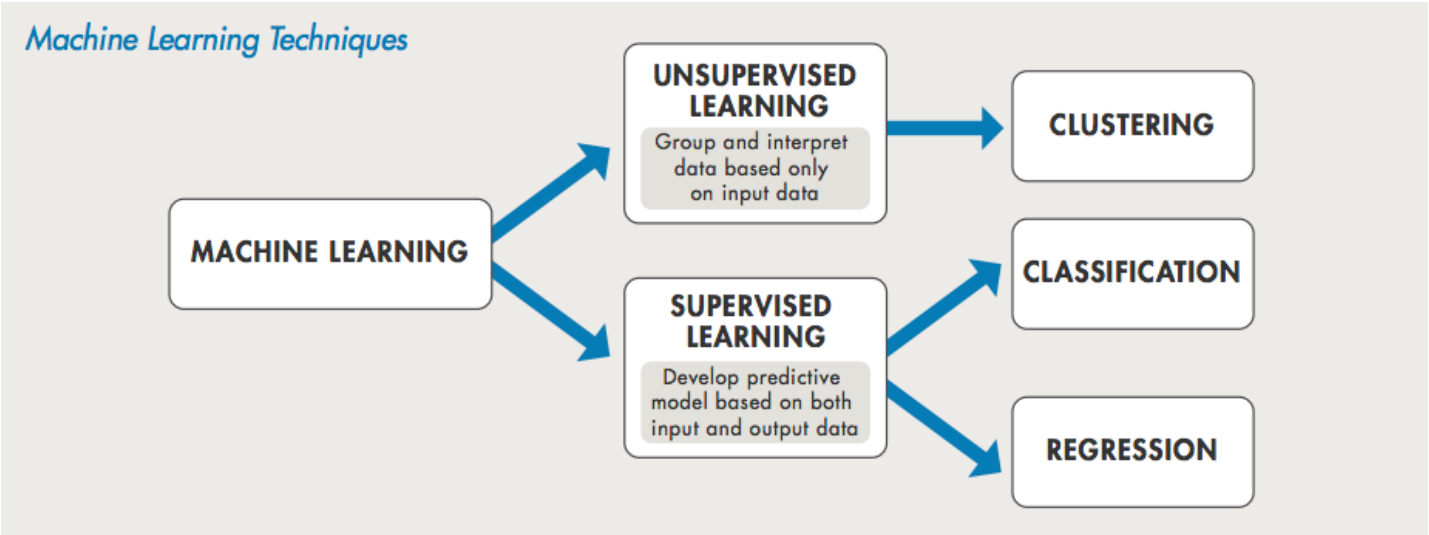
A Mean Field Games approach to cluster analysis

Fabio Camilli (Sapienza)

Joint work with Laura Aquilanti (Sapienza), Simone Cacace (Sapienza), Raul di Maio (I-Consulting), Adriano Festa (Politecnico Torino)

The aim is to provide an approach through the **Mean Field Games** theory to a classical problem in unsupervised Machine Learning, **the Cluster Analysis**

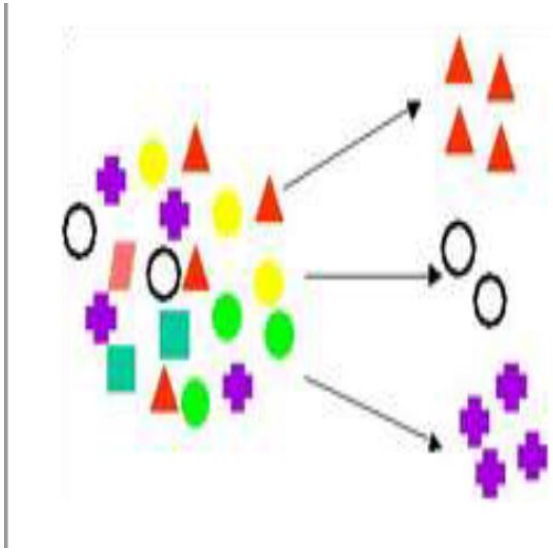


(a) Supervised versus Unsupervised

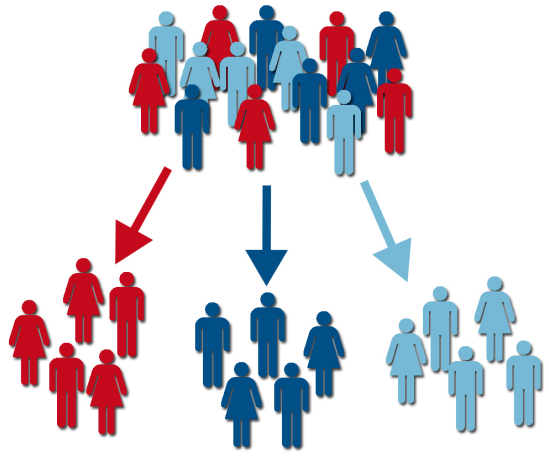
- In Supervised ML, we have prior knowledge of the output values for a set of data points. The goal is to learn a function that best approximates the relationship between input and output observable in the data.
- In Unsupervised ML, we do not have labelled outputs and the aim is to infer a specific structure within a set of data points.

Cluster Analysis

Clustering is the process of grouping a set of objects into classes of similar objects.
A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters.



(b)



(c)

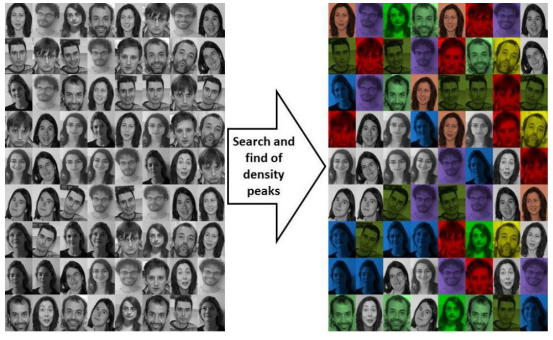
Cluster analysis is used for

- **Extracting set of patterns from the data set.**
- **Pre-process rough sample data for supervised ML**

Some typical applications are

Image Processing and Pattern Recognition

(d) color quantization



(e) face recognition

Market research $\rightarrow$ dividing costumers into homogeneous groups; grouping financial characteristics of companies;

Astronomy $\rightarrow$ classify different groups of stars and find unusual objects;

Biology $\rightarrow$ find groups of genes sharing similar functions.

Broadly speaking, clustering algorithms can be divided into two classes

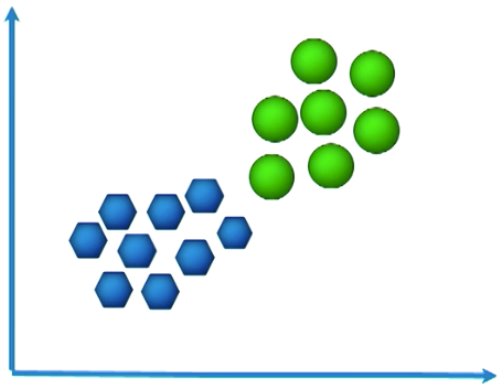
- **HARD Clustering:** each data point either belongs to a single cluster.
- **SOFT Clustering:** each data point has a certain probability to belong to each cluster

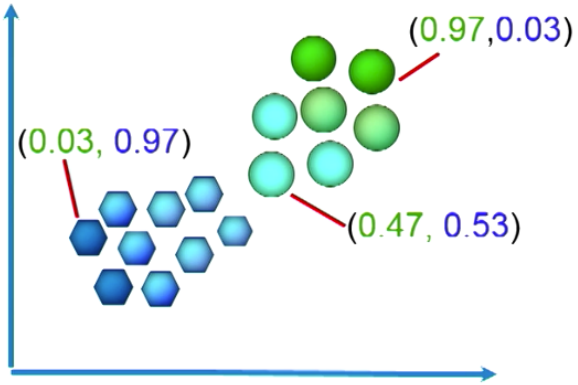
Hard clustering

Each observation belongs to exactly one cluster

Soft clustering

An observation can belong to more than one cluster to a certain degree (e.g. likelihood of belonging to the cluster)





(f) Hard versus soft clustering

Algorithms for cluster analysis

I will shortly review two well known techniques for cluster analysis

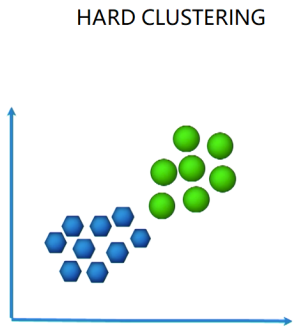
- **Hard Clustering:** K-means problem and Lloyd's algorithm;
- **Soft Clustering:** Mixture models and Expectation-Maximization algorithm.

We will see below that each of the previous techniques corresponds to a specific approach via MFG theory

Hard clustering via K-means

Given a data set $\mathcal{X} = \{x_1, \ldots, x_I\}$, $x_i \in \mathbb{R}^d$ and $\text{card}(\mathcal{X}) = I$, and fixed the number of clusters K, we aim to minimize the functional

$$J(c, \mu) = \sum_{i=1}^{I} \sum_{k=1}^{K} \mathbb{1}_{\{c_i = k\}} |x_i - \mu_k|^2 ,$$



with respect to

- the vector of **cluster assignment**
  $c = (c_1, c_2, \ldots, c_I)$, $c_i \in \{1, \ldots K\}$,
  i.e. $c_i = k \Leftrightarrow |x_i - \mu_k| < |x_i - \mu_j|$, $\forall j = 1, \ldots, K$
- the vector of **cluster barycentres**
  $\mu = (\mu_1, \mu_2, \ldots, \mu_K)$, $\mu_k \in \mathbb{R}^d$

In practice, by minimizing J, we partition the observations into K clusters

$$V(\mu_k) = \{x \in \mathbb{R}^d : |x - \mu_k| = \min_{j=1,\ldots,K} |x - \mu_j|\},$$

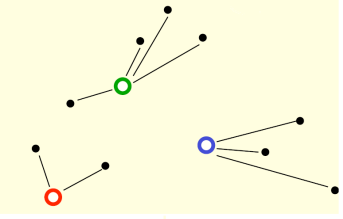
in such a way that each observation belongs to the cluster with the nearest barycentre.

Lloyd's algorithm

Starting from an arbitrary assignment μ^0, at n^{th} iteration:

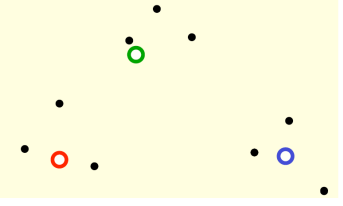
1. **Cluster assignment**:
   Assign the point x_i to the closest barycentre, i.e.

$$c_i^n = \arg\min_j |x_i - \mu_j^n|^2 \qquad \forall i = 1, \ldots, I.$$

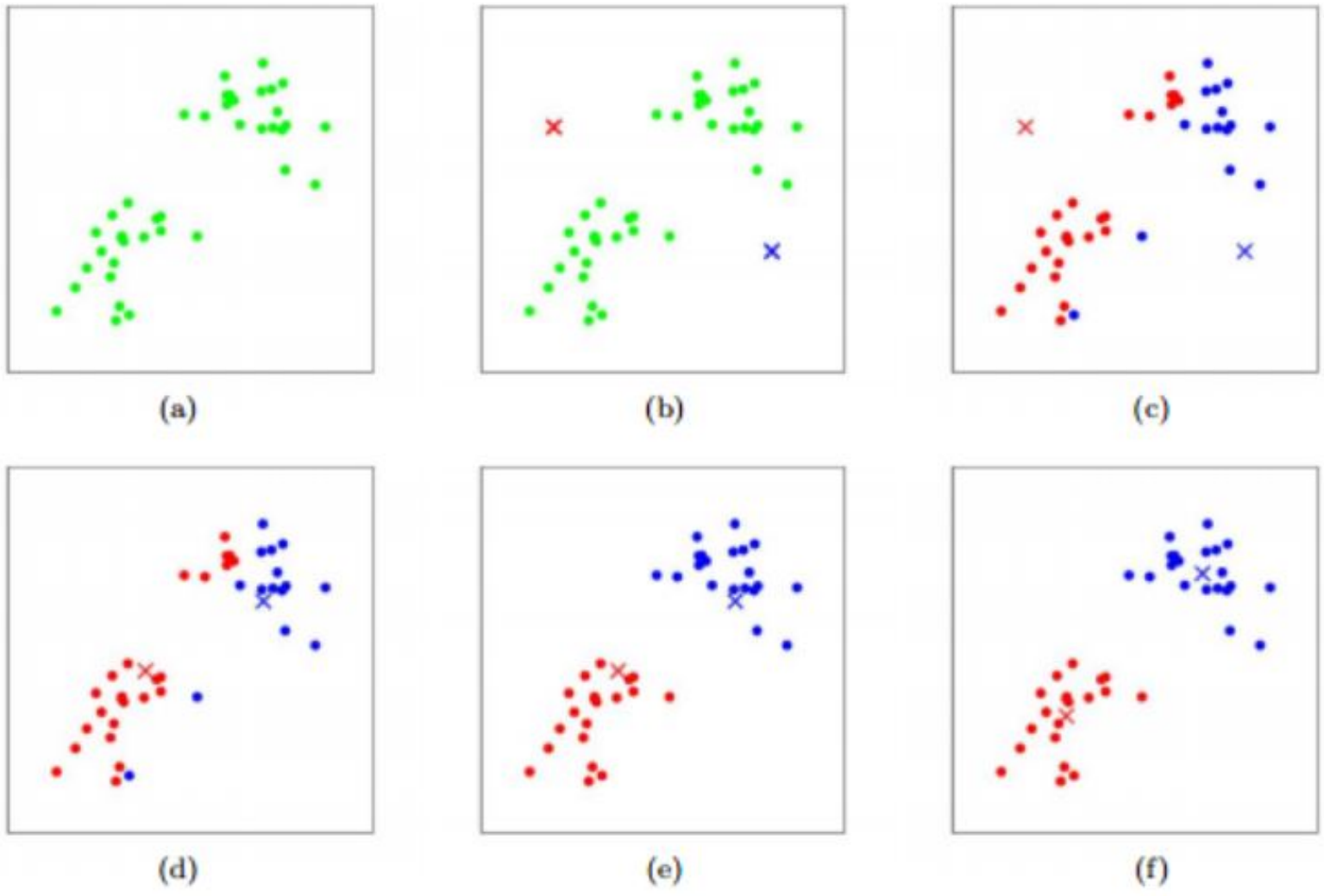


2. **Barycentre update**: Given c^n, we compute the new barycenters of the region $\{x_i : c_i^n = k\}$

$$\Rightarrow \mu_k^{n+1} = \frac{\sum_{i=1}^{I} x_i \mathbb{1}_{\{c_i^n = k\}}}{\sum_{i=1}^{I} \mathbb{1}_{\{c_i^n = k\}}} \qquad \forall k = 1, \ldots, K.$$



