Chapter 1

Cluster Analysis

1.1. Introduction

Cluster analysis or *clustering*, which is an important tool in a variety of scientific areas including pattern recognition, information retrieval, microarrays and data mining, is a family of exploratory data analysis methods that can be used to discover structures in data. These methods seek to obtain a reduced representation of the initial data and, along with principal component analysis, factor analysis and multidimensional scaling, are one form of data reduction. The aim of cluster analysis is the organization of the set into *homogeneous classes* or *natural classes*, in a way which ensures that objects within a class are similar to one another. For example, in statistics, cluster analysis can identify several populations within a heterogeneous initial population, thereby facilitating a subsequent statistical study; in natural science, the clustering of animal and plant species, first proposed by Linnaeus (an 18th-Century Swedish naturalist), is a famous example of cluster analysis; in the study of social networks, clustering may be used to recognize communities within large groups of people and, at a more

Co-Clustering: Models, Algorithms and Applications Gérard Govaert and Mohamed Nadif

general level, simply naming of objects can be seen as a form of clustering.

The attempt to formally define clustering, as a basis for an automated process, raises a number of questions. How can we define the objects (elements, cases, individuals or observations) to be classified? What is a cluster? How are clusters structured? How can different partitions be compared? Most often, the first step consists of defining the notion of proximity, a measure of closeness that can be similarity, dissimilarity or distance, among the objects to be clustered: two objects are close when their dissimilarity or distance is small or their similarity is large. Sometimes these proximities are the form in which the data naturally occur. In most clustering problems, however, each of the objects under investigation will be described by a set of *variables* or *attributes*, and the first step, possibly the most important, in clustering is to define these proximities. Then, a numerical function, usually known as a *criterion*, measuring the homogeneity of the clusters must be defined.

A classical example of a criterion used when the objects $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are described by d continuous variables is the *within-group sum of squares*, also called *within-group inertia*. In this situation, each individual being characterized by a vector $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{id})$, the data take the form of a matrix $\mathbf{x}$ of dimension (n, d) defined by values x_{ij}, where i belongs to a set I of n observations and j belongs to a set J of d continuous variables. The within-group sum of squares can therefore be written as

$$I_W(\mathbf{z}) = \frac{1}{n} \sum_{i,k} z_{ik} \mathbf{d}^2(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_k) = \frac{1}{n} \sum_{i,k} z_{ik} ||\boldsymbol{x}_i - \overline{\boldsymbol{x}}_k||^2, \qquad [1.1]$$

where $\overline{\boldsymbol{x}}_k$ is the mean vector of the kth cluster and $\mathbf{d}$ is the Euclidean distance. Using the within-group covariance matrix

$\mathbf{S}_W = \frac{1}{n}\sum_k z_{.k}\mathbf{S}_k$, where $z_{.k}$ is the size of the kth cluster and $\mathbf{S}_k$ is the covariance matrix of the kth cluster

$$\mathbf{S}_k = \frac{1}{z_{.k}}\sum_i z_{ik}(\boldsymbol{x}_i - \overline{\boldsymbol{x}}_k)(\boldsymbol{x}_i - \overline{\boldsymbol{x}}_k)^t,$$

this criterion can also be written as $I_W(\mathbf{z}) = \mathrm{trace}(\mathbf{S}_W)$. The closer the within-group sum-of-squares criterion is to 0, the more homogeneous the partition will be. In particular, this criterion will be equal to 0 for a partition where each object is a cluster.

The problem can then appear very simple: from the finite set of partitions, select the partition that optimizes the numerical criterion. Unfortunately, the number of partitions is too large for them to be enumerated in a realistic time frame, because of combinatorial complexity. Generally, heuristics are used that, rather than giving the best solution, give a "good" solution close to the optimal solution and lead to local optimization. For instance, the two most commonly used clustering algorithms, namely the k-means algorithm for obtaining partitions and Ward's hierarchical clustering method for obtaining hierarchies, use the within-group sum of squares criterion $\mathrm{trace}(\mathbf{S}_W)$ derived from the within-group covariance matrix $\mathbf{S}_W$, and used the Euclidean metric as a measure of proximity.

In recent years, what used to be an algorithmic, heuristic and geometric focus has tended to give way to a more statistical approach using probabilistic clustering models to formalize the intuitive notion of a natural class [BOC 89]. This approach allows precise analysis and can provide a statistical interpretation of certain metrical criteria whose different variants are not always clear (such as the within-group sum of squares criterion $\mathrm{trace}(\mathbf{S}_W)$), as well as yielding new variants corresponding to precise hypotheses. It also represents a formal framework for tackling difficult

problems such as determining the number of classes or validating the obtained clustering structure. We should bear in mind that in many cases the set to be segmented is merely a sample drawn from a much larger population, and that the conclusions drawn from clustering the sample are to be extrapolated to the entire population. Here, clustering becomes meaningless in the absence of a probabilistic model justifying this extrapolation. All probabilistic approaches to clustering first assume that the data represent a random sample $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ from among a population, and then use an analysis of the probability distribution of this population to define a clustering. A number of different probabilistic clustering methods have been proposed, but the most traditional approach is the use of mixture models, which forms the main subject of this chapter.

Section 1.2 presents a brief review of the main approaches to clustering. Sections 1.3 and 1.4 deal with, respectively, the probability mixture models and the EM algorithm, the standard tool for estimating the parameters of such models. Section 1.5 describes how clustering may be carried out using a mixture model. The four subsequent sections describe several classical situations including Gaussian mixture models for continuous variables, and the latent class model for binary variables, categorical variables and contingency table, and in section 1.10, we study the implementation of these different methods.

1.2. Miscellaneous clustering methods

1.2.1. *Hierarchical approach*

In this section, it will generally be assumed that all the relevant relationships within the set to be classified are summarized by a dissimilarity $\mathbf{d}$. The aim of hierarchical methods is to construct a sequence of partitions of a set

varying from partitions of singletons to the whole set. There are two principal approaches. The divisive approach starts with just one cluster containing all the objects. In each successive iteration, clusters are split into two or more further clusters, usually until every object is alone within a cluster. Note that other stop conditions can be used, and the division into clusters is governed by whether or not a particular property is satisfied. For example, in taxonomy, animals may be separated into vertebrates and invertebrates. The agglomerative approach, in contrast to the divisive approach, starts out from a set of n clusters, with each object forming a singleton cluster. Then, in each successive iteration, the closest clusters are merged until just one cluster remains. Using a dissimilarity D among groups, the closest clusters are the two clusters that are the *closest* with respect to D. According to the definition of D, several agglomerative criteria exist, but the most commonly used are the single linkage or nearest-neighbor criterion [SIB 73], the complete linkage or furthest-neighbor criterion [SOR 48] and the average linkage criterion [SOK 58]. When the objects are described by continuous variables, it is also possible to use Ward's method.

1.2.2. *The k-means algorithm*

This section deals with the k-means algorithm, which is the classical method in partitional clustering when the data are a set of objects $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ described by d continuous variables. The objective of partitional (or non-hierarchical clustering) is to define the partition of a set of objects into clusters, that the objects in a cluster are more "similar" to each other than to objects in other clusters. Starting from g initial cluster centers, the k-means algorithm involves the two following steps up to the convergence: assign each object in Ω to the nearest cluster center; use the centroids of the different clusters as the new cluster centers. It can easily be

shown that this algorithm yields a stationary sequence of partition decreasing the within-group sum-of-squares criterion.

The term k-means actually covers a whole family of methods, and the algorithm previously described is only one example. Bock [BOC 07] has carried out an interesting survey of some historical issues related to the k-means algorithm. We can cite Dalenius [DAL 50, DAL 51], Lloyd's algorithm [LLO 57] in the context of scalar quantization in the one-dimensional case and Steinhaus [STE 56] for data in $\mathbb{R}^d$ in the multidimensional case. Different strategies have been used: first, we have batch algorithms that process all the objects of the sample at each iteration, and incremental algorithms that process only one object at each iteration. Second, we have on-line or off-line training. In on-line training, each object is discarded after it has been processed (on-line training is always incremental). In off-line training, all the data are stored and can be accessed repeatedly (batch algorithms are always off-line). The term *sequential* is ambiguous, referring sometimes to incremental algorithms and sometimes to on-line learning. On-line k-means variants are particularly suitable when not all the data to be classified are available at the outset. The k-means algorithm previously described is an example of a batch algorithm. This batch version, which is the one used most often, was proposed by Forgy [FOR 65], Jancey [JAN 66] or Linde *et al.* [LIN 80] in vector quantization. The algorithm of MacQueen [MAC 67], who was the first to use the name "k-means", and the *neural gas* algorithm [MAR 91b] are examples of on-line k-means.

If the aim is to find the partition minimizing the within-group sum-of-squares criterion, the k-means algorithm does not necessarily provide the best result, but simply a sequence of partitions, the value of whose criterion

will decrease, thus giving a local optimum. Since, in practice, convergence is reached very quickly (often in fewer than 10 iterations even with a large data set), the user can run k-means several times, with different random initializations and retain the best partition, i.e. that optimizes the criterion.

If the number of clusters is not known, several solutions are possible to solve this very difficult problem. For example, the best partition is sought for several numbers of classes and the number of classes are selected by choosing an elbow on the scree plot. It is also possible to add additional constraints relating, for example, to the number of objects by cluster or to the volume of a cluster (see, for instance, the Isodata algorithm [BAL 67]). Finally, there are other approaches using statistical methods, such as hypothesis tests or selection model criteria. This last approach, apparently the most interesting, consists of penalizing the criterion by a function depending on the number of classes, making the criterion "independent" of this number of classes. For instance, minimizing the within-group sum of squares can be viewed as maximizing a likelihood (see section 1.6), and selection model criteria such as the Akaike information criterion (AIC) or the Bayesian information criterion (BIC) can be used to determine the number of clusters.

Finally, it can be interesting to use k-means and Ward's method simultaneously: the two methods are similar in that they both attempt to minimize the within-group sum-of-squares criterion. This leads us to propose strategies using the two approaches, such as, for example, the hybrid method proposed by Wong [WON 82].

1.2.3. *Other approaches*

The *dynamic cluster method* proposed by Diday [DID 71, DID 76] is a generalization of the k-means

algorithm based on the quite powerful idea that the cluster centers are not necessarily centroids of clusters in $\mathbb{R}^d$ and to replace them by centers that may take a variety of forms, depending on the problem to be solved. The k-medoids algorithm ([KAU 87]) is a typical dynamic cluster method where the cluster centers are objects of the set to cluster. Different versions have been proposed: partition around medoids (PAM) ([KAU 90]) and CLARANS ([NG 94]), which are more efficient for large volumes of data. This method is particularly well adapted when the data are given as a dissimilarity matrix d. The k-modes algorithm [HUA 97, NAD 93] for categorical data is another example in which the centers are vectors of categories. A final example (*adaptive distance*, [DID 74, DID 77]), in which each center is a pair of point and distance, determines partition and distance simultaneously allowing the shapes of clusters to be taken into account.

Fuzzy clustering [RUS 69], developed to handle the notion of overlapped clusters, generalizes the classical approach in clustering by assuming that each element $\boldsymbol{x}_i$ can belong to more than one cluster with different levels. A fuzzy partition can therefore be represented by a fuzzy classification matrix $\mathbf{c} = \{c_{ik}\}$ satisfying the following conditions: $\forall i, k,\ c_{ik} \in [0, 1]$, $\forall k,\ \sum_i c_{ik} > 0$ and $\forall i,\ \sum_k c_{ik} = 1$. Bezdeck [BEZ 81] proposed the *fuzzy k-means* algorithm, which can be viewed as a fuzzy version of k-means. The parameter estimation of a mixture model (see section 1.3) can also be viewed as a fuzzy clustering, and the associated EM algorithm is a more statistically formalized method that includes the notion of partial membership in classes. It has better convergence properties and is in general preferred to fuzzy k-means.

