Model-based clustering of functional data

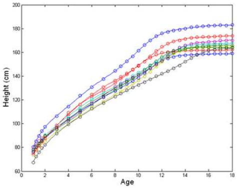
Julien JACQUES

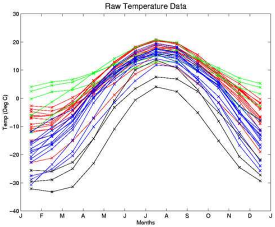
Laboratoire P. Painlevé, UMR CNRS 8524 & Université Lille 1
MODAL, INRIA Lille Nord Europe

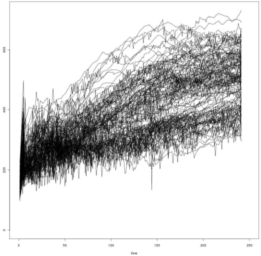
December 8th 2011

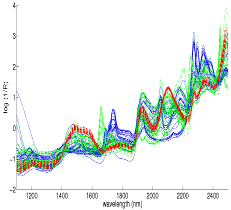
joint work with Charles BOUVEYRON (Paris 1)

Some functional data:









Clustering

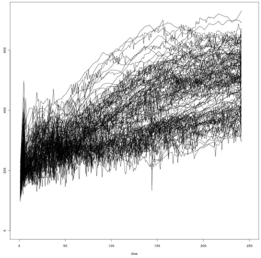
Task of assigning a set of objects into groups (*clusters*).

Objects in a cluster are more similar to each other than to those in other clusters.

Clustering

Task of assigning a set of objects into groups (*clusters*).

Objects in a cluster are more similar to each other than to those in other clusters.

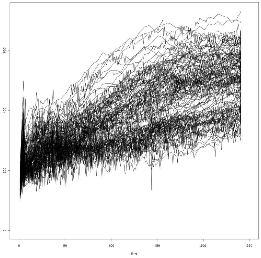


Introduction

Clustering

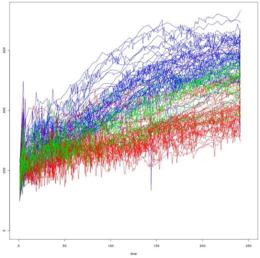
Task of assigning a set of objects into groups (*clusters*).

Objects in a cluster are more similar to each other than to those in other clusters.



$\Rightarrow$

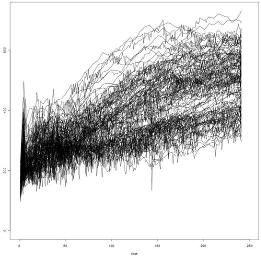
clustering



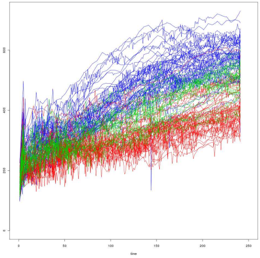
Clustering

Task of assigning a set of objects into groups (*clusters*).

Objects in a cluster are more similar to each other than to those in other clusters.



$\Rightarrow$
clustering



Clustering: unsupervised classification, data segmentation...

Parametric clustering techniques for curves are generally performed in two steps

- The discretization step aims to describe the functions in a finite dimensional space:
  - direct discretization $(X_{t_1}, \ldots, X_{t_p})$,
  - approximation of curves into a space spanned by a finite basis of functions

$$X(t) = \sum_{j=1}^{J} \alpha_j \Phi_j(t)$$

  - use of on functional principal components (FPCA),
- The clustering step usually applies a multivariate clustering technique on the discretized version of the data:
  - k-means,
  - hierarchical clustering,
  - model-based clustering.

Two steps are not satisfactory

- discretization step is done independently on the clustering task,
- how to choose between the discretization techniques and the clustering ones in a unsupervised context ?

Clustering techniques for functional data

Two steps are not satisfactory

- discretization step is done independently on the clustering task,
- how to choose between the discretization techniques and the clustering ones in a unsupervised context ?

Recent clustering techniques are designed for functional data :

- discretization depending on the clustering task
  - James & Sugar [2003]: cluster-dependent spline decomposition,
  - Bouveyron & J. [2011]: parsimonious modeling of cluster-dependent FPCA,
- approximation of the notion of density
  - J. & Preda [preprint]: model-based clustering using approximation of the notion of density for functional random variable.

Plan

1. Preliminary on model-based clustering

2. Parsimonious modeling of cluster-dependent FPCA
   - The model
   - Model inference

3. Numerical applications
   - Introductory example: Canada weather
   - Mars surface characterization

Plan

1. Preliminary on model-based clustering

2. Parsimonious modeling of cluster-dependent FPCA
   - The model
   - Model inference

3. Numerical applications
   - Introductory example: Canada weather
   - Mars surface characterization

Gaussian model-based clustering

Observed data

$\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ with $\forall 1 \leq i \leq n, \quad \boldsymbol{X}_i = (X_{i1}, \ldots, X_{ip}) \in \mathbb{R}^p$

Clustering

consists in grouping each $\boldsymbol{X}_i$ into one of the K clusters $\mathcal{G}_1, \ldots, \mathcal{G}_K$ (K *known*).

Let $\boldsymbol{Z}_i = (Z_{i1}, \ldots, Z_{iK})$ indicates the cluster belonging:

- $Z_{ik} = 1$ if $\boldsymbol{X}_i$ belongs to $\mathcal{G}_k$,
- $Z_{ik} = 0$ otherwise.