3. **Stopping criterion**: If
   $\sup_k |\mu_k^{n+1} - \mu_k^n| >$ error $\rightarrow$ *iterate*

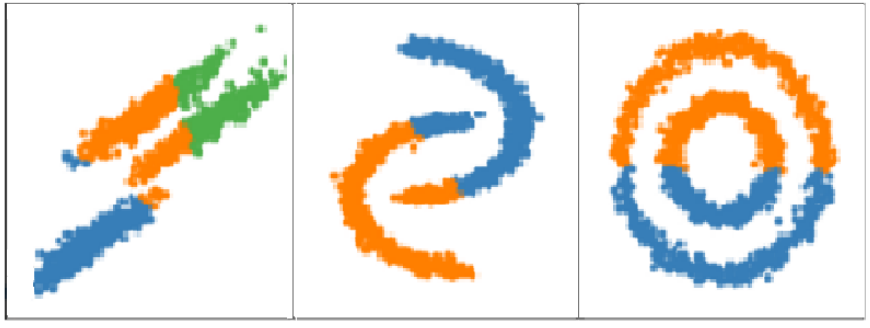


(a) Dataset (b) Random initial centroids.
(c-f) Two iterations of k-means

Advantages and disadvantages

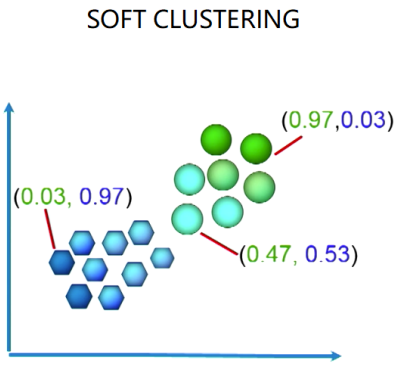
- ADVANTAGES:
  - Very fast (only need to compute the distances between point and barycenters).
  - simple to implement.

- DISADVANTAGES:
  - multiple solutions based on the initialization;
  - number of clusters is selected a priori;
  - all clusters have circular shapes, hence the algorithm fails in different cases.



Soft clustering via Finite Mixture model

Mixture model considers probabilistic partitions of data in cluster. We assume that the data set $\mathcal{X} = \{x_1, \ldots, x_l\}$ represents a set of (independent and identically distributed) observations of a continuous or discrete random variable X. We aim to represent the probability dsitribution of the r.v. X as a convex combination of parametrized probability density functions



$$p(x) = \sum_{k=1}^{K} \alpha_k p_k(x; \theta_k), \quad x \in \mathbb{R}^d$$

- K: number of components of p and α, fixed a priori;
- α_k: weights satisfying $\sum_{k=1}^{K} \alpha_k = 1, \ \alpha_k \in [0, 1]$;
- θ_k: parameters which defines the k-th pdfs.

Two classical examples of parametrized mixture models

Ex. I: Continuous sample space,

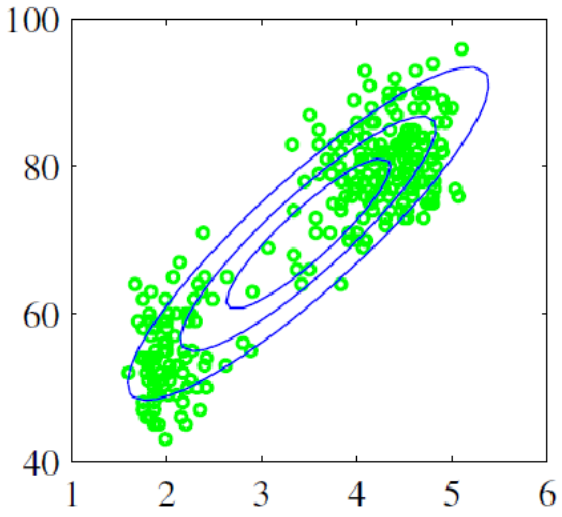
- $p_k(x; \theta_k)$ are Gaussian distributions
- $\theta_k = (\mu_k, \Sigma_k)$, mean and covariance

Ex. II: Discrete sample space

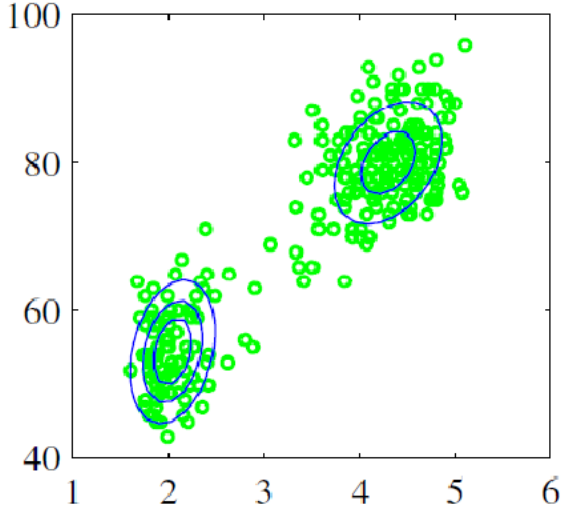
- $p_k(x; \theta_k)$ are Bernoulli distribution
- $\theta_k = \mu_k$, Bernoulli parameter

Why Mixture Models?

$$p(x) = \mathcal{N}(x; \mu, \Sigma)$$



$$p(x) = \alpha_1 \mathcal{N}(x; \mu_1, \Sigma_1) + \alpha_2 \mathcal{N}(x; \mu_2, \Sigma_2)$$



Blu contour represents a single probability density.
On the left we see a **single Gaussian** distribution and on the right a combination of **two Gaussians**.
The first distribution fails to capture the two clumps in the data and indeed places much of its probability mass in the center even though data are very sparse.

Given

$$p(x) = \sum_{k=1}^{K} \alpha_k p_k(x; \theta_k), \quad x \in \mathbb{R}^d$$

Aim: Find α, θ such that $p(x)$ represents the data set $\mathcal{X}$ faithfully

How: Maximize w.r.t. α and θ the log-likelihood functional

$$\mathcal{L}(\alpha, \theta) = \sum_{i=1}^{I} \sum_{k=1}^{K} \gamma_k(x_i) \ln(\alpha_k p_k(x_i; \theta_k))$$

Tool: Use Expectation-Maximization algorithm.

The responsabilities $\gamma_k(x_i)$ represent the probability that a point x_i of the data set is generated by the k^{th} component of the mixture and can used to divide the data set in clusters.

The EM algorithm in the Gaussian case

Starting from an arbitrary assignment $\alpha^0,\ \mu^0, \Sigma^0$, at n^{th} iteration we have:

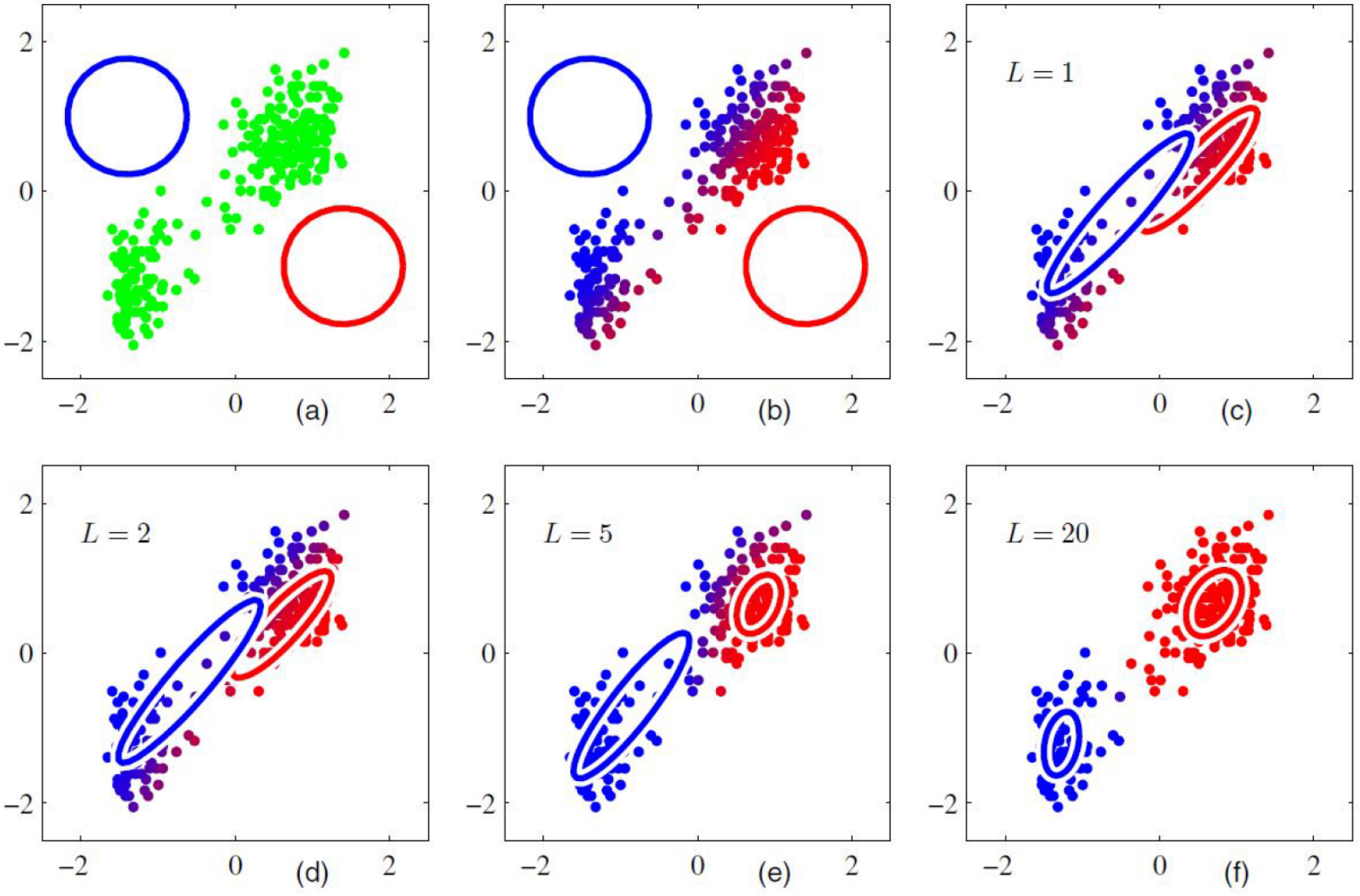
1. **E-step**: Given $\mu_k^{n-1},\ \Sigma_k^{n-1},\ \alpha_k^{n-1}$, $k = 1, \ldots, K$, compute

$$\gamma_k^n(x_i) = \frac{\alpha_k^{n-1} p(x_i; \mu_k^{n-1}, \Sigma_k^{n-1})}{\sum_{j=1}^K \alpha_j^{n-1} p(x_i; \mu_j^{n-1}, \Sigma_j^{n-1})}, \quad \text{(Bayes' Thm.)}$$

