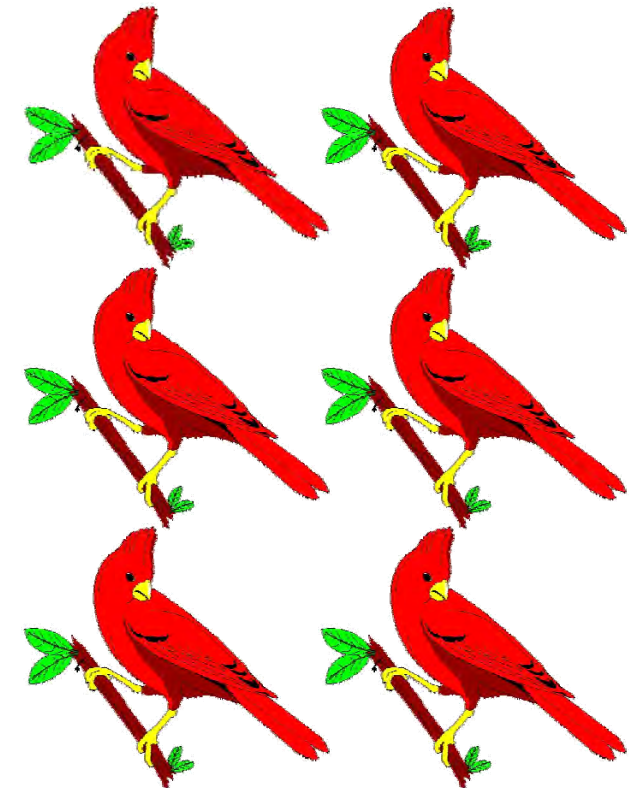
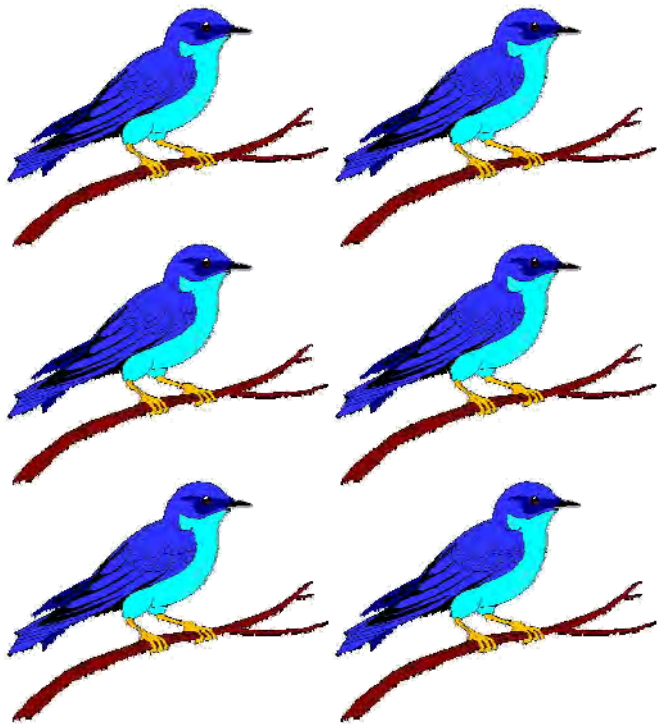


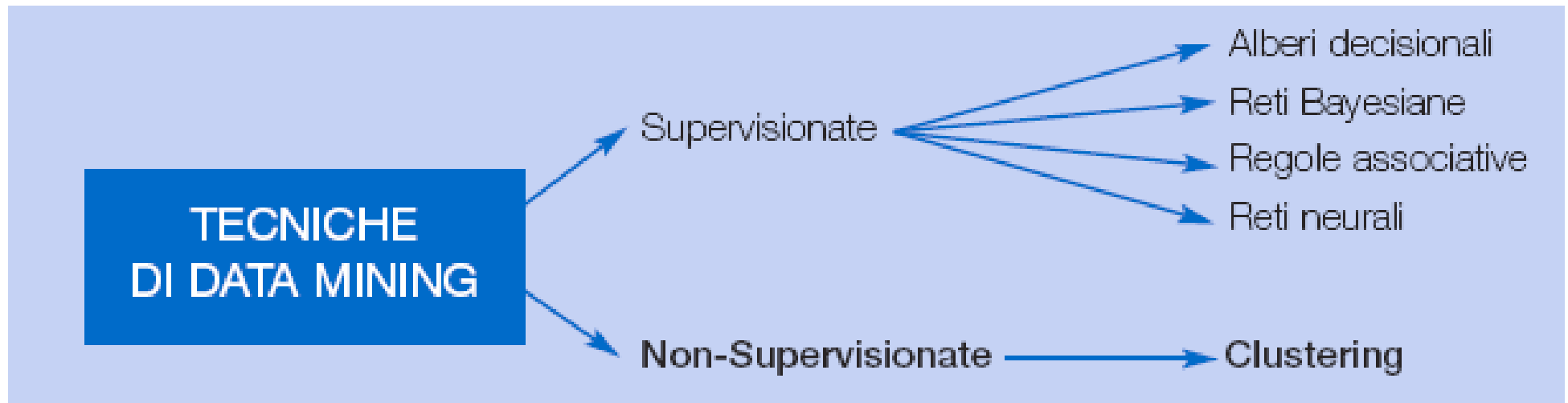
FUZZY CLUSTERING

CLUSTERING: Raggruppamento di oggetti per similarità
(*hard clustering*)

FUZZY CLUSTERING: Modulazione del grado di similarità (può essere parziale)
(*soft clustering*)



Cosa è il Clustering



- 👉 Il **Clustering** (*cluster analysis*, R. Tryon 1939) è una tecnica di *data mining* per il raggruppamento di elementi di un insieme di dati.
- 👉 Si basa sul concetto di distanza tra due elementi. Perciò è necessario definire una metrica. La sua scelta influenza molto il risultato
- 👉 Gli algoritmi di *clustering* raggruppano gli elementi sulla base della loro distanza reciproca cercando di definire raggruppamenti più o meno omogenei

Definizione delle metriche più usate

👉 **Distanza euclidea:** somma delle distanze fra \mathbf{x} e \mathbf{y} al quadrato, eventualmente pesate (\mathbf{w})

$$d_2(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)$$

👉 **Distanza assoluta** (city block, Manhattan): differenza media

$$d_1(\mathbf{x}_i, \mathbf{x}_j) = |\mathbf{x}_i - \mathbf{x}_j|$$

👉 **Distanza di Mahalanobis:** norma euclidea pesata dalla matrice di covarianza delle variabili \mathbf{x}_i e \mathbf{x}_j

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{w} (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{W}^{-1} (\mathbf{x}_i - \mathbf{x}_j)$$

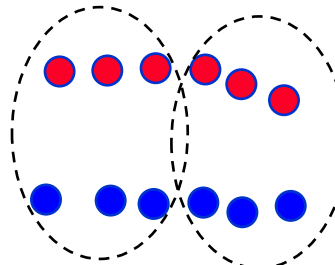
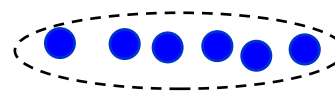
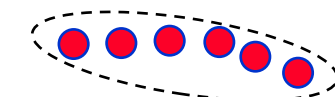
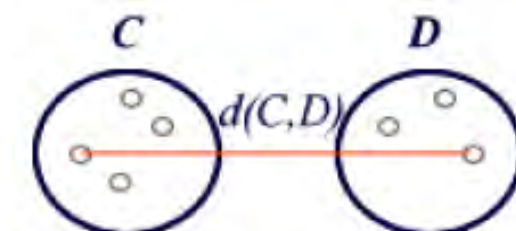
Criteri di Clustering

👉 **Single linkage (nearest neighbour):** la distanza fra due clusters è determinata dalla distanza dei due elementi più vicini.

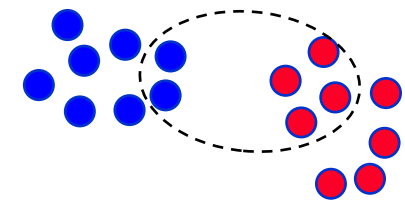
⇒ Questa metrica tende a produrre clusters a catena allungata

👉 **Complete linkage (furthest neighbours):** la distanza fra due clusters è determinata dalla massima distanza fra gli elementi più lontani.

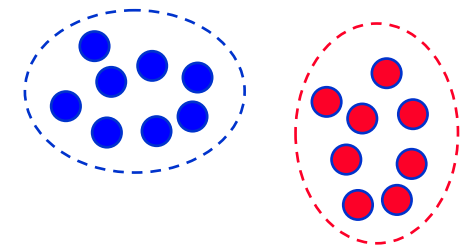
⇒ Questa metrica tende a formare clusters compatti, ma è inadatto per raggruppamenti allungati



single linkage



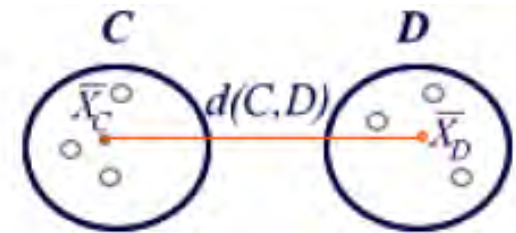
complete linkage



Criteri di Clustering

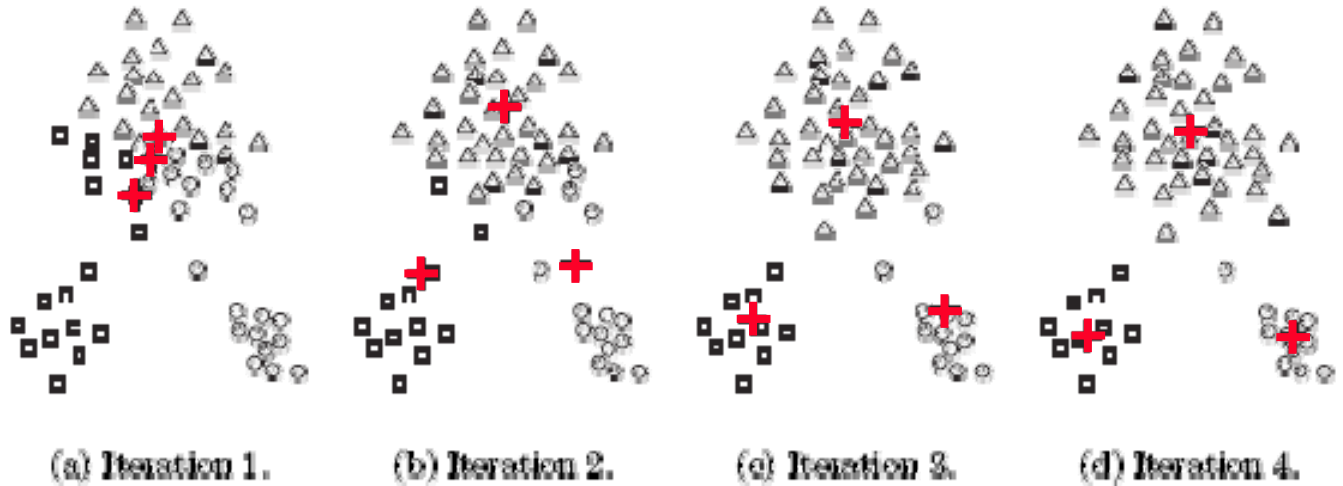
👉 **Metodo del centroide (centroid linkage):** la distanza fra due clusters è determinata dalla distanza fra i “centroidi” determinati come i valori medi delle dati nel cluster

$$d_{ik}^2 = \|x_k - v_i\|$$



👉 **Metodo K-means:** metodo iterativo

- ⇒ Si sceglie a-priori il numero di clusters in cui partizionare l'insieme
- ⇒ Si selezionano casualmente K punti e si considerano come centroidi iniziali
- ⇒ Si assegna ogni successivo punto al centroide più vicino
- ⇒ Si ricalcolano i centroidi
- ⇒ Il processo ha termine quando i centroidi sono stabili



Clustering deterministico (*hard clustering*)

☞ Raggruppamento di oggetti in funzione della similarità

⇒ Oggetti “simili” vanno nello stesso cluster

☞ La *Similarità* dei dati *i*-esimo al *k*-esimo cluster μ_{ik} è hard (SI/NO)

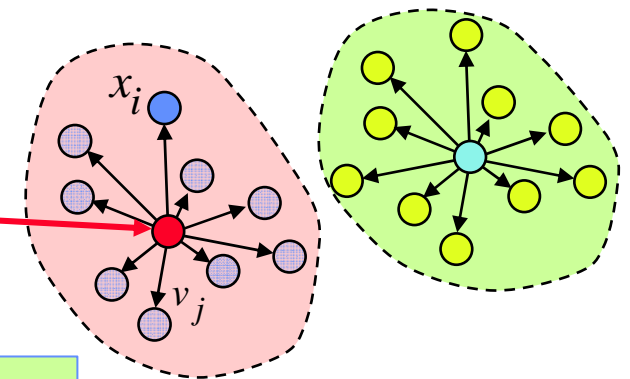
⇒ Funzione di appartenenza (membership) $\mu_{ik} = \begin{cases} 1 & SI \\ 0 & NO \end{cases}$

☞ Con dati numerici definiti in spazi normati la similarità può essere definita in termini di distanza secondo una norma $d_{ik}^2 = \|x_k - v_i\|$

☞ Si considerano le distanze da un punto definito centroide o prototipo del cluster (il più *rappresentativo*)

☞ La distribuzione dei dati nei clusters è fatta in modo da minimizzare un funzionale di partizione dipendente dalle distanze

$$J_{hc} = \min_{\mu_{ik}} \sum_{i=1}^c \sum_{k=1}^N d_{ik}^2$$



Limiti dell'hard clustering

☞ Dato che l'appartenenza è “hard” (SI/NO), non c'è alcuna modulazione del grado di appartenenza

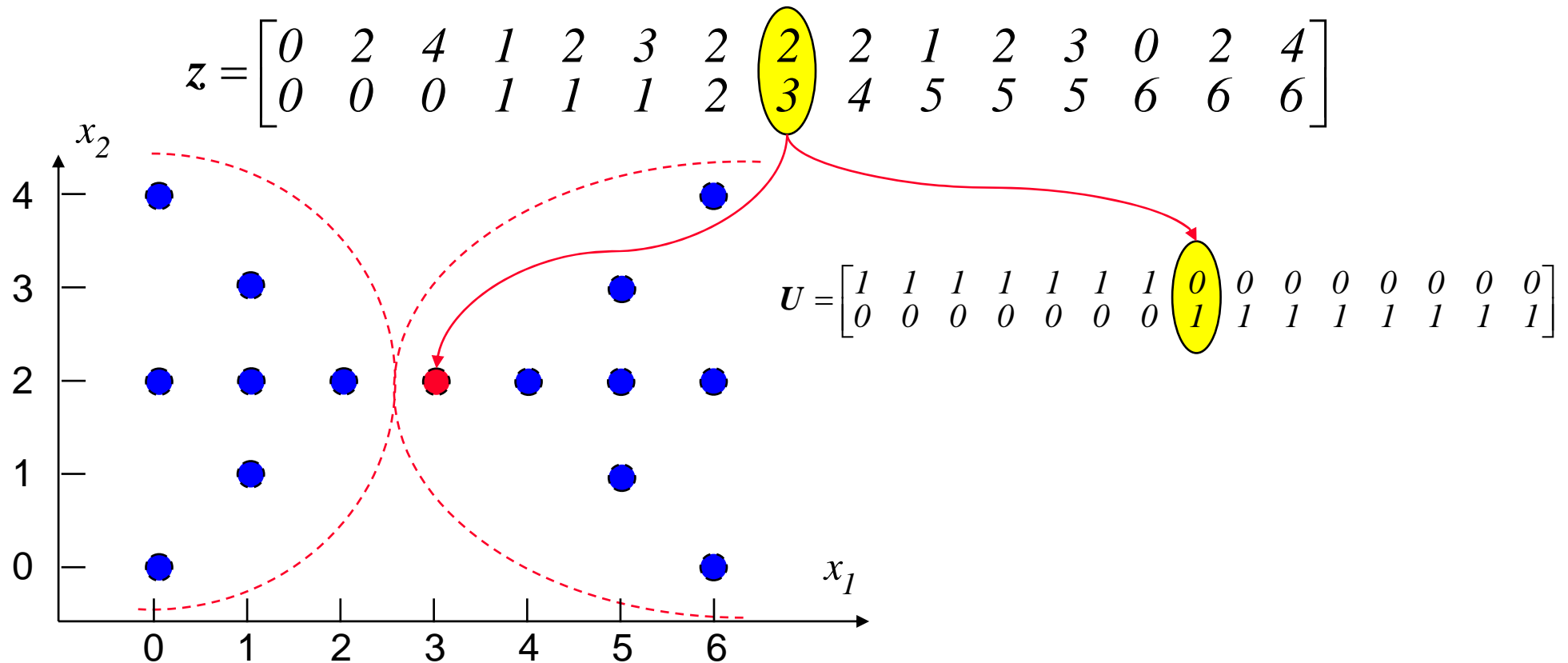
⇒ Tutti gli elementi sono considerati ugualmente appartenenti



☞ Gli hard clusters non riescono a considerare i “casi limite” (“borderline” cases)

Tipico fallimento dell'hard clustering

I cosiddetti *Butterfly data* sono un noto benchmark per i metodi di clustering. I dati possono venir raggruppati in due clusters ben definiti, eccetto il punto di mezzo la cui appartenenza è incerta. La partizione hard non riesce a risolvere questo dato, perché deve assegnarlo (per intero!) ad uno dei due clusters.