The model

Each cluster of data is assumed to arise from a p-variate Gaussian distribution

$$\boldsymbol{X}_{|\boldsymbol{Z}_k=1} \sim \mathcal{N}_p(\mu_k, \Sigma_k)$$

- marginal distribution is a mixture density

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \phi_k(\boldsymbol{x}; \mu_k, \Sigma_k)$$

  - π_k are the mixing proportions
  - $\phi_k(\cdot; \mu_k, \Sigma_k)$ is the density of $\mathcal{N}_p(\mu_k, \Sigma_k)$

- Bayes rule or *Maximum A Posteriori* rule classifies $\boldsymbol{x}$ into $\mathcal{G}_k$ maximizing:

$$t_k(\boldsymbol{x}) \propto \pi_k \phi_k(\boldsymbol{x}; \mu_k, \Sigma_k).$$

Estimation: maximum likelihood

$\theta = (\pi_k, \mu_k, \Sigma_k)_{k=1,\ldots,K}$ is estimated by maximizing the likelihood of $\underline{\boldsymbol{x}} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$

Log-likelihood

$$l(\theta, \underline{\boldsymbol{x}}) = \sum_{i=1}^{n} \ln \left(\sum_{k=1}^{K} \pi_k \phi_k(\mathbf{x}_i, \mu_k, \Sigma_k) \right).$$

Gaussian model-based clustering

Estimation: maximum likelihood

$\theta = (\pi_k, \mu_k, \Sigma_k)_{k=1,\dots,K}$ is estimated by maximizing the likelihood of $\underline{\boldsymbol{x}} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$

Log-likelihood

$$l(\theta, \underline{\boldsymbol{x}}) = \sum_{i=1}^{n} \ln \left(\sum_{k=1}^{K} \pi_k \phi_k(\mathbf{x}_i, \mu_k, \Sigma_k) \right).$$

$\Rightarrow \ln \sum$ is hard to maximize.

Gaussian model-based clustering

Estimation: maximum likelihood

$\theta = (\pi_k, \mu_k, \Sigma_k)_{k=1,\ldots,K}$ is estimated by maximizing the likelihood of $\underline{\boldsymbol{x}} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$

Log-likelihood

$$l(\theta, \underline{\boldsymbol{x}}) = \sum_{i=1}^{n} \ln \left(\sum_{k=1}^{K} \pi_k \phi_k(\mathbf{x}_i, \mu_k, \Sigma_k) \right).$$

$\Rightarrow \ln \sum$ is hard to maximize.

The maximisation will be easier if $\underline{\boldsymbol{z}} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)$ was known. Assuming $\underline{\boldsymbol{z}}$ is known, we define the completed log-likelihood:

$$l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \ln\left(\pi_k \phi_k(\mathbf{x}_i, \mu_k, \Sigma_k)\right).$$

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

But as $\underline{\boldsymbol{z}}$ is unknown, it is estimated !

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

But as $\underline{\boldsymbol{z}}$ is unknown, it is estimated !

Algorithme EM *(CEM version)*

- `Init:  randomize` $\underline{\boldsymbol{z}}$

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

But as $\underline{\boldsymbol{z}}$ is unknown, it is estimated !

Algorithme EM *(CEM version)*

- `Init: randomize` $\underline{\boldsymbol{z}}$
- `M step: compute`

$$\theta^{(h+1)} = \underset{\theta}{\operatorname{argmax}}\, l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$$

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

But as $\underline{\boldsymbol{z}}$ is unknown, it is estimated !

Algorithme EM *(CEM version)*

- Init: randomize $\underline{\boldsymbol{z}}$
- M step: compute

$$\theta^{(h+1)} = \underset{\theta}{\operatorname{argmax}}\, l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$$

- E step: estimate $\underline{\boldsymbol{z}}$ according to $\theta^{(h+1)}$

$$t_{ik} = \frac{\pi_k^{(h+1)} \phi_k(\boldsymbol{x}; \mu_k^{(h+1)}, \Sigma_k^{(h+1)})}{\sum_{k=1}^K \pi_k^{(h+1)} \phi_k(\boldsymbol{x}; \mu_k^{(h+1)}, \Sigma_k^{(h+1)})} \qquad \text{and } \hat{z}_{ik} = 1 \text{ for } k = \underset{\ell}{\operatorname{argmax}}\, t_{i\ell}$$

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

But as $\underline{\boldsymbol{z}}$ is unknown, it is estimated !

Algorithme EM *(CEM version)*

- `Init: randomize` $\underline{\boldsymbol{z}}$
- `M step: compute`

$$\theta^{(h+1)} = \operatorname*{argmax}_{\theta} l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$$

- `E step: estimate` $\underline{\boldsymbol{z}}$ `according to` $\theta^{(h+1)}$

$$t_{ik} = \frac{\pi_k^{(h+1)} \phi_k(\boldsymbol{x}; \mu_k^{(h+1)}, \Sigma_k^{(h+1)})}{\sum_{k=1}^{K} \pi_k^{(h+1)} \phi_k(\boldsymbol{x}; \mu_k^{(h+1)}, \Sigma_k^{(h+1)})} \qquad \text{and } \hat{z}_{ik} = 1 \text{ for } k = \operatorname*{argmax}_{\ell} t_{i\ell}$$

`repeat M and E steps until` $l(\hat{\theta}, \underline{\boldsymbol{x}})$ `convergence.`

Estimation - EM algorithm

The EM algorithm maximizes $l_c(\theta, \underline{\boldsymbol{x}}, \underline{\boldsymbol{z}})$ rather than $l(\theta, \underline{\boldsymbol{x}})$.

But as $\underline{\boldsymbol{z}}$ is unknown, it is estimated !

Algorithme EM

- `Init: randomize` $\underline{\boldsymbol{z}}$
- `M step: compute`

$$\theta^{(h+1)} = \underset{\theta}{\operatorname{argmax}} E_{\theta^{(h)}}[l_c(\theta, \underline{\boldsymbol{X}}, \underline{\boldsymbol{Z}}) | \underline{\boldsymbol{X}} = \underline{\boldsymbol{x}}]$$

where $\theta^{(h)}$ is the estimation of θ at this step of the algo.

- `E step: compute` $E_{\theta^{(h)}}[\underline{\boldsymbol{z}}]$ `according to` $\theta^{(h+1)}$

$$\hat{z}_{ik} = t_{ik} = \frac{\pi_k^{(h+1)} \phi_k(\boldsymbol{x}; \mu_k^{(h+1)}, \Sigma_k^{(h+1)})}{\sum_{k=1}^K \pi_k^{(h+1)} \phi_k(\boldsymbol{x}; \mu_k^{(h+1)}, \Sigma_k^{(h+1)})}.$$

`repeat M and E steps until` $l(\hat{\theta}, \underline{\boldsymbol{x}})$ `convergence.`

Estimation - selection of the number K of clusters

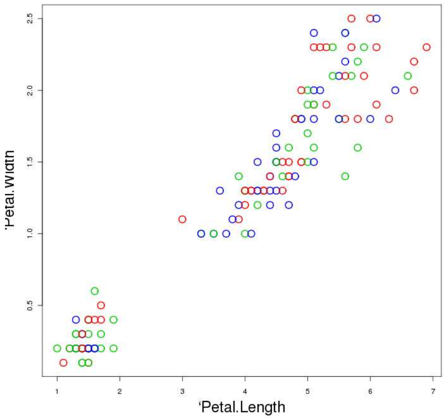
We can use a penalized likelihood criterion :

$$BIC = -2l(\hat{\theta}) + \nu \ln n$$

where ν is the number of model parameters.