2. **M-step**: Update the parameters α, μ, Σ, by setting for $k = 1, \ldots, K$,

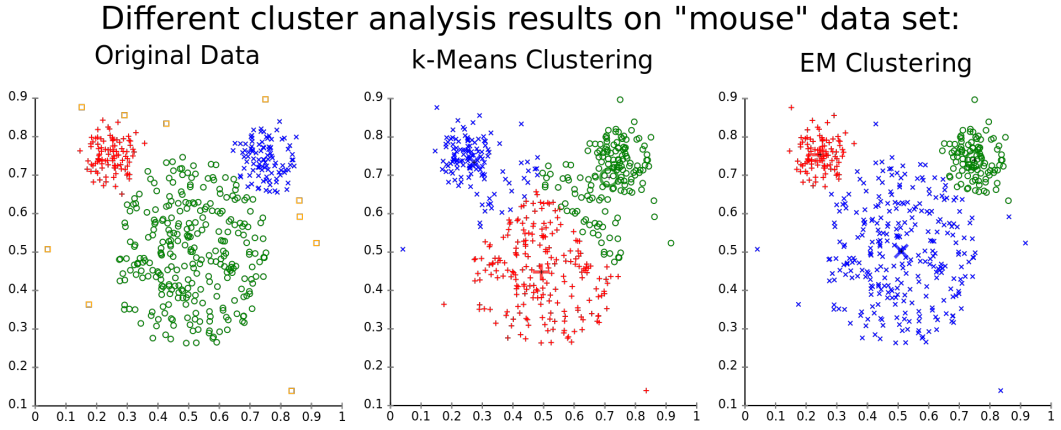
$$\alpha_k^n = \frac{\sum_{i=1}^I \gamma_k^n(x_i)}{I}, \quad \mu_k^n = \frac{\sum_{i=1}^I x_i \gamma_k^n(x_i)}{\sum_{i=1}^I \gamma_k^n(x_i)},$$

$$\Sigma_k^n = \frac{\sum_{i=1}^I \gamma_k^n(x_i)(x_i - \mu_k^n)^t(x_i - \mu_k^n)}{\sum_{i=1}^I \gamma_k^n(x_i)}.$$


2
0
−2
−2
0
2
(a)
(b)
$L = 1$
(c)
$L = 2$
(d)
$L = 5$
(e)
$L = 20$
(f)

Advantages and disadvantages

- **ADVANTAGES:** More flexible in terms of cluster covariance than K-Means: the clusters can take any ellipsoidal shape, rather than being restricted to circles.



(g) EM versus K-means

- **DISADVANTAGES:**
  - The number of clusters is selected a priori;
  - different clustering results for different initializations of the algorithm;
  - fails on some specific examples.

References for cluster analysis

- C.M. Bishop, *Pattern recognition and Machine Learning*, Information Science and Statistics, Springer, New York, 2006.
- J.A. Bilmes, A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov model, Technical Report ICSI-TR-97-021, University of Berkeley, 2000.
- L. Bottou and Y. Bengio, Convergence properties of the K-means algorithms, Adv. Neural Inf. Process. Syst. 82 (1995), 585-592.
- A. Saxena, M. Prasad, A. Gupta, N. Bharill, O.P. Patel, A. Tiwari, M.J. Er, W. Ding and C.T. Lin, A review of clustering techniques and developments, Neurocomputing, 267 (2017), 664–681.

Mean Field Games theory - a brief introduction

Mean Field Games theory aims to study strategic interactions among an infinite number of agents that are rational, small, homogenous and identical and are described by a density function m.
In the basic model, the representative agent controls the stochastic dynamics

$$\begin{cases} dX_t = a_t dt + \sqrt{2\varepsilon} dW_t, & t > 0 \\ X_0 = x. \end{cases}$$

where W_t is a Brownian motion and the control law a_t represents the control which an agent chooses in order to minimize the long time average cost functional

$$J(x, a) = \lim_{T \to +\infty} \frac{1}{T} \mathbb{E}_x \left\{ \int_0^T \left[L(X_s, a_s) + F(X_s, m(X_s)) \right] ds \right\},$$

$L(x, a)$ is the Lagrangian and $F(x, m)$ is the coupling term depending on the distribution m of the other agents.

Nash equilibria are characterized by a 2^{nd} order ergodic Mean Field Games system

$$\begin{cases} -\varepsilon\Delta u(x) + H(x, Du(x)) + \lambda = F[m](x), & x \in \mathbb{R}^d, \quad \text{(HJB)} \\ \varepsilon\Delta m(x) + \operatorname{div}(D_p H(x, Du(x))m(x)) = 0, & x \in \mathbb{R}^d, \quad \text{(FP)} \\ m \geq 0, \ \int_{\mathbb{R}^d} m(x)dx = 1, \ \int_{\mathbb{R}^d} u(x)dx = 0. \end{cases}$$

- The first equation is a Hamilton-Jacobi-Bellman equation, the second a Fokker-Planck equation and u, λ, m are the unknowns.
- (u, λ) describe the value function of the players at position x, while m represents the distribution when they choose the optimal strategy
- the coupling is given by $F[m]$ in the first equation and the term Du inside the divergence in the second equation
- $H(x, p) = \sup_{q \in \mathbb{R}^d}\{pq - L(x, q)\}$ is the Hamiltonian given by the Legendre transform of L.
- $\int_{\mathbb{R}^d} u(x)dx = 0$, $m \geq 0$, $\int_{\mathbb{R}^d} m(x)dx = 1$ are normalization conditions.

The MFG approach to cluster analysis: soft clustering and mixture models

Let $\mathcal{X}$ be a **big data set** described by a probability density function $f : \mathbb{R}^d \to \mathbb{R}$, $\int_{\mathbb{R}^d} f(x)dx = 1, f(x) \geq 0$. The aim is to find a mixture $m(x) = \sum_{k=1}^{K} \alpha_k m_k(x)$ that best fits f.

We consider data points as agents and we subdivide the undistinguished population m into sub-populations, each one described by a density functions m_k, by means of a multi-population Mean Field Games model.

The similarity, or proximity, among the members of a same population is encoded in the cost functional of the optimal control problems for each population, which push the agents to aggregate around the closer barycentre of the given distribution m_k.

Note that, in the standard multi-population Mean Field Games model, populations are distinguished from the beginning.

A representative agent of k^{th} population follows the dynamics

$$\begin{cases} dX_k(s) = a_k(s)ds + \sqrt{2\varepsilon}dW_k(s), & s > 0 \\ X_k(0) = x. \end{cases}$$

and $a_k(s)$ is chosen in order to minimize the cost functional

$$J_k(x, \alpha, m) = \lim_{T\to+\infty} \mathbb{E}_x \frac{1}{T} \int_0^T \left[\frac{1}{2}|a_k(s)|^2 + F_k(X_s, m(X_s))\right] ds,$$

$$F_k(x, m) = \frac{1}{2}(x - \mu_k)^t(\Sigma_k{}^{-1})^t(\Sigma_k{}^{-1})(x - \mu_k).$$

barycenter: $\mu_k = \dfrac{\int_{\mathbb{R}^d} x\gamma_k(x)f(x)dx}{\int_{\mathbb{R}^d} \gamma_k(x)f(x)dx}$

covariance: $\Sigma_k = \dfrac{\int_{\mathbb{R}^d}(x - \mu_k)^t(x - \mu_k)\gamma_k(x)f(x)dx}{\int_{\mathbb{R}^d} \gamma_k(x)f(x)dx}$

weights: $\alpha_k = \displaystyle\int_{\mathbb{R}^d} \gamma_k(x)f(x)dx,$

fraction on total mass: $\gamma_k(x) = \dfrac{\alpha_k m_k(x)}{m(x)}$

$$F_k(x, m, m_k) = \frac{1}{2}(x - \mu_k)^t(\Sigma_k{}^{-1})^t(\Sigma_k{}^{-1})(x - \mu_k).$$

We observe that:

- The potential F_k forces the data points to distribute with an higher probability around the nearest point μ_k, with an attenuation factor given by the variance Σ_k.
- The coupling among the various populations is given by the dependence of μ_k, Σ_k on

$$\gamma_k(x) = \frac{\alpha_k m_k(x)}{m(x)}$$

which depends on the total measure m and can be interpreted as the probability that a point of the data set x is generated by the k^{th} component of the mixture.

The corresponding multi-population MFG system is for $k = 1, \dots, K$.

$$\begin{cases} -\varepsilon \Delta u_k(x) + \frac{1}{2}|Du_k(x)|^2 + \lambda_k = \frac{1}{2}(x-\mu_k)^t(\Sigma_k^{-1})^t(\Sigma_k^{-1})(x-\mu_k), \\ \varepsilon \Delta m_k(x) + \operatorname{div}(m_k(x)Du_k(x)) = 0, \\ \alpha_k = \int_{\mathbb{R}^d} \gamma_k(x) f(x) dx, \\ m_k \geq 0, \int m_k(x)dx = 1, u_k(\mu_k) = 0, \end{cases}$$

Because the Hamiltonian and the coupling cost are quadratic, the solution to the MFG is a mixture of Gaussian densities

$$m(x) = \sum_{k=1}^{K} \alpha_k m(x; \mu_k, \Sigma_k)$$

where

$$m_k(x; \mu_k, \Sigma_k) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_k|^{\frac{1}{2}}} e^{\frac{1}{2}(x-\mu_k)^t \Sigma_k^{-1}(x-\mu_k)}$$

Note that α_k, μ_k, Σ_k are unknown and are obtained by solving the MFG system.

Proposition

Let $\{(u_k, \lambda_k, m_k, \alpha_k)\}_{k=1}^K$ be a solution of the MFG system with $\varepsilon = 1$. Then the parameters $\{\alpha_k, \mu_k, \Sigma_k\}_{k=1}^K$ of the mixture

$$m(x) = \sum_{k=1}^{K} \alpha_k m(x; \mu_k, \Sigma_k)$$

give a critical point of (a continuous version) of the log-likelihood functional

$$\mathcal{L}(\alpha, \mu, \Sigma) = \int_{\mathbb{R}^d} \sum_{k=1}^{K} \gamma_k(x) \ln \left(\alpha_k m_k(x; \mu_k, \Sigma_k) \right) f(x) dx$$

with $\gamma_k(x) = \frac{\alpha_k m_k(x)}{m(x)}$ being the corresponding responsibilities.

Conversely, each critical point of functional log-likelihood can be characterized through a solution of the MFG system.

A general MFG system for cluster analysis

The previous Gaussian model can be generalized in several directions. Given a a bounded set D which contains the support of the data set f, we consider the MFG system

$$\begin{cases} -\varepsilon\Delta u_k(x) + H(x, Du_k(x)) + \lambda_k = F_k(x, m), & x \in D, \\ \varepsilon\Delta m_k(x) + \mathrm{div}(m_k(x) D_p H(x, Du_k(x))) = 0, & x \in D, \\ \partial_n u_k(x) = 0, & x \in \partial D, \\ \varepsilon\partial_n m_k(x) + m_k(x) D_p H(x, Du_k(x)) \cdot n = 0, & x \in \partial D, \\ \alpha_k = \int \gamma_k(x) f(x) dx, \\ m_k(x) \geq 0,\ \int_D m_k(x) dx = 1,\ \int_D u_k(x) dx = 0, \end{cases}$$

for $k = 1, \ldots, K$, where n is the outward normal to the boundary of D, ∂_n the normal derivative.

A typical Hamiltonian H_k is

$$H_k(x,p) = R|p|^\gamma - V_k(x)$$

with $\gamma > 1$, $R > 0$, $V_k \in C^2(D)$. We assume that the coupling cost F_k, $k = 1,\dots,K$, is a nonnegative, regular function depending on $\{m_k\}_{k=1}^K$, but not on $\{u_k\}_{k=1}^K$.

Typical examples of cost functions are

- $F_k(x,m) = F_k(x,\mu_k,\sigma_k)$,
  where $\mu_k = \frac{\int_{\mathbb{R}} x\gamma_k(x)f(x)dx}{\int_{\mathbb{R}} \gamma_k(x)f(x)dx}$, $\sigma_k^2 = \frac{\int (x-\mu_k)^2\gamma_k(x)f(x)dx}{\int \gamma_k(x)f(x)dx}$.
- $F_k(x,m) = m_k(x)\ln\left(\frac{q_k(x)}{m_k(x)}\right)$ where q_k depends on the data set f (Kullback-Leibler divergence)

Under the previous assumptions the general MFG admits a solution $(u_k,\lambda_k,m_k,\alpha_k)$, $k = 1,\dots,K$.

