*CIMPA Research School : Data Science for Engineering and Technology*
*Tunis 2019*

# Bregman divergences
## a basic tool for pseudo-metrics building for data structured by physics

### 4- Clustering with Bregman divergences

Stéphane ANDRIEUX

*ONERA - France*

*Member of the National Academy of Technologies of France*

# *k*-means algorithm : back to history

Partitioning $n$ observations into $k$ clusters in which each observation belongs to the cluster with the nearest mean.

→

It is a non-supervised learning (except for choosing $k$ !)

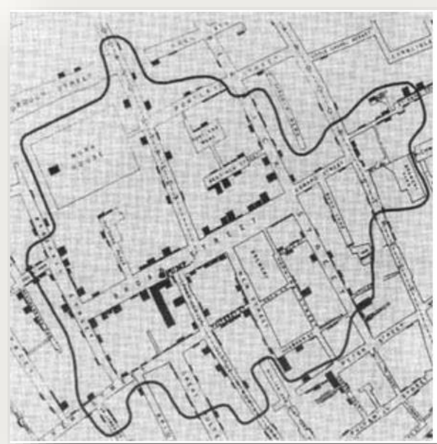
Partition of the data space into Voronoi cells.

1644 *Descartes* 1850 *Dirichlet* 1907 *Voronoi*

Physician John Snow analyzed the 1854 cholera epidemic in London

Each bar represents a death at that address

→



Sources of drinking water, pumps drawing the boundary of equal distance between a pump and other pumps

→

Strong correlation of deaths with proximity to a particular water pump

Identification of the infected pump

# *k*-means algorithm

**Definition: *k*-means clustering**

Given a set $S$ of $n$ observations $(x_1, x_2, \ldots, x_n)$ and a $k$ a given integer much smaller than $n$, $k$-means aims at partition the $n$ observations into $k$ sets $\{S_1, S_2, \ldots, S_k\}$ so as to minimize the within-cluster sum of squares (WCSS),distances of the elements of each set $S_i$ and its centroid $\mu_i$

$$Min \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2 , \quad \mu_i = \arg\min_{\mu} \sum_{x_j \in S_i} \left\| x_j - \mu \right\|^2$$

Or to minimize equivalently $\displaystyle \sum_{i=1,k} \frac{1}{2n_i} \sum_{x_j \in S_i} \sum_{y_l \in S_i} \left\| x_i - y_l \right\|^2$ $\qquad n_i = \text{Card } S_i$

$$\sum_{i=1,n} \left\| x_i - y \right\|^2 = \sum_{i=1,n} \left\| x_i - \mu \right\|^2 + n \left\| y - \mu \right\|^2 \longrightarrow \sum_{i=1,n} \left\| x_i - y \right\|^2 = \sum_{i=1,n} \left\| x_i - \mu + \mu - y \right\|^2$$

$$= \sum_{i=1,n} \left( \left\| x_i - \mu \right\|^2 + \left\| y - \mu \right\|^2 + 2 \langle x_i - \mu, \mu - y \rangle \right)$$

$$= \sum_{i=1,n} \left\| x_i - \mu \right\|^2 + n \left\| y - \mu \right\|^2 + 2 \left\langle \sum_{i=1,n} x_i - n\mu , \mu - y \right\rangle$$

$$\sum_{j=1,n} \sum_{i=1,n} \left\| x_i - y_j \right\|^2 = n \sum_{i=1,n} \left\| x_i - \mu \right\|^2 + n \sum_{j=1,n} \left\| y_i - \mu \right\|^2 = 2n \sum_{i=1,n} \left\| x_i - \mu \right\|^2$$

# *k*-means algorithm

$$Min \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2 \;,\; \mu_i = \arg\min_{\mu} \sum_{x_j \in S_i} \left\| x_j - \mu \right\|^2$$

## Loyd's Algorithm

**Assignment step:**

Assign each observation $x_i$ to a new cluster $S_j^N$ which $\mu_j^N$ has the least distance to $x_i$

Or to minimize equivalently

$$S_j^N = \left\{ x_i \in S, \left\| x_i - \mu_j^N \right\|^2 \leq \left\| x_i - \mu_l^N \right\|^2, \forall l \leq k \right\}$$