The *self-organizing map* (SOM) or *Kohonen map* [KOH 82] was first inspired by the adaptive formation of topology-conserving neural projection in the brain. Its aim is

to generate a mapping of a set of high-dimensional input signals onto a one- or two-dimensional array of formal neurons. Each neuron becomes representative of some input signals, such that the topological relationship among input signals in the input space is reflected, as faithfully as possible, in the arrangement of the corresponding neurons in the array (also called output space). When using this method for clustering, it is possible either to match each neuron with a unique cluster or to match many neurons to one cluster. In the latter case, the Kohonen algorithm produces a reduced representation of the original data set, and clustering algorithms may operate on this new representation. In the SOM literature, we refer to the clusters by the nodes or neurons, each of which has a weight in $\mathbb{R}^d$. The weights refer to the cluster means. The principal advantage of SOM is that it preserves the topology clustering. Generally, the neurons are arranged as a one- or two-dimensional rectangular grid preserving relations among the objects, also referred to as units. SOM is therefore a useful tool for visualizing clusters and evaluating their proximity in a reduced space. The Kohonen map can be viewed as an extension of the on-line k-means algorithm and, like k-means, requires the number of clusters (nodes of the grid) to be fixed and initial values to be selected. Different strategies can be used but initialization using principal component analysis (PCA) would appear to be an attractive and interesting approach.

High-dimensional data present a particular challenge to clustering algorithms. This is because of the so-called curse of dimensionality that leads to the sparsity of the data: in high-dimensional space, all pairs of points tend to be almost equidistant from one another. As a result, it is often unrealistic to define distance-based clusters in a meaningful way. Usually, clusters cannot be found in the original feature space because several features may be irrelevant for clustering, owing to correlation or redundancy. However,

clusters are usually embedded in lower dimensional subspaces, and different sets of features may be relevant for different sets of objects. Thus, objects can often be clustered differently in subspaces varying from the original feature space. Different approaches have been proposed. *Subspace clustering* seeks to find clusters in different subspaces within a data set. An example of this kind of approach is the CLIQUE algorithm [AGR 98]. It is a density-based method that can automatically find subspaces of the highest dimensionality such that high-density clusters exist in those subspaces. *Subspace ranking* aims at identifying all subspaces of a (high-dimensional) feature space that contain interesting clustering structures. The subspaces should be ranked according to this interestingness. *Projected clustering* is a method whereby the subsets of dimensions selected are specific to the clusters themselves. We can also cite the high-dimensional data clustering (HDDC) approach of Bouveyron [BOU 07]. In this approach, a family of Gaussian mixture models designed for high-dimensional data combining the ideas of subspace clustering and parsimonious modeling are used to develop a clustering method based on the EM algorithm.

Kernel clustering methods have attracted much attention in recent years [FIL 08]. They involve transforming a low-dimensional input space into a high-dimensional kernel-deduced feature space in which patterns are more likely to be linearly separable [SCH 02]. They have certain advantages when handling nonlinear separable data sets. We can also cite *spectral clustering* techniques that make use of the spectrum of the similarity matrix of the data to perform dimensionality reduction for clustering in fewer dimensions, and more generally, graph clustering techniques.

In this section, different approaches to clustering, essentially classical and generally based on numerical

criteria, have been reviewed. Unfortunately, defining these criteria and using them successfully is not always easy. To overcome these difficulties exist other approaches, such as the mixture model approach, which is undoubtedly a very useful contribution to clustering. It offers considerable flexibility, gives a meaning to certain criteria and sometimes leads to replacing criteria with new criteria with fewer drawbacks. In addition, it provides solutions to the problem of the number of clusters. The next section describes this approach.

1.3. Model-based clustering and the mixture model

The clustering methods described in the previous two sections are mainly heuristic techniques derived from empirical methods, usually optimizing measurement criteria. Implementing these solutions entails choosing not only a metric reflecting the dissimilarity among the objects in the set to be segmented, but also a criterion deriving from this metric capable of measuring the degree of cohesion and separation among classes. A rapid perusal of the lists of metrics and criteria proposed in the clustering literature will be enough to convince most readers that these are not easy choices. A number of different probabilistic clustering methods have been proposed, but the use of finite mixture densities, which provides a sensible statistical model for the clustering process, is now widespread.

Since their first use by Newcomb in 1886 for the detection of outlier points, and then by Pearson in 1894 to identify two separate populations of crabs, finite mixtures of distributions have been employed to model a wide variety of random phenomena. These models assume that measurements are taken from a set of individuals, each of which belongs to one of a number of different classes, while any individual's particular class is unknown. We might, for instance, know

the sizes of fish in a sample, but not their sex, which is difficult to ascertain. Mixture models can thus address the heterogeneity of a population, and are especially well suited to the problem of clustering. This is an area where much research has been done. McLachlan and Peel's book [MCL 00] is a highly detailed reference for this domain that has seen considerable developments over the last few years. We will first briefly recall the model and the problems of estimating its parameters.

Finite mixture models, which assume that every class is characterized by a probability distribution, are highly flexible models that can take account of a variety of situations including heterogeneous populations and outlier elements. Because of the EM algorithm, which is particularly well suited to this kind of context, a number of mixture models have been developed in the field of statistics, and the use of mixture models in clustering has been studied by authors including Scott and Symons [SCO 71], Marriott [MAR 75], Symons [SYM 81], McLachlan [MCL 82] and McLachlan and Basford [MCL 88]. The mixture model approach is attractive for several reasons. It corresponds to our intuitive idea of a population composed of several classes, it is strongly linked to reference methods such as the k-means algorithm and it is able to handle a wide variety of special situations in a more or less natural way. It is this approach that forms the subject of this chapter.

In a finite probability mixture model, the data $\mathrm{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ are taken to constitute a sample of n independent instances of a random variable $\boldsymbol{X}$ in $\mathbb{R}^d$. The productivity density function (pdf) can be expressed as

$$f(\boldsymbol{x}_i) = \sum_k \pi_k f_k(\boldsymbol{x}_i), \quad \forall i \in I,$$

where g is the number of components, f_k are the pdf of each component and π_k are the mixture proportions ($\pi_k \in]0,1[\ \forall k$ and $\sum_k \pi_k = 1$). The principle of a mixture model is to suppose, given the proportions $\pi_1, \ldots, \pi_g$ and the distributions f_k of each class, that the data are generated according to the following mechanism:

– z: each individual is allotted to a class according to a categorical distribution with parameters $\pi_1, \ldots, \pi_g$;

– x: each $\boldsymbol{x}_i$ is assumed to arise from a random vector with pdf f_k.

In addition, it is usually assumed that the components' pdf f_k belong to a parametric family of pdf $f(., \boldsymbol{\alpha})$. The pdf of the mixture can therefore be written as

$$f(\boldsymbol{x}_i, \boldsymbol{\theta}) = \sum_k \pi_k f(\boldsymbol{x}_i; \boldsymbol{\alpha}_k), \quad \forall i \in I,$$

where $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_g, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_g)$ is the parameter of the model. For example, the pdf of a mixture model for two univariate Gaussian distributions of variance 1 in $\mathbb{R}$ is written as

$$f(x_i; \pi, \mu_1, \mu_2) = \pi \varphi(x_i; \mu_1, 1) + (1 - \pi)\varphi(x_i; \mu_2, 1),$$

where $\varphi(.; \mu, \sigma^2)$ is the pdf of the univariate Gaussian distribution of mean μ and variance σ^2. Figure 1.1 uses the pdf obtained from a mixture of three Gaussian components in $\mathbb{R}^2$ to illustrate this concept of a probability mixture.

Much effort has been devoted to the estimation of parameters for the mixture model, following the work of Pearson, whose use of the method of moments to estimate the five parameters $(\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \pi)$ of a univariate Gaussian mixture model with two components required him to solve polynomial equations of degree nine. There have been a

number of studies [MCL 88, TIT 85] and different estimation methods have been envisaged. Apart from the method of moments, we also find graphic methods, the maximum likelihood method and Bayesian approaches. In this chapter, we will restrict ourselves to examining the maximum likelihood method using the EM algorithm, which is currently the most widely used. Before examining this method in section 1.4, we first draw the reader's attention to certain difficulties which the estimation of parameters of a mixture model presents.

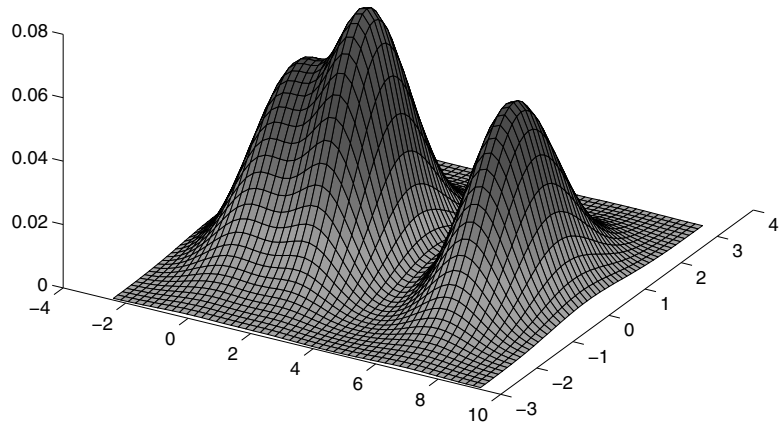

Figure 1.1. *Gaussian mixture in* $\mathbb{R}^2$

In certain situations, such as in the case of the crabs data previously described where the idea of the component has a precise physical basis, the number of components may be completely determined. Most often, however, the number of components is not known and must itself be estimated. It should be noted that if the number of components is taken to be an additional parameter, the mixture model may be seen as a semi-parametric compromise between a classical parametric estimation problem, when the number of components corresponds to a fixed constant, and a

non-parametric estimation problem, in this case via the kernel method, when the number of components is equal to the size of the sample. We assume from here onwards that number g of components is known, and later we will look at the proposed solutions for making this difficult choice.

If the problem is to be of any interest, the pdf of the mixture needs to be identifiable, which means that any two mixtures whose densities are the same must have the same parameters. A number of studies have addressed this problem and several difficulties arise. The first difficulty is due to the numbering of the classes. For example, in the case of a mixture with two components, the parameters $((\pi_1, \pi_2), (\alpha_1, \alpha_2)$ and $((\pi_2, \pi_1), (\alpha_2, \alpha_1)$, although different, obviously yield the same pdf: a mixture is consequently never identifiable. The difficulties to which this situation gives rise will depend on the estimation algorithms. In the case of the EM algorithm that we will use, it simply does not matter, however, this cannot be said of the Bayesian approach, where this situation is known as the "*switching problem*". The second, considerably more awkward, difficulty may arise from the very nature of the component pdf. It may easily be established that a mixture of uniform or binomial distributions is not identifiable. Mixtures of Gaussian, exponential and Poisson distributions, however, are identifiable.

