Recupero di dati ed elaborazione di segnali e immagini per bioinformatica

MODULO: Riconoscimento e Recupero dell'informazione per Bioinformatica

Manuele Bicego

Corso di Laurea in Bioinformatica
Dipartimento di Informatica - Università di Verona

Metodologie di Clustering

Disegnare un sistema di clustering

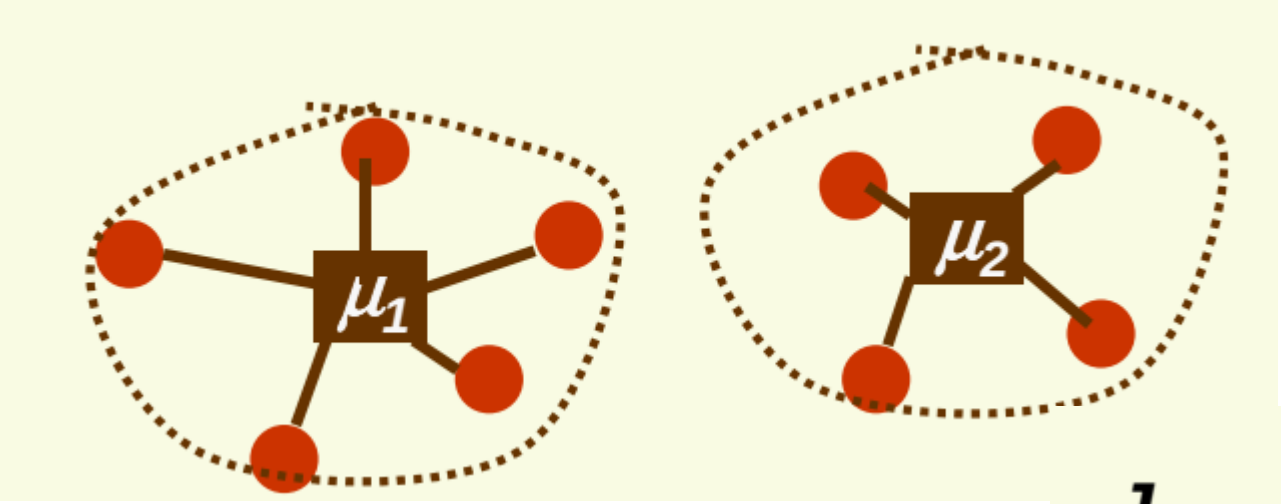
- Il tipico approccio per disegnare un sistema di clustering consiste in due passi:
  - Definire un criterio per misurare quanto “buono” è un dato clustering
  - Definire un algoritmo per calcolare il clustering (ad esempio ottimizzando il criterio definito nel passo precedente)

- Problema: il numero di clustering possibili è ENORME (numero delle possibili partizioni di un insieme)

  Esempio: 100 oggetti, 5 clusters → 10e68 possibilità!

Criteri di Clustering

- Esempio di un criterio: il *sum-of-squared errors* SSE (per rappresentazioni vettoriali)
- Idea: gli oggetti che appartengono al cluster devono essere vicini al suo centroide