A MFG version of the EM algorithm

- (**Inizialization**) Choose randomly $\alpha_1^0, \dots, \alpha_k^0$, $m_1^0 \dots, m_k^0$;
- (**E-step**) Compute the responsibilities $\gamma_1^n, \dots, \gamma_k^n$,

$$\gamma_k^n(x) = \frac{\alpha_k^n m_k^n(x)}{m^n(x)}$$

- (**M-step**) Solve the k (decoupled) MFG systems

$$\begin{cases} -\varepsilon \Delta u_k(x) + \frac{1}{2}|Du_k|^2 + \lambda_k = \frac{1}{2}\left|\frac{x-\mu_k^n}{(\sigma_k^n)^2}\right|^2, & x \in \mathbb{R}, \\ \varepsilon \Delta m_k(x) + \operatorname{div}(m_k(x) Du_k(x)) = 0, & x \in \mathbb{R}, \\ \alpha_k = \int_{\mathbb{R}} x \gamma_k^n(x) dx \\ m_k > 0, \ \int m_k(x)dx = 1, \ \int_D u_k(x)dx = 0 \end{cases}$$

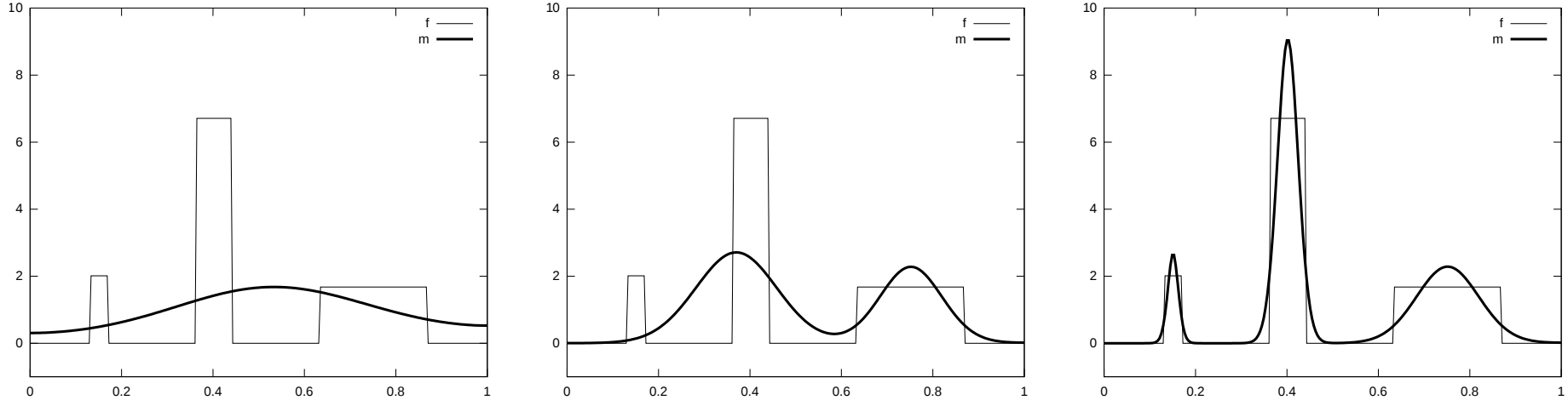
where

$$\mu_k^n = \frac{\int_{\mathbb{R}} x \gamma_k^n(x) f(x) dx}{\int_{\mathbb{R}} \gamma_k^n(x) f(x) dx}, \quad (\sigma_k^n)^2 = \frac{\int_{\mathbb{R}} (x-\mu_k^n)^2 \gamma_k^n(x) f(x) dx}{\int_{\mathbb{R}} \gamma_k^n(x) f(x) dx}$$

- (**Stopping criterion**) If $\sup_k |\mu_k^{n+1} - \mu_k^n| >$ error, iterate

Test 1. Piecewise constant data set

We consider a piece-wise constant distribution f on $\Omega = [0, 1]$, composed by three plateaux of different widths and heights, such that $\int_0^1 f(x)\,dx = 1$.



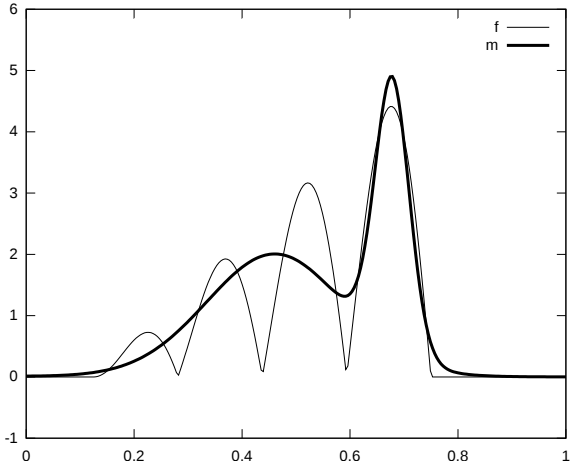
(a) (b) (c)

The thin line represents f, while the thick line represents the mixture $m = \sum_{k=1}^{K} \alpha_k m_k$, for $K = 1, 2, 3$ from (a) to (c).

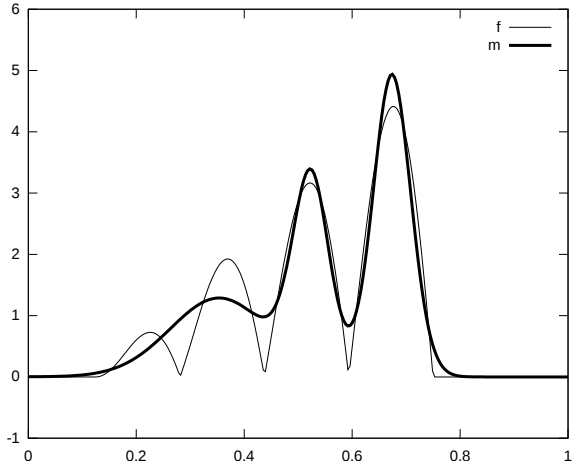
The mean and the variance of each m_k adapt to the data, according to the given number K of mixture components.

Test 2. Oscillating data set

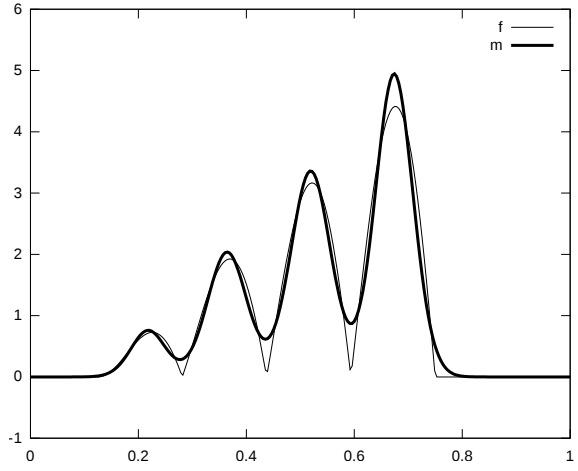
f is given by suitably scaling and translating the function $x \sin(4\pi x)$ for $x \in [0, 1]$, so that f has compact support and $\int_0^1 f(x)dx = 1$.



(a)



(b)



(c)

We show the solutions corresponding to $K = 2, 3, 4$ from (a) to (c). The peaks of f are sequentially approximated as the number K of mixture components increases, according to their heights and the underlying masses.

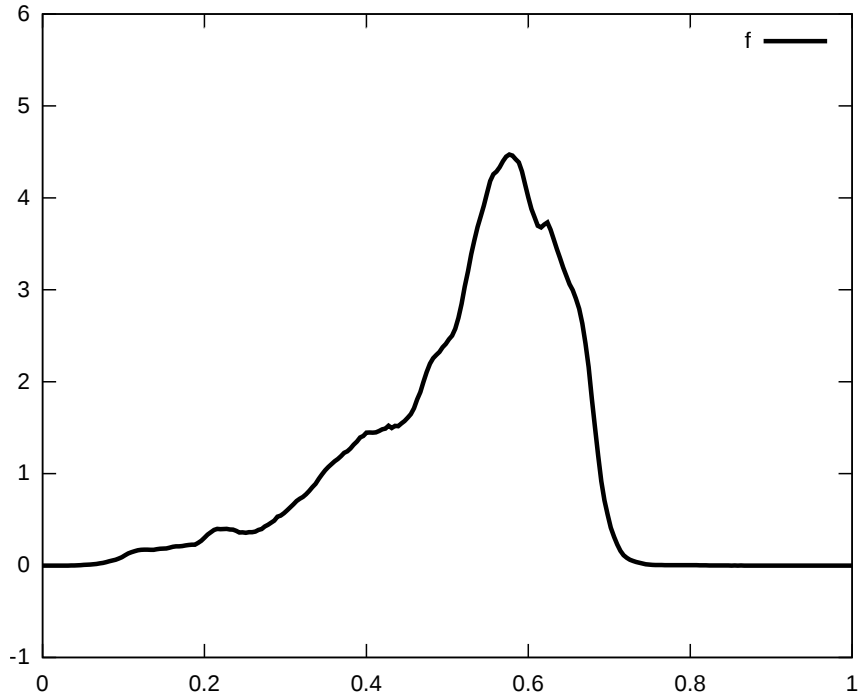
Test 3. An application to color quantization

Consider the case of an image in gray scales, i.e. each pixel contains a level of gray represented by a value in the interval $[0, 1]$. To generate the data set distribution f:

x-axis: grey level in $[0, 1]$

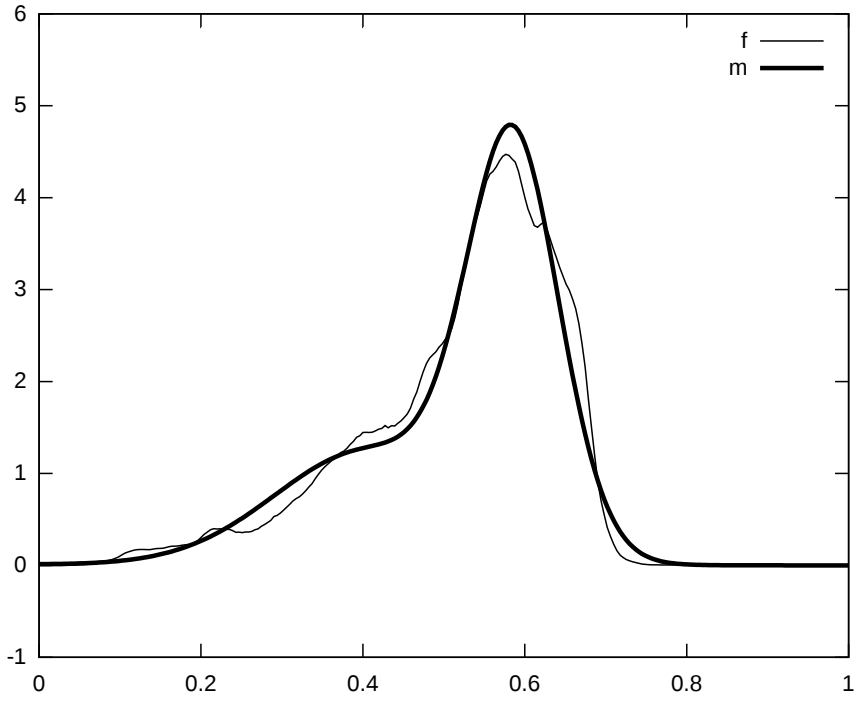
y-axis: their frequency in the pixels of the image

(a)

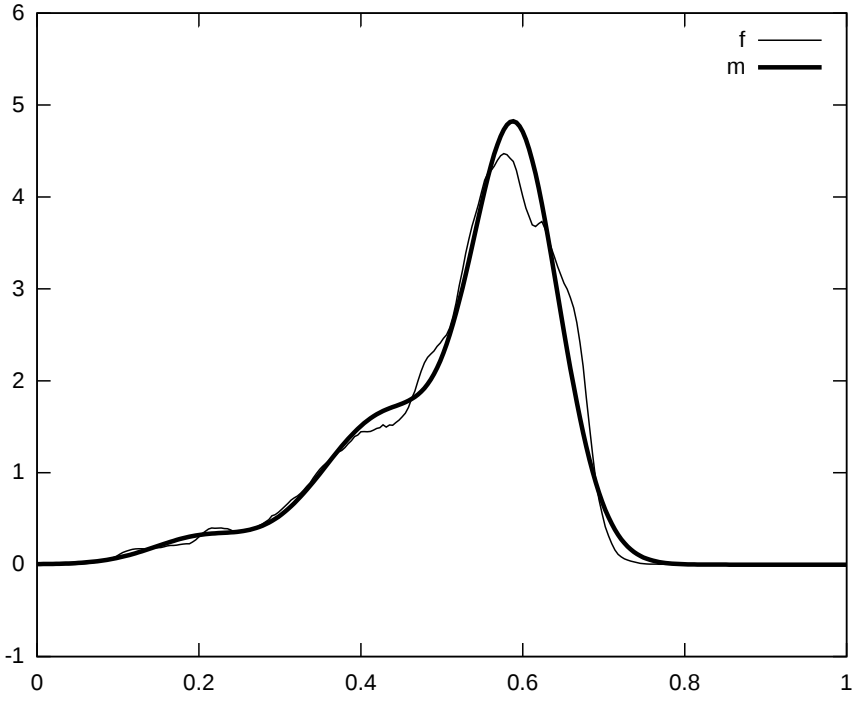


(b)

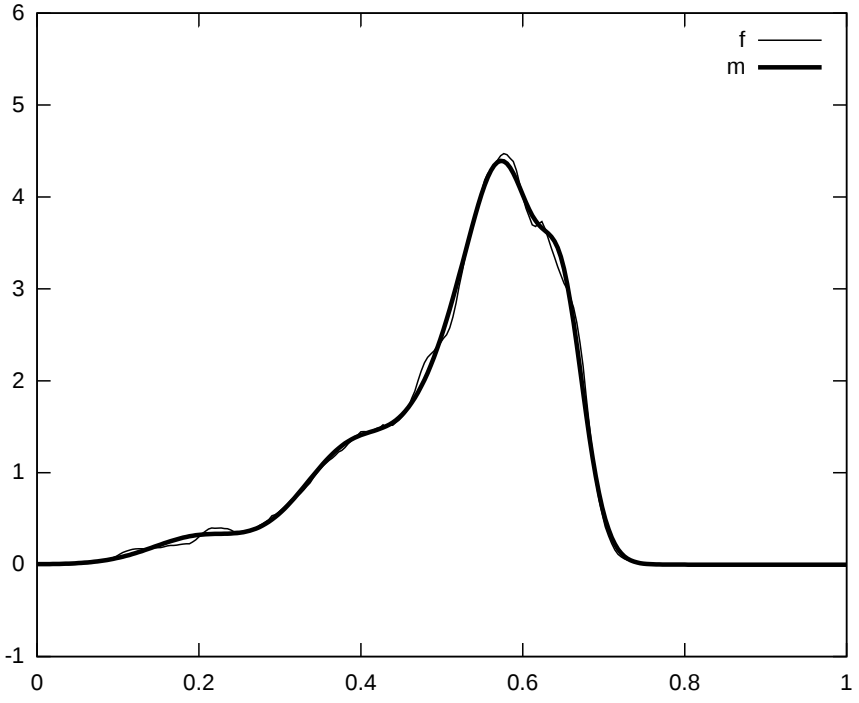
A black and white image (a) and its gray scales distribution (b).