**Update step**

Calculate the new centroids $\mu_j^{N+1}$ of the new clusters $S_j^N$

$$\mu_j^{N+1} = \frac{1}{\mathrm{Card}\, S_j^N} \sum_{x_l \in S_j^N} x_l$$

**Convergence criterion**

Based on the evolution of the centroids:

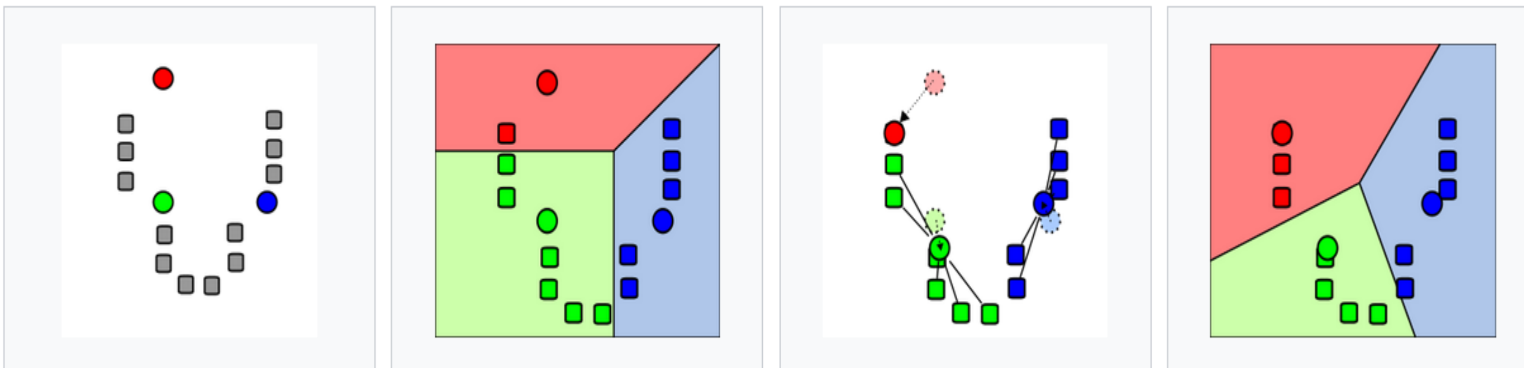
$$\sum_{j=1,k} \left\| \mu_j^{N+1} - \mu_j^{N+1} \right\|^2 \leq \varepsilon_{tol}^2$$

Or to minimize equivalently

# *k*-means algorithm

## Loyd's Algorithm

$$Min \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2 , \quad \mu_i = \arg\min_{\mu} \sum_{x_j \in S_i} \left\| x_j - \mu \right\|^2$$



1. $k$ initial "means" (in this case $k$=3) are randomly generated within the data domain (shown in color).

2. $k$ clusters are created by associating every observation with the nearest mean. The partitions here represent the

