High-Dimensional Data Clustering

Charles Bouveyron, Stéphane Girard, Cordelia Schmid

High-Dimensional Data Clustering

C. Bouveyron[a,b], S. Girard[a] and C. Schmid[b]

[a] *LMC-IMAG, Université Grenoble 1, BP. 53, 38041 Grenoble Cedex 9, France*

[b] *INRIA Rhône-Alpes, 655 avenue de l'Europe, 38330 Saint-Ismier Cedex, France*

Abstract

Clustering in high-dimensional spaces is a difficult problem which is recurrent in many domains, for example in image analysis. The difficulty is due to the fact that high-dimensional data usually live in different low-dimensional subspaces hidden in the original space. This paper presents a family of Gaussian mixture models designed for high-dimensional data which combine the ideas of dimension reduction and parsimonious modeling. These models give rise to a clustering method based on the Expectation-Maximization algorithm which is called High-Dimensional Data Clustering (HDDC). In order to correctly fit the data, HDDC estimates the specific subspace and the intrinsic dimension of each group. Our experiments on artificial and real datasets show that HDDC outperforms existing methods for clustering high-dimensional data.

Key words: Model-based clustering, high-dimensional data, Gaussian mixture models, subspace selection, dimension reduction, parsimonious models.

1 Introduction

Clustering in high-dimensional spaces is a recurrent problem in many fields of science, for example in image analysis. Indeed, the data used in image analysis are often high-dimensional and this penalizes clustering methods. Popular clustering methods are based on the Gaussian Mixture Model (GMM) [24] and show a disappointing behavior when the size of the dataset is too small compared to the number of parameters to estimate. This well-known phenomenon is called *curse of dimensionality* and was introduced by Bellman [3]. We refer to [26, 27] for a theoretical study of the effect of dimensionality in the supervised framework.

To avoid overfitting, it is necessary to find a balance between the number of parameters to estimate and the generality of the model. In this paper we propose a Gaussian mixture model which takes into account the specific subspace in which each cluster is located and therefore limits the number of parameters to estimate. The Expectation-Maximization (EM) algorithm [10] is used for parameter estimation and the intrinsic dimension of each group is determined automatically with the scree-test of Cattell. This allows to derive a robust clustering method in high-dimensional spaces, called High Dimensional Data Clustering (HDDC). In order to further limit the number of parameters, it is possible to make additional assumptions on the model. We can for example assume that classes are spherical in their subspaces or fix some parameters to be common between classes. Finally, HDDC is evaluated and compared to standard clustering methods on artificial and real datasets. We show that our approach outperforms existing clustering methods.

This paper is organized as follows. Section 2 presents the state of the art on clustering of high-dimensional data. Section 3 introduces our parameterization of the Gaussian mixture model. Section 4 presents the clustering method HDDC, *i.e.* the estimation of the parameters of the models and of the hyper-parameters. Experimental results for our clustering method on simulated and real datasets are reported in Section 5.

2 Related work on high-dimensional clustering

Standard methods to overcome the *curse of dimensionality* consist in reducing the dimension of the data and/or to use a parsimonious Gaussian mixture model. More recently, methods which find clusters in different subspaces have been proposed. In this section, we give a brief survey of these works in clustering of high-dimensional data.

2.1 Dimension reduction

Many methods use global dimension reduction techniques to overcome problems due to high dimensionality. A widely used solution is to reduce the dimension-

ality of data before using a classical clustering method. Dimension reduction techniques can be divided into techniques for *feature extraction* and *feature selection*. Feature extraction techniques build new variables carrying a large part of the global information. Among these techniques, the most popular is Principal Component Analysis (PCA) [19] which is often used in data mining and image analysis. However, PCA is a linear technique, *i.e.* it only takes into account linear dependences between variables. Recently, many non-linear techniques have been proposed such as Kernel PCA [30], non-linear PCA [15, 17] and neural networks based techniques [9, 20, 29, 34]. In [31], the dimension reduction problem was considered in the Quadratic Discriminant Analysis framework. In contrast, feature selection techniques find an appropriate subset of the original variables to represent the data. A survey on feature selection can be found in [16]. A recent approach [28] proposes to combine global feature selection and model-based clustering. These global dimension reduction techniques are often advantageous in terms of performance, but suffer from the drawback of losing information which could be discriminant. Indeed, the clusters are usually hidden in different subspaces of the original feature space and a global approach cannot capture this.

2.2 Parsimonious models

Another solution is to use models which require the estimation of fewer parameters. For example, the eigenvalue decomposition of the covariance matrices [2, 8] allows to re-parameterize the covariance matrix of the classes in their eigenspaces. By fixing some parameters to be common between classes, this parameterization yields parsimonious models which generate clustering methods based on the EM algorithm. A review on parsimonious models can be found in [14]. These approaches are based on various Gaussian models from the most complex one (a full covariance matrix for each group) to the simplest one (a spherical covariance matrix for all groups which leads to the classical k-means method) and thus allow to fit different types of data. However, these methods cannot efficiently solve the problem of the high-dimensionality when clusters live in low-dimensional subspaces and when many dimensions of the original space are irrelevant.

2.3 Subspace clustering

A recent extension of traditional clustering is to find the subspace in which the data of each cluster live. On the one hand, many subspace clustering methods use heuristic search techniques to find the subspaces. They are usually based on grid search methods and find dense clusterable subspaces. CLIQUE [1] was one of the first algorithms proposed to find clusters within subspaces of the dataset. A review on this type of methods can be found in [25]. On the other hand, methods based on mixtures of factor analyzers [24, 35] rely on a latent variables model and on an EM based procedure to cluster high-dimensional data. More recently, Bocci *et al.* [5] proposed a similar approach to cluster dissimilarity data. The model of these methods can be viewed as a mixture of constrained Gaussian densities in class-specific subspaces. The advantage of such a model is to capture correlations without estimating full covariance matrices. In this paper, we propose an unified approach for subspace clustering in the Gaussian mixture model framework which includes these approaches and involves additional regularizations as in parsimonious models.

3 Gaussian mixture models for high-dimensional data

Clustering divides a given dataset $\{x_1, ..., x_n\}$ of n data points in $\mathbb{R}^p$ into k homogeneous groups (see [18] for a review). A popular clustering technique uses Gaussian mixture models, which assume that each class is represented by a Gaussian probability density. Data $\{x_1, ..., x_n\} \in \mathbb{R}^p$ are therefore modeled by a density of the form:

$$f(x, \theta) = \sum_{i=1}^{k} \pi_i \phi(x, \theta_i), \tag{1}$$

where ϕ is a p-variate normal density with parameter $\theta_i = \{\mu_i, \Sigma_i\}$ and π_i are the mixing proportions. This model requires to estimate full covariance matrices and therefore the number of parameters increases with the square of the dimension. However, due to the *empty space* phenomenon [33] we can assume that high-dimensional data live in subspaces with a dimensionality lower than the one of

the original space. We therefore propose to work in low-dimensional class-specific subspaces and to limit the number of parameters to estimate in order to adapt model-based clustering to high-dimensional data.

3.1 The Gaussian mixture model $[a_{ij}b_iQ_id_i]$

As in the classical Gaussian mixture model framework, we assume that class conditional densities are Gaussian $\mathcal{N}(\mu_i, \Sigma_i)$ with means μ_i and covariance matrices Σ_i, for $i = 1, ..., k$. Let Q_i be the orthogonal matrix of eigenvectors of Σ_i. The class conditional covariance matrix Δ_i is therefore defined in the eigenspace of Σ_i by:

$$\Delta_i = Q_i^t \, \Sigma_i \, Q_i. \tag{2}$$

The matrix Δ_i is thus a diagonal matrix which contains the eigenvalues of Σ_i. We further assume that Δ_i is divided into two blocks:

$$\Delta_i = \begin{pmatrix} \boxed{\begin{matrix} a_{i1} & & 0 \\ & \ddots & \\ 0 & & a_{id_i} \end{matrix}} & \mathbf{0} \\ \mathbf{0} & \boxed{\begin{matrix} b_i & & 0 \\ & \ddots & \\ 0 & & b_i \end{matrix}} \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} d_i \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} (p - d_i) \end{matrix}$$

where $a_{ij} > b_i$, $\forall j = 1, ..., d_i$. The class specific subspace $\mathbb{E}_i$ is spanned by the d_i eigenvectors associated to the eigenvalues a_{ij} with $\mu_i \in \mathbb{E}_i$. We also define the subspace $\mathbb{E}_i^\perp$ such that $\mathbb{E}_i \oplus \mathbb{E}_i^\perp = \mathbb{R}^p$. In this subspace $\mathbb{E}_i^\perp$, the variance is modeled by the single parameter b_i. Let $P_i(x) = \tilde{Q}_i \tilde{Q}_i^{\,t}(x - \mu_i) + \mu_i$ and $P_i^\perp(x) = (Q_i - \tilde{Q}_i)(Q_i - \tilde{Q}_i)^t(x - \mu_i) + \mu_i$ be the projection of x on $\mathbb{E}_i$ and $\mathbb{E}_i^\perp$ respectively, where $\tilde{Q}_i$ is made of the d_i first columns of Q_i supplemented by zeros. Figure 1 summarizes these notations. Following the notation system of [8], the mixture model presented above will be denoted by $[a_{ij}b_iQ_id_i]$ in the sequel.

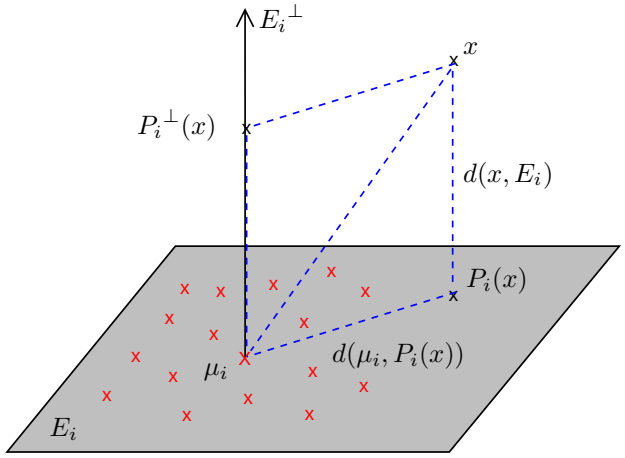

Figure 1: The subspaces $\mathbb{E}_i$ and $\mathbb{E}_i^{\perp}$ of the ith mixture component.