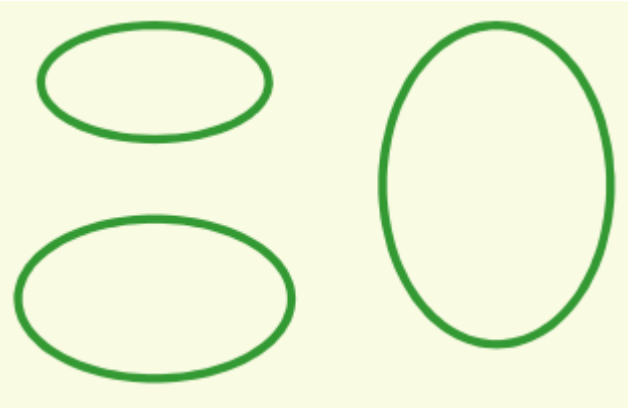
Criterio da minimizzare

$$J_{SSE} = \sum_{i=1}^{c} \sum_{x \in D_i} \| x - \mu_i \|^2$$

Il numero di cluster (c) è fissato

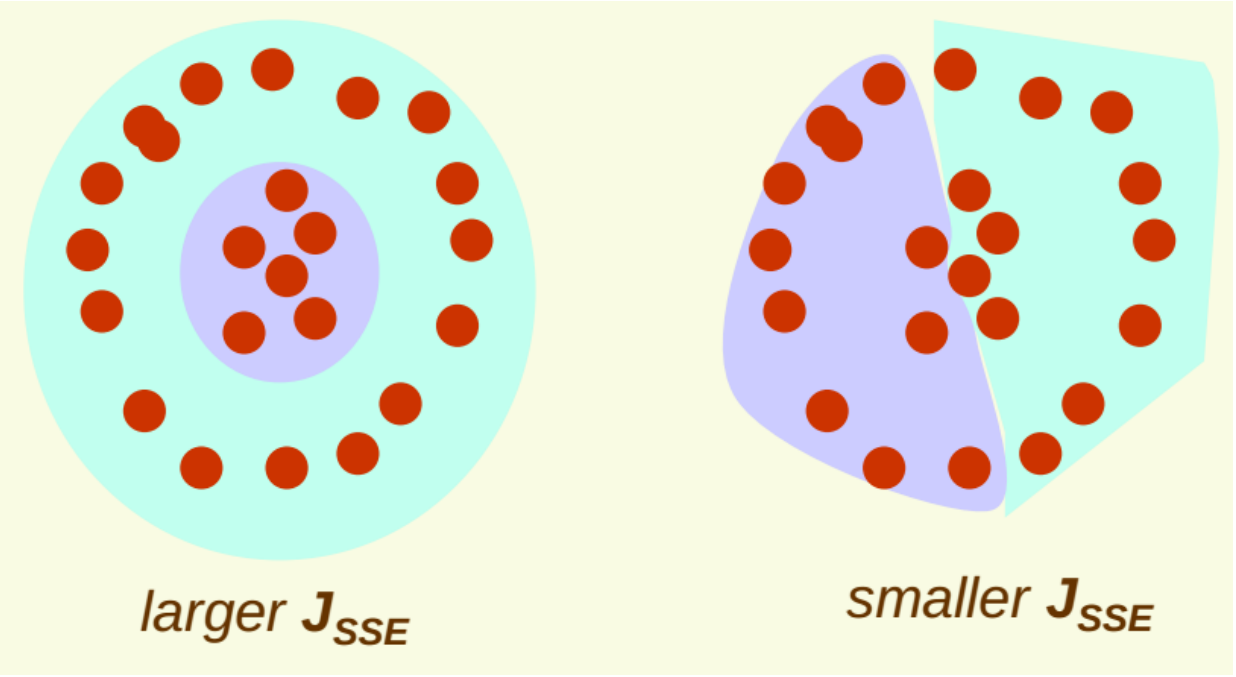
Criteri di Clustering

Questo criterio va bene se i cluster sono compatti e ragionevolmente separati



Sorgono problemi quando questo non vale…

Esempio: l'anello esterno non è compatto



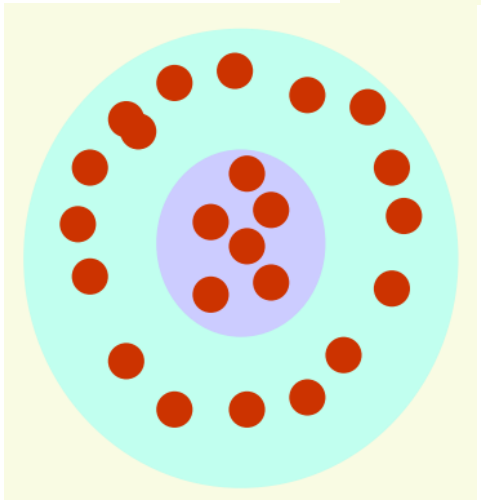
Criteri di Clustering

Altro problema. Questo criterio assume che i cluster siano più o meno della stessa dimensione, non è adeguato quando i cluster hanno dimensioni differenti

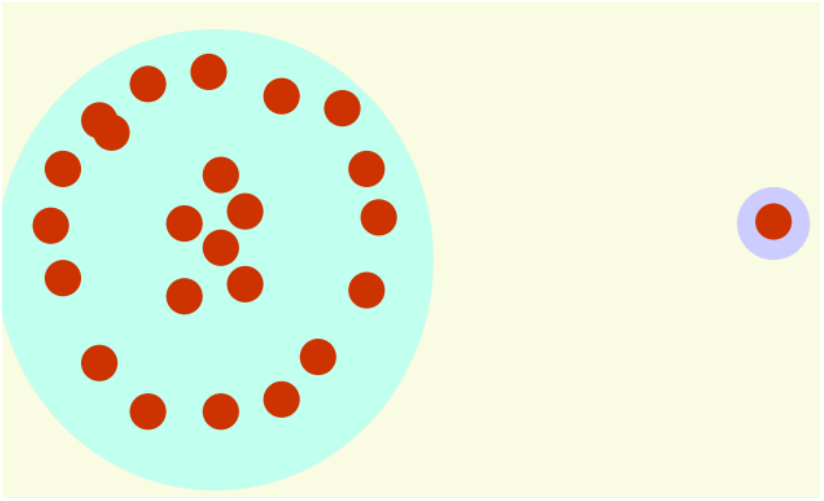
Criteri di Clustering

- Scegliere il criterio più adeguato è ovviamente difficile (il concetto di cluster è definito in modo "vago")
- Alcuni criteri sono adeguati in alcuni scenari ma non in altri, è necessario usare informazioni a priori!!

Esempio: $$J_{\max} = \sum_{i=1}^{c} n_i \left[\max_{y \in D_i, x \in D_i} \| x - y \|^2 \right]$$



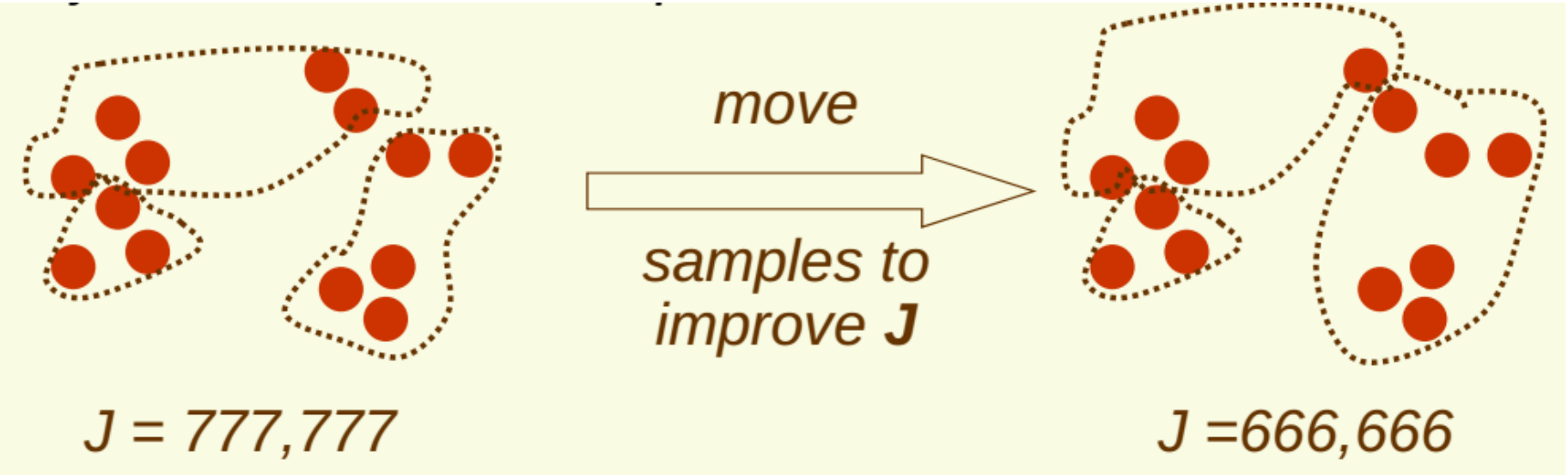
Buono in questo caso



Non robusto agli *outliers*

Algoritmi di clustering

- Rappresenta l'algoritmo utilizzato per trovare il clustering ottimale (problema: lo spazio di ricerca è enorme!!)
- Scelta classica: un algoritmo iterativo
  - Trovare una ragionevole partizione iniziale
  - Ripetere: spostare oggetti da un gruppo ad un altro in modo che la funzione obiettivo J migliori



Nota preliminare

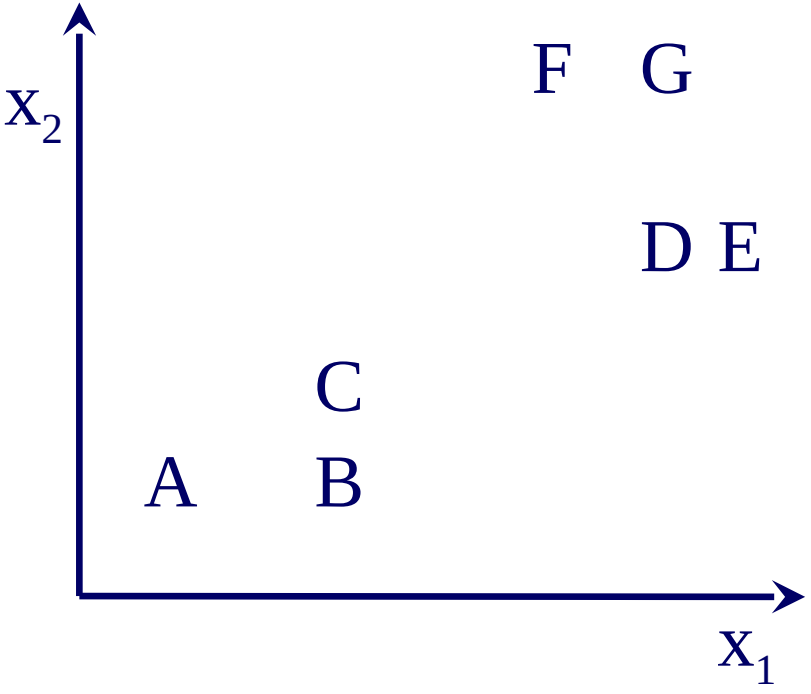
- Esistono moltissimi algoritmi di clustering
- Questi algoritmi possono essere analizzati da svariati punti di vista
- La suddivisione principale tuttavia è quella che raggruppa i metodi di clustering in due categorie: metodi partizionali e metodi gerarchici