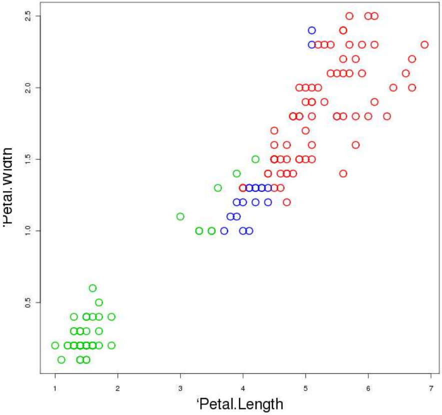
Estimation - illustration

Example of the EM convergence on the famous *iris* dataset.



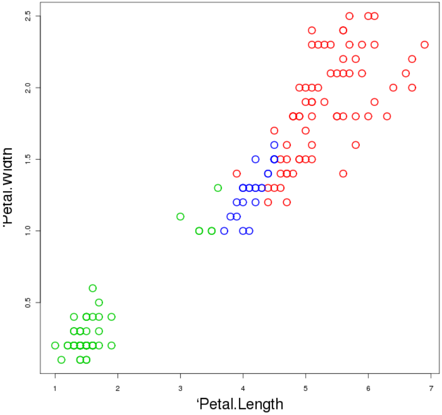
Estimation - illustration

Example of the EM convergence on the famous *iris* dataset.



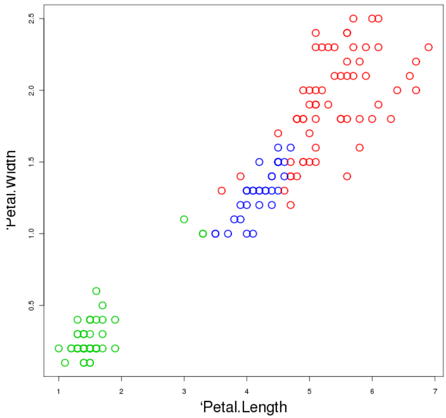
Estimation - illustration

Example of the EM convergence on the famous *iris* dataset.



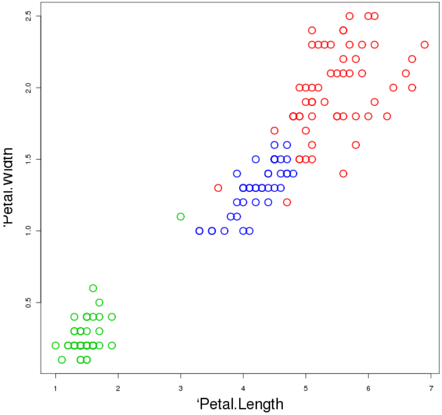
Estimation - illustration

Example of the EM convergence on the famous *iris* dataset.



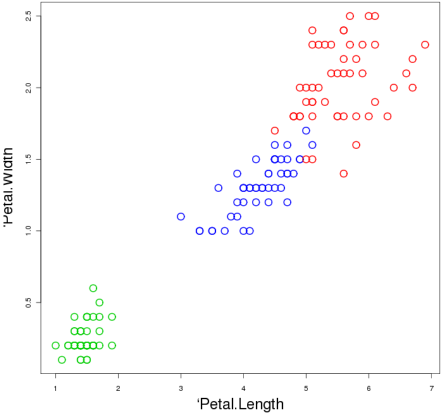
Estimation - illustration

Example of the EM convergence on the famous *iris* dataset.



Estimation - illustration

Example of the EM convergence on the famous *iris* dataset.



Plan

1 Preliminary on model-based clustering

2 Parsimonious modeling of cluster-dependent FPCA
- The model
- Model inference

3 Numerical applications
- Introductory example: Canada weather
- Mars surface characterization

Plan

Transformation of the observed curves

- Data : $\{x_1, ..., x_n\} \in L_2[0, T]$ indep. realiz. of $X = \{X(t)\}_{t\in[0,T]}$

Transformation of the observed curves

- Data : $\{x_1, ..., x_n\} \in L_2[0, T]$ indep. realiz. of $X = \{X(t)\}_{t\in[0,T]}$
- Observations : for each x_i, only $x_{ij} = x_i(t_{ij})$ are observed for $\{t_{ij} : j = 1, \ldots, m_i\}$.

Transformation of the observed curves

- Data : $\{x_1, ..., x_n\} \in L_2[0, T]$ indep. realiz. of $X = \{X(t)\}_{t \in [0,T]}$
- Observations : for each x_i, only $x_{ij} = x_i(t_{ij})$ are observed for $\{t_{ij} : j = 1, \ldots, m_i\}$.
- Basis expansion : *reconstruct the functional form of the data*

$$X(t) = \sum_{j=1}^{p} \gamma_j(X)\psi_j(t),$$

$\gamma = (\gamma_1(X), ..., \gamma_p(X))$ is a random vector in $\mathbb{R}^p$ (p known)

Transformation of the observed curves

- Data : $\{x_1, ..., x_n\} \in L_2[0, T]$ indep. realiz. of $X = \{X(t)\}_{t\in[0,T]}$
- Observations : for each x_i, only $x_{ij} = x_i(t_{ij})$ are observed for $\{t_{ij} : j = 1, \ldots, m_i\}$.
- Basis expansion : *reconstruct the functional form of the data*

$$X(t) = \sum_{j=1}^{p} \gamma_j(X)\psi_j(t),$$

$\gamma = (\gamma_1(X), ..., \gamma_p(X))$ is a random vector in $\mathbb{R}^p$ (p known)
$\Rightarrow x_i$ will be described by $\gamma_i = (\gamma_{i1}, ..., \gamma_{ip})$.

A group-specific functional latent model

Let $\{x_{i_1}, ..., x_{i_{n_k}}\}$ being n_k curves of $\mathcal{G}_k$ described by $\{\gamma_1, ..., \gamma_{n_k}\} \in \mathbb{R}^p$.

Assumptions

- $\{\gamma_1, ..., \gamma_{n_k}\}$ indep. realiz. of $\Gamma \in \mathbb{R}^p$.