Fuzzy clustering

- ☞ Nel *fuzzy clustering* l'appartenenza può variare in modo continuo fra **0** (nessuna similarità) a **1** (completa similarità). Perciò:
1. Ogni dato può (parzialmente) appartenere a più clusters
 2. La somma delle appartenenze di ogni punto a tutti i clusters deve essere = 1
 3. La somma delle appartenenze di tutti i punti a ciascun cluster non può eccedere N
- ☞ Il funzionale di partizione ora include *l'appartenenza fuzzy* μ_{ik} elevata all'esponente fuzzy m

$$J_{fc}(c, m) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m d_{ik}^2$$

- ☞ Per risolvere il problema di partizione si deve introdurre un vincolo costituito dalla seconda condizione

$$\sum_{i=1}^c \mu_{ik} = 1 \quad k \in (1, N)$$

Fuzzy C-Means

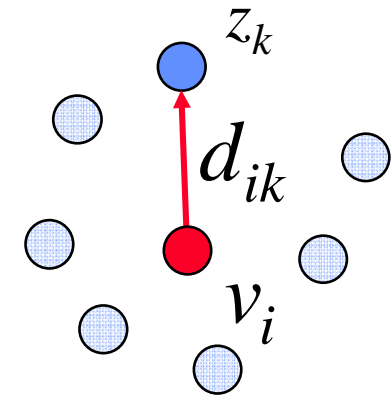
- 👉 La prima partizione fuzzy, chiamata **Fuzzy C-Means**, fu introdotta da Bezdek (1981)
- 👉 E' basata sulla minimizzazione vincolata del funzionale di partizione J_{fc} con il vincolo della totale appartenenza

$$J_{fc}(c, m) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m d_{ik}^2 \quad \text{subject to} \quad \sum_{i=1}^c \mu_{ik} = 1 \quad k \in (1, N)$$

- 👉 L'esponente $m \in [1, \infty)$ definisce la *fuzziness* della partizione
- 👉 Per la distanza si considera la norma euclidea

$$d_{ik}^2 = (\mathbf{z}_k - \mathbf{v}_i)^T (\mathbf{z}_k - \mathbf{v}_i)$$

- 👉 La distanza d_{ik} è calcolata fra ciascun punto \mathbf{z}_k e ciascun centro dei clusters \mathbf{v}_i e dipende dalla membership μ_{ik}



- 👉 ***L'algoritmo trova contemporaneamente le membership μ_{ik} e i centri dei clusters \mathbf{v}_i***

Soluzione del clustering FCM

- Il problema viene risolto iterativamente determinando in alternanza le memberships μ_{ik} e i centri dei cluster v_i , fino a che la partizione non converge
- Come si vede dallo schema le due quantità sono una funzione dell'altra

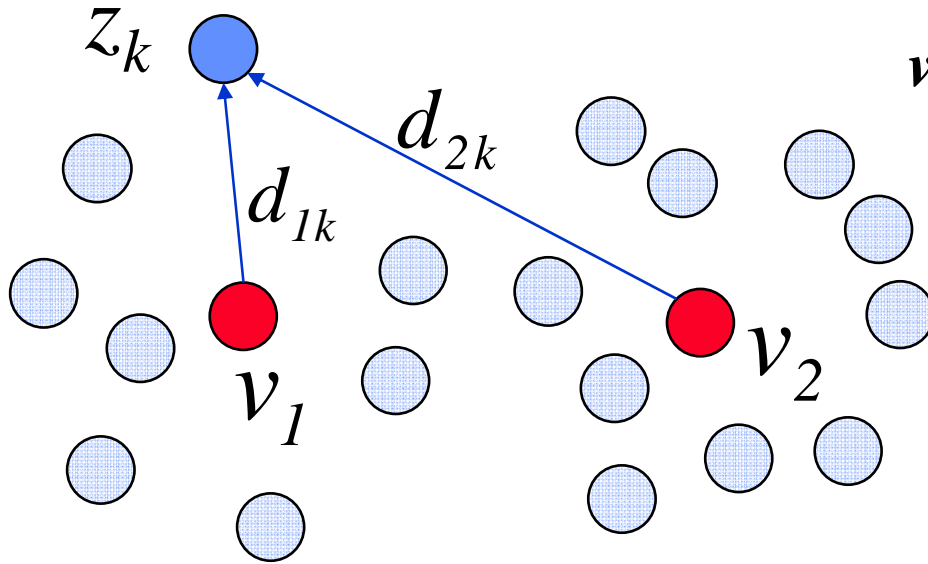
Le memberships determinano i centroidi

$$V : v_i = \frac{\sum_{k=1}^N (\mu_{ik})^m z_k}{\sum_{k=1}^N (\mu_{ik})^m} \quad i \in (1, c)$$

$$U : \mu_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}} \quad k \in (1, N)$$

I centroidi determinano le distanze

Esempio



$$\mathbf{v}_1 = \frac{\sum_{k=i}^N (\mu_{1k})^m \mathbf{z}_k}{\sum_{k=i}^N (\mu_{1k})^m}$$

$$\mathbf{v}_2 = \frac{\sum_{k=i}^N (\mu_{2k})^m \mathbf{z}_k}{\sum_{k=i}^N (\mu_{2k})^m}$$

$$d_{1k} = (\mathbf{z}_k - \mathbf{v}_1)^T (\mathbf{z}_k - \mathbf{v}_1)$$

$$d_{2k} = (\mathbf{z}_k - \mathbf{v}_2)^T (\mathbf{z}_k - \mathbf{v}_2)$$

$$\mu_{1k} = \frac{1}{\left(\frac{d_{1k}}{d_{1k}}\right)^{\frac{2}{m-1}} + \left(\frac{d_{1k}}{d_{2k}}\right)^{\frac{2}{m-1}}}$$

A red arrow points from the number 1 in the numerator to the first term in the denominator.

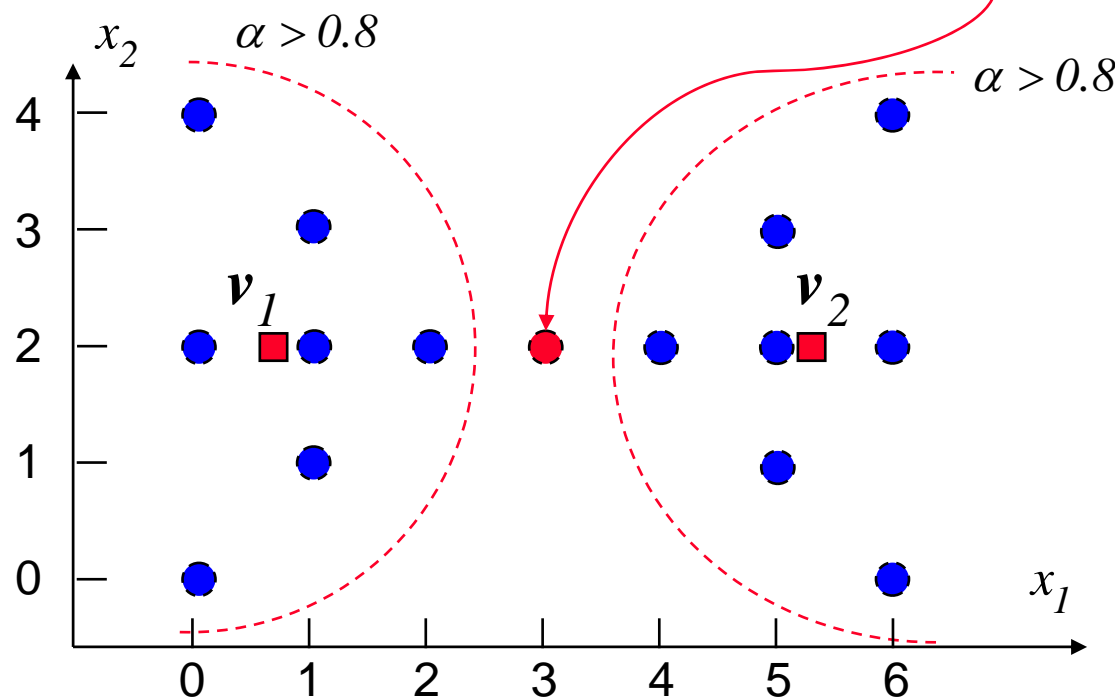
$$\mu_{2k} = \frac{1}{\left(\frac{d_{2k}}{d_{1k}}\right)^{\frac{2}{m-1}} + \left(\frac{d_{2k}}{d_{2k}}\right)^{\frac{2}{m-1}}}$$

A red arrow points from the number 1 in the numerator to the second term in the denominator.

Di nuovo i *Buttefly data*

I *Buttefly data* sono trattati molto bene dal FCM , perché al punto medio (3,2) viene assegnata una **membership 0.5**, cioè uguale appartenenza ad ambedue i cluster. Con una fuzziness $m = 2$ la matrice di partizione è

$$z = \begin{bmatrix} 0.86 & 0.97 & 0.86 & 0.94 & 0.99 & 0.94 & 0.88 & 0.5 & 0.12 & 0.06 & 0.01 & 0.06 & 0.14 & 0.03 & 0.14 \\ 0.14 & 0.03 & 0.14 & 0.06 & 0.01 & 0.06 & 0.12 & 0.5 & 0.88 & 0.94 & 0.99 & 0.94 & 0.86 & 0.97 & 0.86 \end{bmatrix}$$

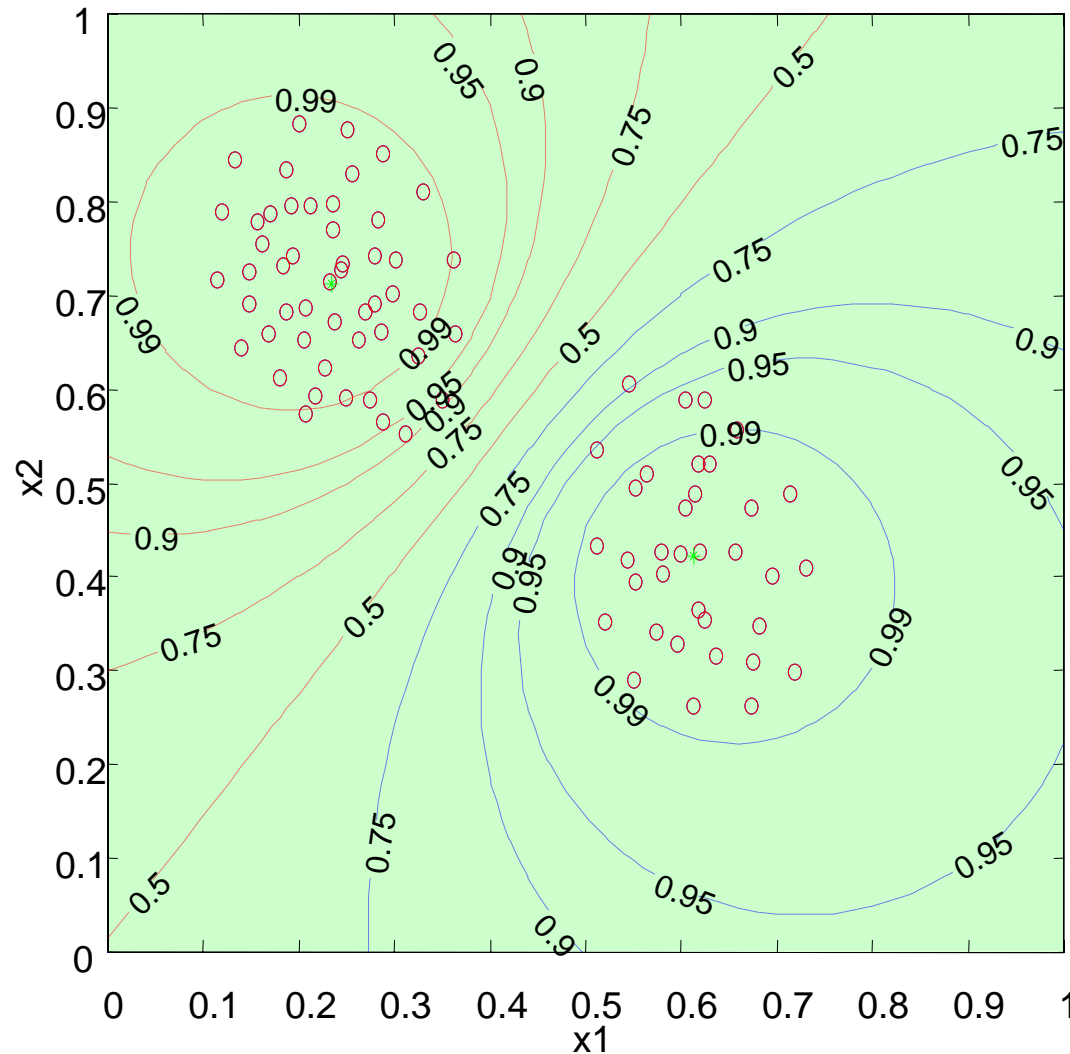


■ = centroidi

$$V = \begin{bmatrix} 0.85 & 5.14 \\ 2.00 & 2.00 \end{bmatrix}$$

\uparrow \uparrow
 v_1 v_2

Esempio di clustering FCM

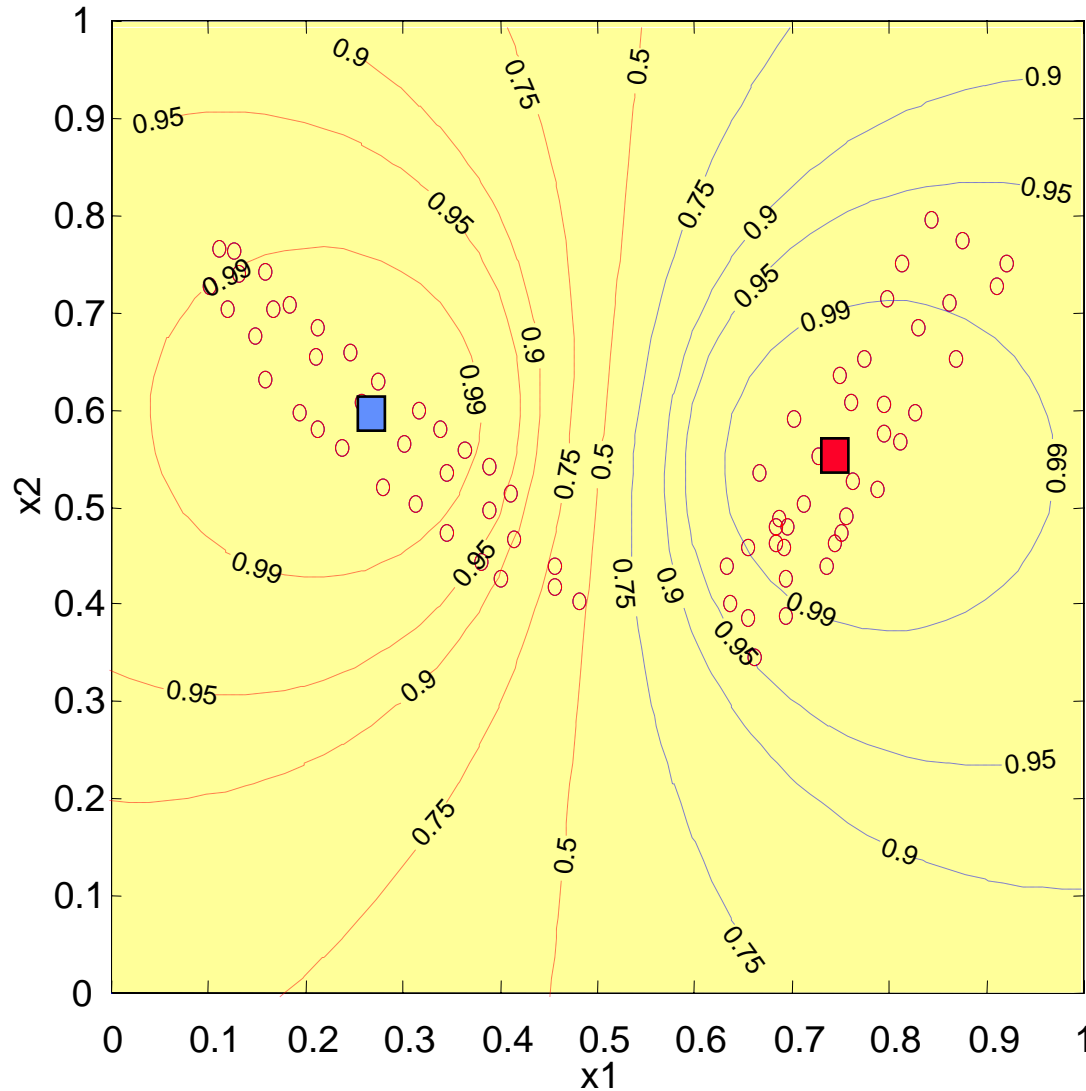


Essendo basato sulla norma euclidea, FCM determina clusters *sferici*.