3.2 The sub-models of $[a_{ij}b_iQ_id_i]$

By fixing some parameters to be common within or between classes, we obtain particular models which correspond to different regularizations. For instance, if we fix the dimensions d_i to be common between the classes, we obtain the model $[a_{ij}b_iQ_id]$ which corresponds to the model of [35]. In the following, "free Q_i" means that Q_i is specific for each class C_i and "common Q_i" means that for each $i = 1, ..., k$, $Q_i = Q$ and consequently the class orientations are the same. We divide the family of the model $[a_{ij}b_iQ_id_i]$ into three categories: models with free orientations, models with common orientations and models with common covariance matrices.

Models with free orientations This category of models assumes that the groups live in subspaces with different orientations, *i.e.* the matrices Q_i are specific to each group. Clearly, the general model $[a_{ij}b_iQ_id_i]$ belongs to this category. Note that, if we assume that $d_i = (p-1)$ for all $i = 1, ..., k$, the model $[a_{ij}b_iQ_id_i]$ reduces to the classical GMM with full covariance matrices for each mixture component which yields in the supervised framework the well known Quadratic Discriminant Analysis (QDA). It is possible to add constraints on the different parameters to obtain more regularized models. For example, if we fix the first d_i eigenvalues to be common within each class, we obtain the more restricted

model $[a_i b_i Q_i d_i]$. The model $[a_i b_i Q_i d_i]$ often gives satisfying results, *i.e.* the assumption that each matrix Δ_i contains only two different eigenvalues, a_i and b_i, seems to be an efficient way to regularize the estimation of Δ_i. Another type of regularization is to fix the parameters b_i to be common between the classes. We thus obtain the model $[a_i b Q_i d_i]$ which assumes that the variance outside of the class-specific subspaces is common. This can be viewed as modeling the noise in $\mathbb{E}_i^{\perp}$ by a common parameter b and this seems natural since the data were obtained in a common acquisition process. This category of models contains also the models $[ab_i Q_i d_i]$, $[abQ_i d_i]$ and all models with free Q_i and common d_i.

Models with common orientations It is also possible to assume that the class orientations are common, *i.e.* $Q_i = Q$ for each $i = 1, ..., k$. However, this assumption does not necessarily imply that the class-specific subspaces are the same. Indeed, if the dimensions d_i are free, the intersection of the k class-specific subspaces is the one of the class with the smallest intrinsic dimension. This assumption can be interesting to model groups with some common properties and with additional specific characteristics. However, several models of this category require a complex iterative estimation based on the FG algorithm [12] and therefore they will be not considered here. Consequently, only the models $[a_i b_i Qd]$, $[ab_i Qd]$ and $[a_i bQd]$ will be considered in this paper since their parameters can be estimated using a simple iterative procedure. Note that a model similar to $[a_{ij} bQd]$ was considered by Flury *et al.* in [13] in the supervised framework with an additional assumption on the means.

Models with common covariance matrices This branch of the family contains only the two models $[a_j bQd]$ and $[abQd]$. Both models indeed assume that the classes have the same covariance matrix $\Sigma = Q\Delta Q^t$. Particularly, if we assume that $d = (p-1)$, the model $[a_j bQd]$ reduces to a Gaussian mixture model (denoted by "Com-GMM" in the following) which yields in the supervised framework the well known Linear Discriminant Analysis (LDA). Remark that if $d < (p-1)$, the model $[a_j bQd]$ can be viewed as the a combination of a dimension reduction technique with a GMM with common covariance matrices, but without losing information since the information carried by the smallest eigenvalues is not discarded.

3.3 Characteristics of the models

The family of models presented above requires only the estimation of d_i-dimensional subspaces and therefore the different models are significantly more parsimonious than the general Gaussian model if $d_i \ll p$. Table 1 summarizes some properties of the models considered here. The second column of this table gives the number of parameters to estimate. The third column provides the asymptotic order of the number of parameters (*i.e.* with the assumption that $k \ll d \ll p$). The fourth column gives the number of parameters for the particular case $k = 4$, $p = 100$ and $\forall i$, $d_i = 10$. The last column indicates whether the Maximum Likelihood (ML) estimators are closed form or not. These characteristics are also given for five Gaussian mixture models: GMM with full covariance matrices for each class (Full-GMM), with common covariance matrices between classes (Com-GMM), with diagonal covariance matrices (Diag-GMM), with spherical covariance matrices (Sphe-GMM). Note that Celeux and Govaert recommend in [8] to make use of the models Diag-GMM and Sphe-GMM in clustering problems. We can observe that all models of our family require the estimation of fewer parameters than both Full-GMM and Com-GMM. In the particular case of 100-dimensional data, made of 4 classes and with common intrinsic dimensions d_i equal to 10, the model $[a_{ij}b_iQ_id_i]$ only requires the estimation of 4 231 parameters whereas Full-GMM and Com-GMM requires respectively the estimation of 20 603 and 5 453 parameters. Remark that the model $[a_{ij}b_iQ_id_i]$, which gives rise to quadratic separation between the groups, requires the estimation of fewer parameters than Com-GMM, which gives rise to linear separation between the groups.

4 High-dimensional data clustering

In this section we derive the EM-based clustering framework for the Gaussian model $[a_{ij}b_iQ_id_i]$ and its sub-models. The clustering approach presented here will be in the following denoted by High-Dimensional Data Clustering (HDDC). We remind that unsupervised classification organizes data in homogeneous groups using only the observed values of the p explanatory variables. Usually, in model-based clustering, the parameters $\theta = \{\pi_1, ..., \pi_k, \theta_1, ..., \theta_k\}$ with $\theta_i = \{\mu_i, \Sigma_i\}$ are estimated by the EM algorithm which repeats iteratively E and M steps.

Model	Number of parameters	Asymptotic order	Nb of prms $k=4$, $d=10$, $p=100$	ML estimation
$[a_{ij}b_iQ_id_i]$	$\rho+\bar{\tau}+2k+D$	kpd	4231	CF
$[a_{ij}bQ_id_i]$	$\rho+\bar{\tau}+k+D+1$	kpd	4228	CF
$[a_ib_iQ_id_i]$	$\rho+\bar{\tau}+3k$	kpd	4195	CF
$[ab_iQ_id_i]$	$\rho+\bar{\tau}+2k+1$	kpd	4192	CF
$[a_ibQ_id_i]$	$\rho+\bar{\tau}+2k+1$	kpd	4192	CF
$[abQ_id_i]$	$\rho+\bar{\tau}+k+2$	kpd	4189	CF
$[a_{ij}b_iQ_id]$	$\rho+k(\tau+d+1)+1$	kpd	4228	CF
$[a_jb_iQ_id]$	$\rho+k(\tau+1)+d+1$	kpd	4198	CF
$[a_{ij}bQ_id]$	$\rho+k(\tau+d)+2$	kpd	4225	CF
$[a_jbQ_id]$	$\rho+k\tau+d+2$	kpd	4195	CF
$[a_ib_iQ_id]$	$\rho+k(\tau+2)+1$	kpd	4192	CF
$[ab_iQ_id]$	$\rho+k(\tau+1)+2$	kpd	4189	CF
$[a_ibQ_id]$	$\rho+k(\tau+1)+2$	kpd	4189	CF
$[abQ_id]$	$\rho+k\tau+3$	kpd	4186	CF
$[a_{ij}b_iQd_i]$	$\rho+\tau+D+2k$	pd	1396	FG
$[a_{ij}bQd_i]$	$\rho+\tau+D+k+1$	pd	1393	FG
$[a_ib_iQd_i]$	$\rho+\tau+3k$	pd	1360	FG
$[a_ibQd_i]$	$\rho+\tau+2k+1$	pd	1357	FG
$[ab_iQd_i]$	$\rho+\tau+2k+1$	pd	1357	FG
$[abQd_i]$	$\rho+\tau+k+2$	pd	1354	FG
$[a_{ij}b_iQd]$	$\rho+\tau+kd+k+1$	pd	1393	FG
$[a_jb_iQd]$	$\rho+\tau+k+d+1$	pd	1363	FG
$[a_{ij}bQd]$	$\rho+\tau+kd+2$	pd	1390	FG
$[a_ib_iQd]$	$\rho+\tau+2k+1$	pd	1357	IP
$[ab_iQd]$	$\rho+\tau+k+2$	pd	1354	IP
$[a_ibQd]$	$\rho+\tau+k+2$	pd	1354	IP
$[a_jbQd]$	$\rho+\tau+d+2$	pd	1360	CF
$[abQd]$	$\rho+\tau+3$	pd	1351	CF
Full-GMM	$\rho+kp(p+1)/2$	$kp^2/2$	20603	CF
Com-GMM	$\rho+p(p+1)/2$	$p^2/2$	5453	CF
Diag-GMM	$\rho+kp$	$2kp$	803	CF
Sphe-GMM	$\rho+k$	kp	407	CF

Table 1: Properties of the HDDC models: $\rho = kp + k - 1$ is the number of parameters required for the estimation of means and proportions, $\bar{\tau} = \sum_{i=1}^{k} d_i[p-(d_i+1)/2]$ and $\tau = d[p-(d+1)/2]$ are the number of parameters required for the estimation of $\tilde{Q}_i$ and $\tilde{Q}$, and $D = \sum_{i=1}^{k} d_i$. For asymptotic orders, we assume that $k \ll d \ll p$. CF means that the ML estimates are closed form. IP means that the ML estimation needs an iterative procedure. FG means that the ML estimation requires the iterative FG algorithm.