1.4. EM algorithm

Maximizing the log-likelihood of a mixture model

$$\mathrm{L}(\boldsymbol{\theta}) = \log \left(\prod_i \sum_k \pi_k f(\boldsymbol{x}_i, \boldsymbol{\alpha}_k) \right)$$

leads to likelihood equations that usually have no analytical solution. It may, nevertheless, be shown that if the parameter

$\boldsymbol{\alpha}_k$ is a vector of real numbers α_{kr}, the solution of these likelihood equations must satisfy

$$\pi_k = \frac{1}{n}\sum_i \widetilde{z}_{ik} \quad \forall k \quad \text{and} \quad \sum_i \widetilde{z}_{ik} \frac{\partial \log f_k(\boldsymbol{x}_i, \boldsymbol{\alpha}_k)}{\partial \alpha_{kr}} = 0 \quad \forall k, r \qquad [1.2]$$

$$\text{with} \quad \widetilde{z}_{ik} = \frac{\pi_k f_k(\boldsymbol{x}_i, \boldsymbol{\alpha}_k)}{\sum_{k'} \pi_{k'} f_{k'}(\boldsymbol{x}_i, \boldsymbol{\alpha}_{k'})}. \qquad [1.3]$$

These equations suggest the following iterative algorithm: (1) start from an initial solution θ; (2) calculate the values $\widetilde{z}_{ik}$ from this parameter using equation [1.3]; (3) update the parameter θ on the basis of these values $\widetilde{z}_{ik}$ using equations [1.2]; continue from (2). If this algorithm converges, therefore, the fixed point obtained will satisfy the likelihood equations. The procedure corresponds, in fact, to the application of the Dempster *et al.* [DEM 77] EM algorithm to the mixture model. Before describing this algorithm, we will define the concept of complete data on which it relies.

1.4.1. *Complete data and complete-data likelihood*

At the outset, we consider that the observed data $\mathbf{x}$ correspond to what is merely a partial knowledge of unknown data $\mathbf{y}$ that are termed *complete data*, the two being linked by a function $\mathbf{x} = T(\mathbf{y})$. The complete data might, for instance, be of the form $\mathbf{y} = (\mathbf{x}, \mathbf{z})$, in which case $\mathbf{z}$ is known as *missing information*. This idea of complete data may either be meaningful for a model, which is the case for the mixture model, or it may be completely artificial. The likelihood $f(\mathbf{y}; \boldsymbol{\theta})$ calculated from these complete data is termed *complete-data likelihood* or, in the case of the mixture model, *classification likelihood*. Starting from the equation $f(\mathbf{y}; \boldsymbol{\theta}) = f(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}) f(\mathbf{x}; \boldsymbol{\theta})$, we obtain the equation

$$\mathrm{L}(\boldsymbol{\theta}) = \mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}) - \log f(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}) \qquad [1.4]$$

between the initial log-likelihood $\mathrm{L}(\boldsymbol{\theta})$ and the complete-data log-likelihood $\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z})$.

1.4.2. *Principle*

The EM algorithm is based on the hypothesis that maximizing the complete-data likelihood is simple. Since this likelihood cannot be calculated – y is unknown – an iterative procedure based on the conditional expectation of the log-likelihood for a value of the current parameter $\boldsymbol{\theta}'$ is used as follows: first, calculating the conditional expectation for the two members of equation [1.4], we obtain the fundamental equation of the EM algorithm

$$\mathrm{L}(\boldsymbol{\theta}) = Q(\boldsymbol{\theta}, \boldsymbol{\theta}') - H(\boldsymbol{\theta}, \boldsymbol{\theta}'),$$

where $Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = \mathbb{E}(\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z})|\mathbf{x}, \boldsymbol{\theta}')$ and $H(\boldsymbol{\theta}, \boldsymbol{\theta}') = \mathbb{E}(\log f(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta})|\mathbf{x}, \boldsymbol{\theta}')$.

Introducing the parameter $\boldsymbol{\theta}'$ allows us to define an iterative algorithm to increase the likelihood. Using Jenssen's inequality, it can be shown that for fixed $\boldsymbol{\theta}'$, the function $H(\boldsymbol{\theta}, \boldsymbol{\theta}')$ is the maximum for $\boldsymbol{\theta} = \boldsymbol{\theta}'$. The value $\boldsymbol{\theta}$ that maximizes $Q(\boldsymbol{\theta}, \boldsymbol{\theta}')$, therefore, satisfies the equation

$$\mathrm{L}(\boldsymbol{\theta}) \geqslant \mathrm{L}(\boldsymbol{\theta}'). \qquad [1.5]$$

The EM algorithm involves constructing, from an initial solution $\boldsymbol{\theta}^{(0)}$, the sequence $\boldsymbol{\theta}^{(p)}$ satisfying $\boldsymbol{\theta}^{(q+1)} = \arg\max Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)})$. Equation [1.5] shows that this sequence causes the criterion $\mathrm{L}(\boldsymbol{\theta})$ to develop.

1.4.3. *Application to mixture models*

For the mixture model, the complete data are obtained by adding the original component $\mathbf{z}_i$ to each individual member of the sample

$$\mathbf{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n) = ((\boldsymbol{x}_1, \boldsymbol{z}_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{z}_n)).$$

Coding $\boldsymbol{z}_i = (z_{i1}, \ldots, z_{ig})$ where, let us recall, z_{ik} equals 1 if i belongs to component k and 0 otherwise, we obtain the following equations

$$f(\mathbf{y}; \boldsymbol{\theta}) = \prod_i f(\boldsymbol{y}_i; \boldsymbol{\theta}) = \prod_i \sum_k \pi_k f(\boldsymbol{x}_i; \boldsymbol{\alpha}_k),$$

$$\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}) = \log(f(\mathbf{y}; \boldsymbol{\theta})) = \sum_{i,k} z_{ik} \log\left(\pi_k f(\boldsymbol{x}_i; \boldsymbol{\alpha}_k)\right),$$

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}') = \sum_{i,k} \mathbb{E}(z_{ik}|\mathbf{x}, \boldsymbol{\theta}') \log\left(\pi_k f(\boldsymbol{x}_i; \boldsymbol{\alpha}_k)\right).$$

Denoting as $\widetilde{z}_{ik}$ the probabilities of belonging $\mathbb{E}(z_{ik}|\mathbf{x}, \boldsymbol{\theta}') = P(z_{ik} = 1|\mathbf{x}, \boldsymbol{\theta}')$, the EM algorithm takes the following form:

– initialize: arbitrarily select an initial solution $\boldsymbol{\theta}$;

– repeat the following two steps until convergence:

- step E (expectation): calculate the probabilities of $\boldsymbol{x}_i$ belonging to the classes, conditionally on the current parameter

$$\widetilde{z}_{ik} = \frac{\pi_k f(\boldsymbol{x}_i, \alpha_k)}{\sum_{k'} \pi_{k'} f(\boldsymbol{x}_i, \alpha_{k'})};$$

- step M (maximization): maximize the log-likelihood conditionally on $\widetilde{z}_{ik}$; the proportions are therefore obtained simply by the equation $\pi_k = \sum_i \widetilde{z}_{ik}/n$, while the parameters $\boldsymbol{\alpha}_k$ are obtained by solving the likelihood equations that depend on the mixture model employed.

1.4.4. *Properties*

Under certain conditions of regularity, it has been established that the EM algorithm always converges to a local likelihood maximum. It shows good practical behavior, but may, nevertheless, be quite slow in some situations. This is the case, for instance, when classes are very mixed. This algorithm, proposed by Dempster *et al.* in a seminal paper [DEM 77], often simple to implement, has gained widespread popularity and given rise to a large number of studies that are thoroughly covered in McLachlan and Krishnan's book [MCL 97].

1.4.5. *EM: an alternating optimization algorithm*

Hathaway [HAT 86] has shown that the EM algorithm applied to a mixture model may be interpreted as an alternating algorithm for optimizing a fuzzy clustering criterion. We make use of this fact below when we examine the links between estimating the parameters of a mixture model and fuzzy clustering. To obtain this result, Hathaway defines the criterion

$$\mathrm{F_C}(\widetilde{\mathbf{z}}, \boldsymbol{\theta}) = \mathrm{L_C}(\boldsymbol{\theta}, \widetilde{\mathbf{z}}) + \mathrm{H}(\widetilde{\mathbf{z}}), \qquad [1.6]$$

where $\mathrm{H}(\widetilde{\mathbf{z}}) = -\sum_{i,k} \widetilde{z}_{ik} \log \widetilde{z}_{ik}$ is the entropy of the distribution $\widetilde{\mathbf{z}}$. Moreover, if we denote as $\widetilde{\mathbf{z}}_{\boldsymbol{\theta}}$ the posterior distribution $\mathbb{P}(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})$, it can be shown, using equation [1.4], that the criterion $\mathrm{F_C}$ can also be expressed as

$$\mathrm{F_C}(\widetilde{\mathbf{z}}, \boldsymbol{\theta}) = L(\boldsymbol{\theta}) - \mathrm{KL}(\widetilde{\mathbf{z}}, \widetilde{\mathbf{z}}_{\boldsymbol{\theta}}), \qquad [1.7]$$

where KL is the Kullback–Liebler divergence between two distributions.

The alternating algorithm for optimizing the criterion $\mathrm{F_C}$, therefore, becomes simple to implement:

– minimizing for fixed $\boldsymbol{\theta}$: equation [1.7] implies that $\widetilde{\mathbf{z}}$ must minimize $\mathrm{KL}(\widetilde{\mathbf{z}}, \widetilde{\mathbf{z}}_{\boldsymbol{\theta}})$ and consequently $\widetilde{\mathbf{z}} = \widetilde{\mathbf{z}}_{\boldsymbol{\theta}}$;

– minimizing for fixed $\widetilde{\mathbf{z}}$: equation [1.6] shows that $\boldsymbol{\theta}$ must maximize the expectation $\mathrm{L_C}(\boldsymbol{\theta}, \widetilde{\mathbf{z}})$.

Therefore, we are dealing with what are precisely the two steps of the EM algorithm. In addition, after each first step, we have $\mathrm{F_C}(\widetilde{\mathbf{z}}, \boldsymbol{\theta}) = \mathrm{F_C}(\widetilde{\mathbf{z}}_{\boldsymbol{\theta}}, \boldsymbol{\theta}) = \mathrm{L}(\boldsymbol{\theta})$, demonstrating that the EM algorithm increases the likelihood.

1.5. Clustering and the mixture model

1.5.1. *The two approaches*

Mixture models may be used in two different ways to obtain a partition of the initial data.

– The first, known as the *mixture approach*, estimates the parameters of the model and then determines the partition by allocating each individual to the class that maximizes the *a posteriori* probability $\widetilde{\mathbf{z}}_{ik}$ computed using these estimated parameters; this allocation is known as the maximum *a posteriori* probability (MAP) method.

– The second, the *classification approach*, was first presented by Scott and Symons [SCO 71] and developed further by Schroeder [SCH 76]; this approach involves creating a partition of the sample such that each class k is made to correspond to a sub-sample respecting the distribution $f(., \boldsymbol{\alpha}_k)$. This requires simultaneous estimation of the model parameters and the desired partition.

In this section, we describe the criterion that the latter approach optimizes, as well as the optimization algorithm usually employed in this situation. We then briefly compare the two approaches and examine links between these types of

methods and the more classical metrical approaches to clustering. We conclude the section by looking at how the mixture model may be interpreted in terms of fuzzy clustering.

1.5.2. *Classification likelihood*

Introducing the z partition in the likelihood criterion is not an obvious step, and various ideas have been proposed. Scott and Symons [SCO 71] defined the criterion

$$\mathrm{L_{CR}}(\boldsymbol{\theta}, \mathbf{z}) = \sum_k \sum_{i|z_{ik}=1} \log f(\boldsymbol{x}_i, \alpha_k)$$

in which the proportions do not appear. Symons [SYM 81], realizing that this criterion tends to yield classes of similar proportions, modified it so as to use the complete-data (or classification) log-likelihood described above

$$\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}) = \sum_k \sum_{i|z_{ik}=1} \log \pi_k f(\boldsymbol{x}_i, \alpha_k) = \sum_{i,k} z_{ik} \log \pi_k f(\boldsymbol{x}_i, \alpha_k)$$

linked to the previous criterion by the equation

$$\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}) = \mathrm{L_{CR}}(\boldsymbol{\theta}, \mathbf{z}) + \sum_k z_{.k} \log \pi_k,$$

where $z_{.k}$ is the cardinal of the class k. The quantity $\sum_k z_{.k} \log \pi_k$ is a penalty term that disappears if all the proportions are made to be identical. The criterion $\mathrm{L_{CR}}(\boldsymbol{\theta}, \mathbf{z})$ can, therefore, be seen as a variant of classification likelihood, restricted to a mixture model where all classes have the same proportion.