Se i dati sono disposti in forme circa sferiche, FCM produce del clusters accettabili, altrimenti la partizione può essere irrealistica.

Questo è il maggior limite di FCM

Quando FCM non va bene



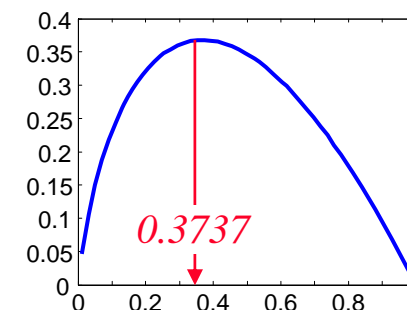
Se i dati sono disposti in insiemi “allungato”, FCM può produrre dei clusters irrealistici, che possono non riflettere il reale raggruppamento dei dati o raggruppare i dati in modo incoerente

■ ■ *centroidi*

Entropia di partizione

- Il principale indice della bontà della partizione è l'**Entropia Fuzzy**
- Essa è definita secondo i principi della Teoria dell'Informazione in funzione dell'incertezza residua risultante dalla partizione

$$H_n = - \frac{1}{1 - \frac{c}{N}} \sum_{i=1}^c \sum_{k=1}^N \mu_{i,k} \cdot \log(\mu_{i,k}) / N$$



⇒ **Minore è il valore di H , tanto maggiore è l'efficacia della partizione**

- H_n può variare da un minimo di 0 (partizione “hard”) in cui non c'è incertezza ad un massimo di $\log(c)$, corrispondente a membership tutte uguali a $1/c$ (massima incertezza)

$$\mu = (0/1)$$

$$\mu = (1/c)$$

0

range di H_n

$\log(c)$

massima certezza

massima incertezza

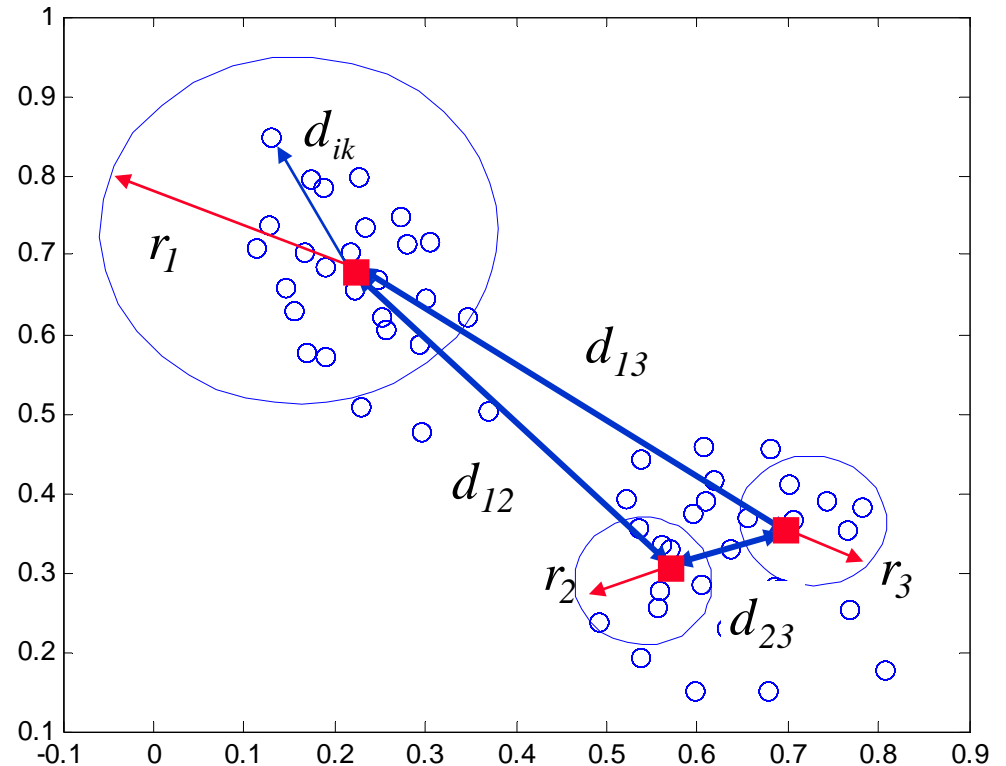
Coefficiente di separazione

- Misura la “compattezza” dei clusters paragonando il loro “raggio” con la distanza fra i centroidi
- Si definisce “raggio” del cluster i -esimo la massima distanza pesata intra-cluster

$$r_i = \max_{1 \leq k \leq N} (\mu_{ik} d_{ik})$$

- Il **Coefficiente di Separazione** è massimo rapporto fra queste e le distanza inter-cluster d_{ij}

$$g = \max_{1 \leq i, j \leq c} \left(\frac{r_i + r_j}{d_{ij}} \right)$$

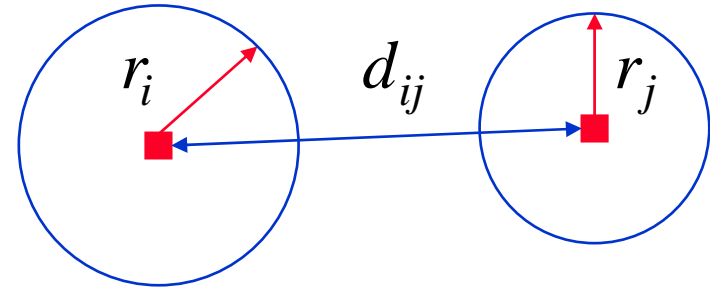


Possibili situazioni di separazione

👉 **Clusters ben separati:**

la minima distanza
è **superiore** alla
massima somma
dei raggi

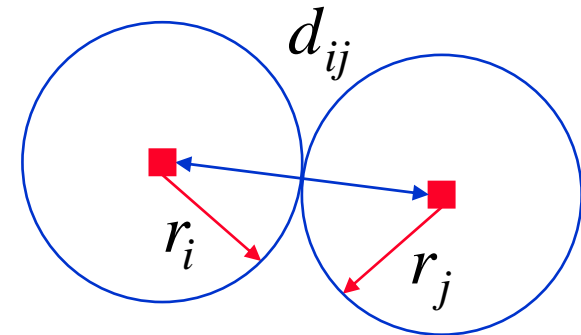
$$g = \max \left(\frac{r_i + r_j}{d_{ij}} \right) < 1$$



👉 **Clusters tangenti:**

la minima distanza
è **uguale** alla
massima somma
dei raggi

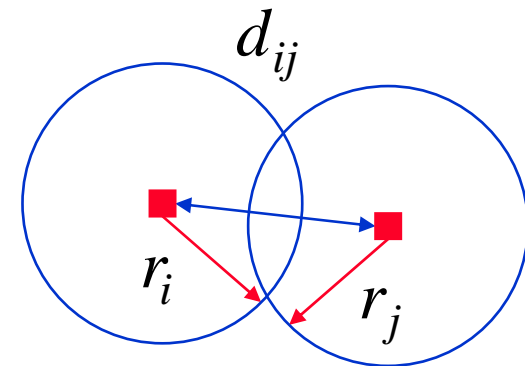
$$g = \max \left(\frac{r_i + r_j}{d_{ij}} \right) = 1$$



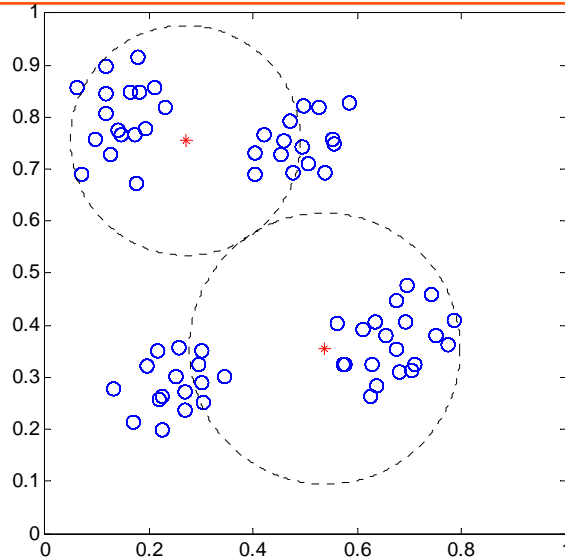
👉 **Clusters intersecanti:**

la minima distanza
è **inferiore** alla
massima somma
dei raggi

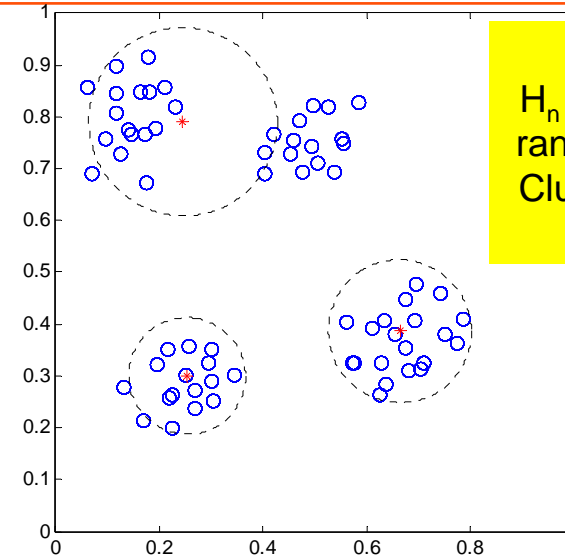
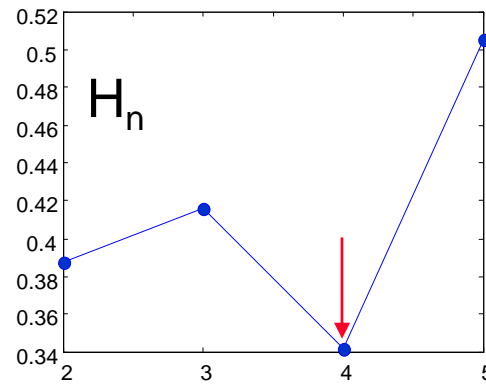
$$g = \max \left(\frac{r_i + r_j}{d_{ij}} \right) > 1$$



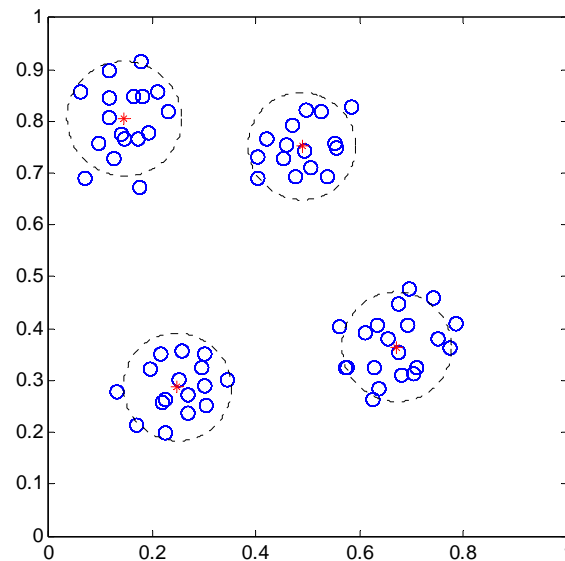
Variazione di H_n con il numero di clusters



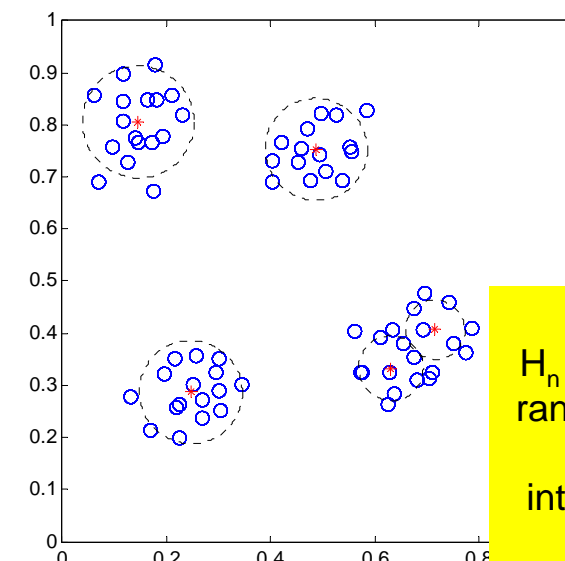
$c = 2$
 $H_n = 0.38757$ nel
 range $(0, 0.69315)$
 Clusters tangenti
 $g = 1.0$