MFG clustering and the corrispondig mixture (K=2)
Grey level corrisponds to the barycenter of the mixtures.
Each image is reconstructed from the corresponding mixture by simply using the responsibilities $\{\gamma_k\}_{k=1,\ldots,K}$. The pixel x it is mapped to the value μ_{k^*}, where $k^* = \arg\max_{k=1,\ldots,K} \gamma_k(x_p)$.



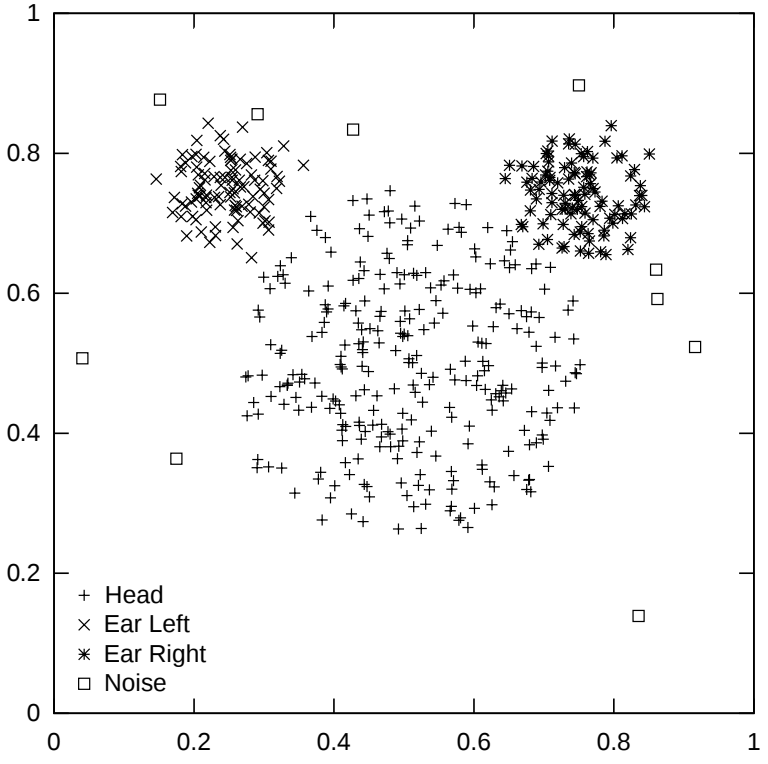
MFG clustering and the corrispondig mixture (K=3)
Grey level corrisponds to the barycenter of the mixtures.



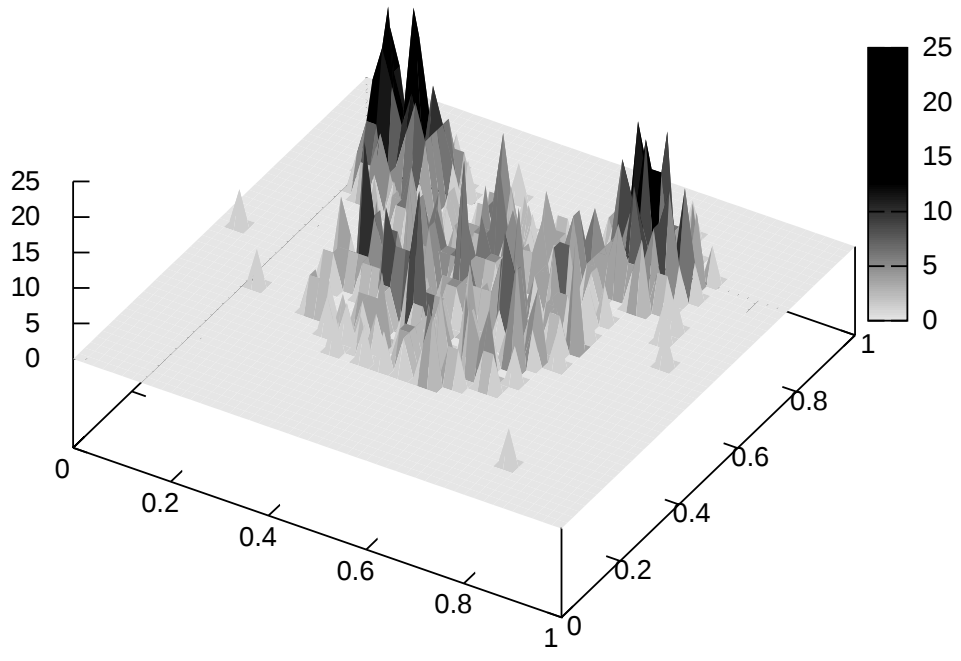
MFG clustering and the corrispondig mixture (K=5)
Grey level corrispond to the barycenter of the mixtures.

Test 4. The Mouse data set

Data from the Elki project, forming a “mouse” similar to a popular comic character. The data set is organized in 3 clusters (plus some random noise), corresponding to the head and the ears of the mouse.



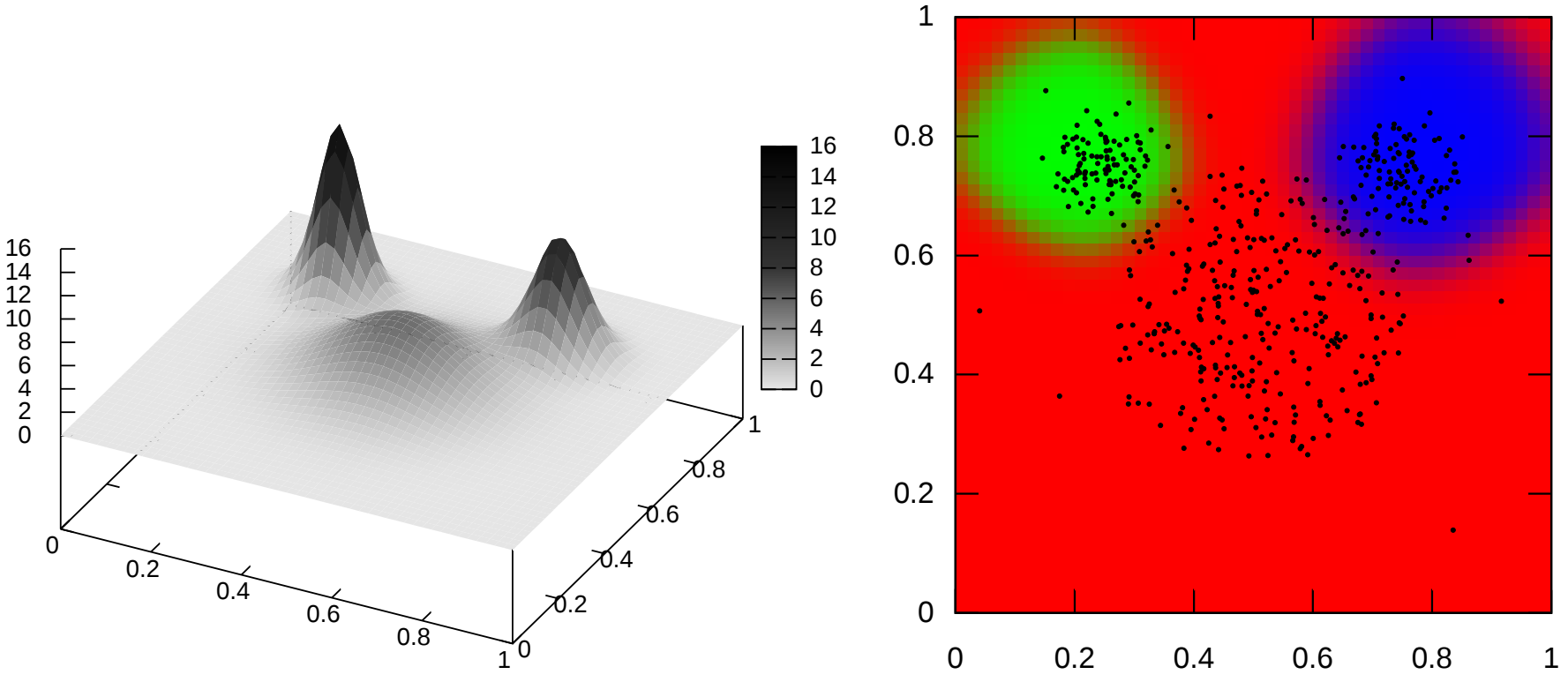
(a)



(b)

The “mouse” data set (a) and the corresponding distribution (b).

For the visual representation, we consider RGB triplets in $[0,1]^3$, and we assign to the three clusters the pure colors red $(1,0,0)$, green $(0,1,0)$ and blue $(0,0,1)$ respectively. Then we use the responsibilities $\{\gamma_k\}_{k=1,2,3} \in [0,1]$ to compute the color of each cell of the grid.



MFG mixture (a) and clustering (b) of the "mouse" data set for $K = 3$.

The MFG approach to cluster analysis: the Bernoulli case

In the previous model, the data set is represented by the samples of a continuous r.v. taking values in $\mathbb{R}^d$. Now we consider a data set $\mathcal{X} = \{x_1, \ldots, x_N\}$ generated by a discrete r.v. taking a finite number of values S, i.e. $x_i \in \{0, \ldots, S\}$.
As before, the aim is to find a mixture model

$$\pi(x) = \sum_{k=1}^{K} \alpha_k \pi_k(x; \theta_k), \quad \text{with } \alpha_k \in [0,1], \ \sum_{k=1}^{K} \alpha_k = 1$$

which gives the best representation of $\mathcal{X}$.

For a Bernoulli mixture model:
$S = 2$, $p_k(x; \theta_k)$ are Bernoulli distributions with $\theta_k = \mu_k$

We introduce the K-populations finite state ergodic MFG system

$$\begin{cases} V_k(i) = \min\limits_{P_i:\, P_{ij}\geq 0, \sum_j P_{ij}=1} \Big\{ \sum\limits_{j=1}^{S} P_{ij}\left(c(P_{ij}) + \varepsilon \log(P_{ij}) + F(i,\theta_k) + V_k(j)\right) \Big\} - \lambda_k, \\ \pi_k(i) = \sum_{j=1}^{S} P^k_{ji}\pi_k(j), \\ \pi_k(i) \geq 0,\ \sum_{i=1}^{S} \pi_k(i) = 1,\ \sum_{i=1}^{S} V_k(i) = 0, \\ \alpha_k = \frac{1}{N}\sum_{n=1}^{N} \gamma_k(x_n), \qquad i \in \{1,\dots,S\} \end{cases}$$

$$P^k_i = \arg\min_{P_i:\, P_{ij}\geq 0, \sum_j P_{ij}=1} \Big\{ \sum_{j=1}^{S} P_{ij}\left(c(P_{ij}) + \varepsilon \log(P_{ij}) + F(i,\theta_k) + V_k(j)\right) \Big\}$$

Bernoulli parameter: $\theta_k = \dfrac{\sum_{n=1}^{N} \gamma_k(x_n) x_n^d}{\sum_{n=1}^{N} \gamma_k(x_n)}, \quad d = 1,\dots,D$

fraction on total mass: $\gamma_k(x_n) = \dfrac{\alpha_k \pi_k(x_n)}{\pi(x_n)} \qquad k = 1,\dots,K,\ x_n \in \mathcal{X}.$