The reader could refer to [23] for further informations on the EM algorithm and its extensions. In particular, the models presented in this paper can be also used in the Classification EM (CEM) and Stochastic EM (SEM) algorithms presented in [7]. Using the parameterization presented in the previous section, the EM algorithm for estimating the parameters $\theta = \{\pi_i, \mu_i, \Sigma_i, a_{ij}, b_i, Q_i, d_i\}$ can be written as detailed in the following.

4.1 The E step

This step computes, at iteration q and for each $i = 1, ..., k$ and $j = 1, ..., n$, the conditional probability $t_{ij}^{(q)} = \mathbb{P}(x_j \in C_i^{(q-1)}|x_j)$ which can be written from (1) and using the Bayes formula as follows:

$$t_{ij}^{(q)} = \mathbb{P}(x_j \in C_i^{(q-1)}|x_j) = \frac{\pi_i^{(q-1)}\phi(x_j, \theta_i^{(q-1)})}{\sum_{\ell=1}^{k} \pi_\ell^{(q-1)}\phi(x_j, \theta_\ell^{(q-1)})}.$$

We can observe that the conditional probability $t_{ij}^{(q)}$ that x_j belongs to the class C_i at iteration q is mainly based on the quantity $\pi_i^{(q-1)}\phi(x_j, \theta_i^{(q-1)})$. Here, we will formulate the probability $t_{ij}^{(q)}$ using the parameters of the model $[a_{ij}b_iQ_id_i]$ presented in the previous section. In order not to overload the equations, we will omit the index of the current iteration q in the remainder of this paragraph. Writing $\phi(x, \theta_i)$ with the new class conditional covariance matrix Δ_i, we obtain:

$$-2\log(\phi(x, \theta_i)) = (x - \mu_i)^t(Q_i\Delta_iQ_i^t)^{-1}(x - \mu_i) + \log(\det \Delta_i) + p\log(2\pi).$$

Moreover, $Q_i^tQ_i = Id$ and consequently:

$$-2\log(\phi(x, \theta_i)) = \left[Q_i^t(x - \mu_i)\right]^t \Delta_i^{-1} \left[Q_i^t(x - \mu_i)\right] + \log(\det \Delta_i) + p\log(2\pi).$$

Given the structure of Δ_i, we obtain:

$$-2\log(\phi(x, \theta_i)) = \|\tilde{Q_i}^t(x-\mu_i)\|^2_{\Lambda_i} + \frac{1}{b_i}\|(Q_i - \tilde{Q_i})^t(x-\mu_i)\|^2 + \log(\det \Delta_i) + p\log(2\pi),$$

where $\|.\|^2_{\Lambda_i}$ is the Mahalanobis distance associated with the matrix $\Lambda_i = \tilde{Q_i}\Delta_i\tilde{Q_i}^t$. Using the definitions of P_i and $P_i^\perp$ (*cf.* Paragraph 3.1) and in view of Figure 1,

we obtain:

$$-2\log(\phi(x,\theta_i)) = \|\mu_i - P_i(x)\|^2_{\Lambda_i} + \frac{1}{b_i}\|x - P_i(x)\|^2 + \log(\det\Delta_i) + p\log(2\pi).$$

The relation $\log(\det\Delta_i) = \sum_{j=1}^{d_i}\log(a_{ij}) + (p-d_i)\log(b_i)$ allows to conclude that:

$$t_{ij} = 1/\sum_{\ell=1}^{k}\exp\left(\frac{1}{2}(K_i(x_j) - K_\ell(x_j))\right),$$

where $K_i(x) = -2\log(\pi_i\phi(x,\theta_i))$ is called the cost function and is defined by:

$$K_i(x) = \|\mu_i - P_i(x)\|^2_{\Lambda_i} + \frac{1}{b_i}\|x - P_i(x)\|^2 + \sum_{j=1}^{d_i}\log(a_{ij}) + (p-d_i)\log(b_i) - 2\log(\pi_i). \tag{3}$$

We can observe that $K_i(x)$ is mainly based on two distances: the distance between the projection of x on $\mathbb{E}_i$ and the mean of the class and the distance between the observation and the subspace $\mathbb{E}_i$. This cost function favours the assignment of a new observation to the class for which it is close to the subspace and for which its projection on the class subspace is close to the mean of the class. The variance terms a_{ij} and b_i balance the importance of both distances. For example, if the data are very noisy, *i.e.* b_i is large, it is natural to balance the distance $\|x - P_i(x)\|^2$ by $1/b_i$ in order to take into account the large variance in $\mathbb{E}_i^\perp$.

4.2 The M step

This step maximizes at iteration q the conditional likelihood and uses the following update formulas. The proportions and the means of the mixture are estimated by:

$$\hat{\pi}_i^{(q)} = \frac{n_i^{(q)}}{n},\ \hat{\mu}_i^{(q)} = \frac{1}{n_i^{(q)}}\sum_{j=1}^{n} t_{ij}^{(q)} x_j,$$

where $n_i^{(q)} = \sum_{j=1}^{n} t_{ij}^{(q)}$. Moreover, the update formula for the covariance matrix of the fuzzy class C_i is:

$$\hat{\Sigma}_i^{(q)} = \frac{1}{n_i^{(q)}}\sum_{j=1}^{n} t_{ij}^{(q)}(x_j - \hat{\mu}_i^{(q)})(x_j - \hat{\mu}_i^{(q)})^t.$$

The estimation of the specific parameters of HDDC is detailed below. Proofs of the following results are given in the appendix.

Models with free orientations The ML estimators of model parameters are closed form for this category of models.

– Subspace $\mathbb{E}_i$: the d_i first columns of Q_i are estimated by the eigenvectors associated with the d_i largest eigenvalues λ_{ij} of $\hat{\Sigma}_i$.

– Model $[a_{ij}b_iQ_id_i]$: the estimator of a_{ij} is $\hat{a}_{ij} = \lambda_{ij}$ and the estimator of b_i is the mean of the $(p - d_i)$ smallest eigenvalues of $\hat{\Sigma}_i$ and can be written as follows:

$$\hat{b}_i = \frac{1}{(p - d_i)} \left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d_i} \lambda_{ij} \right). \tag{4}$$

– Model $[a_{ij}bQ_id_i]$: the estimator of a_{ij} is $\hat{a}_{ij} = \lambda_{ij}$ and the estimator of b is:

$$\hat{b} = \frac{1}{(p - \xi)} \left(\mathrm{Tr}(\hat{W}) - \sum_{i=1}^{k} \hat{\pi}_i \sum_{j=1}^{d_i} \lambda_{ij} \right). \tag{5}$$

– Model $[a_ib_iQ_id_i]$: the estimator of b_i is given by (4) and the estimator of a_i is:

$$\hat{a}_i = \frac{1}{d_i} \sum_{j=1}^{d_i} \lambda_{ij}. \tag{6}$$

– Model $[ab_iQ_id_i]$: the estimator of b_i is given by (4) and the estimator of a is:

$$\hat{a} = \frac{1}{\xi} \sum_{i=1}^{k} \hat{\pi}_i \sum_{j=1}^{d_i} \lambda_{ij}. \tag{7}$$

– Model $[a_ibQ_id_i]$: the estimators of a_i and b are respectively given by (6) and (5).

– Model $[abQ_id_i]$: the estimators of a and b are respectively given by (7) and (5).

– Models with common dimensions: the estimators of the models with common dimensions d_i can be obtained from the previous ones by replacing the values d_i

by d for each $i = 1, ..., k$. In this case, equations (5) and (7) can be simplified as:

$$\hat{a} = \frac{1}{d} \sum_{j=1}^{d} \lambda_j, \tag{8}$$

$$\hat{b} = \frac{1}{(p-d)} \left(\mathrm{Tr}(\hat{W}) - \sum_{j=1}^{d} \lambda_j \right), \tag{9}$$

where λ_j is the jth largest eigenvalue of the estimated within-covariance matrix $\hat{W} = \sum_{i=1}^{k} \hat{\pi}_i \hat{\Sigma}_i$.

– Model $[a_j b_i Q_i d]$: the estimator of a_j is $\hat{a}_j = \lambda_j$ and the estimator of b_i is given by (4).

– Model $[a_j b Q_i d]$: the estimator of a_j is $\hat{a}_j = \lambda_j$ and the estimator of b is given by (9).

Models with common orientations Here, we assume in addition that the dimensions d_i are common between classes. The following ML estimators require an iterative procedure.

– Subspace $\mathbb{E}_i$: Given a_i and b_i, the d first columns of Q are estimated by the eigenvectors associated to the d largest eigenvalues of the matrix M defined by:

$$M(a_1, ..., a_k, b_1, ..., b_k) = \sum_{i=1}^{k} n_i (\frac{1}{b_i} - \frac{1}{a_i}) \hat{\Sigma}_i.$$

– Model $[a_i b_i Q d]$: given Q, the estimator of a_i and b_i are:

$$\hat{a}_i(Q) = \frac{1}{d} \sum_{j=1}^{d} q_j^t \hat{\Sigma}_i q_j, \tag{10}$$

$$\hat{b}_i(Q) = \frac{1}{(p-d)} \left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d} q_j^t \hat{\Sigma}_i q_j \right). \tag{11}$$

– Model $[a_i b Q_i d_i]$: given Q, the estimator of a_i is given by (10) and the estimator of b is:

$$\hat{b}(Q) = \frac{1}{(p-d)} \left(\mathrm{Tr}(\hat{W}) - \sum_{j=1}^{d} q_j^t \hat{W} q_j \right). \tag{12}$$

– Model $[ab_iQd]$: given Q, the estimator of b_i is defined by (11) and the estimator of a is:

$$\hat{a}(Q) = \frac{1}{d}\sum_{j=1}^{d} q_j^t \hat{W} q_j. \tag{13}$$

– Model $[a_ibQd]$: given Q, the estimators of a_i and b are respectively given by (10) and (12).