A group-specific functional latent model

Let $\{x_{i_1},...,x_{i_{n_k}}\}$ being n_k curves of $\mathcal{G}_k$ described by $\{\gamma_1,...,\gamma_{n_k}\} \in \mathbb{R}^p$.

Assumptions

- $\{\gamma_1,...,\gamma_{n_k}\}$ indep. realiz. of $\Gamma \in \mathbb{R}^p$.
- $\{x_{i_1},...,x_{i_{n_k}}\}$ are sample paths of a stochastic process which can be described in a sufficient manner in a low-dimensional subspace $\mathbb{E}_k[0,T]$ of $L_2[0,T]$ with dimension $d_k \leq p$.

A group-specific functional latent model

Let $\{x_{i_1}, ..., x_{i_{n_k}}\}$ being n_k curves of $\mathcal{G}_k$ described by $\{\gamma_1, ..., \gamma_{n_k}\} \in \mathbb{R}^p$.

Assumptions

- $\{\gamma_1, ..., \gamma_{n_k}\}$ indep. realiz. of $\Gamma \in \mathbb{R}^p$.
- $\{x_{i_1}, ..., x_{i_{n_k}}\}$ are sample paths of a stochastic process which can be described in a sufficient manner in a low-dimensional subspace $\mathbb{E}_k[0, T]$ of $L_2[0, T]$ with dimension $d_k \leq p$.
- $\{\varphi_{kj}\}_{j=1,...,d_k}$ a basis of $\mathbb{E}_k[0, T]$,

A group-specific functional latent model

Let $\{x_{i_1}, ..., x_{i_{n_k}}\}$ being n_k curves of $\mathcal{G}_k$ described by $\{\gamma_1, ..., \gamma_{n_k}\} \in \mathbb{R}^p$.

Assumptions

- $\{\gamma_1, ..., \gamma_{n_k}\}$ indep. realiz. of $\Gamma \in \mathbb{R}^p$.
- $\{x_{i_1}, ..., x_{i_{n_k}}\}$ are sample paths of a stochastic process which can be described in a sufficient manner in a low-dimensional subspace $\mathbb{E}_k[0, T]$ of $L_2[0, T]$ with dimension $d_k \leq p$.
- $\{\varphi_{kj}\}_{j=1,...,d_k}$ a basis of $\mathbb{E}_k[0, T]$,
- $\{\lambda_1, ..., \lambda_{n_k}\}$ expansion coefficients of curves in $\{\varphi_{kj}\}_{j=1,...,d_k}$.

A group-specific functional latent model

Let $\{x_{i_1}, ..., x_{i_{n_k}}\}$ being n_k curves of $\mathcal{G}_k$ described by $\{\gamma_1, ..., \gamma_{n_k}\} \in \mathbb{R}^p$.

Assumptions

- $\{\gamma_1, ..., \gamma_{n_k}\}$ indep. realiz. of $\Gamma \in \mathbb{R}^p$.
- $\{x_{i_1}, ..., x_{i_{n_k}}\}$ are sample paths of a stochastic process which can be described in a sufficient manner in a low-dimensional subspace $\mathbb{E}_k[0, T]$ of $L_2[0, T]$ with dimension $d_k \leq p$.
- $\{\varphi_{kj}\}_{j=1,...,d_k}$ a basis of $\mathbb{E}_k[0, T]$,
- $\{\lambda_1, ..., \lambda_{n_k}\}$ expansion coefficients of curves in $\{\varphi_{kj}\}_{j=1,...,d_k}$.
- $\{\lambda_1, ..., \lambda_{n_k}\}$ indep. realiz. of $\Lambda \in \mathbb{R}^{d_k}$.

A group-specific functional latent model

Let $\{x_{i_1}, ..., x_{i_{n_k}}\}$ being n_k curves of $\mathcal{G}_k$ described by $\{\gamma_1, ..., \gamma_{n_k}\} \in \mathbb{R}^p$.

Assumptions

- $\{\gamma_1, ..., \gamma_{n_k}\}$ indep. realiz. of $\Gamma \in \mathbb{R}^p$.
- $\{x_{i_1}, ..., x_{i_{n_k}}\}$ are sample paths of a stochastic process which can be described in a sufficient manner in a low-dimensional subspace $\mathbb{E}_k[0, T]$ of $L_2[0, T]$ with dimension $d_k \leq p$.
- $\{\varphi_{kj}\}_{j=1,...,d_k}$ a basis of $\mathbb{E}_k[0, T]$,
- $\{\lambda_1, ..., \lambda_{n_k}\}$ expansion coefficients of curves in $\{\varphi_{kj}\}_{j=1,...,d_k}$.
- $\{\lambda_1, ..., \lambda_{n_k}\}$ indep. realiz. of $\Lambda \in \mathbb{R}^{d_k}$.
- Γ and Λ linked by

$$\Gamma = U_k \Lambda + \varepsilon,$$

where U_k a $p \times d_k$ matrix and $\varepsilon \in \mathbb{R}^p$ an indep. noise term.

A group-specific functional latent model

Distributional assumptions

- $\Lambda \sim \mathcal{N}(m_k, S_k)$, where $m_k \in \mathbb{R}^{d_k}$ and $S_k = \mathrm{diag}(a_{k1}, ..., a_{kd_k})$.
- $\varepsilon \sim \mathcal{N}(0, \Xi_k)$,

A group-specific functional latent model

Distributional assumptions

- $\Lambda \sim \mathcal{N}(m_k, S_k)$, where $m_k \in \mathbb{R}^{d_k}$ and $S_k = \mathrm{diag}(a_{k1}, ..., a_{kd_k})$.
- $\varepsilon \sim \mathcal{N}(0, \Xi_k)$,
- $\Rightarrow \quad \Gamma \sim \mathcal{N}(\mu_k, \Sigma_k)$, with $\mu_k = U_k m_k$ and $\Sigma_k = U_k S_k U_k^t + \Xi_k$.

A group-specific functional latent model

Distributional assumptions

- $\Lambda \sim \mathcal{N}(m_k, S_k)$, where $m_k \in \mathbb{R}^{d_k}$ and $S_k = \mathrm{diag}(a_{k1}, \ldots, a_{kd_k})$.
- $\varepsilon \sim \mathcal{N}(0, \Xi_k)$,
- $\Rightarrow \quad \Gamma \sim \mathcal{N}(\mu_k, \Sigma_k)$, with $\mu_k = U_k m_k$ and $\Sigma_k = U_k S_k U_k^t + \Xi_k$.