Gerarchico vs partizionale

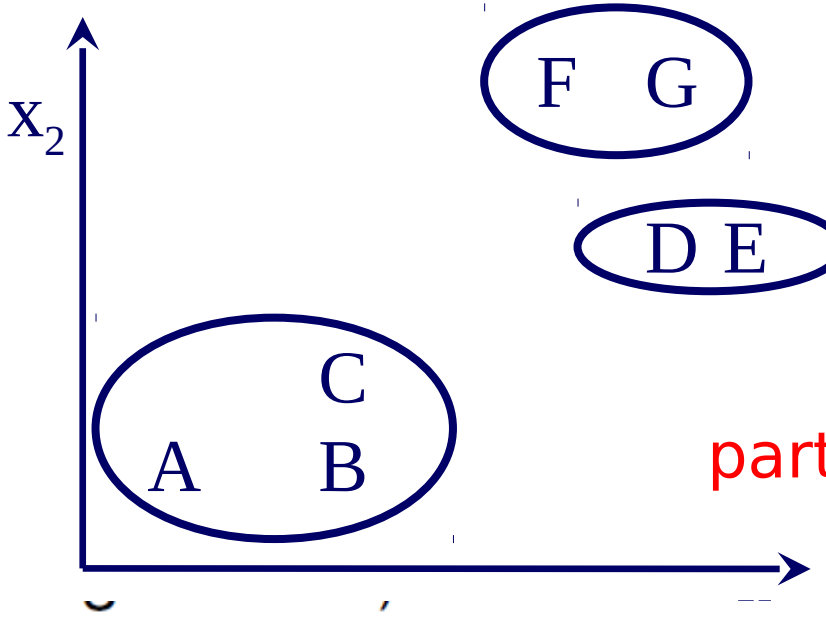
La differenza risiede nel tipo di risultato dell'operazione di clustering

- **Clustering Partizionale**: il risultato è una singola partizione dei dati (insieme di cluster disgiunti la cui unione ritorna il data set originale)

- **Clustering Gerarchico**: il risultato è una serie di partizioni innestate (un "dendrogramma")

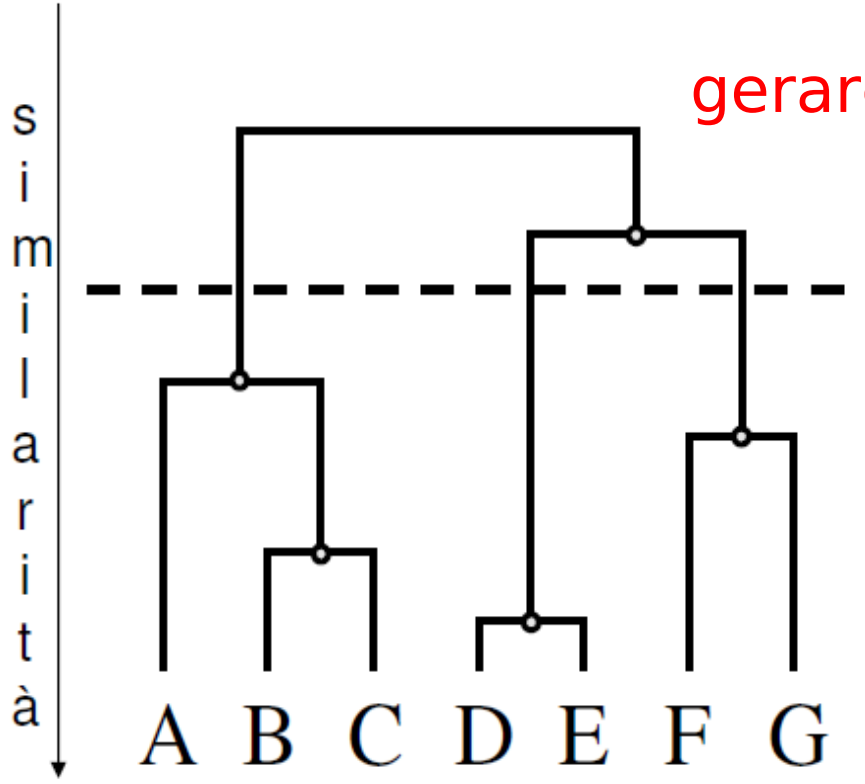


problema originale



partizionale

gerarchico



Clustering Partizionale

VANTAGGI:

- Ottimo riassunto dei dati: identifica i gruppi naturali presenti nel dataset
- Ideale per dataset grandi, molto più veloce dei metodi gerarchici

SVANTAGGI

- tipicamente richiede che i dati siano rappresentati in forma vettoriale
- scegliere il numero di cluster è un problema (esistono metodi per determinarlo in modo automatico)
- tipicamente estrae solo cluster convessi

Esempi: K-means (e sue varianti), PAM, ISODATA, DBSCAN,..

Clustering gerarchico

VANTAGGI

- ⇨ riesce ad evidenziare le relazioni tra i vari pattern del dataset (più informativo del partizionale)
- ⇨ tipicamente richiede in ingresso una matrice di prossimità (non necessita quindi di dati in forma vettoriale)
- non è necessario settare a priori il numero di cluster
- riesce a caratterizzare anche clusters di forma non convessa

SVANTAGGI

- ⇨ è improponibile per dataset grandi
- ⇨ molti degli algoritmi gerarchici sono greedy (subottimali)

Esempi: Complete Link, Single Link, Ward Link,...