$c = 3$
 $H_n = 0.41592$ nel
 range $(0, 1.0986)$
 Clusters separati
 $g = 0.6$

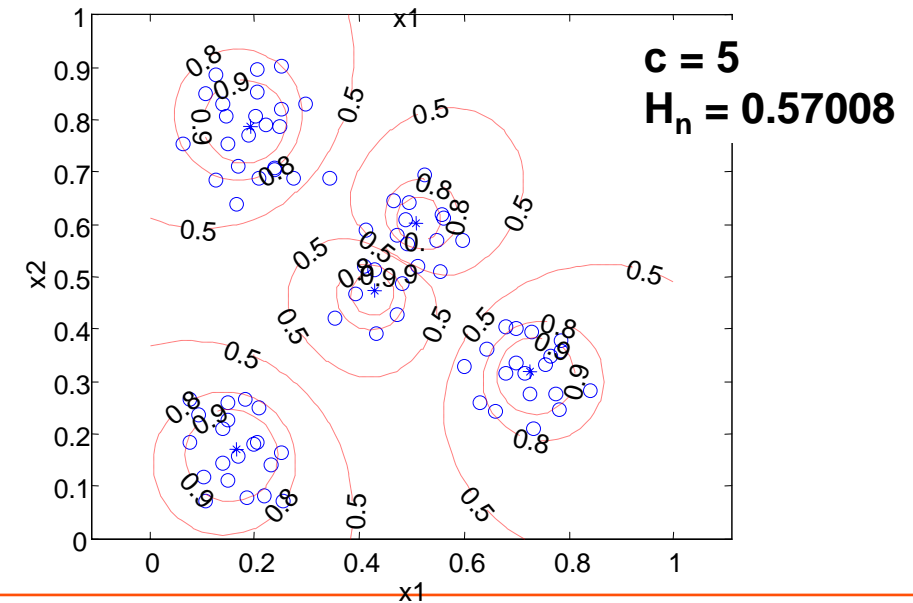
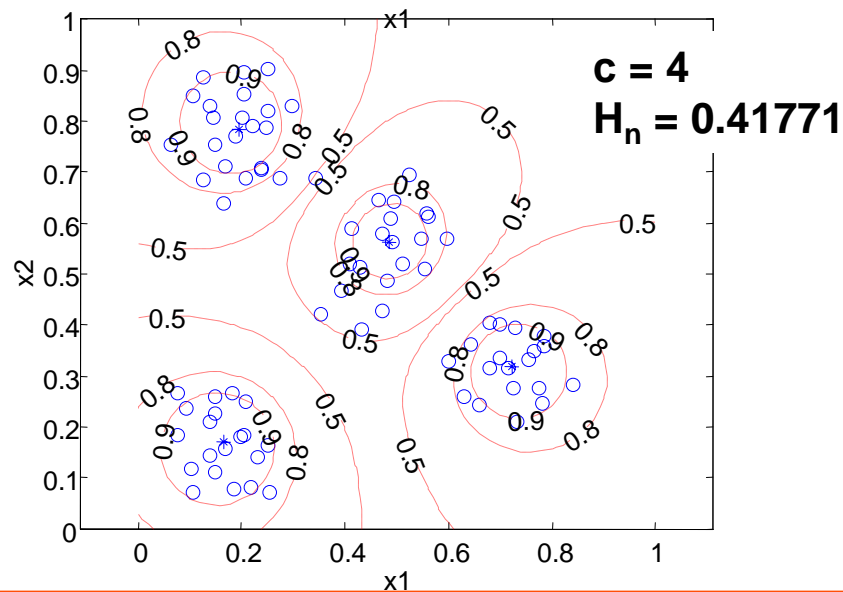
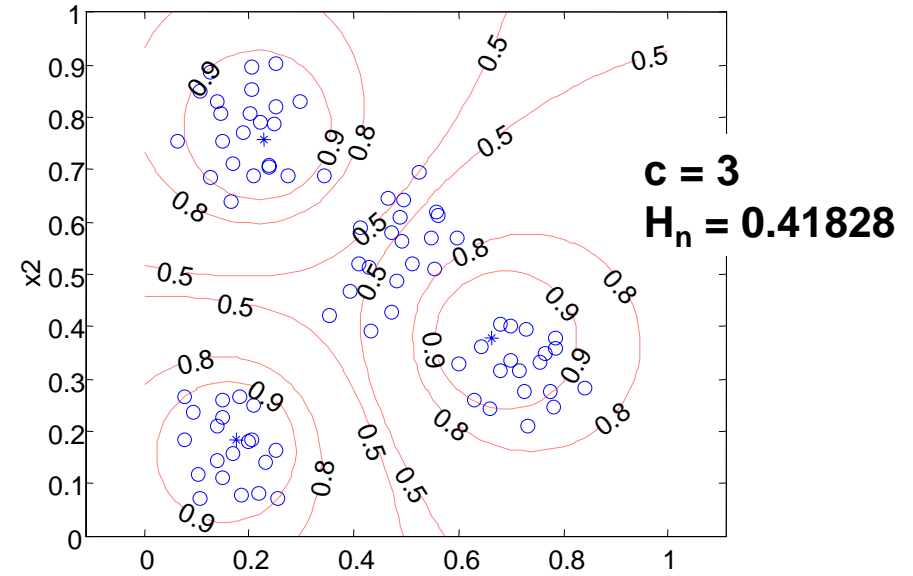
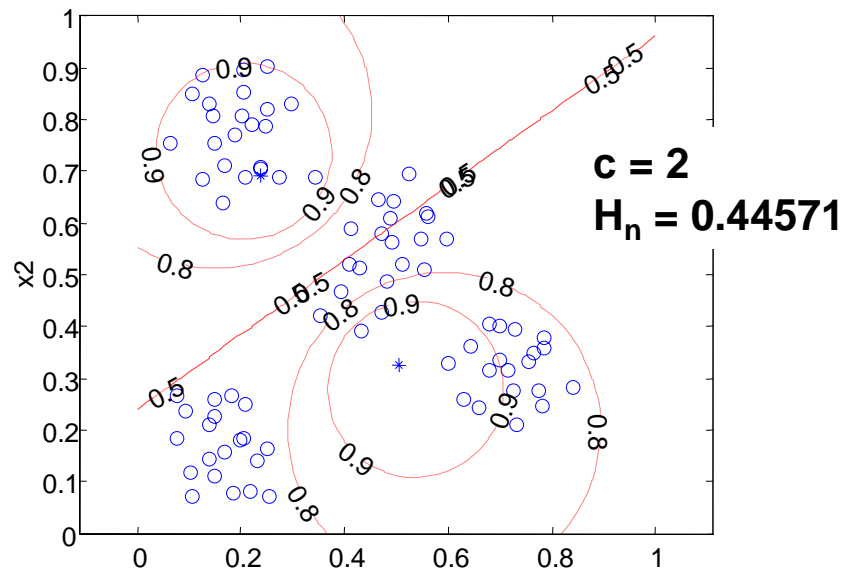


$c = 4$
 $H_n = 0.34158$ nel
 range $(0, 1.3863)$
 Clusters separati
 $g = 0.62$



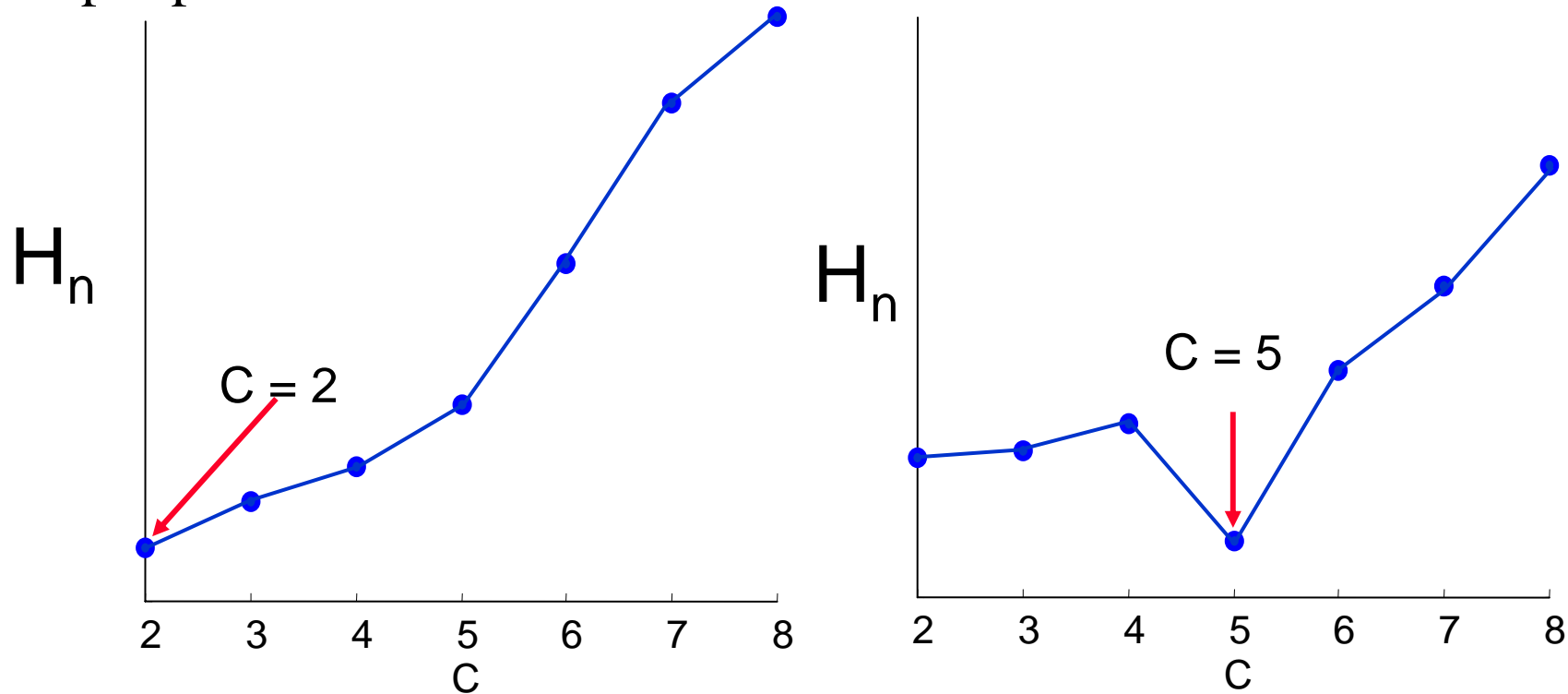
$c = 5$
 $H_n = 0.50551$ nel
 range $(0, 1.6094)$
 Clusters
 intersecanti $g =$
 1.054

Linee di livello dei clusters e H_n al variare di c



Come usare H_n per determinare il miglior C

- ☞ Per $c = 2$ si ha spesso un minimo. Si deve considerare questo minimo solo se H_n è crescente monotonicamente
- ☞ Se esistono altri minimi per $c > 2$ allora si considera quello per il c più piccolo



Altri criteri di scelta del numero di clusters

👉 **Fuzzy hypervolume (FHV):** se Σ_i sono le matrici di covarianza dei clusters, il primo minimo di V_h indica la migliore partizione

$$V_h = \sum_{i=1}^c [\det(\Sigma_i)]^{\frac{1}{2}}$$

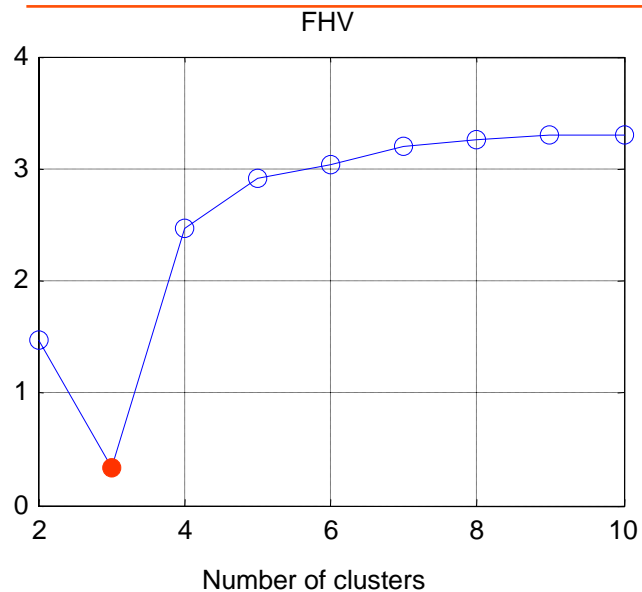
👉 **Average Partition Density (APD):** se S_i è la somma delle membership dei dati contenuti nell'ellissoide di raggio pari alla deviazione standard del cluster. Il primo massimo indica la migliore partizione

$$D_A = \frac{1}{c} \sum_{i=1}^c \frac{S_i}{[\det(\Sigma_i)]^{\frac{1}{2}}}$$

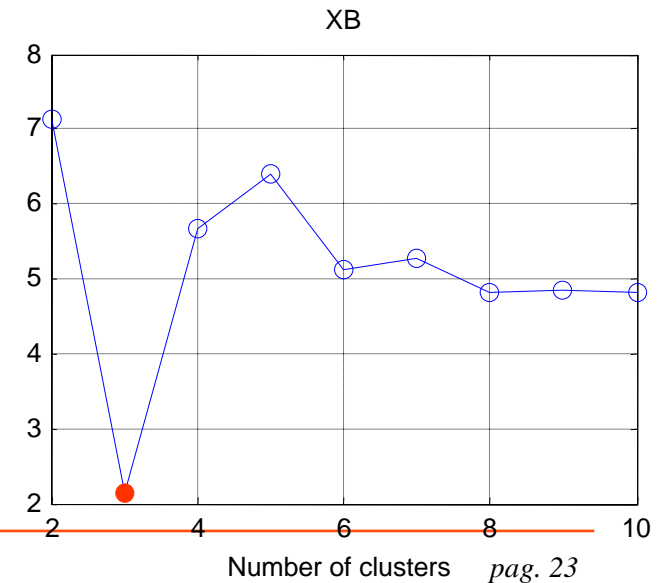
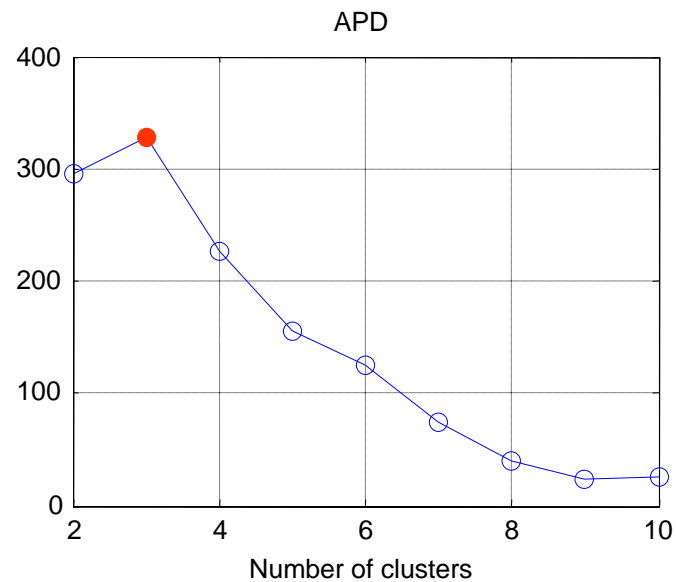
👉 **Xie and Beni criterion (XB):** basato sul rapporto fra distanze euclidee pesate con le memberships. Il primo minimo indica la migliore partizione

$$XB(c) = \frac{\sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^m \|\mathbf{x}_j - \mathbf{v}_i\|^2}{N \min_{ij} \|\mathbf{x}_j - \mathbf{v}_i\|^2}$$

Esempio di misure di partizione

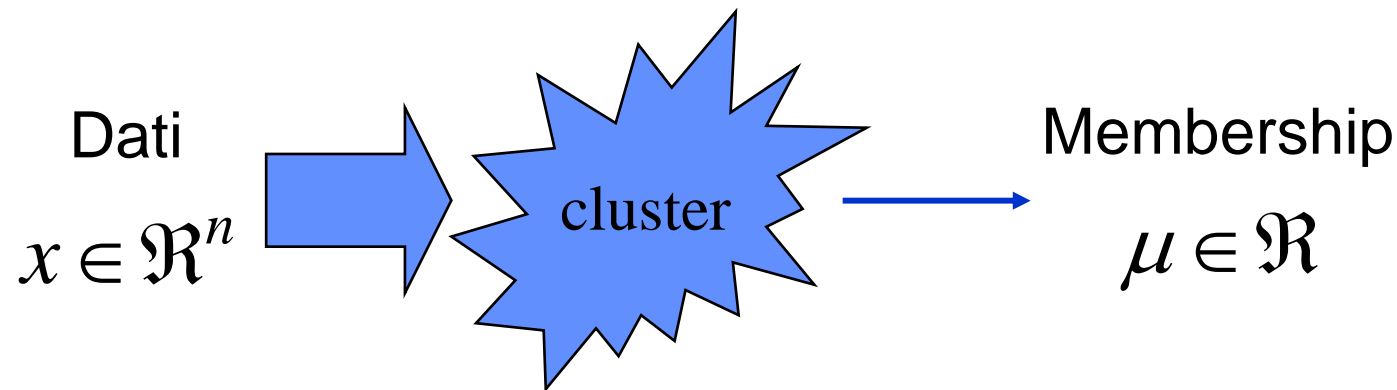


Generalmente tutti i criteri sono concordi nell'indicare un identico numero ottimale di partizioni



Utilizzo del Fuzzy Clustering (FC)

- ☞ Il FC può essere usato per rivelare relazioni di somiglianza in insiemi complessi di dati
 - ⇒ L'aspetto *fuzzy* aiuta a classificare i dati in *più o meno* appartenenti ad un certo gruppo
- ☞ Nel caso di dati multidimensionali, l'analisi FC consente una forte riduzione della complessità, perché l'uscita è sempre e comunque una membership scalare