1.5.3. *The CEM algorithm*

When seeking to maximize classification likelihood, it is possible to use a clustering version of the EM algorithm, obtained by adding a clustering step. This yields the very general clustering algorithm known as classification EM (CEM) [CEL 92], defined as follows:

– step 0: arbitrarily select an initial solution θ;

– step E: compute $\widetilde{z}_{ik}$ as in the EM algorithm;

– step C: obtain the $\mathbf{z}$ partition by allocating each $\boldsymbol{x}_i$ to the class that maximizes $\widetilde{z}_{ik}$ (MAP); this is equivalent to modifying the $\widetilde{z}_{ik}$ by replacing them with the nearest 1 or 0 values;

– step M: maximize the likelihood depending on the z_{ik}; the estimations of the maximum likelihood among the π_k and the $\boldsymbol{\alpha}_k$ are obtained using the classes of the partition $\mathbf{z}$ as sub-samples, where the proportions are given by the formula $\pi_k = \frac{1}{n} z_{.k}$, the $\boldsymbol{\alpha}_k$ being computed according to the particular mixture model selected.

Here, we have an alternating *dynamic cluster methods* [DID 79] type optimization algorithm, where the E and the C steps correspond to the allocation step, and the M step corresponds to the representation step.

It can be shown that this algorithm is stationary and that it increases the complete-data likelihood at each iteration, given some very general assumptions.

1.5.4. *Comparison of the two approaches*

The clustering approach, which determines the parameters at each iteration using truncated mixture model samples, yields a biased and inconsistent estimation, since

the number of parameters to be estimated increases as the size of the sample increases. Different authors have studied this problem and shown that it is usually preferable to use the mixture approach.

However, when the classes are well separated and membership relatively small, the clustering approach can sometimes give better results [CEL 93, GOV 96]. Moreover, the CEM algorithm is considerably faster than the EM algorithm, and it may be necessary to use it when computation time is limited, for example in real-time operations, or for very large volumes of data.

Finally, the clustering approach has the advantage of being able to present a large number of clustering algorithms as special cases of the CEM algorithm, which allows it to incorporate them into a probabilistic clustering approach. We will see in section 1.6.3, for example, that the k-means algorithm can be seen as a simple special case of the CEM algorithm. In particular, we will show that the optimized criteria, the within-group sum-of-squares criterion for continuous data and the information criterion for qualitative data correspond to the classification likelihood of a particular mixture model. These correspondences, studied in [GOV 89] and [GOV 90b], can be formalized by the following theorem.

THEOREM 1.1.– If the clustering criterion can be expressed as

$$W(\mathbf{z}, \boldsymbol{\lambda}, D) = \sum_{i,k} z_{ik} \Delta(\boldsymbol{x}_i, \boldsymbol{\lambda}_k),$$

where $\mathbf{z}$ is a partition of the set to be segmented, $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_g)$ and $\boldsymbol{\lambda}_k$ are representatives of the class k, and Δ is a measure of the dissimilarity between an object $\boldsymbol{x}$ and the representative of a class, and if there exists a real r such that the quantity $\int r^{-\Delta(\boldsymbol{x},\boldsymbol{\lambda})} d\boldsymbol{x}$ is independent of $\boldsymbol{\lambda}$, this criterion is therefore equivalent to the classification

likelihood criterion of a mixture model with densities of the form $f(\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{s} r^{-\Delta(\boldsymbol{x},\boldsymbol{\lambda})}$, s being a positive constant.

This theorem may be used equally well for continuous data as for discrete (binary or qualitative) data – either a Lebesgue measure or a discrete measure will be used accordingly. This theorem is important insofar as a great many clustering criteria can be put into this very general form; for example, it is the case for the intraclass inertia criterion, whose class representative is its center of gravity and where the distance D is the square of the Euclidean distance. It can also help us to fix the fields of application of these criteria and to suggest others.

1.5.5. *Fuzzy clustering*

In fuzzy clustering, it is no longer the case that an object either belongs or does not belong to a particular class. Instead, there are degrees of belonging. Formally, fuzzy clustering is characterized by a matrix $\mathbf{c}$ with terms c_{ik} satisfying $c_{ik} \in [0, 1]$ and $\sum_k c_{ik} = 1$. Bezdek's "fuzzy k-means" [BEZ 81], one of the most commonly encountered, involves minimizing the criterion

$$\mathrm{W}(\mathbf{c}) = \sum_{i,k} c_{ik}^{\gamma} \mathbf{d}^2(\boldsymbol{x}_i, \mathbf{g}_k),$$

where $\gamma > 1$ is a coefficient for adjusting the degree of fuzziness, $\mathbf{g}_k$ is the center of the class and $\mathbf{d}$ is the Euclidean distance. It is required that γ be different from 1, otherwise the function W is minimal for values of $c_{ik} = 0$ or 1 and thus we have the usual within-group sum-of-squares criterion. The values usually recommended are between 1 and 2. Minimizing this criterion is achieved using an algorithm that alternates between the two following steps:

1) compute the centers: $\mathbf{g}_k = \frac{\sum_i c_{ik}^{\gamma} \boldsymbol{x}_i}{\sum_i c_{ik}}$;

2) compute the fuzzy partition: $c_{ik} = \frac{C_i}{\|\boldsymbol{x}_i - \mathbf{g}_k\|^{\frac{2}{\gamma-1}}}$ with $C_i = \sum_{k'} \frac{1}{\|\boldsymbol{x}_i - \mathbf{g}_{k'}\|^{\frac{2}{\gamma-1}}}$.

Validating this kind of approach and, in particular, choosing the coefficient γ can be tricky. Therefore, estimating the parameters of a mixture model is an alternative, more natural, way of addressing this problem. The estimation of the *a posteriori* probabilities $\widetilde{z}_{ik}$ of objects belonging to each class directly provides a fuzzy clustering, and the EM algorithm, applied to the mixture model, may be seen as a fuzzy clustering algorithm.

As mentioned above, Hathaway [HAT 86] went even further and showed that seeking to obtain a fuzzy partition and the parameter θ using an optimization alternated with a fuzzy clustering criterion leads precisely to the two steps of the EM algorithm, which can therefore be considered as a fuzzy clustering algorithm. One may obtain the same result simply by applying the results from section 1.4.5 to the mixture model. Given that here the probability distribution $\widetilde{\mathbf{z}}$ is defined by the vector (c_{ik}) and that we simply have $\mathbb{E}_{\mathbf{c}}(\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z})) = \mathrm{L_C}(\theta, \mathbf{c})$, we show that the EM algorithm alternately maximizes the criterion

$$\mathrm{W}(\mathbf{c}, \boldsymbol{\theta}) = \mathrm{L_C}(\boldsymbol{\theta}, \mathbf{c}) + \mathrm{H}(\mathbf{c}),$$

where $\mathrm{L_C}$ is the complete-data log-likelihood function where the partition $\mathbf{z}$ has been replaced by the fuzzy partition $\mathbf{c}$

$$\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{c}) = \sum_{i,k} c_{ik} \log\left(\pi_k f_k(\boldsymbol{x}_i; \boldsymbol{\alpha})\right)$$

and H is the entropy function

$$\mathrm{H}(\mathbf{c}) = -\sum_{i,k} c_{ik} \log c_{ik}.$$

It is easy to show that if the entropy term of the criterion W is removed, then, "hard" partitions are obtained at each step. The resulting algorithm is simply the CEM algorithm: the difference between the EM and CEM algorithms is the presence of the entropy term. If, when EM converges, the components are highly separated, the fuzzy partition $\mathbf{z}(\boldsymbol{\theta})$ is close to a partition and we have

$$H(\mathbf{z}(\boldsymbol{\theta})) \approx 0$$

and

$$L(\boldsymbol{\theta}) = W(\mathbf{z}(\boldsymbol{\theta}), \boldsymbol{\theta}) = \mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}(\boldsymbol{\theta})) + H(\mathbf{z}(\boldsymbol{\theta})) \approx \mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}(\boldsymbol{\theta})).$$

1.6. Gaussian mixture model

We will now examine what happens to this approach when each class is modeled by a Gaussian distribution, which is a classical solution for continuous data.

1.6.1. *The model*

The pdf of the mixture can be written as $f(\boldsymbol{x}_i; \boldsymbol{\theta}) = \sum_k \pi_k \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, where φ is the pdf of the Gaussian multivariate distribution

$$\varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}_k|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^t \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)\}$$

and $\boldsymbol{\theta}$ is the vector $(\pi_1, \ldots, \pi_g, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_g)$ formed by the proportions π_k and the parameters $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$, which

are, respectively, the mean vector and the covariance matrix of class k.

When the sample size is small, or when the dimension of the space is large, the number of parameters must be reduced so as to obtain more parsimonious models. To this end, the spectral decomposition of the matrices [BAN 93, CEL 95] may be used, allowing the covariance matrices to be parameterized uniquely as $\mathbf{\Sigma}_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^t$, where the diagonal matrix $\mathbf{A}_k$ with determinant 1 and decreasing values defines the *shape* of the class, the orthogonal matrix $\mathbf{D}_k$ defines the *direction* of the class and the positive real number λ_k represents the *volume* of the class. Thus, the mixture model is parameterized by the centers $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_g$, the proportions $\pi_1, \ldots, \pi_g$, the volumes $\lambda_1, \ldots, \lambda_g$, the shapes $\mathbf{A}_1, \ldots, \mathbf{A}_g$ and the directions $\mathbf{D}_1, \ldots, \mathbf{D}_g$ of each class.

For example, when the data are in a plane, $\mathbf{D}$ is a rotation matrix defined by an angle α and $\mathbf{A}$ is a diagonal matrix with diagonal terms a and $1/a$. Figure 1.2 shows the equidensity ellipse of this distribution depending on the values α, λ and a.

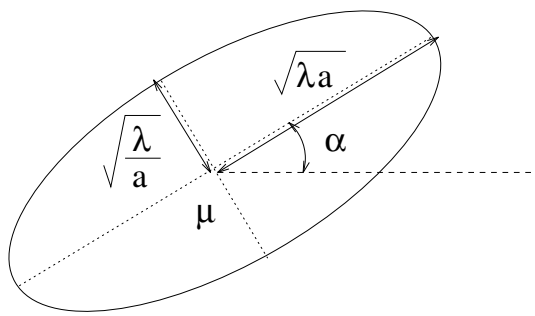

Figure 1.2. *Parameterization of a Gaussian class in the plane*

Using this parameterization, it becomes possible to propose solutions that can be seen as a middle way between, on the one hand, restrictive hypotheses (covariance matrices

proportional to the identity matrix, or covariance matrices identical for all classes) and, on the other hand, very general constraint-free hypotheses [BAN 93, CEL 95].

This parameterization also highlights two distinct notions that are often conflated under the rather vague heading of *size*: these are, first, the proportion of individuals present within a class and, second, the volume that a class occupies in space. It is quite possible for a class to have a small volume and a high proportion or, alternatively, a large volume but a low proportion.

We will now look at what happens to the CEM algorithm and the classification likelihood criterion in the case of the Gaussian mixture model. It should be noted that a similar approach could be applied to the EM algorithm.

