datasets. The noise grid goes from $\epsilon = 0$, by 0.05 steps, to $\epsilon = 0.7$, since beyond this limit none of the tested methods is able to recover the true partition, the cluster structure behind being almost non-existent.

Overall, MMPCA performs really well when compared to competitors, demonstrating robustness both to noise and unbalanced clusters. The best competitor seems to be `GMM.LDA`, which, while basic, is advantaged by the knowledge of $(K^\star, d^\star)$, despite lacking a model selection criterion. The NMFEM method, which is the closest to our model, seems to perform quite correctly for low noise levels, but exhibits poor stability and efficiency with respect to noise. Moreover, it really seems to suffer from the high-dimensional setting, with fewer observations than variables. The stability of MMPCA over NMFEM advocates for the Bayesian approach, putting a prior on $\boldsymbol{X}$, which allow smoothing the dimensionality effect. The differences may also arise from the marginal versus classification likelihood maximization, and the algorithms used for optimization. The mixture of multinomial is really sensitive to noise and to the high-dimensional setting as well, thus supporting the idea of a latent topic factorization of the true parameters. The clustering obtained by LDA performs poorly, which is not surprising since LDA is not a clustering model for count data. Finally, NMF and HTScluster also exhibit a strong stability to noise while clearly underperforming compared to other methods for this scenario.

3.4.4 Model selection

While the results above are encouraging for MMPCA, they are conducted with the true values $(K^\star, d^\star) = (6, 4)$. This section evaluates the capacity of the ICL criterion proposed in Section 3.3.4 for every value of λ with $\epsilon = 0$, since this corresponds to the true model. The results are shown in Table 3.1, computed on 50 datasets for each value of λ. It demonstrates a good performance for $\lambda = 1$ and 0.85, while seeming sensitive to unbalanced clusters, as shown by the poor performance when $\lambda = 0.7$. Interestingly, for $\lambda = 0.7$, the criterion still selects $K = 4$ or $K = 5$, indicating that it could not capture smaller clusters, the high-dimensional setting with few data points for the smallest cluster complicating the asymptotic

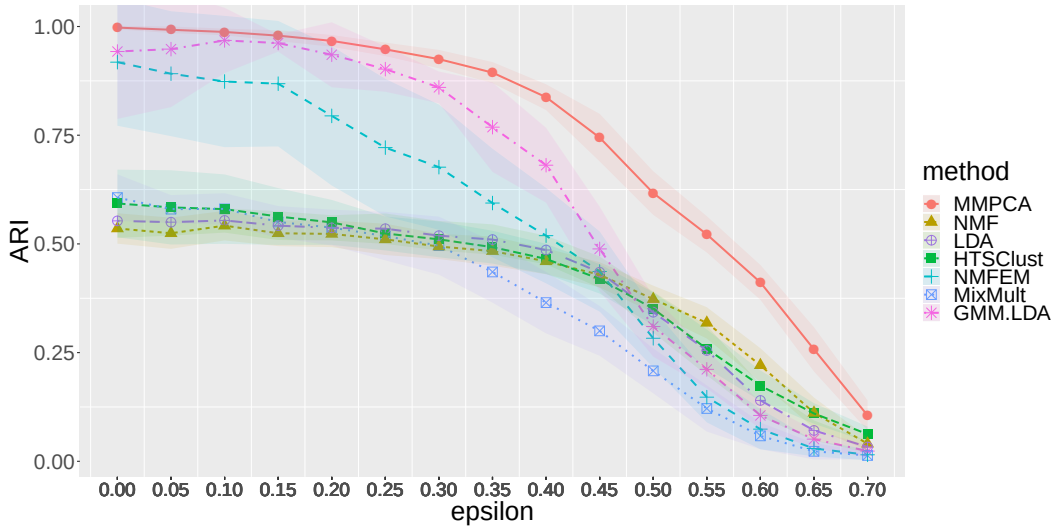


Figure 3.4: $\lambda = 1$. Mean ARI per noise level ϵ, with error bars. Each score is calculated on 50 simulated datasets.

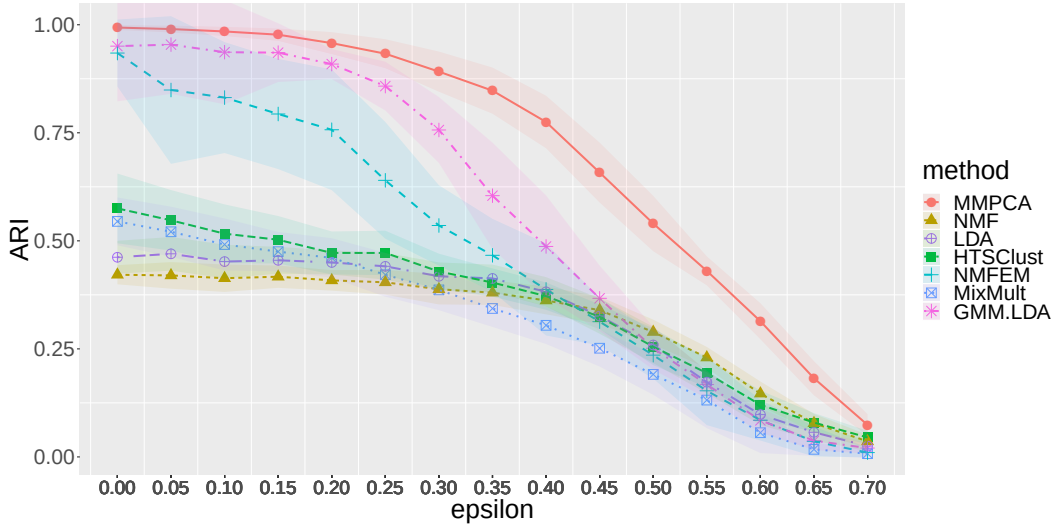


Figure 3.5: $\lambda = 0.85$. Mean ARI per noise level ϵ, with error bars. Each score is calculated on 50 simulated datasets.

in approximations.

3.4.5 Sensitivity to sample size

This last experiment aims at comparing the sensibility of every method to the dimensionality of the problem. Keeping the setting of Section 3.4.3, with $\lambda = 0.85$ and $\epsilon = 0.2$, 50 datasets

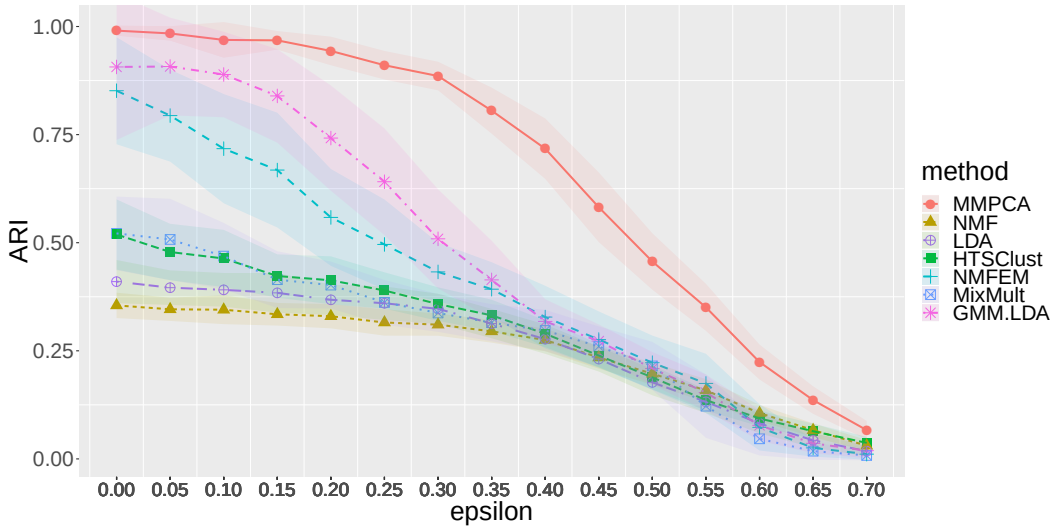


Figure 3.6: $\lambda = 0.7$. Mean ARI per noise level ϵ, with error bars. Each score is calculated on 50 simulated datasets.

K\d	2	3	**4**	5
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	100	0
7	0	0	0	0
8	0	0	0	0

$\lambda = 1$

K\d	2	3	**4**	5
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	2	0
6	0	0	98	0
7	0	0	0	0
8	0	0	0	0

$\lambda = 0.85$

K\d	2	3	**4**	5
2	0	0	0	0
3	0	28	0	0
4	0	50	8	0
5	0	0	6	0
6	0	0	8	0
7	0	0	0	0
8	0	0	0	0

$\lambda = 0.7$

Table 3.1: Percentage of correct selections with ICL on 50 simulated datasets. The actual number of clusters and topics are $K^\star = 6$ and $d^\star = 4$.

are simulated with an increasing sample size. Results are shown in Figure 3.7, in terms of the n/p ratio. MMPCA clearly demonstrates a great stability beyond $n/p = 0.1$, while `GMM.LDA` seems to be more sensitive, even at large sample sizes as the error bars demonstrate. It also indicates that NMFEM can perform well in this experimental setting, which was expected, although it still needs far more observations than the aforementioned methods to reach the

same performance. Basic mixture of multinomials also presents some amelioration with an increased sample size, yet still suffering from the high dimensionality of the problem. As for NMF and HTSclust, they present a remarkable stability in this scenario, not seeming to benefit from the increasing number of observations.

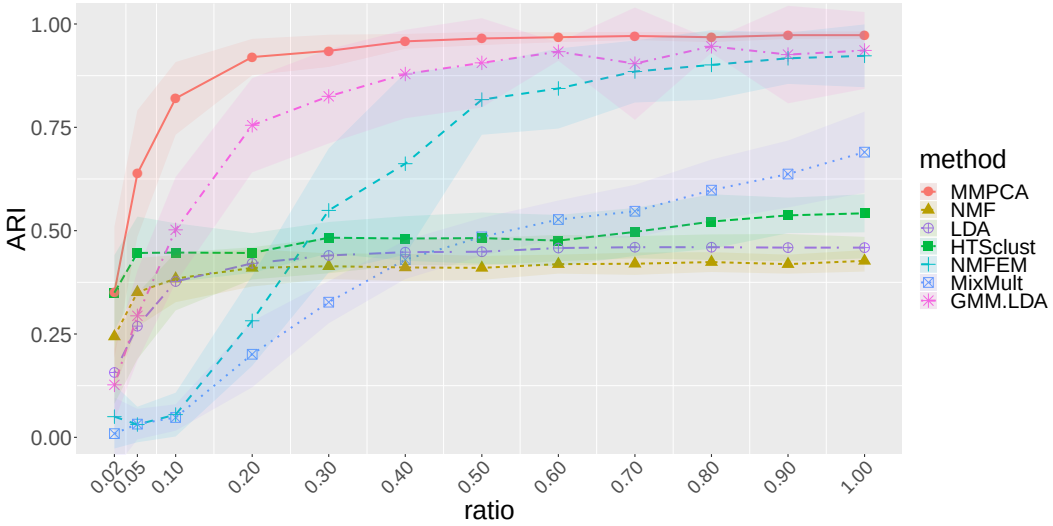


Figure 3.7: Stability with respect to sample size.

3.4.6 Computational complexity

Finally, Figure 3.8 shows the computational time of Algorithm 2 for increasing values of $N \in \{50, 100, \ldots, 1000\}$, and for $K = 6$, $d = 4$ and $p = 915$. As we can see, Algorithm 2 exhibits a linear growth with N, as discussed in Section 3.3.5. Moreover, the figure shows the complexity of running LDA with $d = 6$ and $d = 4$ topics. As we can see, relying on LDA.K6 for clustering, on LDA.K4 for topic modeling, or both at the same time, induces computational times of the same order of magnitude as Algorithm 2.

3.5 Applications to the clustering of anatomopathological reports

With 58,000 new cases in 2018 in France (Defossez et al. 2019), breast cancer is the most common malignant disease in women. Earlier diagnosis and better adjuvant therapy have substantially improved patient outcomes. The pathologist establishes the diagnosis and provides prognostic and predictive factors of response to treatments. This is done by observing microscope slides of biological samples from both core needle biopsies and surgical specimens. Indeed, the microscopical aspect of cellular constituents and architecture are a fundamental part of diagnosis. Thus, information such as the histological type of the lesion (Lakhani 2012), the histopathological grading (Ellis and Elston 2006), or molecular classification (Sorlie et al. 2003), are recorded in medical reports. The latter are heterogeneous, unstructured textual data, varying both with the pathologist writing style and with the change in medical conventions throughout time. Although we have access to the pathologist conclusion on the lesion type, *i.e.* the label, it is of interest to perform a deeper analysis to understand the variety and richness of information present.

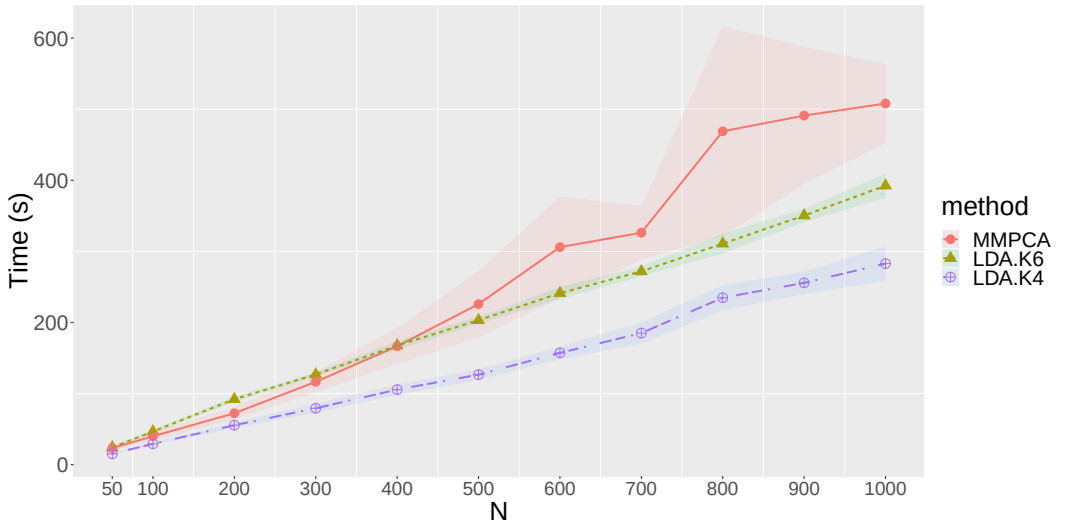


Figure 3.8: Mean computational time for 10 runs of Algorithm 2, LDA with $d = 6$ and $d = 4$ topics. The number of observations is increasing while the dimension is fixed to $p = 915$. The simulation setting is the same as the one in Section 3.4.3 with $\epsilon = 0$ and $\lambda = 1$.

The dataset considered here consists in about 900 medical reports from the anatomopathological service of Institut Curie, a French hospital specialized in Cancer treatment. These reports describe histological lesions in tissues sampled from the core needle breast biopsy. The lesions considered can be of two types: either benign, meaning there is no need for a medical care, or malignant lesion requiring specific care such as surgery, chemotherapy, and/or radiotherapy. World health organization classification of tumors of the breast divides malignant breast carcinomas in several types, including two main sub-categories (Lakhani 2012): non-special type (NST, e.g. ductal) and lobular. In this study, only these two subtypes of invasive cancer are considered. Removing the conclusion from all documents, we only keep the descriptive part, and are interested in clustering those anatomopathological reports to understand the information present in them. For this, Algorithm 2 was run with $K = 2, \ldots, 10$ and $d = 2, \ldots, 7$ on a document-term matrix consisting of unigrams and a short hand designed word list. The vocabulary size is 302 and the ICL criterion of Proposition 3.6 chose $K = 7$ clusters and $d = 5$ topics.

In order to make a qualitative analysis of the results, Table 3.2 shows the number of label assignments for each cluster. The algorithm has found a clear separation between benign lesions in cluster 4, lobular invasive carcinoma in cluster 1, and NST invasive carcinoma split in the 5 smaller clusters. Observing the three NST documents in Cluster 4 revealed that they focus a lot on describing benign lesions with minor invasive ones, thus explaining their clustering. Moreover, the smaller NST clusters are quite interesting since we recover some of the known prognostic and predictive factors of carcinomatous lesions. Indeed, cluster 5 is the biggest cluster and corresponds to a high-grade invasive NST carcinoma which is expected. Cluster 7 contains a lot of description of the stroma, which is known to have a major impact on response to the chemotherapy and patient outcome. As for the architecture aspect, Cluster 3 and 6 contains reports with well-differentiated architectures for the former and undifferentiated for the latter, implying a higher level of malignity. When looking at

Cluster 2, we may see that there is a lot of microcalcifications and in-situ[†] cancerous lesions in the report descriptions. This can be explained by the fact that almost all samples present in this cluster came from a particular type of breast biopsy: macrobiopsy. These are almost exclusively used to search for cancerous lesions after the detection of microcalcifications in a breast mammography. Indeed, microcalcifications are considered as suspect in the development of cancerous tumors, especially the in-situ NST ones. This is interesting to know that we can recover information such as the type of medical exam from the description of tumorous lesions when it does not appear in the text .

	Benign	Non special type carcinoma (e.g. ductal)	Lobular carcinoma
1	0	0	43
2	1	31	1
3	0	106	0
4	231	3	0
5	0	211	0
6	0	126	0
7	0	113	0

Table 3.2: Confusion matrix of document label along cluster.

Making use of the property described in Proposition 3.1, we estimate the topic matrix $\boldsymbol{U}$ and the cluster topic proportions $\boldsymbol{X}$ on the 7 meta-documents aggregated according to the final clustering. The variational estimate of all $\boldsymbol{x}_k$, consisting of the normalized γ_k, is given in Table 3.3, while the most probable words per topic are shown in Figure 3.9. The topic analysis provides a deeper insight and concordant results with the qualitative analysis above.

Topic 1. This topic focus on general descriptive aspects of a tumor. In particular, words like "tumoral", "tumor", or "cytonuclear" are commonly used in medical reports when describing a tumor lesion. A word like "abundant" is related to stroma description, which explains why Cluster 7 is peaked toward this topic.

Topic 2. With keywords like "invasive ductal carinoma" corresponding to the lesion type and "poorly", "high" corresponding to the histopathological grading of the tumor (Ellis and Elston 2006), this topic correspond to high-grade invasive ductal carinoma. Interestingly, Cluster 5 is completely peaked towards topic 2, and the analysis of the grade reveals that most of them are from intermediate to high.

Topic 3. The keywords "independent cells" and "fibro-elastic stroma" are commonly used to describe "invasive lobular carcinoma" lesion. As expected, Cluster 1 is entirely peaked toward this topic since it contains all invasive lobular carcinoma.

Topic 4. Containing some keywords like "in situ", "high", "intermediate" or "necrosis", this topic is clearly related to the lexical field of in-situ lesions that can be associated with invasive cancer. We can see that Cluster 2, 3 and 6 are associated to this topic. It was known for Cluster 2 since it involves microcalcifications. However it brings some more information about the two other clusters.

[†]In-situ cancers are pre-invasive lesions that get their name from the fact that they have not yet started to spread. Invasive cancer tissues can contain both invasive and in-situ lesions in the same slide.

	Topic1	Topic2	Topic3	Topic4	Topic5
$\boldsymbol{x}_1$	0.00	0.01	**0.98**	0.00	0.00
$\boldsymbol{x}_2$	0.19	0.11	0.04	0.38	0.29
$\boldsymbol{x}_3$	0.13	0.09	0.01	**0.76**	0.00
$\boldsymbol{x}_4$	0.01	0.00	0.01	0.01	**0.97**
$\boldsymbol{x}_5$	0.00	**1.00**	0.00	0.00	0.00
$\boldsymbol{x}_6$	0.05	**0.65**	0.03	0.26	0.01
$\boldsymbol{x}_7$	**0.74**	0.12	0.03	0.11	0.00

Table 3.3: The matrix of estimated $(\boldsymbol{x}_k)_{1,\dots,7}$. The topics are associated to those described in Figure 3.9.

Topic 5. This topic is characteristic of the benign lesions lexical field. The keywords "cylindric metaplasia", "fibrocystic" or "simple" are related to benign breast lesions that are all grouped inside Cluster 4. It also contains "microcalcification" which is characteristic of Cluster 2 as explained above.

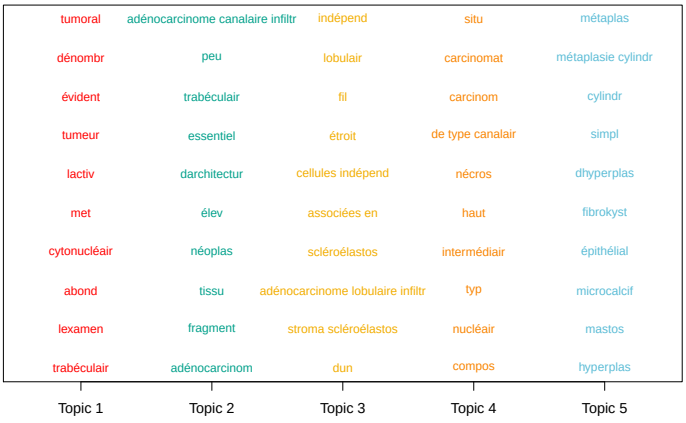


Figure 3.9: Most probable words per topics estimated on the aggregated Document Term Matrix.

3.6 Conclusion

In this work, we introduced a new algorithm for the clustering of count data based on a mixture of MPCA distributions, allowing associating the dimension reduction aspect of topic modeling with model-based clustering. The methodology maximizes a variational bound of an integrated classification likelihood of the model in a greedy fashion, handling both parameter inference and discrete optimization with respect to the partition. In addition, an ICL-like model selection criterion was proposed to select the number of clusters and topics. Experiments on simulated data were used to assess the interest of the proposed approach, its performances comparing favorably with other methods in different scenarios. Notably,

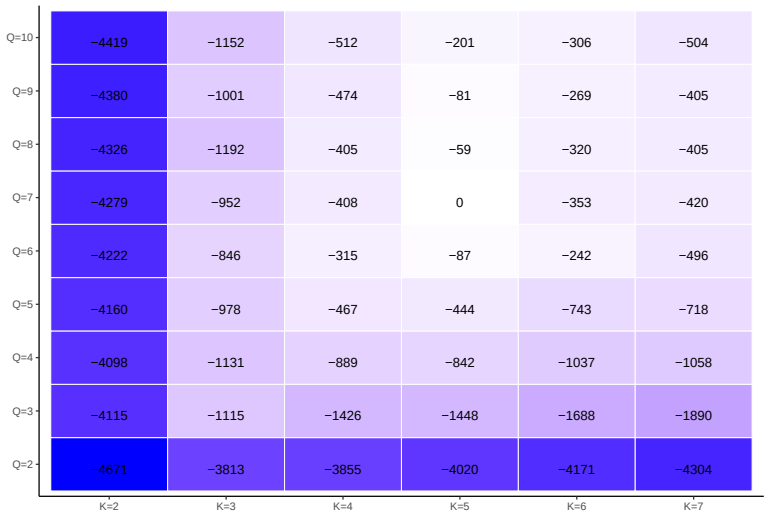


Figure 3.10: Model selection for MMPCA on the Curie anatomopathological report datasets. The ICL criterion values are displayed with the maximum value subtracted for visualization purpose.

a real data application in medical report clustering illustrated the capacity to unveil some relevant structure from count data.

4

Discriminative Gaussian subspace clustering and the Bayesian Fisher EM algorithm

In the previous chapter, we introduced a Bayesian treatment of a factorized mixture of multinomials for count data. In this chapter, we propose a Bayesian extension of the

discriminative latent mixture model (DLM, Bouveyron and Brunet 2012a), related to MCFA. A clustering algorithm, the Bayesian Fisher EM, is derived, and a model selection criterion is proposed via the integrated classification likelihood. Simulation on numerical data shows a significant gain of our methodology.

4.1 Introduction

Consider a data matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$ consisting of continuous observations $\boldsymbol{y}_i$ that we want to cluster into K classes. As discussed in Chapter 2, the Gaussian mixture model is well suited for such a task, with:

$$p(\boldsymbol{Y} \mid \boldsymbol{\pi}, \boldsymbol{m}, \boldsymbol{S}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{m}_k, \boldsymbol{S}_k). \tag{4.1}$$

However, it suffers from a form of the famous *curse of dimensionality*, as its covariance matrices $\boldsymbol{S}_k$ involve a number of parameters growing with the squared of the dimension. In such a context, the number of observations required to fit high-dimensional data may be very large.

On the one hand, some approaches rely on unsupervised dimension reduction such as PCA of factor analysis to project the data prior to model fitting (Ghosh and Chinnaiyan 2002). However, such transformations do not take into account the clustering task at hand and might induce a loss of relevant discriminative information, in addition to losing the principled approach to model-based clustering. Indeed, Chang (1983) designed a 2-component simulation setting in dimension $p = 15$ where the groups are best discriminated on the space defined by the first and the last components, as represented in Figure 4.1. Another example of this phenomenon is also given in McLachlan and Peel (2004, sec. 8.2) in dimension $p = 5$. Model-free heuristics have also been proposed for subspace clustering, seeking for regions of high-density within the observed space. The CLIQUE algorithm (Agrawal et al. 1998) is a popular instance of such methods, and the building block for many others. We refer to Parsons et al. (2004) for a comprehensive review on this subject.

On the other hand, a wealth of literature has focused on developing parsimonious models of Equation (4.1) consisting of low-rank factorization of the covariance matrices, reducing the number of parameters to estimate. Section 2.4 contains a detailed introduction of these models, based on a factor analysis formulation, which may be interpreted as dimension reduction methods, searching to cluster the data in low-dimensional subspace(s). Maximum likelihood inference is always preferred in these models, usually via an EM algorithm. However, only a subset of these models share a common subspace for all the data, which is of interest for visualization purpose. Moreover, without further clustering information, the estimated latent subspace may be biased toward density estimation, preserving the variance of the observed data as much as possible, rather than clustering and explicit separation of the groups. These objectives are not always aligned as Figure 4.1 suggests, and in order to circumvent this issue, several works introduced the notion of a discriminative subspace.

4.1.1 Discriminative subspace: from classification to clustering

In the supervised framework, where $\boldsymbol{Z}$ is observed, the tension between signal *representation* and signal *classification* is well known and analogous to the distinction between density estimation and clustering. Fukunaga (1990, chap. 10) discusses this in detail, introducing

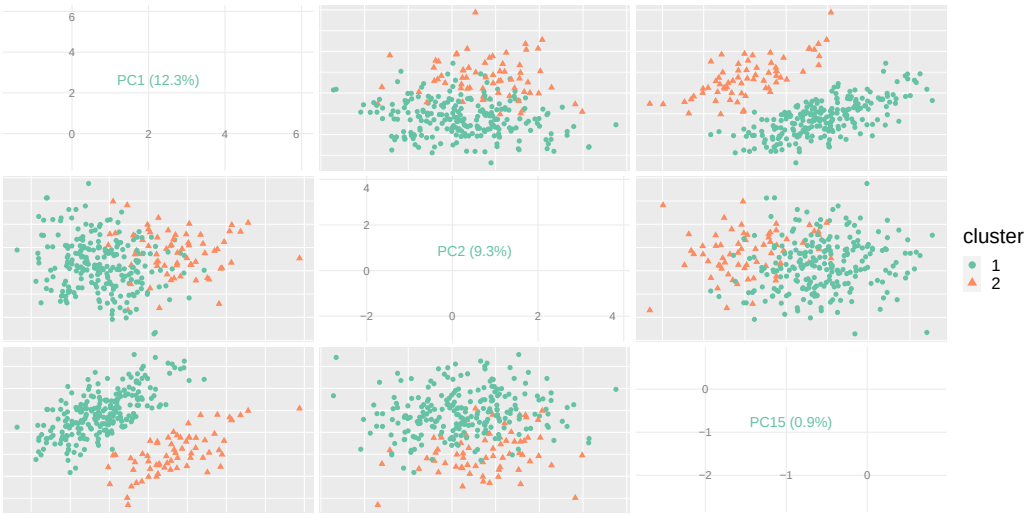


Figure 4.1: Chang (1983) data set with $n = 300$ observations projected on the 1st, 2nd and 15th principal components respectively. Colors and shapes indicate the true cluster membership. We see that the last principal component contains important discriminative information in terms of clustering, while the second principal component is not suited for the task.

the notion of class separability along with 4 different possible criteria to measure it. The idea is to find a linear subspace $\boldsymbol{U}$ in which the group means are well separated while the within-class variance is small. The most popular criterion for such a task is a generalization of Fisher's Linear Discriminant Analysis (Duda et al. 2000, chap. 4):

$$\mathrm{F}(\boldsymbol{U}) = \mathrm{Tr}\left[(\boldsymbol{U}^\top \boldsymbol{S}_W \boldsymbol{U})^{-1} \boldsymbol{U}^\top \boldsymbol{S}_B \boldsymbol{U}\right], \tag{4.2}$$

where $\boldsymbol{S}_W = (1/n)\sum_k \sum_{i:\boldsymbol{z}_{ik}=1} (\boldsymbol{y}_i - \boldsymbol{m}_k)(\boldsymbol{y}_i - \boldsymbol{m}_k)^\top$ is the within-class covariance matrix of the data, while $\boldsymbol{S}_B = (1/n)\sum_k n_k (\boldsymbol{m}_k - \bar{\boldsymbol{y}})(\boldsymbol{m}_k - \bar{\boldsymbol{y}})^\top$ is the between-class covariance. This criterion computes the trace of the ratio between the within-class and between-class covariance matrices in the latent space, and its maximization with respect to $\boldsymbol{U}$ translates the goal of a discriminative subspace. Without orthogonality constraints, the maximization of Equation (4.2) happens to be equivalent to a generalized eigenvalue problem $\boldsymbol{S}_B \boldsymbol{u} = \lambda \boldsymbol{S}_W \boldsymbol{u}$ which can be solved efficiently, even when $\boldsymbol{S}_W$ is singular (Ye 2005). Moreover, since the rank of the matrix $\boldsymbol{S}_B$ is at most $K - 1$, there is only $d \leq K - 1$ dimensions of interest.

In an unsupervised context, $\boldsymbol{Z}$ is unknown, and the scatter matrices $\boldsymbol{S}_W$ and $\boldsymbol{S}_B$ cannot be formed. Still, building on these ideas, some works have been proposed to adapt the criterion. In the goal of feature selection for clustering, Dy and Brodley (2004) compared maximum likelihood approaches with the maximization criterion of Equation (4.2), highlighting the interest of both in different contexts. Feature selection can be cast in the framework of dimension reduction where $\boldsymbol{U}$ is forced to be a $(0, 1)$-matrix with columns' non-zero index indicating the subset of d selected variables (Nie et al. 2008). For clustering applications, Torre and Kanade (2006) proposed the Discriminative Cluster Analysis, combining k-means and linear discriminant analysis. In a visualization approach, Scrucca (2010) proposed to project the data in a subspace minimizing the Fisher criterion, using the partition given by the model of Equation (4.1). It relies on a modified version of $\boldsymbol{S}_B$ taking into account variability between within-class covariance, and demonstrates good vi-

sualization power. However, this method is post-inference and still requires to fit a GMM in the observation space which is prohibitive in real high-dimensional scenarios. Finally, Bouveyron and Brunet (2012a) proposed the discriminative latent mixtures (DLM) model, which treats the estimation of the latent subspace as a separate problem from maximum likelihood estimation. The proposed model is close to the MCFA of Baek and McLachlan (2008), although inference is different and done via the Fisher EM algorithm which mixes the EM strategy with an F-step. In the latter, the current posterior membership probabilities τ_{ik} are used to compute the scatter matrices $\boldsymbol{S}_W$ and $\boldsymbol{S}_B$, and $\boldsymbol{U}$ is supposed to be discriminant, maximizing Equation (4.2) with orthogonality constraints.

Note that, albeit not directly related to high-dimensional estimation, the idea of incorporating clustering information is popular in the context of mixture modeling. It dates back to the CEM algorithm (Celeux and Govaert 1992) aiming at maximizing the classification likelihood for inference. It is also present in the ICL criterion of Section 2.5 for selecting the number of clusters. This trade-off between clustering and density estimation led to the proposition of differentiating between the notion of mixture component and cluster in GMM (Baudry et al. 2010), the former being a combination of the latter.

4.1.2 Contribution and organization of the chapter

In Section 4.2, we introduce a Bayesian formulation of the DLM model putting a prior on the mean in the latent space, with a hyper-parameter λ controlling the between-class variance. Following Bouveyron and Brunet (2012a), we derive a family of sub-models with constraints on the latent covariance matrices, and discuss its links to existing methods. Then, the posterior now being intractable, Section 4.3 introduces a variational extension of the Fisher EM algorithm for simultaneous clustering and dimension reduction. Section 4.4 assesses the corresponding Bayesian Fisher EM algorithm in several high-dimensional scenarios on both simulated and real data, along with a detailed comparison with state-of-the-art model-based subspace clustering models.

4.2 The Bayesian discriminative latent mixture

4.2.1 Discriminative latent mixture

Bouveyron and Brunet (2012a) proposed the following generative model, relying on the idea that $K-1$ properly chosen dimensions are sufficient to discriminate between K classes. It is based on a factor analysis like formulation with a linear Gaussian model:

$$\begin{aligned} \boldsymbol{z}_i &\sim \mathcal{M}_K(1, \boldsymbol{\pi}), \\ \boldsymbol{x}_i \mid \{z_{ik} = 1\} &\sim \mathcal{N}_d(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \\ \boldsymbol{y}_i &= \boldsymbol{U}\boldsymbol{x}_i + \boldsymbol{\epsilon}_i, \qquad \boldsymbol{\epsilon}_i \mid \{z_{ik} = 1\} \sim \mathcal{N}_p(\boldsymbol{0}_p, \boldsymbol{\Psi}_k). \end{aligned} \tag{DLM}$$

Here, the matrix $\boldsymbol{U}$ is constrained to be column-orthonormal $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}_d$, and form the basis of a low-dimensional subspace of dimension $d \leq \min(K-1, p)$, which is called the *discriminative subspace*. When the latent variables are integrated out, the marginal distribution is a constrained GMM:

$$\boldsymbol{y}_i \sim \sum_{k=1}^{K} \pi_k \mathcal{N}_p(\boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{U}\boldsymbol{\Sigma}_k\boldsymbol{U}^\top + \boldsymbol{\Psi}_k),$$

This model relates to the MCFA model presented in Section 2.4.1, except that the latent dimension is constrained to be at most $K-1$, and $\boldsymbol{U}$ is considered to be discriminative in the sense of Fisher's criterion. Moreover, the noise matrices $\boldsymbol{\Psi}_k$ are not constrained to be common across clusters nor diagonal anymore, but rather to be isotropic in the orthogonal of the subspace. Formally, define $\boldsymbol{D} = [\boldsymbol{U}, \boldsymbol{V}]$ where $\boldsymbol{V} \in \mathbb{R}^{p\times(p-d)}$ is the orthogonal complement of $\boldsymbol{U}$ in $\mathbb{R}^p$. Then $\boldsymbol{\Psi}_k$ is assumed to respect:

$$\boldsymbol{U}^\top \boldsymbol{\Psi}_k \boldsymbol{U} = \mathbf{0}_{d\times d}, \qquad \boldsymbol{V}^\top \boldsymbol{\Psi}_k \boldsymbol{V} = \beta_k \boldsymbol{I}_{p-d}, \qquad \boldsymbol{V}^\top \boldsymbol{\Psi}_k \boldsymbol{U} = \mathbf{0}_{(p-d)\times d}.$$

These constraints amount to say that the covariance matrix $\boldsymbol{S}_k$ is block diagonal after being rotated by $\boldsymbol{D}$. In other terms, writing $\boldsymbol{\Delta}_k = \boldsymbol{D}^\top \boldsymbol{S}_k \boldsymbol{D}$, the DLM model assumes that:

$$\boldsymbol{\Delta}_k = \operatorname{diag}(\boldsymbol{\Sigma}_k, \beta_k \boldsymbol{I}_{p-d}) = \begin{pmatrix} \boldsymbol{\Sigma}_k & \mathbf{0} \\ \mathbf{0} & \beta_k \boldsymbol{I}_{p-d} \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}a\\b\end{matrix}}\right\} d \le K-1 \\ \left.\vphantom{\begin{matrix}a\\b\end{matrix}}\right\} (p-d) \end{matrix}.$$

This hypothesis implies that the discriminative subspace contains the relevant clustering information, while the noise variance lies in the orthogonal directions. The model parameters are $\boldsymbol{\vartheta} = (\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{U})$.

4.2.2 A Bayesian formulation and the family of sub-models

We propose a Bayesian extension of the DLM model where a prior is put on $\boldsymbol{\mu}_k$ as in the standard Bayesian Gaussian mixture models:

$$p(\boldsymbol{\mu} \mid \boldsymbol{\nu}, \lambda) = \prod_{k=1}^{K} \mathcal{N}_d(\boldsymbol{\mu}_k \mid \boldsymbol{\nu}, \lambda \boldsymbol{I}_d). \tag{4.3}$$

Here λ is a hyper-parameter controlling the spreading of the $\boldsymbol{\mu}_k$'s in the latent space. The rest of the model and assumptions is unchanged and we refer to this Bayesian version as $\text{BDLM}_{[\boldsymbol{\Sigma}_k \beta_k]}$, which is represented as a graphical model in Figure 4.2. The set of parameters is then $\boldsymbol{\vartheta} = (\boldsymbol{\pi}, \boldsymbol{\Sigma}, \boldsymbol{U}, \boldsymbol{\beta})$ of dimension

$$\gamma = K - 1 + K\frac{d(d+1)}{2} + pd - \frac{d(d+1)}{2} + K,$$

and Section 4.3 discusses inference and clustering.

Considering specific constraints on the matrix $\boldsymbol{\Delta}_k$, one can derive a family of sub-models for the BDLM as in the original DLM. Akin to the spectral constraints of Banfield and Raftery (1993) described in Section 2.4.1, one can assume a combination of hypotheses on the structure of the latent space covariance $\boldsymbol{\Sigma}_k$ and the noise covariance $\boldsymbol{\Psi}_k$, for a total of 12 models. First homoscedasticity constraints of the type $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$ or $\boldsymbol{\Psi}_k = \boldsymbol{\Psi}$ may be considered, denoted as $\text{BDLM}_{[\boldsymbol{\Sigma}\beta_k]}$, $\text{BDLM}_{[\boldsymbol{\Sigma}_k\beta]}$, and $\text{BDLM}_{[\boldsymbol{\Sigma}\beta]}$ where the subscript k denotes heteroscedasticity of the concerned parameter. Moreover, the covariance $\boldsymbol{\Sigma}_k$ can be further assumed to be diagonal $\operatorname{diag}(\alpha_{k1}^2, \ldots, \alpha_{kd}^2)$ leaving 4 possibilities depending on the homoscedasticity hypothesis denoted as $\text{BDLM}_{[\alpha_{kh}\beta_k]}$, $\text{BDLM}_{[\alpha_h\beta_k]}$, $\text{BDLM}_{[\alpha_{kh}\beta]}$ and

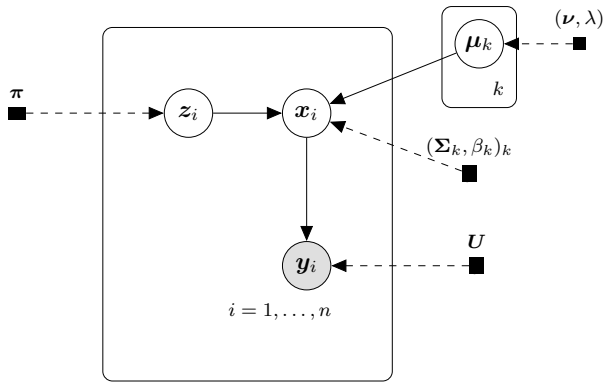


Figure 4.2: Graphical model representation of the Bayesian discriminative latent mixture model.

$\text{BDLM}_{[\alpha_h\beta]}$. Finally, the latent covariance may be considered isotropic $\boldsymbol{\Sigma}_k = \alpha_k^2 \boldsymbol{I}_d$, and the corresponding 4 sub-models are noted as $\text{BDLM}_{[\alpha_k\beta_k]}$, $\text{BDLM}_{[\alpha\beta_k]}$, $\text{BDLM}_{[\alpha_k\beta]}$ and $\text{BDLM}_{[\alpha\beta]}$. A comprehensive summary of these sub-models, along with their number of free parameters is given in Table 4.1.

Model	Number of free parameters γ	$p = 100$, $K = 4$
Full-GMM	$K - 1 + Kp + K\frac{p(p+1)}{2}$	20603
Sphe-GMM	$K - 1 + 2Kp$	803
$\text{BDLM}_{[\boldsymbol{\Sigma}_k\beta_k]}$	$K - 1 + K\frac{d(d+1)}{2} + pd - \frac{d(d+1)}{2} + K$	325
$\text{BDLM}_{[\boldsymbol{\Sigma}_k\beta]}$	$K - 1 + K\frac{d(d+1)}{2} + pd - \frac{d(d+1)}{2} + 1$	322
$\text{BDLM}_{[\boldsymbol{\Sigma}\beta_k]}$	$K - 1 + \frac{d(d+1)}{2} + pd - \frac{d(d+1)}{2} + K$	307
$\text{BDLM}_{[\boldsymbol{\Sigma}\beta]}$	$K - 1 + \frac{d(d+1)}{2} + pd - \frac{d(d+1)}{2} + 1$	304
$\text{BDLM}_{[\alpha_{kh}\beta_k]}$	$K - 1 + Kd + pd - \frac{d(d+1)}{2} + K$	313
$\text{BDLM}_{[\alpha_{kh}\beta]}$	$K - 1 + Kd + pd - \frac{d(d+1)}{2} + 1$	310
$\text{BDLM}_{[\alpha_h\beta_k]}$	$K - 1 + d + pd - \frac{d(d+1)}{2} + K$	304
$\text{BDLM}_{[\alpha_h\beta]}$	$K - 1 + d + pd - \frac{d(d+1)}{2} + 1$	301
$\text{BDLM}_{[\alpha_k\beta_k]}$	$K - 1 + K + pd - \frac{d(d+1)}{2} + K$	305
$\text{BDLM}_{[\alpha_k\beta]}$	$K - 1 + K + pd - \frac{d(d+1)}{2} + 1$	302
$\text{BDLM}_{[\alpha\beta_k]}$	$K - 1 + 1 + pd - \frac{d(d+1)}{2} + K$	302
$\text{BDLM}_{[\alpha\beta]}$	$K - 1 + 1 + pd - \frac{d(d+1)}{2} + 1$	299

Table 4.1: The BDLM family of sub-models with their associated number of free parameters, along with related model-based subspace clustering models. The dimension of the latent space is fixed to $d = K - 1$ in the last examples. See also Table 2.1 on page 41 for a comparison with related common subspace clustering methods.

4.2.3 Link with parsimonious Gaussian models

As an extension of the DLM model, the BDLM model inherits its connections to the other model-based subspace clustering models described in Section 2.4.1. Indeed, they share the same *global* linear Gaussian model formulations, with a common loading matrix $\boldsymbol{U}$. The MCFA model (Baek and McLachlan 2008) is the closest to the DLM model although the assumption on the nature of the subspace and the noise matrix $\boldsymbol{\Psi}_k$ are different. This model also shares deep ties with the heteroscedastic factor mixture analysis (HFMA Montanari and Viroli 2010) where the latent scores again follow a mixture of Gaussian distributions and a common loading matrix. However, the linear transformation $\boldsymbol{U}$ is not constrained to be orthonormal here, and other identifiability constraints are put on $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ such that the scores are standardized in the latent space. The PGMM family of McNicholas and Murphy (2008) also contains 4 constrained MFA models CUU, CCU, CUC and CCC, with common loadings across clusters $\boldsymbol{S}_k = \boldsymbol{W}\boldsymbol{W}^\top + \boldsymbol{\Psi}_k$. Note that these models do not put the mixture in the latent space but in the observation one, which does not allow putting further constraints on $\boldsymbol{\Sigma}_k$. Finally, Bouveyron et al. (2007a) proposed a family of 28 constrained Gaussian mixtures for high-dimensional data, among which 14 share common orientation matrices.

Related to our approach, a fully Bayesian extension of the MCFA model was proposed in Wei and Li (2013), putting a Dirichlet prior on the mixture proportions $\boldsymbol{\pi}$, a standard Gaussian on each column of $\boldsymbol{U}$ and a Gaussian-inverse-Wishart prior on $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. However, the marginal distribution of $\boldsymbol{x}_i$ in this model is now a mixture of Student t-distribution which differs from the mixture of Gaussian in our model. In addition, the factor loading matrix is no longer assumed to be column-orthonormal, and as in MCFA the subspace spanned by $\boldsymbol{U}$ is not considered discriminant. The authors proposed to rely on a variational Bayes EM algorithm to approximate the posterior distribution of the parameters.