For example, it is possible to use the following iterative procedure to estimate the parameters associated to the model $[a_ib_iQd]$:

– Initialization: the d first columns of $Q^{(0)}$ are the eigenvectors associated with the d largest eigenvalues of $\hat{W}$.

– Until convergence: $a_i^{(\ell)} = \hat{a}_i(Q^{(\ell-1)})$, $b_i^{(\ell)} = \hat{b}_i(Q^{(\ell-1)})$ and the d first columns of $Q^{(\ell)}$ are the eigenvectors associated to the d largest eigenvalues of the matrix $M(a_1^{(\ell)}, ..., a_k^{(\ell)}, b_1^{(\ell)}, ..., b_k^{(\ell)})$.

Models with common covariance matrices In this category of models, the parameters can be estimated in closed form.

– Subspace $\mathbb{E}_i$: the d first columns of the matrix Q are the eigenvectors associated to the d largest eigenvalues of $\hat{W}$.

– Model $[a_jbQd]$: the estimator of a_j is $\hat{a}_j = \lambda_j$ and the estimator of b is given by (9).

– Model $[abQd]$: the estimator of a and b are respectively given by (8) and (9).

4.3 Hyper-parameters estimation

Within the M step, we also have to estimate the intrinsic dimensions of each subclass. This is a difficult problem with no unique technique to use. Our approach is based on the eigenvalues of the class conditional covariance matrix Σ_i of the class C_i. The jth eigenvalue of Σ_i corresponds to the fraction of the full variance carried by the jth eigenvector of Σ_i. We estimate the class specific dimension d_i, $i = 1, ..., k$, with the empirical method scree-test of Cattell [6] which analyzes the differences between eigenvalues in order to find a break in the scree. The selected dimension is the one for which the subsequent differences are smaller

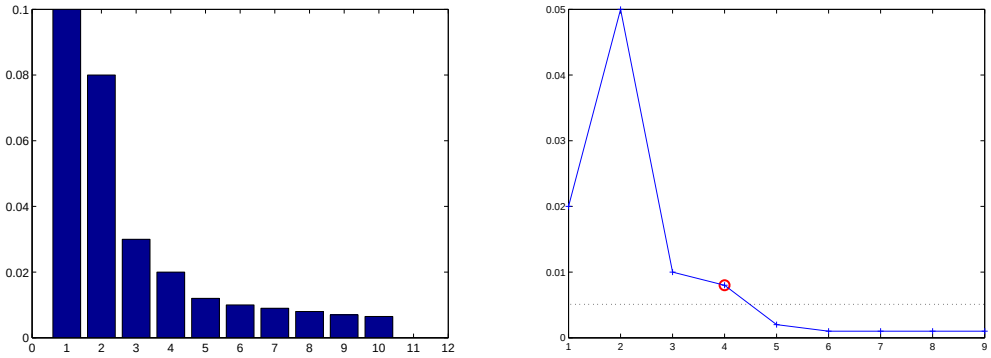

Figure 2: Estimation of the intrinsic dimension d_i using the scree-test of Cattell: plot of ordered eigenvalues of Σ_i (left) and differences between consecutive eigenvalues (right).

than a threshold. Figure 2 illustrates this method: the graph on the right shows that the differences between eigenvalues after the fourth one are smaller than the threshold (dashed line). Thus, in this case, four dimensions will be chosen and this corresponds indeed to a break in the scree (left graph). In our experiments, the threshold is chosen using the probabilistic criterion BIC [32] which consist in minimizing:

$$\text{BIC}(m) = -2\log(L) + \nu(m)\log(n),$$

where $\nu(m)$ is the number of parameters of the model m given in Table 1 for HDDC, L is the likelihood and n is the number of observations. In the case of common intrinsic dimensions between the groups, the dimension d is directly determined using BIC. The second hyper-parameter to estimate in any clustering method is the number of groups k. A classical approach is to use BIC to select the number of groups and we follow this way in the experiments presented in Section 5.

4.4 Numerical considerations

We can observe in (3) that the cost function K_i does not use the projection on the subspace $\mathbb{E}_i^{\perp}$ and consequently does not require the estimation of the $(p - d_i)$ smallest eigenvalues of $\hat{\Sigma}_i$ and their associated eigenvectors. In addition, the computation of ML estimators of parameters b_i takes benefit of the relation

$\sum_{j=d_i+1}^{p} \lambda_{ij} = \text{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d_i} \lambda_{ij}$, for each $i = 1, ..., k$. This avoid numerical problems due to the singularity of the covariance matrix $\hat{\Sigma}_i$. Indeed, if the covariance matrix $\hat{\Sigma}_i$ is not of full rank, the smallest eigenvalues are equal to zero and the corresponding eigenvectors are ill-estimated. Furthermore, the models Full-GMM and Com-GMM need to invert the covariance matrices $\hat{\Sigma}_i$ which usually fails because of the singularity of the matrices. It exists nevertheless some solutions, as the ones presented in [21], to overcome this problem for the models Full-GMM and Com-GMM. In contrast, this problem does not arise with HDDC since the M step, based on the cost function K_i, does not require the inversion of the covariance matrices $\hat{\Sigma}_i$. Finally, for each class, the d_i first eigenvalues and their associated eigenvectors are computed using the Arnoldi method [22] which provides only the largest eigenvalues and the associated eigenvectors of a ill-conditioned matrix.

5 Experimental results

In this section, we present results for artificial and real datasets illustrating the main features of HDDC. In the following experiments, HDDC will be compared to 3 classical Gaussian mixture models: GMM with full covariance matrices for each class (Full-GMM), with diagonal covariance matrices (Diag-GMM), with spherical covariance matrices (Sphe-GMM). A numerical regularization was necessary to invert the covariance matrices in the clustering method associated to the model Full-GMM, so that it is able to work with data of dimension larger than 50.

5.1 Simulation study: model selection

Given that HDDC is a model-based clustering method, the well-known criterion BIC can be used for selecting the best adapted model to the data. Here, we used BIC and the cluster recognition rate to compare the different models of HDDC. The cluster recognition rate can be computed since we know the true partitions and is the maximum rate over the correct matchings between the true groups and the found clusters. It is impossible to report in this section numerical experiments for all the discussed models. Therefore, we limit ourselves to models

Simulated data model	HDDC model					
	$[a_{ij}b_iQ_id_i]$	$[a_{ij}bQ_id_i]$	$[a_ib_iQ_id_i]$	$[a_ibQ_id_i]$	$[ab_iQ_id_i]$	$[abQ_id_i]$
$[a_{ij}b_iQ_id_i]$	357	373	**349**	359	**349**	360
$[a_{ij}bQ_id_i]$	403	404	397	**396**	397	397
$[a_ib_iQ_id_i]$	389	419	**377**	391	**377**	394
$[a_ibQ_id_i]$	438	440	**419**	**419**	420	420
$[ab_iQ_id_i]$	399	433	**380**	402	384	403
$[abQ_id_i]$	456	451	428	**427**	434	433

Table 2: BIC value for the HDDC models on different simulated datasets (the best ones are in bold).

Simulated data model	HDDC model					
	$[a_{ij}b_iQ_id_i]$	$[a_{ij}bQ_id_i]$	$[a_ib_iQ_id_i]$	$[a_ibQ_id_i]$	$[ab_iQ_id_i]$	$[abQ_id_i]$
$[a_{ij}b_iQ_id_i]$	0.967	0.828	0.973	0.919	**0.975**	0.903
$[a_{ij}bQ_id_i]$	0.730	0.727	0.779	**0.782**	0.758	0.751
$[a_ib_iQ_id_i]$	0.979	0.871	0.983	0.929	**0.986**	0.917
$[a_ibQ_id_i]$	0.826	0.800	**0.882**	0.863	0.875	0.865
$[ab_iQ_id_i]$	0.965	0.825	**0.980**	0.844	0.952	0.822
$[abQ_id_i]$	0.712	0.752	**0.797**	0.793	0.711	0.707

Table 3: Cluster recognition rate for the HDDC models on different simulated datasets (the best ones are in bold).

with free orientations since we believe that these models are able to tackle different situations. We performed extensive simulations (50 replications for each of the 6 data models) and then used the 6 different models with free orientations in HDDC to cluster the simulated data. For each dataset, we simulated 3 Gaussian densities in $\mathbb{R}^{100}$ according to one of the 6 models with free orientations, *i.e.* free matrices Q_i, and with the following parameters: $\{d_1, d_2, d_3\} = \{2, 5, 10\}$, $\{\pi_1, \pi_2, \pi_3\} = \{0.4, 0.3, 0.3\}$ and close means and random matrices Q_i. Each one of the 6 datasets was made of 1000 points. Tables 2 and 3 present respectively the BIC value and the cluster recognition rate on average for the 6 considered HDDC models on the different simulated datasets. First of all, it appears that BIC and the cluster recognition rate select in general the same models and this confirm that BIC is a useful tool in model-based clustering. Not surprisingly, the models used to simulate the data obtain small BIC values and satisfying cluster recognition rates. However, it appears that the model $[a_ib_iQ_id_i]$ is usually selected by BIC as the best model and its cluster recognition rates are very good for each type of simulated data. Thus, the model $[a_ib_iQ_id_i]$ seems to have the right

Nb of groups k	Dimensions d_i	BIC value
2	2,16	414
3	**2,5,10**	**407**
4	2,2,5,10	414
5	2,5,5,10,12	416
6	2,5,6,10,10,12	424

Table 4: Selection of the number of groups using BIC with the model $[a_i b_i Q_i d_i]$ of HDDC: data are made of 3 groups with intrinsic dimensions $d_i = \{2, 5, 10\}$.

number of degrees of freedom and the assumptions that Δ_i has only 2 different eigenvalues is an efficient way to regularize the estimation of Δ_i. Note that the models $[a_i b Q_i d_i]$ and $[a b_i Q_i d_i]$ are also often selected by BIC and provide good cluster recognition rates.