Alcuni algoritmi di clustering

Sommario

- Basic Sequential Algorithmic Scheme (BSAS)
- K-Means (e sue varianti)
- Algoritmi gerarchici agglomerativi (Single Link, Complete Link)
- Misture di Gaussiane (cenni)

Basic Sequential Algorithmic Scheme (BSAS)

Basic Sequential Algorithmic Scheme (BSAS)

Caratteristiche

- Algoritmo **partizionale** di tipo **sequenziale**: i pattern vengono processati in modo sequenziale (uno dopo l'altro)

Idea principale

- i pattern vengono processati una volta sola, uno dopo l'altro (l'ordine può essere casuale)
- ogni pattern processato viene assegnato ad un cluster esistente oppure va a creare un nuovo cluster (sulla base della similarità con i cluster formati fino a quel momento)

BSAS: algoritmo

Notazioni

- $\mathbf{x}_i$: vettore di punti, $\{\mathbf{x}_1, \ldots \mathbf{x}_N\}$ dataset da clusterizzare
- C_j: j-esimo cluster
- m: numero di cluster trovati ad un determinato istante

Parametri da definire:

- $d(\mathbf{x}, C)$: distanza tra un punto e un insieme (un cluster)
  - ⇨ Max: distanza massima
  - ⇨ Min: distanza minima
  - ⇨ Average: distanza media
  - ⇨ center-based: distanza dal “rappresentante”
- Θ: soglia di dissimilarità

BSAS: algoritmo

m=1

C_m = x_1

for i = 2 to N

trova C_k tale che $d(\mathrm{x}_i, C_k) = min_{1 \leq j \leq m} d(\mathrm{x}_i, C_j)$

if $d(\mathrm{x}_i, C_k)$ > θ

m = m+1

C_m = $\{\mathrm{x}_i\}$

else

$C_k = C_k \cup \{\mathrm{x}_i\}$

(se necessario aggiornare i rappresentanti)

end if

end for

BSAS

VANTAGGI:

- Approccio di clustering molto semplice e intuitivo
- Il numero di cluster non è conosciuto a priori ma viene stimato durante il processo
- Funziona anche per dati non vettoriali (si basa solo sulla definizione di distanza)
- Funziona sia con la distanza che con la similarità (basta cambiare min con max e > con <)

BSAS

SVANTAGGI:

- L'ordine con cui vengono processati i pattern è cruciale (ordini diversi possono produrre risultati diversi)
- Usando la versione "distanza da rappresentanti", e usando come rappresentanti le medie, i cluster che escono sono compatti (funziona solo per cluster convessi)
- la scelta della soglia θ è cruciale
  - ⇨ θ troppo piccola, vengono determinati troppi cluster
  - ⇨ θ troppo grande, troppo pochi cluster

BSAS

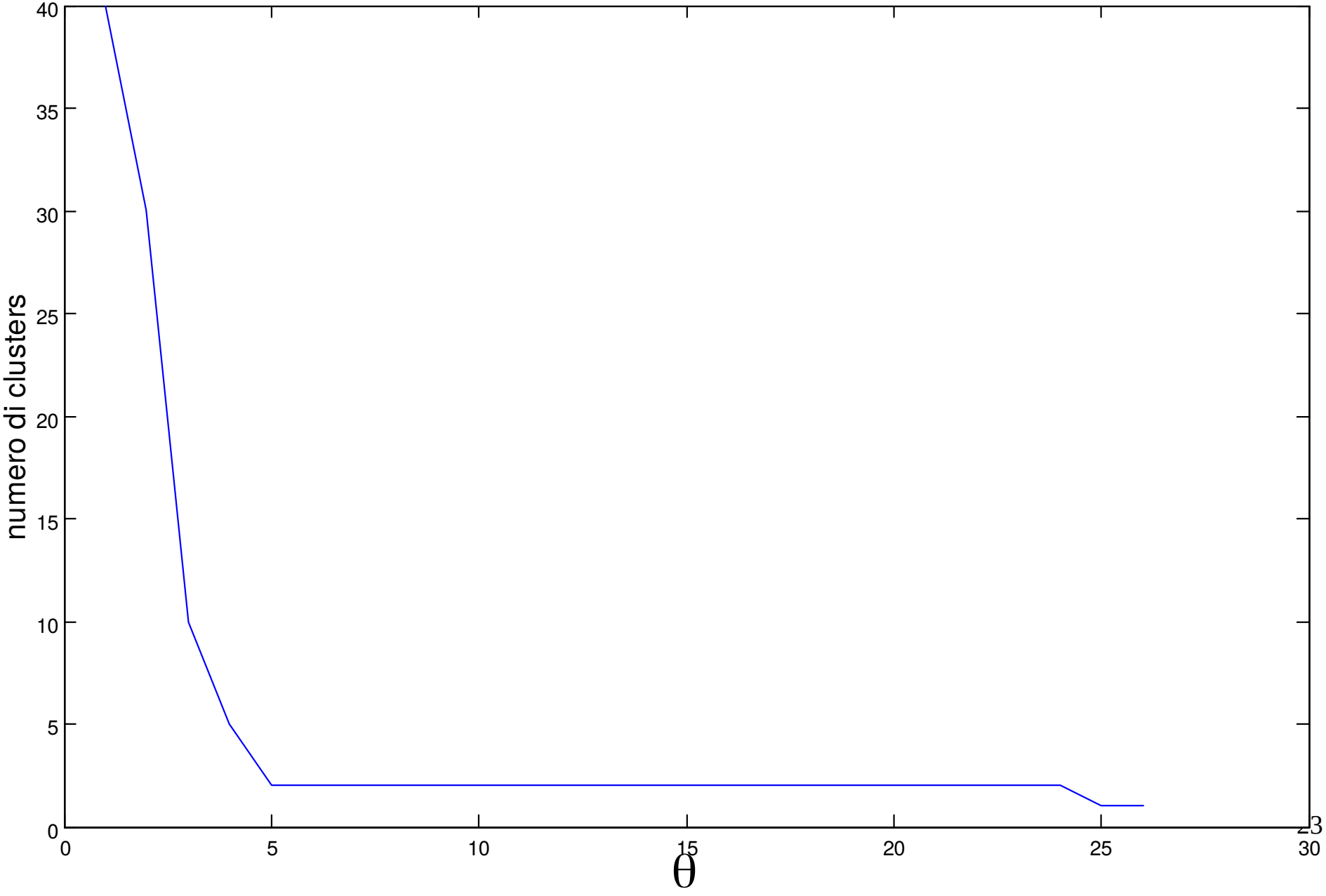
- Metodo per calcolare la soglia ottimale:

```
for θ = a to b step c
    • Eseguire s volte l'algoritmo BSAS, ogni volta
      processando i pattern con un ordine differente
    • Stimare mθ come il numero più frequente di cluster
      trovati
end for

• Visualizzare il numero di cluster mθ vs il parametro θ
• La soglia ottimale è quella corrispondente alla regione
  "piatta" più lunga (si sceglie la soglia in mezzo alla
  regione)
```

- ~~dettagli~~
  - ⇨ *a* è la distanza minima tra i punti, *b* la distanza massima
  - ⇨ assumiamo che "esista" un clustering

BSAS



K-Means

K-means

Caratteristiche

- Algoritmo più famoso di clustering partizionale
- E' un algoritmo "center-based": ogni cluster è rappresentato da un "centro"
- Ottimizza una funzione di errore

Idea principale

- Ogni cluster è rappresentato dalla sua media
- Si parte da una clusterizzazione iniziale (casuale)
- Ad ogni iterazione
  - si calcolano le medie dei clusters del passo precedente;
  - si ridetermina la clusterizzazione assegnando ogni pattern alla media più vicina
- si continua fino a convergenza

K-means: l’algoritmo

(alla lavagna)

K-means

VANTAGGI:

- Algoritmo semplice, intuitivo, molto famoso e utilizzato

- E’ molto efficiente nel clusterizzare dataset grandi, perché la sua complessità computazionale è linearmente dipendente dalla dimensione del data set

K-means

SVANTAGGI

- il numero di cluster deve essere fissato a priori
- l'ottimizzazione spesso porta ad un ottimo "locale"
- l'inizializzazione è cruciale: una cattiva inizializzazione porta ad un clustering pessimo
- Si possono ottenere solo cluster con forma convessa
- Lavora solo su dati vettoriali numerici (deve calcolare la media)
- Non funziona bene su dati altamente dimensionali (soffre del problema della curse of dimensionality)

Varianti del K-means

ISODATA (*Iterative Self-Organizing Data Analysis Techniques*): tecnica che permette lo splitting e il merging dei cluster risultanti

- Ad ogni iterazione effettua dei controlli sui cluster risultanti:
  - un cluster viene diviso se la sua varianza è sopra una soglia prefissata, oppure se ha troppi punti
  - due cluster vengono uniti se la distanza tra i due relativi centroidi è minore di un'altra soglia prefissata, oppure se hanno troppo pochi punti
- la scelta delle soglie è cruciale, ma fornisce anche una soluzione alla scelta del numero di cluster

Varianti del K-means

PAM (Partitioning around the medoids):

- l'idea è quella di utilizzare come "centri" del K-means i "medoidi" invece che le medie
- Medoide di un cluster: punto del dataset più vicino alla media
- Vantaggio: non si usa come rappresentante del cluster un elemento che non esiste (la media non è un punto "vero")

Varianti del K-means

DPAM: (Distance PAM): variante per dati “non vettoriali”

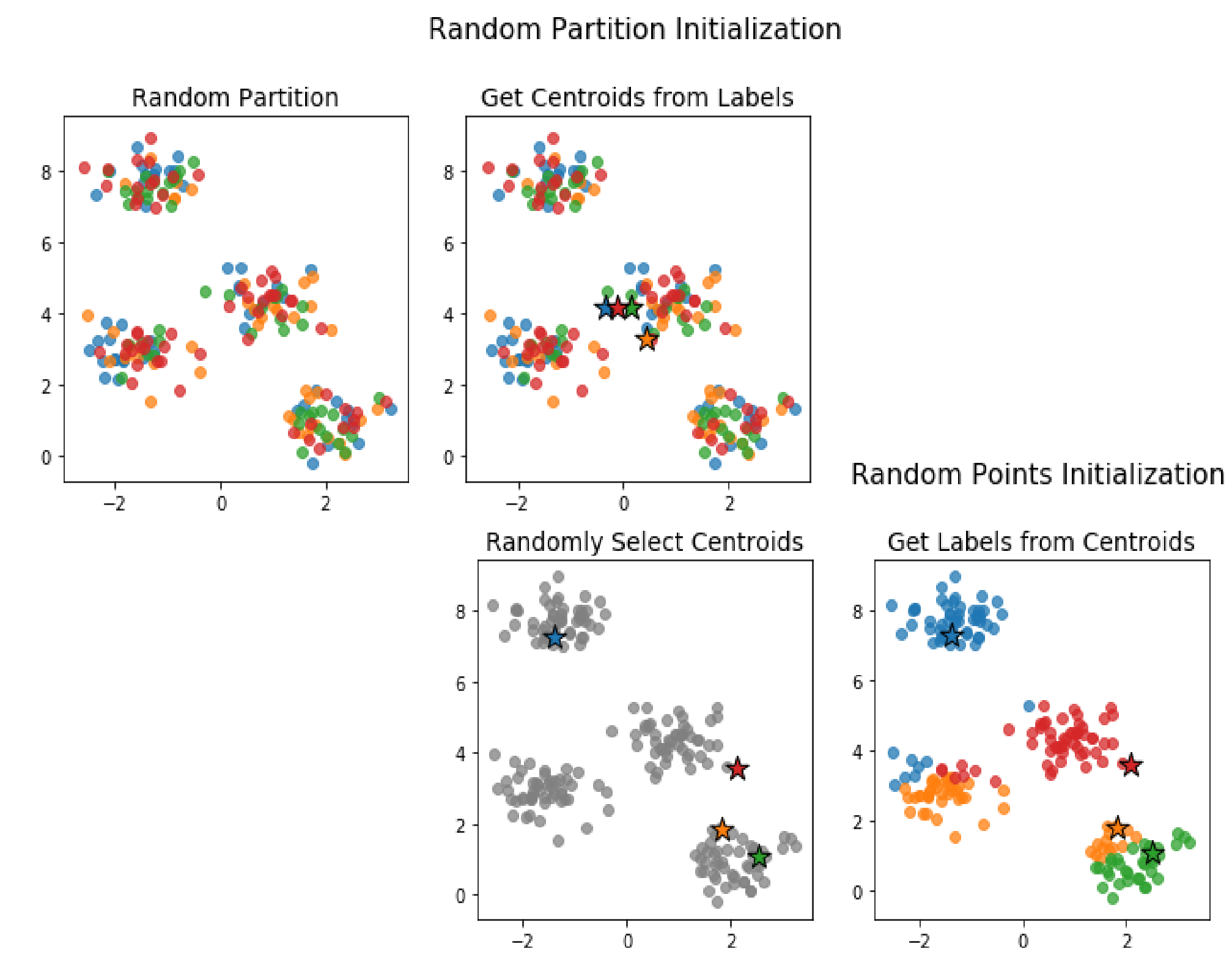
- l’idea è quella di utilizzare come “centri” del K-means gli oggetti “più centrali” di un cluster
  - Oggetto più centrale: oggetto a distanza minima da tutti gli altri oggetti del cluster
- Non è più necessario calcolare le medie, ma si lavora solo con le distanze:
  - per stimare il rappresentante uso la distanza minima da tutti gli oggetti del cluster;
  - per l’assegnamento ad un cluster uso la distanza minima dal rappresentante
- In questo modo si può lavorare anche con dati non vettoriali, serve solo una misura di distanza tra questi dati

Varianti del K-means

Nota: l'inizializzazione del K-means può essere un problema

- Inizializzazioni diverse possono portare a soluzioni diverse
- Soluzione tipica: si ripete il K-means partendo da diverse inizializzazioni casuali, e si tiene la soluzione che porta al minimo valore della funzione di errore