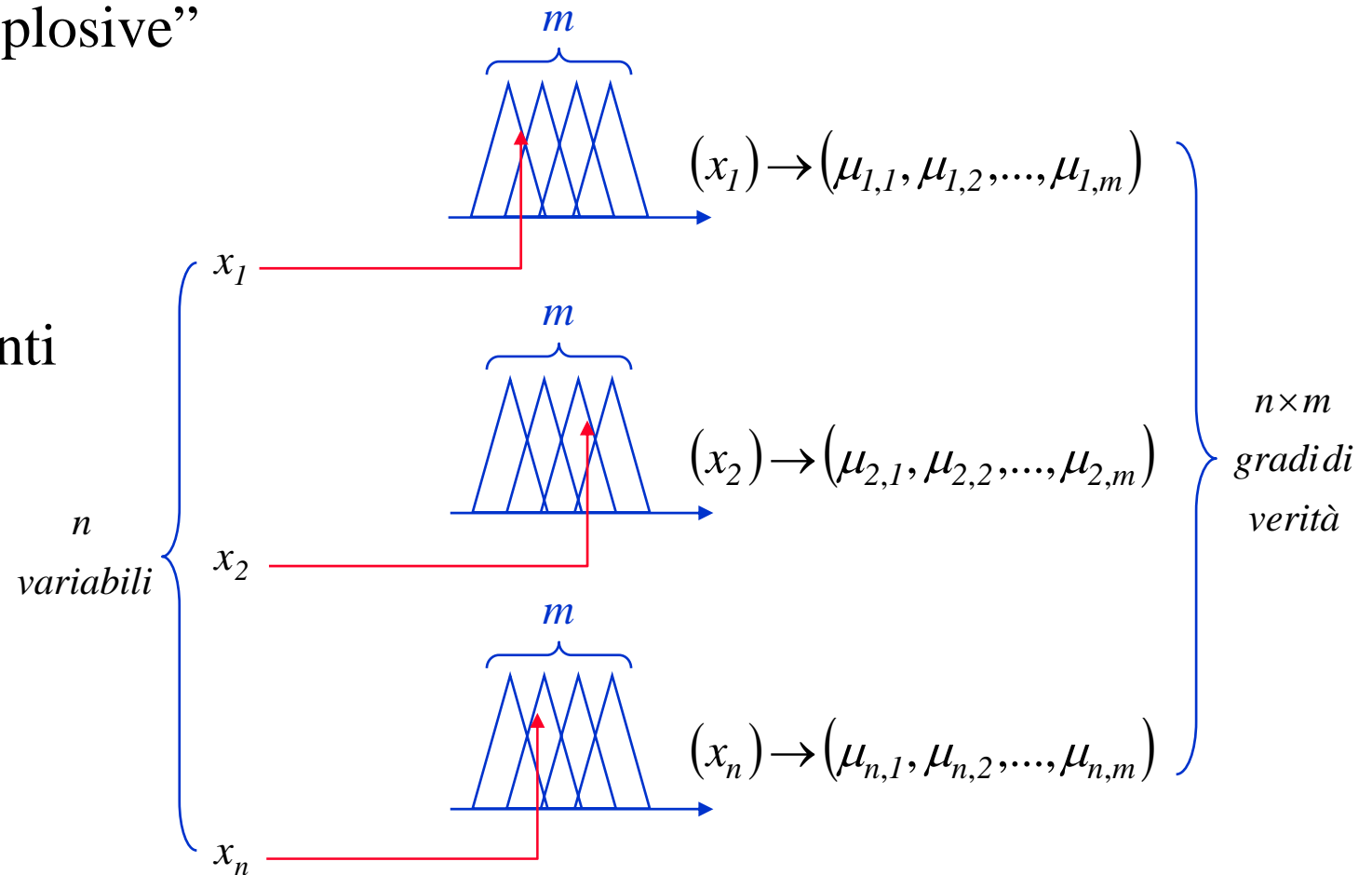


- ☞ Il FC può essere usato come metodo *diagnostico* per scoprire quando certe concomitanze fra dati implicano una data situazione significativa per il sistema
- ☞ Il FC può essere utilizzato per costruire automaticamente rappresentatori fuzzy da usare come antecedenti di modelli o regolatori fuzzy

La maledizione della dimensionalità

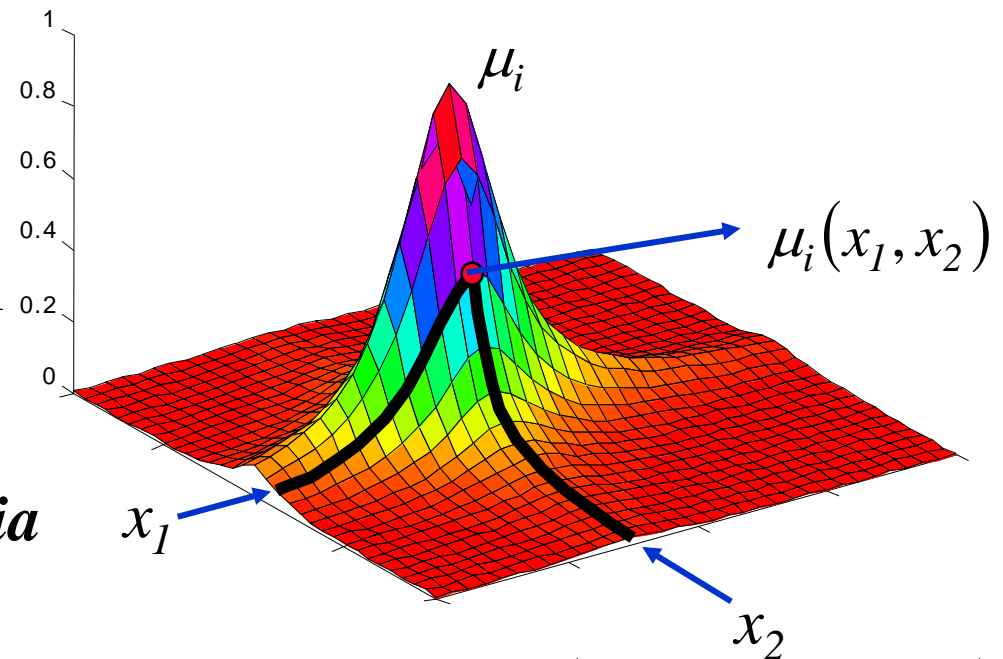
☞ Nei metodi numerici si fa spesso riferimento alla “*curse of dimensionality*” (Bellman) quando il problema tende a diventare di dimensioni “esplosive”

☞ Ad esempio nel caso fuzzy, se si hanno molti antecedenti e molte membership da n variabili si passa a $n \times m$ membership



Vantaggi della fuzzificazione mediante cluster

- ☞ Se i dati sono raggruppati in clusters, le membership sono ricavabili dai centroidi $\{v_1, v_2, \dots, v_c\}$, perciò $\mathbf{x} \in \mathcal{R}^n \rightarrow \mu \in \mathcal{R}^c$
- ☞ Nel caso multivariabile, il grado di verità risultante μ è dato dall'appartenenza **congiunta** dei dati ad un cluster, indipendente dal numero di variabili in ingresso
- ☞ Perciò il numero di regole non dipende più dal numero di variabili antecedenti ma solo dal numero di clusters \rightarrow **Enorme parsimonia di regole di inferenza**



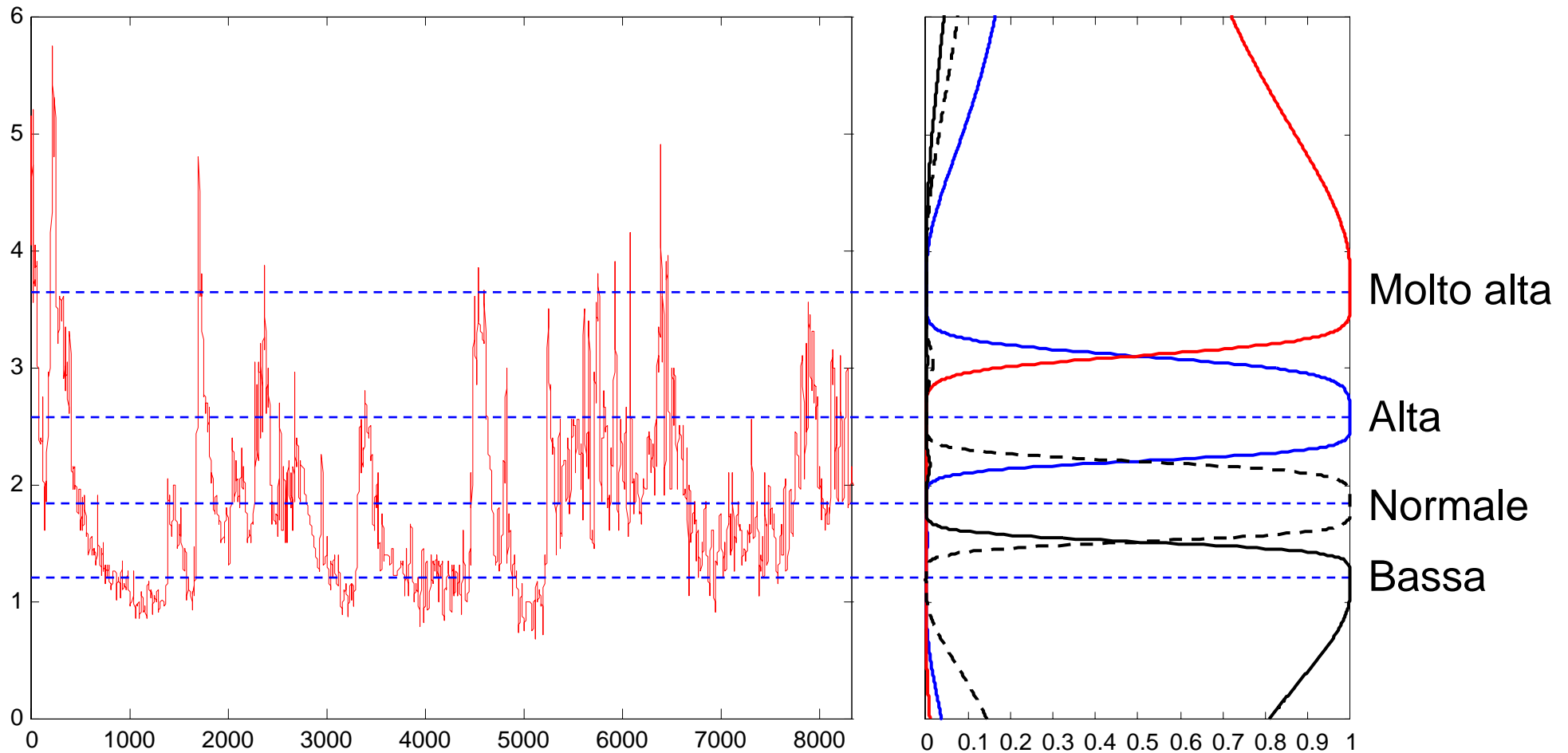
$$R_i : \text{if } \{x_1, x_2, \dots, x_n\} \subset C_i \text{ then } y = f_i(x_1, x_2, \dots, x_n)$$

Classificazione di una quantità scalare

- ☞ Si esegue una clusterizzazione sull'unica dimensione dei dati
- ☞ Eventualmente si effettuano dei test per cercare il numero ottimale di clusters c
- ☞ Ottenuti i centroidi $\{v_1, v_2, \dots, v_c\}$, eventualmente ereditati da una precedente clusterizzazione, si esegue una classificazione di dati utilizzando in sequenza l'intero supporto $x \in (x_{min}, x_{max})$
- ☞ In tal modo si ha la membership di un qualunque valore del supporto rispetto all' i -esimo
centroide $v_i \in \{v_1, v_2, \dots, v_c\}$
- ☞ Questa formula può essere usata per classificare ulteriori dati e per definire un sistema inferenziale

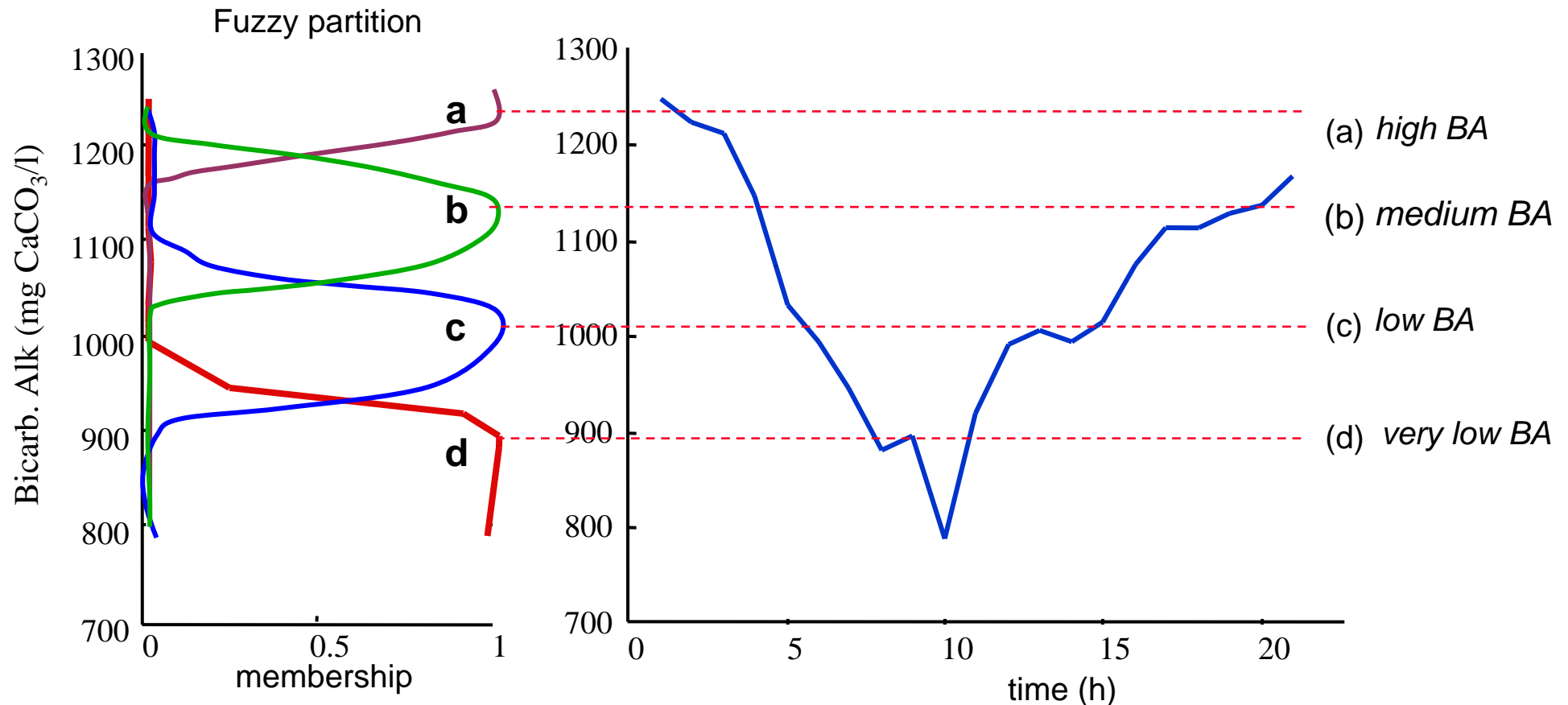
$$\mu_i = \frac{1}{\sum_{j=1}^c \left(\frac{d_i}{d_j} \right)^{\frac{2}{m-1}}}$$

Classificazione di dati di torbidità



Nota: a causa del vincolo $\sum \mu_i = 1$ le μ agli estremi tendono ad essere tutte uguali e pari a $1/c$.
Perciò questa classificazione è valida prevalentemente nell'intervallo centrale del supporto

Generazione di membership di funzione scalare



In questo caso si fissa c ad un valore opportuno e si clusterizza l'andamento della variabile da fuzzificare.

Il FCM fornisce automaticamente le membership a sinistra. Queste serviranno per classificare successivi valori, ad es. per un modello fuzzy o per il controllo.