5.2 Simulation study: hyper-parameters selection

Here, we are interested in the selection of the number of groups and of the intrinsic dimension of the clusters. In this experiment, we simulated 3 Gaussian densities in $\mathbb{R}^{100}$ according to the model $[a_i b_i Q_i d_i]$ with the following parameters: $\{d_1, d_2, d_3\} = \{2, 5, 10\}$, $\{\pi_1, \pi_2, \pi_3\} = \{0.4, 0.3, 0.3\}$, $\{a_1, a_2, a_3\} = \{150, 100, 75\}$, $\{b_1, b_2, b_3\} = \{15, 15, 15\}$, close means and random matrices Q_i. The dataset was made of 1000 points. Table 4 presents the choices of group intrinsic dimensions for the different values of k and the corresponding BIC values. First of all, it appears that the criterion BIC can be successfully used for choosing the number of clusters as in standard Gaussian mixture models. Indeed, we computed the BIC value associated to the model $[a_i b_i Q_i d_i]$ for different values of k, the number of groups, and BIC indicates that the most likely value is $k = 3$ which is correct. In addition, the intrinsic dimensions d_i, estimated by HDDC for $k = 3$, are indeed the ones of the simulated data. It is also interesting to observe the evolution of the estimation of dimensions d_i according to the number of clusters. For instance, if we consider the case of a mixture of 2 Gaussian densities, HDDC seems to correctly fit the first 2-dimensional cluster and create a second cluster made of the two other real groups. In addition, the estimated dimension of this second cluster is approximately the sum of the intrinsic dimensions of the

two real groups. Similarly, for $k = 4$, HDDC divides the first real group into two new clusters with intrinsic dimensions equal to 2. We can conclude that our approach for dimension estimation works well and allows to correctly identify the cluster subspaces.

5.3 Simulation study: influence of the dimensionality

In this paragraph, we highlight the dimensionality effect on the different clustering methods. For this last simulation study, we simulated 3 Gaussian densities in $\mathbb{R}^p$, $p = 20, ..., 100$, according to the model $[a_i b_i Q_i d_i]$ with the same parameters as in the previous experiment. The performance of methods is measured by the average cluster recognition rate computed on 50 replications. The studied clustering methods were initialized using the same random partition. Figures 3 and 4 respectively show the influence of the dimensionality on the BIC value and the cluster recognition rate for different Gaussian mixture models: model $[a_i b_i Q_i d_i]$ of HDDC, Full-GMM, Diag-GMM and Sphe-GMM. It is not surprising to observe on Figure 3 that BIC selects the model $[a_i b_i Q_i d_i]$ as the best model since the data are simulated according to this model. However, it interesting to remark that, the more the dimension increases, the larger the difference between the BIC values of the different models is, and that in favour of the model $[a_i b_i Q_i d_i]$. Figure 4 shows that data dimensionality does not influence the performance of HDDC which is very close to the performance of the Bayes decision rule (computed with the true densities). In addition, HDDC provides a cluster recognition rate similar to Full-GMM in low dimensions. Full-GMM is known to be very sensitive to the data dimensionality and, indeed, gives bad results as soon as the dimension increases. The models Diag-GMM and Sphe-GMM cannot correctly fit the data since they are too parsimonious for this complex dataset. However, we can observe that Sphe-GMM is not sensitive to the data dimensionality whereas Diag-GMM is. To summarize, HDDC is not sensitive to the dimensionality and works very well both in low and in high-dimensional spaces. In addition, the model $[a_i b_i Q_i d_i]$ outperforms models requiring a higher number of parameters (Full-GMM) and models requiring a smaller number of parameters (Diag-GMM and Sphe-GMM).

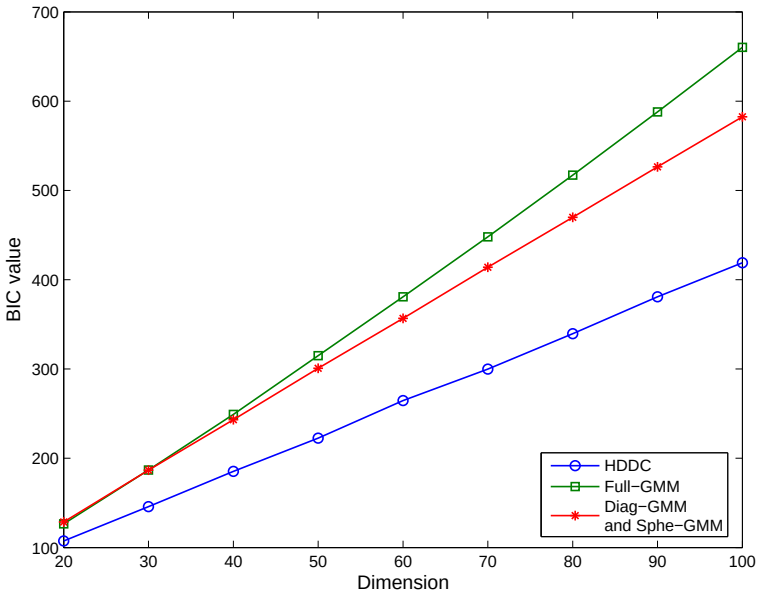

Figure 3: Influence of the dimensionality on the BIC value for different Gaussian mixture models.

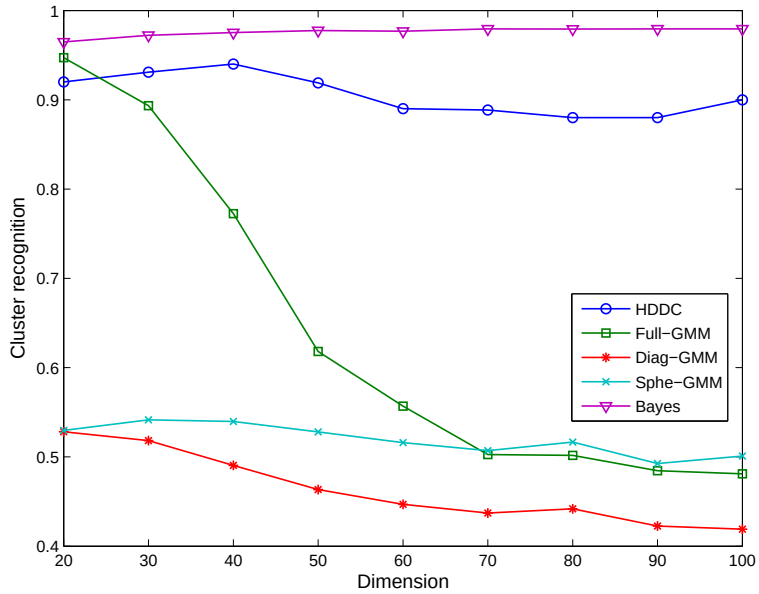

Figure 4: Influence of the dimensionality on cluster recognition rate for different Gaussian mixture models.

5.4 Real data study: comparison with variable selection

In this experiment, we compare HDDC with the method of variable selection for model-based clustering presented by Raftery *et al.* in [28], denoted by VS-GMM in the following. The authors considered the variable selection problem as a model selection problem. Selection is made using approximate Bayes factors and combined with a greedy search algorithm. In addition, it is possible to perform this variable selection on the original variables, but also on the principal components using PCA as a pre-processing step. In order to compare HDDC to this variable selection technique, we used the same dataset as in [28]. The Leptograpsus crabs dataset consists of 200 subjects equally distributed into 4 classes: Orange Male, Orange Female, Blue Male and Blue Female. There are 5 variables for each subject: width of frontal lip (FL), rear width (RW), length along the mid-line of the carapace (CL), maximum of the width of the carapace (CW) and body depth (BD) in mm. The left panel of Figure 5 shows the Crabs data projected on the two first principal axes and the big circles represent the cluster means.

Table 5 gives the classification error rate for the classical model Sphe-GMM, the VS-GMM method and HDDC. The second column of this table indicates on which variables is performed the clustering. HDDC obtains a cluster recognition rate equal to 95% and the variable selection method of Raftery *et al.* obtains 93.5% whereas the classical model Sphe-GMM obtains a cluster recognition rate equal to 60.5%. HDDC found that each cluster lives in a 1-dimensional subspace embedded into the original 5-dimensional space. The right panel of Figure 5 shows the specific subspaces (blue lines) of the 4 mixture components obtained with the model $[a_i b_i Q_i d_i]$ of HDDC. For this illustration, we chose to project the data on the two first principal components since results obtained with VS-GMM on these variables are better than on the original ones. We can observe that the specific axes of the different clusters are very correlated and this explains that HDDC provides a better clustering result than the variable selection method VS-GMM. Figure 6 presents the 12 steps of the EM algorithm in order to show the evolution of the group-specific subspaces into HDDC. We can conclude that HDDC is able to fit the clusters in their specific subspaces whereas the other methods select only the best dimensions for clustering.

Model	Variables	Cluster recognition rate
Sphe-GMM	Original	0.605
VS-GMM	Original	0.925
Sphe-GMM	Princ. comp.	0.605
VS-GMM	Princ. comp.	0.935
HDDC $[a_i b_i Q_i d_i]$	Original	**0.950**

Table 5: Classification results for the Crabs data: comparison of different model-based clustering methods.