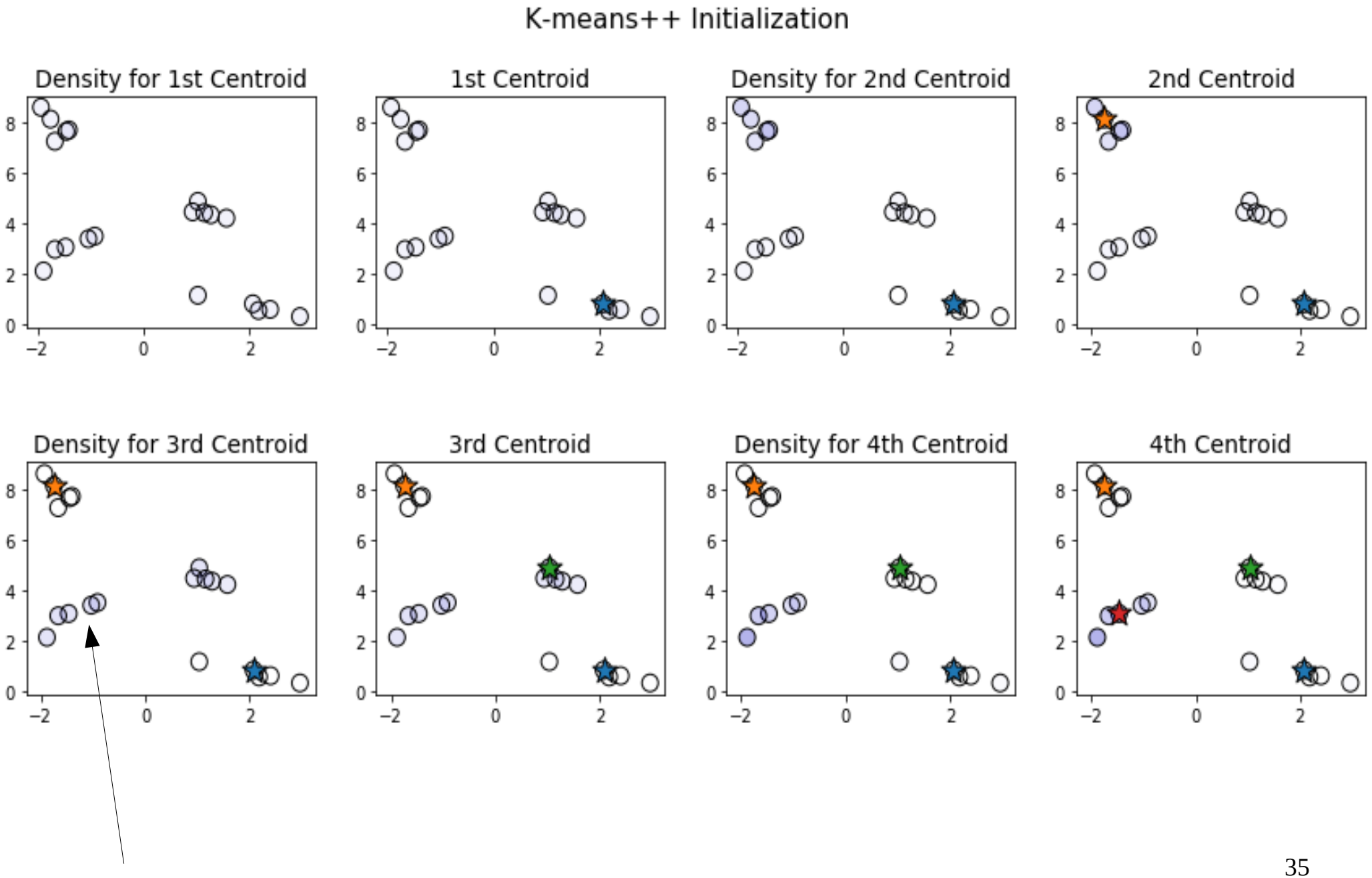
- Due possibili inizializzazioni:
  - scegliere in modo casuale i cluster e derivare le medie (Random Partition Initialization)
  - scegliere le medie come punti casuali del dataset (Random Points Initialization)


Random Partition Initialization
Random Partition
Get Centroids from Labels
Random Points Initialization
Randomly Select Centroids
Get Labels from Centroids

Varianti del K-means

K-means ++: K-means con una inizializzazione “furba”: l’algoritmo inizializza le medie in modo simile al Random Points Initialization, ma i punti non sono scelti a caso:

- La prima media è un punto scelto in modo casuale (probabilità uniforme su tutti i punti)
- La seconda media è scelta con una probabilità non uniforme: ogni punto ha una probabilità proporzionale alla sua distanza dalla prima media (è più facile che venga scelto un punto lontano dalla prima media)
- La terza media è scelta favorendo i punti lontani dalle prime due
- In questo modo le medie sono “ben distribuite”



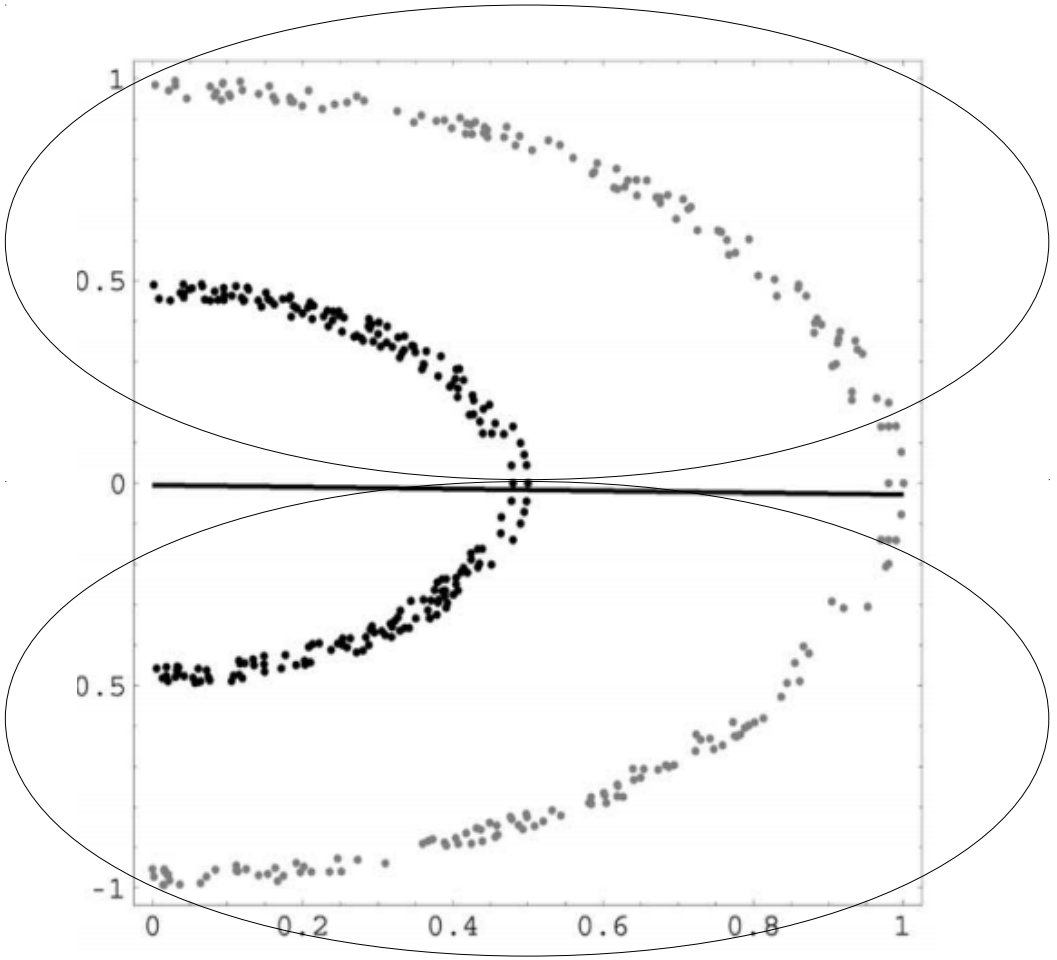
Intensità del colore: probabilità di essere scelto