4.3 Clustering with the Bayesian Fisher EM algorithm

In the following we propose a clustering algorithm based on the joint maximization of the Fisher criterion and the observed-data likelihood. The latter is intractable, and contrary to the DLM, the posterior distribution of the latent variables $(\boldsymbol{Z}, \boldsymbol{\mu})$ is intractable, thus we propose to rely on a variational approximation. After having derived the specific form of the CAVI updates for this model, we give the proper formulas for the M-step. Then, following Bouveyron and Brunet (2012a), we propose to choose, at each iteration, $\boldsymbol{U}$ as the best current discriminative subspace maximizing the Fisher criterion. Thus, the proposed clustering algorithm for BDLM is named Bayesian Fisher EM (BFEM) and alternates between 3 steps:

- The VE-step which finds an approximation of the posterior $p(\boldsymbol{Z}, \boldsymbol{\mu} \mid \boldsymbol{Y}; \boldsymbol{\vartheta})$ in the mean-field family.
- The M-step where the mixture parameters are estimated in the latent space by maximizing the variational lower bound.
- The F-step where $\boldsymbol{U}$ is chosen to maximize the current variational Fisher criterion.

4.3.1 Variational approximation

Similarly to the Bayesian formulation of standard GMM, the observed-data likelihood is no longer tractable. Indeed, the latter is written as:

$$p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) = \int_{\boldsymbol{\mu}} p(\boldsymbol{\mu}) \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \, \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k) \, \mathrm{d}\boldsymbol{\mu}. \tag{4.4}$$

Unfortunately, each $\boldsymbol{\mu}_k$ now appears in all n factors of the integrand, thus the integral does not reduce to products and sums of d-dimensional trivial integrals over $\boldsymbol{\mu}_k$. Another way of seeing the difficulty is to swap the integrals over $\boldsymbol{Z}$ and $\boldsymbol{\mu}$, leaving:

$$p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) = \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}) \prod_{k=1}^{K} \int_{\boldsymbol{\mu}_k} p(\boldsymbol{\mu}_k) \prod_{i : z_{ik}=1} \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k) \, \mathrm{d}\boldsymbol{\mu}_k. \tag{4.5}$$

Now each integral may be computed thanks to Gaussian conjugacy. However, there are K^n possible configurations to sum over, which is not computationally feasible.

Thus, the posterior $p(\boldsymbol{Z}, \boldsymbol{\mu} \mid \boldsymbol{Y})$ is not tractable either, and is only known up to its normalizing constant in Equation (4.4), which prevents from calculating its moments. This fact is well known in Bayesian treatment of mixtures, and leads to either MCMC algorithms or approximate inference (Fruhwirth-Schnatter et al. 2019, section 5.2). Taking the notations of Section 2.2, with $\boldsymbol{\eta} = (\boldsymbol{\mu}, \boldsymbol{Z})$, we posit the following mean-field approximation:

$$q(\boldsymbol{\mu}, \boldsymbol{Z}) = \prod_{k=1}^{K} q(\boldsymbol{\mu}_k) \prod_{i=1}^{n} q(\boldsymbol{z}_i).$$

Then, we recall the variational lower bound:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) \geq \mathcal{J}(q, \boldsymbol{\vartheta}) = \mathbb{E}_q \left[\log p(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\mu} \mid \boldsymbol{\vartheta}) \right] + \mathrm{H}(q).$$

The following propositions give the optimal form of the CAVI updates for each individual distribution. Once again, the distribution $q^\star(\boldsymbol{z}_i)$ happens to be multinomial at the optimum, while $q^\star(\boldsymbol{\mu}_k)$ is Gaussian. In the VE-step, the updates of Equations (4.6) and (4.7) are cycled over until a local maximum of $\mathcal{J}(q)$ is reached.

Proposition 4.1 (Proof in Appendix B.1 on page 139)**.** *The coordinate update for the variational distribution $q(\boldsymbol{z}_i)$ is*

$$q^\star(\boldsymbol{z}_i) = \mathcal{M}_K(\boldsymbol{z}_i \mid 1, \boldsymbol{\tau}_i), \tag{4.6}$$

with $\forall i, k$,

$$\tau_{ik} \propto \pi_k \exp\left\{ \mathbb{E}_{\boldsymbol{\mu}_k} \left[\log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\mu_k, \boldsymbol{S}_k) \right] \right\}.$$

Proposition 4.2 (Proof in Appendix B.2 on page 139)**.** *The coordinate update for the variational distribution $q(\boldsymbol{\mu}_k)$ is*

$$q^\star(\boldsymbol{\mu}_k) = \mathcal{N}_d(\boldsymbol{\mu}_k \mid \tilde{\boldsymbol{\mu}}_k, \tilde{\mathbf{M}}_k), \tag{4.7}$$

with $\forall k$,

$$\begin{aligned}
\tilde{n}_k &= \sum_i \tau_{ik}, \\
\tilde{\mathbf{M}}_k &= \left(\lambda^{-1}\boldsymbol{I}_d + \tilde{n}_k \boldsymbol{\Sigma}_k^{-1}\right)^{-1}, \\
\tilde{\boldsymbol{\mu}}_k &= \boldsymbol{\nu} + \tilde{\mathbf{M}}_k \boldsymbol{\Sigma}_k^{-1} \left(\boldsymbol{U}^\top (\sum_i \tau_{ik} \boldsymbol{y}_i) - \tilde{n}_k \boldsymbol{\nu} \right).
\end{aligned}$$

Finally, note that the expression of τ_{ik} in Equation (4.6) involves an expectation in the observation space that may be reworked in order to avoid inverting the $p \times p$ matrix $\boldsymbol{S}_k$. This is done by taking advantage of the specific block structure of the latter, hence only relying on the inverse of the $d \times d$ matrix $\boldsymbol{\Sigma}_k$:

$$\begin{aligned}
\tau_{ik} &\propto \pi_k \exp\left\{ \mathbb{E}_{\boldsymbol{\mu}_k}\left[\log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\mu_k, \boldsymbol{S}_k)\right] \right\}, \\
&\propto \pi_k \exp\left\{ -\frac{1}{2}\Big(p\log(2\pi) + \log|\boldsymbol{S}_k| + (\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k)^\top \boldsymbol{U}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top(\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k) \right. \right. \\
&\qquad \left. \left. + \frac{1}{\beta_k}(\|\boldsymbol{y}_i\|_2 - \|\boldsymbol{U}^\top \boldsymbol{y}_i\|_2) + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k \boldsymbol{\Sigma}_k^{-1}\right] \Big) \right\}.
\end{aligned} \tag{4.8}$$

4.3.2 The M-step

In the M-step, the bound is maximized with respect to the latent space parameters $(\boldsymbol{\Sigma}, \boldsymbol{\beta})$ and the mixture proportions $\boldsymbol{\pi}$. Note that $\boldsymbol{U}$ is treated as a fixed, distinct parameter here, which will be dealt with in the next section. The following proposition gives the form of the lower bound as a function of $\boldsymbol{\vartheta}$.

Proposition 4.3 (Proof in Appendix B.3 on page 141)**.** *In model* $\text{BDLM}_{[\boldsymbol{\Sigma}_k \beta_k]}$*, the variational lower bound as a function of* $\boldsymbol{\vartheta}$ *may be written as:*

$$\begin{aligned}
\mathcal{J}(\boldsymbol{\vartheta}) = \text{const} - \frac{1}{2}\sum_{k=1}^{K} \tilde{n}_k \Bigg\{ & -2\log(\pi_k) + \log|\boldsymbol{\Sigma}_k| + (p-d)\log(\beta_k) \\
& + \operatorname{Tr}\left[\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right] + \frac{1}{\beta_k}\left(\operatorname{Tr}\left[\hat{\boldsymbol{C}}_k\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right]\right) \Bigg\},
\end{aligned} \tag{4.9}$$

where

$$\hat{\boldsymbol{C}}_k = \frac{1}{\tilde{n}_k}\sum_{i=1}^{n} \tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k)(\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k)^\top + \boldsymbol{U}\tilde{\mathbf{M}}_k\boldsymbol{U}^\top.$$

At iteration (t), in the M-step, the mixture proportions are estimated classically as in other mixture models:

$$\hat{\pi}_k^{(t)} = \frac{\tilde{n}_k^{(t)}}{n}. \tag{4.10}$$

The remaining parameters $(\boldsymbol{\Sigma}_k, \beta_k)$ depend on the chosen sub-model and the following proposition details the estimates corresponding to the each of the 12 sub-models.

Proposition 4.4 (Proof in Appendix B.4 on page 143)**.** *The M-step estimates for* $\boldsymbol{\Sigma}_k$ *and* β_k *at iteration* (t) *are:*

- *Model* $\text{BDLM}_{[\boldsymbol{\Sigma}_k \beta_k]}$*:*

$$\hat{\boldsymbol{\Sigma}}_k^{(t)} = \boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}, \qquad \hat{\beta}_k^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}_k^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.11}$$

- *Model* $\text{BDLM}_{[\boldsymbol{\Sigma}_k \beta]}$*:*

$$\hat{\boldsymbol{\Sigma}}_k^{(t)} = \boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}, \qquad \hat{\beta}^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.12}$$

- *Model* $\text{BDLM}_{[\boldsymbol{\Sigma} \beta_k]}$*:*

$$\hat{\boldsymbol{\Sigma}}^{(t)} = \boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}, \qquad \hat{\beta}_k^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}_k^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.13}$$

- *Model* $\text{BDLM}_{[\boldsymbol{\Sigma} \beta]}$*:*

$$\hat{\boldsymbol{\Sigma}}^{(t)} = \boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}, \qquad \hat{\beta}^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.14}$$

- *Model* $\text{BDLM}_{[\alpha_{kh} \beta_k]}$*:*

$$\hat{\alpha}_{kh}^{(t)} = \boldsymbol{u}_h^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{u}_h, \qquad \hat{\beta}_k^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}_k^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.15}$$

- *Model* $\text{BDLM}_{[\alpha_{kh} \beta]}$*:*

$$\hat{\alpha}_{kh}^{(t)} = \boldsymbol{u}_h^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{u}_h, \qquad \hat{\beta}^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.16}$$

- *Model* $\text{BDLM}_{[\alpha_h \beta_k]}$*:*

$$\hat{\alpha}_h^{(t)} = \boldsymbol{u}_h^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{u}_h, \qquad \hat{\beta}_k^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}_k^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.17}$$

- *Model* $\text{BDLM}_{[\alpha_h \beta]}$*:*

$$\hat{\alpha}_h^{(t)} = \boldsymbol{u}_h^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{u}_h, \qquad \hat{\beta}^{(t)} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}^{(t)}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.18}$$

- *Model* $\text{BDLM}_{[\alpha_k\beta_k]}$*:*

$$\hat{\alpha}_k^{(t)} = \frac{1}{d}\operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right], \qquad \hat{\beta}_k^{(t)} = \frac{\operatorname{Tr}\left[\hat{\boldsymbol{C}}_k^{(t)}\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.19}$$

- *Model* $\text{BDLM}_{[\alpha_k\beta]}$*:*

$$\hat{\alpha}_k^{(t)} = \frac{1}{d}\operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right], \qquad \hat{\beta}^{(t)} = \frac{\operatorname{Tr}\left[\hat{\boldsymbol{C}}^{(t)}\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.20}$$

- *Model* $\text{BDLM}_{[\alpha\beta_k]}$*:*

$$\hat{\alpha}^{(t)} = \frac{1}{d}\operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right], \qquad \hat{\beta}_k^{(t)} = \frac{\operatorname{Tr}\left[\hat{\boldsymbol{C}}_k^{(t)}\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.21}$$

- *Model* $\text{BDLM}_{[\alpha\beta]}$*:*

$$\hat{\alpha}^{(t)} = \frac{1}{d}\operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right], \qquad \hat{\beta}^{(t)} = \frac{\operatorname{Tr}\left[\hat{\boldsymbol{C}}^{(t)}\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}^{(t)} \boldsymbol{U}\right]}{p-d}. \tag{4.22}$$

Here, $\boldsymbol{u}_h$ denotes the h-th column of $\boldsymbol{U}$ which is computed in the F-step at iteration (t) and:

$$\hat{\boldsymbol{C}}_k^{(t)} = \frac{1}{\tilde{n}_k^{(t)}} \sum_{i=1}^{n} \tau_{ik}^{(t)} (\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k^{(t)})(\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k^{(t)})^\top + \boldsymbol{U}\tilde{\mathbf{M}}_k^{(t)}\boldsymbol{U}^\top, \tag{4.23}$$

$$\hat{\boldsymbol{C}}^{(t)} = \frac{1}{n} \sum_{k=1}^{K} \tilde{n}_k^{(t)} \hat{\boldsymbol{C}}_k^{(t)}. \tag{4.24}$$

4.3.3 The Fisher step

As explained above, the subspace $\boldsymbol{U}$ is supposed to be discriminant in the sense of the Fisher criterion. The partition $\boldsymbol{Z}$ being unknown, the scatter matrices in Equation (4.2) cannot be formed. Following Bouveyron and Brunet (2012a) we propose to replace them by the soft within and between-class scatter matrices:

$$\tilde{\boldsymbol{S}}_W^{(t+1)} = \frac{1}{n} \sum_{k=1}^{K} \frac{1}{\tilde{n}_k^{(t)}} \sum_{i=1}^{n} \tau_{ik}^{(t)} \left(\boldsymbol{y}_i - \tilde{\boldsymbol{m}}_k^{(t)}\right) \left(\boldsymbol{y}_i - \tilde{\boldsymbol{m}}_k^{(t)}\right)^\top,$$

$$\tilde{\boldsymbol{S}}_B^{(t+1)} = \frac{1}{n} \sum_{k=1}^{K} \tilde{n}_k^{(t)} \left(\tilde{\boldsymbol{m}}_k^{(t)} - \bar{y}\right) \left(\tilde{\boldsymbol{m}}_k^{(t)} - \bar{y}\right)^\top,$$

$$\tilde{\boldsymbol{m}}_k^{(t)} = \frac{1}{\tilde{n}_k^{(t)}} \sum_{i=1}^{n} \tau_{ik}^{(t)} \boldsymbol{y}_i.$$

Note that these matrices only involve the variational distribution of $\boldsymbol{Z}$, although the latter also depends on $q^{(t)}(\boldsymbol{\mu})$ through the fixed point algorithm of the VE-step. Moreover, at any

iteration (t) we recover the classical identity of linear discriminant analysis $\boldsymbol{S}_T = \boldsymbol{S}_W^{(t)} + \boldsymbol{S}_B^{(t)}$, where $\boldsymbol{S}_T$ is sample covariance matrix $\boldsymbol{S}_T = (1/n)\sum_{i=1}^n (\boldsymbol{y}_i - \bar{y})(\boldsymbol{y}_i - \bar{y})^\top$ which does not depend on the clustering and is constant throughout the algorithm.

Then $\boldsymbol{U}$ is supposed to maximize the following criterion:

$$\boldsymbol{U}^{(t)} = \underset{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d}{\arg\max}\, \mathrm{F}(\boldsymbol{U}) = \mathrm{Tr}\left[(\boldsymbol{U}^\top \boldsymbol{S}_T \boldsymbol{U})^{-1} \boldsymbol{U}^\top \tilde{\boldsymbol{S}}_B^{(t)} \boldsymbol{U}\right]. \tag{4.25}$$

This criterion is slightly different from the one in Equation (4.2) since $\tilde{\boldsymbol{S}}_W^{(t)}$ has been replaced by $\boldsymbol{S}_T$. Although the solutions are not exactly the same, this is justified using the identity above since the problems of minimizing $\mathrm{Tr}[\boldsymbol{U}^\top \boldsymbol{S}_W \boldsymbol{U}]$ or $\mathrm{Tr}[\boldsymbol{U}^\top \boldsymbol{S}_T \boldsymbol{U}]$ are the same (Fukunaga 1990, chap. 10). It is often used in practice (Ye 2005), and computationally efficient in this case since $\boldsymbol{S}_T$ and its inverse need to be computed only once at the beginning of the algorithm.

Without the orthonormality constraints, the problem in Equation (4.25) is directly solved by taking the leading d eigenvectors of the generalized eigenvalue problem $\tilde{\boldsymbol{S}}_B^{(t)} \boldsymbol{u}_h = \gamma_h \boldsymbol{S}_T \boldsymbol{u}_h$ (Ghojogh et al. 2019). If $\boldsymbol{S}_T$ is invertible this can be done by computing the d leading eigenvectors of $\boldsymbol{S}_T^{-1} \tilde{\boldsymbol{S}}_B^{(t)}$. However, since $\boldsymbol{S}_T^{-1} \tilde{\boldsymbol{S}}_B$ is not necessarily symmetric, the solution is not orthonormal with respect to the regular scalar product, but rather verifies $\boldsymbol{U}^\top \boldsymbol{S}_T \boldsymbol{U} = \boldsymbol{I}_d$. Unfortunately, there is no direct solution for the problem of Equation (4.25) with the constraint $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d$. In the supervised context, algorithms have been derived to solve this problem which is called orthogonal Linear Discriminant Analysis (OLDA). Foley and Sammon (1975) proposed an iterative algorithm to successively find $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d$ in the 2-class problem. It was later generalized for arbitrary values of K by Okada and Tomita (1985), and coined the orthonormal discriminant vectors (ODV) by Hamamoto et al. (1991). Note that simultaneous algorithms also exist to optimize with respect to $\boldsymbol{U}$, based on successive eigen and QR-decompositions of carefully designed matrices (see. Ye 2005; Lu et al. 2016).

Relying on the ODV method, Bouveyron and Brunet (2012a) proposed an iterative algorithm starting from $\boldsymbol{u}_1$, the leading eigenvector of $\boldsymbol{S}_T^{-1} \tilde{\boldsymbol{S}}_B^{(t)}$, and greedily maximizing the criterion by computing the r-th direction as the solution of the unconstrained problem in the orthogonal of the current subspace $\mathcal{B}_{r-1} = \mathrm{vect}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{r-1})$. An orthogonal basis $V_r = (\boldsymbol{v}_r, \ldots, \boldsymbol{v}_p)$ of $\mathcal{B}_{r-1}^\perp$ can be found by the Gram-Schmidt procedure:

$$\boldsymbol{v}_l = \alpha_l (\boldsymbol{I}_p - \sum_{l'=1}^{l-1} \boldsymbol{v}_{l'} \boldsymbol{v}_{l'}^\top) \psi_l, \quad \forall l = r, \ldots, p.$$

where $\boldsymbol{v}_l = \boldsymbol{u}_l$ for $l = 1, \ldots, r-1$, α_l is a normalization constant such that $\|\boldsymbol{v}_l\|_2 = 1$ and ψ_l are linearly independent vectors of $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{r-1}$. Then the matrix $\boldsymbol{P}_r = (\boldsymbol{v}_r, \ldots, \boldsymbol{v}_p)$ is used to project the scatter matrix in the orthogonal subspace $\mathcal{B}_{r-1}^\perp$:

$$\begin{aligned} \boldsymbol{S}_{Tr} &= \boldsymbol{P}_r^\top \boldsymbol{S}_T \boldsymbol{P}_r, \\ \tilde{\boldsymbol{S}}_{Br}^{(t)} &= \boldsymbol{P}_r^\top \tilde{\boldsymbol{S}}_B^{(t)} \boldsymbol{P}_r. \end{aligned}$$

Finally, the leading eigenvector $\boldsymbol{a}_r$ of the generalized eigenvalue problem $\tilde{\boldsymbol{S}}_{Br}^{(t)} \boldsymbol{a}_r = \gamma_r \boldsymbol{S}_{Tr} \boldsymbol{a}_r$ is computed, and the r-th discriminant vector is chosen as

$$\boldsymbol{u}_r = \frac{\boldsymbol{P}_r \boldsymbol{a}_r}{\|\boldsymbol{a}_r\|_2}. \tag{4.26}$$

Thus, $\boldsymbol{u}_r$ meet the constraints $\boldsymbol{u}_r^\top \boldsymbol{u}_h = 0, \forall h < r$. This iterative procedure is repeated until $r = d$ discriminant vectors are found.

4.3.4 Hyper-parameters estimation

The hyper-parameters $(\lambda, \boldsymbol{\nu})$ may be set by the user and kept fixed during the whole procedure. For instance, when the data is centered, $\bar{y} = 0$, then $\bar{x} = \boldsymbol{U}^\top \bar{y} = 0$ thus one could set $\boldsymbol{\nu} = 0$. However, λ controls the variance of $\boldsymbol{\mu}_k$ and setting it by hand can lead to poor performances. On the one hand, a too small value would not allow the space to be discriminant. On the other hand, when $\lambda \to +\infty$, the prior becomes non-informative. A quick asymptotic analysis of the variational distribution $q(\boldsymbol{\mu}_k)$ of Proposition 4.2 confirms this as:

$$\tilde{\mathbf{M}}_k \xrightarrow[\lambda \to +\infty]{} \frac{1}{\tilde{n}_k} \boldsymbol{\Sigma}_k, \qquad \tilde{\boldsymbol{\mu}}_k \xrightarrow[\lambda \to +\infty]{} \boldsymbol{\Sigma}_k \boldsymbol{\Sigma}_k^{-1} \frac{1}{\tilde{n}_k} \textstyle\sum_i \tau_{ik} \boldsymbol{U}^\top \boldsymbol{y}_i = \hat{\boldsymbol{\mu}}_k^{DLM}.$$

Thus, the variational posterior mean becomes the maximum-likelihood estimate of $\boldsymbol{\mu}_k$ in the frequentist formulation DLM. Under the hypothesis that $\tilde{n}_k \to +\infty$ as $n \to +\infty$, the variational approximation of the posterior becomes a Dirac mass at $\hat{\mu}_k^{DLM}$. This fact is somewhat similar to the well-known behavior of the posterior in Bayesian formulations of ridge regression when the prior becomes vague (Bishop 2006, p. 153).

Here, we propose a parametric empirical Bayes approach (Morris 1983), using the variational bound as a surrogate for the type-II likelihood as it is commonly done in other well-known hierarchical Bayesian models (Blei et al. 2003; Airoldi et al. 2008). The following proposition gives the form of the empirical Bayes estimates $(\hat{\boldsymbol{\nu}}, \hat{\lambda})$ maximizing $\mathcal{J}(\boldsymbol{\nu}, \lambda) \approx \log p(\boldsymbol{Y} \mid \boldsymbol{\nu}, \lambda)$.

Proposition 4.5 (Proof in Appendix B.5 on page 146)**.** *The following updates maximize the variational lower bound with respect to* $(\boldsymbol{\nu}, \lambda)$*:*

$$\hat{\boldsymbol{\nu}} = \frac{\sum_{k=1}^K \tilde{\boldsymbol{\mu}}_k}{K}, \tag{4.27}$$

$$\hat{\lambda} = \frac{\sum_{k=1}^K \|\tilde{\boldsymbol{\mu}}_k - \hat{\boldsymbol{\nu}}\|_2^2 + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k\right]}{dK} \tag{4.28}$$

4.3.5 Stopping criterion and properties

Starting from a subspace $\boldsymbol{U}^{(0)}$, the BFEM algorithm iterates over the VE-step, M-step and F-step updates, in this order. The algorithm is described in Algorithm 3 and this section discusses initialization, convergence and useful properties of the algorithm. Let us begin by discussing the link to the original FEM algorithm.

LINK TO THE ORIGINAL FISHER EM The proposed algorithm is largely inspired by the original FEM algorithm. However, note that the M-step updates of Bouveyron and Brunet (2012a) use:

$$\tilde{\boldsymbol{C}}_k = \frac{1}{\tilde{n}_k} \sum_{i=1}^n \tau_{ik} (\boldsymbol{y}_i - \tilde{\boldsymbol{m}}_k)(\boldsymbol{y}_i - \tilde{\boldsymbol{m}}_k)^\top, \quad \text{with: } \tilde{\boldsymbol{m}}_k = \frac{1}{\tilde{n}_k} \sum_{i=1}^n \tau_{ik} \boldsymbol{y}_i. \tag{4.29}$$

The latter does not exactly correspond to the matrix $\hat{\boldsymbol{C}}_k$ of Propositions 4.3 and 4.4, since it uses $\tilde{\boldsymbol{m}}_k$ instead of $\hat{\boldsymbol{m}}_k = \boldsymbol{U}\boldsymbol{\mu}_k$. In particular, this has the consequence that the matrix $\tilde{\boldsymbol{C}}_k$ does not directly depend on $\boldsymbol{U}$, whereas it does in $\hat{\boldsymbol{C}}_k$. Therefore, our algorithm computes the true optimal updates in the M-step, while the FEM algorithm relies on the approximation $\hat{\boldsymbol{C}}_k \approx \tilde{\boldsymbol{C}}_k$.

CONVERGENCE AND STOPPING CRITERION Since the F-step does not maximize the variational bound with respect to $\boldsymbol{U}$, the latter is no longer monotonically increasing. This is also the case for the original FEM algorithm, hence we propose to rely on the same stopping criterion: Aitken's accelerated criterion (McLachlan and Krishnan 2007, p. 145). The latter was introduced as an acceleration method for EM when the sequence of likelihood is linearly converging. Here, we replace the likelihood sequence with the variational bound $\{\mathcal{J}^{(t)}\}_t$. Then, define the Aitken accelerated estimate of $\mathcal{J}^\star$ is defined for $t \geq 2$ as:

$$l_A^{(t+1)} = \mathcal{J}^{(t)} + \frac{1}{1 - c^{(t+1)}} \left(\mathcal{J}^{(t+1)} - \mathcal{J}^{(t)} \right), \quad \text{with: } c^{(t+1)} = \frac{\mathcal{J}^{(t+1)} - \mathcal{J}^{(t)}}{\mathcal{J}^{(t)} - \mathcal{J}^{(t-1)}}. \tag{4.30}$$

Then, the stopping criterion is defined as $|l_A^{(t+1)} - l_A^{(t)}| < \epsilon$, where ϵ is a user-defined tolerance parameter. Since there is no guarantee that the sequence $\{\mathcal{J}^{(t)}\}_t$ is increasing here, a maximum number of iterations is also provided by the user as an alternative stopping criterion, as it is always done in a VEM algorithm anyway.

Another possible stopping condition is the absolute change of the Fisher criterion of Equation (4.25) between two successive F-step: $|\,\mathrm{F}(\boldsymbol{U}^{(t+1)}) - \mathrm{F}(\boldsymbol{U}^{(t)})|\,/\,|\,\mathrm{F}(\boldsymbol{U}^{(t)})|$. The latter was shown to have good performance for clustering applications (Bouveyron and Brunet 2012b).

INITIALIZATION The BFEM algorithm needs an initial variational distribution $q^{(0)}$ defined by its starting variational parameters $(\boldsymbol{\tau}, \tilde{\boldsymbol{\mu}}, \tilde{\mathbf{M}})$, an initial set of parameters $(\boldsymbol{\Sigma}, \boldsymbol{\beta}, \boldsymbol{\pi})$ and an initial subspace $\boldsymbol{U}^{(0)}$. We recommend initializing by setting $\boldsymbol{\tau}^{(0)} = \boldsymbol{Z}^{(0)}$, a partition obtained by any suitable clustering algorithm, *e.g.* random or k-means partitions. Then, the matrix $\tilde{\boldsymbol{S}}_B^{(0)}$ can be formed to solve the problem in Equation (4.25), giving an initial $\boldsymbol{U}^{(0)}$. Next, the initial parameters $(\boldsymbol{\Sigma}, \boldsymbol{\beta}, \boldsymbol{\pi})$ are obtained by using the frequentist M-step of Bouveyron and Brunet (2012a) with $\boldsymbol{U}^{(0)}$. The remaining variational parameters $(\tilde{\boldsymbol{\mu}}_k, \tilde{\mathbf{M}}_k)$ can then be set using Proposition 4.2 with $\boldsymbol{\tau}^{(0)}$, $\boldsymbol{U}^{(0)}$ and $\boldsymbol{\vartheta}^{(0)}$. As for the hyper-parameters, we initialize $\boldsymbol{\nu}$ as $\boldsymbol{\nu}^{(0)} = (1/n) \sum_{i=1}^n {\boldsymbol{U}^{(0)}}^\top \boldsymbol{y}_i$, and set $\lambda^{(0)} = 10^3$ as a vague prior for the first iteration, which is refined by empirical Bayes estimation throughout the algorithm.

Naturally, as in every algorithm with non-convex objective, the procedure can fall into poor local maxima of the bound. Thus, we recommend several restart with different initialization. In the experiments of Section 4.4, we try several k-means initialization and take the one achieving the greatest variational lower bound.

COMPUTATIONAL COMPLEXITY The ODV method necessitates $d(p-1)$ Gram-Schmidt operations overall, and d generalized eigenvalue problems to solve, although one only needs to find the first leading eigenvectors which can be done efficiently (Ge et al. 2016). Since $d \leq K - 1$ is supposed to be small compared to p, this is not too computationally expensive. Moreover, this has to be compared to the computational cost of maximizing the lower bound of Proposition 4.3 with respect to $\boldsymbol{U}$ as in a traditional M-step. Indeed, there

is no closed-form solution for this problem and relying of gradient descent can rapidly become cumbersome since it necessitates relying on the steepest descent in the Stiefel manifold $St(p,d) = \{\boldsymbol{U} \in \mathbb{R}^{p\times d},\ \boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}_d\}$.

Still, Section 4.3.7 introduces an alternative Fisher criterion which only necessitates performing one singular value decomposition at each step (t). The empirical computing time of the BFEM algorithm and competing methods are shown in Figure 4.8 on page 92 for the experimental settings of Section 4.4.

Algorithm 3: Pseudo code of the BFEM algorithm

Data: $\boldsymbol{Y}$
Result: A clustering $\boldsymbol{Z}$ and a discriminative subspace $\boldsymbol{U}$
Input: K, $\boldsymbol{Z}^{(0)}$, ϵ_{VE}, ϵ_M, T_M, T_{VE}, F-procedure, $\lambda^{(0)}$

`// Initialization`
Set $\boldsymbol{\tau} \leftarrow \boldsymbol{Z}^{(0)}$
Compute $\boldsymbol{S}_T$, $\tilde{\boldsymbol{S}}_B^{(0)}$ and subspace $\boldsymbol{U}$ with F-procedure
Compute initial parameters $(\boldsymbol{\pi}, \boldsymbol{\Sigma}, \boldsymbol{\beta})$ with frequentist M-step
Compute variational parameters $(\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{M}})$ with Proposition 4.2
Set $\boldsymbol{\nu} \leftarrow \frac{1}{n}\sum_{i=1}^n \boldsymbol{U}^\top \boldsymbol{y}_i$ and $\lambda \leftarrow \lambda^{(0)}$
Set $L[0] \leftarrow \mathcal{J}(\boldsymbol{\vartheta}, q, \boldsymbol{\nu}, \lambda)$

`// Optimization`
for $t \leftarrow 1$ **to** T_M **do**

  `// F-step`
  Compute $\tilde{\boldsymbol{S}}_B^{(t)}$
  Update $\boldsymbol{U}$ with F-procedure

  `// VE-step (fixed point algorithm)`
  for $v \leftarrow 1$ **to** T_{VE} **do**
    Set temp $\leftarrow \mathcal{J}(q)$
    Update $\boldsymbol{\tau}$ using Proposition 4.1
    Update $(\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{M}})$ using Proposition 4.2
    if $|(\mathcal{J}(q) - \text{temp}) \,/\, \mathcal{J}(q)| < \epsilon_{VE}$ **then Break**;
  end

  `// M-step`
  Update $(\boldsymbol{\pi}, \boldsymbol{\Sigma}, \boldsymbol{\beta})$ with Equation (4.10) and Proposition 4.4

  `// Empirical Bayes`
  Update $(\boldsymbol{\nu}, \lambda)$ with Proposition 4.5

  `// Compute the ELBO`
  $L[t] \leftarrow \mathcal{J}(\boldsymbol{\vartheta}, q, \boldsymbol{\nu}, \lambda)$
  if $t \geq 2$ **then**
    $c \leftarrow \frac{L[t]-L[t-1]}{L[t-1]-L[t-2]}$
    $\text{aitken}[t] \leftarrow L[t-1] + \frac{1}{1-c}\big(L[t] - L[t-1]\big)$
    if $|\,\text{aitken}[t] - \text{aitken}[t-1]| < \epsilon_M$ **then Break**;
  end
end
Return q, $\boldsymbol{\vartheta} = (\boldsymbol{U}, \boldsymbol{\pi}, \boldsymbol{\Sigma}, \boldsymbol{\beta})$

4.3.6 Model selection

CHOOSING THE LATENT DIMENSION d As in the supervised case, the rank of $\tilde{\boldsymbol{S}}_B^{(t)}$ is at most $K-1$, hence $d \leq K-1$. We recommend setting it to $d = K-1$ for inference, as it is preferable to have redundant information than to lose discriminant directions. This presents the advantage to report the problem of selecting d to the one of selecting K.

Moreover, following Okada and Tomita (1985), the discriminant vectors found by the ODV procedure may be ordered according to the value of their 1-dimensional fisher criterion

$$\mathrm{F}(\boldsymbol{u}_r) = \frac{\boldsymbol{u}_r^\top \tilde{\boldsymbol{S}}_B^{(t)} \boldsymbol{u}_r}{\boldsymbol{u}_r^\top \boldsymbol{S}_T \boldsymbol{u}_r} = \gamma_r,$$

where γ_r is the largest eigenvalue of the r-th problem solved, leaving:

$$\mathrm{F}(\boldsymbol{u}_1) \geq \ldots \geq \mathrm{F}(\boldsymbol{u}_d). \tag{4.31}$$

Thus, for visualization purpose, one can choose $d = 2$ or $d = 3$, and project the data onto the corresponding subspace. Another solution, as is commonly done in PCA, is to show several combinations of discriminant axes, for instance with a matrix of 2-dimensional scatter plots. Naturally, the visualization quality is dependent on the clustering quality, thus a poor visualization can guide the user to restart analysis with a different initialization.

CHOOSING THE NUMBER OF CLUSTERS K In a clustering perspective, we propose to rely on the integrated classification likelihood described in Section 2.5.2 to choose K and the submodel. Recall that the criterion of Biernacki et al. (2000) is defined as:

$$\mathrm{ICL}_{BIC}(\mathcal{M}, K) = \log p(\boldsymbol{Y}, \hat{\boldsymbol{Z}} \mid \hat{\boldsymbol{\vartheta}}, \mathcal{M}, K) - \frac{\gamma_{\mathcal{M},K}}{2} \log(n), \tag{4.32}$$

where we take $\hat{\boldsymbol{\vartheta}}$ to be the parameter estimates at the end of BFEM. Although the marginal likelihood is intractable as explained in Section 4.3.1, the first term above is the classification likelihood which is tractable in the BDLM models. Actually, it can be computed with the variational lower bound $\mathcal{J}$, replacing τ_{ik} by $\hat{z}_{ik}$ in the formulas of Propositions 4.1 and 4.2. A detailed proof of this fact is given in Appendix B.6 on page 147.

4.3.7 An alternative Fisher criterion

The ODV method proposed to maximize the criterion in Equation (4.25) can be viewed as a greedy method, sequentially solving 1-dimensional problems $\boldsymbol{u}_r = \arg\max_{\boldsymbol{u}} \mathrm{F}(\boldsymbol{u})$ under the growing set of constraints $\{\boldsymbol{u}\top\boldsymbol{u}_h = 0, \forall h < r\}$. Such a method is not guaranteed to globally converge as emphasized in Hamamoto et al. (1991). Moreover, relying on the Gram-Schmidt procedure can sometimes lead to numerical instabilities in the BFEM algorithm. Thus, Bouveyron and Brunet (2011) proposed an alternative Fisher criterion, searching for the orthogonal projection matrix $\boldsymbol{U} \in \mathbb{R}^{p \times d}$ minimizing the following reconstruction error:

$$\boldsymbol{U}^{(t)} = \underset{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d}{\arg\min} \|\boldsymbol{S}_T^{-1} \tilde{\boldsymbol{S}}_B^{(t)} - \boldsymbol{U}\boldsymbol{U}^\top \boldsymbol{S}_T^{-1} \tilde{\boldsymbol{S}}_B^{(t)}\|_F^2 \tag{4.33}$$

This optimization problem has a somewhat PCA like flavor except the matrix we wish to reconstruct is not the original data $\boldsymbol{Y}$ but the measure of class separability: $\boldsymbol{S}_T^{-1} \tilde{\boldsymbol{S}}_B^{(t)}$. The classical results still holds, and the optimal $\boldsymbol{U}^{(t)}$ is given as the leading d left singular-vectors

of $\boldsymbol{S}_T^{-1}\tilde{\boldsymbol{S}}_B^{(t)} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}$. Note that, since the product of these two symmetric matrices is not symmetric, the singular value decomposition is different from its spectral decomposition.

This modified F-step can be used to replace the ODV procedure at step (t). Thus, one only has to perform a partial singular-value decomposition (SVD) on $\boldsymbol{S}_T^{-1}\tilde{\boldsymbol{S}}_B^{(t)}$ at each step. Since $\boldsymbol{S}_T^{-1}$ is computed only once, this is particularly efficient.

4.4 Numerical experiments

This section compares the performance of different subspace clustering on simulated and classical real data benchmarks. We considered 6 different algorithms:

1. The proposed BFEM algorithm, with the ODV procedure described in Section 4.3.3 for the F-step. The results are also displayed for the alternative Fisher criterion of Section 4.3.7, which we refer to as the SVD procedure,

2. The Fisher EM algorithm of Bouveyron and Brunet (2012a), implemented in the **FisherEM** R Package (Bouveyron et al. 2020a). Again, we also show results for the F-step using the ODV procedure as well as the SVD,

3. The EM algorithm for the PGMM of McNicholas and Murphy (2008) with model CCU, corresponding to the low-rank constraint $\boldsymbol{S}_k = \boldsymbol{W}\boldsymbol{W} + \boldsymbol{\Psi}$ of the covariance matrix. An implementation is available in the eponymous R package **pgmm** (McNicholas et al. 2019),

4. The EM algorithm for MCFA of Baek et al. (2009) and implemented in the **EMMIXmfa** R pckage (Rathnayake et al. 2019),

5. The EM algorithm for the HDDC model of Bouveyron et al. (2007a) with model $[\cdot QD]$, so that the learned subspaces are common, as in the other methods. We used the implementation available in the **HDclassif** R package (Bergé et al. 2019),

6. A k-means baseline.

A total of 8 distinct algorithms are tested, since both models BFEM and FEM may have two distinct F-step procedures. In Sections 4.4.2 and 4.4.3, we use colors to distinguish between models, and line marker types to differentiate between the ODV and SVD method. For the sake of readability, we do not show the results the HFMA model of Montanari and Viroli (2010) in the following figures since it did not perform well on our experimental settings. This might be due to the different constraints on the subspace means and covariance, making it more distant to the BDLM model than other subspace clustering methods.

Throughout the rest of this section, unless stated otherwise, each method has the same 10 restarts consisting of 10 different k-means results. The one achieving the greatest likelihood is kept, and the clustering is done with a MAP estimate over the posterior of $\boldsymbol{Z}$. The maximum of iterations is set to 100 everywhere and the same absolute tolerance of 10^{-6} is used. The fixed point algorithm in the VE-step has a tolerance of 10^{-6} but a maximum number of iterations set to 3.

Concerning the choice of d, it is set to $K-1$ for both FEM and BFEM. The HDDC models have an internal heuristic to choose the best intrinsic dimension d_k of each cluster, and we use the BIC for the MCFA and PGMM as suggested in their original papers. The clustering results are reported using the Adjusted Rand Index (ARI, Hubert and Arabie 1985), a label independent measure of statistical similarity between two partitions. An ARI

of 0 means that the two partitions are statistically independent, while identical partitions (up to label switching) give an ARI of 1. Hence, the higher the ARI, the better.

4.4.1 An introductory example

In order to illustrate the interest of discriminative subspaces, we begin by the setting of Chang (1983) discussed in the introduction and Figure 4.1. There are $n = 300$ observations and $K = 2$ clusters in the data, defined as follows:

$$\boldsymbol{y}_i = -0.5r + r\mathbb{1}_{\{z_{ik}=1\}} + \mathcal{N}_p(\mathbf{0}, \boldsymbol{S}),$$

with $\forall j = 1, \ldots, 15$:

$$r_j = 0.95 - 0.05j,$$
$$\boldsymbol{S}_{jj} = 1 \text{ and } \forall j' \neq j, \boldsymbol{S}_{jj'} = -0.13 f_j f_{j'} \text{ with } f_j = \begin{cases} -0.9 & j \leq 8 \\ 0.5 & j > 8 \end{cases}.$$

Thus, it is a 2 component Gaussian mixture in dimension $p = 15$, with $\boldsymbol{m}_1 = -0.5r$ and $\boldsymbol{m}_2 = 0.5r$ and homoscedastic covariance $\boldsymbol{S}_1 = \boldsymbol{S}_2 = \boldsymbol{S}$. We emphasize that this simulation is not favoring any of the tested methods, except maybe for the standard GMM since the simulation is according to this model.

We ran each method with the true number of clusters $K = 2$, and used model selection for the choice of d for the concerned methods. The average results over 100 simulated datasets are represented for each method in Table 4.2. We do not distinguish the ODV and SVD methods here, since they lead to the same results on this simple example. One can see that the proposed discriminative subspace approach yields a better clustering in this setting, with a slight advantage over the frequentist version. In particular, extensions of pPCA like MCFA or PGMM do not allow to recover the correct partitions. This highlights the interest of discriminative subspaces even in different scenarios. The HDDC algorithm exhibit the same performance as BFEM. However, we point out that it selects intrinsic dimensions $d_k = 14$ to do so, which are the maximum values in this model. In contrast, BFEM works with $d = 1$ enabling to visualize the latent space in Figure 4.3. Finally, since the dimension p is still reasonable compared to n, a standard GMM with spectral constraints may be fitted and performs well. Here, the BIC criterion selects the EEE model which means ellipsoidal, equal volume, shape and orientation $\boldsymbol{S}_k = \lambda \boldsymbol{D}\boldsymbol{\Delta}\boldsymbol{D}$, and corresponds to the true model.

Kmeans	BFEM	FEM	HDDC	MCFA	PGMM	Mclust
0.24 ± 0.1	$\mathbf{1 \pm 0}$	0.98 ± 0.11	$\mathbf{1 \pm 0}$	0.62 ± 0.07	0.42 ± 0.22	0.97 ± 0.12

Table 4.2: Mean ARI and standard errors for 100 simulations of Chang's setting.

Figure 4.3 shows the 1-dimensional discriminative subspace found by BFEM on one simulation, with colors indicating cluster membership. Moreover, the solid lines represent the Gaussian density in each cluster $p(\boldsymbol{x} \mid \tilde{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k)$, along with the empirical within-cluster distribution as a histogram. Unsurprisingly, the subspace induce well-separated clusters and the empirical within-cluster distribution has a Gaussian shape fitting the theoretical one. Finally, the evolution of the evidence lower bound during the BFEM algorithm is displayed in Figure 4.4. As expected, it is not monotonically increasing, especially in the first step, although the evolution is quite smooth. In addition, convergence happens before the limit of 100 iterations is reached.

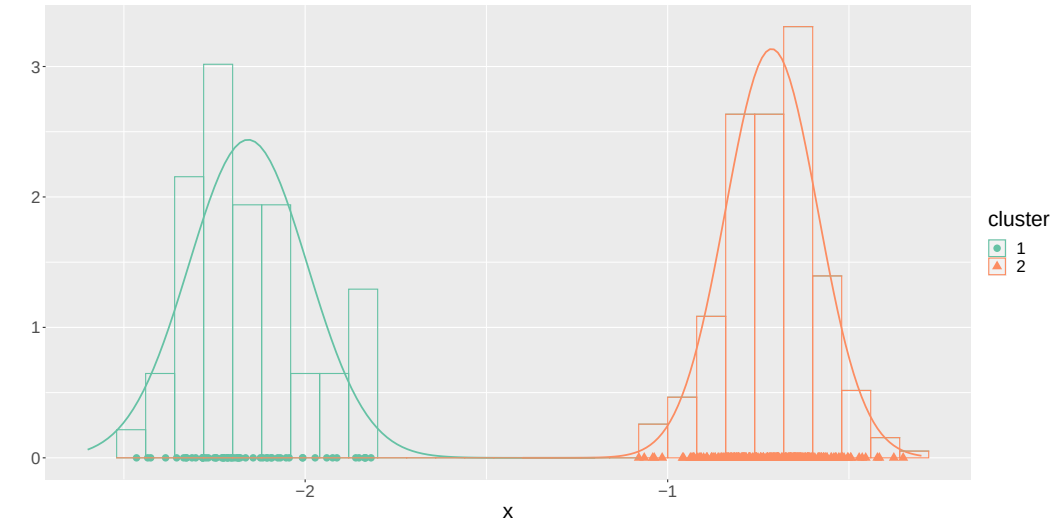


Figure 4.3: Projection of Chang's data in the 1-dimensional subspace found by BFEM, colors indicate the estimate cluster memberships. Solid lines represent the learned within-cluster Gaussian distributions, while the histograms represent the empirical ones.