Parsimony assumptions By analogy to HDDC (Bouveyron *et al.* 2007)

- Ξ_k is assumed to be such that $\Delta_k = Q_k^t \Sigma_k Q_k$ can be written

$$\Delta_k = \left(\begin{array}{ccc|cccc} a_{k1} & & 0 & & & & \\ & \ddots & & & \mathbf{0} & & \\ 0 & & a_{kd_k} & & & & \\ \hline & & & b_k & & & 0 \\ & & & & \ddots & & \\ & \mathbf{0} & & & & \ddots & \\ & & & 0 & & & b_k \end{array}\right) \begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} d_k \\ \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\} (p - d_k) \end{array}$$

with $Q_k = [U_k, V_k]$ orthogonal and $a_{kj} > b_k$ for $j = 1, \ldots, d_k$.

The clustering model FunHDDC

Clustering background

- Let $Z_i = (Z_{i1}, \ldots, Z_{iK})$ indicates the group of the ith curve:
  $Z_{ik} = 1$ if the ith curve belongs to $\mathcal{G}_k$, 0 otherwise.
- Z_i are unobserved.
- Clustering task: predict the value of Z_i for each observed curve x_i.

The clustering model FunHDDC

Clustering background

- Let $Z_i = (Z_{i1}, \ldots, Z_{iK})$ indicates the group of the ith curve: $Z_{ik} = 1$ if the ith curve belongs to $\mathcal{G}_k$, 0 otherwise.
- Z_i are unobserved.
- Clustering task: predict the value of Z_i for each observed curve x_i.

Clustering model

- Each curve x_i is assumed to be sample path of X, admitting a basis expansion γ_i whose marginal distribution is:

$$p(\gamma) = \sum_{k=1}^{K} \pi_k \phi(\gamma; \mu_k, \Sigma_k),$$

  - ϕ is the Gaussian density function,
  - $\mu_k = U_k m_k$,
  - $\Sigma_k = Q_k \Delta_k Q_k^t$,
  - $\pi_k = P(Z_k = 1)$ is the prior probability of the group $\mathcal{G}_k$.

This model is quoted $\mathrm{FunHDDC}_{[a_{kj} b_k Q_k d_k]}$.

The FunHDDC model and its submodels

Parsimonious submodels can be defined by constraining model parameters within or between groups:

- fixing the first d_k diagonal elements of Δ_k to be common within each class

$$\Rightarrow \text{FunHDDC}_{[a_k b_k Q_k d_k]}$$

- fixing b_k to be common between the classes

$$\Rightarrow \text{FunHDDC}_{[a_{kj} b Q_k d_k]}$$

$$\Rightarrow \text{FunHDDC}_{[a_k b Q_k d_k]}$$

which both assume that the behavior of the error components outside the class specific subspaces is common.

Plan

FunHDDC: an EM-based algorithm

- unsupervised problem $\rightarrow$ direct maximization of the likelihood unfeasible,

$\Rightarrow$ EM algorithm:

- E step:
  computes the expectation of the complete log-likelihood conditionally on the current value of the model parameter $\theta^{(q-1)}$,
- M step:
  estimates the model parameter by maximizing the expectation of the complete likelihood conditionally on the posterior probabilities $t_{ik}^{(q)}$ computed in E step.

Model inference: the funHDDC algorithm

The E step in fact reduces to the computation of the posterior probabilities $t_{ik} = P(Z_i = k | X = x_i)$:

$$t_{ik}^{(q)} = 1 / \sum_{\ell=1}^{K} \exp\left(H_k^{(q-1)}(\gamma_i) - H_\ell^{(q-1)}(\gamma_i)\right),$$

with $H_k^{(q-1)}(\gamma)$ defined as:

$$H_k^{(q-1)}(\gamma) = ||\mu_k^{(q-1)} - P_k(\gamma)||_{D_k}^2 + \frac{1}{b_k^{(q-1)}}||\gamma - P_k(\gamma)||^2 + \sum_{j=1}^{d_k} \log(a_{kj}^{(q-1)}) + (p - d_k)\log(b_k^{(q-1)}) - 2\log(\pi_k^{(q-1)}),$$

where P_k is the projection operator on the latent space $\mathbb{E}_k$

Model inference: the funHDDC algorithm

The M step consists in updating estimates of model parameters:

- the mixture proportions are estimated by $\pi_k^{(q)} = n_k^{(q)}/n$, with $n_k^{(q)} = \sum_{i=1}^{n} t_{ik}^{(q)}$,
- the group means are estimated by $\mu_k^{(q)} = \frac{1}{n_k^{(q)}} \sum_{i=1}^{n} t_{ik}^{(q)} \gamma_i$,

Model inference: the funHDDC algorithm

The M step consists in updating estimates of model parameters:

- the mixture proportions are estimated by $\pi_k^{(q)} = n_k^{(q)}/n$, with $n_k^{(q)} = \sum_{i=1}^n t_{ik}^{(q)}$,
- the group means are estimated by $\mu_k^{(q)} = \frac{1}{n_k^{(q)}} \sum_{i=1}^n t_{ik}^{(q)} \gamma_i$,
- the d_k first columns of Q_k are updated by the eigenvectors associated with the largest eigenvalues of $W^{\frac{1}{2}} C_k^{(q)} W^{\frac{1}{2}}$ where $W = (w_{jk})_{1 \leq j,k \leq p} = \int_0^T \psi_j(t)\psi_k(t)dt$,

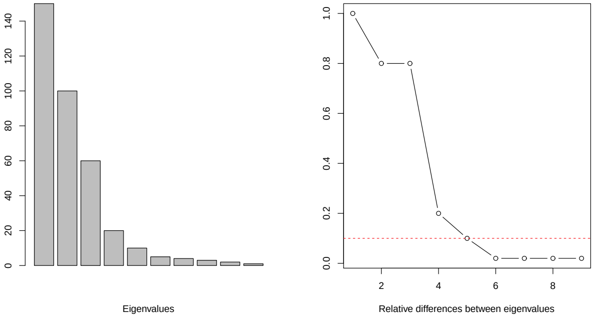
Model inference: the funHDDC algorithm

The M step consists in updating estimates of model parameters:

- the mixture proportions are estimated by $\pi_k^{(q)} = n_k^{(q)}/n$, with $n_k^{(q)} = \sum_{i=1}^{n} t_{ik}^{(q)}$,
- the group means are estimated by $\mu_k^{(q)} = \frac{1}{n_k^{(q)}} \sum_{i=1}^{n} t_{ik}^{(q)} \gamma_i$,
- the d_k first columns of Q_k are updated by the eigenvectors associated with the largest eigenvalues of $W^{\frac{1}{2}} C_k^{(q)} W^{\frac{1}{2}}$ where $W = (w_{jk})_{1 \leq j,k \leq p} = \int_0^T \psi_j(t)\psi_k(t)dt$,
- the variance parameters $a_{kj}, j = 1, ..., d_k$, are updated by the d_k largest eigenvalues of $W^{\frac{1}{2}} C_k^{(q)} W^{\frac{1}{2}}$,
- the variance parameters b_k are updated by $b_k^{(q)} = \mathrm{trace}(W^{\frac{1}{2}} C_k^{(q)} W^{\frac{1}{2}}) - \sum_{j=1}^{d_k} \hat{a}_{kj}^{(q)}$.

Model inference: estimation of hyper-parameters

The intrinsic dimensions d_k are estimated using the scree-test of Cattell which looks for a break in the eigenvalue scree.



The number K of groups is determined using the BIC criterion.

Plan

1. Preliminary on model-based clustering
2. Parsimonious modeling of cluster-dependent FPCA
   - The model
   - Model inference
3. Numerical applications
   - Introductory example: Canada weather
   - Mars surface characterization

Plan

An introductory example: Canada weather

The Canadian weather dataset:

- it is a classical set of time series presented in details in [Ramsay & Silverman],
- it consists in the daily measured temperatures at 35 Canadian weather stations across the country,
- 35 curves measured at 365 times.

Experimental protocol:

- we ran funHDDC for different numbers of groups and we kept the result with the highest BIC value,
- the most general model $[a_k b_k Q_k d_k]$ was used.

An introductory example: Canada weather

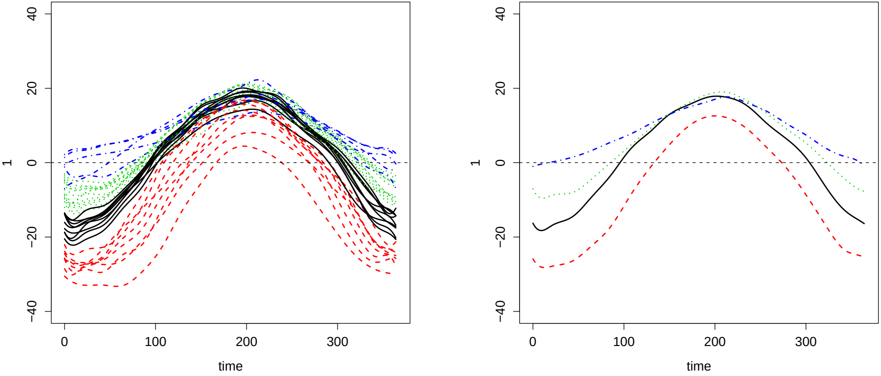
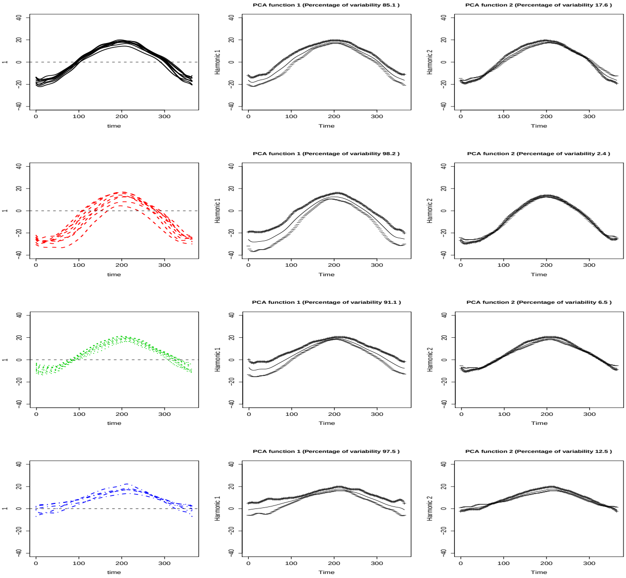


Fig. - Clustering in 4 groups (left) and group means (right).

Fig. - Geographical positions of the weather stations with their group labels.

An introductory example: Canada weather



An introductory example: Canada weather

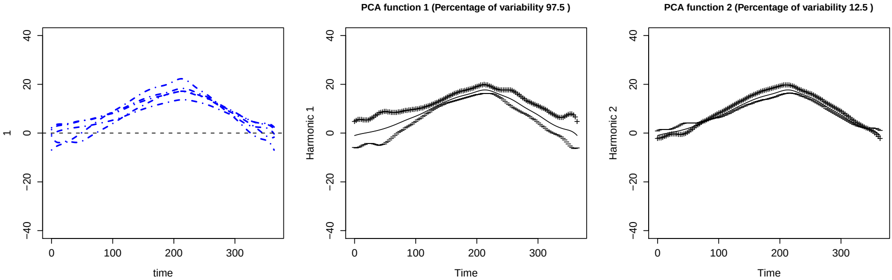


Fig. - Group 4 (mostly Pacific stations)

- PCA function 1: *high-variance during winter*,
- PCA function 2: *time-shift effect*.

An introductory example: Canada weather

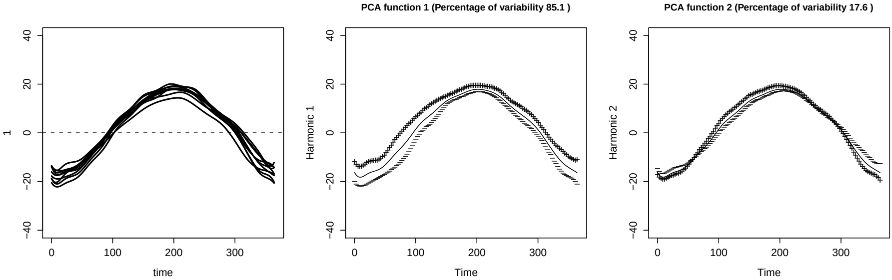


Fig. - Group 1 (mostly continental stations)

- PCA function 2: $+$ *and* $-$ *inversion*.