Varianti del K-means

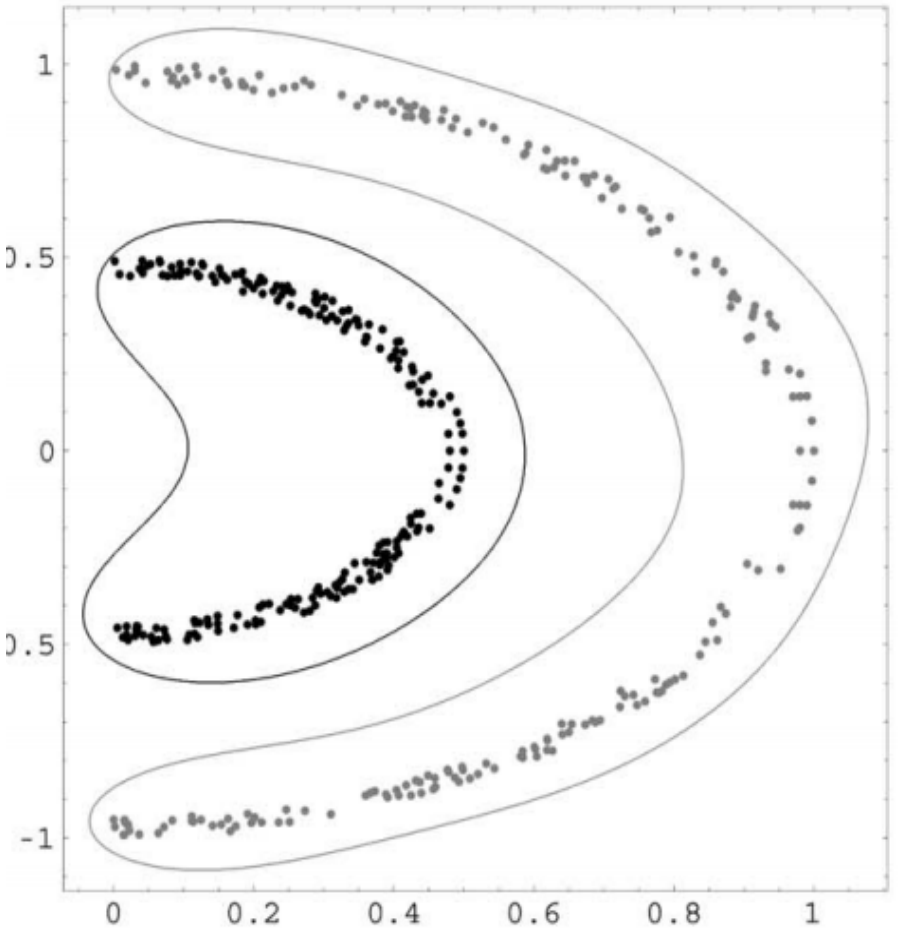
K-means “generalizzato”: K-means dove i cluster non sono più rappresentati dalle medie ma da classificatori

- Partendo da un’inizializzazione casuale, ad ogni iterazione:
  - per ogni cluster del passo precedente si addestra un classificatore
    - Classificatore uno contro tutti
    - Classificatore “One-class”
  - si ricalcolano i cluster assegnando ogni pattern al classificatore che ha la “confidenza” più alta (per esempio probabilità a posteriori più alta)
- Dipendentemente dalla flessibilità del classificatore si riescono a trovare anche cluster non convessi

Varianti del K-means



K means classico



K means “generalizzato”
(classificatore: one class SVM)

Clustering gerarchico agglomerativo

Clustering gerarchico agglomerativo

Caratteristiche

- Algoritmi di clustering gerarchico, cioè che generano una serie di partizioni innestate
- Rappresentazione di un clustering gerarchico: il dendrogramma

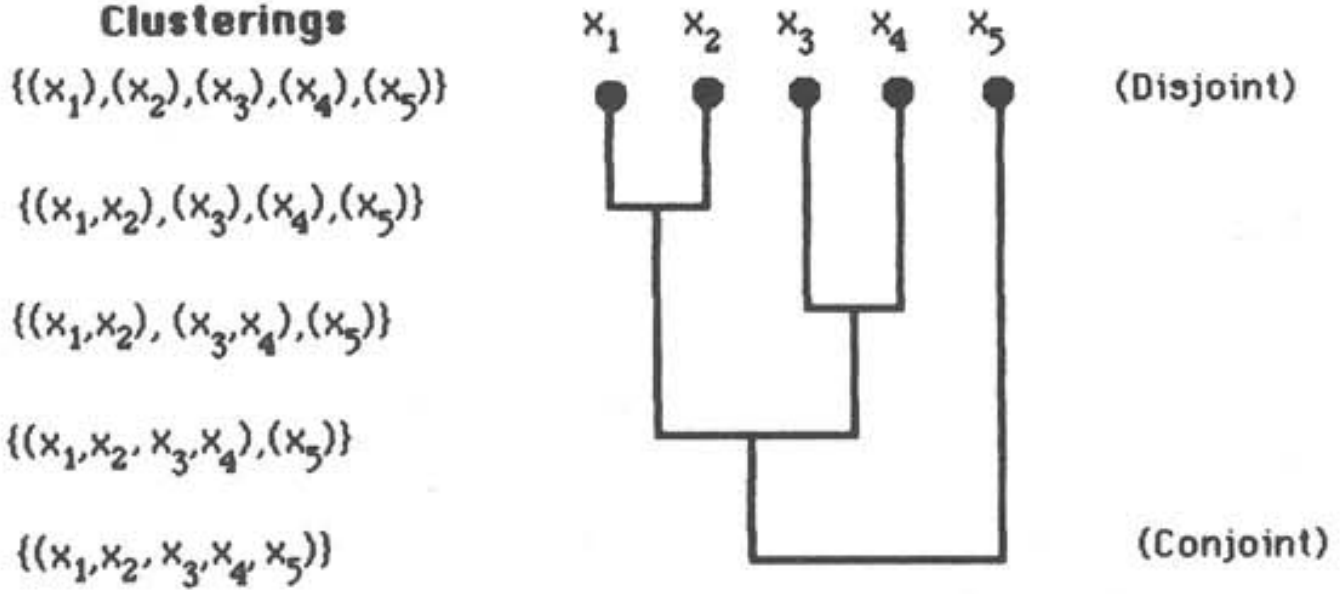


Figure 3.2 Example of dendrogram.