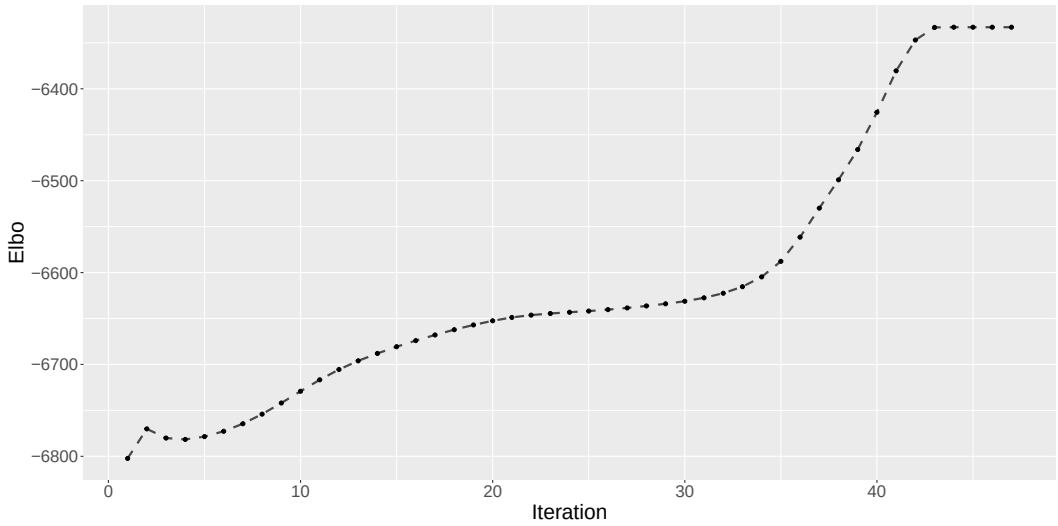


Figure 4.4: Evolution of the evidence lower bound during a run of the BFEM algorithm with ODV procedure on Chang's dataset.

4.4.2 Sensitivity to the dimension

We now propose to investigate the behavior of subspace clustering methods on increasingly high-dimensional problems. In this setting, we simulate $\boldsymbol{X}$ from $K = 3$ Gaussian components in dimension $d = 2$. The respective means and covariance matrices are

$$\boldsymbol{\mu}_k = 3\,(0, k)^\top \qquad \boldsymbol{\Sigma}_k = \begin{pmatrix} 1.5 & 0.75 \\ 0.75 & 0.45 \end{pmatrix} \qquad \boldsymbol{\pi} = (0.4, 0.3, 0.3)^\top.$$

Figure 4.5 illustrates a particular simulation of $n = 900$ data points. As can be seen, it corresponds to the particular case where the clusters are parallel Gaussian ellipses, differentiated with a mean-shift along the x-axis.

Next, we propose to simulate according to the DLM model. First, a matrix $\boldsymbol{A} \in \mathbb{R}^{p \times p}$ is simulated as $\boldsymbol{A} \sim \otimes_{j,j'} \mathcal{N}(0, 100)$. An orthogonal transformation $\boldsymbol{D}$ is computed afterward, as the Q-matrix of the QR-decomposition. Then, for each observation i, a $(p-d)$ dimensional standard Gaussian noise $\boldsymbol{\epsilon}_i$ is simulated. Finally, the data points are created as the linear transformation $\boldsymbol{y}_i = \boldsymbol{D}(\boldsymbol{x}_i^\top, \boldsymbol{\epsilon}_i^\top)^\top$. The first principal components are expected to behave poorly in terms of class separation in this scenario, which is illustrated in Figure 4.6 for $p = 50$. Indeed, the directions of greatest variations include noisy directions that contribute more to the variance than the second signal dimension.

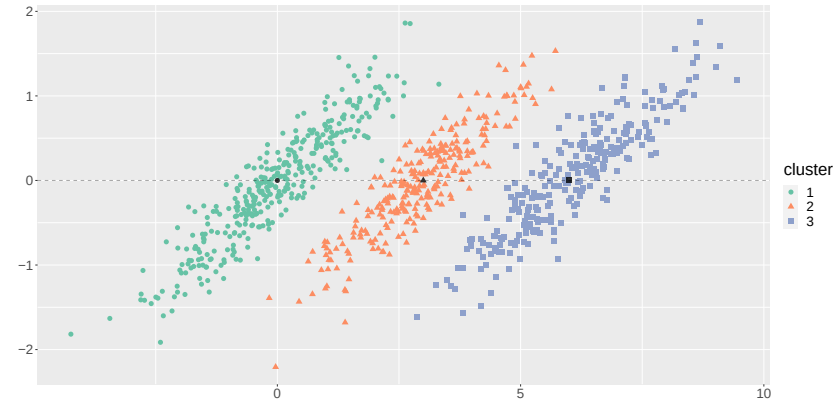


Figure 4.5: Simulation of the 3-components GMM of Section 4.4.2.

Figure 4.6: Subspaces found by PCA on a simulation with $p = 50$, color and shape indicate the true cluster membership. Once again, the most discriminative subspace is given by the first and last components, the other corresponding to noisy directions.

We investigate the behavior of each method as the dimension p increases from 5 to 155. Mean ARI and standard variations were computed on a 10-spaced linear grid, for 100 simulated datasets at each level p. As stated above, we use colors to differentiate between

the BFEM and FEM, and line marker types to distinguish between OVD (solid) and SVD (dashed) for both algorithms. Concerning MCFA and PGMM, due to the increasing computational cost of the experiments, the subspace dimension was set to the true value $d = 2$. The results are displayed in Figure 4.7 and shows several things. First, the BFEM and FEM with the ODV method are very robust in this scenario, with a perfect recovery at each level p. Other subspace clustering methods quickly decrease beyond $p = 15$ with performances comparable, or below k-means. Thus it underlines a limit of likelihood-based approach, as noisy dimensions are being fitted in the subspace when p increases. The discriminative approach, injecting clustering information in the search for the optimal subspace, is robust in this context as the optimal subspace is not necessarily aligned with greatest variance directions. Another interesting fact is the sensitivity to noise of the Fisher EM with the SVD procedure, which displays a rather unstable behavior. We note that BFEM with the SVD procedure, while suboptimal, is still displaying a strong stability in high-dimensional settings, with an ARI decreasing only after $p = 85$

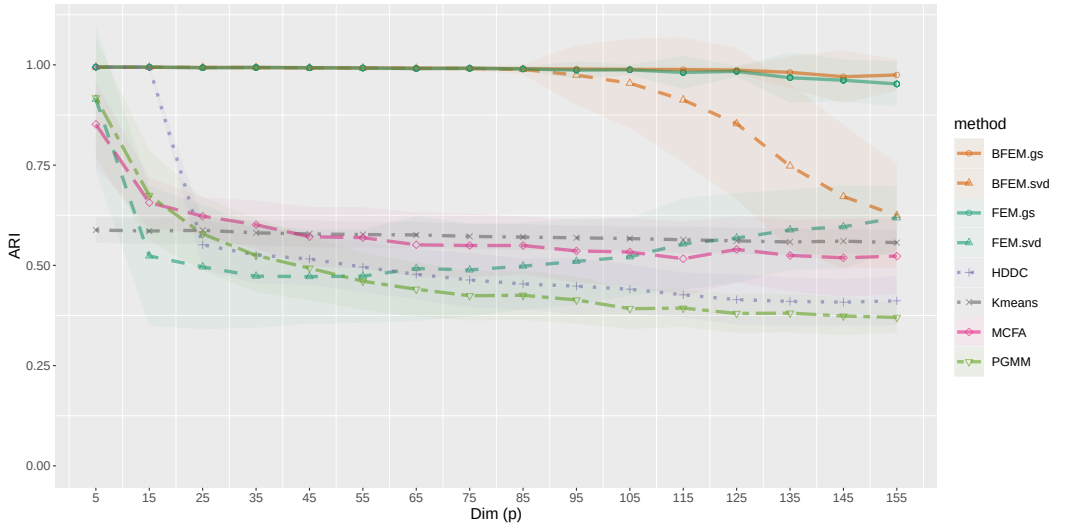


Figure 4.7: Evolution of the mean ARI over 100 different runs for each method, with an increasing dimensionality p and $n = 900$.

4.4.3 Signal-to-noise ratio

We place ourselves in the same setting as section 4.4.2, only this time the dimension is fixed to a high-dimensional scenario $p = 150$. We propose to investigate the impact of the noise, with $\boldsymbol{\epsilon}_i$ now drawn from a centered Gaussian distribution with covariance $\beta \boldsymbol{I}_{p-d}$. The latter may be interpreted as controlling the signal-to-noise ratio (SNR) which can be defined as the ratio between signal variance in the subspace of dimension $d = 2$ and the noise variance* β. Since all clusters have the same subspace covariance, we define the signal variance as $\mathrm{Tr}[\boldsymbol{\Sigma}]$, the inertia of a cluster cloud point in the latent subspace. The SNR is best expressed

*One could use $(p - d)\beta$ as the actual variance of the signal, taking into account the fact that there are $(p - d)$ noisy directions. However, since $p - d$ is fixed here, it only acts as a scaling factor for the SNR.

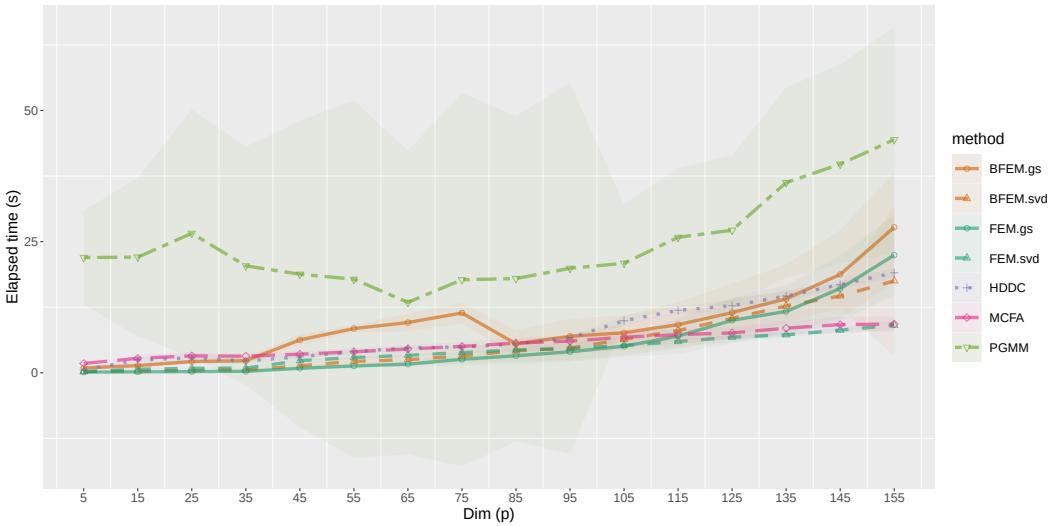


Figure 4.8: Mean elapsed time in seconds for one run of each method, computed on 100 datasets for each level p.

in decibels (dB), which corresponds to ten times the decimal logarithm of the variance ratio.

$$SNR = 10 \times \log_{10}\left(\frac{\mathrm{Tr}\left[\mathbf{\Sigma}\right]}{\beta}\right).$$

A value of 0 means that both variances are equal, and an increase (resp. decrease) of 3 dB means that the variance of the noise was divided (resp. multiplied) by 2. For example, a SNR of 3 dB means that the variance of the signal is 2 times that of the noise ($\beta \approx 1$), and a SNR of -6 means that β is 4 times greater than the signal variance ($\beta \approx 8$).

Figure 4.9 shows the mean ARI and standard deviations for an increasing SNR from -6 to 15 with a 0.5-spaced linear grid. Again 100 datasets are simulated for each level. Several comments are in order. First, for high values of the SNR, which we refer to as the noiseless regime, the clustering problem of Section 4.4.2 becomes trivial, except for k-means which is disadvantaged by the non-spherical shapes of clusters and Fisher EM which seems to display a surprising instability in the noiseless limit. From the preceding section, we know that the Fisher EM algorithm with the SVD procedure is not robust to high-dimension. However, this shows that the ODV procedure also suffers from instability in the frequentist setting. This may be due to poor conditioning of the soft between-class scatter matrix arising in this case. Apart from this somewhat surprising fact, the behaviors of other subspace clustering methods such as HDDC, MCFA and PGMM are expected. However, their performances quickly decrease, even for reasonable values of the SNR where the noise variance is orders of magnitude below the signal. A contrario, the BFEM displays a strong stability, and the SVD method seems to be applicable as long as the noise variance remains reasonably below the signal. In addition, BFEM with the ODV procedure is the most stable of all, with perfect recoveries even when the SNR is 0 dB, *i.e.* the equal variance case. Eventually, no clustering structure can be recovered below 0 dB, as the signal is completely overwhelmed by noisy directions, and the ARI of each method quickly decreases to 0.

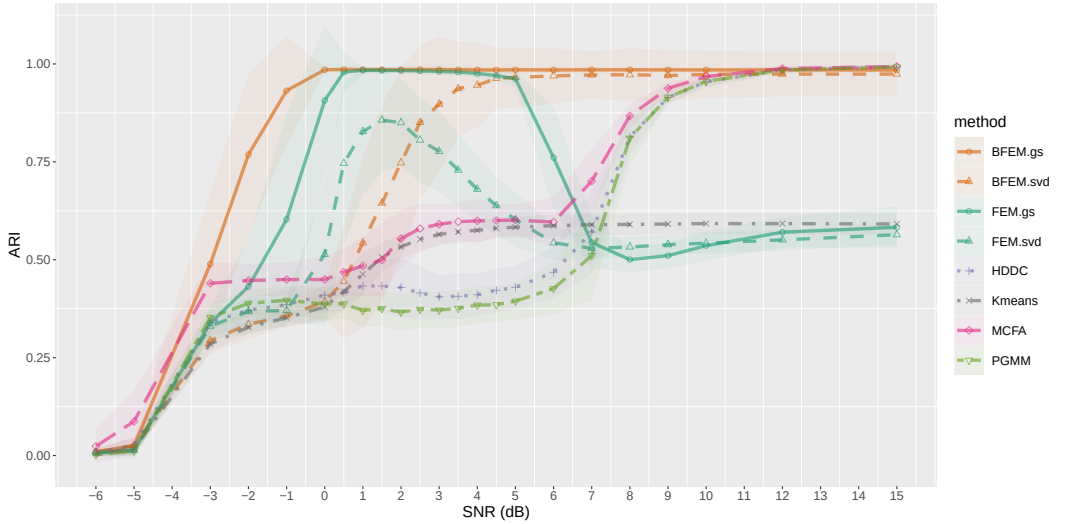


Figure 4.9: Evolution of the mean ARI over 100 different runs for an increasing signal-to-noise ratio, $p = 150$ and $n = 900$.

4.4.4 Model selection

Here, we investigate the ability of the ICL criterion of Section 4.3.6 to choose both the number of clusters and the model. We use the setting of Section 4.4.2, which corresponds to $p = 150$, $K = 3$ and a model $\text{BDLM}_{[\boldsymbol{\Sigma}\beta]}$. Two different levels of SNR are tried: 3 dB, which corresponds to the setting of Figure 4.7 with $\beta = 1$, and -2 dB, which is a more complicated case for BFEM as shown in Figure 4.9.

The results are shown in Table 4.3, we see that the performance of model selection are perfect for the first setting, and still very satisfying in the more difficult scenario with 90% of correct selection of the pair $(K, \mathcal{M})$, and 98% of correct selection of K.

4.4.5 Real data benchmarks

Here, we consider classical real-data benchmarks considered in subspace clustering literature:

- Fisher's iris is a traditional real dataset used to assess clustering algorithms, although it cannot be deemed as a high-dimensional problem. It consists in 150 observations of 3 iris species, 50 each, described by 4 variables.
- The Italian Wine dataset contains the description of 178 wines 27 variables related to *e.g.* color or alcohol (Weinen 1986). There are 3 types of wines, and we wish to know to which extent the 27 variables can help relate to the type of wine. This dataset is also a famous introductory dataset for subspace clustering method (McNicholas and Murphy 2008; Bouveyron et al. 2019).
- The USPS358 dataset is a more realistic example of high-dimensional data clustering. It is a subset of the US postal dataset from UCI, which originally contained 16×16 images of scanned digits from 0 to 9, with only digits 3, 5 and 8 known to be the most difficult to discriminate. There are $n = 1,756$ images, described by $p = 256$ pixels indicating gray level value. In this scenario, we want to recover the 3 different digit classes.

***SNR*=3**

$K \setminus \mathcal{M}$	$[\Sigma_k\beta]$	$[\alpha_{kh}\beta]$	$[\alpha_k\beta]$	$[\boldsymbol{\Sigma\beta}]$	$[\alpha\beta]$	$[\alpha_h\beta]$
2	0	0	0	0	0	0
3	0	0	0	**100%**	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0

***SNR*= − 2**

$K \setminus \mathcal{M}$	$[\Sigma_k\beta]$	$[\alpha_{kh}\beta]$	$[\alpha_k\beta]$	$[\boldsymbol{\Sigma\beta}]$	$[\alpha\beta]$	$[\alpha_h\beta]$
2	0	0	2%	0	0	0
3	8%	0	0	**90%**	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0

Table 4.3: Percentage of correct model selection for BFEM with $p = 150$ and varying SNR. The true model is $\text{BDLM}_{[\Sigma\beta]}$ with $K = 3$.

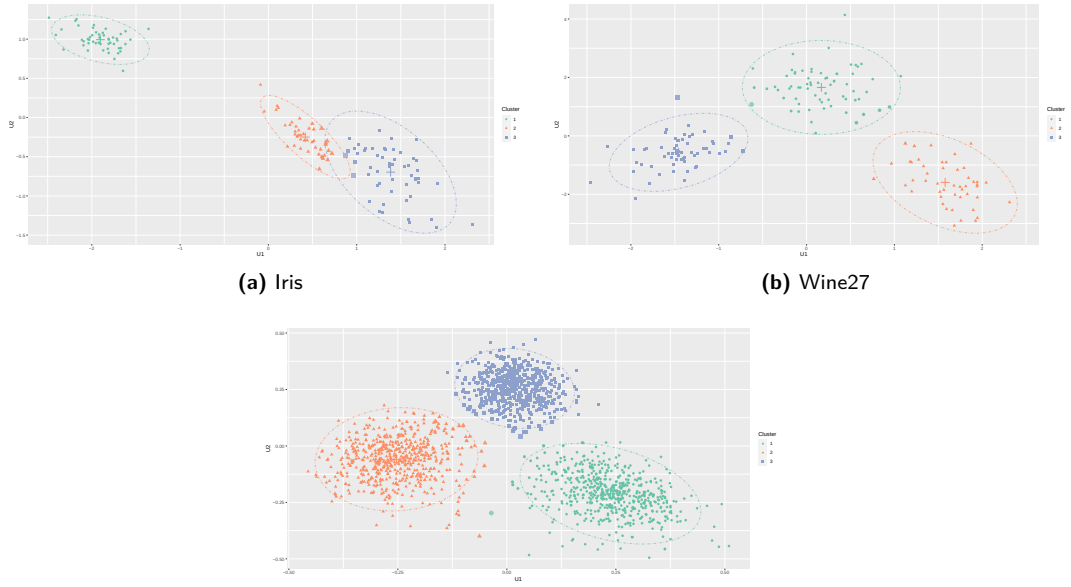
These datasets are available in the **MBCbook** R package. We ran each method with the number of clusters we seek in the corresponding dataset. The latent dimension of PGMM and MCFA was chosen by BIC with $d \in \{1, 2, 3\}$ for Iris and $d \in \{1, \ldots, 10\}$ for Wine27 and USPS358. All models were allowed for HDDC, including the one with different subspaces, and we used the ICL as the model selection criterion. Table 4.4 shows the results. We see that, while PGMM performs better on Iris and Wine27, BFEM displays a real interest on the high-dimensional of clustering USPS358 in dimension $p = 256$, achieving the top performance. Moreover, for the Wine27 dataset, MCFA and PGMM respectively chose $d = 6$ and $d = 4$ while HDDC selected a $[a_j bQd]$ model with $d_k = d = 4$ in each class. A contrario, the FEM an BFEM works with $d = 2$ and the discriminative subspace can therefore be plotted entirely in a two-dimensional graphic, as shown in Figure 4.10.

4.5 Conclusion and perspectives

In this chapter, we introduced a Bayesian formulation of the discriminative latent mixture model, and proposed a variational algorithm for clustering high-dimensional data. Building on the Fisher EM, it relies on the introduction of an additional F-step, although the posterior membership probabilities differ at each step. The hyper-parameter λ, controlling between class separation in the latent space, is estimated via an empirical Bayes strategy, and an ICL criterion is derived for model selection. A detailed experimental setting in high-dimension is

	Iris	Wine27	USPS358
Kmeans	0.73	0.90	0.64
BFEM	0.90	0.95	**0.76**
FEM	0.88	0.93	0.66
HDDC	0.90	0.93	0.35
MCFA	0.92	0.96	0.28
PGMM	**0.94**	**0.98**	0.38
Mclust	0.90	0.93	0

Table 4.4: ARI performance on real datasets



(a) Iris **(b)** Wine27

(c) USPS358

Figure 4.10: Real datasets projected onto the two-dimensional subspace learned by BFEM.

designed, comparing performances with both the frequentist Fisher EM and other state-of-the-art Gaussian subspace clustering algorithms. Our algorithm shows both superiority to the latter as well as a significant improvement to the Fisher EM, especially when the SVD method is used.

Chapter 6 discusses possible extensions of the BFEM algorithm which we sketch here. First, one could seek to perform variable selection in addition to clustering and dimension reduction. To that end, sparse extensions of the Fisher EM were proposed in Bouveyron and Brunet-Saumard (2014) based on the regression formulation of Qiao et al. (2009) with a Lasso-like penalty. A natural extension would be to propose a sparse BFEM algorithm relying on the same formulation. Existing literature on model-based feature selection being rather extensive, we prefer to leave a comprehensive and detailed study with carefully designed experimental settings to future work.

Second, the trace of ratio formulation in Equation (4.2) is itself a simplification of the

ratio of trace problem:

$$\max_{\boldsymbol{U}^\top \boldsymbol{U}=\boldsymbol{I}_d} \frac{\mathrm{Tr}[\boldsymbol{U}^\top \boldsymbol{S}_B \boldsymbol{U}]}{\mathrm{Tr}[\boldsymbol{U}^\top \boldsymbol{S}_W \boldsymbol{U}]},$$

which does not accept a closed-form solution and necessitate iterative algorithms to solve. Recent works highlight the better discriminative power of this new formulation in the supervised context, and new iterative algorithms have been designed (Guo et al. 2003; Wang et al. 2007; Kokiopoulou et al. 2011). Based on these empirical results, it would be interesting to modify the F-step in order to maximize such a criterion. Given the diversity of possible algorithms and their need for calibration, we leave a detailed study to future work in order not to overload the figures.

5

Hierarchical clustering in discrete latent variable models with the exact integrated classification likelihood

In this chapter, we investigate model-based hierarchical clustering via a direct maximization of the integrated classification likelihood. More precisely, we focus on the class of DLVMs where an exact ICL, ICL_{ex}, can be derived as discussed in Section 2.5.2. Our contribution is twofold. First, we address the known problem of poor local optima and sensibility to initialization of greedy heuristics. To that end, we propose a genetic algorithm, carefully combining and merging different solutions, allowing for an efficient exploration over the space of partitions. Second, we propose a hierarchical algorithm relying on a new approximation of ICL_{ex}. The latter considers the asymptotic of the partition evidence, $\log p(\boldsymbol{Z} \mid \alpha)$, as the Dirichlet hyper-parameter α goes to 0, viewing the latter as a regularization parameter controlling the granularity of the partition. The output is a nested hierarchy of partitions allowing an exploration of results at coarser scales, and the ordering of the clusters improving visual representations of the clustering results.

5.1 Introduction

5.1.1 A reminder on discrete latent variable models

As discussed in Section 2.2, model-based clustering with discrete latent variable models is a principled approach for clustering, with a variety of flexible models depending on the data at hand (Bouveyron et al. 2019). This class encompasses finite mixture models (McLachlan and Peel 2000), but also related models that do not exactly fit the definition of a FMM. We described popular instances such as the popular stochastic block model (SBM) for network analysis (Wang and Wong 1987; Nowicki and Snijders 2001) and its extensions (see Karrer and Newman 2011, for instance), as well as the latent block model (LBM) (Govaert and Nadif 2010) for co-clustering. The general definition of a DLVM assumes that the observations provided in $\boldsymbol{Y}$ are drawn from a two-step process: first, the latent partition $\boldsymbol{Z}$ is drawn independently from a product of multinomial distributions parameterized by $\boldsymbol{\pi}$. Then, the observations are supposed to be independent given the whole partition. The classification likelihood is written as:

$$p(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\vartheta}) = \prod_{\boldsymbol{z} \in \boldsymbol{Z}} p(\boldsymbol{z} \mid \boldsymbol{\pi}) \underbrace{\prod_{\boldsymbol{y} \in \boldsymbol{Y}} p(\boldsymbol{y} \mid \boldsymbol{Z}; \boldsymbol{\theta})}_{\text{factorized}}, \tag{5.1}$$

In the case of mixture models, the observations are n independent random vectors $\boldsymbol{Y} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n\}$ in $\mathbb{R}^p$, which can be summarized in a data matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$. In this context, each observation is assigned to a latent multinomial variable $\boldsymbol{z}_i \in \{0, 1\}^K$, defining its cluster assignment. The latter is independently drawn from a multinomial distribution, with proportions $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$, where K denotes the number of components or clusters. Then, an observation $\boldsymbol{y}_i$ follows some conditional distribution depending on the value of $\boldsymbol{z}_i$, and the sampling process for all i is as follows:

$$\begin{aligned} \boldsymbol{z}_i \mid \boldsymbol{\pi} &\sim \mathcal{M}(1, \boldsymbol{\pi}), \\ \boldsymbol{y}_i \mid z_{ik} = 1, \boldsymbol{\theta}_k &\sim p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_k). \end{aligned} \tag{5.2}$$

The parameter $\boldsymbol{\pi}$ controls the prior probability of belonging to each group, while the mixture parameters control the distribution in the k-th cluster, and depend on the observational model at hand. For instance, in a Gaussian mixture we have $\boldsymbol{\theta}_k = (\boldsymbol{m}_k, \boldsymbol{S}_k)$, respectively the mean and covariance matrix.

In the case of the stochastic block model, the observations are the edges $\boldsymbol{Y} = \{y_{ij}\}$, where y_{ij} represents the presence of absence of an edge. It can be binary, $y_{ij} \in \{0, 1\}$, or weighted $y_{ij} \in \mathbb{R}$ (Mariadassou et al. 2010). Observing the edges, *e.g.* the topology of the graph, we wish to cluster the nodes $\{1, \dots, n\}$. Thus, each node i is assigned to a cluster latent variable $\boldsymbol{z}_i$ and the edges are supposed to be conditionally independent given the partition, with a conditional distribution depending only on the clusters of their out and end-nodes. The sampling process is then written as:

$$y_{ij} \mid z_{ik} z_{jl} = 1, \boldsymbol{\theta}_{kl} \sim p(y_{ij} \mid \boldsymbol{\theta}_{kl}). \tag{5.3}$$

As in mixture models, the parameter $\boldsymbol{\pi}$ controls the group proportions and the latent partition is drawn independently from $\mathcal{M}_K(1, \boldsymbol{\pi})$. The set of parameters $\boldsymbol{\theta}_{kl}$ is now specific to the pair (k, l) of clusters, and depends on the specific model. For instance, in the case of a binary SBM $\theta_{kl} \in [0, 1]$. In Section 2.2.2.a, we discussed the differences with standard finite mixture models, due to the marginal dependencies between edges arising when marginalizing over $\boldsymbol{Z}$.

In co-clustering, the observations $\boldsymbol{Y} = \{y_{ij}\}$ are supposed to be given in a data matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$, and one seeks a bipartition $\boldsymbol{Z} = (\boldsymbol{Z}^r, \boldsymbol{Z}^c)$ with K_r clusters over the n rows and K_c clusters over the p columns. The latent block model (LBM, Govaert and Nadif 2010) supposes conditional independence of entries $\boldsymbol{y}_{ij}$ given $\boldsymbol{z}_i^r$ and $\boldsymbol{z}_j^c$. The sampling scheme is given as:

$$y_{ij} \mid z_{ik}^r z_{jl}^c = 1, \boldsymbol{\theta}_{kl} \sim p(y_{ij} \mid \boldsymbol{\theta}_{kl}), \tag{5.4}$$

and is very close to SBM. Indeed, the latter may be viewed as a particular instance of LBM when $n = p$ and $\boldsymbol{Z}^c = \boldsymbol{Z}^r$. Moreover, the row partition $\boldsymbol{Z}^r$ and column partition $\boldsymbol{Z}^c$ are supposed to be respectively drawn *i.i.d.* from $\mathcal{M}_{K_r}(1, \boldsymbol{\pi}^r)$ and $\mathcal{M}_{K_c}(1, \boldsymbol{\pi}^c)$. Thus, the distribution of $\boldsymbol{Z}$ is a product of multinomials parameterized by $\boldsymbol{\pi} = (\boldsymbol{\pi}^r, \boldsymbol{\pi}^c)$, hence fitting the definition of Equation (5.1).

5.1.2 Greedy maximization of the exact ICL criterion

As discussed in Section 2.5.2, the ICL criterion was introduced for the purpose of model selection in the specific case of model-based clustering. Biernacki et al. (2000) first used a combination of Laplace and Stirling approximations on $\log p(\boldsymbol{Y}, \boldsymbol{Z} \mid K, \mathcal{M})$ to find an asymptotic criterion called ICL_{BIC} due to its close link to the BIC.

Recent works have also considered exact expressions of the ICL, putting a factorized conjugate prior distribution over the model parameters $p(\boldsymbol{\pi}, \boldsymbol{\theta} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) = p(\boldsymbol{\pi} \mid \boldsymbol{\alpha}) p(\boldsymbol{\theta} \mid \boldsymbol{\beta})$, and defined as:

$$\begin{aligned} \text{ICL}_{ex}(\boldsymbol{Z}; \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \log \left(\int_{\boldsymbol{\theta}} \int_{\boldsymbol{\pi}} p(\boldsymbol{X} | \boldsymbol{Z}, \boldsymbol{\theta}) p(\boldsymbol{\theta} | \boldsymbol{\beta}) p(\boldsymbol{Z} | \boldsymbol{\pi}) p(\boldsymbol{\pi} | \boldsymbol{\alpha}) \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\boldsymbol{\pi} \right), \\ &= \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\beta}) + \log p(\boldsymbol{Z} \mid \boldsymbol{\alpha}). \end{aligned} \tag{5.5}$$

The $\boldsymbol{\alpha}$ parameters control the conjugate distribution, which, in the case of a DLVM, is a Dirichlet* over group proportions $\mathcal{D}_K(\boldsymbol{\alpha})$. This part is thus common to all DLVMs in the sense that it does not depend on the observational model on $\boldsymbol{Y}$. If a symmetric Dirichlet is chosen, with $\alpha_k = \alpha$, the second term can be made explicit using the independence of the

*Or a product of Dirichlet distributions $\mathcal{D}_{K_r}(\boldsymbol{\alpha}_r) \times \mathcal{D}_{K_c}(\boldsymbol{\alpha}_c)$ in the case of co-clustering with the LBM. Except for an additional notation burden, the rest of the discussion easily extends to this case, which is discussed in detail in the Section 5.4.

elements of $\boldsymbol{Z}$:

$$\mathrm{ICL}_{ex}(\boldsymbol{Z}; \alpha, \boldsymbol{\beta}) = \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\beta}) + \log \left(\frac{\Gamma(K\alpha) \prod_{k=1}^{K} \Gamma(\alpha + n_k)}{\Gamma(\alpha)^K \Gamma(n + \alpha K)} \right), \tag{5.6}$$

with $n_k = \sum_i z_{ik}$. Usually, the hyper-parameter α is set to 1 or $\frac{1}{2}$ to specify uniform or Jeffreys prior.

The hyper-parameters $\boldsymbol{\beta}$ control the conjugate prior over the mixture parameters $\boldsymbol{\theta}$, and depends on the generative model at hand. Naturally, such criterion is restricted to particular DLVMs, where such conjugate distributions are easy to derive, so that the first term in Equation (5.5) is analytic. However, this class is quite large and expressions are available for the mixture of multinomials (Biernacki et al. 2010) and Gaussian mixture (Bertoletti et al. 2015), while being virtually feasible for any mixture of exponential families as they admit natural conjugate priors (Gelman et al. 2004, p. 42). Exact ICL criteria were also derived for the SBM (Côme and Latouche 2015), the LBM (Wyse et al. 2017) and degree-corrected variants (Newman and Reinert 2016; Riolo et al. 2017).

In between the frequentist and the Bayesian approaches, a new line of work started to consider direct maximization of ICL_{ex} with respect to the partition $\boldsymbol{Z}$, avoiding the inference step over the parameters $(\boldsymbol{\pi}, \boldsymbol{\theta})$. In order to solve this discrete and combinatorial optimization problem, greedy heuristics were successfully tested to directly optimize this criterion over the space of possible partitions. These approaches consist in hill-climbing algorithms, starting from an initial partition and greedily swapping clusters until some local maximum is met. Eventually, in the end, some clusters are merged up to the point where no more merge moves can maximize the ICL_{ex}. Such type of algorithms performs model selection and clustering at the same time and are computationally attractive compared to approximate or exact inference alternatives. This approach dates back to Tessier et al. (2006), for the latent class model. It was then extended in Côme and Latouche (2015) for SBM, and applied to other DLVMs such as Gaussian mixture models (Bertoletti et al. 2015), LBM (Wyse et al. 2017) and dynamic variants of SBM (Corneli et al. 2016; Zreik et al. 2016).

5.1.3 Spurious local maxima and genetic clustering algorithms

The aforementioned greedy maximization procedure comes with a cost. In practice, the objective is highly multimodal, and the combinatorial nature of the search space multiplies the presence of poor local maxima in which the method gets stuck. To tackle this problem, Côme and Latouche (2015) proposed to perform several restarts with different initializations, while Bertoletti et al. (2015) specifically designed a batch version of the greedy heuristic, swapping groups of several nodes in order to avoid some the local maxima.

Based on a similar observation, Tessier et al. (2006) suggested that simple greedy hill climbing heuristics on the ICL_{ex} could be improved by the use of genetic evolutionary algorithms (GA, Eiben and Smith 2004). This methodology borrows from biological evolution principles, combining solutions via crossover operators, allowing random modifications and discarding poor solutions, in an analogy to genetic inheritance, mutations and natural selection. Thus, they allow to efficiently explore the partition space and to avoid the pitfalls of spurious maxima through the recombination and mutation operators. Apart from the work of Tessier et al. (2006) for the latent class model, evolutionary algorithms were proposed for

Gaussian model-based clustering, maximizing the (non integrated) classification likelihood (Andrews and McNicholas 2013), and in the context of feature selection (Scrucca 2016). More generally, the specific use of GA for clustering problems is not new (Cole 1998), and we refer to Hruschka et al. (2009) for a recent and detailed review on the subject. In the specific case of ICL_{ex} maximization, they present an appealing method to improve greedy hill climbing heuristics (Tessier et al. 2006), or, recently put in these words:

> *Several authors have considered the direct optimization of the exact ICL in $\boldsymbol{Z}$ without estimating $\boldsymbol{\vartheta}$ [...] the proposed greedy algorithms are highly sensitive to the numerous local optima and have only been experimented with for moderate sample sizes. This is the reason why evolutionary algorithms are expected to be useful but they need to be calibrated (to choose the tuning parameters) and are expensive in computing time.*
>
> ~ Fruhwirth-Schnatter et al. (2019, p. 137)

5.1.4 Hierarchical clustering using the ICL

Having derived a solution with a given number of clusters K, it may be of interest to construct a hierarchy from its clusters, allowing the exploration of partitions at different granularity levels. Model-based hierarchical clustering extends the idea of non-parametric and similarity-based hierarchical clustering strategy, such as Ward's methods (Ward Jr 1963) or complete-link (Sokal and Michener 1958) and single-link (Sneath 1957) clustering. The first work of Murtagh and Raftery (1984) extends Ward's criterion as the likelihood in an isotropic Gaussian mixture models, and was later extended to the general case of spectral constraints $\boldsymbol{S}_k = \lambda_k \boldsymbol{D}_k \boldsymbol{\Delta}_k \boldsymbol{D}_k^\top$ (Banfield and Raftery 1993; Fraley 1998). In this spirit, Zhong and Ghosh (2003) proposed an extension of Ward's distance as the difference of log-likelihoods before and after a merge, along with ways to approximate it when the inference step is too costly to be done for each fusion.

More recently, model selection criteria were proposed as objective functions in hierarchical clustering algorithms. Heller and Ghahramani (2005) proposed a hierarchical Bayesian clustering algorithm, based on hypothesis testing. Marginal likelihoods of clusters are computed at each stage, using conjugate priors involving similar expressions as in the ICL_{ex}. Explicitly working with a ICL_{BIC} criterion, Baudry et al. (2010) proposed a soft hierarchical clustering algorithm for finite mixture models. Relying on an asymptotic approximation rather than exact derivation, it chooses the merge inducing lowest posterior entropy for the cluster memberships probabilities. Thus, the latter is used to assess clustering quality, and the output is a hierarchy of soft partitions. In the context of network analysis, Peixoto (2014) proposed a greedy hierarchical clustering algorithm for a hierarchical formulation of the SBM, using another model selection criterion: the *description length*. Although the criterion differs, the author shows that it matches the ICL_{ex} when the prior on the connection probabilities of the SBM is replaced by a nested sequence of priors and hyper-priors. Finally, the greedy hill climbing heuristics possess a final merge stage but they stop when a local maximum of ICL_{ex} is reached, and are unable to complete a full hierarchy.

5.1.5 Contributions and organization of the chapter

This chapter builds on two main contributions to propose a two-step methodology for hierarchical clustering.

First, Section 5.2 addresses the issue of spurious local maxima in greedy maximization of ICL_{ex}. We propose a hybrid genetic algorithm mixing an evolutionary strategy with local search to optimize the ICL_{ex} criterion, efficiently exploring the space of partitions. The novelty and efficiency of this approach reside in the representation of solutions as set partitions, and in the crossover operator used to recombine solutions, carefully preserving their structure. This algorithm is adaptable to a wide variety of DLVMs, as soon as swap and merge moves can be efficiently computed.

Second, Section 5.3 introduces an agglomerative hierarchical algorithm considering ICL_{ex} as a function of the hyper-parameter α and relying on a new approximation, using the asymptotic of the log-Gamma function when α goes to 0. We show that decreasing α can unlock fusions in the sense that coarser partitions achieve a greater ICL value. Starting from an ICL-dominant solution at a given level α, typically 1, the proposed heuristic extracts a set of nested clustering, each of which is dominant with respect to this new criterion over some range of α values. In addition, this strategy enables the construction of a cluster dendrogram, giving a natural ordering for the clusters which is interesting for visualization purposes, particularly on real datasets.

These two contributions are generically applicable in the framework of DLVMs for which conjugate prior can be easily derived. In Section 5.4, we derive ICL_{ex} expressions and discuss model-dependent questions such as the specification of hyper-parameters $\boldsymbol{\beta}$. Specifically, the treated DLVMs are the mixture of multinomials, SBM, degree-corrected SBM and LBM. Moreover, these algorithms are naturally linked, working with similar objectives, and the first one can be used as an initialization for the second to extract a hierarchical clustering. One of the particularities of this approach is that it only extracts the relevant part of the dendrogram, since the latter typically starts with an optimal partition obtained at $\alpha = 1$ or $1/2$. Therefore, it avoids the analysis of uninformative fusions commonly encountered in the first stages of classical hierarchical agglomerative clustering algorithms. This approach is also computationally efficient and may handle large datasets which could be hard to grasp with classical fully hierarchical algorithms.

Section 5.5 gives a detailed investigation of the two algorithms behavior on simulated and real datasets, along with a thorough comparison with related model-based clustering algorithms.

As a motivating example for the proposed two-step methodology, we simulate a random SBM graph with $n = 1500$ nodes and a hierarchical cluster structure with 3 big clusters each composed of 5 small clusters. The small clusters have an intra-connectivity probability of 0.1 and a probability of connecting a node from the same big cluster of 0.025. Moreover, two random nodes may be connected with a probability of 0.001. Figure 5.1 presents the result of a greedy optimization with a random starting partition with twenty clusters. The results of the proposed hybrid optimization algorithm and the same results after a reordering of the clusters with the hierarchical heuristic. As clearly shown by this example, the greedy heuristic with a random starting point suffers from under-fitting with only six clusters extracted among the 15 simulated. The hybrid algorithm does not suffer from the same problem in this example, and recovers correctly the 15 simulated clusters. Finally, the hierarchical ordering enables a clear visualization of the hierarchical structure of this dataset, that is also clearly depicted in the extracted dendrogram presented in Figure 5.2.

Finally, an open-source **R** package (R Core Team 2019) **greed** providing a reference implementation of the algorithms introduced in this chapter is also available. The implementation is extendable and new models can be integrated. The main computationally demanding methods were developed in **Cpp** thanks to the **Rcpp** package (Eddelbuettel and Balamuta 2017) taking advantages of sparse matrix computational efficiency thanks to the **RcppAr-**

madillo and **Matrix** packages (Eddelbuettel and Sanderson 2014; Bates and Maechler 2019) which offer a natural interface with the **Armadillo** Cpp library for linear algebra with sparse matrix (Sanderson and Curtin 2019). Eventually, the **future** package (Bengtsson 2019) was used to enable easy parallelization of the computations of the proposed hybrid genetic algorithm.

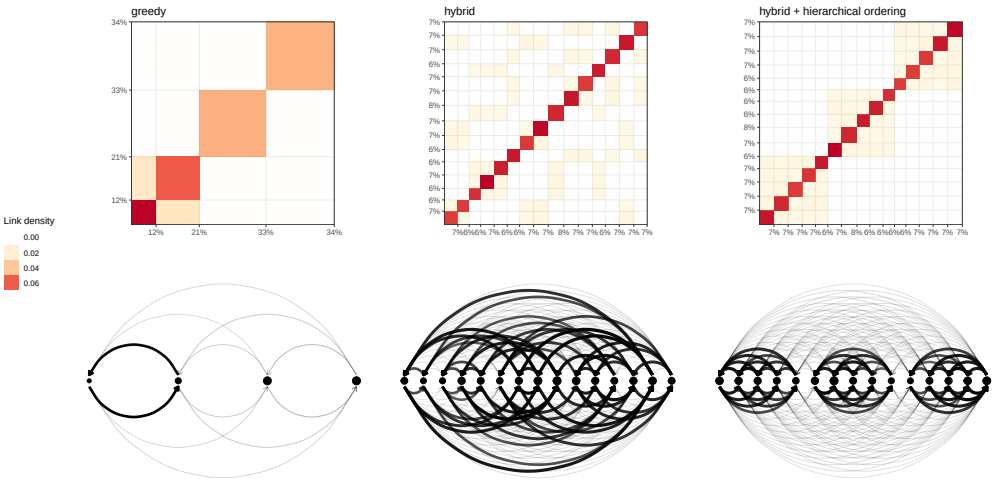


Figure 5.1: Motivating example for the proposed algorithms. Block matrix representation of the solutions (upper row) and cluster node link diagram (bottom row) obtained with (from left to right) a greedy algorithm with a random starting point, the proposed hybrid algorithm and the same clustering but with clusters rearranged thanks to the hierarchical ordering.

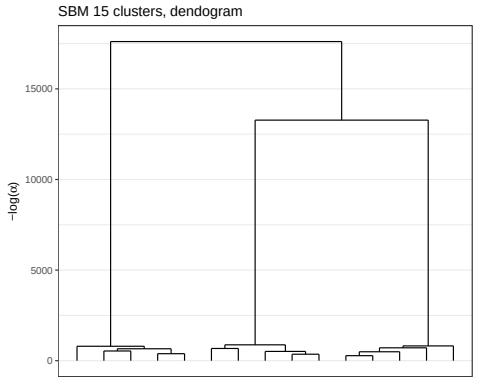


Figure 5.2: Motivating example for the proposed algorithms. Clusters dendrogram extracted with the hierarchical regularization path heuristic.

5.2 A hybrid genetic algorithm for DLVMs

As explained above, several works rely on the ICL_{ex} criterion as an objective function to maximize with respect to the partition $\boldsymbol{Z}$. These are mainly based on greedy hill climbing

algorithms: starting from a carefully chosen over-segmented initial partition, or *seed*, swaps and eventually merges are applied to increase the criterion. In addition to the competitive computational complexity and the ease of implementation, these algorithms may be seen as an automatic way to perform model selection, as clusters may be emptied during the process. In the SBM case, Côme and Latouche (2015) propose a thorough comparison with state-of-the-art methods that illustrates the interest of such algorithms.

However, a major drawback of this approach is its dependency to the initialization. Indeed, defining a relevant initial partition is not trivial, and the method may lead to underfitting as demonstrated in the introductory example in Figure 5.1. Here, the issue seems to lie in the lack of exploration of the partition space, and genetic algorithms (GAs) have been proposed to improve the exploration. Starting from a given solution, the latter evolve a population of candidate solutions by selecting some of the most promising ones, combining them, and mutating them until a specified number of generations or some stopping criterion is met. As described in Eiben and Smith (2004, Chapter 2), the fundamental components of such algorithms are the solution representation, the selection strategy and the variation operators used for recombination and mutation. However, while GAs are very good at identifying near-optimal regions of the search space, they can take a relatively long time to reach a local optimum in the region of interest. In order to improve their exploitation capacity, a number of works suggested hybridizing GAs with efficient local search algorithms capable of improving solutions between each generation (see Eiben and Smith 2004, Chapter 10). These evolutionary methods have been named in various ways, such as hybrid GAs, memetic GAs, and genetic local search algorithms.

In the case of ICL_{ex} maximization, existing greedy heuristics may be seen as such local search algorithms, locally improving a partition, and we build on this idea to propose a hybrid GA. In the following, we discuss the practical choices made when designing the genetic algorithm. Moreover, we emphasize that, in this section, the prior parameters are considered to be fixed to uninformative or default values, and we only optimize $\mathrm{ICL}_{ex}(\boldsymbol{Z})$ with respect to the partition $\boldsymbol{Z}$.

