Model-based Statistical Learning

Pr. Charles BOUVEYRON

Professor of Statistics
Chair of the Institut 3IA Côte d'Azur
Université Côte d'Azur & Inria

charles.bouveyron@univ-cotedazur.fr
@cbouveyron

The mixture of PPCA

Tipping & Bishop proposed in 1996 the mixture of PPCA

$$Z \sim \mathcal{M}(1; \pi) \qquad \pi = (\pi_1, \dots, \pi_K)$$

$$X_{|Z=k} = \mu U_k + \varepsilon_k \qquad \text{where } U_k \text{ is a } p \times d \text{ matrix}$$

$$\varepsilon_k \sim \mathcal{N}(0, \sigma_k^2 I_p)$$

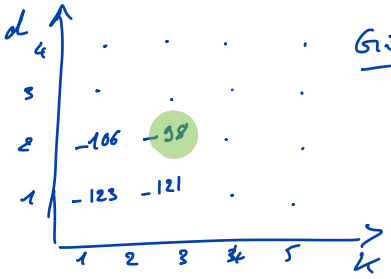
$$Y|Z=k \sim \mathcal{N}_d(\mu_k, I_d)$$

$$\Rightarrow X \sim \sum_{k=1} \pi_k \phi(\mu_k U_k, U_k^t U_k + \sigma_k^2 I_p)$$

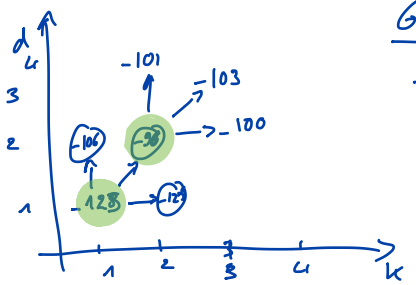
The mixture of PPCA

The influence of MPPCA can be done using the EM algorithm.

The choice of both K and d can be done using model selection, with BIC for instance.



Grid search



Glouton algo

HDDC (High Dimensional Data Clustering, Bouveyron 06)

The idea of HDDC is to extend MPPCA on 2 directions:

1) allows the intrinsic dimensions of the subspaces to be different : $d \rightarrow d_k$.

2) allows sub-models of MPPCA to fit into different situations with fewer data.

HDDC

the model of HDDC : $Z \sim \mathcal{M}(1, \pi)$

$X_{|Z=k} = Y U_k + \varepsilon_k$ where U_k is a $p \times d_k$ orthonormal matrix

$\varepsilon_k \sim \mathcal{N}(0, \sigma_k^2 I_p)$

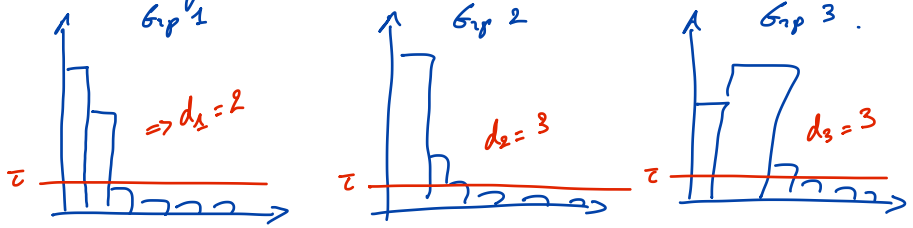
$Y \sim \mathcal{N}(\mu_k, a_k I_{d_k})$ where $a_k \in \mathbb{R}^{d_k}$.

$$\Rightarrow X \sim \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mu_k U_k, (a_k U_k)^t (a_k U_k) + \sigma_k^2 I_p)$$

HDDC

In fancy this HDDC model is still possible with EM, but model selection is now very difficult because we have $(K+1)$ discrete hyper-parameters to choose.

A solution to avoid this combinatorial problem: the scree-test of Cattell



$\Rightarrow$ Model selection here resumes to choosing K and τ.

In HDDC, it is also possible to get more parsimonious models by constraining some model parameters to be common between or within groups.

$\forall k,\ d_k = d \Rightarrow$ MPPCA

$\forall k,\ \sigma_k^2 = \sigma^2 \Rightarrow [a_{kj}\ b\ Q_k\ d_k)$

$a_{kj} = a_k \Rightarrow [a_k\ b_k\ Q_k\ d_k]$

$\sigma_k^2 = \sigma^2\ \&\ a_{kj} = a_k \Rightarrow [a_k\ b\ Q_k\ d_k)$

$(\ldots)$

$\Rightarrow [a\ b\ Q\ d]$

} 28 submodels which can be selected thanks to model selection.