3. The centroid of each of the $k$ clusters becomes the new mean.

4. Steps 2 and 3 are repeated until convergence has been reached.

## Indicators

By Huyghens theorem

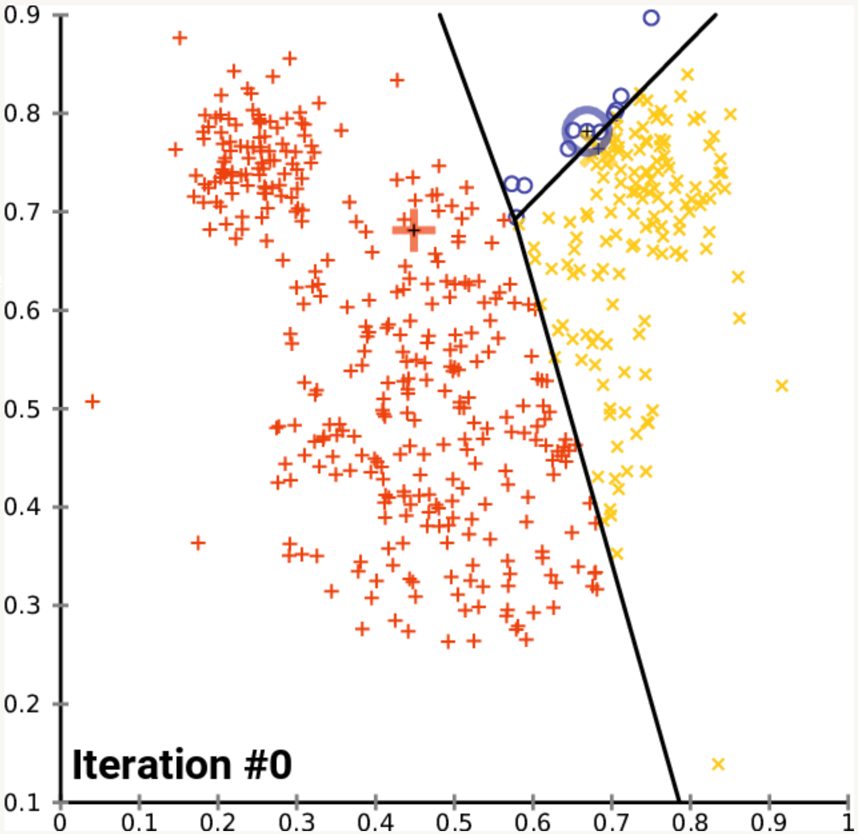
$$\sum_{x_i \in S} \left\| x_i - \mu \right\|^2 = \sum_{i=1}^{k} n_i \left\| \mu_i - \mu \right\|^2 + \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2$$

*Clusters separability indicator* *Clusters compacity indicator.*

# $k$-means algorithm

$$Min\sum_{i=1}^{k}\sum_{x_j\in S_i}\left\|x_j-\mu_i\right\|^2\ ,\ \ \mu_i=\arg\min_{\mu}\sum_{x_j\in S_i}\left\|x_j-\mu\right\|^2$$

Or to minimize e




Loyd's Algorithm

# *k*-means algorithm : the problem of initialization

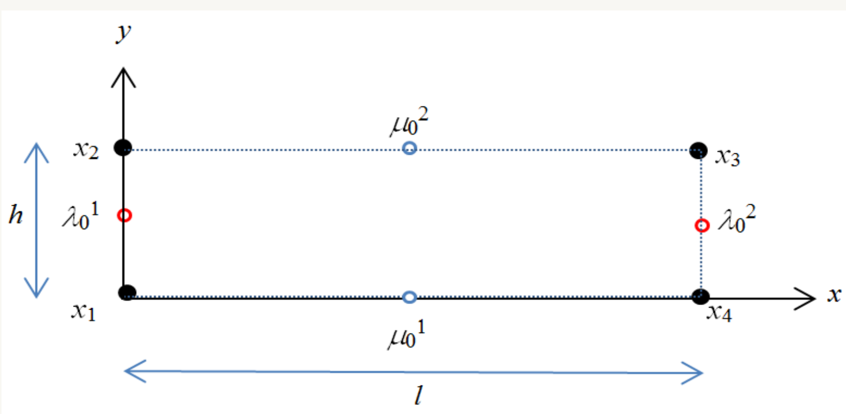
$$Min\sum_{i=1}^{k}\sum_{x_j\in S_i}\left\|x_j-\mu_i\right\|^2 \ ,\ \mu_i=\arg\min_{\mu}\sum_{x_j\in S_i}\left\|x_j-\mu\right\|^2$$