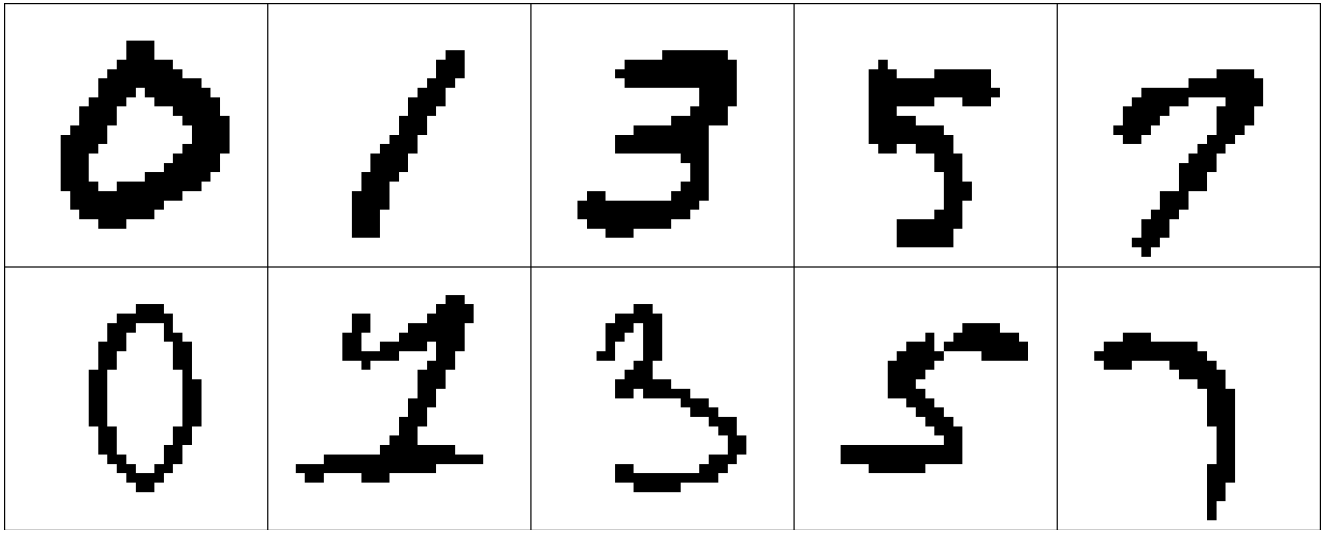
The vector $\theta_k \in \mathbb{R}^S$ represents the average value of the data set with respect to the distribution π_k and interaction among the sub-populations is encoded in the weights α_k and in the coupling cost $F(i,\theta_k)$

More generally, $X = (X^1, \ldots, X^D)$ can be a D-dimensional vector, each component taking S different values. In this case, V_k, π_K, λ_k are D-dimensional vectors.
For example, for a Bernoulli mixture model, we have a 2-states K-populations MFG system for each $d = 1, \ldots, D$

$$
\begin{cases}
V_k^d(0) = \min_{p\in[0,1]}\{p(-\frac{1-p}{2} + \varepsilon \log(p) + V_k^d(0)) \\
\qquad +(1-p)(-\frac{p}{2} + \varepsilon \log(1-p) + V_k^d(1))\} - \lambda_k^d + (\theta_k^d)^2 \\
V_k^d(1) = \min_{q\in[0,1]}\{(1-q)(-\frac{q}{2} + \varepsilon \log(1-q) + V_k^d(0)) \\
\qquad +q(-\frac{1-q}{2} + \varepsilon \log(q) + V_k^d(1))\} - \lambda_k^d + (1-\theta_k^d)^2 \\
\pi_k^d(0) = p\pi_k^d(0) + (1-q)\pi_k^d(1) \\
\pi_k^d(1) = (1-p)\pi_k^d(0) + q\pi_k^d(1) \\
\pi_k^d \geq 0,\ \sum_{x\in\{0,1\}} \pi_k^d(x) = 1,\ \sum_{x\in\{0,1\}} V_k^d(x) = 0 \\
\alpha_k = \frac{1}{N}\sum_{n=1}^N \gamma_k(x_n)
\end{cases}
$$

Example: a dataset of handwritten digits

We consider as dataset the MNIST database of handwritten digits, containing 60000 images of the digits $\{\mathbf{0}, \ldots, \mathbf{9}\}$, each composed by 28×28 pixels of monochrome images, turned in binary vectors of size $D = 784$.

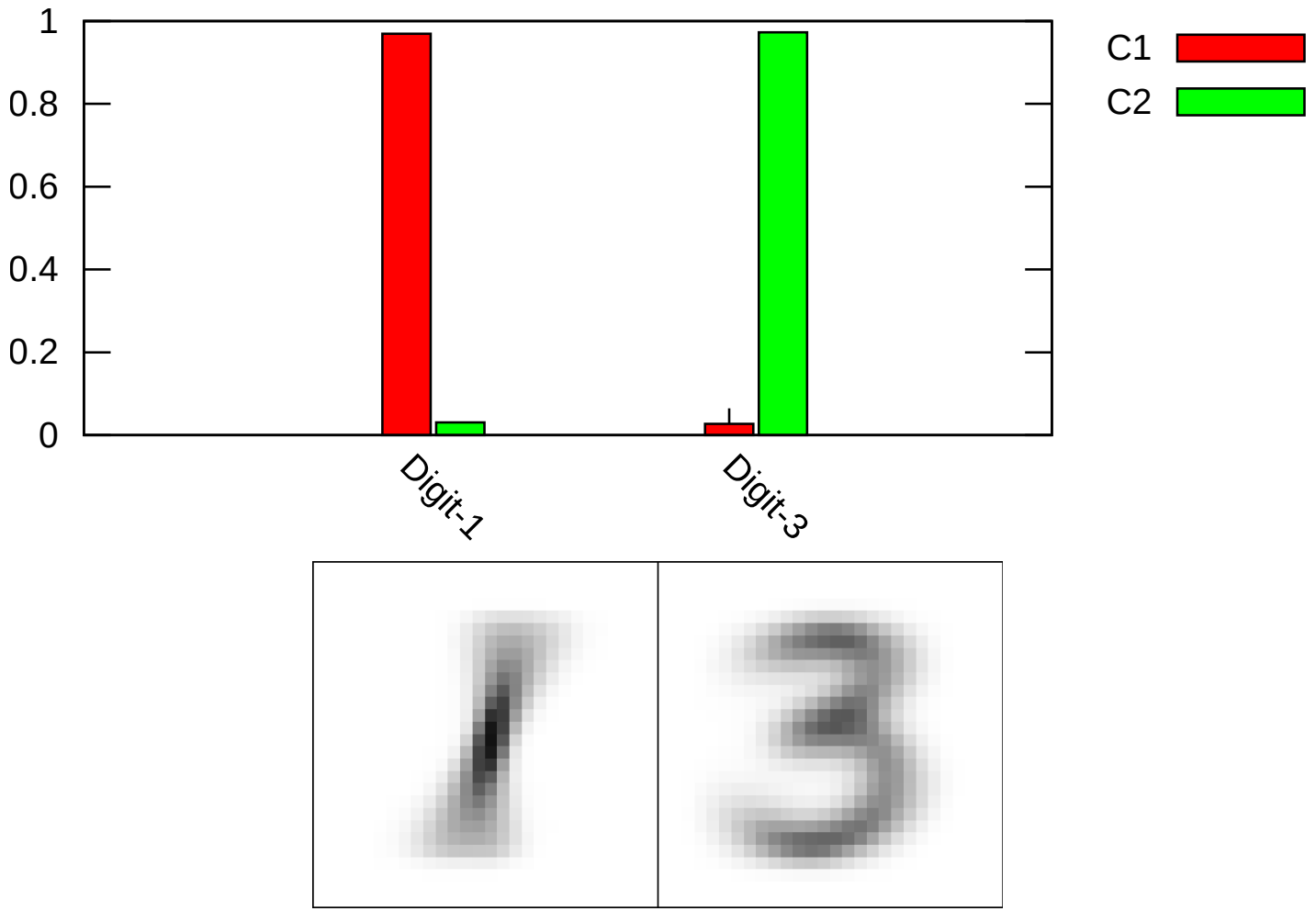


Different samples of hand-written digits from the MNIST database. Each sample is labelled by the number of the corresponding digit, to check the correctness of the clusterization

To cluster the images, we use a finite state MFG with Bernoulli distribution, i.e. $S = 2$, and number of components $D = 784$

Test 1: Digits 1 and 3

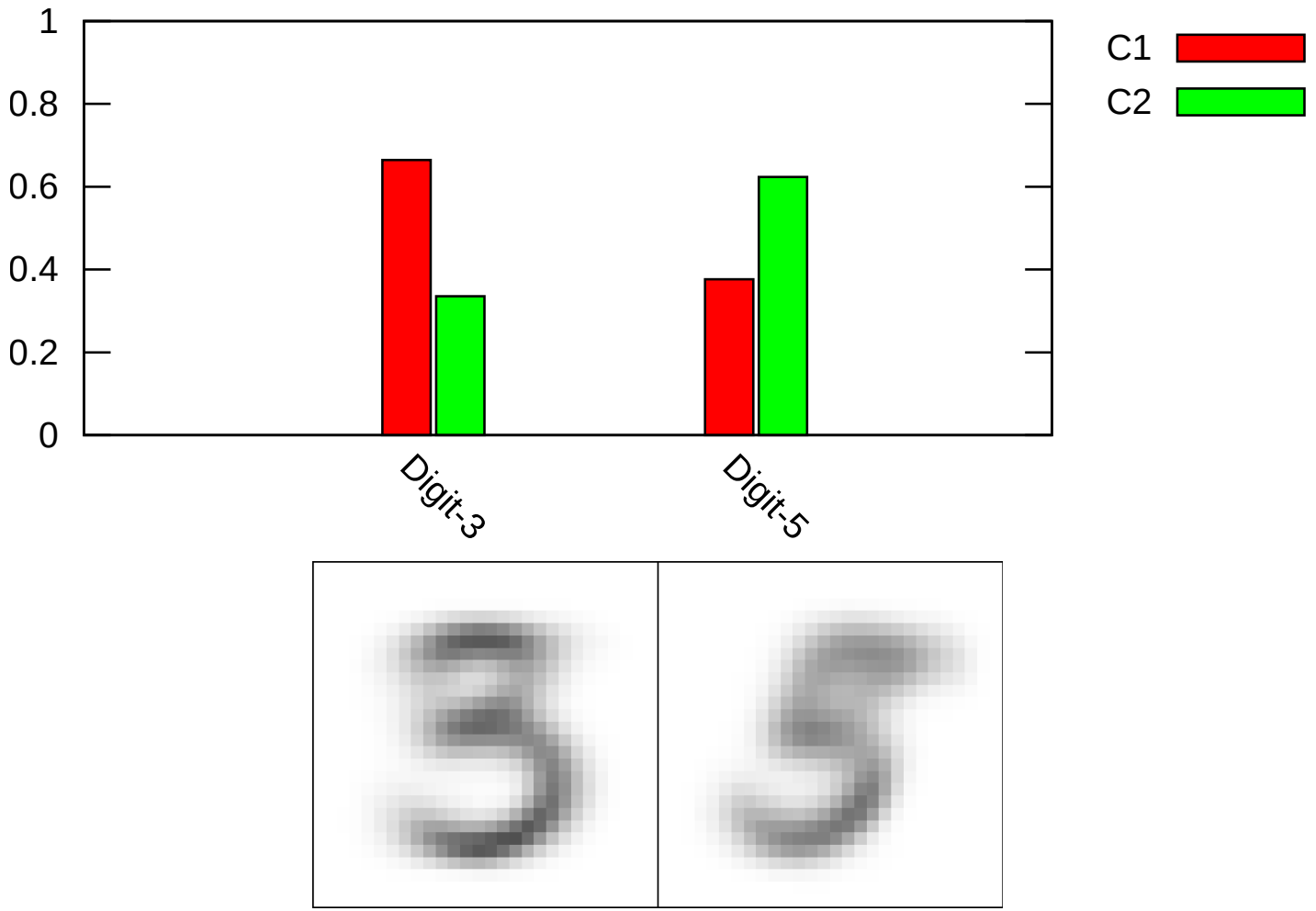
Digits **1**, **3** with $K = 2$. In Figure 5, the clusterization histogram and the corresponding Bernoulli parameters.



Clusterization histogram and the corresponding Bernoulli parameters.