1.6.2. *CEM algorithm*

1.6.2.1. *Clustering step*

Each $\boldsymbol{x}_i$ is allocated to the class that maximizes the probability of membership $\widetilde{z}_{ik} = \pi_k \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) / (\sum_{k'} \pi_{k'} \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_{k'}, \boldsymbol{\Sigma}_{k'}))$, that is to say $\pi_k \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ or, equivalently, the class that minimizes $-\log(\pi_k \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k))$, which can be written as

$$\mathrm{d}^2_{\boldsymbol{\Sigma}_k^{-1}}(\boldsymbol{x}_i, \boldsymbol{\mu}_k) + \log |\boldsymbol{\Sigma}_k| - 2 \log \pi_k, \qquad [1.8]$$

where $\mathrm{d}^2_{\boldsymbol{\Sigma}_k^{-1}}(\boldsymbol{x}_i, \boldsymbol{\mu}_k)$ is the quadratic distance $(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^t \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)$.

1.6.2.2. *Step M*

Here, for a given partition $\mathbf{z}$, we have to determine the parameter $\boldsymbol{\theta}$ that maximizes $\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z})$, which is equal (ignoring one constant) to

$$-\frac{1}{2}\sum_k \left(\sum_i z_{ik}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^t \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k) + z_{.k}\log|\boldsymbol{\Sigma}_k| - 2z_{.k}\log\pi_k \right).$$

The parameter $\boldsymbol{\mu}_k$ is thus necessarily the center of gravity $\overline{\boldsymbol{x}}_k = \frac{1}{z_{.k}}\sum_i z_{ik}\boldsymbol{x}_i$ and the proportions, if they are not constrained, satisfy $\pi_k = z_{.k}/n$. The parameters $\boldsymbol{\Sigma}_k$ must then minimize the function

$$F(\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_g) = \sum_k z_{.k}\left(\mathrm{trace}(\mathbf{S}_k\boldsymbol{\Sigma}_k^{-1}) + \log|\boldsymbol{\Sigma}_k|\right), \qquad [1.9]$$

where

$$\mathbf{S}_k = \frac{1}{z_{.k}}\sum_i z_{ik}(\boldsymbol{x}_i - \overline{\boldsymbol{x}}_k)(\boldsymbol{x}_i - \overline{\boldsymbol{x}}_k)^t$$

is the covariance matrix of the class k. We now examine three particular situations.

1.6.3. *Spherical form, identical proportions and volumes*

We now look at the most straightforward situation where all classes have a Gaussian spherical distribution with the same volume and the same proportion. The covariance matrices are written as $\boldsymbol{\Sigma}_k = \lambda\mathbf{D}_k\mathbf{I}_d\mathbf{D}_k^t = \lambda\mathbf{I}_d \ \forall k$, and formula [1.8] shows that individuals can be allotted to the different classes simply by using the usual Euclidean distance $\mathbf{d}^2(\boldsymbol{x}_i, \boldsymbol{\mu}_k)$. Function F, therefore, becomes

$$F(\lambda) = \frac{1}{\lambda}\sum_k z_{.k}\,\mathrm{trace}(\mathbf{S}_k) + nd\log\lambda = \frac{1}{\lambda}n\,\mathrm{trace}(\mathbf{S}_W) + nd\log\lambda,$$

where $\mathbf{S}_W = \frac{1}{n}\sum_k z_{.k}\mathbf{S}_k$ is the within-group covariance matrix, thus giving us $\lambda = \frac{\text{trace}(\mathbf{S}_W)}{d}$. The classification likelihood is written as

$$\mathrm{L_C}(\boldsymbol{\theta}, \mathbf{z}) = -\frac{nd}{2}\log \text{trace}(\mathbf{S}_W) + cst.$$

Maximizing the classification likelihood is therefore equivalent to minimizing the within-group sum-of-squares criterion $\text{trace}(\mathbf{S}_W)$. Moreover, the CEM algorithm is simply the k-means algorithm. This means that to use the within-group sum-of-squares criterion is to assume that classes are spherical and have the same proportion and the same volume.

1.6.4. *Spherical form, identical proportions but differing volumes*

We now take the model described above and modify it slightly to include classes with different volumes. The covariance matrices are now written $\boldsymbol{\Sigma}_k = \lambda_k \mathbf{I}_p$, and formula [1.8] shows that individuals are allotted to classes according to the distance

$$\frac{1}{\lambda_k} D^2(\boldsymbol{x}_i, \boldsymbol{\mu}_k) + d\log\lambda_k.$$

The distance from a point to the center of a class has been modified by an amount that depends on the volume of the class. This modification has important repercussions; for example, the regions of separation, which in the previous case were hyperplanes, become hyperspheres. It may be shown that the minimized criterion can be written as

$$\sum_k \log \text{trace}(\mathbf{S}_k).$$

With this model, we can very easily recognize situations such as the situation shown in Figure 1.3. Here, the two classes have been simulated with two spherical Gaussian distributions which have the same proportions but widely differing volumes. The result obtained using the classical intraclass inertia criterion corresponds to a separation of the population by a straight line and therefore bears no relation at all to the simulated partition. With the variable-volume model, the obtained partition, shown by the circle, is very close to the initial clustering.

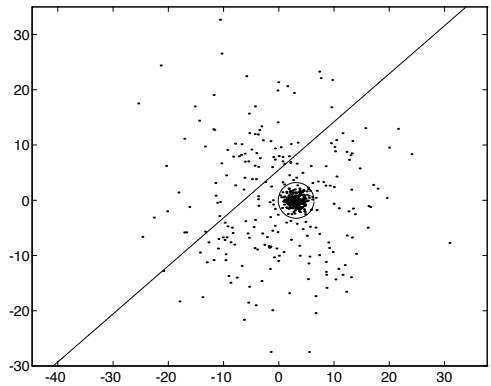

Figure 1.3. *Example of classes with different volumes*

It will be noted that without the help of the mixture model, it would have been difficult, on the basis of a simple metrical interpretation, to come up with the distance and the criterion used in this approach.

1.6.5. *Identical covariance matrices and proportions*

Our final example is where all classes have the same form and the same proportion. The covariance matrix of each class can thus be written as $\mathbf{\Sigma}_k = \mathbf{\Sigma}$. It can be shown that individuals are now allotted to classes on the basis of the distance $\mathrm{d}^2_{\mathbf{\Sigma}^{-1}}(\boldsymbol{x}_i, \boldsymbol{\mu}_k)$ and that the criterion to be minimized

may be written as $|\mathrm{S}_W|$, which serves to justify the use of this criterion, sometimes proposed in a metrical context [FRI 67], without reference to the Gaussian model.

1.7. Binary data

We now turn, still within a broad discussion of clustering methods based on probability distribution mixture models, to the clustering of sets of individuals measured using binary variables.

1.7.1. *Binary mixture model*

As the Gaussian model is often chosen to model each component of the mixture when variables are continuous, the log-linear model [AGR 90, BOC 86] is a natural choice when variables are binary. The complete or saturated log-linear model, where each class has a multinomial distribution with 2^d values, is not really applicable in the case of a mixture. Instead, we use a log-linear model with sufficient constraints. The simplest example is the independence model that assumes that, conditionally on membership of a class, the binary variables are independent. From this, we obtain the latent class model [GOO 74, LAZ 68], which we will now examine.

In the binary case, each x_{ij} has a Bernoulli distribution whose probability distribution takes the form

$$f(x_{ij};\alpha_{kj}) = (\alpha_{kj})^{x_{ij}}(1-\alpha_{kj})^{1-x_{ij}} \text{ where } \alpha_{kj} = P(x_{ij}=1|k).$$

Therefore, the distribution of the class k is

$$f_k(\boldsymbol{x}_i;\boldsymbol{\alpha}_k) = \prod_j (\alpha_{kj})^{x_{ij}}(1-\alpha_{kj})^{1-x_{ij}} \text{ where } \boldsymbol{\alpha}_k = (\alpha_{k1},\ldots,\alpha_{kp}),$$

and the mixture model chosen considers that the data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ constitute a sample of independent instances from a random $\{0,1\}^d$ probability vector

$$f(\boldsymbol{x}_i; \boldsymbol{\theta}) = \sum_k \pi_k f(\boldsymbol{x}_i; \theta_k) = \sum_k \pi_k \prod_j (\alpha_{kj})^{x_{ij}} (1 - \alpha_{kj})^{1 - x_{ij}},$$

where the parameter $\boldsymbol{\theta}$ is constituted by the proportions $\pi_1, \ldots, \pi_g$ and by the parameter vector $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_g)$ of each component.

The problem that arises is how to estimate these parameters, and possibly the origin class as well. As in the case of the Gaussian model, the estimation may be obtained using the EM or the CEM algorithms described above. The only differences concern the computation of the parameters α_{kj} that becomes $\alpha_{kj} = \frac{\sum_i \widetilde{z}_{ik} x_i^j}{\sum_i \widetilde{z}_{ik}}$ for the first, and $\alpha_{kj} = \frac{\sum_i z_{ik} x_i^j}{z_{.k}} =$ % of 1 for the second. Intensive comparisons between the two algorithms EM and CEM were performed by Govaert and Nadif [GOV 96].

1.7.2. *Parsimonious model*

The number of parameters of this latent class model is equal to $(g - 1) + g * d$, where g, it will be recalled, is the number of classes and d is the number of binary variables. In the case of the complete log-linear model, however, the number of parameters is equal to 2^d. For example, when $g = 5$ and $d = 10$, the number of parameters for the two models is, respectively, 54 and 1,024. Given that one of the identifiability conditions of the model is that the number of states is greater than the number of parameters, there is a clear interest in being able to propose even more

parsimonious models. To this end, the model may be restated as follows

$$f(\boldsymbol{x};\boldsymbol{\theta}) = \sum_k \pi_k \prod_j (\varepsilon_{kj})^{|x_{ij}-a_{kj}|}(1-\varepsilon_{kj})^{1-|x_{ij}-a_{kj}|},$$

where $\begin{cases} a_{kj}=0,\ \varepsilon_{kj}=\alpha_{kj} & \text{if } \alpha_{kj}<0.5 \\ a_{kj}=1,\ \varepsilon_{kj}=1-\alpha_{kj} & \text{otherwise.} \end{cases}$

The parameter $\boldsymbol{\alpha}_k$ is, thus, replaced by the two following parameters:

– a binary vector $\mathbf{a}_k$ representing the center of the class and which is the most frequent binary value for each variable;

– a vector $\boldsymbol{\varepsilon}_k$ belonging to the set $]0,1/2[^d$ that defines the dispersion of the component, and represents the probability of any particular variable's having a value different from that of the center.

We are, thus, led to the parameters used by Aitchinson and Aitken [AIT 76] for discrimination with non-parametric estimation via the kernel method. Starting from this formulation, we arrive at parsimonious situations by stipulating certain constraints: the $[\varepsilon]$ model is defined by stipulating that the dispersion should not depend either on the component or on the variable, the $[\varepsilon_k]$ model by stipulating that it should depend only on the component, and the $[\varepsilon^j]$ model by stipulating that it should depend only on the variable.

For example, in the simplest case, the $[\varepsilon]$ model, given identical proportions ($\pi_k = 1/g$), the clustering approach results in the complete-data log-likelihood being maximized

$$\mathrm{L_C}(\boldsymbol{\theta},\mathbf{z}) = \log\frac{\varepsilon}{1-\varepsilon}\sum_{i,k} z_{ik}\mathbf{d}(\boldsymbol{x}_i,\mathbf{a}_k) + nd\log(1-\varepsilon),$$

which is to say that the criterion

$$\mathrm{W}(\mathbf{z}, \boldsymbol{\theta}) = \sum_{i,k} z_{ik} \mathbf{d}(\boldsymbol{x}_i, \mathbf{a}_k) \quad \text{where} \quad \mathbf{d}(\boldsymbol{x}_i, \mathbf{a}_k) = \sum_j |x_{ij} - a_{kj}|$$

is minimized.