5.2.1 Recombination of solutions with the cross-partition operator

The first step towards building a GA is to define a way to represent candidate solutions inside the algorithm. The latter is also called the *genotype space*, with genotypes as points in this space. This choice is fundamental as it guides the variation operators such as the recombination operator, also known as crossover, which combines two parent genotypes into a new one, and the mutation operator, which randomly modifies genotypes. In the case of clustering, the original space of solutions contains $\boldsymbol{Z}$, implicitly defining a partition $\mathcal{P} = \{\boldsymbol{C}_1, ..., \boldsymbol{C}_K\}$ of $\{1, \dots, n\}$ into K clusters. Tessier et al. (2006) use integer encoding, which consists in a vector of length n where each individual is assigned to an integer $k \in \{1 \dots, K\}$ representing its cluster assignment. However, this approach presents a major drawback. Indeed, akin to the label switching problem in statistical inference, the ICL_{ex} objective function is invariant under a permutation of the cluster indices, and this representation is therefore heavily redundant. Thus, as emphasized in Hruschka et al. (2009), popular crossover operators based on crossover points will not consider this specificity and will completely break the structure of the solution. This is notably the case in Tessier et al. (2006), leading to slow evolution of the population of solutions. We propose to circumvent this issue by directly choosing the space of partitions as the genotype space, defining crossover and mutation operators on it. Such operators will not suffer from label switching, and will preserve the clustering structure present in the genotypes.

CROSSOVER OPERATOR The crossover operator defines how two parent genotypes $\mathcal{P}^1 = \{\boldsymbol{C}_1^1, ..., \boldsymbol{C}_{K_1}^1\}$ and $\mathcal{P}^2 = \{\boldsymbol{C}_1^2, ..., \boldsymbol{C}_{K_2}^2\}$ are combined together to form an offspring. We propose to use the cross-partition, defined as the set of all possible intersections between the elements of the two partitions:

$$\mathcal{P}^1 \times \mathcal{P}^2 := \left\{ \boldsymbol{C}_k^1 \cap \boldsymbol{C}_l^2 \,,\, \forall k \in \{1, ..., K_1\}, l \in \{1, ..., K_2\} \right\} \setminus \{\emptyset\} \,.$$

This operator produces a new partition with at most $K_1 \times K_2$ clusters, which is a refinement of $\mathcal{P}^1$ and $\mathcal{P}^2$ in the sense that both parents may be reconstructed using merge operations. It is also the first common ancestor of both $\mathcal{P}^1$ and $\mathcal{P}^2$ in the partition lattice. Hence, its interest is twofold. First, as in the motivating example, if both parent partitions are under-fitted, crossing them allows the algorithm to go backward in the partition lattice, considering finer clustering. Second, it is particularly appropriate for the hybridization with greedy heuristics. Indeed, unnecessary clusters may be created when the crossed solutions are around the best one. Then, a greedy local search based on merge moves may be used to remove these clusters efficiently.

5.2.2 Selection, mutation and the hybrid algorithm

The remaining aspects of the genetic algorithm concern the selection procedure and the mutation operator. As the population size V is kept fixed throughout the algorithm, selection defines which parent genotypes are combined together to form offspring. Several options were tested which did not greatly changed the performances of the algorithm, and we decided to keep a rank-based selection policy (see Eiben and Smith 2004, pp.81-82). In this scheme, the selected genotypes for building the next generation are chosen according to a probability proportional to their rank in terms of ICL_{ex}.

As for the mutation operator, it randomly acts on the elements of a genotype, here the clusters of a partition. Together with the recombination operator, it allows introducing variability in the algorithm allowing for a better exploration. Again, a desirable property is the refinement of a given partition, and a natural mutation to consider is to split a cluster in two new ones at random. Then, local searches consisting in swaps and merges can either undo a poor split or explore new directions. The resulting hybrid greedy algorithm is represented as pseudo-code in Algorithm 4.

COMPUTATIONAL EFFICIENCY From a computational perspective, the crossover and mutation operator can easily be parallelized since they are independent for each pair of solutions to combine. In addition, while already efficient, this first version was optimized by taking advantage of a special feature of the problem. Indeed, after having formed the crossed partition, one may determine the pairs of clusters (k, l) that have a common parent either in $\mathcal{P}^1$ or $\mathcal{P}^2$:

$$\left\{ (\boldsymbol{C}_k, \boldsymbol{C}_l) \in (\mathcal{P}_1 \times \mathcal{P}_2)^2 : \exists \boldsymbol{C} \in \mathcal{P}^1 \cup \mathcal{P}^2, (\boldsymbol{C}_k \cap \boldsymbol{C} \neq \emptyset) \text{ and } (\boldsymbol{C}_l \cap \boldsymbol{C} \neq \emptyset) \right\},$$

only allowing merge and swap movements between them. This allows gaining a factor K, which can be interesting for a large number of clusters, especially in the first iterations of the algorithm. The rationale behind this restriction is that both initial partitions may be recovered if needed, while the inspection of a non-negligible quantity of merge and swap moves having a low chance of being relevant can be avoided.

This hybrid genetic algorithm allows the extraction of a natural clustering when the number of clusters is unknown, by carefully exploring the space of partitions and exploiting relevant solutions. The trade-off between the two is controlled by a few tuning parameters, namely the population size and the probability of mutation, and the computational complexity, which is model-dependent, is competitive with other approaches. The experiments carried in Section 5.5 will demonstrate its performances in real and simulated settings.

5.3 Hierarchical extension from regularization path

In this section, we introduce the second contribution of this chapter: a greedy agglomerative algorithm for hierarchical clustering, based on an approximation of ICL_{ex}. Hereafter, ICL_{ex} is viewed not only as a function of the partition $\boldsymbol{Z}$ but also of the hyper-parameter α. The asymptotic behavior of the log-gamma function near 0 is used to derive a simple functional form for the criterion as a function of α. The resulting approximation is called ICL_{lin} due to its log-linear dependency in α. Then, α is used as a regularization parameter which unlocks access to simpler, coarser, solutions. The algorithm produces a hierarchy of nested partitions along with the sequence of the regularization parameters which enabled the fusions : $(\boldsymbol{Z}^{(k)}, \alpha^{(k)})_{k=K,\dots,1}$. Eventually, the extracted partitions may be investigated, and the hierarchical structure used to get a pseudo-ordering of the initial clusters to enhance the graphical representation of the clustering results.

Algorithm 4: Hybrid genetic algorithm

Data: population size: V, probability of mutation: pm, maximum number of generations: $maxgen$, dataset $\boldsymbol{Y}$
Result: a partition $\mathcal{P}^*$

Build a population $G = \{\mathcal{P}^1, ..., \mathcal{P}^V\}$ of initial solutions using greedy swap
$nbgen = 1$ **while** $nbgen < maxgen$ **do**
  add the best solution $\mathcal{P}^*$ in the population to the new generation $G_n = \{\mathcal{P}^*\}$
  sample according to their rank in terms of ICL, $(V-1)$ pairs of solution in G
  for *each sampled pairs* $(\mathcal{P}^1, \mathcal{P}^2)$ *of partitions* **do**
    build the cross partition $\mathcal{P}$ of $\mathcal{P}^1$ and $\mathcal{P}^2$
    $\mathcal{P} = \mathcal{P}^1 \times \mathcal{P}^2$
    update $\mathcal{P}$ using greedy merge
    if $random < pm$ **then**
      sample a cluster of $\mathcal{P}$ and split it randomly in two
    end
    update $\mathcal{P}$ using greedy swap
    add $\mathcal{P}$ to the new generation $G_n = \{G_n, \mathcal{P}\}$
  end
  replace the population by the new generation $G = G_n$
  $nbgen \leftarrow nbgen + 1$
end
return the best solution $\mathcal{P}^*$ of G_n

5.3.1 A new approximation for the exact ICL

As shown in Equation (5.6), ICL_{ex} decomposes as the sum of two terms. The first one is $\log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\beta})$, the conditional integrated log-likelihood of the data, given the partition $\boldsymbol{Z}$. It will be denoted by $D(\boldsymbol{Z})$ and only depends on the observed data $\boldsymbol{Y}$, the partition $\boldsymbol{Z}$, and the model specification. The second term is the integrated log-likelihood of $\boldsymbol{Z}$ and depends on the Dirichlet hyper-parameter α:

$$\log p(\boldsymbol{Z} \mid \alpha, K) = \log\Gamma(\alpha K) + \sum_{k=1}^{K} \log\Gamma(\alpha + n_k) - K \log\Gamma(\alpha) - \log\Gamma(n + \alpha\, K). \tag{5.7}$$

Here, the dependency between K and $\boldsymbol{Z}$ is made explicit, the former representing the number of clusters in the latter. Then, we consider the asymptotic behavior of the expression above when α becomes small. First, recall that the log-gamma function behaves as minus the natural logarithm near 0:

$$\log\Gamma(\alpha) = \log(\alpha^{-1}\,\Gamma(\alpha + 1)) \;\approx_0\; -\log(\alpha)\,. \tag{5.8}$$

Then, considering K fixed, we can use this approximation on $\log\Gamma(\alpha)$ and $\log\Gamma(\alpha K)$ respectively. Finally, we use $\log\Gamma(n_k + \alpha) \approx \log\Gamma(n_k)$ and $\log\Gamma(n + \alpha K) \approx \log\Gamma(n)$ when alpha is close to 0. Combining these approximations, a simpler expression of Equation (5.7) as a log-linear function of α, can be derived:

$$\log p(\boldsymbol{Z} \mid \alpha, K) \approx_0 (K-1)\log(\alpha) - \log(K) + \sum_{k=1}^{K} \log\Gamma(n_k) - \log\Gamma(n)\,.$$

The algorithm introduced in this paper relies on this approximation, and the corresponding criterion is named ICL_{lin} where *lin* stands for linear:

$$\mathrm{ICL}_{lin}(\boldsymbol{Z}, \alpha) = D(\boldsymbol{Z}) + (K-1)\log(\alpha) - \log(K) + \sum_{k=1}^{K} \log\Gamma(n_k) - \log\Gamma(n)\,.$$

All quantities that do not depend on α may be grouped in an intercept:

$$I(\boldsymbol{Z}) \coloneqq D(\boldsymbol{Z}) - \log(K) + \sum_{k=1}^{K} \log\Gamma(n_k) - \log\Gamma(n)\,. \tag{5.9}$$

Then, the log-linearity of our new criterion appears explicitly:

$$\mathrm{ICL}_{lin}(\boldsymbol{Z}, \alpha) = (K-1)\log(\alpha) + I(\boldsymbol{Z})\,. \tag{5.10}$$

Naturally, the quality of this approximation depends on how small both α and αK are. For the first one, the approximation of Equation (5.8) is quite mild, even for standard α value such as 1 or $\frac{1}{2}$. As for αK, while its value may be relatively far from 0 for $\alpha = 1$, we verify in practice that α rapidly decreases several orders of magnitude below 1 as of the first fusion. This ensures that the approximation is correct throughout the procedure.

5.3.2 Hierarchy construction

Looking at the functional form of the previous approximation, a natural goal is to search for the Pareto front in the $(\log\alpha,\, \mathrm{ICL}_{lin}(\boldsymbol{Z},\alpha))$ plane. The latter corresponds to a set of dominating partitions with respect to ICL_{lin}, for a certain range of α values in $]0,1]$, or equivalently for a range of $\log(\alpha)$ values in $]-\infty,0]$. Formally, we define the Pareto front as:

$$P = \{(\boldsymbol{Z}^\star, I_\alpha^\star) \;:\; \forall\alpha \in I_\alpha^\star,\, \forall \boldsymbol{Z} \neq \boldsymbol{Z}^\star,\, \mathrm{ICL}_{lin}(\boldsymbol{Z}^\star,\alpha) \geq \mathrm{ICL}_{lin}(\boldsymbol{Z},\alpha)\}, \tag{5.11}$$

where $I_\alpha^\star$ are intervals of $]0,1]$. Finding this set of dominating partitions and ranges is not a trivial task. However, the difficulty is reduced if we consider a dominant partition $\boldsymbol{Z}$ for certain level α, and restrict ourselves to look for partitions that results from merges of $\boldsymbol{Z}$. Indeed, we will show that it is quite easy, for a given a partition $\boldsymbol{Z}$, to find the hyper-parameter $\alpha^\star$ and the pair (g^*,h^*) of clusters to merge, such that the obtained coarser partition $\boldsymbol{Z}_{g^*\cup h^*}$ will dominate $\boldsymbol{Z}$, along with any other partition $\boldsymbol{Z}_{g\cup h}$, over $]0,\alpha^*]$. Starting from an initial clustering $\boldsymbol{Z}^{(K)}$, these locally optimal merges can be used to build a heuristic, in the spirit of hierarchical agglomerative clustering, that will extract a sequence of nested partitions to approximate the Pareto front defined by Equation (5.11). While this heuristic is not guaranteed to extract the Pareto front, it may still provide good results, especially starting from a dominant partition, *e.g.* obtained by maximizing $\mathrm{ICL}_{ex}(\boldsymbol{Z},1)$ with the hybrid optimization algorithm introduced in the previous section. Intuitively, if a partition $\boldsymbol{Z}$ is locally dominant for some α value, there is a good chance that the next dominant partition for some $\alpha' < \alpha$ will be a coarse version of $\boldsymbol{Z}$. Indeed, to surpass a dominant solution in α', the new dominant solution must be coarser in order to benefit from a reduced decreasing slope, while it must also have a high intercept $I(\boldsymbol{Z}')$. Solutions built from merging two clusters of $\boldsymbol{Z}$ are coarser, therefore fulfilling the first requirement. Moreover, since $\boldsymbol{Z}$ is already dominant, we may also hope that a coarser version of it also has a high intercept, and therefore dominates other partitions for this new α' value. Let us therefore detail this heuristic, and the conditions under which a fusion opportunity exists.

5.3.2.a Fusion opportunity

For any given partition $\boldsymbol{Z}^{(k)}$, with $k \geq 2$ clusters, let us define $\mathbb{Z}^{(k-1)}$ as the space of all the partitions with $(k-1)$ clusters that are coarser than $\boldsymbol{Z}^{(k)}$:

$$\mathbb{Z}^{(k-1)} = \left\{ \boldsymbol{Z}_{g\cup h} \;:\text{the partition } \boldsymbol{Z}^{(k)} \text{ with clusters g and h merged, } g \neq h \right\}.$$

Note that we will use the terminology *mother* partition for $\boldsymbol{Z}^{(k)}$ and *child* partition for any element of $\mathbb{Z}^{(k-1)}$.

As pointed out previously, with $\boldsymbol{Z}^{(k)}$ fixed, the function $\mathrm{ICL}_{lin}(\boldsymbol{Z}^{(k)},\,\cdot)$ is log-linear with slope $(k-1)$ and intercept $I(\boldsymbol{Z}^{(k)})$. This implies that the slope of the ICL_{lin} functions decreases incrementally to 0 as k decreases to 1. Figure 5.3 illustrates this behavior of ICL_{lin}, with respect to the number of clusters k. It can easily be seen that the slopes decrease until k reaches 1 which corresponds to an horizontal line.

From Equation (5.10) we are able to derive the expression of the variation of the ICL_{lin}

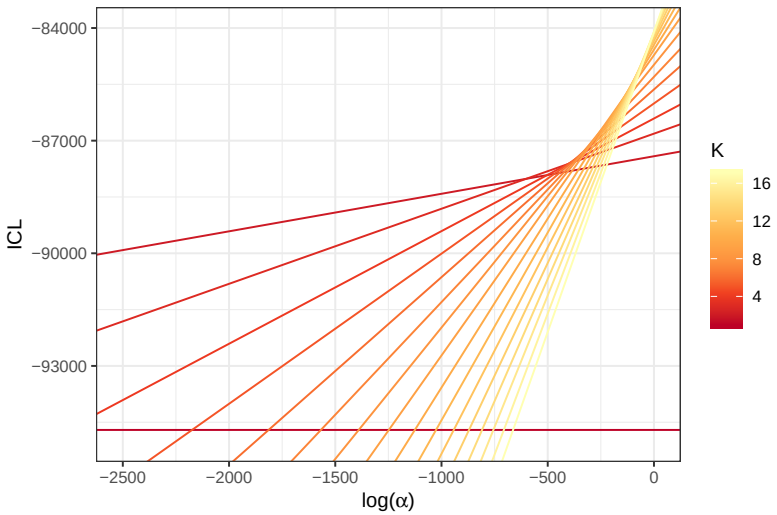


Figure 5.3: Lines of slope $k-1$ representing the functions $\log\alpha \mapsto ICL(\boldsymbol{Z}^{(k)}, \log\,\alpha)$ for a collection of partitions $\boldsymbol{Z}^{(k)}$ with a decreasing number of clusters $k = 21, \ldots, 1$. We see that the ICL_{lin} order changes as α decreases, favoring coarser partitions. The x-axis slice at $\log\alpha = 0$ corresponds to the intercepts $I(\boldsymbol{Z}^{(k)})$.

between a mother partition $\boldsymbol{Z}^{(k)}$ and any of its child $\boldsymbol{Z}_{g\cup h}$ as a function of α :

$$\begin{aligned}\Delta_{g\cup h}(\alpha) &= \mathrm{ICL}_{lin}\left(\boldsymbol{Z}_{g\cup h}, \alpha\right) - \mathrm{ICL}_{lin}\left(\boldsymbol{Z}^{(k)}, \alpha\right) , \\ &= -\log(\alpha) \;+\; I(\boldsymbol{Z}_{g\cup h}) - I(\boldsymbol{Z}^{(k)}) . \end{aligned} \tag{5.12}$$

Graphically, $\log\alpha \mapsto \Delta_{g\cup h}(\log\alpha)$ is the difference between two straight lines, of slope $k-2$ and $k-1$ respectively. Moreover, the zero of Equation (5.12) can be easily derived and will be denoted by $\alpha_{g,\,h}$:

$$\Delta_{g\cup h}(\alpha_{g,\,h}) = 0 \iff \log(\alpha_{g,\,h}) \coloneqq I(\boldsymbol{Z}_{g\cup h}) - I(\boldsymbol{Z}^{(k)}) . \tag{5.13}$$

In geometric terms, we know that below this level, the child partition $\boldsymbol{Z}_{g\cup h}$ dominates its mother $\boldsymbol{Z}^{(k)}$ in terms of ICL_{lin}. Thus, for any mother partition $\boldsymbol{Z}^{(k)}$, we are capable of computing the tipping points $(\alpha_{g,h})_{g<h}$ for the $\frac{k(k-1)}{2}$ possible child partitions. To find the best fusion, we recall the form of the ICL_{lin} for any partition $\boldsymbol{Z}_{g\cup h} \in \mathbb{Z}^{(k-1)}$ from Equation (5.10):

$$\mathrm{ICL}_{lin}(\boldsymbol{Z}_{g\cup h}, \alpha) = (k-2)\log(\alpha) \;+\; I(\boldsymbol{Z}_{g\cup h}) , \quad \forall g, h .$$

So it is clear that, viewed as functions of $\log\alpha$, the ICL_{lin} of all child partitions in $\mathbb{Z}^{(k-1)}$ are parallel straight lines of slopes $(k-2)$, only differing by their intercepts. This guarantees us that there exists a unique partition, uniformly dominating in α, in $\mathbb{Z}^{(k-1)}$. Formally:

$$\begin{aligned}\exists!\boldsymbol{Z}_{g^\star\cup h^\star} \in \mathbb{Z}^{(k-1)} \text{ s.t. } : \forall\alpha > 0, \forall \boldsymbol{Z}_{g\cup h} \in \mathbb{Z}^{(k-1)} \\ \mathrm{ICL}_{lin}\left(\boldsymbol{Z}_{g^\star\cup h^\star}, \alpha\right) \geq \mathrm{ICL}_{lin}\left(\boldsymbol{Z}_{g\cup h}, \alpha\right) .\end{aligned} \tag{5.14}$$

This partition corresponds to the one with the greatest intercept which, by Equation (5.13), also happens to be the one intersecting with $\boldsymbol{Z}^{(k)}$ at the greatest $\alpha_{g,h}$:

$$(g^\star, h^\star) = \arg\max_{g,h} I(\boldsymbol{Z}_{g\cup h}) = \arg\max_{g,h} I(\boldsymbol{Z}_{g\cup h}) - I(\boldsymbol{Z}^{(k)}) = \arg\max_{g,h} \alpha_{g,h}.$$

This discussion describes how to find the best fusion, going from a partition $\boldsymbol{Z}^{(k)}$ to $\boldsymbol{Z}^{(k-1)} = \boldsymbol{Z}_{g^\star \cup h^\star}$ by setting $\alpha^{(k-1)} = \alpha_{g^\star,h^\star}$. Taking this greedy approach, one may perform such locally optimal merges sequentially in a fast and efficient bottom-up procedure until all clusters have been merged into a unique cluster. Hence, we can see how α acts as a regularization parameter, enabling for fusions. Taking an initial partition $\boldsymbol{Z}^{(K)}$ and a given initial $\alpha^{(K)}$, typically 1, this will provide a set of nested clustering solutions $(\boldsymbol{Z}^{(k)}, \alpha^{(k)})_{k=K,\dots,1}$.

5.3.2.b Post-processing

The previous strategy outputs a hierarchy, meaning a set of nested clustering with a number of clusters ranging from K to 1. Each merge performed by the algorithm is stored into a binary tree, keeping track of the hierarchical relations between clusters. However, one important point to observe is that some of the partitions extracted by this agglomerative greedy algorithm may not be dominant anywhere in $\alpha \in]0, 1]$, with respect to the others. This corresponds to situations where combining several merges in one step is better than performing them sequentially. Indeed, in geometrical terms, there is no guarantee that the intersection between the ICL_{lin} of $\boldsymbol{Z}^{(k)}$ and $\boldsymbol{Z}^{(k-1)}$ is at a greater α than between $\boldsymbol{Z}^{(k)}$ and $\boldsymbol{Z}^{(k-2)}$. Or, equivalently, there is no guarantee that the sequence $(\alpha^{(k)})_k$ is non-increasing. This is quite natural since ICL_{lin} is a penalized criterion, thus it does not necessarily increase with the model complexity. Since such partitions cannot belong to the approximated Pareto front, we propose to remove them. Indeed, they are easy to track since they correspond to merge k where $\alpha^{(k-1)} > \alpha^{(k)}$. Then, having extracted the $F \leq K$ dominating partitions, it is possible to recompute the α_f where they cross each other to get a sequence $(\boldsymbol{Z}_f, \alpha_f)_{f=F,\dots,1}$ with a non-increasing sequence $(\alpha_f)_f$. Although the index of $\boldsymbol{Z}_f$ does not indicate its number of clusters anymore, the sequence still consists in a hierarchy of nested partitions, which are now ordered in terms of ICL_{lin} in their ranges of dominance: $\text{ICL}_{lin}(\boldsymbol{Z}_f, \alpha) > \text{ICL}_{lin}(\boldsymbol{Z}_l, \alpha)$, $\forall\, f \neq l$, $\forall \alpha \in [\alpha_{f-1}, \alpha_f]$. Figure 5.4 illustrates this post-processing, where the ICL_{lin} lines associated with each $\boldsymbol{Z}^{(k)}$ extracted by the greedy agglomerative algorithm are depicted with their corresponding dominance ranges, and the nowhere dominant partitions are highlighted.

5.3.2.c Visualization

Along with its property discussed above, the proposed algorithm possesses interesting graphical features for the visualization of both the hierarchy, with a dendrogram, as well as the initial clustering $\boldsymbol{Z}^{(K)}$ using the partial ordering of the leaves.

DENDROGRAM The sequence $(\alpha_f)_{f=F,\dots,1}$ may be used for the construction of a dendrogram representing the cluster merge tree from $\boldsymbol{Z}^{(K)}$ to $\boldsymbol{Z}^{(1)}$, with the non-increasing sequence $(-\log(\alpha_f))_f$ in the y-axis. Thus, the hierarchical structures of the clusters can be visualized as well as the amount of regularization needed for each fusion(s). Indeed, as discussed above, the y-axis can then be seen as the drop in ICL_{lin} induced by each merge, acting as an analog of the traditional *dissimilarity* in agglomerative strategies. Figure 5.5 presents the obtained dendrogram for the Book network of Section 5.5.3.

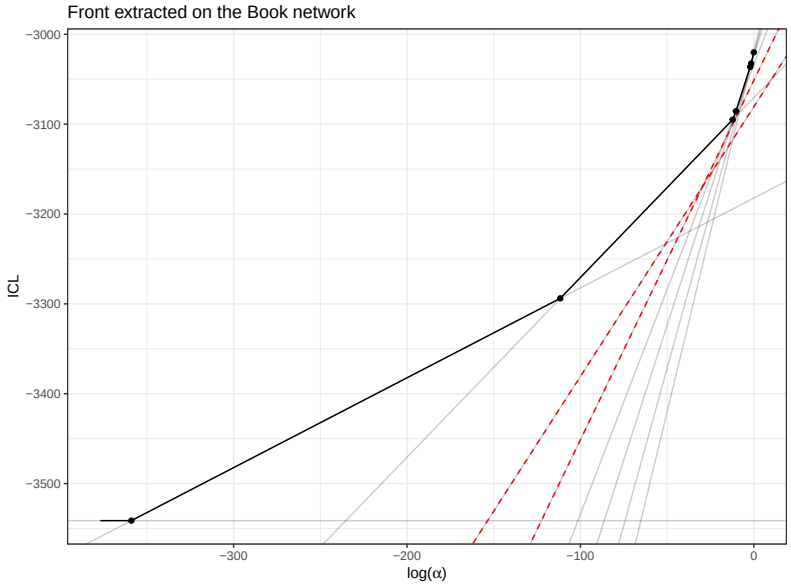


Figure 5.4: $\mathrm{ICL}_{lin}(\boldsymbol{Z}, \alpha)$ as a function of $\log(\alpha)$ for every partition extracted by the greedy hierarchical algorithm on the **Books** co-purchasing network (see Section 5.5.3 for dataset details), with a dc-SBM model. The partitions that do not have any range of dominance are highlighted with dashed red lines, and the dominant ranges with solid black lines. The intersection between dominant partitions correspond to the recomputed tipping, dominance shifting points $(\alpha_f)_f$. The initial partition $\boldsymbol{Z}^{(K)}$ was built using Algorithm 4.

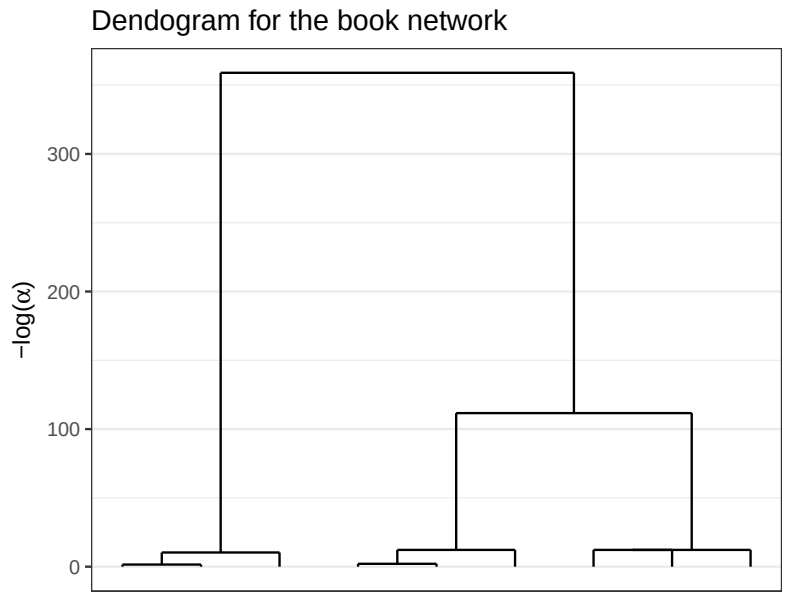


Figure 5.5: Dendrogram representation of the extracted hierarchy for the **Books** co-purchasing network (see Section 5.5.3 for dataset details).

LEAVES ORDERING Another interesting feature of the proposed procedure is the partial ordering of the initial clustering $\boldsymbol{Z}^{(K)}$ that can be obtained from the merge tree structure. Indeed, for a binary tree with K leaves, there are 2^{K-1} permutations of its leaves that are compatible with its structure. In other words, there are 2^{K-1} possible dendrograms representing the same hierarchy. However, some are more relevant than others and we seek to find the optimal tree consistent ordering (or permutation) σ that minimizes the sum of merge costs between successive clusters at $\alpha = 1$:

$$\sigma = \arg\min_{\sigma} \sum_{k=1}^{K-1} \Delta_{\sigma(k)\cup\sigma(k+1)}. \tag{5.15}$$

An efficient algorithm based on dynamic programming (Bar-Joseph et al. 2001) is already available to solve this optimization problem. As shown in Figure 5.1, such ordering of the initial clusters may be used advantageously to draw node-link diagrams or block adjacency matrix, enhancing visualization and simplifying the interpretation of the clustering results. This approach is used in the **greed** package to provide the final ordering of the clusters.

5.4 Deriving exact ICL: application to some DLVMs

So far, the discussion has been purposely general in order to express the generic aspect of the proposed methodologies. The following section discusses the detail of ICL_{ex} derivation for some discrete latent variable models introduced in Chapter 2. As defined in Equation (5.6), the only quantity needed to explicit a particular model is $\log p(\boldsymbol{Y}|\boldsymbol{Z}, \boldsymbol{\beta})$, namely the supposed generative model at hand in Equation (5.1).

5.4.1 Mixture of multinomials

As discussed in Chapter 1, multivariate count data arise in many scientific fields in the form of frequency counts, such as word occurrence in text analysis, read counts in RNAseq data, or species abundance data in ecology. Formally, an observation $\boldsymbol{y}_i$ is supposed to be a count vector in $\mathbb{N}^p$, where y_{ij} represents the count of modality j, with total count $c_i = \sum_{j=1}^{p} y_{ij}$. Here, we consider the mixture of multinomials (MoM) model which was introduced in Section 2.4.2 for the clustering of discrete data. In a Bayesian context, we define a symmetric conjugate Dirichlet prior on each parameter $\boldsymbol{\theta}_k$ and the generative model of Equation (5.2) is given by:

$$\begin{aligned} \boldsymbol{\theta}_k &\sim \mathcal{D}_p(\boldsymbol{\beta} = (\beta, \ldots, \beta)), \\ \boldsymbol{y}_i | z_{ik} = 1, \boldsymbol{\theta} &\sim \mathcal{M}_p(c_i, \boldsymbol{\theta}_k). \end{aligned} \tag{5.16}$$

Then, each parameter $\boldsymbol{\theta}_k$ can be marginalized out exactly, giving a Dirichlet-multinomial distribution (Minka 2000) per cluster.

Proposition 5.1 (Proof in Appendix C.2 on page 149)**.** *Under the mixture of multinomials model of Equation* (5.16)*, we have:*

$$\log p(\boldsymbol{Y}|\boldsymbol{Z}) = \sum_{k} \log \left(\frac{\Gamma(\beta p) \prod_{j=1}^{p} \Gamma(o_{kj} + \beta)}{\Gamma(\beta)^p \, \Gamma(c_k + \beta p)} \right) + \log B(\boldsymbol{Y}), \tag{5.17}$$

with $o_{kj} = \sum_{i=1}^{n} z_{ik} y_{ij}$, $c_k = \sum_{j=1}^{p} o_{kj}$ *and* $B(\boldsymbol{Y})$ *is a constant that does not depend on* $\boldsymbol{Z}$ *or* $\boldsymbol{\beta}$.

Tessier et al. (2006) and Biernacki et al. (2010) analogously derived an exact ICL criterion for the latent class model (LCM) which is closely related to MoM. The LCM model also fits in the proposed framework. The derivation of greedy updates for merge or swap moves does not present difficulties for these models. As for setting the β hyper-parameter, uninformative prior or Jeffreys prior can be used by setting β to 1 or $\frac{1}{2}$.

5.4.2 Stochastic block models and degree correction

We now describe the derivations for the standard binary SBM of Section 2.2.2.a as well as its degree-corrected variant.

BINARY SBM In the binary SBM framework, y_{ij} are Bernoulli random variables indicating the presence or absence of an edge. As mentioned above, the probability of a connection between the nodes i and j only depends on their cluster assignments $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$. Hence, there is a connection probability parameter θ_{kl} for each pair of clusters. Ultimately, a Bayesian formulation of SBM is given by:

$$\begin{aligned} \theta_{kl} &\sim \text{Beta}(\eta^0, \zeta^0), \\ y_{ij} | z_{ik} z_{jl} = 1, \boldsymbol{\theta} &\sim \mathcal{B}(\theta_{kl}), \end{aligned} \tag{5.18}$$

where the Beta prior on the connection probabilities is used as a conjugate of the Bernoulli distribution with hyper-parameter $\boldsymbol{\beta} = (\eta^0, \zeta^0)$. Côme and Latouche (2015) derived an exact ICL criterion for this model, relying on Beta-Bernoulli conjugacy.

Proposition 5.2 (Proof in Côme and Latouche (2015), Appendix A)**.** *Under the SBM model, we have:*

$$\log p(\boldsymbol{Y}|\boldsymbol{Z}) = \sum_{k,l} \log \left(\frac{\Gamma(\eta^0 + \zeta^0)\Gamma(\eta_{kl})\Gamma(\zeta_{kl})}{\Gamma(\eta^0)\Gamma(\zeta^0)\Gamma(\eta_{kl} + \zeta_{kl})} \right), \tag{5.19}$$

with $\eta_{kl} = \eta^0 + \sum_{i \neq j} z_{ik} z_{jl} y_{ij}$ *and* $\zeta_{kl} = \zeta^0 + \sum_{i \neq j} z_{ik} z_{jl} (1 - y_{ij})$.

Again, a commonly accepted value for setting the hyper-parameter $\boldsymbol{\beta}$ is $\eta^0 = \zeta^0 = 1$ or $1/2$, for a uniform or Jeffreys prior respectively.

DEGREE CORRECTION Real world networks tend to exhibit a specific degree distribution, with some nodes having a number of links greatly superior to the average. In the SBM, all nodes inside a cluster are statistically equivalent, hence a simple SBM model may have some difficulty in reproducing such heterogeneous degree distributions. Karrer and Newman (2011) proposed a slight modification of the SBM to respect the degree sequences of the observed graph. It can be expressed as an SBM generative model, where the connection probability between two nodes now also depends on node parameters $\boldsymbol{\Phi}$ in order to introduce disparity between the nodes. This new model is called degree-corrected stochastic block model (dc-SBM). We introduce a slightly more general version of this model for directed graphs similar to the model introduced in Zhu et al. (2014), where the parameters $\boldsymbol{\Phi}^-$ and $\boldsymbol{\Phi}^+$ govern the out-degree and in-degree distributions of nodes respectively. Then, defining

the degree prior distributions as in Newman and Reinert (2016) and Riolo et al. (2017), the model writes as follows:

$$\begin{aligned} \boldsymbol{\Omega}_{kl} &\sim \mathcal{E}(\beta^{-1}), \\ \boldsymbol{\Phi}_k^+,\, \boldsymbol{\Phi}_k^- \mid \boldsymbol{Z} &\sim \mathcal{U}(\mathbb{S}_k), \\ y_{ij} | z_{ik} z_{jl} = 1, \boldsymbol{\Omega}, \boldsymbol{\Phi} &\sim \mathcal{P}(\Phi_i^- \Omega_{kl} \Phi_j^+). \end{aligned} \tag{5.20}$$

Here, $\boldsymbol{\Phi}_k^{\cdot} = (\phi_i^{\cdot})_{i:z_{ik}=1}$, and $\mathbb{S}_k = n_k \Delta_{n_k}$ the rescaled simplex of dimension $(n_k - 1)$ induced by the constraints $\sum_i \phi_i^{\cdot} z_{ik} = n_k$. The latter must be set for the model to be identifiable. In this model the Bernoulli distribution of edges is replaced by a Poisson, in part to ease the computations, and the exponential distribution is used to leverage standard Gamma-Poisson conjugacy as $\mathcal{E}(\beta^{-1}) = \Gamma(1, \beta)$. This model may therefore handle multi-edges as well as standard graphs, a Poisson with a small mean making a good approximation for the Bernoulli (Zhao et al. 2012). The ICL_{ex} of this model can be derived thanks to the conjugacy between exponential and Poisson distribution, and after some calculus reported in the Appendix.

Proposition 5.3 (Proof in Appendix C.3 on page 150)**.** *Under the dc-SBM model we have:*

$$\begin{aligned} \log p(\boldsymbol{Y}|\boldsymbol{Z}) = &\sum_k \log \left(\frac{(n_k - 1)!\, n_k^{dg_k^+}}{(n_k + dg_k^+ - 1)!} \frac{(n_k - 1)!\, n_k^{dg_k^-}}{(n_k + dg_k^- - 1)!} \right) \\ &+ \sum_{k,l} \log \left(\frac{(\nu_{kl})!\, \beta^{\nu_{kl}}}{(\beta n_k n_l + 1)^{\nu_{kl}+1}} \right) + \log B(\boldsymbol{Y}), \end{aligned} \tag{5.21}$$

where $\nu_{kl} = \sum_{i,j} z_{ik} z_{jl} y_{ij}$ is the total counts in block (k, l), $d_i^- = \sum_j y_{ij}$ and $d_j^+ = \sum_i y_{ij}$ correspond to node i out-degree and in-degree respectively, and dg_k^-, dg_k^+ to their sums in cluster k. $B(\boldsymbol{Y})$ is a constant detailed in the Appendix, that does not depend on $\boldsymbol{Z}$ or $\boldsymbol{\beta}$.

Contrary to the previous models where proper Jeffreys or uniform prior could be used, the exponential distribution does not admit a conventional non-informative prior. An acceptable solution to fix β is, however, proposed in Newman and Reinert (2016), where the authors use the mean connection probability of the network. From a practical point of view, deriving swap and merge updates is also quite easy for these models, even though some care is needed to avoid unnecessary computations (Côme and Latouche 2015) and can be done efficiently using sparse matrices.

5.4.3 Co-clustering and latent block model

Co-clustering aims at clustering simultaneously the rows and columns of a data matrix $\boldsymbol{Y}$ of size $n \times p$ into homogeneous groups. For example, in text analysis one may be interested into grouping documents and words together. Section 2.2.2.c introduced the latent block model (LBM, Govaert and Nadif 2010), a popular generative model to perform such task, forming a flexible class of models depending on the supposed observational model (Wyse et al. 2017). The main feature of the LBM is its block generation hypothesis:

$$\begin{aligned} &\boldsymbol{z}_i^r \sim \mathcal{M}_{K_r}(1, \boldsymbol{\pi}^r),\ \boldsymbol{z}_j^c \sim \mathcal{M}_{K_c}(1, \boldsymbol{\pi}^c), \\ &y_{ij} \mid z_{ik}^r z_{jl}^c = 1, \boldsymbol{\theta} \sim p(\cdot \mid \boldsymbol{\theta}_{kl}). \end{aligned}$$

Here, $\boldsymbol{Z}^r$ and $\boldsymbol{Z}^c$ are binary matrices defining a partition of the n rows in K_r clusters and of the p columns into K_c clusters respectively. The LBM may be handled similarly as other DLVMs, with a slight variation of the prior to handle the bipartition aspect:

$$p(\boldsymbol{\pi} \mid \alpha) = \mathcal{D}_{K_r}(\boldsymbol{\pi}^r \mid \alpha) \times \mathcal{D}_{K_c}(\boldsymbol{\pi}^c \mid \alpha). \tag{5.22}$$

With such a prior, the likelihood of the bipartition integrated with respect to $\boldsymbol{\pi}$ is factorized $p(\boldsymbol{Z} \mid \alpha) = p(\boldsymbol{Z}^r \mid \alpha)p(\boldsymbol{Z}^c \mid \alpha)$ and writes as:

$$p(\boldsymbol{Z} \mid \alpha) = \frac{\Gamma(\alpha K_r) \prod_{k=1}^{K_r} \Gamma(\alpha + n_k)}{\Gamma(\alpha)^{K_r}\, \Gamma(n + \alpha\, K_r)} \times \frac{\Gamma(\alpha K_c) \prod_{l=1}^{K_c} \Gamma(\alpha + m_l)}{\Gamma(\alpha)^{K_c}\, \Gamma(p + \alpha\, K_c)}. \tag{5.23}$$

Again, this part is common to any LBM, and independent on the observational model at hand. Thus, the only quantity needed to derive ICL_{ex} for the LBM is $\log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\beta})$. The latter is often explicit when working with standard distributions for y_{ij}, leveraging on known conjugacy results. This is notably the case for standard discrete data distributions using Beta-Bernoulli or Gamma-Poisson conjugacy. Other types of distributions may be considered, *e.g.* for continuous data y_{ij}, and exponential family distributions are good candidates to derive natural conjugate priors on $\boldsymbol{\theta}_{kl}$.

Moreover, as already emphasized, the SBM and LBM are very similar and a degree-corrected LBM can also be derived for discrete Poisson observations as follows:

$$\begin{aligned} \boldsymbol{\Omega}_{kl} &\sim \mathcal{E}(\beta^{-1}), \\ \boldsymbol{\Phi}^r_k \mid \boldsymbol{Z}^r &\sim \mathcal{U}(\mathbb{S}_k), \\ \boldsymbol{\Phi}^c_l \mid \boldsymbol{Z}^c &\sim \mathcal{U}(\mathbb{S}_l), \\ y_{ij} | z^r_{ik} z^c_{jl} = 1, \boldsymbol{\Omega}, \boldsymbol{\Phi}^r, \boldsymbol{\Phi}^c &\sim \mathcal{P}(\Phi^r_i \Omega_{kl} \Phi^c_j). \end{aligned} \tag{5.24}$$

Then, an ICL_{ex} can be derived which closely resembles the one of Proposition 5.3, using similar arguments and calculations.

Proposition 5.4 (Proof in Appendix C.4 on page 152)**.** *Under the dc-LBM model we have:*

$$\begin{aligned} \log p(\boldsymbol{Y}|\boldsymbol{Z}) = &\sum_k \log\left(\frac{(n_k-1)!\, n_k^{r_k}}{(n_k + r_k - 1)!}\right) + \sum_l \log\left(\frac{(m_l-1)!\, m_l^{c_l}}{(m_l + c_l - 1)!}\right) \\ &+ \sum_{k,l} \log\left(\frac{\nu_{kl}!}{(\beta n_k m_l + 1)^{\nu_{kl}+1}}\right) + \log B(\boldsymbol{Y}), \end{aligned} \tag{5.25}$$

where $\nu_{kl} = \sum_{i,j} z^r_{ik} z^c_{jl} y_{ij}$*. Here,* $r_k = \sum_{ij} z^r_{ik} y_{ij}$ *and* $c_l = \sum_{ij} z^c_{jl} y_{ij}$ *correspond to row and column cluster degrees, and* $B(\boldsymbol{Y})$ *is a constant detailed in the Appendix, that does not depend on* $\boldsymbol{Z}$ *or* $\boldsymbol{\beta}$*.*

Merge and swap updates for dc-LBM closely resemble those of dc-SBM and can be derived in the same fashion. Moreover, the prior parameter β can be set using the same approach as for dc-SBM. However, dealing with bi-partitions induces some particular constraints for both the genetic and hierarchical algorithms. The next paragraph details how they can be extended to co-clustering.

DEALING WITH BIPARTITIONS The hybrid algorithm presented in Section 5.2 can be easily extended to co-clustering described above. In this case, we work with a partition $\mathcal{P}$ of $\{1, \ldots, n+p\}$ with the additional constraints that it decomposes into two disjoint sets of clusters that corresponds to a partition of $\{1, \ldots, n\}$ and $\{n+1, \ldots, n+p\}$ respectively (one for the rows and one for the columns):

$$\mathcal{P} = \{\boldsymbol{C}_1^r, \ldots, \boldsymbol{C}_{K_r}^r, \boldsymbol{C}_1^c, \ldots, \boldsymbol{C}_{K_c}^c\} : \begin{cases} \bigcup_k \boldsymbol{C}_k^r & = \{1, \ldots, n\}, \\ \bigcup_l \boldsymbol{C}_l^c & = \{n+1, \ldots, n+p\} \end{cases} . \tag{5.26}$$

This can be easily achieved by defining $\mathrm{ICL}_{ex}(\mathcal{P}) = -\infty$ for partitions that do not fulfill this constraint and by initializing the algorithm with admissible solutions. This is sufficient to ensure that the obtained solutions will also be compatible with the constraints, since the admissible set of partitions is closed under the crossover and mutation operations used by the algorithm.

Furthermore, the hierarchical methodology can also be extended easily to bi-partitions. Indeed, Equation (5.23) leaves a factorized integrated likelihood for $p(\boldsymbol{Z} \mid \alpha)$, with a common parameter α. Thus, the ICL_{lin} approximation of Equation (5.10) is still log-linear in α and writes:

$$\mathrm{ICL}_{lin}(\boldsymbol{Z}, \alpha) = (K_r - 1)\log(\alpha) + (K_c - 1)\log(\alpha) + I(\boldsymbol{Z}), \tag{5.27}$$

with $I(\boldsymbol{Z}) = I(\boldsymbol{Z}^r) + I(\boldsymbol{Z}^c)$ the intercepts defined in Equation (5.9). Hence, with the constraint that a merge cannot be done between rows and columns clusters, one can look for the best row or column fusion to do at each step, therefore building two dendrograms in parallel, with a shared $(\alpha_f)_f$ sequence.