Test 2: Digits 3 and 5

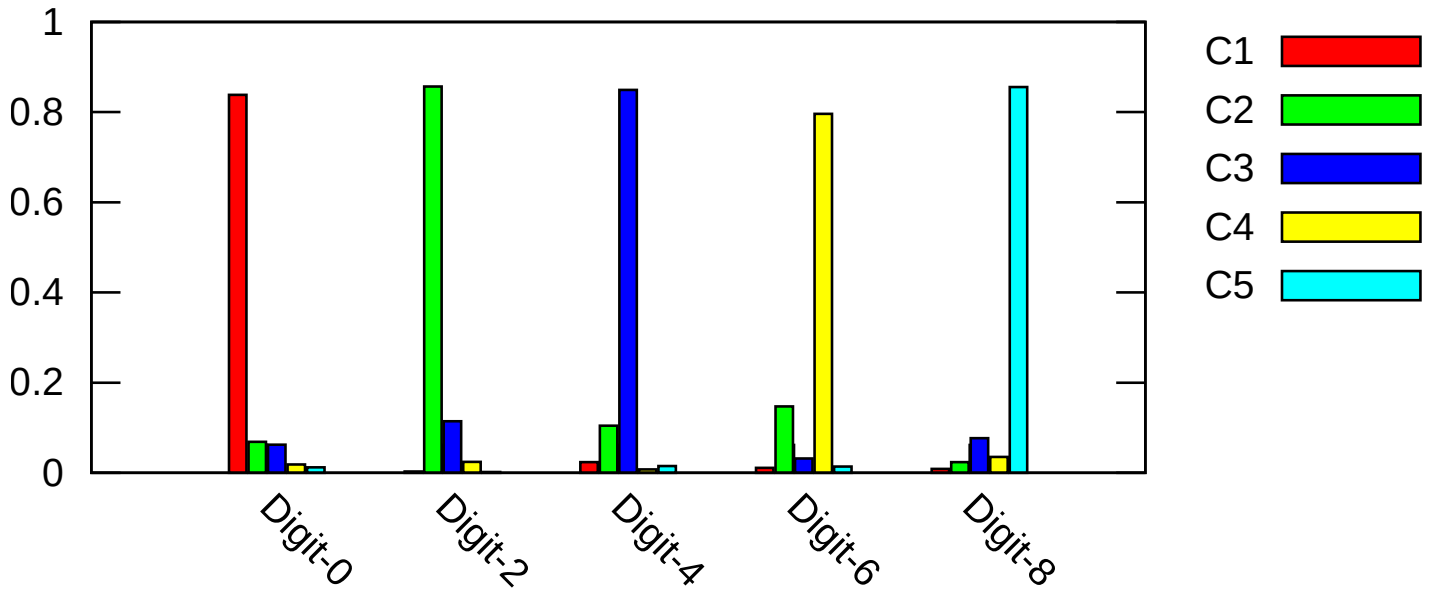
Same example with the digits 3 and 5. The clusterization is slightly ambiguous, since, in average, the samples of the two types are more similar to each other.

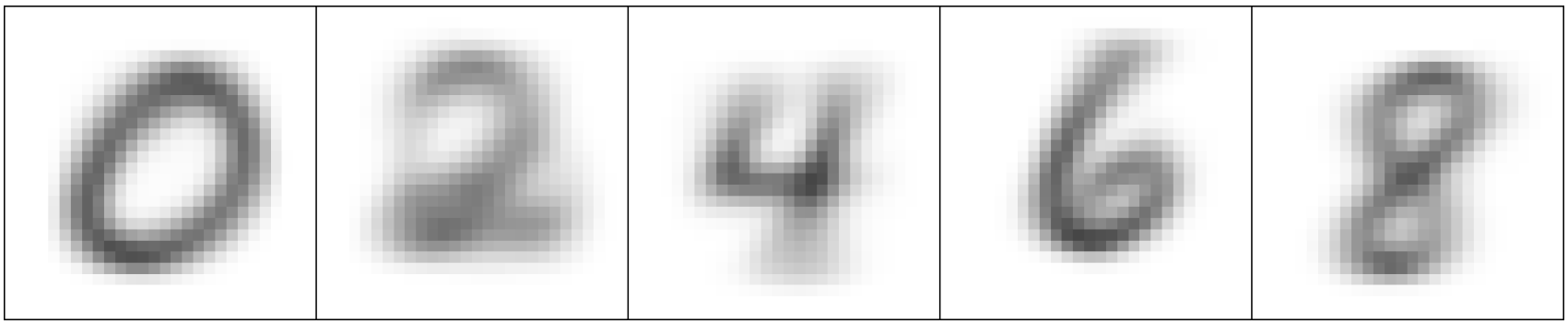


Clusterization histogram and the corresponding Bernoulli parameters.

Test 3: Even digits with $K = 5$

Consider the case $K = 5$ with **0**, **2**, **4**, **6**, **8**. In Figure 7, we observe that the chosen digits are, in average, different from each other, so that they are quite well clusterized.





Clusterization histogram for even digits and the corresponding Bernoulli

The MFG approach to cluster analysis: hard clustering

Let $\mathcal{X}$ be a **big data set** described by a probability density function $f : \mathbb{R}^d \to \mathbb{R}$, $\int_{\mathbb{R}^d} f(x)dx = 1, f(x) \geq 0$ and $supp\{f\}$ contained in a bounded set Ω.

Aim: Subdivide a data set into K clusters such that each data point belongs to the cluster with the nearest barycenter.

Heuristic derivation of the MFG system for the Hard Clustering problem

It is well known, in classical cluster theory, that the hard clustering K-means problem can be seen as the limit of Gaussian mixture model when the variance parameter of the mixture model goes to 0.

We exploit a similar idea to deduce a PDE system for hard clustering: we pass to the limit in the soft clustering MFG system

$$\begin{cases} -\varepsilon \Delta u_k(x) + \frac{1}{2}|Du_k(x)|^2 + \lambda_k = \frac{1}{2}(x-\mu_k)^t(\Sigma_k^{-1})^t(\Sigma_k^{-1})(x-\mu_k), \\ \varepsilon \Delta m_k(x) + \mathrm{div}(m_k(x)Du_k(x)) = 0, \\ \alpha_k = \int_{\mathbb{R}^d} \gamma_k(x) f(x) dx, \\ m_k \geq 0, \int m_k(x)dx = 1, u_k(\mu_k) = 0, \end{cases}$$

for $\Sigma_k = \sigma I$ and for $\varepsilon/\sigma^2 \to 0$.

Eliminating in the limit system the densities m_k which reduce to Dirac functions at the barycenter μ_k, we get the the system of K first order HJ equations

$$\begin{cases} |Du_k| = 1 \quad x \in \mathbb{R}^d, \\ u_k(\mu_k) = 0, \\ \mu_k = \frac{\int_{\mathbb{R}^d} x \mathbb{1}_{S^k}(x) f(x) dx}{\int_{\mathbb{R}^d} \mathbb{1}_{S^k}(x) f(x) dx}, \\ S^k = \{x \in \mathbb{R}^d : \ u_k(x) = \min_{j=1,\dots,K} u_j(x)\} \end{cases}$$

The coupling among the equation is through the sets S^k.

Consider the continuous K-means functional

$$\mathcal{I}(y_1, \ldots, y_k) = \sum_{k=1}^{K} \int_{V(y_k)} |x - y_k|^2 f(x) dx,$$

where $V(y_k) = \{x \in \mathbb{R}^d : |x - y_k| = \min_{j=1,\ldots,K} |x - y_j|\}$.

Theorem

(i) Let $(y_1, \ldots, y_K)$ be a critical point of the functional $\mathcal{I}$ with clusters $V(y_k)$. Then, there exists a solution of the system of HJ equations such that $\mu_k = y_k$ and $S^k = V(y_k)$.

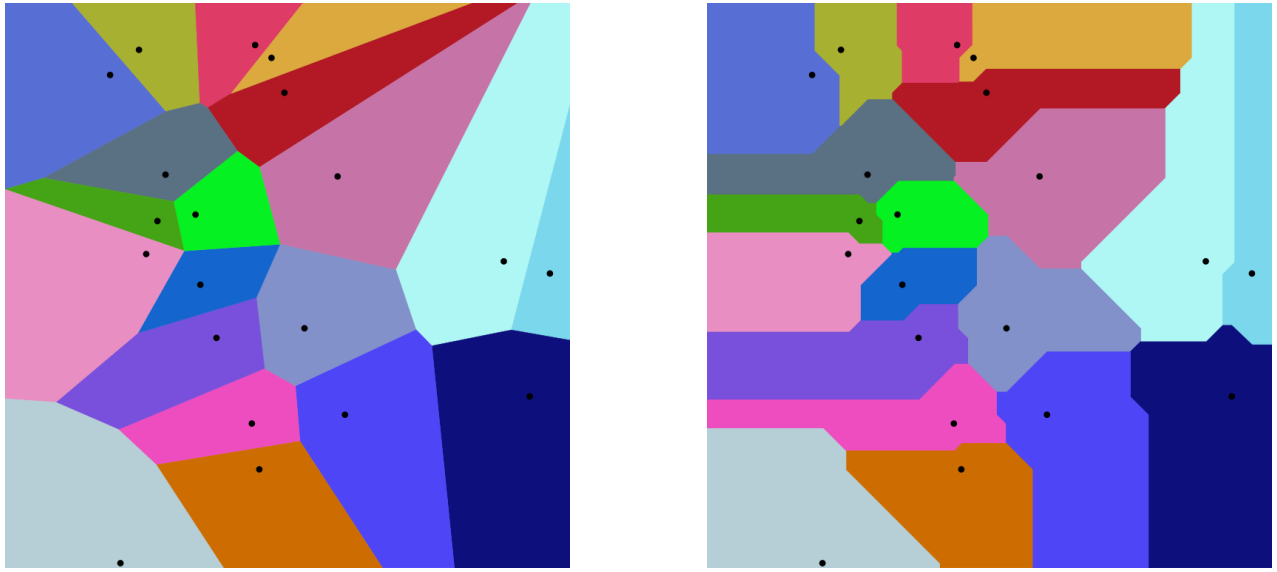
(ii) Given a solution $u = (u_1, \ldots, u_K)$ of of the system of HJ equations, then $(\mu_1, \ldots, \mu_K)$ is a critical point of $\mathcal{I}$ with clusters $V(y_k) = S^k$.

A geometric interpretation of the K-means problem

Given a set of generators $\{y_k\}_{k=1}^K$, $y_k \in \overline{\Omega}$, the Voronoi region corresponding to y_k is defined by

$$V(y_k) = \{x \in \Omega : \ d(x, y_k) = \min_{j=1,\ldots,K} d(x, y_j)\}$$

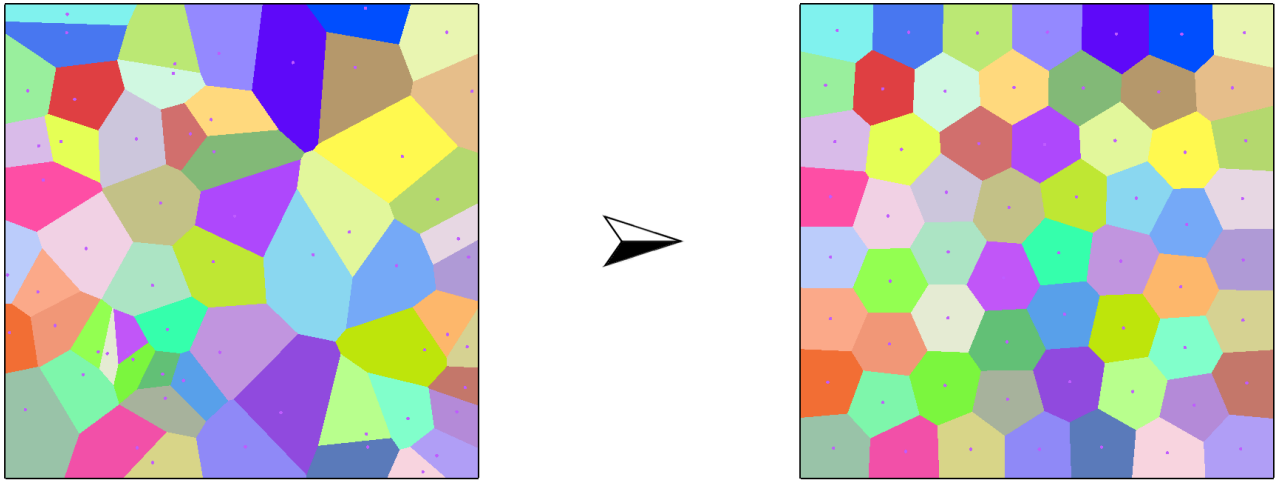
and the family $\{V(y_k)\}_{k=1}^K$ determines a tessellation of Ω.



Voronoi tesselation: Left: Euclidean distance; Right: Manhattan distance.

Definition: A Voronoi tessellation $\{V(y_k)\}_{k=1}^{K}$ of Ω is said to be a **centroidal Voronoi tessellation (CVT)** if, for each $k = 1, \ldots, K$, the generator y_k of $V(y_k)$ coincides with the centroid of $V(y_k)$, i.e.

$$y_k = \frac{\int_{V(y_k)} s ds}{\int_{V(y_k)} ds}.$$



Left: Voronoi tessellation; Right: Centroidal Voronoi tessellation.