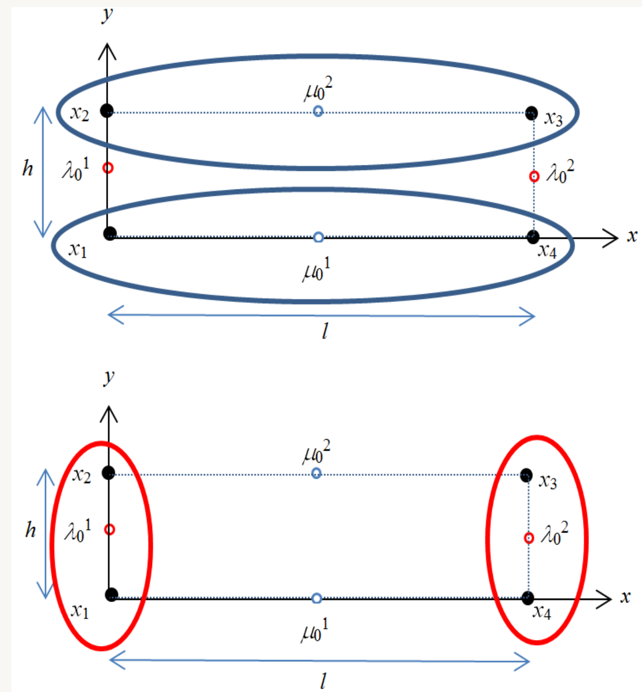
Loyd's Algorithm necessitates an initialization of the *k* first centroids

And is very sensitive to the initialization

Or to minimize equivalently

Simple example with one iteration convergence
and two different initialization



# *k*-means algorithm : the problem of initialization

$$Min\sum_{i=1}^{k}\sum_{x_j\in S_i}\left\|x_j-\mu_i\right\|^2\ ,\ \ \mu_i=\underset{\mu}{\arg\min}\sum_{x_j\in S_i}\left\|x_j-\mu\right\|^2$$

Better initialization than random initialization the *k-means*++ algorithm

Choose the first center $\mu_1^0$ uniformly at random within the data set $S$

Or to minimize equivalently

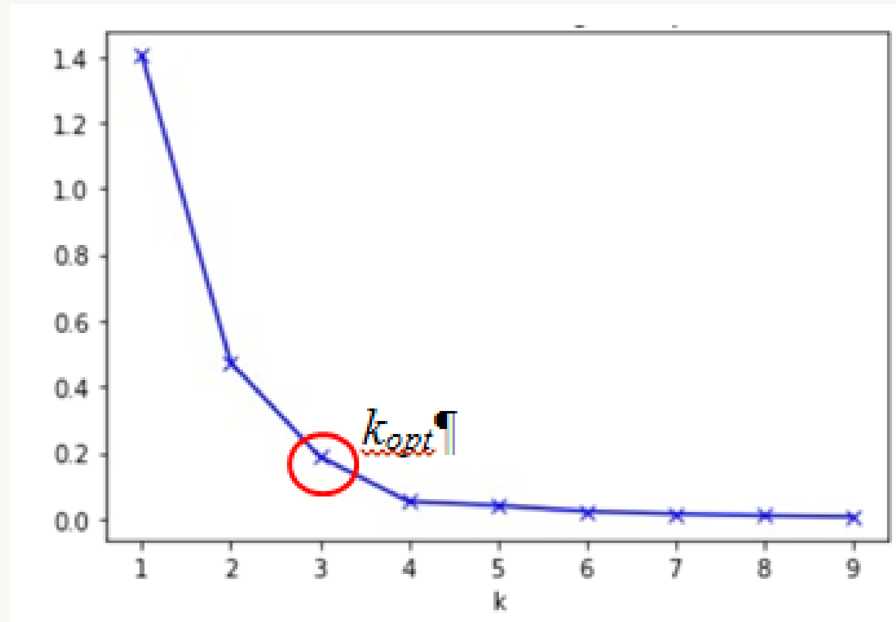
For each data point $x_j$ in S, compute $\left\|x_j-\mu_1^0\right\|^2$

Choose the new center $\mu_2^0$ at random in $S$, using the weighted probability distribution proportional to $\left\|x_j-\mu_1^0\right\|^2$

Repeat until $k$ centers have been chosen

# $k$-means algorithm : the problem of choosing $k$

$$CC\left(\{S_i\}_{i=1,k}\right) = \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2$$



# $k$-medoids algorithm

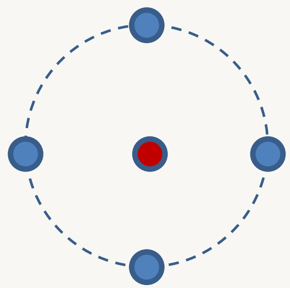
**Definition** Medoid of a finite set of points a distance $d$.