5.5 Numerical experiments

Having described several popular instances of discrete latent variable models that can be handled by the proposed methodology, this section investigates the behavior of the proposed algorithms in simulated and real settings with several models. First, simulations are performed to compare the hybrid optimization algorithm with other algorithms able to handle the same task. The results of the hybrid algorithm and competitors are then compared on real datasets, prior to an analysis of the hierarchical results produced by the proposed methodology on the same datasets.

5.5.1 Medium-scale SBM simulations

To investigate the performances of the hybrid algorithm, we pursue with our motivating example defined in Section 5.1. The simulation consists of a SBM graph with 1500 nodes with 15 clusters hierarchically designed : 3 big clusters each divided into 5 small clusters. Figure 5.6 (left) presents the evolution of the ICL criterion among the different generations of solutions build by the algorithm. As clearly shown by this figure, the criterion improves at each generation until it reaches a plateau around the fourth generation. A comparison of the algorithm with other solutions is also performed on the same problem by running the different algorithms with one hundred simulated graphs. The hybrid algorithm is compared with a greedy algorithm with random starting point, a greedy algorithm with multiple random starting partitions, a regularized spectral algorithm (Qin and Rohe 2013) (which is run with the true number of clusters since it does not perform model selection), and a greedy algorithm initialized with the spectral algorithm. For all the variants of the greedy algorithm

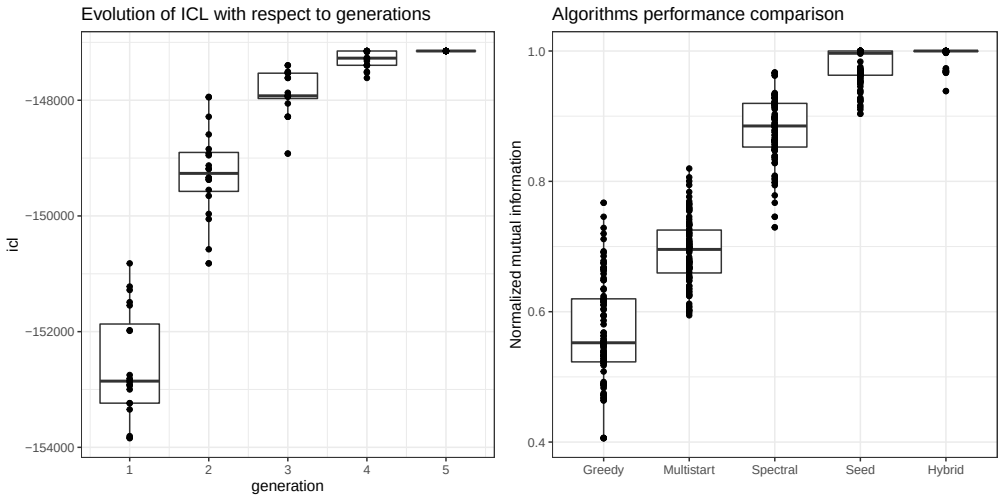


Figure 5.6: Evolution of ICL_{ex} with respect to the generation for one run of the hybrid algorithm (left), NMI between simulated and reconstructed clusters for one hundred simulations for the different algorithms (right).

and our hybrid proposal default values were used for their parameters: initial number of clusters equal to twenty, size of the population equal to fifty, probability of mutation equal to 0.25 and maximum number of generations fixed to ten. The comparison is made in terms of normalized mutual information (NMI, Vinh et al. 2010) between the extracted and simulated clusters. The NMI allows comparing partitions with a different number of clusters, as is needed in this setting, and an NMI of 1 means a perfect match between two partitions. One hundred simulated graphs with the same parameters as those given in the motivating example of Section 5.1 are used to compare the algorithms. As expected, the greedy algorithm with random starting point suffers from quite severe under-fitting and gives an NMI around 0.55, using multistart helps a little and the solutions then are around an NMI of 0.7. The spectral algorithm does also improve with an NMI around 0.85. Eventually, the two best algorithms are the simple greedy algorithm carefully initialized (here using the results of the spectral algorithm with twenty clusters) and our proposed hybrid algorithm which recovers almost perfectly the simulated partitions in all of the simulations (93% of perfect recovery) whereas some simulations are still not perfectly recovered by the greedy algorithm with careful initialization (51% of perfect recovery).

5.5.2 Medium-scale mixture of multinomials simulations

As a second scenario, we focused on a mixture of multinomials model. The simulation setup was as follows: 15 clusters with equal proportions were generated. The sample size was fixed to 500 and the number of possible outcomes for the multinomials to 100. The multinomial parameters were set such that each cluster has a uniform distribution on $\{1, \ldots, 100\}$ except for 10 randomly chosen outcomes that have their probabilities multiplied by 4. Eventually the number of draws for each multinomial sample was set to 50. The simulation was performed one hundred times and for each generated dataset the solutions found by the different variants of the greedy heuristic, an EM algorithm (from the **mixtools** R package) with model selection performed with AIC and BIC were recorded. We may first look at the number of clusters extracted by each algorithm. Figure 5.7 presents the bar graphs of the

number of extracted clusters for each of the algorithms over the 100 generated datasets.

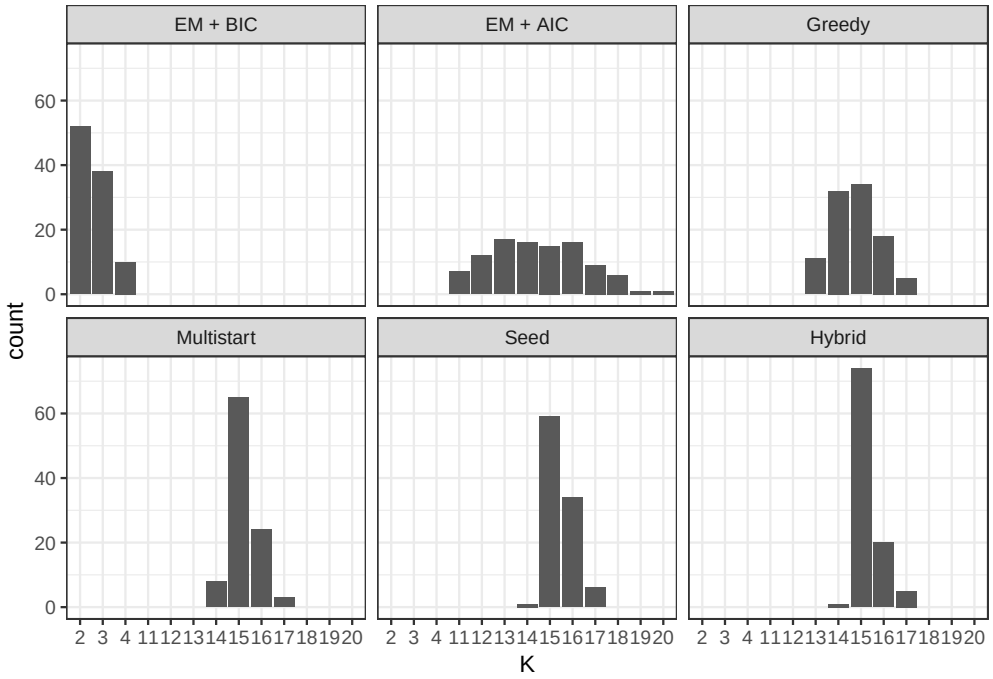


Figure 5.7: Bar graphs of the number of extracted clusters over one hundred simulated datasets for the different algorithms. The datasets were generated with $K = 15$.

The solutions found using an EM algorithm and BIC or AIC for model selection suffer from a lot of variance. AIC gives more satisfactory results on this problem but the number of extracted clusters is still quite variable, between 10 and 22. BIC leads to too simple models with fewer than 5 clusters in all the simulations. Some of these results can be explained by the random initialization of the EM algorithm. Greedy maximization of ICL gives better results in this problem and found the correct number of clusters in around 60% of the simulations with the multistart version of the algorithm (which is a little bit better than the version seeded with a simple k-means). Eventually, the hybrid algorithm found the correct number of clusters in more than 75% of the simulations and is therefore also better here. If we inspect the results with respect to the NMI with the simulated labels, or with the obtained ICL values as shown in Figure 5.8, the ranking of the different solutions does not differ. The hybrid algorithm leads to the best results even though the differences with the seeded version of the greedy algorithm are less important with respect to these metrics in this experiment.

5.5.3 Clustering real network data

The performances of the proposed solution were also investigated with real datasets. Classical graph clustering datasets were first analyzed:

- **Blog**: a directed network from Adamic and Glance (2005) of hyperlinks between 1222 blogs on US politics, recorded during the 2004 presidential election,

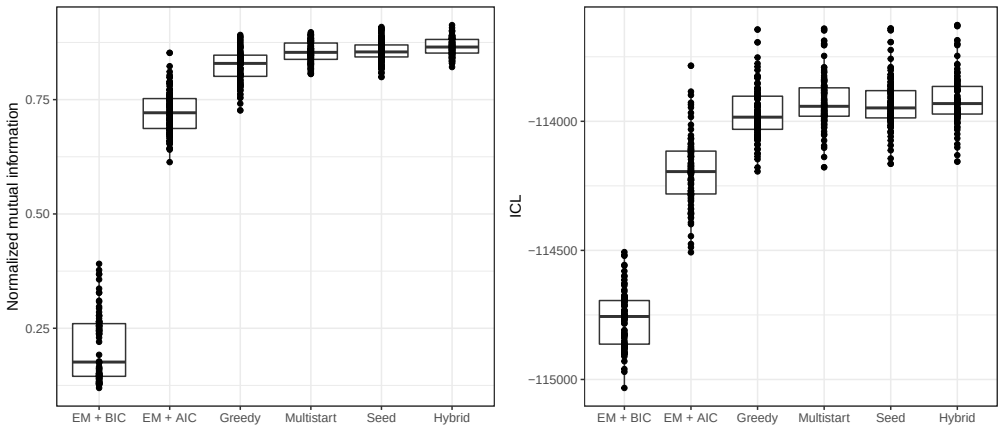


Figure 5.8: NMI between simulated and extracted clusters and ICL for the different algorithms on the mixture of multinomial simulation over one hundred simulations.

- **Books**: a network of 105 books about US politics also published around the time of the 2004 presidential election and sold by the online bookseller Amazon.com (edges between books represent frequent co-purchasing of books by the same buyers),
- **Jazz**: an undirected network of 198 jazz bands (Gleiser and Danon 2003),
- **Football**: an undirected network of American football games between 115 colleges during the regular Fall 2000 season (Newman and Girvan 2004).

All of these classical datasets were downloaded from Mark Newman datasets page[†]. Two co-clustering datasets were also benchmarked:

- **French parliament**: this dataset concerns the votes of 593 French deputies during a part of the current legislature and covers 1839 ballots, the data were extracted from the French national assembly open data api[‡] and gathered into a binary matrix where the presence of a one indicates a positive vote of a deputy for a specific ballot.
- **Jazz bands / musicians**: is a recreation of the raw data in Gleiser and Danon (2003). These raw data were extracted by scrapping the same source namely *The Red Hot Jazz Archive*[§]. For each available band, the list of its members was extracted leading to a binary matrix of 4475 musicians and 965 bands. For all the performed analyses, we removed all the musicians that played in fewer than 3 bands and all the bands with fewer than 3 musicians, leaving a final matrix of 690 musicians and 539 bands.

These datasets were produced for this chapter and are available together with the classical network datasets in the R package **greed** accompanying the chapter. For each of these datasets, and in order to get some information on the variability of the results, we ran the algorithms 25 times, with a dc-SBM model for networks and dc-LBM model for co-clustering datasets, and the resulting ICL_{ex} values were recorded. The algorithms are the same as previously: greedy with multiple random starts, seeded greedy (spectral algorithm

[†]available at http://www-personal.umich.edu/ mejn/netdata/
[‡]available at http://data.assemblee-nationale.fr/
[§]available at http://www.redhotjazz.com/

for dc-SBM and independent k-means on rows and columns for dc-LBM) and our proposed hybrid approach. To study the impact of the population size on the results of the hybrid algorithm, this parameter was also set to vary in $\{20, 40, 80\}$. These numbers are quite small with respect to the ones commonly encountered in pure GA, which is allowed by the use of hybridization with local search reducing the need for a large population.

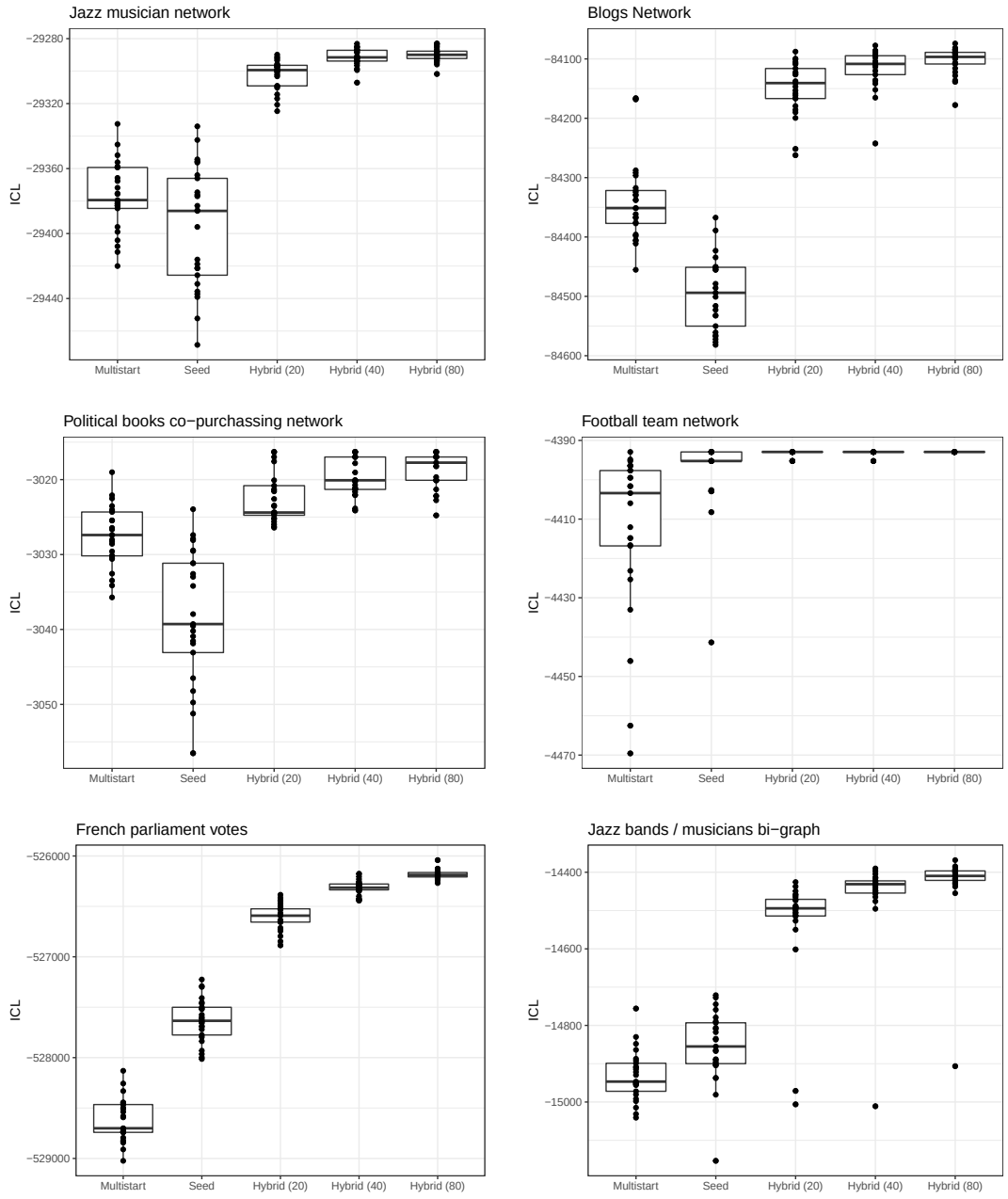


Figure 5.9: Boxplots of the ICL_{ex} values obtained from 25 runs of the different algorithms on the six different datasets.

The results are presented with boxplots in Figure 5.9. For all the datasets, the best results are achieved by the hybrid algorithm with a population of 80 partitions. For each experiment, while a bigger population size leads to better results with less variation, a small population size of 20 already achieves a significant improvement over the multiple and seeded strategies. Indeed, an important performance gap in terms of ICL_{ex} is visible between the three hybrid solutions and the two others. Moreover, some datasets like Jazz, Blogs and Political books highlights the interest of the multiple restart over the seeded strategy. This is expected for the experiments with directed networks (**Blogs**, **Books**), where the seed partitions are found using an undirected network model. Thus, it advocates for the use of directed model whenever possible for these datasets. This last experiment on the proposed hybrid algorithm clearly shows a benefit of using such an approach on real data. In the next section, we illustrate the interest of the hierarchical algorithm, giving a more detailed discussion about the clustering results on real datasets.

5.5.4 Hierarchical analysis of real datasets

In continuity with the motivating example of Figure 5.2, the interest of the hierarchical procedure is illustrated on the real datasets introduced previously. Starting from the best solution of Algorithm 4, with a population size of 40, we build the hierarchy and the dendrogram for each of the examples. We start by describing the results on the four graphs datasets, then detailing the French parliament votes co-clustering one.

NEWTORK CLUSTERING Figure 5.10 shows the results of the proposed two-step methodology with the dc-SBM as the underlying model, highlighting its analytical and visual interest. Columns represent datasets and the first row corresponds to the adjacency matrices of each network, with the rows/columns arranged per cluster numbers and the color indicating the link density between clusters. Notice that clusters are reordered according to the leaf ordering of the dendrogram, bringing *linked* clusters next to each other, enhancing the visualization of the block clustering. Next, the second row represents the cluster node link diagram, another representation of a graph clustering where the size of nodes is proportional to cluster size and the width of arrows to link density between clusters. Once again, we use the leaf ordering provided by the binary tree. The latter is then plotted as a dendrogram in the third row, emphasizing the amount of regularization (drop in α) needed for each fusion.

A possible heuristic to spot interesting levels in the dendrogram could be to cut it at a certain level $\alpha^{(f_h)}$ where the amounts of regularization needed for the next fusion is considered too important, relatively to the amount needed for past fusions. The fourth row represents the same adjacency matrices as in the first row, except the new clustering $\boldsymbol{Z}_{f_h}$ is now used. For the Blogs network, starting from a solution with 18 clusters, the heuristic finds a lot of potential fusions for reasonable α levels, leaving 2 clusters at the selected clustering. We emphasize that the *real* number of clusters, annotated by the expert, is also 2 (conservatives and liberals). Likewise, for the Books dataset, the heuristic selects 3 clusters which is the number of different categories of books present in the data. The Football network has a more pronounced and balanced community structure, with the initial partition $\boldsymbol{Z}^{(12)}$ near the ground truth number of clusters, which is 11, thus explaining the relatively regular jump distribution in α. The heuristic cuts the dendrogram after the second fusion at 10 clusters. As for the Jazz network, it starts with 21 clusters and we propose to cut at 4 clusters according to the heuristic, with the corresponding $\boldsymbol{Z}^{(4)}$ presenting an interesting block structure. Overall, this highlights the relevance of the proposed hierarchical

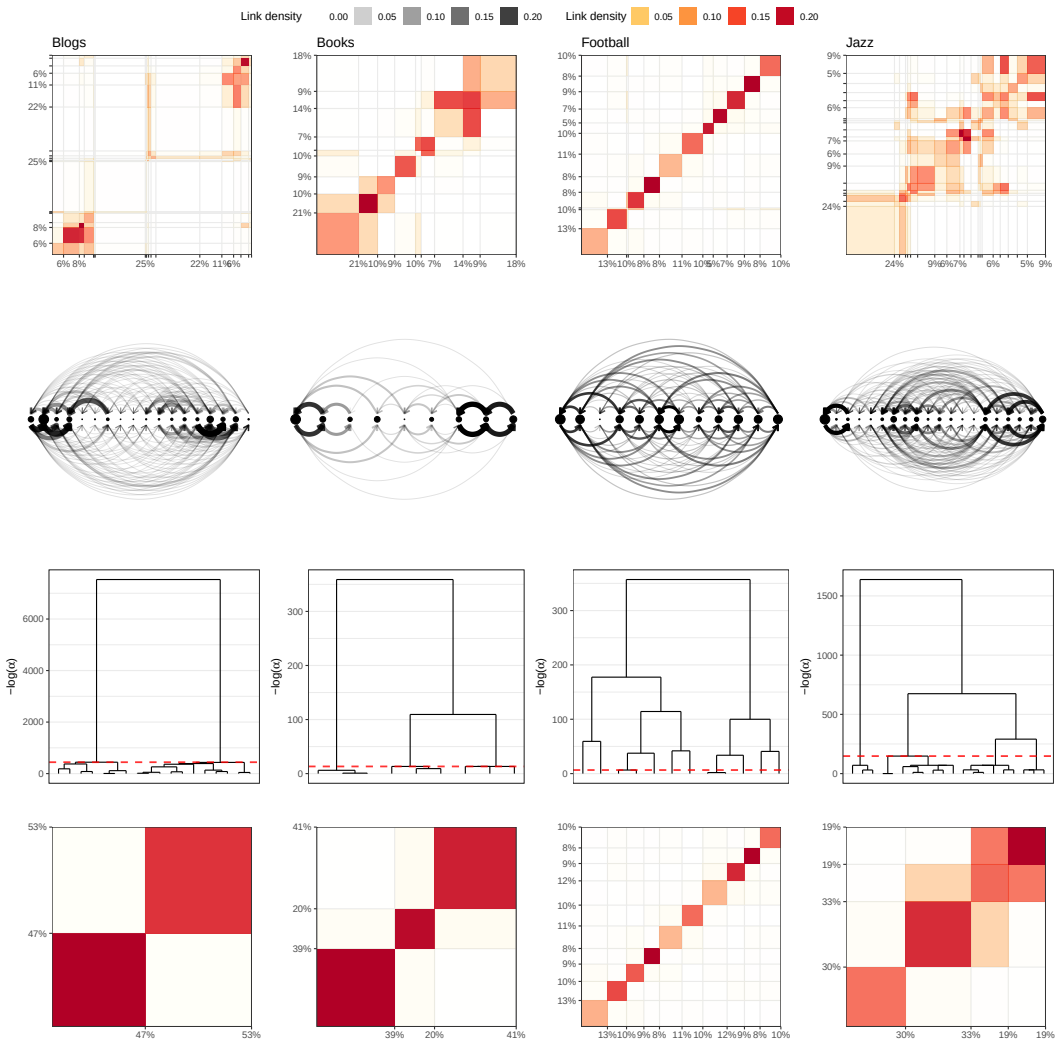


Figure 5.10: Illustration of the hierarchical agglomerative strategy on four real networks: blogs, books, football and jazz. First row: aggregated adjacency matrix according to the initial partition $\boldsymbol{Z}^{(K)}$, with cluster reordering given by the leaf ordering of the dendrogram. Second row: node link diagram of $\boldsymbol{Z}^{(K)}$. Third row: dendrogram of the hierarchy extracted from the initial partition. Fourth row: exploration of some clustering $\boldsymbol{Z}^{(f_h)}$ alongside the hierarchy.

agglomeration in term of clustering quality and interpretability as well.

CO-CLUSTERING ON FRENCH ASSEMBLY VOTES We illustrate the hierarchical heuristic on the French assembly votes co-clustering dataset. The initial partition $\boldsymbol{Z}^{(K)}$ found by Algorithm 4 has 116 clusters divided in 70 row clusters and 46 columns clusters. These are quite large numbers for a dataset of this size, and one might want to explore solutions with fewer row clusters. As explained above, the hierarchical algorithm can build two separate dendrograms for rows and columns, which are linked by their merging sequence $(\alpha_f)_f$. Then, using the same heuristic on the sequence, we can cut both dendrogram at the same level, thus

determining a number of row and column clusters. In this example, we chose to cut at 26 clusters overall, leaving 13 rows and 13 columns clusters. Inspecting the row clustering, we found it consistent with the true labels, which are the political party memberships. Some members of Parliament (MPs) in different opposition groups from the left (communists, socialists) are gathered in a single cluster, whereas MPs from the majority group (LREM) are split into 5 different clusters, with some having centrists or right-wing opposition members. This agrees with the current separations and relationships in the French Parliament and the French political field.

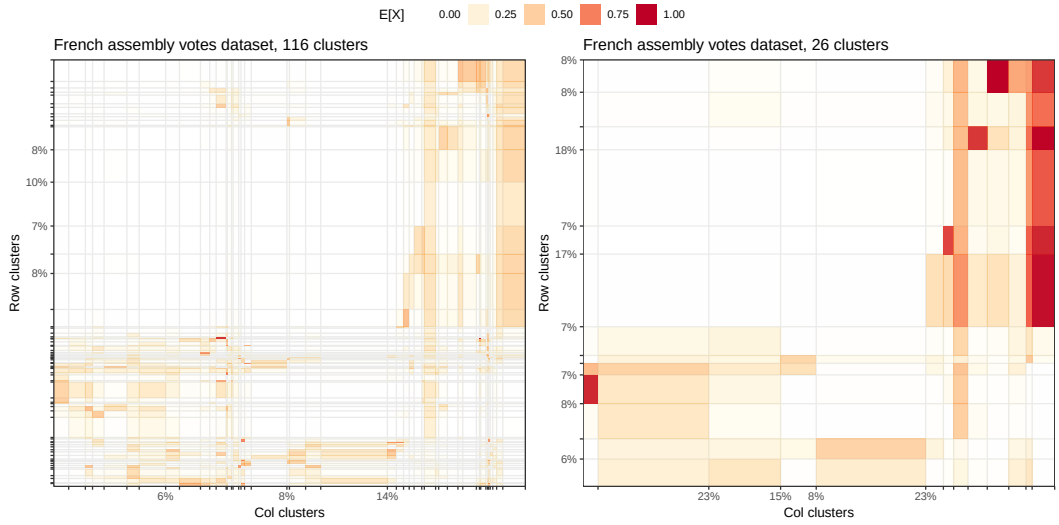


Figure 5.11: Block matrix representation of the **French Parliament** dataset after cluster reordering (left) and coarser clustering extraction (right).

5.6 Conclusion

In this chapter, we have proposed a new algorithm for clustering with discrete latent variables models, along with a hierarchical clustering algorithm to find a hierarchy of clusters. Both methods share the ICL as an objective criterion to maximize, and their interest lies on their computational efficiency as well as the wide variety of models they can be applied to. We presented some of the most common ones for discrete data or graphs clustering, as well as an extension for co-clustering. For all these models, numerical experiments assess the superiority of the clustering algorithm over existing methods. In addition, experiments on real datasets were conducted to illustrate the interest of the method in real-world applications. The hierarchical heuristic completes the methodology, giving access to coarser partitions than the one found by the genetic algorithm, by including a Dirichlet hyper-parameter α in the objective criterion. The resulting hierarchy may be visualized as a dendrogram, and explored as well as the amount of regularization needed for each fusion. Moreover, we illustrated how the leaf ordering of the dendrogram may be used to reorder clusters in the initial partition, enhancing the visualization of any clustering.

Although Gaussian mixtures fit into the DLVM framework and could be included in this work, the difficulty of setting non-informative hyper-priors must be addressed carefully,

as the clustering results is greatly influenced by these. Therefore, we leave their in-depth treatment and numerical simulations to future work.

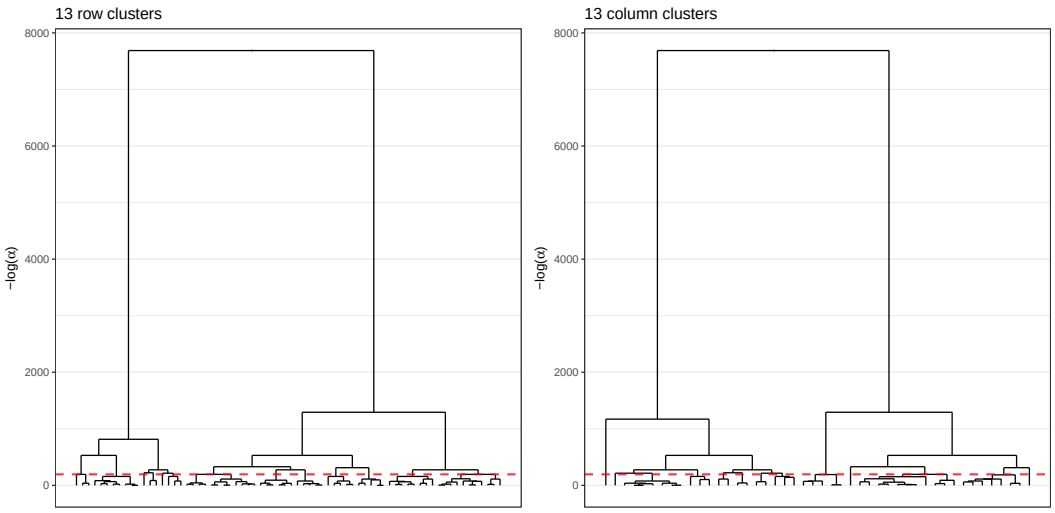


Figure 5.12: Row clusters dendrogram (left), and columns clusters dendrogram (right) for the **French Parliament** dataset. The dashed red line represents the height used to cut the dendrogram and to extract a coarser clustering.

6

Conclusion and perspectives

The first two chapters reviewed general ideas and modern challenges in model-based clustering with discrete latent variable models, which constitutes the generic framework of this thesis. While Chapters 3 to 5 contains the original contributions of this thesis, the following chapter reflects on them and discusses a series of leads currently investigated for future works.

6.1 Summary of the contributions

High-dimensional multivariate data represent a challenging task in statistics, especially for unsupervised methods such as clustering. Chapters 3 and 4 have showed how mixture modeling could benefit from the integration of dimension reduction to design robust clustering algorithms for discrete and continuous data. Chapter 5 presented a general methodology for hierarchical clustering with the exact ICL, applicable to any model for which this quantity is tractable. In particular, this approach goes beyond multivariate data, as it applies to graph clustering with stochastic block models. These clustering algorithms compared favorably to state-of-the-art methodologies on several numerical scenarios. Moreover, several applications on real data were considered, notably on medical data in collaboration with Institut Curie (Chapter 3). Finally, these works led to the development of R packages to ensure accessible and reproducible research. Every algorithm is available online: the MMPCA algorithm of

Chapter 3 via the **MoMPCA** R package on CRAN*, and the BFEM algorithm of Chapter 4 at https://github.com/nicolasJouvin/FisherEM which is written as an extension of the **FisherEM** package and will soon be available on CRAN. Etienne Côme developed an efficient implementation of the genetic and hierarchical algorithms of Chapter 5 in the **greed** R package, presently available at https://github.com/comeetie/greed.

These contributions motivated the writing of several scientific articles, among which one was published in an international peer-reviewed journal

- **Greedy clustering of count data through a mixture of multinomial PCA** (joint work with Guillaume Bataillon, Charles Bouveyron, Pierre Latouche and Alain Livartowski), *Computational Statistics* (2020),

and two were submitted

- **Hierarchical clustering with discrete latent variable models and the integrated classification likelihood** (joint work with Charles Bouveyron, Etienne Côme and Pierre Latouche), Preprint HAL-02530705 (2020),
- **A Bayesian Fisher-EM algorithm for discriminative Gaussian subspace clustering** (joint work with Charles Bouveyron and Pierre Latouche), Preprint HAL-03047930 (2020).

6.2 Future works

The proposed methodologies and frameworks lead to several research directions. Here, we detail our ongoing works on the field of high-dimensional data clustering.

6.2.1 Two extensions of the Bayesian Fisher EM algorithm

6.2.1.a Sparse extensions to BFEM and variable selection

If the dimension reduction aspect of BFEM allows tackling high-dimensional problems, interpreting the discriminative space is still a challenging problem. Indeed, the d discriminative axis are linear combinations of the p original variables, which makes the analysis of individual variable contributions hard when p is large. Simple thresholding heuristic on the loadings coefficients may be used to eliminate some low-contributing variables, although previous work advise against this approach (Cadima and Jolliffe 1995). In the case of the Fisher EM algorithm, Bouveyron and Brunet-Saumard (2014) adapted the supervised case of Qiao et al. (2009), to recover sparse loadings. The latter casts the maximization of the Fisher criterion as a regression problem, and introduce a l_1-type penalty to ensure sparsity. Such an approach is directly adaptable in our framework, and we sketch the main ideas below.

Recall the Fisher criterion introduced in Section 4.3.3, Equation (4.25):

$$F(\boldsymbol{U}) = \mathrm{Tr}\left[\left(\boldsymbol{U}^\top \boldsymbol{S}_T \boldsymbol{U}^\top\right)^{-1} \boldsymbol{U}^\top \tilde{\boldsymbol{S}}_B^{(t)} \boldsymbol{U}^\top\right].$$

*See also https://github.com/nicolasJouvin/MoMPCA

Introducing the following matrices:

$$\boldsymbol{H}_T = \frac{1}{\sqrt{n}} \left(\boldsymbol{Y} - \mathbf{1}_n \bar{\boldsymbol{y}}\right)^\top \in \mathbb{R}^{p \times n},$$
$$\tilde{\boldsymbol{H}}_B^{(t)} = \frac{1}{\sqrt{n}} \left[\sqrt{\tilde{n}_1^{(t)}} \left(\tilde{\boldsymbol{m}}_1^{(t)} - \bar{\boldsymbol{y}}\right), \ldots, \sqrt{\tilde{n}_K^{(t)}} \left(\tilde{\boldsymbol{m}}_K^{(t)} - \bar{\boldsymbol{y}}\right)\right] \in \mathbb{R}^{p \times K},$$

the following identities hold:

$$\boldsymbol{S}_T = \boldsymbol{H}_T \boldsymbol{H}_T^\top \qquad\qquad \tilde{\boldsymbol{S}}_B^{(t)} = \tilde{\boldsymbol{H}}_B^{(t)} (\tilde{\boldsymbol{H}}_B^{(t)})^\top. \tag{6.1}$$

In the supervised setting, Qiao et al. (2009) proposed a reformulation of the Fisher criterion as a regression problem.

Proposition 6.1. *(Qiao et al. 2009, Theorem 1) The optimal $\boldsymbol{U}^\star$ such that*

$$\boldsymbol{U}^\star = \arg\max_{\boldsymbol{U}} F(\boldsymbol{U}),$$

spans the same subspace as the solution $\boldsymbol{B}^{(t)}$ of the following regression problem:

$$(\boldsymbol{A}^{(t)}, \boldsymbol{B}^{(t)}) = \arg\min_{\boldsymbol{A}^\top \boldsymbol{A} = \boldsymbol{I}_d, \boldsymbol{B}} \sum_{k=1}^{K} \|\boldsymbol{R}_T^{-\top} \tilde{\boldsymbol{H}}_{B,k}^{(t)} - \boldsymbol{A}\boldsymbol{B}^\top \tilde{\boldsymbol{H}}_{B,k}^{(t)}\|_F + \rho \sum_{h=1}^{d} \boldsymbol{\beta}_h^\top \boldsymbol{S}_T \boldsymbol{\beta}_h. \tag{6.2}$$

Here, $\boldsymbol{A} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_d] \in \mathbb{R}^{p \times d}$, $\boldsymbol{B} = [\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_d] \in \mathbb{R}^{p \times d}$, $\boldsymbol{R}_T \in \mathbb{R}^{p \times p}$ is a upper triangular matrix obtained from the Cholesky decomposition of $\boldsymbol{S}_T$ (i.e. $\boldsymbol{S}_T = \boldsymbol{R}_T^\top \boldsymbol{R}_T$), $\tilde{\boldsymbol{H}}_{B,k}$ is the k-th column of $\tilde{\boldsymbol{H}}_B$, and $\rho > 0$.

The proof is detailed in Qiao et al. (2009), and involves keeping $\boldsymbol{B}$ then $\boldsymbol{A}$ fixed in the optimization to respectively solve d separated ridge regression problems in $\boldsymbol{B}$, and one projection Procruste problem in $\boldsymbol{A}$ (Gower and Dijksterhuis 2004, Chapter 5). Then, using this reformulation, l_1-type penalization may be added to Equation (6.2), inducing sparsity in $\boldsymbol{B}^{(t)}$:

$$(\boldsymbol{A}^{(t)}, \boldsymbol{B}^{(t)}) = \arg\min_{\boldsymbol{A}^\top \boldsymbol{A} = \boldsymbol{I}_d, \boldsymbol{B}} \sum_{k=1}^{K} \|\boldsymbol{R}_T^{-\top} \tilde{\boldsymbol{H}}_{B,k}^{(t)} - \boldsymbol{A}\boldsymbol{B}^\top \tilde{\boldsymbol{H}}_{B,k}^{(t)}\|_F + \rho \sum_{h=1}^{d} \boldsymbol{\beta}_h^\top \boldsymbol{S}_T \boldsymbol{\beta}_h + \lambda \sum_{h=1}^{d} \|\boldsymbol{\beta}_h\|_1.$$

Bouveyron and Brunet-Saumard (2014) used this reformulation of the problem in their F-step, using the LARS algorithm to solve the LASSO problem (Efron et al. 2004). Of course, since the orthonormality of $\boldsymbol{B}^{(t)}$ is not guaranteed in this case, an extra-step is added to obtain the F-step $\boldsymbol{U}^{(t)}$ as the best orthogonal approximation of $\boldsymbol{B}^{(t)}$:

$$\boldsymbol{U}^{(t)} = \arg\min_{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d} \|\boldsymbol{B}^{(t)} - \boldsymbol{U}\|_F. \tag{6.3}$$

This last optimization program is a nearest orthogonal Procruste problem (Gower and Dijksterhuis 2004, Chapter 4) and is solved in closed form as $\boldsymbol{U}^{(t)} = u^{(t)} v^{(t)\top}$, where the $u^{(t)}$ and $v^{(t)}$ matrices come from the SVD of $\boldsymbol{B}^{(t)} = u^{(t)} \Lambda v^{(t)\top}$.

The final loading matrix after convergence is then thresholded at some value and each axis of the discriminative subspace have only a few contributing variables. The number of free parameters in $\boldsymbol{U}$ can then be modified to take into account zero coefficients, modifying

the penalty in the ICL criterion. Based on the empirical results provided in Bouveyron and Brunet-Saumard (2014) and the improvements of Chapter 4, this sparse F-step should lead to good results. From a practical point of view, the introduction of early sparsity may lead to poor results. Thus, taking an initialization $\boldsymbol{U}^{(0)}$ provided by the standard BFEM of Chapter 4 should bypass this problem. Finally, we note that more elaborate type of regularization could be used, adding a ridge l_2-type penalty term, and replacing LARS with the elastic-net algorithm (Zou and Hastie 2005).

6.2.1.b From the trace of ratios to the ratio of traces problem

The original Fisher criterion considered the 2-class separation problem, where the data is projected in one-dimensional subspace described by some vector $\boldsymbol{u}$:

$$\boldsymbol{u} = \arg\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top \boldsymbol{S}_B \boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{S}_W \boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top \boldsymbol{S}_B \boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{S}_T \boldsymbol{w}}, \tag{6.4}$$

where the equivalence between the two optimization problems is given by the identity $\boldsymbol{S}_T = \boldsymbol{S}_B + \boldsymbol{S}_W$. Fukunaga (1990) proposed an extension to the K class separation, where $\boldsymbol{U}$ contains $d < K - 1$ discriminant vectors, and which is widely used, including in Chapter 4. This criterion is known as the trace of ratio problem (Jia et al. 2009):

$$\boldsymbol{U}_{TROP} = \arg\max_{\boldsymbol{U}} \operatorname{Tr}\left[\left(\boldsymbol{U}^\top \boldsymbol{S}_T \boldsymbol{U}^\top\right)^{-1} \boldsymbol{U}^\top \boldsymbol{S}_B \boldsymbol{U}^\top\right]. \tag{TROP}$$

Remember that without orthogonality constraints, $\boldsymbol{U}_{TROP}$ is obtained in closed form as the d leading eigenvectors of the generalized eigenvalue problem $\boldsymbol{S}_B \boldsymbol{u} = \lambda_u \boldsymbol{S}_T \boldsymbol{u}$. Adding orthogonality constraints led to the development of the so-called Foley-Sammon transform (Foley and Sammon 1975), along with algorithms such as the ODV procedure described in Section 4.3.3. The latter iteratively derives the columns of $\boldsymbol{U}_{TROP}$ as solutions of the one-dimensional problem in Equation (6.4) adding orthogonality constraints with respect to currently computed vectors.

However, this criterion is actually the relaxation of a more difficult one, the ratio of trace (Guo et al. 2003; Wang et al. 2007; Ngo et al. 2012):

$$\boldsymbol{U}_{ROTP} = \arg\max_{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d} \frac{\operatorname{Tr}\left[\boldsymbol{U}^\top \boldsymbol{S}_B \boldsymbol{U}^\top\right]}{\operatorname{Tr}\left[\boldsymbol{U}^\top \boldsymbol{S}_W \boldsymbol{U}^\top\right]} = \arg\max_{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d} \frac{\operatorname{Tr}\left[\boldsymbol{U}^\top \boldsymbol{S}_B \boldsymbol{U}^\top\right]}{\operatorname{Tr}\left[\boldsymbol{U}^\top \boldsymbol{S}_T \boldsymbol{U}^\top\right]}. \tag{ROTP}$$

The latter may be thought as an alternative generalization of the Fisher criterion to d-dimensional problems, which preserves the agnostic choice between $\boldsymbol{S}_W$ and $\boldsymbol{S}_T$ in the optimization. Several works argued that solving the (ROTP) problem leads to a subspace with better discriminative power than the (TROP) problem (Wang et al. 2007; Ngo et al. 2012). However, this new problem does not admit any closed form solution, although it is shown to be equivalent to the *trace difference* problem:

Proposition 6.2. *(Guo et al. 2003, Theorem 2) Solving Problem* (ROTP) *is equivalent to find the root (zero point) of the following trace difference function:*

$$f(\gamma) = \max_{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d} \operatorname{Tr}\left[\boldsymbol{U}^\top \left(\boldsymbol{S}_B - \gamma \boldsymbol{S}_T\right) \boldsymbol{U}\right]. \tag{6.5}$$

The optimal $\gamma^\star$ such that $f(\gamma^\star) = 0$ can then be used to find the optimal $\boldsymbol{U}_{ROTP}$ as:

$$\boldsymbol{U}_{ROTP} = \underset{\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_d}{\arg\max} \operatorname{Tr}\left[\boldsymbol{U}^\top \left(\boldsymbol{S}_B - \gamma^\star \boldsymbol{S}_T\right) \boldsymbol{U}\right].$$

Different optimization algorithms build on Proposition 6.2 and propose to search for the root of the trace difference function. Guo et al. (2003) derived a bisection heuristic starting from an interval $I = [\gamma_1, \gamma_2]$ such that $0 \in f(I)$, and refining it by dichotomy. Wang et al. (2007) proposed a naive Newton-Raphson algorithm, and theoretical analysis as well as improvements were proposed in Jia et al. (2009) and Ngo et al. (2012).

Thus, introducing the same soft matrix $\tilde{\boldsymbol{S}}_B^{(t)}$ as in Chapter 4, the F-step of BFEM could be modified to search for $\boldsymbol{U}_{ROTP}^{(t)}$ instead of $\boldsymbol{U}_{TROP}^{(t)}$. Based on the empirical observations in the supervised case, this could lead to better discriminative performances of the subspace although one needs to be careful with sensitivity to initialization in the clustering case, as the labels iteratively change in the BFEM algorithm. An implementation of the aforementioned algorithms is available in the **maotai** R package and experimentation are still ongoing since they need to be calibrated.

6.2.2 Clustering categorical data with a mixture of multinomial multiple correspondence analysis

Categorical data is a fundamental tool in medical or social sciences (Agresti 2003), often involving the design of tests or surveys with multiple-choice questions. In this context, n individuals are observed through Q categorical variables $\boldsymbol{y}_{iq}$, with a varying number of modalities p_q. The latter is often coded as a dummy variable, $\boldsymbol{y}_{iq} \in \{0,1\}^{p_q}$, where $y_{iqj} = 1$ if individual i chooses the j-th modality for the q-th question. This representation may be summed up in an indicator matrix $\boldsymbol{Y} = [\boldsymbol{Y}_1 | \dots | \boldsymbol{Y}_Q]$. Here, each matrix $\boldsymbol{Y}_q \in \{0,1\}^{n \times p_q}$ contains the indicator vectors of the q-th variable in its rows, as represented in Table 6.1.

$$\boldsymbol{Y} = \left[\begin{array}{cc|ccc|ccc} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

Table 6.1: Example of categorical data representation with $n = 10$, $Q = 3$ and $p_1 = 2$, $p_2 = p_3 = 3$.

In the following, we detail how the framework of mixture modeling integrating linear dimension reduction can be applied to such data. In addition, we sketch a variational EM algorithm for clustering, relying on a local variational bound of the LogSumExp function.