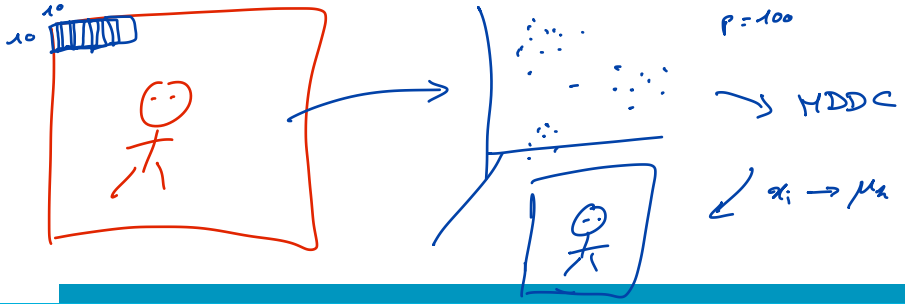
$\Rightarrow$ the R library HDclassif implements it.

$\Rightarrow$ the Python package HDDA ——— for Python
github.com/mfanuel/HDDA

In the supervised case: HDDA

In the supervised case, HDDA implements all the 28 models (including MDPCA) for supervised classification.

A remark: HDDC for image denoising turns out to be extremely performant: HDTRi method implements it.



Conclusion on subspace clustering / classif.:

⊕ extremely performant models for HD data.

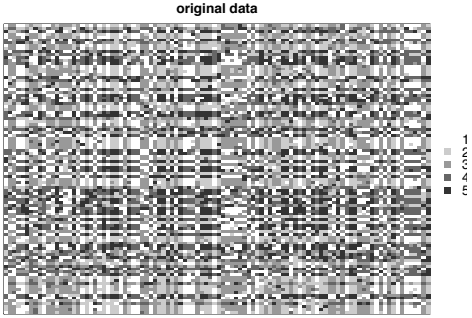
⊖ no easy way to visualise the data in the latent subspaces.

↳ the Fisher-EM algorithm aims to solve this visualization problem while keeping the idea of subspace clustering.

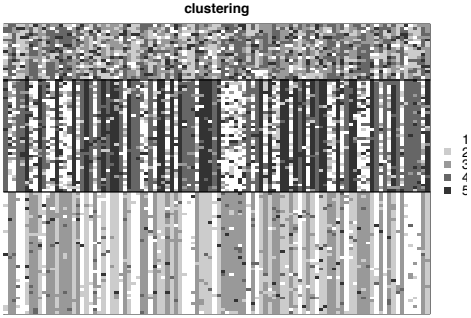
Outline

Co-clustering

When the number of variables is high, the interpretation of the obtained clustering can remain difficult:

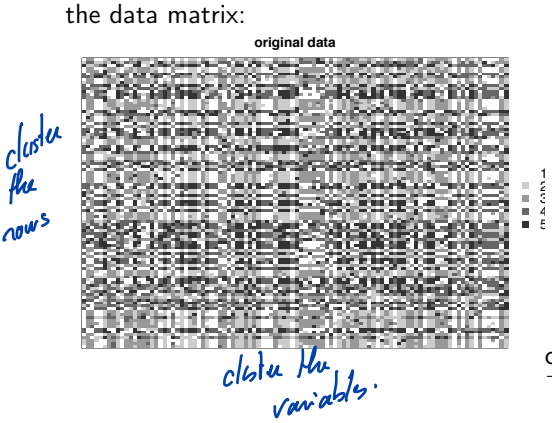


clustering

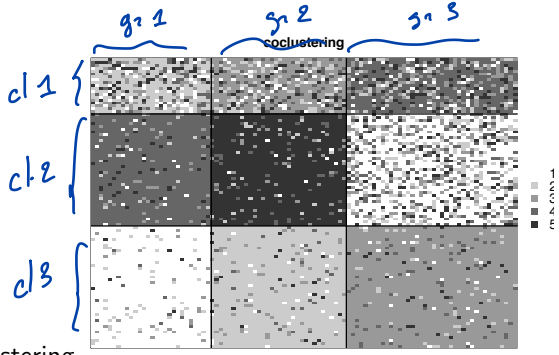


Co-clustering

In such a case, co-clustering may help by clustering simultaneously the rows and the columns of the data matrix:



co-clustering



The latent block model (LBM)

The LBM model is designed for clustering both rows and columns of a data set:

Z : cluster variable for rows (into K groups)

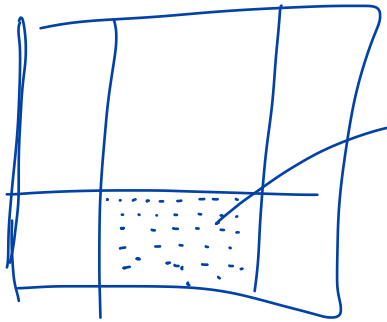
W : ——— columns (into L groups)

$Z \sim \mathcal{M}(1; \pi)$ and $W \sim \mathcal{M}(1; \rho)$

The latent block model (LBM)

Assumption : knowing z and w, the individuals into a block defined by z and w are all assumed to be independant of the same univariate pdf.

$$X_{ij} \mid z_{ik} = 1, w_{jl} = 1 \sim f_{kl}(x_{ij}; \theta_{kl})$$



$\sim f_{22}(x_{ij}; \theta_{22})$ which is a univariate distr.

The latent block model (LBM)

As an illustration, the LBM model for binary data:

$X_{ij} \in [0,1]$

$Z \sim \mathcal{M}(1; \pi)$

$W \sim \mathcal{M}(1; \rho)$

$X_{ij} \mid Z_{ik}=1, W_{j\ell}=1 \sim \mathcal{B}(p_{k\ell})$

$\Rightarrow$ the nb of parameters to estimate:

$\pi \rightarrow K-1$

$\rho \rightarrow L-1$

$p_{k\ell} \rightarrow KL.$

$\Big\}$ $KL + (K-1) + (L-1)$

The latent block model (LBM)

⊕ This model is extremely parsimonious in term of models parameters and, therefore, has no other way to find very good clusterings for rows and columns.

⊕ The LBM model can be adapted to various types of data by changing the univariate distribution.

- Binary data → Bernoulli
- count data → Poisson
- → continuous data → Gaussian.
- categorical → Multinomial
- compositional → Dirichlet

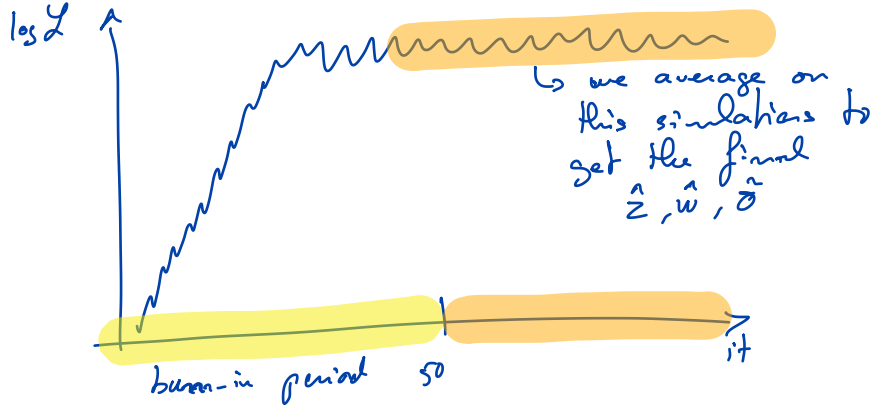
Inference of the LBM model

Unfortunately, it is not possible to directly use the EM algorithm because we have 2 latent variables.

One approach is to use a stochastic version of EM: the SEM-Gibbs algorithm.

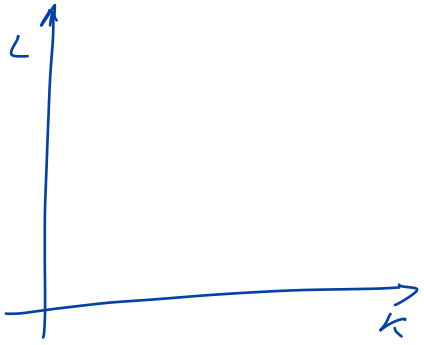
- SE step : we simulate Z and W according to the posterior conditional distribution (MCMC idea)
- M step : we find estimates of model parameters by maximizing the likelihood knowing some simulated values for Z and W.

This generates a sequence of posterior distributions which converges toward the actual posterior one.



Model selection

Model selection criteria (Bic, Aic, ICL) can be used to select K and L.



Grid search or the glouton approach.

Application with R

the blockcluster package implements the LBM for several data types.

$$X_{|Z=k} = YU + \varepsilon_k$$

but U is such that it discriminates best the clusters