The medoid $\bar{\mu}_d$ of a set of $N$ points of IR$^n$, $S$ with respect the distance $d$ is the point belonging to $S$

$$\bar{\mu}_d = \underset{s \in S}{\arg\min} \sum_{i=1,N} d(x_i, s) \qquad \neq \qquad \mu = \frac{1}{n} \sum_{i=1,n} x_i$$

Centroid for $d(x,y) = \|x-y\|^2$

Or to minimize equivalently



Pathologic case!

**Definition: $k$-medoids clustering**

Given a set $S$ of $n$ observations $(x_1, x_2, \ldots, x_n)$, and a $k$ a given integer much smaller than $n$, $k$-medoids aims at partition the $n$ observations into $k$ sets $\{S_1, S_2, \ldots, S_k\}$ so as to minimize the within-cluster sum of squares (WCSS),distances of the elements of each set $S_i$ and its medoid $\mu_i$

$$Min \sum_{i=1}^{k} \sum_{x_j \in S_i} d(x_j - \bar{\mu}_i) \,, \quad \bar{\mu}_i = \underset{\mu \in S}{\arg\min} \sum_{x_j \in S_i} d(x_j, \mu)$$

# $k$-medoids algorithm

## Partitioning Around Medoids (PAM)

<u>Assignment step:</u>

Assign each observation $x_i$ to a new cluster $S_j^N$ which $\bar{\mu}_j^N$ in $S$ has the least distance to $x_i$,

$$S_j^N = \left\{ x_i \in S, \left\| x_i - \bar{\mu}_j^N \right\|^2 \leq \left\| x_i - \bar{\mu}_l^N \right\|^2, \forall l \leq k \right\}$$

<u>Swap step</u>

Or to minimize equivalently

For each cluster $S_j^N$, pick randomly a non-medoid point $x_r^N \neq \mu_j^N$ and recompute the global cost by exchanging $x_r^N$ and $\mu_j^N$

$$E(x_r^N) = \sum_{i \neq h}^{k} \sum_{x_j \in S_i} d(x_j - \bar{\mu}_i^N) + \sum_{x_j \in S_i} d(x_j, x_r^N)$$