6.2.2.a Multiple correspondence analysis

Exploratory analysis of such data has a long history, especially geometric approaches in the line of the SVD approach to PCA described in Section 2.3.1.a. In particular, the "French

school" of data analysis led by Escofier-Cordier (1969) and Benzécri (1973) proposed the multiple correspondence analysis (MCA), which is a form of factor analysis decomposing the responses of individuals on principal directions retaining the most of variability in the chi-square sense. It may also be cast as a form of generalized SVD on the matrix $\boldsymbol{Y}$ after centering and a carefully chosen weighting. Introducing the column mean vector as $\boldsymbol{m}$ such that $m_{qj} = n^{-1} \sum_i \boldsymbol{y}_{iqj}$, and columns weights as $\boldsymbol{D}_{\boldsymbol{m}} = \text{diag}(m_{qj})_{qj}$, MCA solves for the SVD of the following matrix (Fithian and Josse 2017, Eq. (9)):

$$\boldsymbol{A} \coloneqq \frac{1}{\sqrt{nQ}} \left(\boldsymbol{Y} - \boldsymbol{1}\boldsymbol{m}^\top \right) \boldsymbol{D}_{\boldsymbol{m}}^{-1/2}, \qquad \boldsymbol{A} = \tilde{\boldsymbol{V}} \boldsymbol{\Lambda} \tilde{\boldsymbol{U}}^\top, \tag{6.6}$$

and takes the rank-d approximation of $\boldsymbol{A} \approx \tilde{\boldsymbol{V}}_d \boldsymbol{\Lambda}_d \tilde{\boldsymbol{U}}_d^\top$. This method may be seen as a generalization of PCA to categorical data, and provides powerful visualization properties to investigate connections between categories and identify individuals with the similar profile of responses (Pagès 2014, p. 53).

Among other examples, MCA had an important impact on Pierre Bourdieu's theory of sociological Fields (see *e.g.* Lebaron and Le Roux 2015, Chapter 3), and is still widely used in contemporary sociological analyses using quantitative methods (Desrosières 2008). A recent perspective on multiple correspondence analysis and its applications may be found in Greenacre and Blasius (2006).

6.2.2.b A probabilistic formulation: Multinomial multiple correspondence analysis

Although the original work of Benzécri was purposely constructed apart from the generative model approach[†] (Husson et al. 2016, p. 2), recent works proposed a probabilistic formulation of MCA. A basic generative model for categorical variable would be the following multinomial model:

$$p(\boldsymbol{Y} \mid \boldsymbol{\theta}_K) = \prod_{i=1}^{n} \prod_{q=1}^{Q} \mathcal{M}_{p_q}(1, \boldsymbol{\theta}_{iq}),$$

which posits that observation $\boldsymbol{y}_{iq}$ takes modality j with probability θ_{iqj}, where $\boldsymbol{\theta}_{iq} \in \Delta_{p_q}$. Without further restriction, the saturated model with free $\boldsymbol{\theta}_{iq}$ is not of any interest as it perfectly (over)fits the data. Fithian and Josse (2017) proposed the following log-bilinear model, also known as multinomial MCA (MMCA), as a rank-d constrained model on $\log(\theta_{iqj})$:

$$\forall i, q, \forall j = 1, \ldots, p_q, \ \log(\theta_{iqj}) = \beta_{qj} + \sum_{h=1}^{d} \boldsymbol{x}_{ih} \boldsymbol{u}_{qjh} + \text{const} \ . \tag{6.7}$$

Introducing $\boldsymbol{\beta}_q$, the main effect of the q-th variable, and decomposing $\boldsymbol{U}$ into Q matrices $\boldsymbol{U}^\top = [\boldsymbol{U}_1^\top | \ldots | \boldsymbol{U}_Q^\top]$, with $\boldsymbol{U}_q \in \mathbb{R}^{p_q \times d}$, the model may be written in matrix form as:

$$\boldsymbol{a}_{iq} = \beta_{qj} + \boldsymbol{U}_q \boldsymbol{x}_i, \quad \log(\theta_{iqj}) = a_{iqj} - \text{lse}(\boldsymbol{a}_{iq}) \text{ with } \text{lse}(\boldsymbol{a}) = \log \left(\sum_l e^{a_l} \right). \tag{6.8}$$

The model parameters are $\boldsymbol{\vartheta} = (\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{U})$, to be estimated by maximizing the log-likelihood of the model, which is a difficult, non-convex problem. After a second order Taylor expansion

[†]With the famous quote: "The model must follow the data, not the other way around" or, in French, "Le modèle doit suivre les données, non l'inverse." (Benzécri 1973, p. 6, Tome 2)

around $\boldsymbol{\vartheta}^{(0)} = (\log(\boldsymbol{m}), \mathbf{0}, \mathbf{0})$, Fithian and Josse (2017, Theorem 2) show that maximizing the latter is equivalent to solving the SVD problem of MCA in Equation (6.6), highlighting the connection between the geometric formulation and the probabilistic model. In addition, a Majorization-Minimization algorithm is derived in Groenen and Josse (2016), for minimizing the negative log-likelihood with an additional regularization term, penalizing for large eigenvalues of $\boldsymbol{X}\boldsymbol{U}^\top$ and avoiding overfitting.

Note that related models were proposed in the literature, considering $\boldsymbol{x}_i$ as a standard Gaussian latent variable, which leads to an intractable observed-data likelihood. In psychometrics, Moustaki and Knott (2000) introduced a latent trait model for mixed-type data with exponential family distributions, and derived an EM algorithm using Gauss-Hermite quadrature to approximate the intractable integrals. However, their work focuses on the case $d = 1$. More recently, Chiquet et al. (2018) proposed a similar model as a generalization of probabilistic PCA to exponential family distributions, with a focus on Poisson distributed observations. The inference relies on a variational EM algorithm, with an approximation of $p(\boldsymbol{X} \mid \boldsymbol{Y}; \boldsymbol{\vartheta})$.

6.2.2.c Mixture of MMCA

The interest of the generative approach in Equation (6.8) is the possibility to introduce clustering via mixture modeling. Indeed, in line with the integration of dimension reduction in mixture models described in Section 2.4, we propose to add a generative layer in $\boldsymbol{x}_i$, leaving a mixture of MMCA (M3CA):

$$\boldsymbol{x}_i \sim \sum_{k=1}^{K} \pi_k \, \mathcal{N}_d(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{M3CA}$$

Note that the subspace $\boldsymbol{U}$ is common across clusters here. The model parameters are now $\boldsymbol{\vartheta} = (\boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and a family of sub-models may be derived considering different types of constraints on $\boldsymbol{\Sigma}_k$, as in Chapter 4. As mentioned above, the observed-data likelihood now involves intractable integrals over $\boldsymbol{x}_i$:

$$p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \int_{\boldsymbol{x}_i} \prod_{q=1}^{Q} \prod_{j=1}^{p_q} \left(\frac{e^{\beta_{qj} + \boldsymbol{x}_i^\top \boldsymbol{u}_{qj}}}{\sum_{l=1}^{p} e^{\beta_{ql} + \boldsymbol{x}_i^\top \boldsymbol{u}_{ql}}} \right)^{y_{iqj}} \mathcal{N}_d(\boldsymbol{x}_i \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \, \mathrm{d}\boldsymbol{x}_i. \tag{6.9}$$

The posterior distribution $p(\boldsymbol{X}, \boldsymbol{Z} \mid \boldsymbol{Y}; \boldsymbol{\vartheta})$ is already factorized in the model, albeit intractable. Thus, one needs to explicitly constrain the functional form of the variational distribution in mean-field inference to obtain an approximation:

$$q \in \mathcal{Q} := \left\{ q(\boldsymbol{X}, \boldsymbol{Z}) = \prod_{i=1}^{N} \mathcal{N}_d(\boldsymbol{x}_i \mid \tilde{\boldsymbol{\mu}}_i, \tilde{\mathbf{M}}_i) \, \mathcal{M}_K(\boldsymbol{z}_i \mid 1, \boldsymbol{\tau}_i) \right\}. \tag{6.10}$$

The classical evidence lower bound still holds in this context:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) \geq \mathcal{J}(q; \boldsymbol{\vartheta}) = \mathbb{E}_q \left[\log p(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{Z} \mid \boldsymbol{\vartheta}) \right] + \mathrm{H}(q).$$

However, the latter is still intractable as it involves expectations of the form $\mathbb{E}_{q(\boldsymbol{x}_i)} \left[-\operatorname{lse}(\boldsymbol{a}_{iq}) \right]$, which are not in closed form. This fact is well-known in the case of binary categorical data, when $p_q = 2$, and Tipping (1999) proposed to rely on the quadratic bound of the sigmoid

function $\sigma(a) = \left(1 + e^{-a}\right)^{-1}$, originally introduced in the supervised case of Bayesian logistic regression (Jaakkola and Jordan 1997):

$$\forall a, \xi, -\log(1 + e^a) \geq \lambda(\xi)(\xi^2 - a^2) + \frac{\xi - a}{2} - \log(1 + e^{\xi}), \text{ with } \lambda(\xi) = \frac{1}{2\xi}\left(\frac{1}{1 + e^{-\xi}} - \frac{1}{2}\right). \tag{6.11}$$

Gollini and Murphy (2014) proposed a mixture of latent trait analyzers for the clustering of binary categorical data, along with a variational EM relying on this bound. This was also coined the ξ-transformation in Latouche et al. (2011, Proposition 5.2), and is a special case of *local* variational inference methods (Bishop 2006, Section 10.5) as it introduces a second lower bound:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{\vartheta}) \geq \mathcal{J}(q; \boldsymbol{\vartheta}) \geq \mathcal{J}(q; \boldsymbol{\vartheta}, \boldsymbol{\xi}). \tag{6.12}$$

A generalization of this lower bound to the multinomial case and lse functions was introduced in Bouchard (2007, Equation (5)). It is based on the fact that $-\operatorname{lse}$ can be lower-bounded by a product of sigmoid functions. Then, applying p_q times the inequality in Equation (6.11) leaves:

$$\forall i, q, \forall \boldsymbol{\xi}_{iq} \in \mathbb{R}^{p_q}, -\operatorname{lse}(\boldsymbol{a}_{iq}) \geq \sum_{j=1}^{p_q} \lambda(\xi_{iqj})(\xi_{iqj}^2 - a_{iqj}^2) + \frac{\xi_{iqj} - a_{iqj}}{2} - \log(1 + e^{\xi_{iqj}}). \tag{6.13}$$

The expectations with respect to $q(\boldsymbol{x}_i)$ of the right-hand side are then easy to compute as quadratic functions of a Gaussian, and a variational EM algorithm can be sketched:

$$\begin{aligned}
q^{(t+1)} &= \arg\max_{q \in \mathcal{Q}} \mathcal{J}\left(q; \boldsymbol{\vartheta}^{(t)}, \boldsymbol{\xi}^{(t)}\right), && \text{(VE-step)} \\
\boldsymbol{\vartheta}^{(t+1)} &= \arg\max_{\boldsymbol{\vartheta}} \mathcal{J}\left(q^{(t+1)}; \boldsymbol{\vartheta}, : \boldsymbol{\xi}^{(t)}\right), && \text{(M-step)} \\
\boldsymbol{\xi}^{(t+1)} &= \arg\max_{\boldsymbol{\xi}} \mathcal{J}(q^{(t+1)}; \boldsymbol{\vartheta}^{(t+1)}, \boldsymbol{\xi}). && \text{(local step)}
\end{aligned}$$

The computations for the M-step updates are still to be derived, especially for $\boldsymbol{\beta}$ and $\boldsymbol{U}$ although one could also resort to numerical optimization. Such an algorithm needs to be carefully calibrated, and the impact of $\boldsymbol{\xi}$ in the optimization needs to be investigated. Finally, we note that a similar model was proposed in Khan et al. (2010), including mixed-type data, although they rely on a different bound called the Bohning bound based on a Taylor series expansion of the lse around a point $\boldsymbol{a}$. Moreover, akin to the differences between MFA and MCFA (see Section 2.4.1), each cluster possesses a categorical loading matrix $\boldsymbol{U}_{kq}$, while the subspace $\boldsymbol{U}_q$ is common across clusters in our formulation.

Such a methodology could greatly improve the understanding of relationships between categorical variables, with a common visualization of the variables and clustered individuals in the same bi-plot of dimension d.

Appendices

A Appendix for Chapter 3

A.1 Constructing meta-observations

Proof of Proposition 3.1 on page 55.

$$
\begin{aligned}
p(\boldsymbol{W}, \boldsymbol{X} \mid \boldsymbol{Z}, \boldsymbol{U}) &= p(\boldsymbol{X}) \times p(\boldsymbol{W} \mid \boldsymbol{X}, \boldsymbol{Z}), \\
&= \prod_{k'} p(\boldsymbol{x}_{k'}) \times \prod_i \prod_k \prod_l \mathcal{M}_p(\boldsymbol{w}_{il},\, 1,\, \boldsymbol{U}\boldsymbol{x}_k)^{z_{ik}}, \\
&= \prod_k p(\boldsymbol{x}_k) \prod_i \prod_j \prod_l (\boldsymbol{u}_j^\top \boldsymbol{x}_k)^{z_{ik} w_{ilj}}, \\
&= \prod_k p(\boldsymbol{x}_k) \prod_j \prod_i (\boldsymbol{u}_j^\top \boldsymbol{x}_k)^{\sum_{l=1}^{c_i} z_{ik} w_{ilj}}, \\
&= \prod_k p(\boldsymbol{x}_k) \prod_j (\boldsymbol{u}_j^\top \boldsymbol{x}_k)^{\sum_{i=1}^{n} \sum_{l=1}^{c_i} z_{ik} w_{ilj}}, \\
&= \prod_k p(\boldsymbol{x}_k) \prod_j (\boldsymbol{u}_j^\top \boldsymbol{x}_k)^{\sum_i z_{ik} y_{ij}},
\end{aligned}
$$

since $y_{ij} = \sum_l w_{ilj}$. Then, put

$$
\begin{aligned}
\tilde{\boldsymbol{W}}_k(\boldsymbol{Z}) &= \{z_{ik}\boldsymbol{w}_{il}, i = 1, \ldots, n, l = 1 \ldots, c_i\}\ , \\
\tilde{\boldsymbol{W}}_{kj}(\boldsymbol{Z}) &= \sum_i^n z_{ik} \sum_{l=1}^{c_i} z_{ik} w_{ilj},
\end{aligned}
$$

and this completes the proof of Proposition 3.1. □

A.2 Derivation of the lower bound

Lower bound and Proposition 3.2. The bound of Equation (3.9) follows from standard derivation of the evidence lower bound in variational inference. Since the log is concave, by Jensen inequality:

$$
\begin{aligned}
\log p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{\pi}, \boldsymbol{U}) &= \log \sum_{\boldsymbol{T}} \int_{\boldsymbol{X}} p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{T} \mid \boldsymbol{\pi}, \boldsymbol{U})\, \mathrm{d}\boldsymbol{X}, \\
&= \log \sum_{\boldsymbol{T}} \int_{\boldsymbol{X}} \frac{p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{T} \mid \boldsymbol{\pi}, \boldsymbol{U})}{q(\boldsymbol{T}, \boldsymbol{X})} q(\boldsymbol{T}, \boldsymbol{X})\, \mathrm{d}\boldsymbol{X}, \\
&= \log \left(\mathbb{E}_q \left[\frac{p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{\pi}, \boldsymbol{U})}{q(\boldsymbol{T}, \boldsymbol{X})} \right] \right) \\
&\geq \mathbb{E}_q \left[\log \frac{p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{\pi}, \boldsymbol{U})}{q(\boldsymbol{T}, \boldsymbol{X})} \right], \\
&:= \mathcal{J}(q(\cdot);\, \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z}).
\end{aligned}
$$

Moreover, the difference between the classification log-likelihood and its bound is exactly the KL divergence between the approximate posterior $q(\cdot)$ and the true one:

$$\log p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{\pi}, \boldsymbol{U}) - \mathcal{J}(q(\cdot); \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z}) = -\mathbb{E}_q \left[\log \frac{p(\boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{U})}{q(\boldsymbol{T}, \boldsymbol{X})} \right].$$

Furthermore, the complete expression is given in Proposition 3.2 as:

$$\begin{aligned}
\mathcal{J}(q(\cdot); \boldsymbol{\pi}, \boldsymbol{U}, \boldsymbol{Z}) &= \underbrace{\mathbb{E}_q \left[\log p(\boldsymbol{W}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{Z}, \boldsymbol{U}) \right] - \mathbb{E}_q \left[\log q(\boldsymbol{T}, \boldsymbol{X}) \right]}_{\mathcal{J}_{\mathrm{LDA}}} + \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}), \\
&= \textstyle\sum_{k=1}^{K} \mathcal{J}_{\mathrm{LDA}}^{(k)}(q; \boldsymbol{U}, \tilde{\boldsymbol{W}}_k(\boldsymbol{Z})) + \sum_{k=1}^{K} \sum_{i=1}^{n} z_{ik} \log(\pi_k),
\end{aligned}$$

where

$$\begin{aligned}
\mathcal{J}_{\mathrm{LDA}}^{(k)}(q; \boldsymbol{U}, \tilde{\boldsymbol{W}}_k(\boldsymbol{Z})) = {} & \log \Gamma(\textstyle\sum_{h=1}^{d} \delta_h) - \sum_{h=1}^{d} \log \Gamma(\delta_h) \\
& + \sum_{h=1}^{d} (\delta_h - 1)(\psi(\gamma_{kh}) - \psi(\textstyle\sum_{h'=1}^{d} \gamma_{kh'})) \\
& + \sum_{i=1}^{N} z_{ik} \sum_{h=1}^{d} \sum_{l=1}^{L_i} \phi_{ilh} \left[\psi(\gamma_{kh}) - \psi(\textstyle\sum_{h'=1}^{d} \gamma_{kh'}) + \sum_{j=1}^{p} w_{ilj} \log(u_{jh}) \right] \\
& - \log \Gamma(\textstyle\sum_{h=1}^{d} \gamma_{kh}) - \sum_{h=1}^{d} \log \Gamma(\gamma_{kh}) \qquad (14) \\
& - \sum_{h=1}^{d} (\gamma_{kh} - 1)(\psi(\gamma_{kh}) - \psi(\textstyle\sum_{h'=1}^{d} \gamma_{kh'})) \\
& - \sum_{i=1}^{n} z_{ik} \sum_{l=1}^{L_i} \phi_{ilh} \log(\phi_{ilh}).
\end{aligned}$$

□

A.3 Optimization of $q(\boldsymbol{Z})$

Proof of Proposition 3.3 on page 57. Using the CAVI updates of Proposition 2.3, we get that the optimal $q(\boldsymbol{t}_{il})$ verifies:

$$\log q(\boldsymbol{t}_{il}) = \mathbb{E}_{\boldsymbol{T}_{-(i,l)}, \boldsymbol{X}} \left[\log p(\boldsymbol{W}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{Z}) \right] + \text{const},$$

where the expectation is taken with respect to all $\boldsymbol{X}$ and $\boldsymbol{T}$ except $\boldsymbol{t}_{il}$, assuming $(\boldsymbol{T}, \boldsymbol{X}) \sim q$. Developing the latter leads to:

$$\log q(\boldsymbol{t}_{il}) = \sum_{h=1}^{d} t_{ilh} \left[\sum_{j=1}^{p} w_{ilj} \log(u_{jh}) + \psi(\gamma_{kh}) - \psi(\textstyle\sum_{h'=1}^{d} \gamma_{kh'}) \right] + \text{const}. \qquad (15)$$

Equation (15) characterizes the log density of a multinomial:

$$q(\boldsymbol{t}_{il}) = \mathcal{M}_d(\boldsymbol{t}_{il} \mid 1, \boldsymbol{\phi}_{il} = (\phi_{il1}, \ldots, \phi_{ild})),$$

where the quantity inside brackets represents the logarithm of the parameter, modulo the normalizing constant. Hence,

$$\forall h, \quad \phi_{ilh} \propto \left(\prod_{j=1}^{p} u_{jh}^{w_{ilj}} \right) \prod_{k=1}^{K} \exp \left\{ \psi(\gamma_{kh}) - \psi \left(\sum_{h'=1}^{d} \gamma_{kh'} \right) \right\}^{z_{ik}}.$$

□

A.4 Optimization of $q(\boldsymbol{X})$

Proof of Proposition 3.4 on page 57. With the same reasoning, the optimal form of $q(\boldsymbol{X})$ is:

$$\begin{aligned} \log q(\boldsymbol{X}) &= \mathbb{E}_{\boldsymbol{T}} \left[p(\boldsymbol{W}, \boldsymbol{T}, \boldsymbol{X} \mid \boldsymbol{Z}) \right] + \text{const}, \\ &= \sum_{k=1}^{K} \left[\sum_{h=1}^{d} (\delta_h - 1) \log(\boldsymbol{x}_{kh}) + \sum_{i=1}^{n} z_{ik} \sum_{l=1}^{L_i} \sum_{h=1}^{d} \phi_{ilh} \log(\boldsymbol{x}_{kh}) \right] + \text{const}, \\ &= \sum_{k=1}^{K} \sum_{h=1}^{d} \left[\delta_h + \sum_{i=1}^{n} z_{ik} \sum_{l=1}^{L_i} \phi_{ilh} - 1 \right] \log(\boldsymbol{x}_{kh}) + \text{const}. \end{aligned} \tag{16}$$

Once again, a specific functional form appears as the log of a product of K independent Dirichlet densities. Then,

$$q(\boldsymbol{X}) = \prod_{k=1}^{K} \mathcal{D}_d \left(\boldsymbol{x}_k \mid \boldsymbol{\gamma}_k = (\gamma_{k1}, \ldots, \gamma_{kd}) \right),$$

with the Dirichlet parameters inside the brackets of Equation (16):

$$\forall (k, h), \quad \gamma_{kh} = \delta_h + \sum_{i=1}^{n} z_{ik} \sum_{l=1}^{L_i} \phi_{ilh}.$$

□

A.5 Optimization of $\boldsymbol{U}$

Proof of Proposition 3.5 on page 57 (I). This a constrained maximization problem with d constraints $\sum_{j=1}^{p} u_{jh} = 1$. Isolating terms of Equation (14) depending on $\boldsymbol{U}$, and denoting constraints multipliers as $(\lambda_h)_h$, the Lagrangian can be written:

$$\begin{aligned} f(\boldsymbol{U}, \lambda) &= \sum_{k=1}^{K} \sum_{i=1}^{n} z_{ik} \sum_{l=1}^{L_i} \sum_{j=1}^{p} \sum_{h=1}^{d} \phi_{ilh} w_{ilj} \log(u_{jh}) + \sum_{h=1}^{d} \lambda_h (\textstyle\sum_j u_{jh} - 1), \\ &= \sum_{i=1}^{n} \sum_{l=1}^{L_i} \sum_{j=1}^{p} \phi_{ilh} w_{ilj} \log(u_{jh}) + \sum_{h=1}^{d} \lambda_h (\textstyle\sum_j u_{jh} - 1). \end{aligned}$$

Setting its derivative to 0 leaves:

$$u_{jh} \propto \sum_{i=1}^{n} \sum_{l=1}^{L_i} \phi_{ilh}\, w_{ilj}.$$

□

A.6 Optimization of $\boldsymbol{\pi}$

Proof of Proposition 3.5 on page 57 (II). The bound depends on $\boldsymbol{\pi}$ only through its clustering term:

$$\log p(\boldsymbol{Z} \mid \boldsymbol{\pi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \log(\pi_k).$$

Once again, this is a constrained optimization problem, and, introducing the Lagrange multiplier λ associated to the constraint $\sum_{k=1}^{K} \pi_k = 1$, we get:

$$\sum_{k=1}^{K} \sum_{i=1}^{n} z_{ik} \log(\pi_k) + \lambda(\textstyle\sum_{k=1}^{K} \pi_k - 1).$$

Setting the derivative with respect to π_k to 0, we get:

$$\pi_k = \frac{\sum_{i=1}^{n} z_{ik}}{N}.$$

□

A.7 Model selection

Proof of Proposition 3.6 on page 60. Assuming that the parameters $(\boldsymbol{\pi}, \boldsymbol{U})$ follows a prior distribution that factorizes as follow:

$$p(\boldsymbol{\pi}, \boldsymbol{U} \mid K, d) = p(\boldsymbol{\pi} \mid K, \alpha)\, p(\boldsymbol{U} \mid d), \tag{17}$$

where

$$p(\boldsymbol{\pi} \mid K, \alpha) = \mathcal{D}_d(\boldsymbol{\pi} \mid \alpha \mathbf{1}_K). \tag{18}$$

Then, the classification log-likelihood is written:

$$\begin{aligned}
\log p(\boldsymbol{W}, \boldsymbol{Z} \mid K, d) &= \log \int_{\boldsymbol{\pi}} \int_{\boldsymbol{U}} p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{\pi} \mid K, d)\ \mathrm{d}\boldsymbol{\pi}\ \mathrm{d}\boldsymbol{U} \\
&= \log \int_{\boldsymbol{\pi}} \int_{\boldsymbol{U}} p(\boldsymbol{W}, \boldsymbol{Z} \mid \boldsymbol{U}, \boldsymbol{\pi}, K, d) p(\boldsymbol{\pi} \mid K, \alpha)\, p(\boldsymbol{U} \mid d)\ \mathrm{d}\boldsymbol{\pi}\ \mathrm{d}\boldsymbol{U} \\
&= \log \int_{\boldsymbol{\pi}} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) p(\boldsymbol{\pi} \mid K, \alpha)\, \mathrm{d}\boldsymbol{\pi} \int_{\boldsymbol{U}} p(\boldsymbol{W} \mid \boldsymbol{Z}, \boldsymbol{U}, K, d) p(\boldsymbol{U} \mid d)\, \mathrm{d}\boldsymbol{U} \\
&= \log \int_{\boldsymbol{\pi}} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) p(\boldsymbol{\pi} \mid K, \alpha)\, \mathrm{d}\boldsymbol{\pi} \\
&\qquad + \log \int_{\boldsymbol{U}} p(\boldsymbol{W} \mid \boldsymbol{Z}, \boldsymbol{U}, K, d) p(\boldsymbol{U} \mid d)\, \mathrm{d}\boldsymbol{U}. \qquad (19)
\end{aligned}$$

The first term in Equation (19) is exact by Dirichlet-Multinomial conjugacy. Setting $\alpha = \frac{1}{2}$ plus a Stirling approximation on the Gamma function as in Biernacki et al. (2000) leads to:

$$\log \int_{\boldsymbol{\pi}} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) p(\boldsymbol{\pi} \mid K, \alpha) \, \mathrm{d}\boldsymbol{\pi} \approx \max_{\boldsymbol{\pi}} \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}, Q) - \frac{K-1}{2} \log(n). \tag{20}$$

As for the second term, a BIC-like approximation as in Bouveyron et al. (2018) gives:

$$\log \int_{\boldsymbol{U}} p(\boldsymbol{W} \mid \boldsymbol{Z}, \boldsymbol{U}, K, d) p(\boldsymbol{U} \mid d) \, \mathrm{d}\boldsymbol{U} \approx \max_{\boldsymbol{U}} \log p(\boldsymbol{W} \mid \boldsymbol{Z}, \boldsymbol{U}, K, d) - \frac{d(p-1)}{2} \log(K).$$

In practice, $\log p(\boldsymbol{W} \mid \boldsymbol{Z}, \boldsymbol{U}, K, d)$ is still intractable, hence we replace it by its variational approximation after convergence of the VEM, $\mathcal{J}^{\star}_{\mathrm{LDA}}$, which is the sum of the meta-observations individual LDA-bounds detailed in Equation (14) (different from $\mathcal{J}$). In the end, it gives the following criterion:

$$\begin{aligned} \mathrm{ICL}(K, d, \boldsymbol{Z}, \boldsymbol{W}) \quad &= \quad \mathcal{J}^{\star}_{\mathrm{LDA}}(q; \boldsymbol{U}, \boldsymbol{Z}) - \frac{d(p-1)}{2} \log(K) \\ &\quad + \max_{\boldsymbol{\pi}} \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}, K) - \frac{K-1}{2} \log(n). \end{aligned} \tag{21}$$

Note that:

$$\max_{\boldsymbol{U}} \log p(\boldsymbol{W} \mid \boldsymbol{Z}, \boldsymbol{U}, K, d) + \max_{\boldsymbol{\pi}} \log p(\boldsymbol{Z} \mid \boldsymbol{\pi}, K) \approx \mathcal{J}^{\star},$$

i.e. the bound after Algorithm 2 converges.

□

B Appendix for Chapter 4

B.1 Optimization of $q(\boldsymbol{Z})$

Proof of Proposition 4.1 on page 78. From Proposition 2.3 we know that the optimal form of an individual distribution is:

$$q^\star(\boldsymbol{z}_i) \propto \exp\left\{\mathbb{E}_{\boldsymbol{z}_{-i},\boldsymbol{\mu}}\left[\log p(\boldsymbol{Y}, \boldsymbol{z}_i, \boldsymbol{z}_{-i}, \boldsymbol{\mu} \mid \boldsymbol{\vartheta})\right]\right\} \tag{22}$$

Taking the log of this expression and leaving out everything that does not depend on $\boldsymbol{z}_i$ leads to the following functional form:

$$\log q^\star(\boldsymbol{z}_i) = \mathbb{E}_{\boldsymbol{\mu}_k}\left[\sum_k z_{ik}\left[\log(\pi_k) + \log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k)\right]\right] + \mathrm{C}, \tag{23}$$

$$= \sum_k z_{ik}\left[\log(\pi_k) + \mathbb{E}_{\boldsymbol{\mu}_k}\left[\log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k)\right]\right] + \mathrm{C}. \tag{24}$$

Here we recognize the functional form of a multinomial distribution:

$$q^\star(\boldsymbol{z}_i) = \mathcal{M}_K(\boldsymbol{z}_i \mid 1, \boldsymbol{\tau}_i), \tag{25}$$

with:

$$\tau_{ik} \propto \pi_k \exp\{\mathbb{E}_{\boldsymbol{\mu}_k}\left[\log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\mu_k, \boldsymbol{S}_k)\right]\}. \tag{26}$$

□

B.2 Optimization of $q(\boldsymbol{\mu})$

Proof of Proposition 4.2 on page 78. Let $k \in \{1, \ldots, K\}$, still from Proposition 2.3 we know that the optimal $q(\boldsymbol{\mu}_k)$ verifies:

$$q^\star(\boldsymbol{\mu}_k) \propto \exp\left\{\mathbb{E}_{\boldsymbol{Z},\boldsymbol{\mu}_{-k}}\left[\log p(\boldsymbol{Y}, \boldsymbol{\mu}_k, \boldsymbol{Z}, \boldsymbol{\mu}_{-k}) \mid \boldsymbol{\vartheta}\right]\right\} \tag{27}$$

Taking the logarithm of this expression and leaving out everything that does not depend on $\boldsymbol{\mu}_k$ leads to the following functional form:

$$\log q^\star(\boldsymbol{\mu}_k) = \log p(\boldsymbol{\mu}_k) + \sum_{i=1}^{n} \mathbb{E}_{\boldsymbol{z}_i, \boldsymbol{\mu}_{-k}} \left[\log p(\boldsymbol{y}_i \mid \boldsymbol{z}_i, \boldsymbol{\mu}_{-k})\right] + \mathrm{C}_1, \tag{28}$$

$$= \log \mathcal{N}_d(\boldsymbol{\mu}_k \mid \boldsymbol{\nu}, \lambda \boldsymbol{I}_d) + \sum_{i=1}^{n} \tau_{ik} \log \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k) + \mathrm{C}_2 \tag{29}$$

$$= -\frac{1}{2}\left[\lambda^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\nu})^\top(\boldsymbol{\mu}_k - \boldsymbol{\nu}) + \sum_{i=1}^{n} \tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top \boldsymbol{S}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)\right] + \mathrm{C}_3, \tag{30}$$

$$= -\frac{1}{2}\left[\lambda^{-1}\boldsymbol{\mu}_k^\top\boldsymbol{\mu}_k + \sum_{i=1}^{n} \tau_{ik}\boldsymbol{\mu}_k^\top \boldsymbol{U}^\top \boldsymbol{S}_k^{-1}\boldsymbol{U}\boldsymbol{\mu}_k \right.$$
$$\left. -2\left(\sum_{i=1}^{n} \tau_{ik}\boldsymbol{y}_i^\top \boldsymbol{S}_k^{-1}\boldsymbol{U}\boldsymbol{\mu}_k + \lambda^{-1}\boldsymbol{\nu}^\top\boldsymbol{\mu}_k\right)\right] + \mathrm{C}_4, \tag{31}$$

$$= -\frac{1}{2}\left[\boldsymbol{\mu}_k^\top(\lambda^{-1}\boldsymbol{I}_d + \tilde{n}_k \boldsymbol{U}^\top \boldsymbol{S}_k^{-1}\boldsymbol{U})\boldsymbol{\mu}_k - 2\left(\sum_{i=1}^{n} \tau_{ik}\boldsymbol{y}_i^\top \boldsymbol{S}_k^{-1}\boldsymbol{U} + \lambda^{-1}\boldsymbol{\nu}^\top\right)\boldsymbol{\mu}_k\right] + \mathrm{C}_4. \tag{32}$$

Putting:

$$\tilde{\mathbf{M}}_k = \left(\lambda^{-1}\boldsymbol{I}_d + \tilde{n}_k \boldsymbol{U}^\top \boldsymbol{S}_k^{-1}\boldsymbol{U}\right)^{-1}, \tag{33}$$

$$\tilde{\boldsymbol{\mu}}_k = \tilde{\mathbf{M}}_k\left(\sum_{i=1}^{n} \tau_{ik}\boldsymbol{U}^\top \boldsymbol{S}_k^{-1}\boldsymbol{y}_i + \lambda^{-1}\boldsymbol{\nu}\right) \tag{34}$$

we can then write:

$$\log q^\star(\boldsymbol{\mu}_k) = -\frac{1}{2}\left[(\boldsymbol{\mu}_k - \tilde{\boldsymbol{\mu}}_k)^\top \tilde{\mathbf{M}}_k^{-1}(\boldsymbol{\mu}_k - \tilde{\boldsymbol{\mu}}_k)\right] + \text{const}\ . \tag{35}$$

We recognize the logarithm of a Gaussian density with mean $\tilde{\boldsymbol{\mu}}_k$ and covariance $\tilde{\mathbf{M}}_k$. Moreover, the expressions of $\tilde{\boldsymbol{\mu}}_k$ and $\tilde{\mathbf{M}}_k$ can be simplified, since $\boldsymbol{S}_k = \boldsymbol{D}\boldsymbol{\Delta}_k\boldsymbol{D}^\top$ with $\boldsymbol{D} = [\boldsymbol{U}, \boldsymbol{V}]$ and $\boldsymbol{V}$ is the orthogonal complement of $\boldsymbol{U}$. Thus $\boldsymbol{S}_k^{-1} = \boldsymbol{D}\boldsymbol{\Delta}_k^{-1}\boldsymbol{D}^\top$ and:

$$\boldsymbol{U}^\top \boldsymbol{S}_k^{-1} = \boldsymbol{U}^\top \boldsymbol{D}\boldsymbol{\Delta}_k^{-1}\boldsymbol{D}^\top = \boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top, \tag{36}$$

$$\boldsymbol{U}^\top \boldsymbol{S}_k^{-1}\boldsymbol{U} = \boldsymbol{\Sigma}_k^{-1}. \tag{37}$$

Thus, a use of Woodbury's identity gives

$$\begin{aligned}\tilde{\mathbf{M}}_k\lambda^{-1}\boldsymbol{\nu} &= \left(\boldsymbol{I}_d + \tilde{n}_k\boldsymbol{I}_d\lambda\boldsymbol{I}_d\boldsymbol{\Sigma}_k^{-1}\right)^{-1}\boldsymbol{\nu},\\ &= \left[\boldsymbol{I}_d - \boldsymbol{I}_d\tilde{n}_k\left(\lambda^{-1}\boldsymbol{I}_d + \tilde{n}_k\boldsymbol{\Sigma}_k^{-1}\right)^{-1}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{I}_d\right]\boldsymbol{\nu},\\ &= \boldsymbol{\nu} - \tilde{\mathbf{M}}_k\boldsymbol{\Sigma}_k^{-1}\tilde{n}_k\boldsymbol{\nu},\end{aligned}$$

and we finally get:

$$\begin{aligned}\tilde{\boldsymbol{\mu}}_k &= \tilde{\mathbf{M}}_k\left(\sum_{i=1}^n \tau_{ik}\boldsymbol{U}^\top\boldsymbol{S}_k^{-1}\boldsymbol{y}_i + \lambda^{-1}\boldsymbol{\nu}\right),\\ &= \tilde{\mathbf{M}}_k\boldsymbol{\Sigma}_k^{-1}\left(\sum_{i=1}^n \tau_{ik}\boldsymbol{U}^\top\boldsymbol{y}_i\right) + \tilde{\mathbf{M}}_k\lambda^{-1}\boldsymbol{\nu},\\ &= \boldsymbol{\nu} + \tilde{\mathbf{M}}_k\boldsymbol{\Sigma}_k^{-1}\left(\sum_{i=1}^n \tau_{ik}\boldsymbol{U}^\top\boldsymbol{y}_i - \tilde{n}_k\boldsymbol{\nu}\right)\end{aligned} \tag{38}$$

□

B.3 Variational lower bound

We recall

$$\mathcal{J}(q,\boldsymbol{\vartheta}) = \mathbb{E}_q\left[\log p(\boldsymbol{Y},\boldsymbol{Z},\boldsymbol{\mu}\mid\boldsymbol{\vartheta})\right] + \mathrm{H}(q),$$

Now that $q^\star$ have been derived in Propositions 4.1 and 4.2, we can compute the variational lower bound explicitly:

$$\begin{aligned}\mathcal{J}(q,\boldsymbol{\vartheta}) = &-\frac{1}{2}\sum_{i=1}^n\sum_{k=1}^K \tau_{ik}\left\{p\log(2\pi) + \log|\boldsymbol{S}_k| + \mathbb{E}_{\boldsymbol{\mu}_k}\left[(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\boldsymbol{S}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)\right]\right\}\\ &+\sum_{k=1}^K \tilde{n}_k\log(\pi_k)\\ &-\frac{1}{2}\sum_{k=1}^K d\log(2\pi) + d\log(\lambda) + \frac{1}{\lambda}\mathbb{E}_{\boldsymbol{\mu}_k}\left[(\boldsymbol{\mu}_k - \boldsymbol{\nu})^\top(\boldsymbol{\mu}_k - \boldsymbol{\nu})\right]\\ &-\sum_i\sum_k \tau_{ik}\log(\tau_{ik})\\ &+\frac{Kd}{2}(\log(2\pi)+1) + \frac{1}{2}\sum_k\log|\tilde{\mathbf{M}}_k|,\end{aligned} \tag{39}$$

with:

$$\begin{aligned}\mathbb{E}_{q^\star}[\boldsymbol{\mu}_k] &= \tilde{\boldsymbol{\mu}}_k,\\ \mathbb{E}_{q^\star}[\boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top] &= \tilde{\boldsymbol{\mu}}_k\tilde{\boldsymbol{\mu}}_k^\top + \tilde{\mathbf{M}}_k,\\ \log|\boldsymbol{S}_k| &= \log|\boldsymbol{\Sigma}_k| + (p-d)\log(\beta_k).\end{aligned} \tag{40}$$

Proof of Proposition 4.3 on page 79. Recall the form of the bound given in Proposition 4.3:

$$\begin{aligned}\mathcal{J}(\boldsymbol{\vartheta}) = \text{const} - \frac{1}{2}\sum_{k=1}^{K}\tilde{n}_k\Bigg\{ &- 2\log(\pi_k) + \log|\boldsymbol{\Sigma}_k| + (p-d)\log(\beta_k) \\ &+ \operatorname{Tr}\left[\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top\hat{\boldsymbol{C}}_k\boldsymbol{U}\right] + \frac{1}{\beta_k}\left(\operatorname{Tr}\left[\hat{\boldsymbol{C}}_k\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top\hat{\boldsymbol{C}}_k\boldsymbol{U}\right]\right)\Bigg\}.\end{aligned} \tag{41}$$

Only the first two lines of Equation (39) depend on $\boldsymbol{\vartheta}$, focusing on the first line:

$$\begin{aligned}&\mathbb{E}_q\left[\log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\mu}; \boldsymbol{\vartheta})\right] \\ =&\sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}\left\{p\log(2\pi) + \log|\boldsymbol{S}_k| + \mathbb{E}_{\boldsymbol{\mu}_k}\left[(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\boldsymbol{S}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)\right]\right\} \\ =&\gamma + \sum_{k=1}^{K}\tilde{n}_k\left\{\log|\boldsymbol{S}_k| + \frac{1}{\tilde{n}_k}\mathbb{E}_{\boldsymbol{\mu}_k}\left[\sum_i \tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\boldsymbol{S}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)\right]\right\},\end{aligned}$$

where $\gamma = np\log(2\pi)$. The terms inside the expectations may be rearranged using the usual *trace trick* to make the empirical cluster covariance matrices appear:

$$\begin{aligned}&\frac{1}{\tilde{n}_k}\sum_i \tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\boldsymbol{S}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k), \\ =&\frac{1}{\tilde{n}_k}\sum_i \tau_{ik}\operatorname{Tr}\left[(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\boldsymbol{S}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)\right], \\ =&\operatorname{Tr}\left[\boldsymbol{S}_k^{-1}\frac{1}{\tilde{n}_k}\sum_i \tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\right], \\ =&\operatorname{Tr}\left[\boldsymbol{S}_k^{-1}\boldsymbol{C}_k\right],\end{aligned} \tag{42}$$

where

$$\boldsymbol{C}_k = \frac{1}{\tilde{n}_k}\sum_i \tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top,$$

is the empirical covariance matrices of cluster k. This trace may be further decomposed using the following lemma which relies on the particular form of $\boldsymbol{S}_k^{-1}$ in the BDLM model.

Lemma .1. *For any square matrix $\boldsymbol{A} \in \mathcal{M}_{p\times p}(\mathbb{R})$, the following identity holds:*

$$\operatorname{Tr}\left[\boldsymbol{S}_k^{-1}\boldsymbol{A}\right] = \operatorname{Tr}\left[\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top\boldsymbol{A}\boldsymbol{U}\right] + \frac{1}{\beta}\left(\operatorname{Tr}\left[\boldsymbol{A}\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top\boldsymbol{A}\boldsymbol{U}\right]\right) \tag{43}$$

Proof. We can split $\boldsymbol{S}_k^{-1}$ in two parts depending on the discriminative subspace and its orthogonal complement: take $\boldsymbol{D} = \tilde{\boldsymbol{D}} + \bar{\boldsymbol{D}}$, with $\tilde{\boldsymbol{D}} = [\boldsymbol{U}, \boldsymbol{0}_{p\times(p-d)}]$ and $\bar{\boldsymbol{D}} = [\boldsymbol{0}_{p\times d}, \boldsymbol{V}]$.

Then, $\boldsymbol{S}_k^{-1} = \tilde{\boldsymbol{D}}\boldsymbol{\Delta}_k^{-1}\tilde{\boldsymbol{D}}^\top + \bar{\boldsymbol{D}}\boldsymbol{\Delta}_k^{-1}\bar{\boldsymbol{D}}^\top$, and:

$$\begin{aligned}\operatorname{Tr}\left[\boldsymbol{S}_k^{-1}\boldsymbol{A}\right] &= \operatorname{Tr}\left[\boldsymbol{\Delta}_k^{-1}\tilde{\boldsymbol{D}}^\top\boldsymbol{A}\tilde{\boldsymbol{D}}\right] + \operatorname{Tr}\left[\boldsymbol{\Delta}_k^{-1}\bar{\boldsymbol{D}}^\top\boldsymbol{A}\bar{\boldsymbol{D}}\right], \\ &= \operatorname{Tr}\left[\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top\boldsymbol{A}\boldsymbol{U}\right] + \frac{1}{\beta}\operatorname{Tr}\left[\boldsymbol{V}^\top\boldsymbol{A}\boldsymbol{V}\right]\end{aligned}$$

Moreover, $\boldsymbol{D}\boldsymbol{D}^\top = \boldsymbol{D}^\top\boldsymbol{D} = I_p$ and $\boldsymbol{D} = \tilde{\boldsymbol{D}} + \bar{\boldsymbol{D}}$, hence:

$$\operatorname{Tr}\left[\boldsymbol{A}\right] = \operatorname{Tr}\left[\boldsymbol{D}^\top\boldsymbol{A}\boldsymbol{D}\right] = \operatorname{Tr}\left[\boldsymbol{U}^\top\boldsymbol{A}\boldsymbol{U}\right] + \operatorname{Tr}\left[\boldsymbol{V}^\top\boldsymbol{A}\boldsymbol{V}\right]$$

This concludes Lemma .1's proof. □

Applying Lemma .1 to Equation (42) with $\boldsymbol{A} = \mathbb{E}_{\boldsymbol{\mu}_k}(\boldsymbol{C}_k)$ leaves:

$$\begin{aligned}\mathbb{E}_q\left[\log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\mu}; \boldsymbol{\vartheta})\right] =& \gamma + \sum_{k=1}^{K}\tilde{n}_k\Bigg\{\log|\boldsymbol{\Sigma}_k| + (p-d)\log(\beta_k) + \operatorname{Tr}\left[\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top\mathbb{E}(\boldsymbol{C}_k)\boldsymbol{U}\right] \\ &+ \frac{1}{\beta}\left(\operatorname{Tr}\left[\mathbb{E}(\boldsymbol{C}_k)\right] - \operatorname{Tr}\left[\boldsymbol{U}^\top\mathbb{E}(\boldsymbol{C}_k)\boldsymbol{U}\right]\right)\Bigg\}.\end{aligned} \tag{44}$$

The matrix $\mathbb{E}_{\boldsymbol{\mu}_k}(\boldsymbol{C}_k)$ is denoted as $\hat{\boldsymbol{C}}_k$ and, using Equation (40), one gets:

$$\begin{aligned}\hat{\boldsymbol{C}}_k &= \mathbb{E}_{\boldsymbol{\mu}_k}(\boldsymbol{C}_k), \\ &= \frac{1}{\tilde{n}_k}\sum_i \tau_{ik}\,\mathbb{E}\left[(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)(\boldsymbol{y}_i - \boldsymbol{U}\boldsymbol{\mu}_k)^\top\right], \\ &= \frac{1}{\tilde{n}_k}\sum_i \tau_{ik}\left(\boldsymbol{y}_i\boldsymbol{y}_i^\top - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k\boldsymbol{y}_i^\top - \boldsymbol{y}_i(\boldsymbol{U}\tilde{\boldsymbol{\mu}}_k)^\top + \boldsymbol{U}\,\mathbb{E}\left[\boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top\right]\boldsymbol{U}^\top\right), \\ &= \frac{1}{\tilde{n}_k}\sum_{i=1}^{n}\tau_{ik}(\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k)(\boldsymbol{y}_i - \boldsymbol{U}\tilde{\boldsymbol{\mu}}_k)^\top + \boldsymbol{U}\tilde{\mathbf{M}}_k\boldsymbol{U}^\top.\end{aligned} \tag{45}$$

Finally, the second line of Equation (39) is simply

$$\mathbb{E}_q[\log p(\boldsymbol{Z} \mid \boldsymbol{\pi})] = -\frac{1}{2}\sum_{k=1}^{K} -2\tilde{n}_k\log(\pi_k). \tag{46}$$

This concludes the proof. □

B.4 M-step

Proof of Proposition 4.4 on page 80. Although there are 12 different sub-models, a lot of the proofs are the same.