An introductory example: Canada weather

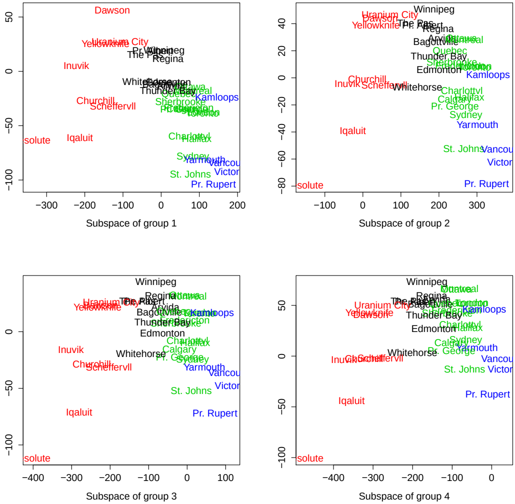


Fig. - Principal scores of the curves into the group-specific functional subspaces.

Plan

1. Preliminary on model-based clustering
2. Parsimonious modeling of cluster-dependent FPCA
   - The model
   - Model inference
3. Numerical applications
   - Introductory example: Canada weather
   - Mars surface characterization

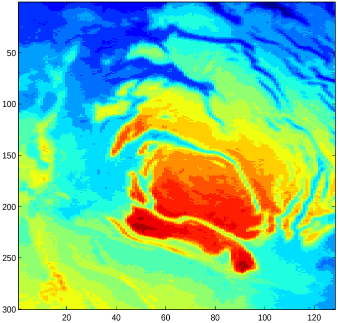
Mars surface characterization

The data

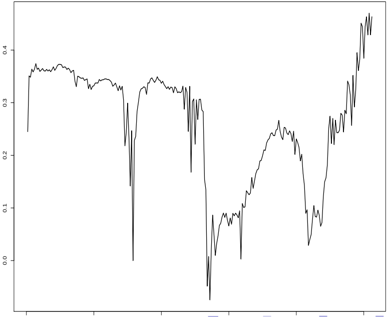
Hyperspectral images (OMEGA instrument, Mars Express spacecraft)

C. Bernard-Michel, S. Douté, M. Fauvel, L. Gardes and S. Girard *Retrieval of Mars surface physical properties frim OMEGA hyperspectral images using regularized sliced inverse regression*, Journal of Geophysical Research, 2009, 114, E06005.

Image 300×128



For each pixel



Mars surface characterization

Goal of the study

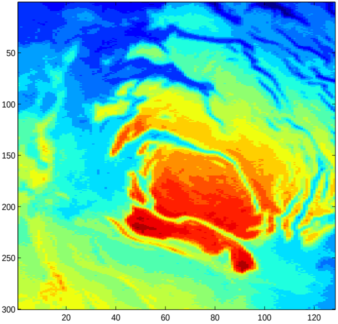
- Characterization of the surface materials,
- $\Rightarrow$ clustering of the 38400 pixels,
- number of groups expected by the experts: 8.

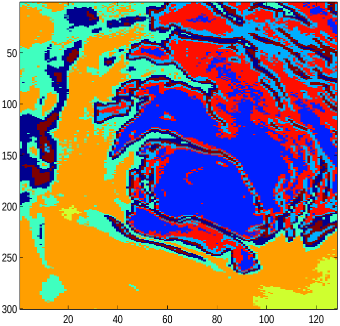
Results with fun-HDDC clustering

- All the submodels lead to relatively similar results,
- BIC tends to select more than 8 groups (about 10-13).

Mars surface characterization

Results obtained with one of the most general model $[a_k b_k Q_k D_k]$



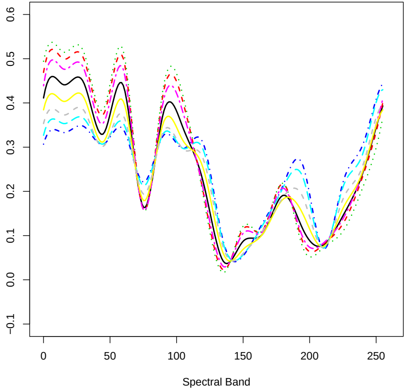


Mars photography and Classification in 8 groups

Consistent with the experts classification (in 8 groups): 51.96%

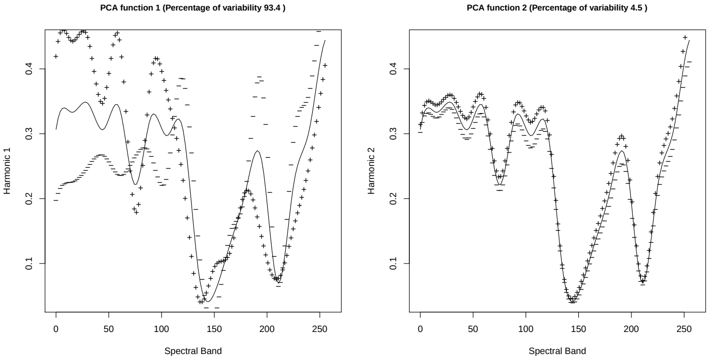
Mars surface characterization

Results obtained with one of the most general model $[a_k b_k Q_k D_k]$



Mean functions of the 8 groups

Results obtained with one of the most general model $[a_k b_k Q_k D_k]$



Class 4 (29.5%)

Conclusion

The funHDDC algorithm:

- is an extension of the multivariate clustering technique HDDC to functional data,
- it is a subspace clustering method which models and clusters the data in a low-dimensional functional subspace,
- it performs similarly or better than 2-step clustering methods while allowing useful interpretations.

Future works:

- extend the technique to multidimensional functions or time series,
- this would be possible by using a Gaussian model with block-diagonal covariance matrices within the group-specific functional subspaces.