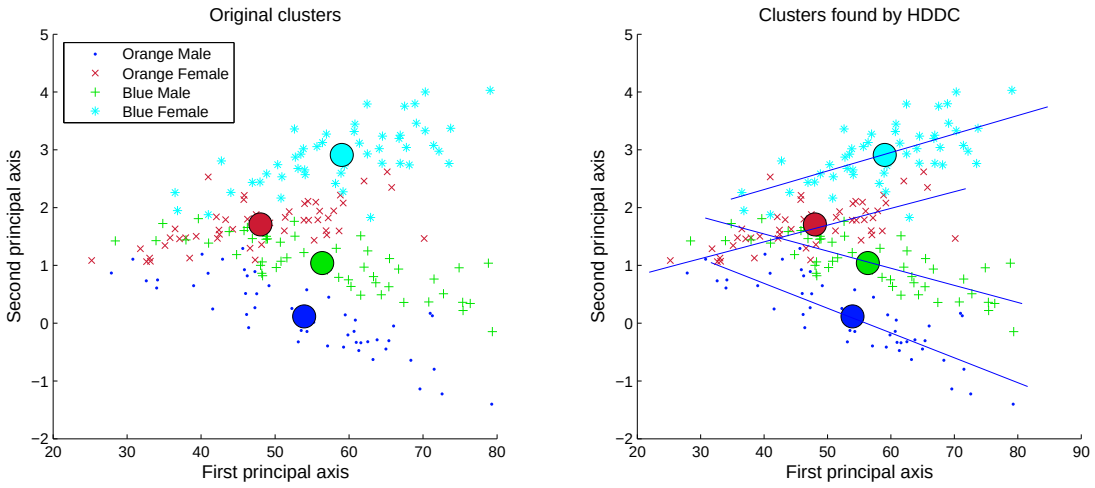

Figure 5: Clustering results using HDDC: on the left panel, crabs data projected on the two first principal axes and, on the right panel, clustering result obtained with the model $[a_i b_i Q_i d_i]$ of HDDC and the estimated specific subspaces of the mixture components (blue lines).

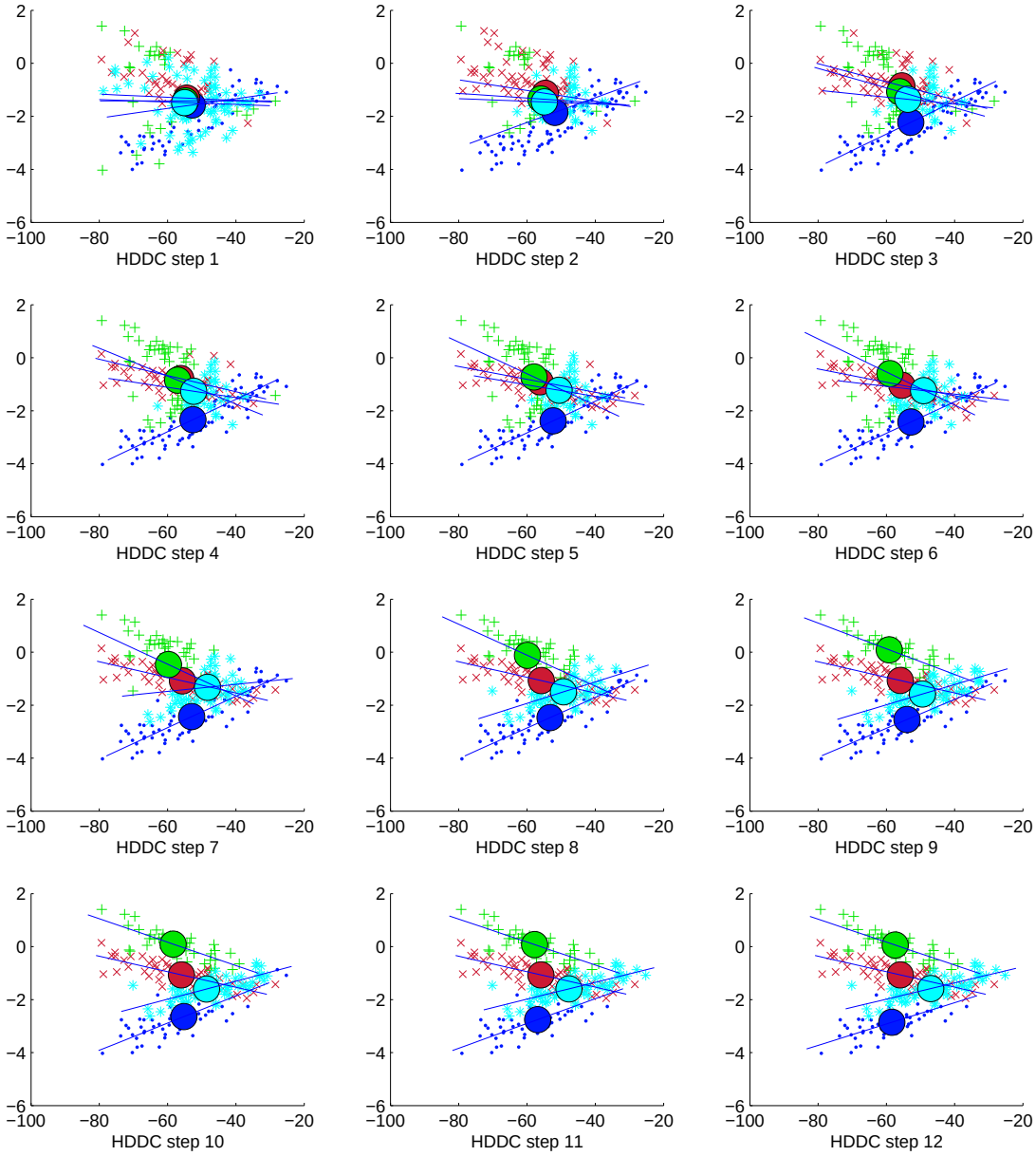

Figure 6: The steps of the EM-based algorithm HDDC on the Crabs dataset and the estimated specific subspaces of the mixture components (blue lines).

5.5 Real data study: Martian surface characterization

Here, we propose to use HDDC to analyze and segment images of the Martian surface. Visible and near infrared imaging spectroscopy is a key remote sensing technique to study and monitor the system of the planets. Imaging spectrometers, which are inboard of an increasing number of satellites, provide high-dimensional hyper-spectral images. Constant technological improvements promote the acquisition of dramatically expanding image collections. In March 2004, the OMEGA instrument (Mars Express, ESA) [4] has collected 310 Gbytes of raw images. The OMEGA imaging spectrometer has mapped the Martian surface with a spatial resolution varying between 300 to 3000 meters depending on spacecraft altitude. It acquires for each resolved pixel the spectrum from 0.36 to 5.2 μm in 256 contiguous spectral channels. OMEGA is designed to characterize the composition of surface materials, discriminating between various classes of silicates, hydrated minerals, oxides and carbonates, organic frosts and ices. For this experiment, we considered a 300×128 image of the Martian surface and a 256-dimensional spectral observation is associated to each of the 38 400 pixels. The image of the studied zone is presented on the left panel of Figure 7. According to the experts, there are $k = 5$ mineralogical classes to identify.

The right image of Figure 7 shows the segmentation obtained with the model $[a_i b_i Q_i d_i]$ of HDDC. First of all, we can observe that the segmentation of HDDC is very precise on the major part of the image. The poor results of the top right part of the image are due to the planet curvature and could be corrected. In particular, the experts of the domain appreciated that our method is able to detect that a melange of ice and carbonate is present around the ice zones (clear zones of the image). Figure 8 shows the spectral means of the 5 classes and this allows the experts to determine the mineralogical and molecular composition of each class. We remind that this study is done without taking into account the spatial relations between the pixels of a image. A natural extension of this work is therefore to combine HDDC with the modeling of the spatial relations using, for example, hidden Markov random fields. This experiment demonstrates that HDDC can be efficiently used on real high-dimensional data and with large datasets. In addition, a major interest of the use of HDDC for this application is to be able to provide posterior probabilities that each pixel belongs to the classes.

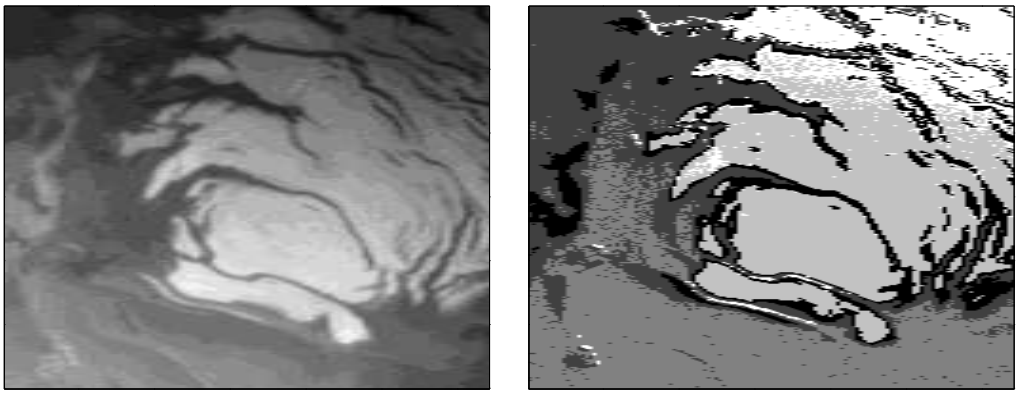

Figure 7: Characterization of the Martian surface composition using HDDC: on the left, image of the studied zone and, on the right, segmentation using HDDC on the 256-dimensional spectral data associated to the image.

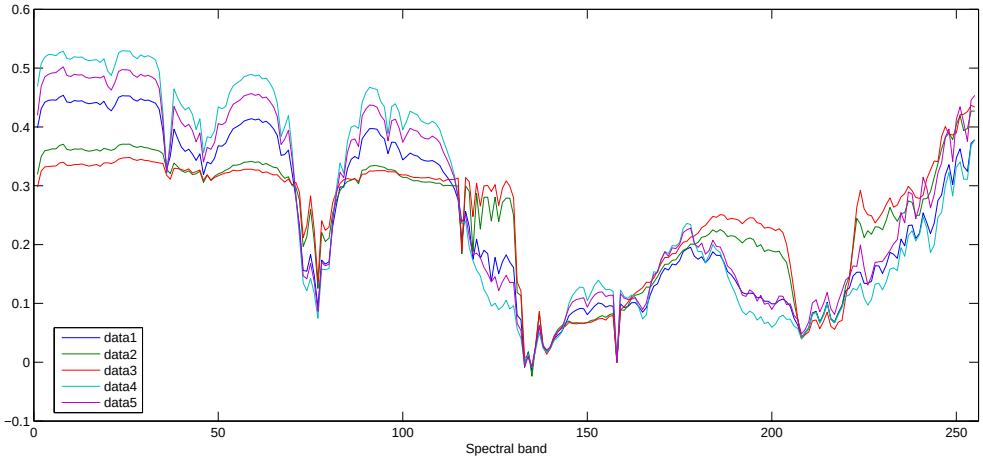

Figure 8: Spectral means of the 5 mineralogical classes found using HDDC.