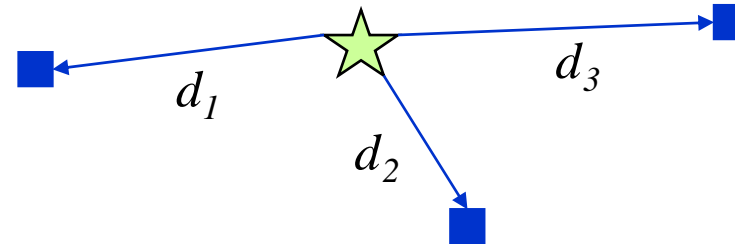
Uso diagnostico del Fuzzy Clustering

- ☞ FC permette di effettuare un'analisi di raggruppamento dei dati in clusters con un *significato comportamentale*
 - ⇒ Es. funzionamento normale, malfunzionamento di tipo A, B, etc.
- ☞ La gradazione fuzzy è importante per decidere quanto il dato è significativo rispetto al dato comportamento
 - ⇒ In questo contesto i centroidi rappresentano l'elemento più significativo per ciascun cluster
- ☞ La diagnosi (che fare?) può essere fatta attraverso un sistema inferenziale che si basa sulle membership dei dati analizzati
- ☞ Come effettuare la ripartizione quando la separazione fra dati non è evidente?
 - ⇒ Va definito un criterio di diminuzione della confusione attraverso la partizione
 - ⇒ Il numero di clusters che minimizza la confusione (incertezza informativa) è il migliore

Classificazione di nuovi dati

- Una precedente clusterizzazione di un nucleo di dati, che possiamo considerare di training, può servire per creare la *base di conoscenza* necessaria a classificare successivi dati
 - ⇒ Risponde alla domanda: a quale cluster appartiene ciascun nuovo dato?
- La classificazione si basa sulla funzione di appartenenza già trovata

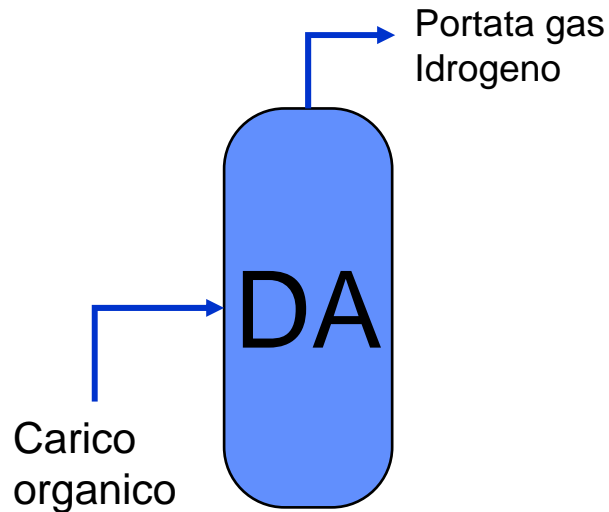
$$\mu_i = \frac{1}{\sum_{j=1}^c \left(\frac{d_i}{d_j} \right)^{\frac{2}{m-1}}}$$



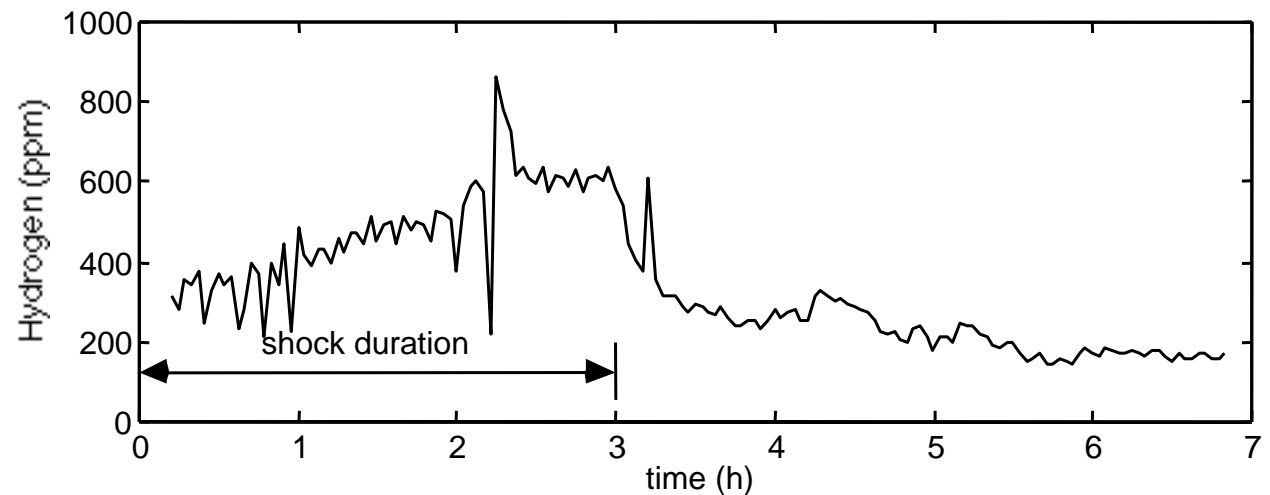
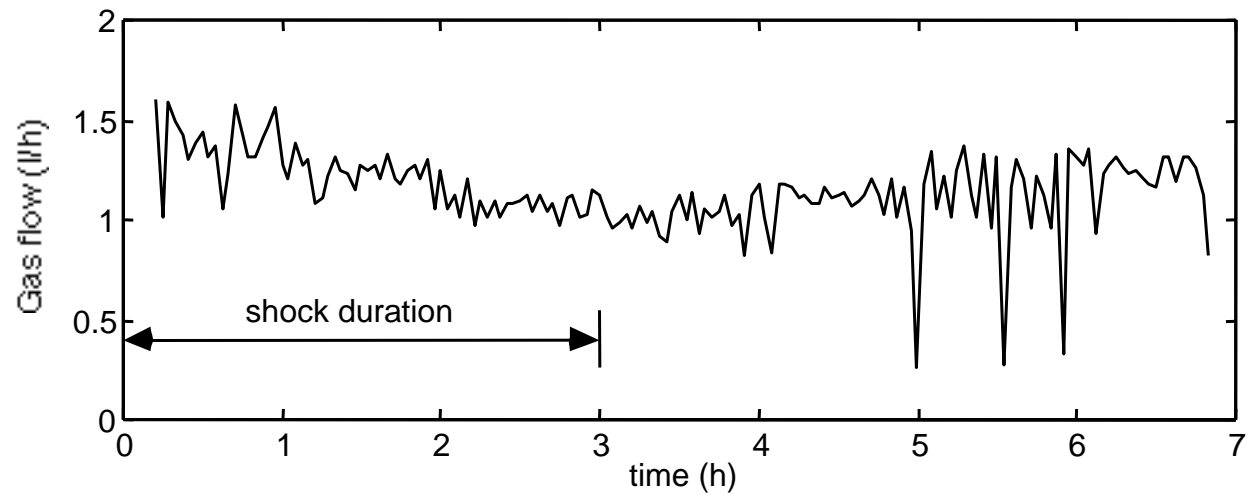
ma il suo uso è più semplice, perché il *centroide è già noto*, perciò le distanze (d_1, d_2, \dots, d_c) sono direttamente calcolabili.

- La precedente, usata *una sola volta per ciascun nuovo dato* (non iterativamente) dà la sua appartenenza alla precedente clusterizzazione
- Tale classificazione può avere un uso *diagnostico*, perché qualifica il nuovo dato rispetto alla base di conoscenza pre-esistente.

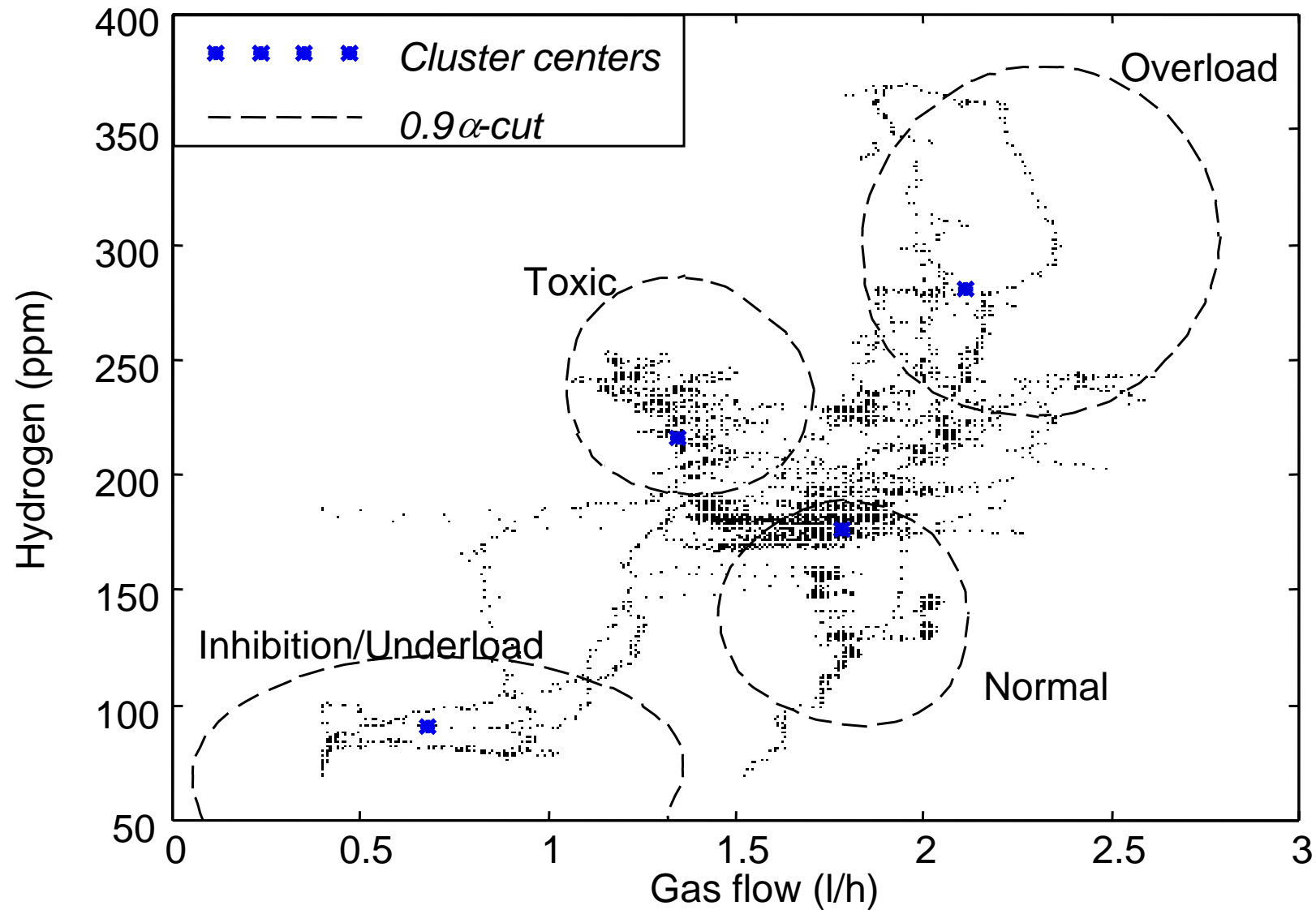
Diagnosi di sovraccarico di un digestore anaerobico



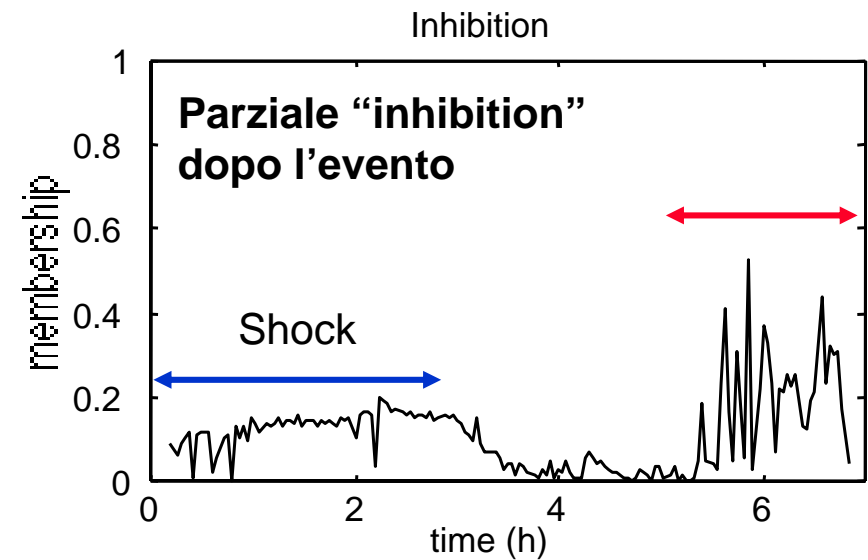
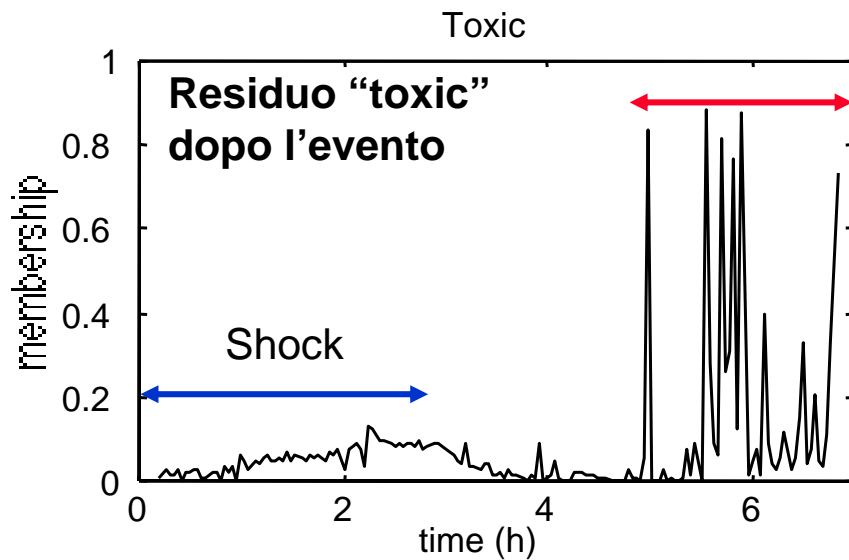
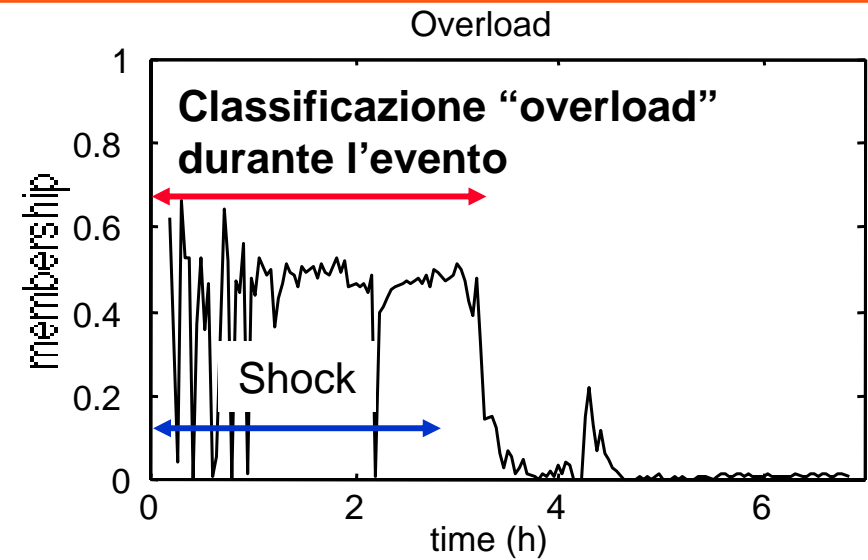
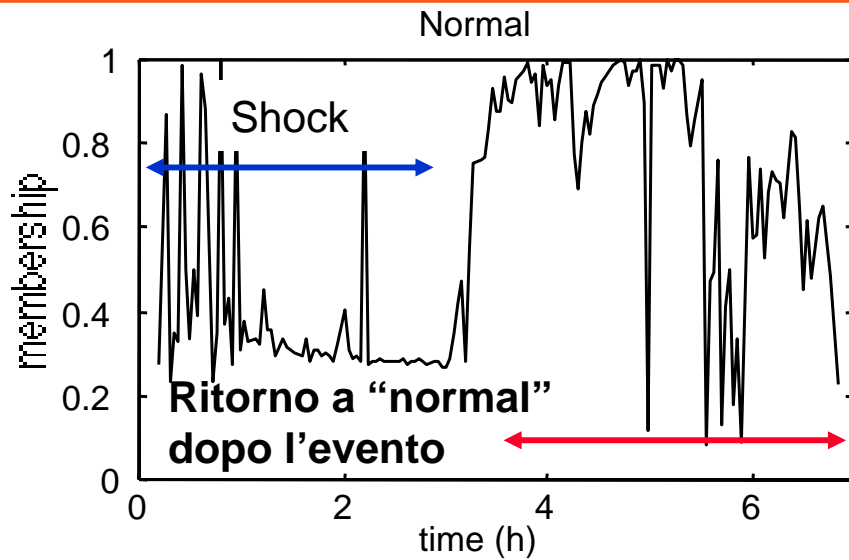
La risposta di un DA ad un eccessivo carico organico (*Overload*) è una inibizione della fase metanogena con diminuzione della portata ed aumento del contenuto di idrogeno