If $E(x_r^N) < E(\bar{\mu}_j^N)$, then swap: $x_r^N \rightarrow \mu_j^N$

<u>Convergence criterion</u>

Based on the non-decreasing of $E$

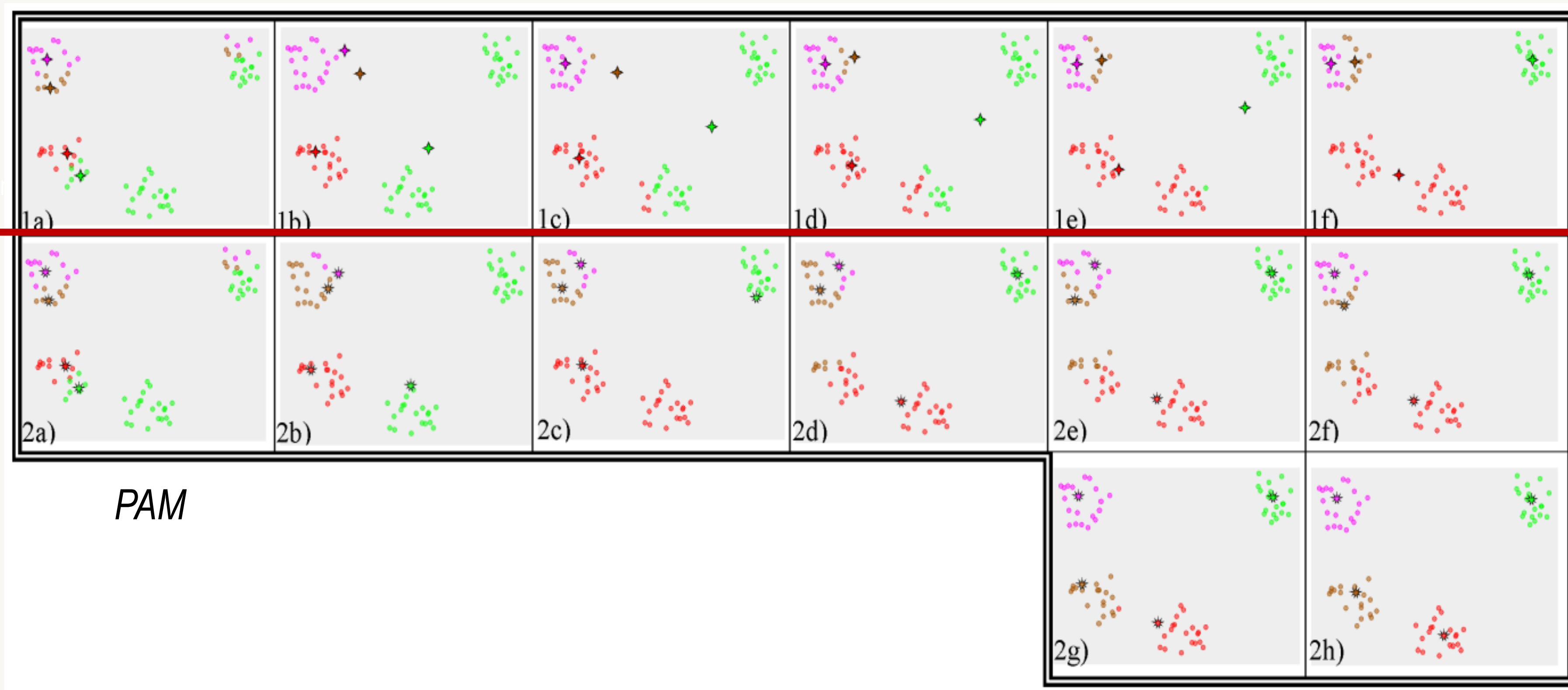
# *k*-medoids algorithm

K-means ++ versus (PAM)

The benefit of using *k*-medoids

(in this case)

*k-means ++*



# Clustering with Bregman divergence

## Probabilistic framework

$X$ a random variable that takes values in a finite set $\mathcal{X} = \{x_i\}_{i=1,n}$

Minimizing the *global distortion* $$\mu = \underset{s \in S}{\arg\min}\, E_v\left[D_J(X,s)\right] = \underset{s \in S}{\arg\min} \sum_{i=1,n} v_i\, D_J(x_i, s)$$

Characterization of *BG* $$\mu = \frac{1}{n}\sum_{i=1,n} v_i x_i$$ Independent of $D_J$

But the *min* still depends on $D_J$

Minimizing the Bregman information of the random variable $X$

$$I_J(X) = E_v\left[D_J(X,\mu)\right] = \min_{s \in S} \sum_{i=1,n} v_i\, D_J(x_i, s)$$

If $M$ is the random variable representing the initial $X$,
$M$ also minimizes the *loss in Bregman Information* $$L_J(M) = I_J(X) - I_J(M)$$

$M$ random variable taking in the finite set $\mathcal{M} = \{\mu_h\}_{h=1,k}$ Induced probability distribution $\pi_h = \sum_{i\, s.t.\, x_i \in X_h} v_i$

# Clustering with Bregman divergence

## Algorithm

Assignment step:

Assign each $x_i$ to a new cluster which has the least Bregman divergence distance to $x_i$,

$$X_h^N = \left\{ x_i \in X, D_J(x_i, \mu_h^N) \le D_J(x_l, \mu_h^N), \forall l \le k \right\}$$

Or to minimize equivalently

Update step

Calculate the new centroid of the new clusters and the corresponding induced probability distributions :

$$\pi_h^{N+1} = \sum_{l\, s.t.\, x_l \in X_h^N} v_l \quad , \quad \mu_h^{N+1} = \frac{1}{\pi_h^N} \sum_{x_l \in X_h^N} x_l$$

Convergence criterion

Based on the evolution of the centroids: $\sum_{h=1,k} \left\| \mu_h^{N+1} - \mu_h^{N+1} \right\|^2 \le \varepsilon_{tol}^2$

Thanks for your attention