Optimization of $\boldsymbol{\beta}$. Let us start with the two possible cases for $\boldsymbol{\beta}$, which are common regardless of the constraint on the latent covariance matrices:

- Model $\text{BDLM}_{[(\cdot)\beta_k]}$: In this case, the variational bound as a function of β_k is:

$$\mathcal{J}(\beta_k) = -\frac{1}{2}\tilde{n}_k \left[(p-d)\log(\beta_k) + \frac{1}{\beta_k}\left(\text{Tr}\left[\hat{\boldsymbol{C}}_k\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right]\right)\right].$$

  Thus, its only stationary point is:

$$\nabla_{\beta_k} \mathcal{J}(\hat{\beta}_k) = 0 \iff \hat{\beta}_k = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}_k\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right]}{p-d}.$$

- Model $\text{BDLM}_{[(\cdot)\beta]}$: In this case, the variational bound as a function of β is:

$$\begin{aligned}\mathcal{J}(\beta) &= -\frac{1}{2}\sum_{k=1}^{K} \tilde{n}_k \left\{ (p-d)\log(\beta) + \frac{1}{\beta}\left(\text{Tr}\left[\hat{\boldsymbol{C}}_k\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right]\right)\right\}, \\ &= -\frac{1}{2} n \left\{ (p-d)\log(\beta) + \frac{1}{\beta}\left(\text{Tr}\left[\frac{1}{n}\sum_{k=1}^{K} \tilde{n}_k \hat{\boldsymbol{C}}_k\right] - \text{Tr}\left[\boldsymbol{U}^\top (\frac{1}{n}\sum_{k=1}^{K} \tilde{n}_k \hat{\boldsymbol{C}}_k)\boldsymbol{U}\right]\right)\right\}.\end{aligned}$$

  And, again, its only stationary point is:

$$\nabla_{\beta} \mathcal{J}(\hat{\beta}) = 0 \iff \hat{\beta} = \frac{\text{Tr}\left[\hat{\boldsymbol{C}}\right] - \text{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}} \boldsymbol{U}\right]}{p-d},$$

  with $\hat{\boldsymbol{C}} = \frac{1}{n}\sum_{k=1}^{K} \tilde{n}_k \hat{\boldsymbol{C}}_k$.

□

Optimization of $\boldsymbol{\Sigma}$. There are now 6 cases to treat, which are the full, diagonal and isotropic covariance matrices where each case can be with or without homoscedasticity. We will need the two following formulas concerning matrix derivation. For any invertible matrix, $\boldsymbol{A} \in \mathbb{R}^{p\times p}$ we have:

$$\nabla_{\boldsymbol{A}} \log|\boldsymbol{A}| = \boldsymbol{A}^{-1}, \tag{47}$$

$$\nabla_{\boldsymbol{A}} \text{Tr}\left[\boldsymbol{A}^{-1}\boldsymbol{B}\right] = -(\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1})^\top. \tag{48}$$

- Model $\text{BDLM}_{[\boldsymbol{\Sigma}_k(\cdot)]}$: We rewrite the bound of Equation (4.9) as a function of $\boldsymbol{\Sigma}_k$:

$$\mathcal{J}(\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K) = -\frac{1}{2}\sum_{k=1}^{K} \tilde{n}_k \left\{ \log|\boldsymbol{\Sigma}_k| + \text{Tr}\left[\boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right]\right\} + \text{const}. \tag{49}$$

  Thus, using Equations (47) and (48) with $\boldsymbol{A} = \boldsymbol{\Sigma}_k$ we get:

$$\nabla_{\boldsymbol{\Sigma}_k} \mathcal{J}(\boldsymbol{\Sigma}_k) = -\frac{\tilde{n}_k}{2}\left(\boldsymbol{\Sigma}_k^{-1} - \boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\boldsymbol{\Sigma}_k^{-1}\right).$$

  Then, a first order condition gives

$$\nabla_{\boldsymbol{\Sigma}_k} \mathcal{J}(\boldsymbol{\Sigma}_k) = 0 \iff \boldsymbol{\Sigma}_k^{-1} = \boldsymbol{\Sigma}_k^{-1}\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\boldsymbol{\Sigma}_k^{-1}, \tag{50}$$

and we obtain the M-step estimate $\hat{\boldsymbol{\Sigma}}_k$ by multiplying left and right by $\boldsymbol{\Sigma}_k$:

$$\hat{\boldsymbol{\Sigma}}_k = \boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}. \tag{51}$$

- Model $\text{BDLM}_{[\boldsymbol{\Sigma}(\cdot)]}$: In this case, the variational bound can be rewritten as:

$$\mathcal{J}(\boldsymbol{\Sigma}) = -\frac{n}{2}\left\{ \log|\boldsymbol{\Sigma}| + \operatorname{Tr}\left[\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^\top\left(\frac{1}{n}\sum_{k=1}^{K}\tilde{n}_k\hat{\boldsymbol{C}}_k\right)\boldsymbol{U}\right]\right\} + \text{const}\ . \tag{52}$$

And finding the root of the gradient leads to:

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{U}^\top \hat{\boldsymbol{C}} \boldsymbol{U}. \tag{53}$$

- Model $\text{BDLM}_{[\alpha_{kh}(\cdot)]}$: In this model, the bound writes as a function of $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K)$:

$$\mathcal{J}(\boldsymbol{\alpha}) = -\frac{1}{2}\sum_{k=1}^{K}\tilde{n}_k\left\{\sum_{h=1}^{d}\log(\alpha_{kh}) + \frac{\boldsymbol{u}_h^\top\hat{\boldsymbol{C}}_k\boldsymbol{u}_h}{\alpha_{kh}}\right\} + \text{const}\ . \tag{54}$$

Thus, the partial derivative with respect to α_{kh} is given as:

$$\nabla_{\alpha_{kh}}\mathcal{J}(\alpha_{kh}) = -\frac{\tilde{n}_k}{2}\left(\frac{1}{\alpha_{kh}} - \frac{\boldsymbol{u}_h^\top\hat{\boldsymbol{C}}_k\boldsymbol{u}_h}{\alpha_{kh}^2}\right),$$

and finding its root gives:

$$\hat{\alpha}_{kh} = \boldsymbol{u}_h^\top\hat{\boldsymbol{C}}_k\boldsymbol{u}_h. \tag{55}$$

- Model $\text{BDLM}_{[\alpha_h(\cdot)]}$: Put $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ and we have,

$$\mathcal{J}(\boldsymbol{\alpha}) = -\frac{n}{2}\left\{\sum_{h=1}^{d}\log(\alpha_h) + \frac{\boldsymbol{u}_h^\top(\frac{1}{n}\sum_{k=1}^{K}\tilde{n}_k\hat{\boldsymbol{C}}_k)\boldsymbol{u}_h}{\alpha_h}\right\} + \text{const}\ . \tag{56}$$

Its gradient with respect to α_h is computed in the same manner as above, and a first-order condition gives:

$$\hat{\alpha}_h = \boldsymbol{u}_h^\top\hat{\boldsymbol{C}}\boldsymbol{u}_h \tag{57}$$

- Model $\text{BDLM}_{[\alpha_k(\cdot)]}$: Introducing $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$, the bound is written as:

$$\mathcal{J}(\boldsymbol{\alpha}) = -\frac{1}{2}\sum_{k=1}^{K}\tilde{n}_k\left\{d\log(\alpha_k) + \frac{1}{\alpha_k}\operatorname{Tr}\left[\boldsymbol{U}^\top\hat{\boldsymbol{C}}_k\boldsymbol{U}\right]\right\} + \text{const}\ . \tag{58}$$

Again, its gradient with respect to α_k is easily computed as:

$$\nabla_{\alpha_k}\mathcal{J}(\alpha_k) = -\frac{\tilde{n}_k}{2}\left(\frac{d}{\alpha_k} - \frac{\operatorname{Tr}\left[\boldsymbol{U}^\top\hat{\boldsymbol{C}}_k\boldsymbol{U}\right]}{\alpha_k^2}\right).$$

Finding it 0 point leaves the following M-step update for $\hat{\alpha}_k$

$$\hat{\alpha}_k = \frac{1}{d} \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}}_k \boldsymbol{U}\right]. \tag{59}$$

- Model $\text{BDLM}_{[\alpha(\cdot)]}$: For this final model, α is a positive scalar and the bound is:

$$\mathcal{J}(\alpha) = -\frac{n}{2}\left\{ d\log(\alpha) + \frac{1}{\alpha} \operatorname{Tr}\left[\boldsymbol{U}^\top (\frac{1}{n}\sum_{k=1}^{K} \tilde{n}_k \hat{\boldsymbol{C}}_k)\boldsymbol{U}\right] \right\} + \text{const}\ . \tag{60}$$

$$\hat{\alpha} = \frac{1}{d} \operatorname{Tr}\left[\boldsymbol{U}^\top \hat{\boldsymbol{C}} \boldsymbol{U}\right]. \tag{61}$$

□

□

B.5 Hyper-parameter estimation

To prove Proposition 4.5, we maximize the variational bound with respect to $(\boldsymbol{\nu}, \lambda)$, from Equation (39) we get:

$$\begin{aligned}
\mathcal{J}(\boldsymbol{\nu}, \lambda) &= -\frac{1}{2}\sum_{k=1}^{K} d\log(2\pi) + d\log(\lambda) + \frac{1}{\lambda}\mathbb{E}_{\boldsymbol{\mu}_k}\left[\|\boldsymbol{\mu}_k - \boldsymbol{\nu}\|_2^2\right], \\
&= -\frac{1}{2}\sum_{k=1}^{K} d\log(2\pi) + d\log(\lambda) + \frac{1}{\lambda}\left[\|\tilde{\boldsymbol{\mu}}_k - \boldsymbol{\nu}\|_2^2 + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k\right]\right].
\end{aligned}$$

Optimization with respect to $\boldsymbol{\nu}$.

$$\nabla_{\boldsymbol{\nu}} \mathcal{J}(\boldsymbol{\nu}) = -\frac{1}{\lambda}\sum_{k=1}^{K}(\tilde{\boldsymbol{\mu}}_k - \boldsymbol{\nu}) \tag{62}$$

Hence,

$$\begin{aligned}
&\nabla_{\boldsymbol{\nu}} \mathcal{J}(\hat{\boldsymbol{\nu}}) = 0, \\
\iff &\hat{\boldsymbol{\nu}} = \frac{\sum_{k=1}^{K} \tilde{\boldsymbol{\mu}}_k}{K}
\end{aligned}$$

Since $\mathcal{J}$ is a concave function of $\boldsymbol{\nu}$ with a negative definite Hessian $\frac{-1}{2\lambda}\boldsymbol{I}_d$, this concludes the proof □

Optimization with respect to λ.

$$\nabla_{\lambda} \mathcal{J}(\lambda) = -\frac{1}{2}\sum_{k=1}^{K} \frac{d}{\lambda} - \frac{1}{\lambda^2}\left[\|\tilde{\boldsymbol{\mu}}_k - \boldsymbol{\nu}\|_2^2 + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k\right]\right].$$

Thus, the first-order condition gives:

$$\begin{aligned}\nabla_\lambda \mathcal{J}(\hat{\lambda}) &= 0,\\ \hat{\lambda} &= \frac{\sum_{k=1}^K \|\tilde{\boldsymbol{\mu}}_k - \boldsymbol{\nu}\|_2^2 + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k\right]}{dK}\end{aligned}$$

The second-order derivative gives us a condition for $\hat{\lambda}$ to be a maximum. Indeed, the following must hold for $\lambda = \hat{\lambda}$:

$$\nabla_\lambda^2 \mathcal{J}(\lambda) = \frac{\left(\sum_{k=1}^K \|\tilde{\boldsymbol{\mu}}_k - \boldsymbol{\nu}\|_2^2 + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k\right]\right)\lambda - d}{2\lambda^3} < 0.$$

Hence, for positive λ, the latter is negative if and only if $\lambda < \frac{\sum_{k=1}^K \|\tilde{\boldsymbol{\mu}}_k - \boldsymbol{\nu}\|_2^2 + \operatorname{Tr}\left[\tilde{\mathbf{M}}_k\right]}{d} = K\hat{\lambda}$. Obviously, $\hat{\lambda}$ verifies this condition for $K \geq 2$ and is thus a maximum of $\mathcal{J}$.

□

B.6 Model selection

For the sake of notations, we drop the dependencies in $\mathcal{M}$ and K here, since the discussion is independent of these quantities. The result stems from the fact that the conditional posterior $p(\boldsymbol{\mu} \mid \hat{\boldsymbol{Z}}, \boldsymbol{Y})$ is tractable in the BDLM model and it equals to $q(\boldsymbol{\mu}_k)$ if $\tau_{ik} = \hat{z}_{ik}$. Thus the variational bound in is tight and equals the classification likelihood.

Formally, we want to show that the variational bound of Equation (39) is equal to $\log p(\boldsymbol{Y}, \hat{\boldsymbol{Z}})$ when $\tau_{ik} = \hat{z}_{ik}$ in Propositions 4.1 and 4.2. When $\boldsymbol{\tau} \leftarrow \hat{\boldsymbol{Z}}$, we have:

$$q(\boldsymbol{\mu}, \boldsymbol{Z} \mid \hat{\boldsymbol{Z}}) = q\left(\boldsymbol{\mu} \mid \tilde{\boldsymbol{\mu}}(\hat{\boldsymbol{Z}}), \tilde{\mathbf{M}}(\hat{\boldsymbol{Z}})\right) \times \delta_{\hat{\boldsymbol{Z}}}(\boldsymbol{Z}), \tag{63}$$

with δ_x the Dirac mass at x, and $(\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{M}})$ computed with $\hat{\boldsymbol{Z}}$ instead of $\boldsymbol{\tau}$:

$$\begin{aligned}\hat{n}_k &= \sum_i \hat{z}_{ik},\\ \tilde{\mathbf{M}}_k(\hat{\boldsymbol{Z}}) &= \left(\lambda^{-1}\boldsymbol{I}_d + \hat{n}_k \boldsymbol{\Sigma}_k^{-1}\right)^{-1},\\ \tilde{\boldsymbol{\mu}}_k(\hat{\boldsymbol{Z}}) &= \boldsymbol{\nu} + \tilde{\mathbf{M}}_k \boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{U}^\top \sum_i \hat{z}_{ik}\boldsymbol{y}_i - \hat{n}_k \boldsymbol{\nu}\right).\end{aligned}$$

It happens that the conditional posterior distribution of $\boldsymbol{\mu}$: $p(\boldsymbol{\mu} \mid \hat{\boldsymbol{Z}}, \boldsymbol{Y})$ is tractable as a

product of K distributions since:

$$\begin{aligned} p(\boldsymbol{\mu} \mid \hat{\boldsymbol{Z}}, \boldsymbol{Y}) &\propto p(\boldsymbol{\mu}) p(\boldsymbol{Y} \mid \hat{\boldsymbol{Z}}, \boldsymbol{\mu}), \\ &\propto \prod_{k=1}^{K} \mathcal{N}_d(\boldsymbol{\mu}_k \mid \boldsymbol{\nu}, \lambda) \times \prod_{i=1}^{n} \prod_{k=1}^{K} \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k)^{\hat{z}_{ik}}, \\ &\propto \prod_{k=1}^{K} \left\{ \mathcal{N}_d(\boldsymbol{\mu}_k \mid \boldsymbol{\nu}, \lambda) \times \prod_{i \in \hat{\mathcal{C}}_k} \mathcal{N}_p(\boldsymbol{y}_i \mid \boldsymbol{U}\boldsymbol{\mu}_k, \boldsymbol{S}_k) \right\}. \end{aligned}$$

Using the same reasoning as in Appendix B.2, the distributions into bracket happens to be (un-normalized) Gaussians with parameter $\left(\tilde{\boldsymbol{\mu}}_k(\hat{\boldsymbol{Z}}), \tilde{\mathbf{M}}_k(\hat{\boldsymbol{Z}})\right)$. Thus we have that:

$$p(\boldsymbol{\mu} \mid \hat{\boldsymbol{Z}}, \boldsymbol{Y}) = q\left(\boldsymbol{\mu} \mid \tilde{\boldsymbol{\mu}}(\hat{\boldsymbol{Z}}), \tilde{\mathbf{M}}(\hat{\boldsymbol{Z}})\right). \tag{64}$$

Finally, writing the expression of the variational bound $\mathcal{J}(\hat{q}; \hat{\boldsymbol{\vartheta}})$ with $\hat{q}$ defined in Equation (63), we get:

$$\begin{aligned} \mathcal{J}(\hat{q}; \hat{\boldsymbol{\vartheta}}) &= \mathbb{E}_{(\boldsymbol{\mu}, \boldsymbol{Z}) \sim \hat{q}} \left[\log\left(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\mu} \mid \hat{\boldsymbol{\vartheta}}\right) \right] + \mathrm{H}(\hat{q}), \\ &= \mathbb{E}_{\boldsymbol{\mu} \sim q\left(\boldsymbol{\mu} \mid \tilde{\boldsymbol{\mu}}(\hat{\boldsymbol{Z}}), \tilde{\mathbf{M}}(\hat{\boldsymbol{Z}})\right)} \left[\mathbb{E}_{\boldsymbol{Z} \sim \delta_{\hat{\boldsymbol{Z}}}} \left[\log\left(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\mu} \mid \hat{\boldsymbol{\vartheta}}\right) \right] \right] + \mathrm{H}\left(q\left(\boldsymbol{\mu} \mid \tilde{\boldsymbol{\mu}}(\hat{\boldsymbol{Z}}), \tilde{\mathbf{M}}(\hat{\boldsymbol{Z}})\right) \right), \\ &= \mathbb{E}_{\boldsymbol{\mu} \sim p(\boldsymbol{\mu} \mid \hat{\boldsymbol{Z}}, \boldsymbol{Y})} \left[\log\left(\boldsymbol{Y}, \hat{\boldsymbol{Z}}, \boldsymbol{\mu} \mid \hat{\boldsymbol{\vartheta}}\right) \right] + \mathrm{H}\left(p(\mu \mid \hat{\boldsymbol{Z}}, \boldsymbol{Y}) \right), \\ &= \log p(\boldsymbol{Y}, \hat{\boldsymbol{Z}}). \end{aligned}$$

Here, the second line used the fact that $\mathrm{H}(\delta_{\hat{\boldsymbol{Z}}}) = 0$, and the last equality used Proposition 2.1 with $\{\boldsymbol{Y}, \hat{\boldsymbol{Z}}\}$ as the observations and $\eta = \boldsymbol{\mu}$ as latent variables, giving the tightness of the lower bound when using the posterior distribution.

C Derivations of exact ICL

C.1 Marginal distribution of $\boldsymbol{Z}$: Dirichlet-Multinomial conjugacy

We recall the expression of ICL_{ex}:

$$\mathrm{ICL}_{ex}(\boldsymbol{Z}) = \log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\beta}) + \log p(\boldsymbol{Z} \mid \boldsymbol{\alpha}).$$

As explained in Equation (2.31) on page 46 and Equation (5.6) on page 100, the second term is analytically tractable when $\boldsymbol{\pi} \sim \mathcal{D}_K(\boldsymbol{\alpha} = (\alpha, \ldots, \alpha))$. This is obtained by an application of standard Dirichlet-Multinomial conjugacy, which is detailed in the following for the sake of completeness.

Let us denote by $C(\boldsymbol{x})$ the normalization constant of the Dirichlet distribution:

$$C(\boldsymbol{x}) = \frac{\prod_{k=1}^{K} \Gamma(x_k)}{\Gamma(\sum_{k=1}^{K} x_k)}.$$

Then, we want to compute the following integral:

$$\begin{aligned}
p(\boldsymbol{Z} \mid \alpha) &= \int_{\boldsymbol{\pi}} p(\boldsymbol{Z} \mid \boldsymbol{\pi}) p(\boldsymbol{\pi} \mid \alpha) \, \mathrm{d}\boldsymbol{\pi}, \\
&= \int_{\boldsymbol{\pi}} \left(\prod_{i=1}^{n} \mathcal{M}_K(\boldsymbol{z}_i \mid 1, \boldsymbol{\pi}) \right) \mathcal{D}_K(\boldsymbol{\pi} \mid \boldsymbol{\alpha}) \, \mathrm{d}\boldsymbol{\pi}, \\
&= \int_{\boldsymbol{\pi}} \left(\prod_{k=1}^{K} \pi_k^{\sum_i z_{ik}} \right) \frac{1}{C(\boldsymbol{\alpha})} \prod_{k=1}^{K} \pi_k^{\alpha - 1} \, \mathrm{d}\boldsymbol{\pi}, \\
&= \frac{C(\boldsymbol{n} + \boldsymbol{\alpha})}{C(\boldsymbol{\alpha})} \int_{\boldsymbol{\pi}} \mathcal{D}_K(\boldsymbol{\pi} \mid \boldsymbol{\alpha} + \boldsymbol{n}) \, \mathrm{d}\boldsymbol{\pi}, \\
&= \frac{C(\boldsymbol{n} + \boldsymbol{\alpha})}{C(\boldsymbol{\alpha})},
\end{aligned}$$

with $n_k = \sum_i z_{ik}$. Thus, we obtain the desired result as:

$$\log p(\boldsymbol{Z} \mid \alpha) = \log \left\{ \frac{\Gamma(\alpha K) \prod_k \Gamma(\alpha + n_k)}{\Gamma(\alpha)^K \Gamma(n + \alpha K)} \right\} \tag{65}$$

C.2 Exact ICL for mixture of multinomials

Proof of Proposition 5.1 on page 112. Here, $\boldsymbol{\theta} = (\boldsymbol{\theta})_k \in \Delta_p^K$ and the conditional likelihood, given $\boldsymbol{Z}$, of the MoM generative model is:

$$p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) = \prod_{k=1}^{K} \prod_{i=1}^{n} \mathcal{M}_p(\boldsymbol{y}_i \mid c_i, \boldsymbol{\theta}_k)^{z_{ik}},$$

We wish to integrate out the parameters $\boldsymbol{\theta} \sim \otimes_k \mathcal{D}_p(\boldsymbol{\beta} = (\beta, \dots, \beta))$. A use of Fubini's formula allows leveraging Dirichlet-Multinomial conjugacy for K different integrals:

$$\begin{aligned}
p(\boldsymbol{Y} \mid \boldsymbol{Z}, \beta) &= \int_{\boldsymbol{\theta}} p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{\theta}, \\
&= \prod_{k=1}^{K} \int_{\boldsymbol{\theta}_k} \prod_{i=1}^{n} \mathcal{M}_p(\boldsymbol{y}_i \mid c_i, \boldsymbol{\theta}_k)^{z_{ik}} \, \mathcal{D}_p(\boldsymbol{\theta}_k \mid \boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{\theta}_k, \\
&= \frac{1}{\prod_{i,j} y_{ij}!} \prod_{k=1}^{K} \int_{\boldsymbol{\theta}_k} \left(\prod_{j=1}^{p} \theta_{kj}^{\sum_i z_{ik} y_{ij}} \right) \frac{1}{C(\boldsymbol{\beta})} \prod_{j=1}^{p} \theta_{kj}^{\beta - 1} \, \mathrm{d}\boldsymbol{\theta}_k, \\
&= \frac{1}{\prod_{i,j} y_{ij}!} \prod_{k=1}^{K} \frac{C(\boldsymbol{o}_k)}{C(\boldsymbol{\beta})} \int_{\boldsymbol{\theta}_k} \mathcal{D}_p(\boldsymbol{\theta}_k \mid \boldsymbol{\beta} + \boldsymbol{o}_k) \, \mathrm{d}\boldsymbol{\theta}_k, \\
&= \frac{1}{\prod_{i,j} y_{ij}!} \times \prod_{k=1}^{K} \frac{\Gamma(\beta p) \prod_{j=1}^{p} \Gamma(o_{kj} + \beta)}{\Gamma(\beta)^p \, \Gamma(c_k + \beta p)},
\end{aligned}$$

with $o_{kj} = \sum_{i=1}^{n} z_{ik} y_{ij}$ and $c_k = \sum_{j=1}^{p} o_{kj}$. Finally, denote

$$B(\boldsymbol{Y}) = \frac{1}{\prod_{i,j} y_{ij}!},$$

which is independent of the partition $\boldsymbol{Z}$. Thus, taking the log concludes the proof. □

C.3 Exact ICL for the degree-corrected SBM

Proof of Proposition 5.3 on page 114. Putting $\boldsymbol{\theta} = (\boldsymbol{\Phi}^+, \boldsymbol{\Phi}^-, \boldsymbol{\Omega})$, the conditional likelihood, given $\boldsymbol{Z}$, of the generative model described in Equation (5.20) writes as:

$$\begin{aligned}
p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) &= \prod_{i,j}^{N} \prod_{k,l}^{K} \mathcal{P}(y_{ij} \mid \Phi_i^+ \Phi_j^- \Omega_{kl})^{z_{ik} z_{jl}}, \\
&= \frac{1}{\prod_{i,j} y_{ij}!} \times \prod_{i,j}^{N} (\Phi_i^+ \Phi_j^-)^{y_{ij}} \prod_{k,l}^{K} \Omega_{kl}^{z_{ik} z_{jl} y_{ij}} \exp(\Phi_i^+ \Phi_j^- \Omega_{kl} z_{ik} z_{jl}), \\
&= \frac{1}{\prod_{i,j} y_{ij}!} \times \prod_{i} (\Phi_i^+)^{d_i^+} \times \prod_{i} (\Phi_i^-)^{d_i^-} \times \prod_{k,l} \Omega_{kl}^{\nu_{kl}} \exp(-n_k n_l \Omega_{kl}),
\end{aligned} \tag{66}$$

$d_i^- = \sum_j y_{ij}$, $d_j^+ = \sum_i y_{ij}$ and $\nu_{kl} = \sum_{i,j} z_{ik} z_{jl} y_{ij}$. Calculating the ICL_{ex} implies to integrate over $\boldsymbol{\theta}$. Notice that Equation (66) is separable as the product of two parts, one depending on $\boldsymbol{\Phi}$ and the other on $\boldsymbol{\Omega}$. In the following, we detail calculations separately for both parts.

INTEGRATING OVER $\boldsymbol{\Phi}$ Recall that $\boldsymbol{\Phi}_k^{\cdot} = (\phi_i^{\cdot})_{i:z_{ik}=1}$ and that:

$$p(\boldsymbol{\Phi}_k^{\cdot}) = \frac{1}{\mathrm{vol}(\mathcal{S}_k)} \mathbb{1}_{\mathcal{S}_k}(\boldsymbol{\Phi}_k^{\cdot}), \quad \text{with } \mathbb{S}_k = \left\{ \boldsymbol{\Phi}_k^{\cdot} \in \mathbb{R}^{n_k} : \sum_{i:z_{ik}=1} \frac{\Phi_i^{\cdot}}{n_k} = 1 \right\}, \tag{67}$$

which is simply the simplex of dimension $(n_k - 1)$, rescaled by a factor n_k. Hence the volume of $\mathbb{S}_k$ is given by:

$$\int_{\mathbb{S}_k} \mathrm{d}\mathbf{\Phi}_k^{\cdot} = \frac{n_k^{n_k}}{(n_k-1)!}. \tag{68}$$

The situation is symmetric for $\mathbf{\Phi}_k^+$ or $\mathbf{\Phi}_k^-$. Thus, calculations are detailed only for the former. One needs to compute:

$$\begin{aligned}
\frac{(n_k-1)!}{n_k^{n_k}} \int_{\mathbb{S}_k} \prod_{i:z_{ik}=1} \left(\Phi_i^+\right)^{d_i^+} \mathrm{d}\mathbf{\Phi}_k^+ &= \frac{(n_k-1)!}{n_k^{n_k}} n_k^{d_k^+} \int_{\mathbb{S}_k} \prod_{i:z_{ik}=1} \left(\frac{\Phi_i^+}{n_k}\right)^{d_i^+} \mathrm{d}\mathbf{\Phi}_k^+, \\
&= \frac{(n_k-1)!}{n_k^{n_k}} n_k^{d_k^+} \int_{\Delta_{n_k}} \prod_{i:z_{ik}=1} (x_i)^{d_i^+} \left|n_k \mathbf{I}_{n_k}\right| \mathrm{d}\mathbf{x}, \\
&= \frac{n_k^{n_k}}{n_k^{n_k}} (n_k-1)!\, n_k^{d_k^+} C(\boldsymbol{a}_k) \int_{\Delta_{n_k}} \mathcal{D}_{n_k}\left(\boldsymbol{x} \mid \boldsymbol{a}_k = (d_i+1)_{i:z_{ik}=1}\right) \mathrm{d}\mathbf{x}, \\
&= (n_k-1)!\, n_k^{d_k^+} \frac{\prod_{i:z_{ik}=1} d_i^+!}{(n_k + d_k^+ - 1)!}, \\
&= \frac{(n_k-1)!}{(n_k + d_k^+ - 1)!} n_k^{d_k^+} \prod_{i:z_{ik}=1} d_i^+!,
\end{aligned} \tag{69}$$

with $d_k^+ = \sum_i z_{ik} d_i^+$. Then,

$$\begin{aligned}
\int_{\Phi^+} p(\Phi^+) \prod_i \left(\Phi_i^+\right)^{d_i^+} \mathrm{d}\mathbf{\Phi}^+ &= \int_{\prod_k \mathbb{S}_k} \prod_k \frac{(n_k-1)!}{n_k^{n_k}} \prod_{i:z_{ik}=1} \left(\Phi_i^+\right)^{d_i^+} \mathrm{d}(\mathbf{\Phi}_1^+, \ldots, \mathbf{\Phi}_K^+), \\
&= \prod_k \frac{(n_k-1)!}{n_k^{n_k}} \int_{\mathbb{S}_k} \prod_{i:z_{ik}=1} \left(\Phi_i^+\right)^{d_i^+} \mathrm{d}\mathbf{\Phi}_k^+, \\
&= \prod_k \frac{(n_k-1)!}{(n_k + d_k^+ - 1)!} n_k^{d_k^+} \prod_i d_i^+!.
\end{aligned} \tag{70}$$

INTEGRATING OVER $\mathbf{\Omega}$ This is done using a standard Gamma-Poisson conjugacy in each pair of clusters. Indeed:

$$\begin{aligned}
\int_{\Omega_{kl}} p(\Omega_{kl}) \Omega_{kl}^{\nu_{kl}} \exp(-n_k n_l \Omega_{kl})\, \mathrm{d}\Omega_{kl} &= \int_{\Omega_{kl}} \frac{1}{\beta} \Omega_{kl}^{\nu_{kl}} \exp\left(-(n_k n_l + \frac{1}{\beta})\Omega_{kl}\right) \mathrm{d}\Omega_{kl}, \\
&= \frac{\Gamma(\nu_{kl}+1)}{\beta\left(n_k n_l + \beta^{-1}\right)^{\nu_{kl}+1}}, \\
&= \frac{\nu_{kl}!\, \beta^{\nu_{kl}}}{\left(\beta n_k n_l + 1\right)^{\nu_{kl}+1}}.
\end{aligned} \tag{71}$$

Utimately, we have:

$$\begin{aligned} p(\boldsymbol{Y} \mid \boldsymbol{Z}) &= \int_{\boldsymbol{\theta}} p(\boldsymbol{Y}, \boldsymbol{\theta} \mid \boldsymbol{Z}) \, \mathrm{d}\boldsymbol{\theta}, \\ &= \frac{\prod_i d_i^+! d_i^-!}{\prod_{ij} y_{ij}!} \prod_k \frac{(n_k - 1)!}{(n_k + d_k^+ - 1)!} n_k^{d_k^+} \frac{(n_k - 1)!}{(n_k + d_k^- - 1)!} n_k^{d_k^-} \\ &\qquad \times \prod_{k,l} \frac{\nu_{kl}! \, \beta^{\nu_{kl}}}{(\beta n_k n_l + 1)^{\nu_{kl}+1}}. \end{aligned} \tag{72}$$

Putting

$$B(\boldsymbol{Y}) = \frac{\prod_i d_i^+! d_i^-!}{\prod_{ij} y_{ij}!},$$

and noticing that the latter does not depend on the partition $\boldsymbol{Z}$ concludes the proof. □

C.4 Exact ICL for the degree-corrected LBM

Proof of Proposition 5.4 on page 115. Putting $\boldsymbol{\theta} = (\boldsymbol{\Omega}, \boldsymbol{\Phi}^r, \boldsymbol{\Phi}^c)$, the conditional likelihood, given $\boldsymbol{Z}$, of the generative model described in Equation (5.24) writes as:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\theta}) = \prod_{i=1}^{n} \prod_{j=1}^{p} \prod_{k}^{K_r} \prod_{l}^{K_c} \mathcal{P}(y_{ij} \mid \Phi_i^r \Phi_j^c \Omega_{kl})^{z_{ik}^c z_{jl}^r}, \tag{73}$$

Calculations for each of the term in Equation (5.25) are similar to Appendix C.3, with a slight difference in the $B(\boldsymbol{Y})$ term. Indeed, the out (resp. in) degrees are now replaced by rows (resp. columns) degrees:

$$B(\boldsymbol{Y}) = \frac{\prod_i r_i! \prod_j c_j!}{\prod_{i,j} y_{ij}!}, \tag{74}$$

with $r_i = \sum_j y_{ij}$ and $c_j = \sum_i y_{ij}$ the row (resp. columns) degrees. □

Bibliography

Adamic, Lada A. and Natalie Glance (2005). "The Political Blogosphere and the 2004 U.S. Election: Divided They Blog". In: *Proceedings of the 3rd International Workshop on Link Discovery*. LinkKDD '05. New York, NY, USA: ACM, pp. 36–43 (cit. on p. 118).

Aggarwal, Charu C and ChengXiang Zhai (2012a). "A survey of text clustering algorithms". In: *Mining text data*. Springer, pp. 77–128 (cit. on p. 7).

Aggarwal, Charu C and ChengXiang Zhai (2012b). *Mining text data*. Springer Science & Business Media (cit. on p. 12).

Agrawal, Rakesh et al. (1998). "Automatic subspace clustering of high dimensional data for data mining applications". In: *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pp. 94–105 (cit. on p. 72).

Agresti, Alan (2003). *Categorical data analysis*. Vol. 482. John Wiley & Sons (cit. on p. 129).

Airoldi, Edoardo M et al. (2008). "Mixed membership stochastic blockmodels". In: *Journal of machine learning research* 9.Sep, pp. 1981–2014 (cit. on p. 83).

Alquier, Pierre and James Ridgway (June 2020). "Concentration of tempered posteriors and of their variational approximations". In: *Ann. Statist.* 48.3, pp. 1475–1497 (cit. on p. 25).

Anders, Simon and Wolfgang Huber (2010). "Differential expression analysis for sequence count data". In: *Genome biology* 11.10, R106 (cit. on p. 7).

Andrews, Jeffrey L and Paul D McNicholas (2013). "Using evolutionary algorithms for model-based clustering". In: *Pattern Recognition Letters* 34.9, pp. 987–992 (cit. on p. 101).

Archambeau, Cédric and Francis R Bach (2009). "Sparse probabilistic projections". In: *Advances in neural information processing systems*, pp. 73–80 (cit. on p. 34).

Auer, Paul L and RW Doerge (2010). "Statistical design and analysis of RNA sequencing data". In: *Genetics* 185.2, pp. 405–416 (cit. on p. 43).

Baek, Jangsun and Geoffrey J McLachlan (2008). "Mixtures of factor analyzers with common factor loadings for the clustering and visualisation of high-dimensional data". In: *Isaac Newton Institute for Mathematical Sciences, Preprints* 3.10 (cit. on pp. 39, 74, 77).

Baek, Jangsun, Geoffrey J McLachlan, and Lloyd K Flack (2009). "Mixtures of factor analyzers with common factor loadings: Applications to the clustering and visualization of high-dimensional data". In: *IEEE transactions on pattern analysis and machine intelligence* 32.7, pp. 1298–1309 (cit. on p. 87).

Balakrishnan, Sivaraman, Martin J. Wainwright, and Bin Yu (Feb. 2017). "Statistical guarantees for the EM algorithm: From population to sample-based analysis". In: *Ann. Statist.* 45.1, pp. 77–120 (cit. on p. 19).

Banerjee, Arindam et al. (2005). "Clustering with Bregman divergences". In: *Journal of machine learning research* 6.Oct, pp. 1705–1749 (cit. on p. 9).

Banfield, Jeffrey D and Adrian E Raftery (1993). "Model-based Gaussian and non-Gaussian clustering". In: *Biometrics*, pp. 803–821 (cit. on pp. 11, 38, 41, 75, 101).

Bar-Joseph, Ziv, David K. Gifford, and Tommi S. Jaakkola (June 2001). “Fast optimal leaf ordering for hierarchical clustering ”. In: *Bioinformatics* 17.1, S22–S29 (cit. on p. 112).
Barabasi, Albert-Laszlo and Zoltan N Oltvai (2004). “Network biology: understanding the cell’s functional organization”. In: *Nature reviews genetics* 5.2, pp. 101–113 (cit. on p. 8).
Bartholomew, David J, Martin Knott, and Irini Moustaki (2011). *Latent variable models and factor analysis: A unified approach*. Vol. 904. John Wiley & Sons (cit. on p. 33).
Bates, Douglas and Martin Maechler (2019). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.2-17 (cit. on p. 103).
Baudry, Jean-Patrick et al. (2010). “Combining Mixture Components for Clustering.” In: *Journal of computational and graphical statistics : a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America* 9 2, pp. 332–353 (cit. on pp. 47, 74, 101).
Bay, Herbert et al. (2008). “Speeded-up robust features (SURF)”. In: *Computer vision and image understanding* 110.3, pp. 346–359 (cit. on p. 6).
Bellman, Richard (1957). *Dynamic Programming*. Princeton University Press (cit. on pp. 12, 37).
Bengtsson, Henrik (2019). *future: Unified Parallel and Distributed Processing in R for Everyone*. R package version 1.13.0 (cit. on p. 103).
Benzécri, Jean-Paul (1973). *L’Analyse des données. Tome 2 : l’analyse des correspondances*. Vol. 2. Dunod Paris (cit. on p. 130).
Bergé, Laurent, Charles Bouveyron, and Stéphane Girard (2019). *HDclassif: An R Package for Model-Based Clustering and Discriminant Analysis of High-Dimensional Data*. R package version 2.2.0 (cit. on pp. 41, 87).
Berger, James O (2013). *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media (cit. on p. 44).
Bertoletti, Marco, Nial Friel, and Riccardo Rastelli (Aug. 2015). “Choosing the number of clusters in a finite mixture model using an exact integrated completed likelihood criterion”. In: *METRON* 73.2, pp. 177–199 (cit. on pp. 48, 100).
Bickel, Peter et al. (2013). “Asymptotic normality of maximum likelihood and its variational approximation for stochastic blockmodels”. In: *The Annals of Statistics* 41.4, pp. 1922–1943 (cit. on p. 28).
Biernacki, Christophe, Gilles Celeux, and Gérard Govaert (2000). “Assessing a mixture model for clustering with the integrated completed likelihood”. In: *IEEE transactions on pattern analysis and machine intelligence* 22.7, pp. 719–725 (cit. on pp. 46, 47, 56, 86, 99, 138).
Biernacki, Christophe, Gilles Celeux, and Gerard Govaert (2010). “Exact and monte carlo calculations of integrated likelihoods for the latent class model”. In: *Journal of Statistical Planning and Inference* 140, pp. 2991–3002 (cit. on pp. 47, 48, 100, 113).
Biernacki, Christophe and Gérard Govaert (1997). “Using the classification likelihood to choose the number of clusters”. In: *Computing Science and Statistics*, pp. 451–457 (cit. on pp. 21, 46).
Bishop, Christopher M (2006). *Pattern recognition and machine learning*. springer (cit. on pp. 16, 20, 27, 38, 39, 83, 132).
Blei, David M, Alp Kucukelbir, and Jon D McAuliffe (2017). “Variational inference: A review for statisticians”. In: *Journal of the American Statistical Association* 112.518, pp. 859–877 (cit. on p. 20).
Blei, David M, Andrew Y Ng, and Michael I Jordan (2003). “Latent dirichlet allocation”. In: *Journal of machine Learning research* 3.Jan, pp. 993–1022 (cit. on pp. 32, 35, 36, 55–57, 60, 83).

Bouchard, Guillaume (2007). "Efficient bounds for the softmax function, applications to inference in hybrid models". In: *Presentation at the Workshop for Approximate Bayesian Inference in Continuous/Hybrid Systems at NIPS-07* (cit. on p. 132).

Bouveyron, Charles and Camille Brunet (2011). "On the estimation of the latent discriminative subspace in the Fisher-EM algorithm". In: (cit. on p. 86).

Bouveyron, Charles and Camille Brunet (2012a). "Simultaneous model-based clustering and visualization in the Fisher discriminative subspace". In: *Statistics and Computing* 22.1, pp. 301–324 (cit. on pp. 13, 41, 72, 74, 77, 81–84, 87).

Bouveyron, Charles and Camille Brunet (2012b). "Theoretical and practical considerations on the convergence properties of the Fisher-EM algorithm". In: *Journal of Multivariate Analysis* 109, pp. 29–41 (cit. on p. 84).

Bouveyron, Charles, Camille Brunet, and Nicolas Jouvin (2020a). *FisherEM: The FisherEM Algorithm to Simultaneously Cluster and Visualize High-Dimensional Data*. R package version 1.5.2 (cit. on p. 87).

Bouveyron, Charles and Camille Brunet-Saumard (2014). "Discriminative variable selection for clustering with the sparse Fisher-EM algorithm". In: *Computational Statistics* 29.3-4, pp. 489–513 (cit. on pp. 13, 95, 126–128).

Bouveyron, Charles, Gilles Celeux, and Stéphane Girard (2011). "Intrinsic dimension estimation by maximum likelihood in isotropic probabilistic PCA". In: *Pattern Recognition Letters* 32.14, pp. 1706–1713 (cit. on p. 34).

Bouveyron, Charles, Stéphane Girard, and Cordelia Schmid (2007a). "High-dimensional data clustering". In: *Computational Statistics & Data Analysis* 52.1, pp. 502–519 (cit. on pp. 41, 77, 87).

Bouveyron, Charles, Stéphane Girard, and Cordelia Schmid (2007b). "High-dimensional discriminant analysis". In: *Communications in Statistics—Theory and Methods* 36.14, pp. 2607–2623 (cit. on p. 41).

Bouveyron, Charles, Pierre Latouche, and Pierre-Alexandre Mattei (2020b). "Exact dimensionality selection for Bayesian PCA". In: *Scandinavian Journal of Statistics* 47.1, pp. 196–211 (cit. on p. 34).

Bouveyron, Charles, Pierre Latouche, and Rawya Zreik (2018). "The stochastic topic block model for the clustering of vertices in networks with textual edges". In: *Statistics and Computing* 28.1, pp. 11–31 (cit. on pp. 27, 48, 53, 56, 58, 138).

Bouveyron, Charles et al. (Dec. 2017). "The Functional Latent Block Model for the Co-Clustering of Electricity Consumption Curves". In: *Journal of the Royal Statistical Society: Series C Applied Statistics* (cit. on p. 29).

Bouveyron, Charles et al. (2019). *Model-Based Clustering and Classification for Data Science: With Applications in R*. Vol. 50. Cambridge University Press (cit. on pp. 5, 37, 93, 98).

Bradley, Paul S, Olvi L Mangasarian, and W Nick Street (1997). "Clustering via concave minimization". In: *Advances in neural information processing systems*, pp. 368–374 (cit. on p. 9).

Bui, Quang Vu et al. (2017). "Combining Latent Dirichlet Allocation and K-means for documents clustering: effect of probabilistic based distance measures". In: *Asian Conference on Intelligent Information and Database Systems*. Springer, pp. 248–257 (cit. on p. 37).