6 Conclusion

We introduced in this paper a family of Gaussian mixture models designed for high-dimensional data which assume that the intrinsic dimensionality of each mixture component is much smaller than the dimensionality of the original space. In addition, the proposed models assume that outside the specific subspace of each group the variance is modeled by a single parameter and corresponds to the noise. It is also possible to add constraints on the parameters within or between the groups and this allows to obtain several regularized models. This parameterization in the eigenspaces of the mixture components gives rise to an EM-based clustering method, called High-Dimensional Data Clustering (HDDC). Experiments on artificial and real datasets demonstrated the effectiveness of the different model of HDDC compared to classical Gaussian mixture models. In particular, the model $[a_i b_i Q_i d_i]$ provides very satisfying results for many types of data.

A Appendix: parameters estimation

First of all, we introduce the following useful formulation of the log-likelihood:

$$-2\log(L) = \sum_{i=1}^{k} n_i \sum_{j=1}^{p} \left(\log(\delta_{ij}) + \frac{1}{\delta_{ij}} q_{ij}^t \hat{\Sigma}_i q_{ij} \right) + c^{st}, \tag{14}$$

where δ_{ij} is the jth diagonal coefficient of Δ_i and q_{ij} is the jth column of Q_i. We refer to [11] for a demonstration of this result.

A.1 Models with free orientations

Subspace $\mathbb{E}_i$: The log-likelihood is to be maximized under the constraint $q_{ij}^t q_{ij} = 1$, which is equivalent to finding a saddle point of the Lagrange function:

$$\mathcal{L} = -2\log(L) - \sum_{j=1}^{p} \theta_{ij}(q_{ij}^t q_{ij} - 1),$$

where θ_{ij} are the Lagrange multipliers. Using the expression (14) of the log-likelihood, the gradient of $\mathcal{L}$ with respect to q_{ij} is:

$$\nabla_{q_{ij}}\mathcal{L} = 2\frac{n_i}{\delta_{ij}}\hat{\Sigma}_i q_{ij} - 2\theta_{ij}q_{ij},$$

and by multiplying this quantity on the left by q_{ij}^t, we obtain:

$$q_{ij}^t\nabla_{q_{ij}}\mathcal{L} = 0 \Leftrightarrow \theta_{ij} = \frac{n_i}{\delta_{ij}}q_{ij}^t\hat{\Sigma}_i q_{ij}.$$

Consequently, $\hat{\Sigma}_i q_{ij} = \frac{\theta_{ij}\delta_{ij}}{n_i}q_{ij}$ and thus q_{ij} is the eigenvector of $\hat{\Sigma}_i$ associated with the eigenvalue $\lambda_{ij} = \frac{\theta_{ij}\delta_{ij}}{n_i} = q_{ij}^t\hat{\Sigma}_i q_{ij}$. As the vectors q_{ij} are eigenvectors of the symmetric matrix $\hat{\Sigma}_i$, this implies that $q_{ij}^t q_{i\ell} = 0$ if $j \neq \ell$. The log-likelihood can therefore be re-written as follows:

$$-2\log(L) = \sum_{i=1}^{k} n_i \left(\sum_{j=1}^{d_i} \left(\log(a_{ij}) + \frac{\lambda_{ij}}{a_{ij}} \right) + \sum_{j=d_i+1}^{p} \left(\log(b_i) + \frac{\lambda_{ij}}{b_i} \right) \right) + c^{st},$$

and, using the relation $\sum_{j=d_i+1}^{p}\lambda_{ij} = \mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d_i}\lambda_{ij}$, we obtain:

$$-2\log(L) = \sum_{i=1}^{k} n_i \left(\sum_{j=1}^{d_i} \log(a_{ij}) + (p-d_i)\log(b_i) + \frac{\mathrm{Tr}(\hat{\Sigma}_i)}{b_i} + \sum_{j=1}^{d_i} \left(\frac{1}{a_{ij}} - \frac{1}{b_i} \right) \lambda_{ij} \right) + c^{st}. \tag{15}$$

Thus, minimizing $-2\log(L)$ with respect to λ_{ij} is equivalent to minimizing the quantity $\sum_{i=1}^{k} n_i \sum_{j=1}^{d_i}(\frac{1}{a_{ij}} - \frac{1}{b_i})\lambda_{ij}$. Since $(\frac{1}{a_{ij}} - \frac{1}{b_i}) < 0$, $\forall j = 1, ..., d_i$, λ_{ij} must therefore be as larger as possible. Thus, the column vector q_{ij}, $\forall j = 1, ..., d_i$, is estimated by the eigenvector associated to the jth largest eigenvalue of $\hat{\Sigma}_i$.

Model $[a_{ij}b_iQ_id_i]$**:** starting from equation (15), the partial derivative of $-2\log(L)$ with respect to a_{ij} and b_i are:

$$-2\frac{\partial\log(L)}{\partial a_{ij}} = n_i\left(\frac{1}{a_{ij}} - \frac{\lambda_{ij}}{a_{ij}^2}\right) \quad \text{and} \quad -2\frac{\partial\log(L)}{\partial b_i} = \frac{n_i(p-d_i)}{b_i} - \frac{n_i}{b_i^2}\left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d_i}\lambda_{ij}\right).$$

The condition $\frac{\partial \log(L)}{\partial a_{ij}} = 0$ implies that $\hat{a}_{ij} = \lambda_{ij}$ and the condition $\frac{\partial \log(L)}{\partial b_i} = 0$ implies that:

$$\hat{b}_i = \frac{1}{(p - d_i)} \left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d_i} \lambda_{ij} \right).$$

Model $[a_{ij} b Q_i d_i]$**:** the partial derivative of $-2\log(L)$ with respect to b is:

$$-2\frac{\partial \log(L)}{\partial b} = \frac{n(p-\xi)}{b} - \frac{1}{b^2} \sum_{i=1}^{k} n_i \left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d_i} \lambda_{ij} \right),$$

and the condition $\frac{\partial \log(L)}{\partial b} = 0$ proves that:

$$\hat{b} = \frac{1}{(p-\xi)} \left(\mathrm{Tr}(\hat{W}) - \sum_{i=1}^{k} \hat{\pi}_i \sum_{j=1}^{d_i} \lambda_{ij} \right).$$

Model $[a_i b_i Q_i d_i]$**:** from (15), the partial derivative of $-2\log(L)$ with respect to a_i is:

$$-2\frac{\partial \log(L)}{\partial a_i} = \frac{n_i d_i}{a_i} - \frac{n_i}{a_i^2} \sum_{j=1}^{d_i} \lambda_{ij},$$

and the condition $\frac{\partial \log(L)}{\partial a_i} = 0$ implies that:

$$\hat{a}_i = \frac{1}{d_i} \sum_{j=1}^{d_i} \lambda_{ij}.$$

Model $[a b_i Q_i d_i]$**:** the partial derivative of $-2\log(L)$ with respect to a is:

$$-2\frac{\partial \log(L)}{\partial a} = \frac{n\xi}{a} - \frac{1}{a^2} \sum_{i=1}^{k} n_i \sum_{j=1}^{d_i} \lambda_{ij},$$

and the condition $\frac{\partial \log(L)}{\partial a} = 0$ gives:

$$\hat{a} = \frac{1}{\xi} \sum_{i=1}^{k} \hat{\pi}_i \sum_{j=1}^{d_i} \lambda_{ij}.$$

Model $[a_j b_i Q_i d]$**:** the partial derivative of $-2\log(L)$ with respect to a_j is:

$$-2\frac{\partial \log(L)}{\partial a_j} = \frac{n}{a_j} - \frac{1}{a_j^2}\sum_{i=1}^{k} n_i \lambda_{ij}.$$

The condition $\frac{\partial \log(L)}{\partial a_j} = 0$ and the relation $\sum_{i=1}^{k} n_i \lambda_{ij} = n\lambda_j$ imply that $\hat{a}_j = \lambda_j$.

A.2 Models with common orientations

Subspace $\mathbb{E}_i$**:** Starting from the likelihood expression (14), we can write:

$$\begin{aligned} -2\log(L) &= \sum_{i=1}^{k} n_i \sum_{j=1}^{d} \left(\log(a_i) + \frac{1}{a_i} q_j^t \hat{\Sigma}_i q_j \right) + \sum_{i=1}^{k} n_i \sum_{j=d+1}^{p} \left(\log(b_i) + \frac{1}{b_i} q_j^t \hat{\Sigma}_i q_j \right) + c^{st}, \\ &= \sum_{i=1}^{k} n_i \left(d\log(a_i) + (p-d)\log(b_i) \right) + \sum_{j=1}^{d} q_j^t A q_j + \sum_{j=d+1}^{p} q_j^t B q_j + c^{st}, \end{aligned}$$

where $A = \sum_{i=1}^{k} \frac{n_i}{a_i} \hat{\Sigma}_i$ and $B = \sum_{i=1}^{k} \frac{n_i}{b_i} \hat{\Sigma}_i$. Note that $\sum_{j=d+1}^{p} q_j^t B q_j$ can be written using the trace of B: $\sum_{j=d+1}^{p} q_j^t B q_j = \mathrm{Tr}(B) - \sum_{j=1}^{d} q_j^t B q_j$. This yields:

$$-2\log(L) = \sum_{i=1}^{k} n_i \left(d\log(a_i) + (p-d)\log(b_i) \right) - \sum_{j=1}^{d} q_j^t (B-A) q_j + \mathrm{Tr}(B) + c^{st} \quad (16)$$

Consequently, the gradient of $\mathcal{L} = -2\log(L) - \sum_{j=1}^{p} \theta_j (q_j^t q_j - 1)$ with respect to q_j is:

$$\nabla_{q_j} \mathcal{L} = -2(B-A)q_j - 2\theta_j q_j,$$

where θ_j is the jth Lagrange multiplier. The relation $\nabla_{q_j} \mathcal{L} = 0$ is equivalent to $(B-A)q_j = -\theta_j q_j$ which means that q_j is eigenvector of the matrix $(B-A)$. In order to minimize the quantity $-2\log(L)$, the d first columns of Q must be the eigenvectors associated with the d largest eigenvalues of $(B-A)$.