Step E of the CEM algorithm, therefore, consists simply of allotting each individual to the class k that minimizes $\mathbf{d}(\boldsymbol{x}_i, \mathbf{a}_k)$. At step M, the parameters a_{kj} for each variable j correspond to the majority binary values in each class k. A class, therefore, corresponds to a binary vector, and the criterion is easy to interpret: it is simply the number of differences among individuals and their representative in the partition $\mathbf{z}$. To use this binary clustering criterion proposed by different authors [GOW 74, GOV 90a] is therefore to assume that the data come from a particular latent class model.

1.7.3. *Examples of application*

To illustrate this approach, we have taken the Stouffer-Toby data set [STO 51], analyzed within the framework of the latent class model by Goodman [GOO 74]. Table 1.1 contains the reactions, classed as one of two possible attitudes, shown by 216 subjects placed in four different conflict situations. We compare the latent class model (here requiring nine parameters) with the log-linear model, with an interaction of order 2, and a very similar number of parameters (11). It is interesting to note that for these data the deviance drops from 7.11 in the case of the linear model to a value of 2.72 in the case of the latent class model. The parameters obtained for the latent class model are shown in Table 1.2.

$S1$	$S2$	$S3$	$S4$	Frequencies
1	1	1	1	42
1	1	1	0	23
1	1	0	1	6
1	1	0	0	25
1	0	1	1	6
1	0	1	0	24
1	0	0	1	7
1	0	0	0	38
0	1	1	1	1
0	1	1	0	4
0	1	0	1	1
0	1	0	0	6
0	0	1	1	2
0	0	1	0	9
0	0	0	1	2
0	0	0	0	20

Table 1.1. *Stouffer-Toby data*

–	p_k	a_{k1}	a_{k2}	a_{k3}	a_{k4}
1	0.279	0.993	0.940	0.927	0.769
2	0.721	0.714	0.330	0.354	0.132

Table 1.2. *Results obtained by EM for the latent class model*

1.8. Categorical variables

We now extend the results of the previous section to categorical data.

1.8.1. *Multinomial mixture model*

As for binary data, we will examine the latent class model and therefore assume that the d qualitative variables are independent, conditionally on their membership to a class. If

α_{kj}^h is the probability that the jth variable takes the modality h when an individual belongs to the class k, therefore the pdf of the mixture can be written as

$$f(\boldsymbol{x}_i;\boldsymbol{\theta}) = \sum_k \pi_k f(\boldsymbol{x}_i;\boldsymbol{\alpha}_k) = \sum_k \pi_k \prod_j \prod_{h=1}^{m_j} (\alpha_{kj}^h)^{x_{ij}^h},$$

where the parameter $\boldsymbol{\theta}$ is defined by the proportions $\pi_1, \ldots, \pi_g$ and by the parameters $\boldsymbol{\alpha}_k = (\alpha_{kj}^h; j = 1, \ldots, d; h = 1, \ldots, m_j)$ of the pdf of each component.

As before, estimating the parameter $\boldsymbol{\theta}$, and possibly estimating the native class of each of the $\boldsymbol{x}_i$, may be achieved by maximizing the likelihood $\mathrm{L}(\boldsymbol{\theta};\mathbf{x})$ using the EM algorithm, or by maximizing the complete-data likelihood $\mathrm{L_C}(\boldsymbol{\theta},\mathbf{z})$ using the CEM algorithm. In the case of the EM algorithm, the computation of the parameters $\boldsymbol{\alpha}_k$ at step M is defined by the equation $\alpha_{kj}^h = \sum_i \widetilde{z}_{ik} x_{ij}^h / \sum_i \widetilde{z}_{ik}$, where $\widetilde{z}_{ik}$ are the probabilities obtained in the usual fashion at step E. In the case of the CEM algorithm, the computation of the parameters $\boldsymbol{\alpha}_k$ becomes $\alpha_{kj}^h = \sum_i z_{ik} x_{ij}^h / z_{.k}$, where $\mathbf{z}$ is the partition obtained by the MAP from the probabilities $\widetilde{z}_{ik}$.

We now look at what happens to the complete-data likelihood criterion when the clustering approach is used with the assumption that the proportions π_k are constant. If we denote $s_{ij}^k = \sum_i z_{ik} x_{ij}^h$, $s_{\cdot}^{jh} = \sum_i x_{ij}^h$, $s_k^{\cdot} = \sum_j \sum_{h=1}^{m_j} s_{ij}^k$ and $s_{\cdot}^{\cdot} = \sum_{i,j} \sum_{h=1}^{m_j} x_{ij}^h = nd$, we can easily show that the equation

$$\mathrm{L_C}(\mathbf{z},\boldsymbol{\theta}) = \sum_{k,j} \sum_{h=1}^{m_j} s_{ij}^k \log \alpha_{kj}^h$$

is obtained.

Given that, at convergence, $\alpha_{kj}^{h} = s_{ij}^{k}/z_{.k}$, it can be shown [CEL 91] that the CEM algorithm maximizes the information criterion [BEN 73a]

$$H(\mathbf{z}, J) = \sum_{k,j} \sum_{h=1}^{m_j} \frac{s_{ij}^{k}}{s_{\cdot}^{\cdot\cdot}} \log \frac{s_{\cdot}^{\cdot\cdot}\, s_{k}^{jk}}{s_{k}^{\cdot\cdot} s_{\cdot}^{jh}}$$

which represents the information from the initial table, retained by the partition $\mathbf{z}$, and yielding results very close to the χ^2 criterion

$$\chi^2(\mathbf{z}, J) = \sum_{k,j} \sum_{h=1}^{m_j} \frac{(s_{\cdot}^{\cdot\cdot}\, s_{ij}^{k} - s_{k}^{\cdot\cdot} s_{\cdot}^{jh})^2}{s_{\cdot}^{\cdot\cdot}\, s_{k}^{\cdot\cdot} s_{\cdot}^{jh}}.$$

Therefore, it follows that to seek a partition into g classes maximizing the information criterion or the χ^2 criterion (approximately equivalent) is to assume that the data derive from a latent class model.

A parallel may be drawn here with the analysis of multiple correspondences. It is not difficult to see that with the geometrical representation used in this factorial analysis, the χ^2 criterion is quite simply the familiar criterion of intraclass inertia.

1.8.2. *Parsimonious model*

The number of parameters $(g - 1) + g * \sum_j (m_j - 1)$ required by the latent class model that we have just described is usually considerably smaller than the number of parameters $\prod_j m_j$ required by the complete log-linear model. For example, for a number of classes g is equal to 5 and a number of qualitative variables d are equal to 10, and where the number of modalities m_j is 4 for all the variables, the number of parameters for the two models is, respectively, 154

and 10^6. In many cases, this number will be quite excessive, and more parsimonious models are called for.

To this end, we begin by remarking that if, for each variable j, the modality of highest probability is denoted as h^*, the model may therefore be re-parameterized as follows

$$f(\boldsymbol{x}_i;\boldsymbol{\theta}) = \sum_k \pi_k \prod_j \left((1-\varepsilon_{kj}^{h^*})^{1-d(x_{ij},a_{kj})} \prod_{h\neq h^*} (\varepsilon_{kj}^{h})^{|x_{ij}^h - a_k^{jh}|} \right),$$

where $\mathbf{a}_{kj} = (a_{kj}^1,\ldots,a_{kj}^{m_j})$ with $a_{kj}^h = 1$ if $h = h^*$ and 0 otherwise, $\boldsymbol{\varepsilon}_{kj} = (\varepsilon_{kj}^1,\ldots,\varepsilon_{kj}^{m_j})$ where $\varepsilon_{kj}^h = 1-\alpha_{kj}^h$ if $h = h^*$ and α_{kj}^h otherwise, and $\delta(x_{ij},a_{kj}) = 0$ if x_{ij} and a_{kj} take the same modality and 1 otherwise. Like for binary data, the vector $\mathbf{a}_k = (\mathbf{a}_{k1},\ldots,\mathbf{a}_{kd})$ may be interpreted as the center of the class k and the vectors $\boldsymbol{\varepsilon}_k^j$ as dispersions. For example, if the parameter α_{kj} is equal to the vector $(0.7, 0.2, 0.1)$, the new parameters become $\mathbf{a}_{kj} = (1,0,0)$ and $\boldsymbol{\varepsilon}_{kj} = (0.3, 0.2, 0.1)$.

With the model restated in this way, it is possible to introduce simple constraints, such as requiring non-majority modalities to have the same dispersion

$$\begin{cases} \varepsilon_{kj}^{h^*} = \varepsilon_{kj} \\ \varepsilon_{kj}^{h} = (1-\varepsilon_{kj})/(m_j-1) \text{ for } h \neq h^*. \end{cases}$$

This gives us a model used in discrimination, where the number of parameters has been reduced from $(g-1) + g * \sum_j (m_j - 1)$ to $(g-1) + \sum_j (m_j - 1)$.

Even more parsimonious models may be obtained if additional constraints are placed on dispersions, for example by requiring that $\varepsilon_{kj}^{h^*}$ should not depend on the variable (model $[\boldsymbol{\varepsilon}_k]$), on the class (model $[\boldsymbol{\varepsilon}_j]$) or on neither the variable nor the class (model $[\boldsymbol{\varepsilon}]$). If we restrict ourselves to this last model $[\boldsymbol{\varepsilon}]$, and if we require proportions to be equal,

the complete-data log-likelihood can be expressed quite simply

$$\mathrm{L_C}(\mathbf{z}, \boldsymbol{\theta}) = \log \frac{\varepsilon}{1-\varepsilon} \sum_k \sum_{i=1}^{n} z_{ik} \mathbf{d}(\boldsymbol{x}_i, \mathbf{a}_k) + nd \log(1-\varepsilon),$$

where $\mathbf{d}(\boldsymbol{x}_i, \mathbf{a}_k)$ is a distance reflecting the number of different modalities between the vector $\boldsymbol{x}_i$ and the center $\mathbf{a}_k$. At the clustering step of the CEM algorithm, the individuals i are thus allotted to the class k that minimizes $\mathbf{d}(\boldsymbol{x}_i, \mathbf{a}_k)$, and at step M, the co-ordinates a_{kj} of the centers $\mathbf{a}_k$ are obtained by taking the majority of modalities. Furthermore, Jollois and Nadif [JOL 02] considered the clustering of categorical data under the classification maximum likelihood approach. In this setting, with a parsimonious multinomial mixture model, they defined a generalization of the k-modes criterion [HUA 98]. They showed that k-modes is just a particular version of CEM and the k-modes criterion is associated with a multinomial mixture model $[\varepsilon]$ with supplementary constraints that are too restrictive: the proportions are assumed to be equal and the variables to have the same number of categories. They conducted experiments showing the superiority of CEM with the model on k-modes when these assumptions are not verified.

In practice, we suggest taking very simple models by class, for example latent class models with a single parameter, and then increasing the number of components if necessary.

When the qualitative variables are ordinal, it is possible either to convert the data into binary data or to use an approach similar to the approach we have just described, taking into account the order that exists among the modalities.

1.9. Contingency tables

To measure the information provided by a contingency table, we need to evaluate the links existing between the two sets I and J as discussed in the Introduction. Several measures of association exist, and one of the most frequently employed is the phi-squared criterion ϕ^2 (see Introduction and Chapter 4). This criterion, used, for example, in correspondence analysis (CA), is defined as follows

$$\phi^2(I,J) = \sum_{i,j} \frac{(p_{ij} - p_{i.}p_{.j})^2}{p_{i.}p_{.j}}.$$

The phi-squared criterion can be used to evaluate the quality of a partition $\mathbf{z}$ of I: to this end, we associate the partition $\mathbf{z}$ with the phi-squared $\phi^2(\mathbf{z},J)$ of the contingency table with g rows and d columns obtained from the initial table in computing the sum of the rows of each cluster. It can be shown that

$$\phi^2(I,J) \geq \phi^2(\mathbf{z},J) \qquad [1.10]$$

and therefore the proposed regrouping necessarily leads to a loss of information. The objective of classification is to find the partition $\mathbf{z}$ that minimizes this loss, i.e. which maximizes $\phi^2(\mathbf{z},J)$. We notice that when the row profiles are equal for each cluster, inequality [1.10] becomes $\phi^2(\mathbf{z},J) = \phi^2(I,J)$, and in this particular case, there is no loss of information. In addition, the problem is meaningful only when the number of clusters is fixed. Otherwise, the optimal partition is simply the partition where each element of I forms a cluster.