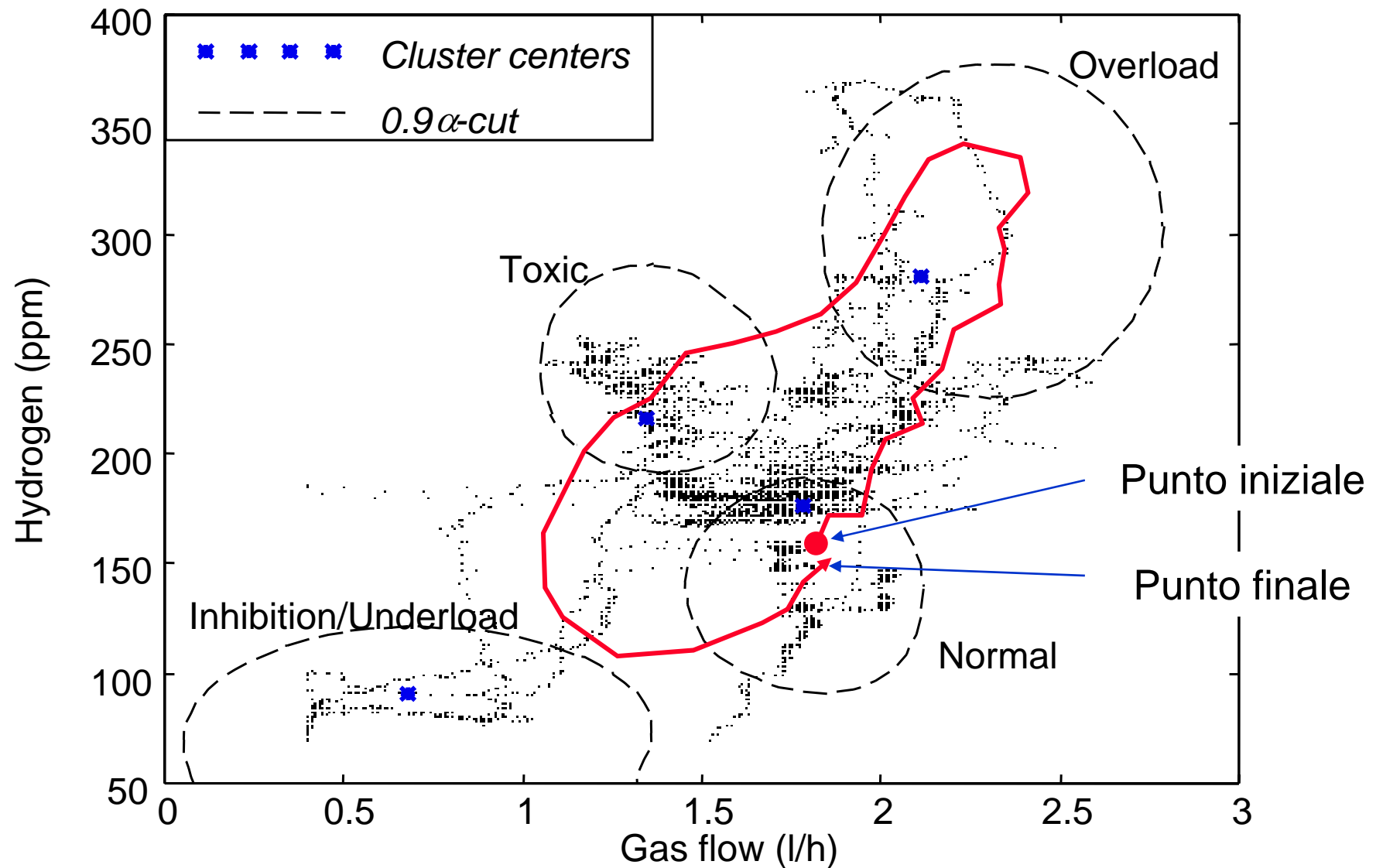
Diagnostica fuzzy di un digestore anaerobico



Classificazione dello shock "Overload"



Percorso dello shock attraverso i clusters



Norme del prodotto interno

☞ In spazi metrici la forma dei clusters è determinata dalla scelta della matrice che induce la norma

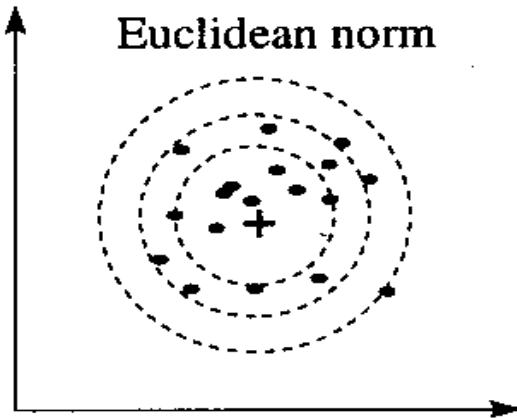
$$d_{ik}^2 = (\mathbf{z}_k - \mathbf{v}_i)^T \mathbf{A} (\mathbf{z}_k - \mathbf{v}_i) \quad \mathbf{A} = \mathbf{R}^{-1}$$

$$\mathbf{A} = \mathbf{I}$$

Matrice identità



Euclidean norm

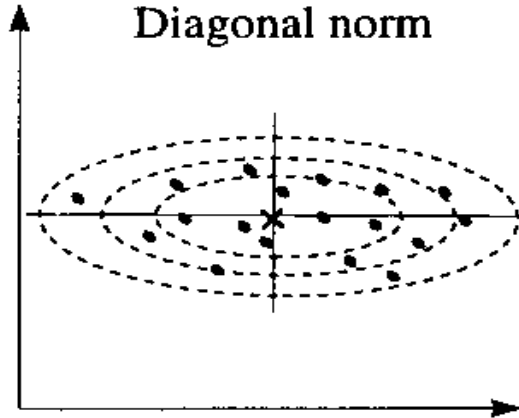


$$\mathbf{A} = \text{diag} \left[1/\sigma_1^2 \quad 1/\sigma_2^2 \quad \dots \quad 1/\sigma_p^2 \right]$$

Matrice diagonale con
reciproco delle varianze dei dati



Diagonal norm

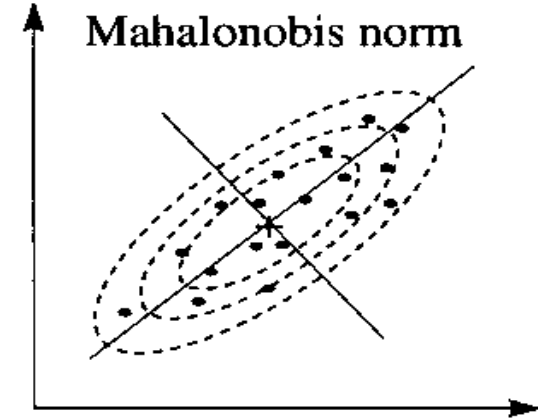


$$\mathbf{R} = \frac{1}{N} \sum_{k=1}^N (\mathbf{z}_k - \bar{\mathbf{z}})(\mathbf{z}_k - \bar{\mathbf{z}})^T$$

Matrice inversa
di covarianza



Mahalanobis norm



Matrici che inducono la norma

👉 **Norma diagonale:** se la matrice \mathbf{A} è l'inversa della varianza dei dati lungo ciascuna coordinata

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \dots \\ z_p \end{bmatrix} \Rightarrow \begin{array}{l} \sigma_1^2 = \text{var}(z_1) \\ \sigma_2^2 = \text{var}(z_2) \\ \dots \\ \sigma_p^2 = \text{var}(z_p) \end{array} \quad \Rightarrow \quad \mathbf{A} = \begin{bmatrix} (1/\sigma_1)^2 & 0 & \dots & 0 \\ 0 & (1/\sigma_2)^2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & (1/\sigma_p)^2 \end{bmatrix}$$

👉 **Norma di Mahalanobis:** se la matrice \mathbf{A} è l'inversa della matrice di covarianza dei dati \mathbf{R}

$$\mathbf{R} = \frac{1}{N} \sum_{k=1}^N (\mathbf{z}_k - \bar{\mathbf{z}})(\mathbf{z}_k - \bar{\mathbf{z}})^T \quad \Rightarrow \quad \mathbf{A} = \mathbf{R}^{-1} \quad \bar{\mathbf{z}} = \text{mean}(\mathbf{z})$$

Clustering con matrice di covarianza fuzzy

☞ Algoritmo di Gustafson e Kessel (estensione di FCM con norma adattiva). E' in grado di

- ⇒ Rivelare clusters di forme diverse (non sferici)
- ⇒ Ciascun cluster ha la propria matrice che induce la norma A_i
- ⇒ Le matrici A_i sono usate come variabili di ottimizzazione per trovare la migliore partizione

☞ La funzione obiettivo diviene

$$J_{gk}(c, m, A_i) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m d_{ikA_i}^2$$

con

$$d_{ikA_i}^2 = (\mathbf{z}_k - \mathbf{v}_i)^T A_i (\mathbf{z}_k - \mathbf{v}_i)$$

☞ Le matrici A_i devono essere vincolate, altrimenti il problema di ottimizzazione è mal posto

- ⇒ Imporre un determinante costante ad ogni $A_i \Rightarrow$ cluster a volume costante

$$A_i : \det(A_i) = \rho_i > 0 \quad i = 1, \dots, c$$

Algoritmo di Gustafson-Kessel

- ☞ Introducendo F_i come la matrice di covarianza fuzzy per l' i -esimo cluster

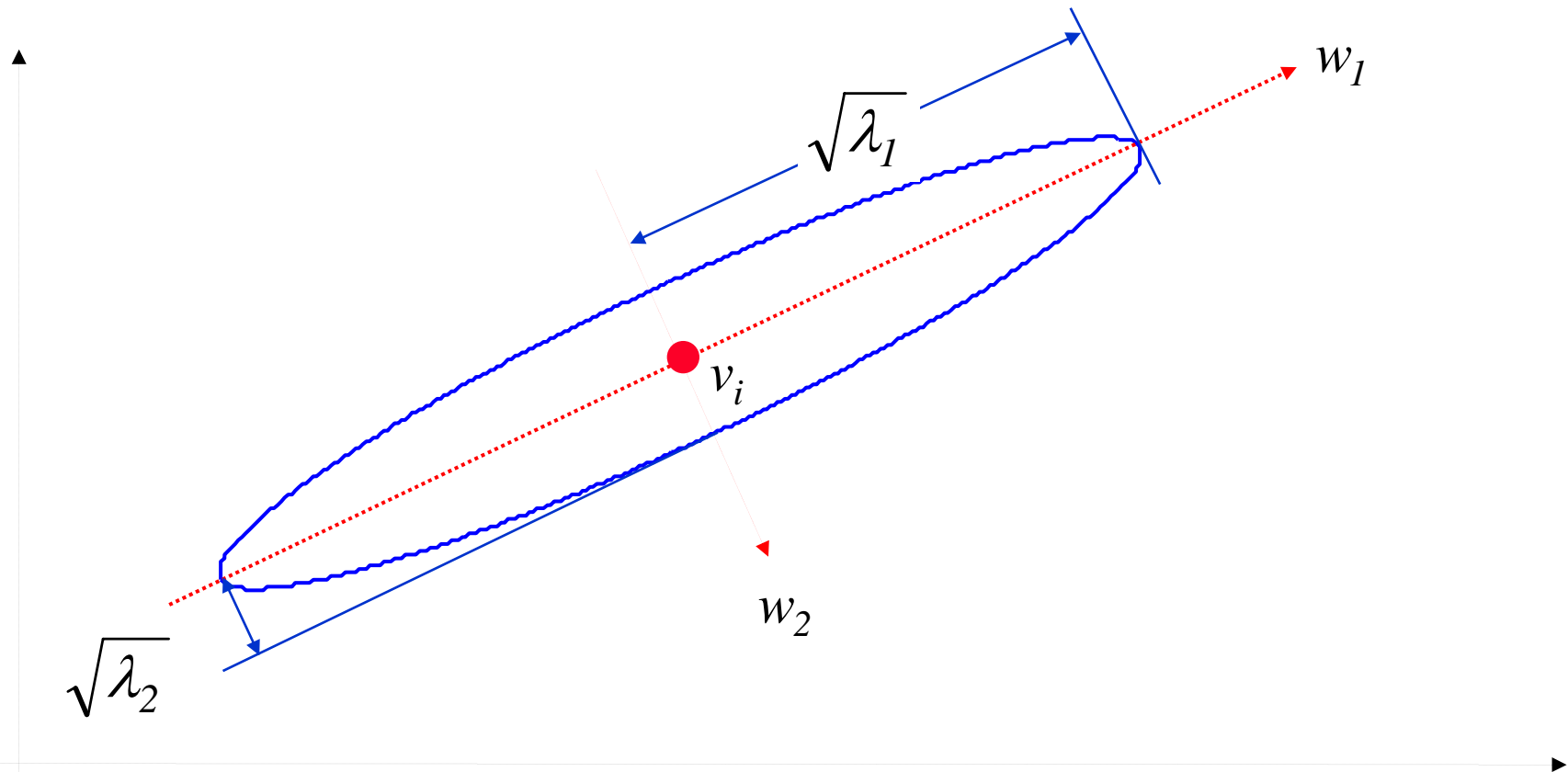
$$F_i = \frac{\sum_{k=1}^N (\mu_{ik})^m (z_k - v_i)(z_k - v_i)^T}{\sum_{k=1}^N (\mu_{ik})^m}$$

- ☞ La matrice che induce la norma A_i è funzione della matrice di covarianza fuzzy

$$A_i = [\rho_i \det(F_i)]^{1/p} \times F_i^{-1}$$

- ☞ In tal modo $d_{ik}^2_{A_i}$ può essere considerata come una norma di Mahalanobis fra il k -esimo punto (z_k) e il i -esimo centroide v_i .