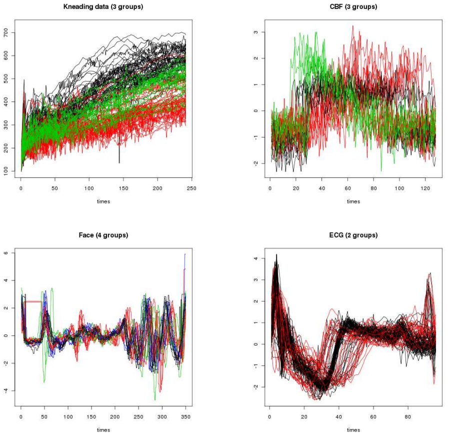
Bibliography

- C. Bernard-Michel, S. Douté, M. Fauvel, L. Gardes and S. Girard *Retrieval of Mars surface physical properties frim OMEGA hyperspectral images using regularized sliced inverse regression*, Journal of Geophysical Research, 2009, 114, E06005.
- C. Bouveyron, S. Girard and C. Schmid *High-Dimensional Data Clustering*, Computational Statistics and Data Analysis, 2007, 52[1], 502–519.
- C. Bouveyron and J. Jacques, *Model-based Clustering of Time Series in Group-specific Functional Subspaces*, Advances in Data Analysis and Classification, 2011, 5[4], 281-300.
- J. Jacques and C. Preda, *Model-based clustering of functional data*, Preprint HAL n°00 628247.
- J.O. Ramsay and B.W. Silverman, *Functional data analysis*, Springer, New York, 2005.
- G.M. James and C.A. Sugar, *Clustering for sparsely sampled functional data*, Journal of the American Statistical Association, 2003, 98[462], 397–408.

Numerical comparisons on benchmark datasets

We used 4 different time series datasets:

- Kneading: 3 groups, 115 curves,
- CBF: 3 groups, 930 curves,
- Face: 4 groups, 112 curves,
- ECG: 2 groups, 200 curves,



Comparison with fclust (James & Sugar, JASA, 2003)

Dataset	Kneading			CBF		
Number of groups	3			3		
Size	50			30		
Method	CCR	BIC	d	CCR	BIC	d
FunHDDC $[a_{kj} b_k Q_k d_k]$	70	-2403	(2,1,1)	63.3	-2430	(1,1,1)
FunHDDC $[a_{kj} b Q_k d_k]$	66.6	-2498	(1,1,1)	63.3	-2498	(1,1,1)
FunHDDC $[a_k b_k Q_k d_k]$	**70**	**-2193**	(1,1,1)	56.6	-2514	(1,1,1)
FunHDDC $[a_k b Q_k d_k]$	66.6	-2402	(1,1,1)	63.3	-2402	(1,1,1)
FunHDDC $[a b_k Q_k d_k]$	66.6	-2195	(1,2,1)	56.6	-2523	(1,1,1)
FunHDDC $[a b Q_k d_k]$	66.6	-2397	(1,1,1)	**63.3**	**-2397**	(1,1,1)
fclust	60			56.6		

Dataset	Face			ECG		
Number of groups	4			2		
Size	24			100		
Method	CCR	BIC	d	CCR	BIC	d
FunHDDC $[a_{kj} b_k Q_k d_k]$	62.5	-2162	(1,1,2,1)	77	-6667	(1,1)
FunHDDC $[a_{kj} b Q_k d_k]$	50	-2286	1,1,1,1)	76	-6428	(1,1)
FunHDDC $[a_k b_k Q_k d_k]$	**62.5**	**-2078**	(2,1,1,1)	77	-6333	(1,1)
FunHDDC $[a_k b Q_k d_k]$	58.3	-2083	(1,2,1,1)	77	-6191	(1,1)
FunHDDC $[a b_k Q_k d_k]$	66.6	-2092	(2,1,2,1)	77	-6317	(1,1)
FunHDDC $[a b Q_k d_k]$	58.3	-2080	(2,1,1,1)	**77**	**-6167**	(1,1)
fclust	41.6			75		

Comparison with two-step methods

FunHDDC	Kneading functional	2-steps methods	Kneading discretized (241 instants)	Kneading spline coeff. (20 splines)	Kneading FPCA scores (4 components)
$[a_{kj} b_k Q_k d_k]$	64.35	HDDC	**66.09**	53.91	44.35
$[a_{kj} b Q_k d_k]$	62.61	MixtPPCA	65.22	64.35	62.61
$[a_k b_k Q_k d_k]$	64.35	mclust	63.48	50.43	60
$[a_k b Q_k d_k]$	62.61	k-means	62.61	62.61	62.61
$[a b_k Q_k d_k]$	64.35	hclust	63.48	63.48	63.48
$[a b Q_k d_k]$	62.61				

FunHDDC	CBF functional	2-steps methods	CBF discretized (128 instants)	CBF spline coeff. (20 splines)	CBF FPCA scores (17 components)
$[a_{kj} b_k Q_k d_k]$	64.84	HDDC	68.60	51.18	68.17
$[a_{kj} b Q_k d_k]$	70.43	MixtPPCA	65.59	51.29	68.27
$[a_k b_k Q_k d_k]$	64.09	mclust	61.18	62.79	68.06
$[a_k b Q_k d_k]$	**70.65**	k-means	64.95	54.09	64.84
$[a b_k Q_k d_k]$	70.65	hclust	60.86	57.96	66.13
$[a b Q_k d_k]$	70.65				

FunHDDC	Face functional	2-steps methods	Face discretized (350 instants)	Face spline coeff. (20 splines)	Face FPCA scores (3 components)
$[a_{kj} b_k Q_k d_k]$	56.25	HDDC	59.82	58.03	63.39
$[a_{kj} b Q_k d_k]$	54.44	MixtPPCA	54.54	61.36	**64.77**
$[a_k b_k Q_k d_k]$	51.78	mclust	62.5	57.14	55.36
$[a_k b Q_k d_k]$	54.44	k-means	59.09	53.41	59.09
$[a b_k Q_k d_k]$	60.71	hclust	50.89	56.25	48.21
$[a b Q_k d_k]$	57.14				

FunHDDC	ECG functional	2-steps methods	ECG discretized (96 instants)	ECG spline coeff. (20 splines)	ECG FPCA scores (19 components)
$[a_{kj} b_k Q_k d_k]$	75	HDDC	74.5	73.5	74.5
$[a_{kj} b Q_k d_k]$	-	MixtPPCA	74.5	73.5	74.5
$[a_k b_k Q_k d_k]$	76.5	mclust	81	80.5	**81.5**
$[a_k b Q_k d_k]$	74.5	k-means	74.5	72.5	74.5
$[a b_k Q_k d_k]$	76.5	hclust	73	76.5	64
$[a b Q_k d_k]$	75				