1.9.1. *MNDKI2 algorithm*

The MNDKI2 algorithm is based on the same geometrical representation of a contingency table as that used in CA. This

representation is justified for several reasons, in particular, because of the similar role played by each of the two dimensions in the analyzed table, and also because of the property of distributional equivalence, which implies stable results when agglomerating elements with similar profiles. In this representation, each row i corresponds to a point vector $\mathbb{R}^d$ defined by the profile p_J^i weighted by the marginal frequency $p_{i.}$. The distances among profiles is not defined by the usual Euclidean metric, but instead by the weighted Euclidean metric, known as the *chi-squared metric* D^2, defined by the diagonal matrix $\mathrm{diag}(\frac{1}{p_{.1}}, \ldots, \frac{1}{p_{.d}})$.

If $\mathbf{z}$ is a partition of the rows, we can define the frequencies $p_{kj} = \sum_i z_{ik} p_{ij}$ and the average row profile of the kth cluster

$$p_J^k = \left(\frac{p_{k1}}{p_{k.}}, \ldots, \frac{p_{kd}}{p_{k.}} \right)^t,$$

where $p_{k.} = \sum_j p_{kj}$. With this representation, we can show after some calculation that the total of squared distances T, the between-cluster sums of squares $B(\mathbf{z})$ and the within-cluster sums of squares $W(\mathbf{z})$ can be written as

$$T = \sum_i p_{i.} D^2(p_J^i, p_J) = \phi^2(I, J),$$

$$B(\mathbf{z}) = \sum_k p_{k.} D^2(p_J^k, p_J) = \phi^2(\mathbf{z}, J),$$

and

$$\mathrm{W}(\mathbf{z}) = \sum_{i,k} z_{ik} p_{i.} D^2(p_J^i, p_J^k).$$

The traditional equation between the total of squared distances, the within-cluster sums of squares and the

between-cluster sums of squares $T = \mathrm{W}(\mathbf{z}) + B(\mathbf{z})$ leads to the following equation

$$\phi^2(I, J) = \mathrm{W}(\mathbf{z}) + \phi^2(\mathbf{z}, J).$$

The term $\mathrm{W}(\mathbf{z})$, therefore, represents the information lost when grouping the elements according to the partition $\mathbf{z}$, and $\phi^2(\mathbf{z}, J)$ corresponds to the information which is preserved. Consequently, since the quantity $\phi^2(I, J)$ does not depend on the partition $\mathbf{z}$, looking for the partition maximizing the criterion $\phi^2(\mathbf{z}, J)$ is equivalent to looking for the partition minimizing criterion $W(\mathbf{z})$. To minimize this criterion, it is possible to apply k-means to the set of profiles with the χ^2 metric. An iterative algorithm, known as MNDKI2, is thus obtained, locally maximizing $\phi^2(\mathbf{z}, J)$.

The question that naturally arises is: which probabilistic model does the criterion $\phi^2(\mathbf{z}, J)$ minimized by the MNDKI2 algorithm correspond to? The answer to this question will not only shed some light on this criterion, but it will also help us to propose other criteria. This is a question that we will focus on. Unfortunately, unlike the standard k-means algorithm, the MNDKI2 algorithm does not correspond to the classification approach associated with a mixture model [GOV 89]. However, using a mixture of multinomial distributions, examined in the following section, we will obtain approximately similar properties.

1.9.2. *Model-based approach*

1.9.2.1. *Multinomial mixture*

A contingency table can be obtained using a mixture of multinomial distributions by the following process of simulation [GOV 07]:

– $\mathbf{z}$: each individual is allotted to a class according to a multinomial distribution with parameters $(\pi_1, \ldots, \pi_g)$;

– $\mathbf{x}_I = (x_{1.}, \ldots, x_{n.})$: generate each row sum $x_{i.}$ according to a discrete distribution ψ such as a Poisson or a binomial distribution;

– $\mathbf{x}$: each $\boldsymbol{x}_i$ is assumed to arise from a multinomial distribution with parameters $x_{i.}$ and $\alpha_{k1}, \ldots, \alpha_{kd}$.

Thus, if $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_g, \alpha_{11}, \ldots, \alpha_{gd})$ denotes the parameter of the model and φ is the multinomial distribution of the kth component, the pdf of this model is written as

$$\begin{aligned} f(\boldsymbol{x}_i; \boldsymbol{\theta}) &= \psi(x_{i.}) \sum_k \pi_k \varphi(\boldsymbol{x}_i; x_{i.}, \alpha_{k1}, \ldots, \alpha_{kd}) \\ &= \psi(x_{i.}) \sum_k \pi_k \frac{x_{i.}!}{x_{i1}! \ldots x_{is}!} \alpha_{k1}^{x_{i1}} \ldots \alpha_{kd}^{x_{id}} \\ &= A \sum_k \pi_k \alpha_{k1}^{x_{i1}} \ldots \alpha_{kd}^{x_{id}}, \end{aligned}$$

where $A = \psi(x_{i.}) \frac{x_{i.}!}{x_{i1}! \ldots x_{id}!}$ does not depend on the parameter $\boldsymbol{\theta}$. The log-likelihood (without the additional constant $\log A$) can therefore be written as

$$L(\boldsymbol{\theta}; \mathbf{x}) = \sum_i \log \sum_k \pi_k \alpha_{k1}^{x_{i1}} \ldots \alpha_{kd}^{x_{id}}, \qquad [1.11]$$

and the complete data log-likelihood is as follows

$$L(\boldsymbol{\theta}; \mathbf{x}, \mathbf{z}) = \sum_{i,k} z_{ik} \left(\ln \pi_k + \sum_j x_{ij} \log \alpha_{kj} \right). \qquad [1.12]$$

The classical problem is, therefore, to estimate the parameter $\boldsymbol{\theta}$ from the sample. In the clustering context, the

mixture model serves to find the component from which each row arises. Next, we see how the EM and CEM algorithms allow us to achieve this goal.

1.9.2.2. *EM algorithm*

For this multinomial mixture model, the application of the EM algorithm described in section 1.4.3 to the sample $\mathbf{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ leads in the M-step to $\alpha_{kj} = \frac{\sum_i \widetilde{z}_{ik} x_{ij}}{\sum_i \widetilde{z}_{ik} x_{i.}}$. The different steps of EM are then expressed in algorithm 1.1.

Algorithm 1.1 Multinomial EM

input: $\mathbf{x}$, g
initialization: $\mathbf{z}$, $\pi_k = \frac{z_{.k}}{n}$, $\alpha_{kj} = \frac{\sum_i z_{ik} x_{ij}}{\sum_i z_{ik} x_{i.}}$
repeat
  E-step. $\widetilde{z}_{ik} \propto \pi_k \alpha_{k1}^{x_{i1}} \ldots \alpha_{kd}^{x_{id}}$
  M-step. $\pi_k = \frac{\widetilde{z}_{.k}}{n}$, $\alpha_{kj} = \frac{\sum_i \widetilde{z}_{ik} x_{ij}}{\sum_i \widetilde{z}_{ik} x_{i.}}$
until convergence
return $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$

In the maximum likelihood approach of the classical mixture model, after we have estimated the parameter $\boldsymbol{\theta}$, we can give a probabilistic clustering of the n rows in terms of their fitted posterior probabilities of component membership, and obtain a partition using a classification step that assigns each object to the component of the mixture to which it has the highest posterior probability of belonging.

1.9.2.3. *CEM algorithm*

Recall that in this classification approach, a C-step that converts the posterior probabilities $\widetilde{z}_{ik}$s to a discrete classification is included prior to performing the M-step. The different steps of the CEM algorithm are then expressed in algorithm 1.2.

Algorithm 1.2 Multinomial CEM

input: $\mathbf{x}$, g
initialization: $\mathbf{z}$, $\pi_k = \frac{z_{.k}}{n}$, $\alpha_{kj} = \frac{\sum_i z_{ik} x_{ij}}{\sum_i z_{ik} x_{i.}}$
repeat
  E-step. $\widetilde{z}_{ik} \propto \pi_k \alpha_{k1}^{x_{i1}} \dots \alpha_{kd}^{x_{id}}$
  C-step. $z_i = \arg\max_k \widetilde{z}_{ik}$
  M-step. $\pi_k = \frac{z_{.k}}{n}$, $\alpha_{kj} = \frac{\sum_i z_{ik} x_{ij}}{\sum_i z_{ik} x_{i.}}$
until convergence
return $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$, $\mathbf{z}$,

Having established an estimate of the parameters, and denoting $p_{kj} = \frac{x_{kj}}{x_{k.}}$, we can express the criterion as

$$\begin{aligned} L(\boldsymbol{\theta}; \mathbf{x}, \mathbf{z}) &= \sum_k z_{.k} \ln \pi_k + \sum_{k,j} x_{kj} \log \frac{x_{kj}}{x_{k.}} \\ &= \sum_k z_{.k} \ln \pi_k + n \sum_{k,j} p_{kj} \log \frac{p_{kj}}{p_{k.} p_{.j}} + n \sum_j p_{.j} \log p_{.j}. \end{aligned} \quad [1.13]$$

Note that the term $\sum_{k,j} p_{kj} \log \frac{p_{kj}}{p_{k.} p_{.j}}$ is the mutual information $\mathcal{I}(\mathbf{z}, J)$ quantifying the information shared between $\mathbf{z}$ and J. This can easily be shown using the definition in terms of entropies

$$\mathcal{I}(\mathbf{z}, J) = \mathrm{H}(\mathbf{z}) + \mathrm{H}(J) - \mathrm{H}(\mathbf{z}, J),$$

where $\mathrm{H}(.)$ is the entropy. This mutual information can be linked to the ϕ^2 criterion as follows: first, using the equalities $\sum_{k,j} p_{k.} p_{.j} = 1$ and $\sum_{k,j} p_{kj} = 1$, we have the equation

$$\sum_{k,j} \frac{(p_{kj} - p_{k.} p_{.j})^2}{p_{k.} p_{.j}} = \sum_{k,j} p_{k.} p_{.j} \left(\left(\frac{p_{kj}}{p_{k.} p_{.j}} \right)^2 - 1 \right).$$

Second, using the approximation $x^2 - 1 \approx 2x \log x$, excellent in the neighborhood of 1 and good in the interval $[0, 3]$, the approximation

$$\sum_{k,j} p_{k.} p_{.j} \left(\left(\frac{p_{kj}}{p_{k.} p_{j.}} \right)^2 - 1 \right) \approx 2 \sum_{k,j} p_{kj} \log \frac{p_{kj}}{p_{k.} p_{.j}}$$

can be obtained [BEN 73b]. Finally, we have the following approximation

$$\sum_{k,j} p_{kj} \log \frac{p_{kj}}{p_{k.} p_{.j}} \approx \frac{1}{2} \sum_{k,j} \frac{(p_{kj} - p_{k.} p_{.j})^2}{p_{k.} p_{.j}}, \qquad [1.14]$$

and therefore,

$$\mathcal{I}(\mathbf{z}, J) \approx \frac{1}{2} \phi^2(\mathbf{z}, J).$$

Therefore, from equations [1.13] and [1.14], when the proportions are fixed, the maximization of $\mathrm{L_C}(\boldsymbol{\theta}; \mathbf{z})$ is equivalent to the maximization of the mutual information $\mathcal{I}(\mathbf{z}, J)$, and approximately equivalent to the maximization of the phi-squared criterion $\phi^2(\mathbf{z}, J)$: the use of the two criteria $\phi^2(\mathbf{z}, J)$ and $I(\mathbf{z}, J)$, therefore, is based on the implicit assumption that the data arise from a mixture of multinomial distributions.