Ellissoide definito da F_i^{-1}



La forma quadratica $(z_k - v_i)^T F_i^{-1} (z_k - v_i)$ definisce un iper-ellissoide attraverso i suoi autovalori w ed i suoi autovalori λ

Algoritmo di Gustafson-Kessel

👉 Calcolo
dei centroidi

$$\mathbf{v}_i = \frac{\sum_{k=1}^N (\mu_{ik})^m \mathbf{z}_k}{\sum_{k=1}^N (\mu_{ik})^m} \quad i \in (1, c)$$

👉 Calcolo delle matrici
di covarianza per
ciascun cluster

$$\mathbf{F}_i = \frac{\sum_{k=1}^N (\mu_{ik})^m (\mathbf{z}_k - \mathbf{v}_i)(\mathbf{z}_k - \mathbf{v}_i)^T}{\sum_{k=1}^N (\mu_{ik})^m}$$

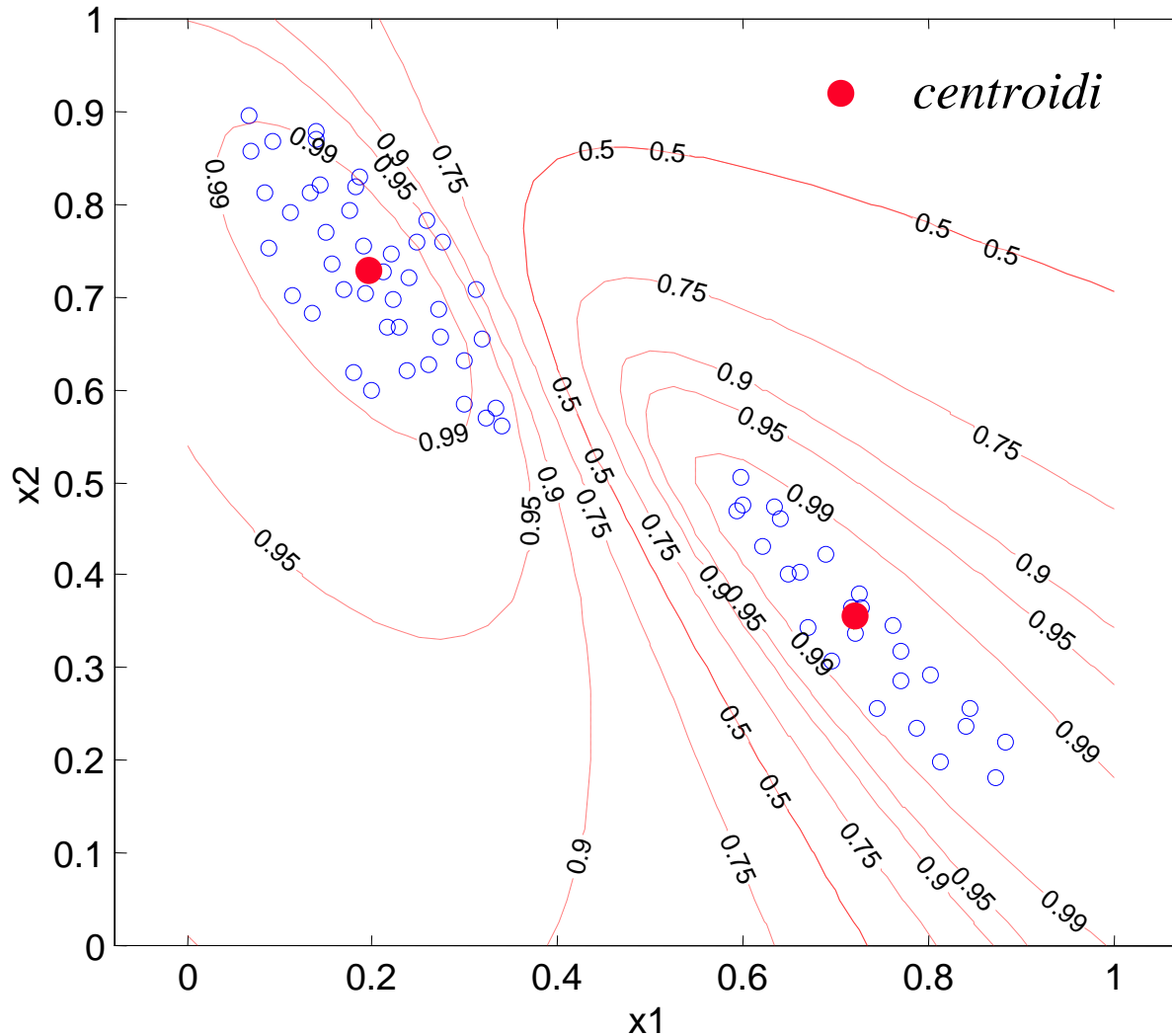
👉 Calcolo delle
distanze

$$d_{ikA_i}^2 = (\mathbf{z}_k - \mathbf{v}_i)^T [\rho_i \det(\mathbf{F}_i)]^{1/p} \times \mathbf{F}_i^{-1} (\mathbf{z}_k - \mathbf{v}_i)$$

👉 Aggiornamento
della partizione

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ikA_i}}{d_{jkA_i}} \right)^{\frac{2}{m-1}}}$$

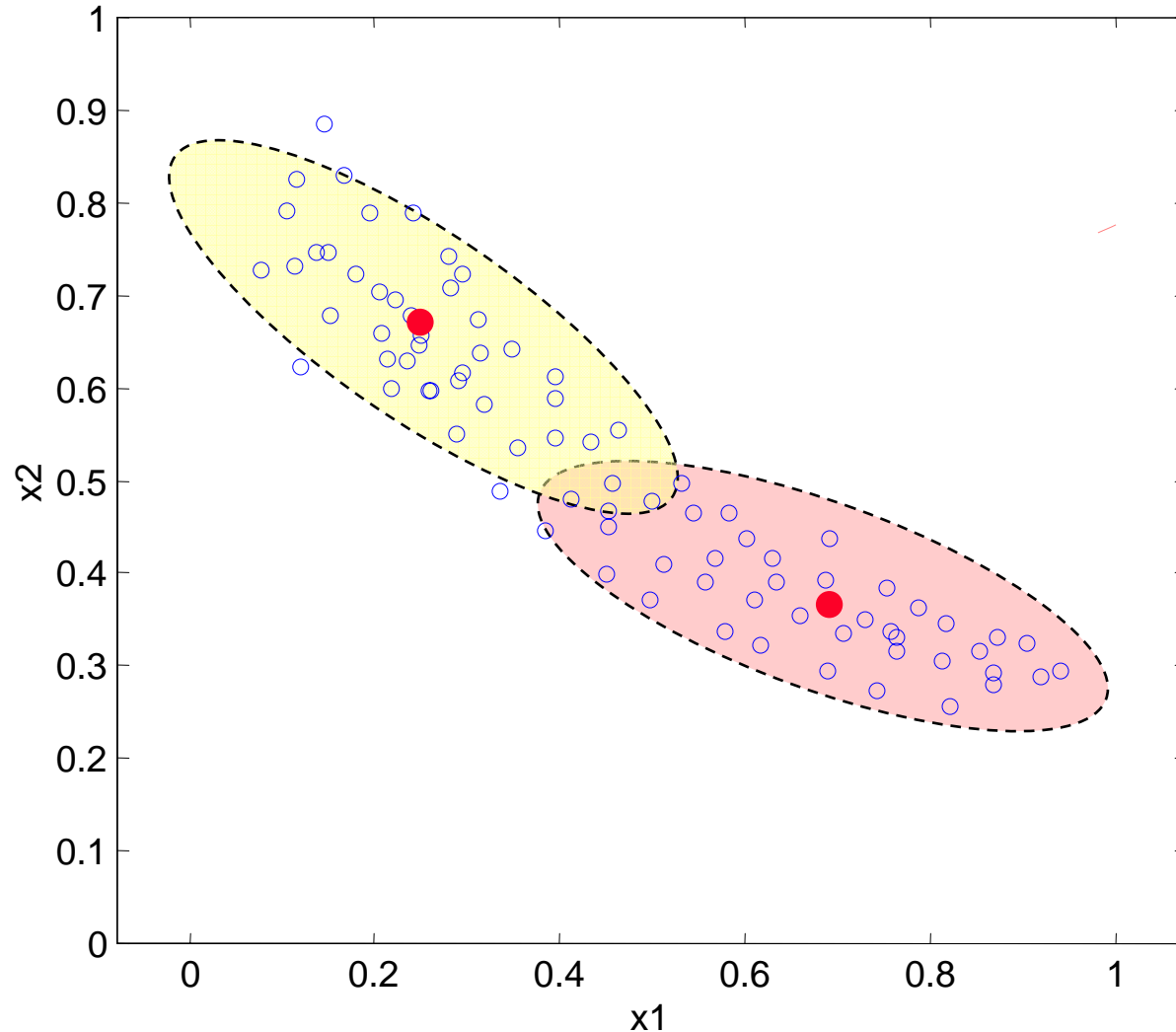
Adattamento del GK ai dati



Il vantaggio di GK rispetto a FCM è che può individuare la forma e l'orientamento dei cluster, adattandosi ai dati.

Lo svantaggio di GK è che $\det(\mathbf{F})$ e \mathbf{F}^{-1} devono essere calcolati ad ogni iterazione

Il GK fornisce clusters di uguale volume



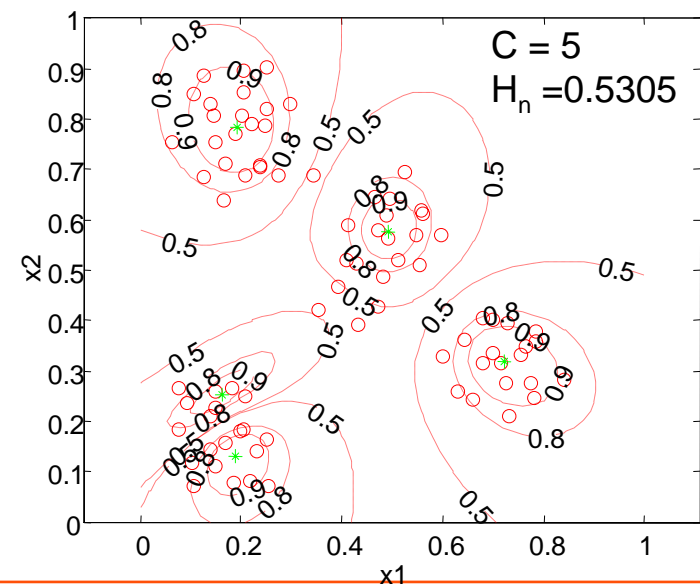
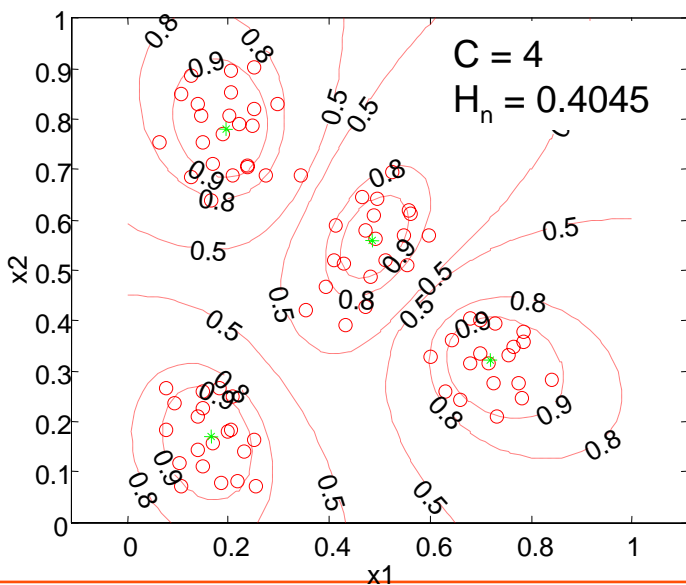
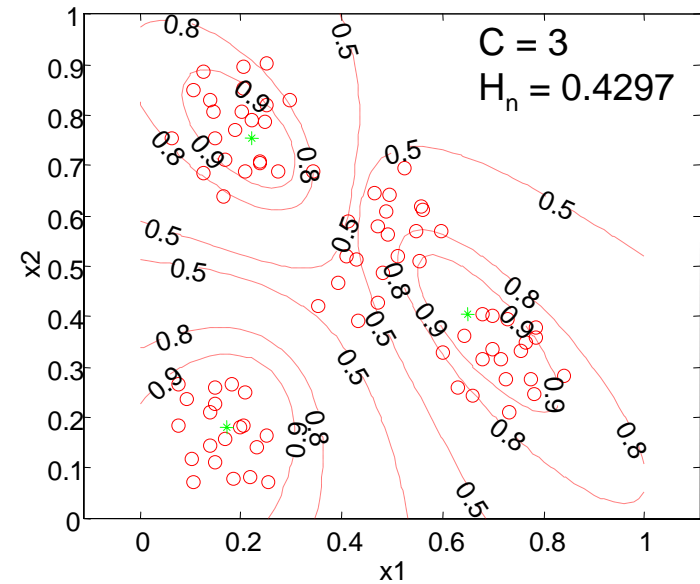
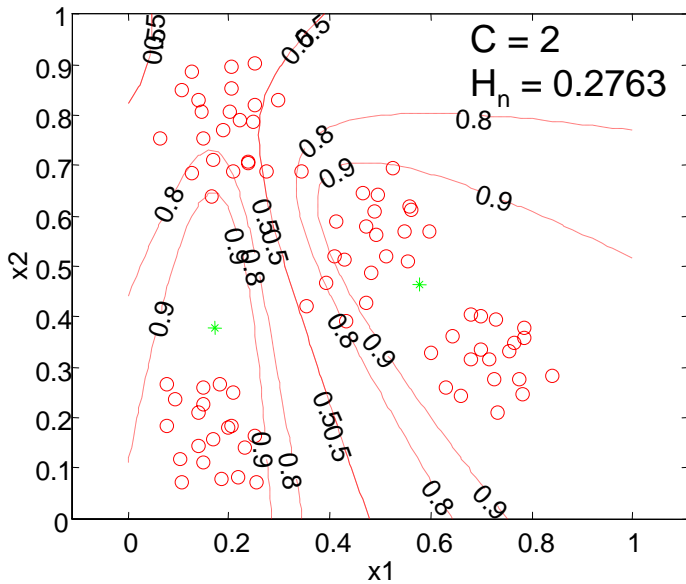
$$A_i = [\rho_i \det(F_i)]^{1/p} \times F_i^{-1}$$

Le matrici che inducono la norma A_i sono funzioni del volume dei clusters.

Tali volumi ρ_i sono imposti dall'utente, che normalmente li pone tutti uguali.

L'algoritmo non li può modificare, perciò il GK non va bene per clusters con volumi molto diversi.

Partizione dei dati precedenti con GK



Fuzzy maximum likelihood estimate (FMLE) clustering

☞ In questo caso la norma è basata sulle stime *fuzzy maximum likelihood*

$$d_{ik\Sigma} = \frac{(\det \Sigma_i)^{\frac{1}{2}}}{\left(\frac{1}{N} \sum_{k=1}^N \mu_{ik} \right)} \exp\left(\frac{1}{2} (\mathbf{z}_k - \mathbf{v}_i)^T \Sigma_i^{-1} (\mathbf{z}_k - \mathbf{v}_i) \right)$$

☞ La covarianza fuzzy dell' i -esimo cluster è definita come

$$\Sigma_i = \frac{\sum_{k=1}^N \mu_{ik} (\mathbf{z}_k - \mathbf{v}_i)(\mathbf{z}_k - \mathbf{v}_i)^T}{\sum_{k=1}^N \mu_{ik}}$$

☞ La differenza con la matrice F di GK è che Σ_i non ha l'esponente fuzzy m .

Miglioramenti di FMLE

- ☞ Lo schema iterativo di FMLE è molto simile al GK
- ☞ La matrice Σ_i è simile alla matrice di covarianza fuzzy F_i di GK, ma non ha l'esponente fuzzy m
- ☞ FMLE è capace di rivelare clusters di forme e dimensioni diverse. Questo grazie a:
 - ⇒ L'introduzione della distanza esponenziale
 - ⇒ La mancanza di vincolo di volume sui clusters
- ☞ La convergenza di FMLE è più critica e tende a convergere a un minimo locale a causa della distanza esponenziale
- ☞ Perciò ha bisogno di una buona inizializzazione (es. GK)

Algoritmo FMLE

👉 Calcola i centroidi

$$\mathbf{v}_i = \frac{\sum_{k=1}^N (\mu_{ik})^m \mathbf{z}_k}{\sum_{k=1}^N (\mu_{ik})^m} \quad i \in (1, c)$$

👉 Calcola la matrice di covarianza

$$\Sigma_i = \frac{\sum_{k=1}^N \mu_{ik} (\mathbf{z}_k - \mathbf{v}_i)(\mathbf{z}_k - \mathbf{v}_i)^T}{\sum_{k=1}^N \mu_{ik}}$$

👉 Calcola la distanza

$$d_{ik\Sigma} = \frac{(\det \Sigma_i)^{\frac{1}{2}}}{\left(\frac{1}{N} \sum_{k=1}^N \mu_{ik} \right)} \exp\left(\frac{1}{2} (\mathbf{z}_k - \mathbf{v}_i)^T \Sigma_i^{-1} (\mathbf{z}_k - \mathbf{v}_i) \right)$$

👉 Aggiorna la partizione

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ikA_i}}{d_{jkA_i}} \right)^{\frac{2}{m-1}}}$$

FLME clusters of differing volume