K-Means problem and centroidal Voronoi tesselations

It can be proved that critical points of the K-means functional

$$\mathcal{I}(y_1, \ldots, y_k) = \sum_{k=1}^{K} \int_{V(y_k)} |x - y_k|^2 dx,$$

where $V(y_k) = \{x \in \mathbb{R}^d : |x - y_k| = \min_{j=1,\ldots,K} |x - y_j|\}$

are the generators of Centroidal Voronoi Tessellation with

$$V(y_k) = \{x \in \mathbb{R}^d : |x - y_k| = \min_{j=1,\ldots,K} |x - y_j|\}$$

A MFG version of the Lloyd's algorithm

- (**Inizialization**) Given an initial guess $(\mu^{(0),1}, \dots, \mu^{(0),k})$:
- (**E-step**) Solve the K (uncoupled) HJ equations

$$\begin{cases} |Du_k^{(n)}| = 1, \\ u_k^{(n)}(\mu^{(n),k}) = 0, \end{cases}$$

for $k = 1, \dots, K$ and compute the Voronoi diagrams

$$S_u^{k,(n)} = \{x \in \Omega : \ u_k^{(n)}(x) = \min_{j=1,\dots,K} u_j^{(n)}(x)\}, \quad k = 1, \dots, K.$$

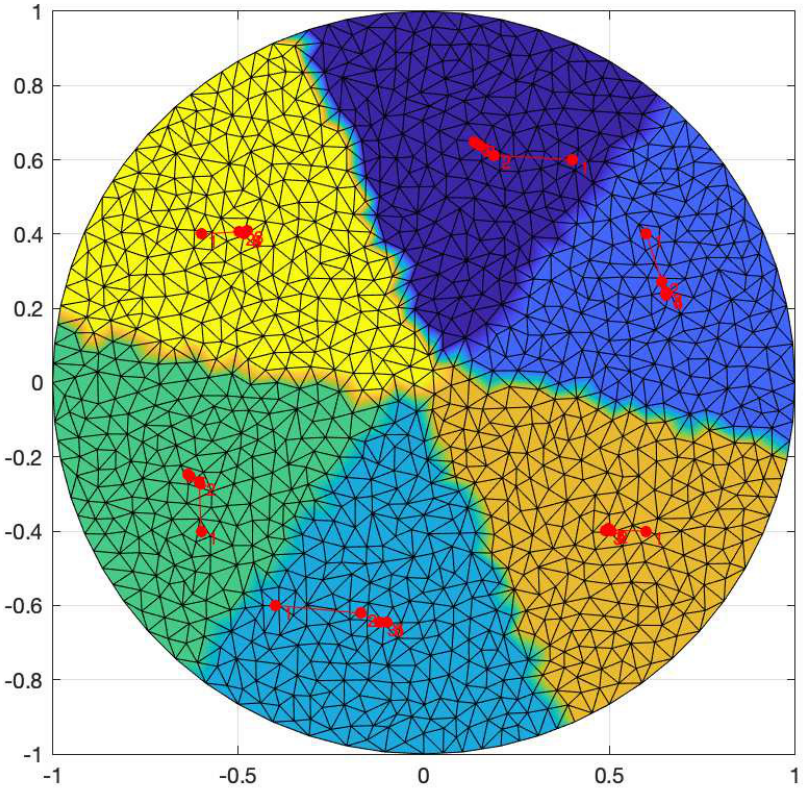
- (**M-step**) Compute the new centroids

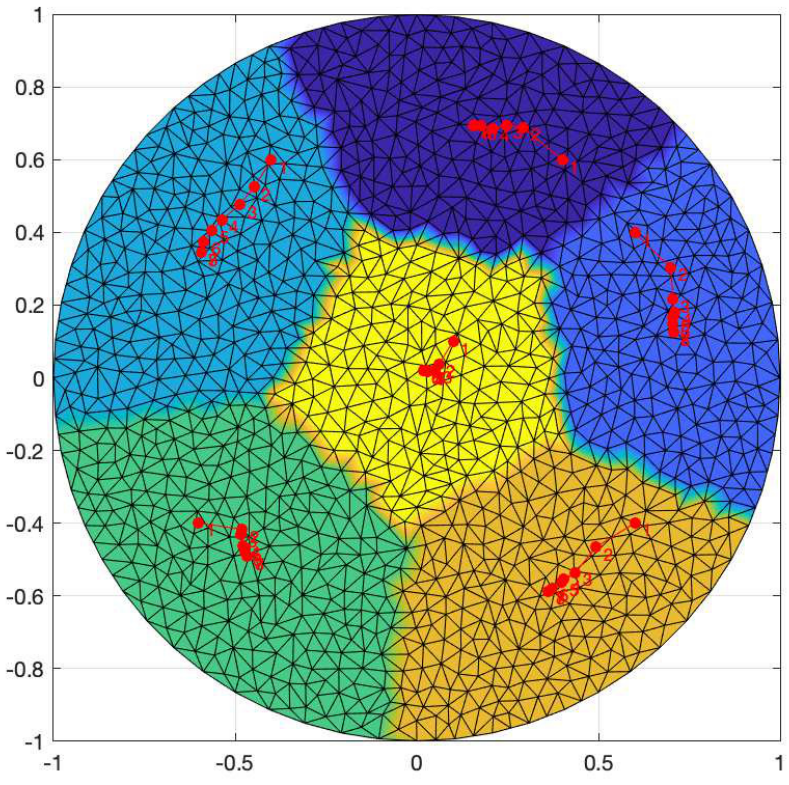
$$\mu^{(n+1),k} = \frac{\int_{\mathbb{R}^d} x \mathbb{1}_{S^{k,(n)}}(x) f(x) dx}{\int_{\mathbb{R}^d} \mathbb{1}_{S^{k,(n)}}(x) f(x) dx}$$

In the first step of the iterative procedure, it is sufficient to solve the problem in the set Ω, the support of the density f.

Test 1:

The density function f is given by a uniform distribution on Ω,





Two Voronoi tessellations with $K = 6$ computed starting from different initial centroids, above/left: $\mu^{(0)} = ([0.4, 0.6], [0.6, 0.4], [0.6, -0.4], [-0.4, -0.6], [-0.6, -0.4], [-0.6, 0.4])$; above/right: $\mu^{(0)} = ([0.4, 0.6], [0.6, 0.4], [0.6, -0.4], [-0.6, -0.4], [-0.4, 0.6], [0.1, 0.1])$

Geodesic Centroidal Voronoi Tesselations

The previous approach can be extended to centroidal Voronoi tessellations related to a **general convex metric d**.

Definition A geodesic Voronoi tessellation $\{V(y_k)\}_{k=1}^K$

$$V(y_k) = \{x \in \Omega : \mathbf{d}(x, y_k) = \min_{j=1,\dots,K} \mathbf{d}(x, y_j)\}$$

is said to be a **geodesic centroidal Voronoi tessellation** if the generator y_k of $V(y_k)$ coincides with the centroid of $V(y_k)$, i.e.

$$\int_{V(y_k)} \mathbf{d}(y_k, x) f(x) dx = \min_{z \in V(y_k)} \int_{V(y_k)} \mathbf{d}(z, x) f(x) dx.$$

The corresponding geodesic K-means functional is

$$\mathcal{I}_d(y_1, \dots, y_k) = \sum_{k=1}^{K} \int_{V(y_k)} \mathbf{d}(y_k, x)^2 f(x) dx$$

The MFG system for geodesic centroidal Voronoi tessellations

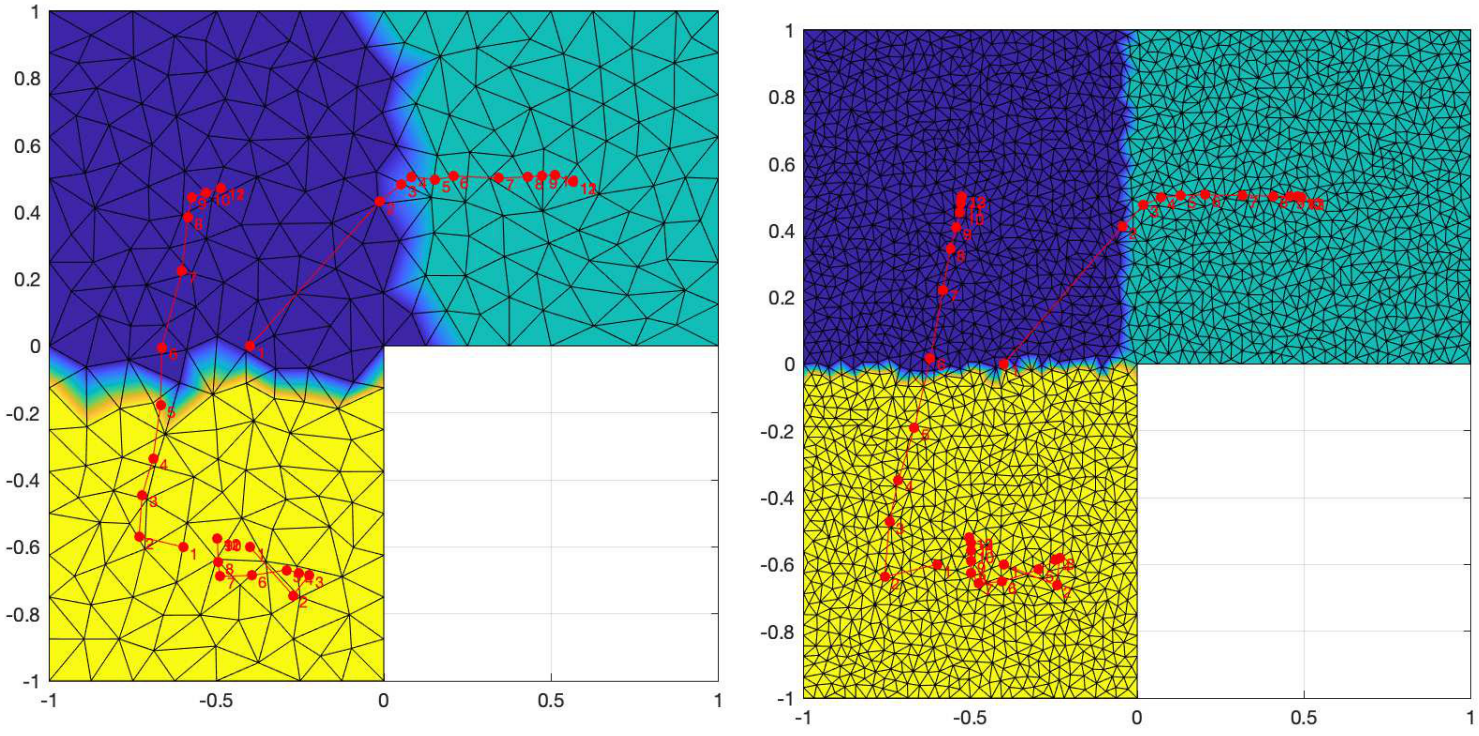
Critical points of the geodesic K-means functional can be characterized by the system of HJ equations

$$\begin{cases} H(x, Du_k) = 1, \qquad x \in \Omega, \\ u_k(\mu_k) = 0, \\ S^k = \{x \in \mathbb{R}^d : \ u_k(x) = \min_{j=1,\dots,K} u_j(x)\}, \\ \int_{S^k} u_k(x) f(x) dx = \min\{\int_{S^k} u_y(x) f(x) dx : u_y \text{ the solution of } H(Du) = 1, \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad u(y) = 0 \text{ with } y \in S^k\} \end{cases}$$

Here H is a convex, positive homogeneous Hamiltonian corresponding to the distance **d**.
For example, if $\mathbf{d}(x, y)$ is the Riemannian distance induced by a positive definite matrix $A(x)$, then $H(x, p) = \sqrt{A(x)p \cdot p}$

Test 2. Chebyshev distance

We consider the Chebyshev distance $\mathbf{d}(x, y) = \max_i(|x_i - y_i|)$ and the density function f given by a uniform distribution on Ω



Above: left panel $\Delta x = 0.01$; right panel: $\Delta x = 0.001$ starting from $\mu^{(0)} = ([-0.6, -0.6], [-0.4, -0.6], [-0.4, 0])$.

Conclusion

- We presented a procedure for computing clusters in hard and soft-clustering analysis by means of a system of PDEs. In some specific case, this approach is an infinite dimensional version of classical methods in finite-dimensional optimization theory
- From a theoretical point of view, it allows to use all the techniques of PDE theory to discover new properties and structures inside cluster analysis;
- From a computational one, the MFG model is very flexible and can be generalized in several directions (different coupling costs)
- It should be possible to interpret other classical algorithms in Machine Learning through the theory of partial differential equations. This is well known in supervised machine learning, but it hasn't been much explored in the unsupervised case

Some references

- Pequito, S., Aguiar, A. P., Sinopoli, B., Gomes, D. A.: Unsupervised learning of finite mixture models using Mean Field Games, in *Annual Allerton Conference on Communication, Control and Computing*, 2011, 321–328,
- J.L. Coron,: *Quelques exemples de jeux à champ moyen*, Ph.D. thesis, Université Paris-Dauphine,2018.
- Aquilanti, L., Cacace, S., Camilli, F., De Maio, R. A Mean Field Games approach to cluster analysis, Appl. Math. Optim. 84 (2021), 299-323
- Aquilanti, L., Cacace, S., Camilli, F., De Maio, R. A Mean Field Games model for finite mixtures of Bernoulli and categorical distributions. J. Dyn. Games 8 (2021), 35-59.
- Camilli, F., Festa, A. A PDE approach to centroidal tessellations of domains, arXiv:2106.10663


Thank You!