1.9.3. *Illustration*

To illustrate the results obtained by MNDKI2, we will use a CA representation. Let us recall that CA is an exploratory multivariate technique that converts a contingency table into a particular type of graphical display in which the rows and the columns of the matrix are depicted as points [BEN 73b, GRE 88b, LEB 84]. It can be used on any two-way table, sparse or not, as the case may be. It projects the rows and columns of a data matrix into points within a graph in a

Euclidean space. The graph is, therefore, used to gain some understanding of the data and to extract information from it. To show the links between MNDKI2 and CA, we will use a comparison of time-budgets as an example [JAM 76]. We have a data matrix where x_{ij} represents the amount of time spent on a variety of activities i by a population j during a given time period. The set I comprises 10 activity clusters: prof (professional), tran (transport), home (housework), child (activities pertaining to childcare), shop (shopping), wash (washing and personal care), meal (mealtime), sleep, tv (television) and leis (other leisure activities). The set J is composed of 28 types of population characterized by gender, country, professional activity and marital status. The vector of letters identifying each population can be interpreted as follows: m or w (man or woman), a or na (active or not active professionally), s or ns (single or not single), us, we, ea or yu (USA, western country, eastern country or Yugoslavia); for instance, *mnsyu* corresponds to a man, not single and from Yugoslavia. We have presented the data in Table 1.3.

Here, we present the best result obtained by MNDKI2 from among 10 random initial positions when the number of clusters in partition $\mathbf{z}$ is 3. The initial $\phi^2(I, J)$ value is 9658.38 and the resulting $\phi^2(\mathbf{z}, J)$ value is 8386.83. The percentage of ϕ^2 accounted for by the partition is very good in this small example: more than 86% of the ϕ^2 is preserved. The clusters in the obtained partition $\mathbf{z}$ are the following: cluster 1: home, child; cluster 2: prof, tran; and cluster 3: sleep, wash, leis, meal, shop, tv.

The column profiles $\frac{p_{ij}}{p_{i.}p_{.j}}$ (with a multiple coefficient $f_{i.}$) reorganized according to $\mathbf{z}$ are reported in Table 1.4. We observe the similarity of the profiles belonging to each cluster. The most interesting values are those that are a long way from the mean 1. They characterize the partition: for example, the category $wnaus$ is a characteristic of the clusters 1 and 2.

	prof	tran	home	child	shop	wash	meal	sleep	tv	leis
maus	610	140	60	10	120	95	115	760	175	315
waus	475	90	250	30	140	120	100	775	115	305
wnaus	10	0	495	110	170	110	130	785	160	430
mnsus	615	141	65	10	115	90	115	765	180	305
wnsus	179	29	421	87	161	112	119	776	143	373
msus	585	115	50	0	150	105	100	760	150	385
wsus	482	94	196	18	141	130	96	775	132	336
mawe	652	100	95	7	57	85	150	807	115	330
wawe	510	70	307	30	80	95	142	815	87	262
wnawe	20	7	567	87	112	90	180	842	125	367
mnswe	655	97	97	10	52	85	152	807	122	320
wnswe	168	22	529	69	102	83	174	825	119	392
mswe	642	105	72	0	62	77	140	812	100	387
wswe	389	34	262	14	92	97	147	848	84	392
mayu	650	140	120	15	85	90	105	760	70	365
wayu	560	105	375	45	90	90	95	745	60	235
wnayu	10	10	710	55	145	85	130	815	60	380
mnsyu	650	145	112	15	85	90	105	760	80	357
wnsyu	260	52	576	59	116	85	117	775	65	295
msyu	615	125	95	0	115	90	85	760	40	475
wsyu	413	89	318	23	112	96	102	774	45	409
maea	650	142	122	22	76	94	100	764	96	334
waea	578	106	338	42	106	94	52	752	64	228
wnaea	24	8	594	72	158	92	128	840	86	398
mnsea	652	133	134	22	68	94	102	762	122	310
wnsea	434	77	431	60	117	88	105	770	73	229
msea	627	148	68	0	88	92	86	770	58	463
wsea	433	86	296	21	128	102	94	758	58	379

Table 1.3. *Transposed time-budget data matrix*

To illustrate the relationship between CA and MNDKI2, we have shown in Figure 1.4 the representation of I on the first two axes that account for 84% of ϕ^2. We can observe that clusters 1 and 2 are strongly opposed and cluster 3 is the middle cluster.

1.10. Implementation

There are a number of software developments implementing the methods described in this chapter, not least

among them is the MIXMOD[1] program. In this section, we give a quick overview of the problems that software implementations need to address.

	prof	tran	home	child	shop	wash	meal	sleep	tv	leis
maus	1359	1624	216	300	1103	1000	985	968	1758	903
waus	1058	1044	901	899	1286	1263	856	987	1155	874
wnaus	22	0	1785	3297	1562	1158	1113	1000	1607	1232
mnsus	1370	1635	234	300	1056	947	984	974	1807	874
wnsus	399	336	1518	2607	1479	1179	1019	988	1436	1069
msus	1304	1334	180	0	1378	1105	856	968	1507	1103
wsus	1074	1091	707	539	1296	1369	822	987	1326	963
mawe	1454	1161	343	210	524	896	1285	1029	1156	947
wawe	1137	813	1108	900	736	1001	1217	1039	875	752
wnawe	45	81	2047	2611	1030	949	1543	1074	1257	1053
mnswe	1461	1127	350	300	478	896	1303	1029	1227	918
wnswe	362	247	1844	1999	906	845	1440	1015	1155	1086
mswe	1432	1220	260	0	570	812	1200	1035	1006	1111
wswe	882	401	961	427	860	1039	1280	1099	858	1143
mayu	1448	1624	433	450	781	947	899	968	703	1046
wayu	1248	1218	1352	1349	827	947	813	949	603	674
wnayu	22	116	2560	1648	1332	895	1113	1038	603	1089
mnsyu	1449	1683	404	450	781	948	899	968	804	1024
wnsyu	579	603	2077	1768	1066	895	1002	987	653	845
msyu	1370	1450	343	0	1057	947	728	968	402	1361
wsyu	928	1041	1156	695	1037	1019	880	994	456	1182
maea	1448	1648	440	659	698	990	856	973	964	957
waea	1310	1251	1239	1280	991	1006	453	974	654	665
wnaea	53	93	2142	2158	1452	969	1096	1070	864	1141
mnsea	1454	1544	483	660	625	990	874	971	1226	889
wnsea	974	899	1564	1810	1082	933	905	987	738	661
mnsea	1397	1717	245	0	809	969	736	981	583	1327
wsea	983	1017	1088	641	1199	1094	820	984	594	1107

Table 1.4. *The* $1,000 \times \frac{p_{ij}}{p_{i.}p_{.j}}$ *profiles reorganized according to the partition* $\mathbf{z}$

1 www.mixmod.org.

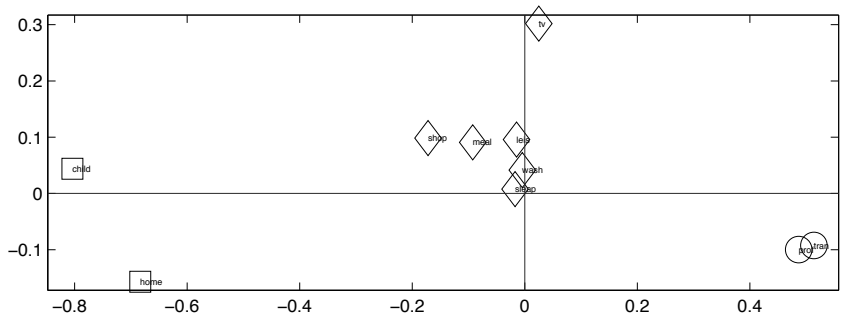

Figure 1.4. *Projection of the columns into the factorial plane spawned by the first and second axes*

1.10.1. *Choice of model and of the number of classes*

Clustering methods are often justified heuristically, and choosing the "right" method or the "right" number of classes can be a problem that is difficult, and often badly stated. The use of clustering methods based on mixture models allows us to place the problem within the more general framework of the selection of probabilistic models.

In the Bayesian context, choosing the most probable model calls for frequently used selection criteria such as Schwarz's [SCH 78] BIC criterion comprising two terms: the first is likelihood, which tends to favor the more complex model, and the second is a penalizing term, an increasing function of the number of the model's parameters. Worth mentioning is the ICL criterion [BIE 00] which, taking the objective of the clustering into account, generally provides good solutions.

1.10.2. *Strategies for use*

Maximizing the likelihood criterion via the EM algorithm or maximizing the clustering likelihood via the CEM algorithm always involves obtaining a series of solutions that see the criterion increase to a local maximum, and which are

therefore dependent on the initial position selected by the algorithm. The strategy usually adopted for obtaining a "good" solution is to run the algorithm several times from different starting points and to retain the best solution. For example, see [BIE 03], where some subtle and effective strategies are examined, including an initial phase in which the algorithm is run a large number of times without waiting for complete convergence.

1.10.3. *Extension to particular situations*

We have seen that the mixture model in clustering can cope with a variety of situations (spherical or non-spherical classes, equal or unequal proportions, etc.) and deal with both continuous and binary data. In this section, we briefly list some clustering problems that the mixture model approach addresses quite naturally, illustrating its adaptability to particular situations.

Noisy data: atypical or outlier data (measurement errors, etc.) generally perturb clustering methods quite considerably. Getting mixture models to take account of noise can be a simple matter, for example by adding a uniformly distributed class or by using distributions less sensitive to atypical elements, such as Laplace distributions.

Incomplete labeling in discrimination: in discrimination we often have, in addition to the learning sample whose class is known, a (sometimes large) set of observations whose class is not known. Making use of these unlabeled observations, which can significantly improve the results of the discrimination, can be easily accomplished by introducing observations whose membership to a class is not brought into question during the iterations of the algorithm to the EM and CEM algorithms.

Spatial data: the mixture model is based on the hypothesis that the vector $\mathbf{z} = (z_1, \ldots, z_n)$ grouping the classes of the different observations is an independent sample. There are, however, more complex situations, such as the segmentation of pixels in image processing, where this hypothesis must be rejected. In these cases, the mixture model may be extended to the clustering of geographically localized multivariate observations such as hidden Markov fields, so as to include this type of data.

Block clustering: the clustering methods described thus far were all designed to classify individuals, or occasionally variables, but there are other methods, often known as block or simultaneous clustering methods, which process the two sets simultaneously and organize the data into homogeneous blocks. Here too, it is possible to extend the use of mixture models [GOV 02] by using a latent block model generalizing the mixture model [GOV 03, GOV 05, GOV 06, GOV 08]. In the following chapters, we will focus on this model.

1.11. Conclusion

In this chapter, we have attempted to show the advantages of using mixture models in clustering. This approach provides a general framework capable of taking into account specificities in the data and in the problem. Moreover, a probabilistic model means being able to harness the entire set of statistical results in proposing solutions to difficult problems such as the choice of the model or the number of classes.

Obviously, one of the difficulties with this approach is in deciding whether the selected mixture model is realistic for the data in question. However, as Everitt [EVE 93] has rightly observed, it is not a difficulty specific to this approach. We cannot avoid choosing a method's underlying hypotheses simply by "concealing" them.