Buntine, Wray (2002). "Variational extensions to EM and multinomial PCA". In: *European Conference on Machine Learning*. Springer, pp. 23–34 (cit. on pp. 34, 52).

Cadima, Jorge and Ian T Jolliffe (1995). "Loading and correlations in the interpretation of principle compenents". In: *Journal of applied Statistics* 22.2, pp. 203–214 (cit. on p. 126).

Carel, Lena (2017). *NMFEM: NMF-EM Algorithm*. R package version 1.0.3 (cit. on p. 42).

Carel, Léna and Pierre Alquier (2017). "Simultaneous dimension reduction and clustering via the NMF-EM algorithm". In: *Advances in Data Analysis and Classification*, pp. 1–30 (cit. on pp. 42, 52, 54, 59, 61).

Casella, George and Roger L Berger (2002). *Statistical inference*. Vol. 2. Duxbury Pacific Grove, CA (cit. on p. 22).

Celeux, Gilles (1985). "The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem". In: *Computational statistics quarterly* 2, pp. 73–82 (cit. on p. 25).

Celeux, Gilles (1998). "Bayesian inference for mixture: The label switching problem". In: *Compstat*. Springer, pp. 227–232 (cit. on p. 22).

Celeux, Gilles, Didier Chauveau, and Jean Diebolt (1995). *On Stochastic Versions of the EM Algorithm*. Research Report RR-2514. INRIA (cit. on p. 25).

Celeux, Gilles, Sylvia Frühwirth-Schnatter, and Christian Robert (2019). "Model Selection for Mixture Models-Perspectives and Strategies". In: *Handbook of Mixture Analysis*. CRC Press (cit. on p. 44).

Celeux, Gilles and Gérard Govaert (1992). "A classification EM algorithm for clustering and two stochastic versions". In: *Computational statistics & Data analysis* 14.3, pp. 315–332 (cit. on pp. 25, 74).

Celeux, Gilles and Gérard Govaert (1995). "Gaussian parsimonious clustering models". In: *Pattern recognition* 28.5, pp. 781–793 (cit. on p. 38).

Celeux, Gilles and Gilda Soromenho (1996). "An entropy criterion for assessing the number of clusters in a mixture model". In: *Journal of classification* 13.2, pp. 195–212 (cit. on p. 46).

Celisse, Alain, Jean-Jacques Daudin, and Laurent Pierre (2012). "Consistency of maximum-likelihood and variational estimators in the stochastic block model". In: *Electronic Journal of Statistics* 6, pp. 1847–1899 (cit. on p. 28).

Chamayou, Grégoire (2013). *Théorie du drone*. La fabrique éditions (cit. on p. 6).

Chan, Tsung-Han et al. (2015). "PCANet: A simple deep learning baseline for image classification?" In: *IEEE transactions on image processing* 24.12, pp. 5017–5032 (cit. on p. 30).

Chang, Wei-Chien (1983). "On using principal components before separating a mixture of two multivariate normal distributions". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 32.3, pp. 267–275 (cit. on pp. 72, 73, 88).

Chien, Jen-Tzung, Chao-Hsi Lee, and Zheng-Hua Tan (Sept. 2017). "Latent Dirichlet Mixture Model". In: *Neurocomputing* (cit. on p. 52).

Chiquet, Julien, Mahendra Mariadassou, and Stéphane Robin (2018). "Variational inference for probabilistic Poisson PCA". In: *The Annals of Applied Statistics* 12.4, pp. 2674–2698 (cit. on pp. 13, 33, 131).

Clarke, Robert et al. (2008). "The properties of high-dimensional data spaces: implications for exploring gene and protein expression data". In: *Nature reviews cancer* 8.1, pp. 37–49 (cit. on p. 12).

Cole, R. M. (1998). "Clustering with Genetic Algorithms". MA thesis. University of Western Australia (cit. on p. 101).

Coleman, Guy Barrett and Harry C Andrews (1979). "Image segmentation by clustering". In: *Proceedings of the IEEE* 67.5, pp. 773–785 (cit. on p. 6).

Côme, Etienne and Pierre Latouche (2015). "Model selection and clustering in stochastic block models based on the exact integrated complete data likelihood". In: *Statistical Modelling* 15.6, pp. 564–589 (cit. on pp. 48, 100, 104, 113, 114).

Corneli, Marco, Charles Bouveyron, and Pierre Latouche (2020). "Co-clustering of ordinal data via latent continuous random variables and not missing at random entries". In: *Journal of Computational and Graphical Statistics* (cit. on p. 29).

Corneli, Marco, Pierre Latouche, and Fabrice Rossi (2016). "Exact ICL maximization in a non-stationary temporal extension of the stochastic block model for dynamic networks". In: *Neurocomputing* 192. Advances in artificial neural networks, machine learning and computational intelligence, pp. 81 –91 (cit. on p. 100).

Cunningham, Ross B and David B Lindenmayer (2005). "Modeling count data of rare species: some statistical issues". In: *Ecology* 86.5, pp. 1135–1142 (cit. on p. 53).

Dasgupta, Abhijit and Adrian E Raftery (1998). "Detecting features in spatial point processes with clutter via model-based clustering". In: *Journal of the American statistical Association* 93.441, pp. 294–302 (cit. on p. 46).

d'Aspremont, Alexandre et al. (2005). "A direct formulation for sparse PCA using semidefinite programming". In: *Advances in neural information processing systems*, pp. 41–48 (cit. on p. 31).

Daudin, J-J, Franck Picard, and Stéphane Robin (2008). "A mixture model for random graphs". In: *Statistics and computing* 18.2, pp. 173–183 (cit. on pp. 10, 27, 28, 47).

Deerwester, Scott et al. (1990). "Indexing by latent semantic analysis". In: *Journal of the American society for information science* 41.6, pp. 391–407 (cit. on p. 32).

Defossez, Gautier et al. (2019). "Estimations nationales de l'incidence et de la mortalité par cancer en France métropolitaine entre 1990 et 2018". In: *Résultats préliminaires. Saint-Maurice (Fra): Santé publique France* (cit. on p. 66).

Dempster, Arthur P, Nan M Laird, and Donald B Rubin (1977). "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1, pp. 1–22 (cit. on pp. 17–19, 56).

Deng, Jia et al. (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255 (cit. on pp. 6, 12, 29).

Desrosières, Alain (2008). "Analyse des données et sciences humaines: comment cartographier le monde social ?" In: *Journal Electronique d'Histoire des Probabilités et de la Statistique* 4.2, p. 21 (cit. on p. 130).

Dhillon, Inderjit S, Yuqiang Guan, and Brian Kulis (2004). "Kernel k-means: spectral clustering and normalized cuts". In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 551–556 (cit. on p. 9).

Diaconis, Persi (1988). "Recent progress on de Finetti's notions of exchangeability". In: *Bayesian statistics* 3, pp. 111–125 (cit. on p. 21).

Dickey, James M (1983). "Multiple hypergeometric functions: Probabilistic interpretations and statistical uses". In: *Journal of the American Statistical Association* 78.383, pp. 628–637 (cit. on p. 35).

Ding, Chris, Tao Li, and Wei Peng (2008). "On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing". In: *Computational Statistics & Data Analysis* 52.8, pp. 3913–3927 (cit. on p. 32).

Ding, Hao et al. (2015). "GRAPHIE: graph based histology image explorer". In: *BMC bioinformatics* 16.S11, S10 (cit. on p. 6).

Donoho, David and Victoria Stodden (2004). "When does non-negative matrix factorization give a correct decomposition into parts?" In: *Advances in neural information processing systems*, pp. 1141–1148 (cit. on p. 32).

Draghici, Sorin (2016). *Statistics and data analysis for microarrays using R and bioconductor*. CRC Press (cit. on p. 6).

Duda, Richard O, Peter E Hart, and David G Stork (2000). *Pattern classification*. John Wiley & Sons (cit. on p. 73).

Durmus, Alain (2016). "High dimensional Markov chain Monte Carlo methods: theory, methods and applications". PhD thesis. Université Paris-Saclay (ComUE) (cit. on p. 12).

Dy, Jennifer G and Carla E Brodley (2004). "Feature selection for unsupervised learning". In: *Journal of machine learning research* 5.Aug, pp. 845–889 (cit. on p. 73).

Eckart, Carl and Gale Young (1936). "The approximation of one matrix by another of lower rank". In: *Psychometrika* 1.3, pp. 211–218 (cit. on p. 31).

Eddelbuettel, Dirk and James Joseph Balamuta (2017). "Extending *R* with *C++*: A Brief Introduction to *Rcpp*". In: *PeerJ Preprints* 5, e3188v1 (cit. on p. 102).

Eddelbuettel, Dirk and Conrad Sanderson (2014). "RcppArmadillo: Accelerating R with high-performance C++ linear algebra". In: *Computational Statistics and Data Analysis* 71, pp. 1054–1063 (cit. on p. 103).

Efron, Bradley et al. (2004). "Least angle regression". In: *The Annals of statistics* 32.2, pp. 407–499 (cit. on p. 127).

Eiben, A. E. and J. E. Smith (2004). *Introduction to Evolutionary Computing, 2^{nd} Edition*. Springer-Verlag (cit. on pp. 100, 104, 105).

Ellis, Ian O and Christopher W Elston (2006). "Histologic grade". In: *Breast pathology*. Elsevier, pp. 225–233 (cit. on pp. 66, 68).

Escofier-Cordier, Brigitte (1969). "L'analyse factorielle des correspondances". fr. In: *Cahiers du Bureau universitaire de recherche opérationnelle Série Recherche* 13, pp. 25–59 (cit. on p. 130).

Everett, B (2013). *An introduction to latent variable models*. Springer Science & Business Media (cit. on p. 16).

Everitt, Brian S., Sabine Landau, and Morven Leese (2011). *Cluster Analysis, Fifth Edition (Wiley Series in Probability and Statistics)*. 5th. Wiley Series in Probability and Statistics. Wiley (cit. on p. 11).

Fithian, William and Julie Josse (2017). "Multiple correspondence analysis and the multilogit bilinear model". In: *Journal of Multivariate Analysis* 157, pp. 87 –102 (cit. on pp. 130, 131).

Foley, Donald H. and John W Sammon (1975). "An optimal set of discriminant vectors". In: *IEEE Transactions on computers* 100.3, pp. 281–289 (cit. on pp. 82, 128).

Fordyce, James A et al. (2011). "A hierarchical Bayesian approach to ecological count data: a flexible tool for ecologists". In: *PloS one* 6.11, e26785 (cit. on p. 7).

Fortunato, Santo (2010). "Community detection in graphs". In: *Physics reports* 486.3-5, pp. 75–174 (cit. on p. 8).

Fraley, Chris (1998). "Algorithms for model-based Gaussian hierarchical clustering". In: *SIAM Journal on Scientific Computing* 20.1, pp. 270–281 (cit. on pp. 11, 101).

Friel, Nial et al. (2017). "Investigation of the widely applicable Bayesian information criterion". In: *Statistics and Computing* 27.3, pp. 833–844 (cit. on p. 44).

Fruhwirth-Schnatter, Sylvia, Gilles Celeux, and Christian P Robert (2019). *Handbook of mixture analysis*. Chapman and Hall/CRC (cit. on pp. 7, 12, 78, 101).

Fukunaga, Keinosuke (1990). *Introduction to Statistical Pattern Recognition (2nd Ed.)* USA: Academic Press Professional, Inc. (cit. on pp. 72, 82, 128).

Gaussier, Eric and Cyril Goutte (2005). "Relation between PLSA and NMF and implications". In: *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 601–602 (cit. on p. 32).

Ge, Rong et al. (2016). "Efficient algorithms for large-scale generalized eigenvector computation and canonical correlation analysis". In: *International Conference on Machine Learning*, pp. 2741–2750 (cit. on p. 84).
Gelman, Andrew et al. (2004). *Bayesian data analysis*. 2nd ed. Chapman & Hall/CRC (cit. on p. 100).
Ghahramani, Zoubin and Geoffrey E Hinton (1996). *The EM algorithm for mixtures of factor analyzers*. Tech. rep. Technical Report CRG-TR-96-1, University of Toronto (cit. on pp. 33, 38).
Ghojogh, Benyamin, Fakhri Karray, and Mark Crowley (2019). "Eigenvalue and generalized eigenvalue problems: Tutorial". In: *arXiv preprint arXiv:1903.11240* (cit. on pp. 30, 82).
Ghosh, Debashis and Arul M Chinnaiyan (2002). "Mixture modelling of gene expression data from microarray experiments". In: *Bioinformatics* 18.2, pp. 275–286 (cit. on p. 72).
Giraud, Christophe (2014). *Introduction to high-dimensional statistics*. Vol. 138. CRC Press (cit. on p. 12).
Gleiser, P. M. and L. Danon (2003). "COMMUNITY STRUCTURE IN JAZZ". In: *Advances in Complex Systems* 06.04, pp. 565–573 (cit. on p. 119).
Goffman, Casper (1969). "And What is Your Erdös Number?" In: *The American Mathematical Monthly* 76.7, pp. 791–791 (cit. on p. 8).
Gollini, Isabella and Thomas Brendan Murphy (2014). "Mixture of latent trait analyzers for model-based clustering of categorical data". In: *Statistics and Computing* 24.4, pp. 569–588 (cit. on p. 132).
Gonzalez, Edward F and Yin Zhang (2005). *Accelerating the Lee-Seung algorithm for non-negative matrix factorization*. Tech. rep. (cit. on p. 32).
Govaert, Gérard and Mohamed Nadif (2003). "Clustering with block mixture models". In: *Pattern Recognition* 36.2, pp. 463–473 (cit. on p. 10).
Govaert, Gérard and Mohamed Nadif (2008). "Block clustering with Bernoulli mixture models: Comparison of different approaches". In: *Computational Statistics & Data Analysis* 52.6, pp. 3233–3245 (cit. on p. 29).
Govaert, Gérard and Mohamed Nadif (2013). *Co-clustering: models, algorithms and applications*. John Wiley & Sons (cit. on p. 28).
Govaert, Gérard and Mohamed Nadif (2010). "Latent Block Model for Contingency Table". In: *Communications in Statistics - Theory and Methods* 39.3, pp. 416–425 (cit. on pp. 29, 98, 99, 114).
Gower, John C and Garmt B Dijksterhuis (2004). *Procrustes problems*. Vol. 30. Oxford University Press on Demand (cit. on p. 127).
Greenacre, Michael and Jorg Blasius (2006). *Multiple correspondence analysis and related methods*. CRC press (cit. on p. 130).
Groenen, Patrick JF and Julie Josse (2016). "Multinomial multiple correspondence analysis". In: *arXiv preprint arXiv:1603.03174* (cit. on p. 131).
Guo, Yue-Fei et al. (2003). "A generalized Foley–Sammon transform based on generalized fisher discriminant criterion and its application to face recognition". In: *Pattern Recognition Letters* 24.1-3, pp. 147–158 (cit. on pp. 96, 128, 129).
Guyon, Isabelle and André Elisseeff (2003). "An introduction to variable and feature selection". In: *Journal of machine learning research* 3.Mar, pp. 1157–1182 (cit. on p. 12).
Hajibabaei, Mehrdad et al. (2007). "DNA barcoding: how it complements taxonomy, molecular phylogenetics and population genetics". In: *TRENDS in Genetics* 23.4, pp. 167–172 (cit. on p. 6).
Hamamoto, Yoshihiko et al. (1991). "A note on the orthonormal discriminant vector method for feature extraction". In: *Pattern recognition* 24.7, pp. 681–684 (cit. on pp. 82, 86).

Hansen, Mark H and Bin Yu (2001). "Model selection and the principle of minimum description length". In: *Journal of the American Statistical Association* 96.454, pp. 746–774 (cit. on p. 44).

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media (cit. on p. 31).

Haughton, Dominique MA (1988). "On the choice of a model to fit data from an exponential family". In: *The Annals of Statistics* 16.1, pp. 342–355 (cit. on p. 45).

Heller, Katherine A and Zoubin Ghahramani (2005). "Bayesian hierarchical clustering". In: *Proceedings of the 22nd international conference on Machine learning*. ACM, pp. 297–304 (cit. on pp. 11, 101).

Hofmann, Thomas (1999). "Probabilistic latent semantic analysis". In: *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., pp. 289–296 (cit. on p. 32).

Hornik, Kurt and Bettina Grün (2011). "topicmodels: An R package for fitting topic models". In: *Journal of Statistical Software* 40.13, pp. 1–30 (cit. on p. 61).

Hotelling, Harold (1933). "Analysis of a complex of statistical variables into principal components." In: *Journal of educational psychology* 24.6, p. 417 (cit. on pp. 29, 30).

Hruschka, E. R. et al. (Mar. 2009). "A Survey of Evolutionary Algorithms for Clustering". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 39.2, pp. 133–155 (cit. on pp. 101, 104).

Hubert, Lawrence and Phipps Arabie (1985). "Comparing partitions". In: *Journal of classification* 2.1, pp. 193–218 (cit. on p. 87).

Husson, François, Julie Josse, and Gilbert Saporta (2016). "Jan de Leeuw and the French school of data analysis". In: *Journal of Statistical Software* (cit. on p. 130).

Jaakkola, Tommi and Michael Jordan (1997). "A variational approach to Bayesian logistic regression models and their extensions". In: *Sixth International Workshop on Artificial Intelligence and Statistics*. Vol. 82. 4 (cit. on p. 132).

Jaakkola, Tommi S and Michael I Jordan (2000). "Bayesian parameter estimation via variational methods". In: *Statistics and Computing* 10.1, pp. 25–37 (cit. on p. 19).

Jégou, Hervé et al. (2010). "Aggregating local descriptors into a compact image representation". In: *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, pp. 3304–3311 (cit. on p. 6).

Jia, Yangqing, Feiping Nie, and Changshui Zhang (2009). "Trace ratio problem revisited". In: *IEEE Transactions on Neural Networks* 20.4, pp. 729–735 (cit. on pp. 128, 129).

Jin, Chi et al. (2016). "Local maxima in the likelihood of gaussian mixture models: Structural results and algorithmic consequences". In: *Advances in neural information processing systems*, pp. 4116–4124 (cit. on p. 25).

Jolliffe, I.T. (2002). *Principal Component Analysis*. 2nd. Springer (cit. on p. 12).

Jouvin, Nicolas (2020). *MoMPCA: Inference and Clustering for Mixture of Multinomial Principal Component Analysis*. R package version 1.0.0 (cit. on p. 53).

Kallenberg, Olav (2006). *Probabilistic symmetries and invariance principles*. Springer Science & Business Media (cit. on p. 21).

Karrer, Brian and Mark EJ Newman (2011). "Stochastic blockmodels and community structure in networks". In: *Physical review E* 83.1, p. 016107 (cit. on pp. 98, 113).

Keribin, Christine (2000). "Consistent estimation of the order of mixture models". In: *Sankhyā: The Indian Journal of Statistics, Series A*, pp. 49–66 (cit. on p. 45).

Keribin, Christine et al. (2015). "Estimation and selection for the latent block model on categorical data". In: *Statistics and Computing* 25.6, pp. 1201–1216 (cit. on p. 29).

Khan, Mohammad Emtiyaz E et al. (2010). "Variational bounds for mixed-data factor analysis". In: *Advances in Neural Information Processing Systems*, pp. 1108–1116 (cit. on p. 132).

Kingma, Diederik P. and Max Welling (2019). "An Introduction to Variational Autoencoders". In: *Foundations and Trends® in Machine Learning* 12.4, pp. 307–392 (cit. on p. 21).

Kokiopoulou, Effrosini, Jie Chen, and Yousef Saad (2011). "Trace optimization and eigenproblems in dimension reduction methods". In: *Numerical Linear Algebra with Applications* 18.3, pp. 565–602 (cit. on pp. 31, 96).

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*, pp. 1097–1105 (cit. on p. 6).

Lakhani, Sunil R (2012). *WHO Classification of Tumours of the Breast*. International Agency for Research on Cancer (cit. on pp. 6, 66, 67).

Lange, Kenneth (2016). *MM optimization algorithms*. SIAM (cit. on pp. 19, 32).

Latouche, Pierre, Etienne Birmelé, and Christophe Ambroise (2011). "Overlapping stochastic block models with application to the french political blogosphere". In: *The Annals of Applied Statistics* 5.1, pp. 309–336 (cit. on pp. 27, 132).

Lebarbier, Émilie and Tristan Mary-Huard (2006). "Une introduction au critère BIC: fondements théoriques et interprétation". In: *Journal de la Société française de statistique* 147.1, pp. 39–57 (cit. on p. 45).

Lebaron, Frédéric and Brigitte Le Roux (2015). "La méthodologie de Pierre Bourdieu en action". In: *Espace culturel, espace social et analyse des données. Paris: Dunod* (cit. on p. 130).

Lee, Daniel D and H Sebastian Seung (1999). "Learning the parts of objects by non-negative matrix factorization". In: *Nature* 401.6755, p. 788 (cit. on p. 31).

Lee, Daniel D and H Sebastian Seung (2001). "Algorithms for non-negative matrix factorization". In: *Advances in neural information processing systems*, pp. 556–562 (cit. on pp. 31, 32, 52, 54).

Leroux, Brian G (1992). "Consistent estimation of a mixing distribution". In: *The Annals of Statistics*, pp. 1350–1360 (cit. on p. 45).

Levy, Omer and Yoav Goldberg (2014). "Neural word embedding as implicit matrix factorization". In: *Advances in neural information processing systems*, pp. 2177–2185 (cit. on p. 32).

Liu, Lin et al. (2016). "An overview of topic modeling and its current applications in bioinformatics". In: *SpringerPlus* 5.1, p. 1608 (cit. on p. 37).

Liu, Zikuan et al. (2006). "Online EM algorithm for mixture with application to internet traffic modeling". In: *Computational statistics & data analysis* 50.4, pp. 1052–1071 (cit. on p. 8).

Lomet, Aurore (2012). "Sélection de modèle pour la classification croisée de données continues". PhD thesis (cit. on p. 29).

Lomet, Aurore, Gerard Govaert, and Yves Grandvalet (2012). "An approximation of the integrated classification likelihood for the latent block model". In: *2012 IEEE 12th International Conference on Data Mining Workshops*. IEEE, pp. 147–153 (cit. on pp. 47, 48).

Lowe, David G (1999). "Object recognition from local scale-invariant features". In: *Proceedings of the seventh IEEE international conference on computer vision*. Vol. 2. Ieee, pp. 1150–1157 (cit. on p. 6).

Lu, Gui-Fu, Jian Zou, and Yong Wang (2016). "A new and fast implementation of orthogonal LDA algorithm and its incremental extension". In: *Neural Processing Letters* 43.3, pp. 687–707 (cit. on p. 82).

MacKay, David J. C. (2002). *Information Theory, Inference & Learning Algorithms*. USA: Cambridge University Press (cit. on p. 44).

MacQueen, James (1967). "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA, pp. 281–297 (cit. on p. 9).

Mairal, Julien et al. (2010). "Online learning for matrix factorization and sparse coding." In: *Journal of Machine Learning Research* 11.1 (cit. on p. 30).

Mariadassou, Mahendra and Catherine Matias (2015). "Convergence of the groups posterior distribution in latent or stochastic block models". In: *Bernoulli* 21.1, pp. 537–573 (cit. on p. 29).

Mariadassou, Mahendra, Stéphane Robin, and Corinne Vacher (2010). "Uncovering latent structure in valued graphs: a variational approach". In: *The Annals of Applied Statistics* 4.2, pp. 715–742 (cit. on pp. 27, 99).

Marin, Jean-Michel et al. (Nov. 2012). "Approximate Bayesian computational methods". In: *Statistics and Computing* 22.6 (cit. on p. 21).

Matias, Catherine and Stéphane Robin (2014). "Modeling heterogeneity in random graphs through latent space models: a selective review". In: *ESAIM: Proceedings and Surveys* 47, pp. 55–74 (cit. on p. 27).

Mattei, Pierre-Alexandre (2017). "Sélection de modèles parcimonieux pour l'apprentissage statistique en grande dimension". PhD thesis (cit. on p. 44).

Mattei, Pierre-Alexandre, Charles Bouveyron, and Pierre Latouche (2016). "Globally sparse probabilistic PCA". In: *Artificial Intelligence and Statistics*, pp. 976–984 (cit. on pp. 12, 13, 34).

Maugis, Cathy, Gilles Celeux, and Marie-Laure Martin-Magniette (2009). "Variable selection for clustering with Gaussian mixture models". In: *Biometrics* 65.3, pp. 701–709 (cit. on p. 12).

McCutcheon, Allan L (1987). *Latent class analysis*. 64. Sage (cit. on p. 43).

McLachlan, Geoffrey and David Peel (2000). *Finite Mixture Models*. John Wiley & Sons, Inc. (cit. on p. 98).

McLachlan, Geoffrey J and Thriyambakam Krishnan (2007). *The EM algorithm and extensions*. Vol. 382. John Wiley & Sons (cit. on pp. 9, 84).

McLachlan, Geoffrey J and David Peel (2004). *Finite mixture models*. John Wiley & Sons (cit. on pp. 9, 21, 25, 46, 72).

McNicholas, Paul D. et al. (2019). *pgmm: Parsimonious Gaussian Mixture Models*. R package version 1.2.4 (cit. on pp. 41, 87).

McNicholas, Paul David and Thomas Brendan Murphy (2008). "Parsimonious Gaussian mixture models". In: *Statistics and Computing* 18.3, pp. 285–296 (cit. on pp. 41, 77, 87, 93).

McParland, Damien and Thomas Brendan Murphy (2019). "Mixture modelling of high-dimensional data". In: *Handbook of Mixture Analysis*. Chapman and Hall/CRC, pp. 239–270 (cit. on p. 37).

Meng, Xiao-Li and David Van Dyk (1997). "The EM algorithm—an old folk-song sung to a fast new tune". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 59.3, pp. 511–567 (cit. on p. 41).

Mika, Sebastian et al. (1999). "Kernel PCA and de-noising in feature spaces". In: *Advances in neural information processing systems*, pp. 536–542 (cit. on p. 31).

Mikolov, Tomas et al. (2013). "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems*, pp. 3111–3119 (cit. on p. 32).
Minka, Thomas (2000). *Estimating a Dirichlet distribution* (cit. on p. 112).
Minka, Thomas P (2001). "Automatic choice of dimensionality for PCA". In: *Advances in neural information processing systems*, pp. 598–604 (cit. on p. 34).
Mirsky, Leon (1960). "Symmetric gauge functions and unitarily invariant norms". In: *The quarterly journal of mathematics* 11.1, pp. 50–59 (cit. on p. 31).
Montanari, Angela and Cinzia Viroli (2010). "Heteroscedastic factor mixture analysis". In: *Statistical Modelling* 10.4, pp. 441–460 (cit. on pp. 40, 77, 87).
Morris, Carl N (1983). "Parametric empirical Bayes inference: theory and applications". In: *Journal of the American statistical Association* 78.381, pp. 47–55 (cit. on p. 83).
Moustaki, Irini and Martin Knott (2000). "Generalized latent trait models". In: *Psychometrika* 65.3, pp. 391–411 (cit. on p. 131).
Mulaik, Stanley A (2009). *Foundations of factor analysis*. CRC press (cit. on p. 30).
Murtagh, Fionn and Adrian E Raftery (1984). "Fitting straight lines to point patterns". In: *Pattern recognition* 17.5, pp. 479–483 (cit. on p. 101).
Nelder, John Ashworth and Robert WM Wedderburn (1972). "Generalized linear models". In: *Journal of the Royal Statistical Society: Series A (General)* 135.3, pp. 370–384 (cit. on p. 33).
Newman, M. E. J. and M. Girvan (Feb. 2004). "Finding and evaluating community structure in networks". In: *Phys. Rev. E* 69 (2), p. 026113 (cit. on pp. 8, 119).
Newman, M. E. J. and Gesine Reinert (Aug. 2016). "Estimating the Number of Communities in a Network". In: *Phys. Rev. Lett.* 117 (7), p. 078301 (cit. on pp. 100, 114).
Ng, Andrew Y, Michael I Jordan, and Yair Weiss (2002). "On spectral clustering: Analysis and an algorithm". In: *Advances in neural information processing systems*, pp. 849–856 (cit. on p. 9).
Ngo, Thanh T, Mohammed Bellalij, and Yousef Saad (2012). "The trace ratio optimization problem". In: *SIAM review* 54.3, pp. 545–569 (cit. on pp. 128, 129).
Nie, Feiping et al. (2008). "Trace ratio criterion for feature selection." In: *AAAI*. Vol. 2, pp. 671–676 (cit. on p. 73).
Nocedal, Jorge and Stephen Wright (2006). *Numerical optimization*. Springer Science & Business Media (cit. on p. 19).
Nowak, Eric, Frédéric Jurie, and Bill Triggs (2006). "Sampling strategies for bag-of-features image classification". In: *European conference on computer vision*. Springer, pp. 490–503 (cit. on p. 6).
Nowicki, Krzysztof and Tom A B Snijders (2001). "Estimation and prediction for stochastic blockstructures". In: *Journal of the American statistical association* 96.455, pp. 1077–1087 (cit. on pp. 26, 98).
O'hara, Robert B and D Johan Kotze (2010). "Do not log-transform count data". In: *Methods in ecology and Evolution* 1.2, pp. 118–122 (cit. on p. 7).
Okada, Toshiniko and Shingo Tomita (1985). "An optimal orthonormal system for discriminant analysis". In: *Pattern Recognition* 18.2, pp. 139–144 (cit. on pp. 82, 86).
Osborne, J (2005). "Notes on the use of data transformations". In: *Practical assessment, research and evaluation* 9.1, pp. 42–50 (cit. on p. 7).
Paatero, Pentti and Unto Tapper (1994). "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values". In: *Environmetrics* 5.2, pp. 111–126 (cit. on p. 31).

Pagès, Jérôme (2014). *Multiple factor analysis by example using R*. CRC Press (cit. on p. 130).
Palla, Gergely, Albert-László Barabási, and Tamás Vicsek (2007). "Quantifying social group evolution". In: *Nature* 446.7136, pp. 664–667 (cit. on p. 7).
Paquet, Ulrich (2008). "Bayesian inference forlatent variable models". PhD thesis (cit. on p. 16).
Parsons, Lance, Ehtesham Haque, and Huan Liu (2004). "Subspace clustering for high dimensional data: a review". In: *Acm Sigkdd Explorations Newsletter* 6.1, pp. 90–105 (cit. on p. 72).
Pearson, Karl (1901). "LIII. On lines and planes of closest fit to systems of points in space". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11, pp. 559–572 (cit. on pp. 6, 29).
Peixoto, Tiago P (2012). "Entropy of stochastic blockmodel ensembles". In: *Physical Review E* 85.5, p. 056122 (cit. on p. 26).
Peixoto, Tiago P (2014). "Hierarchical block structures and high-resolution model selection in large networks". In: *Physical Review X* 4.1, p. 011047 (cit. on p. 101).
Podosinnikova, Anastasia, Francis Bach, and Simon Lacoste-Julien (2015). "Rethinking lda: moment matching for discrete ica". In: *Advances in Neural Information Processing Systems*, pp. 514–522 (cit. on p. 35).
Qiao, Zhihua, Lan Zhou, and Jianhua Z Huang (2009). "Sparse Linear Discriminant Analysis with Applications to High Dimensional Low Sample Size Data." In: *International Journal of Applied Mathematics* 39.1 (cit. on pp. 95, 126, 127).
Qin, Tai and Karl Rohe (2013). "Regularized Spectral Clustering under the Degree-Corrected Stochastic Blockmodel". In: *Proceedings of Nips* (cit. on p. 116).
R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria (cit. on p. 102).
Raftery, Adrian E (1995). "Bayesian model selection in social research". In: *Sociological methodology*, pp. 111–163 (cit. on p. 45).
Raftery, Adrian E and Nema Dean (2006). "Variable Selection for Model-Based Clustering". In: *Journal of the American Statistical Association* 101.473, pp. 168–178 (cit. on p. 12).
Ramos, Juan (2003). "Using tf-idf to determine word relevance in document queries". In: *Proceedings of the first instructional conference on machine learning*. Vol. 242. Piscataway, NJ, pp. 133–142 (cit. on p. 7).
Rand, William M (1971). "Objective criteria for the evaluation of clustering methods". In: *Journal of the American Statistical association* 66.336, pp. 846–850 (cit. on p. 61).
Ranganath, Rajesh, Sean Gerrish, and David Blei (2014). "Black box variational inference". In: *Artificial Intelligence and Statistics*, pp. 814–822 (cit. on p. 21).
Rathnayake, Suren et al. (2019). *EMMIXmfa: Mixture Models with Component-Wise Factor Analyzers*. R package version 2.0.11 (cit. on pp. 40, 87).
Rau, Andrea et al. (Nov. 2011). *Clustering high-throughput sequencing data with Poisson mixture models*. Research Report RR-7786. INRIA, p. 36 (cit. on pp. 43, 61).
Rau, Andrea et al. (Jan. 2015). "Co-expression analysis of high-throughput transcriptome sequencing data with Poisson mixture models". In: *Bioinformatics* 31.9, pp. 1420–1427 (cit. on pp. 43, 52).
Rigouste, Loïs, Olivier Cappé, and François Yvon (2007). "Inference and evaluation of the multinomial mixture model for text clustering". In: *Information processing & management* 43.5, pp. 1260–1280 (cit. on p. 42).
Ringnér, Markus (2008). "What is principal component analysis?" In: *Nature biotechnology* 26.3, pp. 303–304 (cit. on p. 30).

Riolo, Maria A. et al. (2017). "Efficient method for estimating the number of communities in a network". In: *Phys. Rev. E* 96 (3), p. 032310 (cit. on pp. 100, 114).
Robert, Christian (2007). *The Bayesian choice: from decision-theoretic foundations to computational implementation*. Springer Science & Business Media (cit. on p. 47).
Robert, Christian and George Casella (2013). *Monte Carlo statistical methods*. Springer Science & Business Media (cit. on p. 21).
Robert, Christian P, George Casella, and George Casella (2010). *Introducing monte carlo methods with r*. Vol. 18. Springer (cit. on p. 23).
Robert, Christian P and Judith Rousseau (2016). "Some comments about A Bayesian criterion for singular models by M. Drton and M. Plummer". In: *arXiv preprint arXiv:1610.02503* (cit. on p. 45).
Roeder, Kathryn and Larry Wasserman (1997). "Practical Bayesian density estimation using mixtures of normals". In: *Journal of the American Statistical Association* 92.439, pp. 894–902 (cit. on p. 46).
Ronald L. Graham Donald Knuth, Oren Patashnik (1988). *Concrete mathematics: a foundation for computer science*. Addison-Wesley Pub (Sd) (cit. on p. 8).
Roweis, Sam T (1998). "EM algorithms for PCA and SPCA". In: *Advances in neural information processing systems*, pp. 626–632 (cit. on p. 33).
Sanderson, Conrad and Ryan Curtin (2019). "Practical sparse matrices in C++ with hybrid storage and template-based expression optimisation". In: *Mathematical and Computational Applications* 24.3 (cit. on p. 103).
Schwarz, Gideon (1978). "Estimating the dimension of a model". In: *The annals of statistics* 6.2, pp. 461–464 (cit. on p. 45).
Scott, David W and James R Thompson (1983). "Probability density estimation in higher dimensions". In: *Computer Science and Statistics: Proceedings of the fifteenth symposium on the interface*. Vol. 528. North-Holland, Amsterdam, pp. 173–179 (cit. on p. 12).
Scrucca, Luca (2010). "Dimension reduction for model-based clustering". In: *Statistics and Computing* 20.4, pp. 471–484 (cit. on p. 73).
Scrucca, Luca (2016). "Genetic algorithms for subset selection in model-based clustering". In: *Unsupervised Learning Algorithms*. Springer, pp. 55–70 (cit. on p. 101).
Scrucca, Luca et al. (2016). "mclust 5: clustering, classification and density estimation using Gaussian finite mixture models". In: *The R journal* 8.1, p. 289 (cit. on p. 38).
Selosse, Margot, Julien Jacques, and Christophe Biernacki (2020). "Model-based co-clustering for mixed type data". In: *Computational Statistics and Data Analysis* 144, p. 106866 (cit. on p. 29).
Shen-Orr, Shai S et al. (2002). "Network motifs in the transcriptional regulation network of Escherichia coli". In: *Nature genetics* 31.1, pp. 64–68 (cit. on p. 8).
Silvestre, Cláudia, Margarida GMS Cardoso, and Mário AT Figueiredo (2014). "Identifying the number of clusters in discrete mixture models". In: *arXiv preprint arXiv:1409.7419* (cit. on p. 48).
Sneath, Peter HA (1957). "The application of computers to taxonomy". In: *Microbiology* 17.1, pp. 201–226 (cit. on pp. 10, 101).
Sokal, R. R. and C. D. Michener (1958). "A statistical method for evaluating systematic relationships". In: *University of Kansas Science Bulletin* 38, pp. 1409–1438 (cit. on pp. 10, 101).
Sorlie, Therese et al. (2003). "Repeated observation of breast tumor subtypes in independent gene expression data sets." In: *Proceedings of the National Academy of Sciences of the United States of America* 100.14, pp. 8418–8423 (cit. on p. 66).

St-Pierre, Anne P, Violaine Shikon, and David C Schneider (2018). "Count data in biology—Data transformation or model reformation?" In: *Ecology and evolution* 8.6, pp. 3077–3085 (cit. on p. 7).

Steinbach, Michael, Levent Ertöz, and Vipin Kumar (2004). "The challenges of clustering high dimensional data". In: *New directions in statistical physics*. Springer, pp. 273–309 (cit. on p. 12).

Steyvers, Mark and Tom Griffiths (2007). "Probabilistic topic models". In: *Handbook of latent semantic analysis* 427.7, pp. 424–440 (cit. on p. 35).

Tessier, Damien et al. (2006). "Evolutionary latent class clustering of qualitative data". In: (cit. on pp. 48, 100, 101, 104, 113).

Tibshirani, Robert (1996). "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 267–288 (cit. on p. 12).

Tierney, Luke and Joseph B Kadane (1986). "Accurate approximations for posterior moments and marginal densities". In: *Journal of the american statistical association* 81.393, pp. 82–86 (cit. on p. 45).

Tipping, Michael E (1999). "Probabilistic visualisation of high-dimensional binary data". In: *Advances in neural information processing systems*, pp. 592–598 (cit. on p. 131).

Tipping, Michael E and Christopher M Bishop (1999a). "Mixtures of probabilistic principal component analyzers". In: *Neural computation* 11.2, pp. 443–482 (cit. on pp. 13, 38).

Tipping, Michael E and Christopher M Bishop (1999b). "Probabilistic principal component analysis". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61.3, pp. 611–622 (cit. on pp. 33, 34).

Torre, Fernando De la and Takeo Kanade (2006). "Discriminative cluster analysis". In: *Proceedings of the 23rd international conference on Machine learning*, pp. 241–248 (cit. on p. 73).

Vaart, Aad W Van der (2000). *Asymptotic statistics*. Vol. 3. Cambridge university press (cit. on p. 45).

Van Erven, Tim and Peter Harremos (2014). "Rényi divergence and Kullback-Leibler divergence". In: *IEEE Transactions on Information Theory* 60.7, pp. 3797–3820 (cit. on p. 7).

Verleysen, John A Lee; Michel (2007). *Nonlinear dimensionality reduction*. Information science and statistics. Springer (cit. on p. 13).

Vinh, Nguyen Xuan, Julien Epps, and James Bailey (2010). "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance". In: *Journal of Machine Learning Research* 11.Oct, pp. 2837–2854 (cit. on p. 117).

Viroli, Cinzia (2012). *FactMixtAnalysis: Factor Mixture Analysis with covariates*. R package version 1.0 (cit. on p. 41).

Wainwright, Martin J and Michael Irwin Jordan (2008). *Graphical models, exponential families, and variational inference*. Now Publishers Inc (cit. on p. 19).

Wallach, Hanna Megan (2008). "Structured topic models for language". PhD thesis. University of Cambridge (cit. on p. 52).

Wang, Alex et al. (2018). "Glue: A multi-task benchmark and analysis platform for natural language understanding". In: *arXiv preprint arXiv:1804.07461* (cit. on p. 29).

Wang, Huan et al. (2007). "Trace ratio vs. ratio trace for dimensionality reduction". In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8 (cit. on pp. 96, 128, 129).

Wang, Yuchung J and George Y Wong (1987). "Stochastic blockmodels for directed graphs". In: *Journal of the American Statistical Association* 82.397, pp. 8–19 (cit. on pp. 26, 98).

Wang, Zhong, Mark Gerstein, and Michael Snyder (2009). "RNA-Seq: a revolutionary tool for transcriptomics". In: *Nature reviews genetics* 10.1, pp. 57–63 (cit. on p. 43).
Ward Jr, Joe H (1963). "Hierarchical grouping to optimize an objective function". In: *Journal of the American statistical association* 58.301, pp. 236–244 (cit. on pp. 11, 101).
Watanabe, Kazuho et al. (2010). "Simultaneous clustering and dimensionality reduction using variational Bayesian mixture model". In: *Classification as a Tool for Research*. Springer, pp. 81–89 (cit. on p. 52).
Watanabe, Sumio (2013). "A widely applicable Bayesian information criterion". In: *Journal of Machine Learning Research* 14.Mar, pp. 867–897 (cit. on p. 45).
Wei, Xin and Chunguang Li (2013). "Bayesian mixtures of common factor analyzers: Model, variational inference, and applications". In: *Signal processing* 93.11, pp. 2894–2905 (cit. on p. 77).
Weinen, Multivariable Datenanalyse zur Sortenklassifizierung von (1986). "Multivariate data analysis as a discriminating method of the origin of wines". In: *Vitis* 25, pp. 189–201 (cit. on p. 93).
Witten, Daniela M and Robert Tibshirani (2010). "A framework for feature selection in clustering". In: *Journal of the American Statistical Association* 105.490, pp. 713–726 (cit. on p. 12).
Wu, C. F. Jeff (Mar. 1983). "On the Convergence Properties of the EM Algorithm". In: *Ann. Statist.* 11.1, pp. 95–103 (cit. on p. 19).
Wyse, Jason, Nial Friel, and Pierre Latouche (2017). "Inferring structure in bipartite networks using the latent blockmodel and exact ICL". In: *Network Science* 5.1, pp. 45–69 (cit. on pp. 48, 100, 114).
Xie, Pengtao and Eric P Xing (2013). "Integrating Document Clustering and Topic Modeling". In: *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence* (cit. on p. 52).
Xu, Wei, Xin Liu, and Yihong Gong (2003). "Document clustering based on non-negative matrix factorization". In: *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. ACM, pp. 267–273 (cit. on pp. 37, 61).
Yamazaki, Keisuke and Sumio Watanabe (2003). "Singularities in mixture models and upper bounds of stochastic complexity". In: *Neural networks* 16.7, pp. 1029–1038 (cit. on p. 45).
Yang, Yun, Debdeep Pati, and Anirban Bhattacharya (2020). "*alpha*-variational inference with statistical guarantees". In: *Annals of Statistics* 48.2, pp. 886–905 (cit. on p. 20).
Ye, Jieping (2005). "Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems". In: *Journal of Machine Learning Research* 6.Apr, pp. 483–502 (cit. on pp. 73, 82).
Yoshida, Ryo, Tomoyuki Higuchi, and Seiya Imoto (2004). "A mixed factors model for dimension reduction and extraction of a group structure in gene expression data". In: *Proceedings. 2004 IEEE Computational Systems Bioinformatics Conference, 2004. CSB 2004.* IEEE, pp. 161–172 (cit. on p. 39).
Yu, Shipeng (2006). "Advanced Probabilistic Models for Clustering and Projection". PhD thesis (cit. on p. 29).
Yu, Shipeng et al. (2005). "A probabilistic clustering-projection model for discrete data". In: *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, pp. 417–428 (cit. on pp. 52, 53, 55).
Zhao, Yunpeng, Elizaveta Levina, and Ji Zhu (2012). "Consistency of community detection in networks under degree-corrected stochastic block models". In: *The Annals of Statistics* 40.4, pp. 2266–2292 (cit. on p. 114).

Zhong, Shi and Joydeep Ghosh (2003). "A unified framework for model-based clustering". In: *Journal of machine learning research* 4.Nov, pp. 1001–1037 (cit. on p. 101).

Zhou, Tao et al. (2007). "Bipartite network projection and personal recommendation". In: *Physical review E* 76.4, p. 046115 (cit. on p. 8).

Zhu, Yaojia, Xiaoran Yan, and Cristopher Moore (2014). "Oriented and degree-generated block models: generating and inferring communities with inhomogeneous degree distributions". In: *Journal of Complex Networks* 2.1, 1–18 (cit. on p. 113).

Zou, Hui and Trevor Hastie (2005). "Regularization and variable selection via the elastic net". In: *Journal of the royal statistical society: series B (statistical methodology)* 67.2, pp. 301–320 (cit. on p. 128).

Zou, Hui, Trevor Hastie, and Robert Tibshirani (2006). "Sparse principal component analysis". In: *Journal of computational and graphical statistics* 15.2, pp. 265–286 (cit. on p. 31).

Zreik, Rawya, Pierre Latouche, and Charles Bouveyron (2016). "The dynamic random subgraph model for the clustering of evolving networks". In: *Computational Statistics* (cit. on pp. 27, 100).

Zwiener, Isabella, Barbara Frisch, and Harald Binder (2014). "Transforming RNA-Seq data to improve the performance of prognostic gene signatures". In: *PloS one* 9.1, e85150 (cit. on p. 7).