Model $[a_i b_i Q d]$**:** Starting from equation (16), the partial derivatives of $-2\log(L)$ with respect to a_i and b_i are:

$$-2\frac{\partial \log(L)}{\partial a_i} = \frac{n_i d}{a_i} - \frac{n_i}{a_i^2}\sum_{j=1}^{d} q_j^t \hat{\Sigma}_i q_j \text{ and } -2\frac{\partial \log(L)}{\partial b_i} = \frac{n_i(p-d)}{b_i} - \frac{n_i}{b_i^2}\left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d} q_j^t \hat{\Sigma}_i q_j\right).$$

The condition $\frac{\partial \log(L)}{\partial a_i} = 0$ and $\frac{\partial \log(L)}{\partial b_i} = 0$ give respectively:

$$\hat{a}_i(Q) = \frac{1}{d}\sum_{j=1}^{d} q_j^t \hat{\Sigma}_i q_j \quad \text{and} \quad \hat{b}_i(Q) = \frac{1}{(p-d)}\left(\mathrm{Tr}(\hat{\Sigma}_i) - \sum_{j=1}^{d} q_j^t \hat{\Sigma}_i q_j\right).$$

Model $[a_i b Q d]$**:** The partial derivative of $-2\log(L)$ with respect to b is:

$$-2\frac{\partial \log(L)}{\partial b} = \frac{n(p-d)}{b} - \frac{n}{b^2}\left(\mathrm{Tr}(\hat{W}) - \sum_{j=1}^{d} q_j^t \hat{W} q_j\right),$$

and the condition $\frac{\partial \log(L)}{\partial b} = 0$ implies that:

$$\hat{b}(Q) = \frac{1}{(p-d)}\left(\mathrm{Tr}(\hat{W}) - \sum_{j=1}^{d} q_j^t \hat{W} q_j\right).$$

Model $[a b_i Q d]$**:** The partial derivative of $-2\log(L)$ with respect to a is:

$$-2\frac{\partial \log(L)}{\partial a} = \frac{nd}{a} - \frac{n}{a^2}\sum_{j=1}^{d} q_j^t \hat{W} q_j,$$

and the condition $\frac{\partial \log(L)}{\partial a} = 0$ proves that:

$$\hat{a}(Q) = \frac{1}{d}\sum_{j=1}^{d} q_j^t \hat{W} q_j.$$

A.3 Models with common covariance matrices

Subspace $\mathbb{E}_i$: The log-likelihood can be written as follows:

$$-2\log(L) = n\left(\sum_{j=1}^{d}\log(a_j) + (p-d)\log(b) + \frac{\mathrm{Tr}(\hat{W})}{b} + \sum_{j=1}^{d}\left(\frac{1}{a_j} - \frac{1}{b}\right)q_j^t\hat{W}q_j\right) + c^{st}.$$

The gradient of $\mathcal{L} = -2\log(L) - \sum_{j=1}^{p}\theta_j(q_j^tq_j - 1)$ with respect to q_j is:

$$\nabla_{q_j}\mathcal{L} = 2n(\frac{1}{a_j} - \frac{1}{b})\hat{W}q_j - 2\theta_j q_j,$$

where θ_j is the jth Lagrange multiplier. The relation $\nabla_{q_j}\mathcal{L} = 0$ implies that $q_j$is eigenvector of $\hat{W}$. In order to minimize $-2\log(L)$, the first columns of Q must be the eigenvectors associated to the d largest eigenvalues of $\hat{W}$.

Model $[a_jbQd]$: The partial derivatives of $-2\log(L)$ with respect to a_j and b are:

$$-2\frac{\partial\log(L)}{\partial a_j} = \frac{n}{a_j} - \frac{n}{a_j^2}q_j^t\hat{W}q_j \quad \text{and} \quad -2\frac{\partial\log(L)}{\partial b} = \frac{n(p-d)}{b} - \frac{n}{b^2}\sum_{j=d+1}^{p}q_j^t\hat{W}q_j.$$

The condition $\frac{\partial\log(L)}{\partial a_i} = 0$ implies that $\hat{a}_j = \lambda_j$. The combination of the condition $\frac{\partial\log(L)}{\partial b} = 0$ with the relation $\sum_{j=d+1}^{p}\lambda_j = \mathrm{Tr}(\hat{W}) - \sum_{j=1}^{d}\lambda_j$ gives the estimator of b:

$$\hat{b} = \frac{1}{(p-d)}\left(\mathrm{Tr}(\hat{W}) - \sum_{j=1}^{d}\lambda_j\right).$$

Model $[abQd]$: The partial derivatives of $-2\log(L)$ with respect to a is:

$$-2\frac{\partial\log(L)}{\partial a} = \frac{nd}{a} - \frac{n}{a^2}\sum_{j=1}^{d}q_j^t\hat{W}q_j,$$

and the condition $\frac{\partial\log(L)}{\partial a} = 0$ implies that:

$$\hat{a} = \frac{1}{d}\sum_{j=1}^{d}\lambda_j.$$

References

[1] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high-dimensional data for data mining application. In *ACM SIGMOD International Conference on Management of Data*, pages 94–105, 1998.

[2] J. Banfield and A. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49:803–821, 1993.

[3] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[4] J.-P. Bibring and 42 co authors. *OMEGA: Observatoire pour la Minéralogie, l'Eau, les Glaces et l'Activité*, page 37 49. ESA SP-1240: Mars Express: the Scientific Payload, 2004.

[5] L. Bocci, D. Vicari, and M. Vichi. A mixture model for the classification of three-way proximity data. *Computational Statistics and Data Analysis*, 50(7):1625 1654, 2006.

[6] R. Cattell. The scree test for the number of factors. *Multivariate Behavioral Research*, 1(2):245–276, 1966.

[7] G. Celeux and G. Govaert. A classification EM algorithm for clustering and two stochastic versions. *Computational Statistics and Data Analysis*, 14:315–332, 1992.

[8] G. Celeux and G. Govaert. Gaussian parsimonious clustering models. *International Journal of Pattern Recognition*, 28(5):781–793, 1995.

[9] P. Demartines and J. Hérault. Curvilinear Component Analysis: a self-organizing neural network for non linear mapping of data sets. *IEEE Transactions on Neural Networks*, 8(1):148–154, 1997.

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

[11] B. Flury. Common principal components in k groups. *Journal of American Statistical Association*, 79:892–897, 1984.

[12] B. Flury and W. Gautschi. An algorithm for simultaneous orthogonal transformation of several positive definite symmetric matrices to nearly diagonal form. *SIAM Journal on Scientific and Statistical Computing*, 7:169–184, 1986.

[13] L. Flury, B. Boukai, and B. Flury. The discrimination subspace model. *Journal of American Statistical Association*, 92(438):758–766, 1997.

[14] C. Fraley and A. Raftery. Model-based clustering, discriminant analysis and density estimation. *Journal of American Statistical Association*, 97:611–631, 2002.

[15] S. Girard. A nonlinear PCA based on manifold approximation. *Computational Statistics*, 15(2):145–167, 2000.

[16] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[17] T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.

[18] A. Jain, M. Marty, and P. Flynn. Data Clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

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

[20] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, New York, 1995.

[21] W. Krzanowski, P. Jonathan, W. McCarthy, and M. Thomas. Discriminant analysis with singular covariance matrices: methods and applications in spectroscopic data. *Journal of Applied Statistics*, 44:101–115, 1995.

[22] R. Lehoucq, D. Sorensen, and C. Yang. *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM Publications, Philadelphia, 1998.

[23] G. McLachlan and T. Krishnan. *The EM algorithm and extensions*. Wiley Interscience, New York, 1997.

[24] G. McLachlan and D. Peel. *Finite Mixture Models*. Wiley Interscience, New York, 2000.

[25] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: a review. *SIGKDD Explor. Newsl.*, 6(1):90–105, 2004.

[26] T. Pavlenko. On feature selection, curse of dimensionality and error probability in discriminant analysis. *Journal of Statistical Planning and Inference*, 115:565–584, 2003.

[27] T. Pavlenko and D. Von Rosen. Effect of dimensionality on discrimination. *Statistics*, 35(3):191–213, 2001.

[28] A. Raftery and N. Dean. Variable Selection for Model-Based Clustering. *Journal of American Statistical Association*, 101(473):168–178, 2006.

[29] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[30] B. Schölkopf, A. Smola, and K. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.

[31] J. Schott. Dimensionality reduction in quadratic discriminant analysis . *Computational Statistics and Data Analysis*, 66:161–174, 1993.

[32] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464, 1978.

[33] D. Scott and J. Thompson. Probability density estimation in higher dimensions. In *Fifteenth Symposium in the Interface*, pages 173–179, 1983.

[34] J. Tenenbaum, V. De Silva, and J. Langford. A global geometric framework for non linear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[35] M. Tipping and C. Bishop. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11(2):443–482, 1999.