Clustering gerarchico agglomerativo

Idea principale

- si parte da una partizione in cui ogni cluster contiene un solo elemento
- si continua a fondere i cluster più “simili” fino ad avere un solo cluster

Nota:

- A seconda di come si implementa il concetto di “cluster più simili” si hanno algoritmi diversi
- Esempi: single link, complete link

Clustering gerarchico agglomerativo

Algoritmo: ne esistono due formulazioni, qui si vede quella basata su matrici

(alla lavagna)

Esempio

Single Link: $d(C_{rs},C_j) = \min\{d(C_r,C_j), d(C_s,C_j)\}$

Complete Link: $d(C_{rs},C_j) = \max\{d(C_r,C_j), d(C_s,C_j)\}$

	1	2	3	4	5
1	0	2.3	3.4	1.2	3.7
2		0	2.6	1.8	4.6
3			0	4.2	**0.7**
4				0	4.4
5					0

	1	2	3,5	4
1	0	2.3	3.4	**1.2**
2		0	2.6	1.8
3,5			0	4.2
4				0

	1,4	2	3,5
1,4	0	**1.8**	3.4
2		0	2.6
3,5			0

	1,2,4	3,5
1,2,4	0	**2.6**
3,5		0

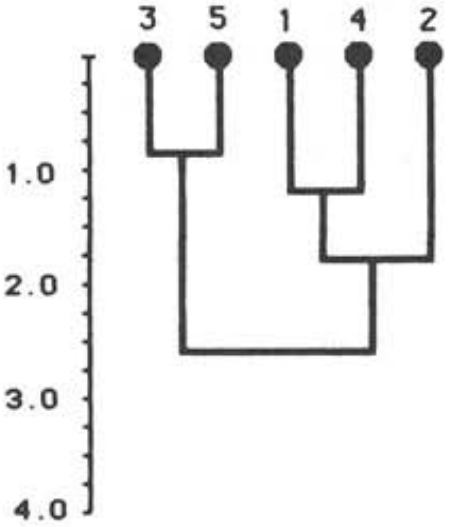
single link

	1	2	3,5	4
1	0	2.3	3.7	**1.2**
2		0	4.6	1.8
3,5			0	4.4
4				0

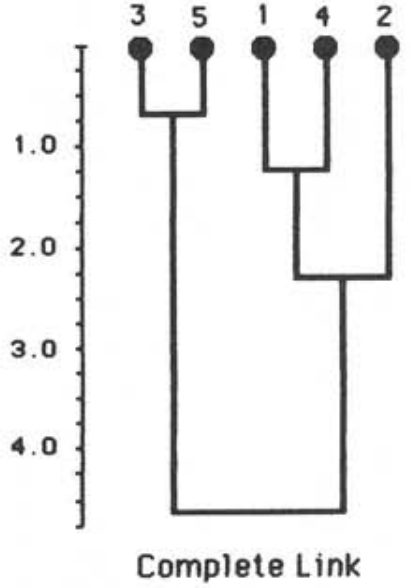
	1,4	2	3,5
1,4	0	**2.3**	4.4
2		0	4.6
3,5			0

	1,2,4	3,5
1,2,4	0	**4.6**
3,5		0

complete link

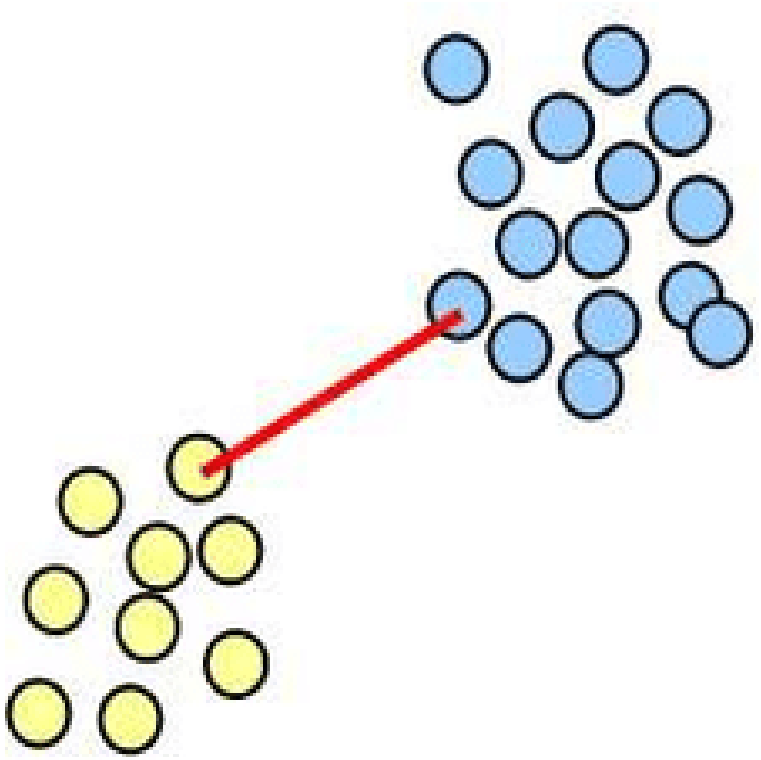


Single Link

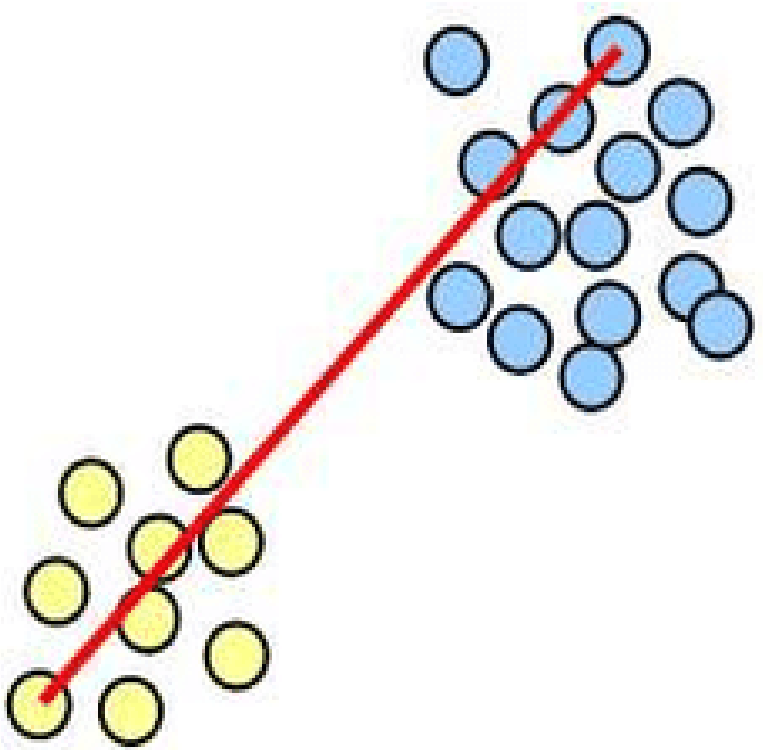


Complete Link

Distanza tra due clusters: differenza tra Single Link e Complete Link

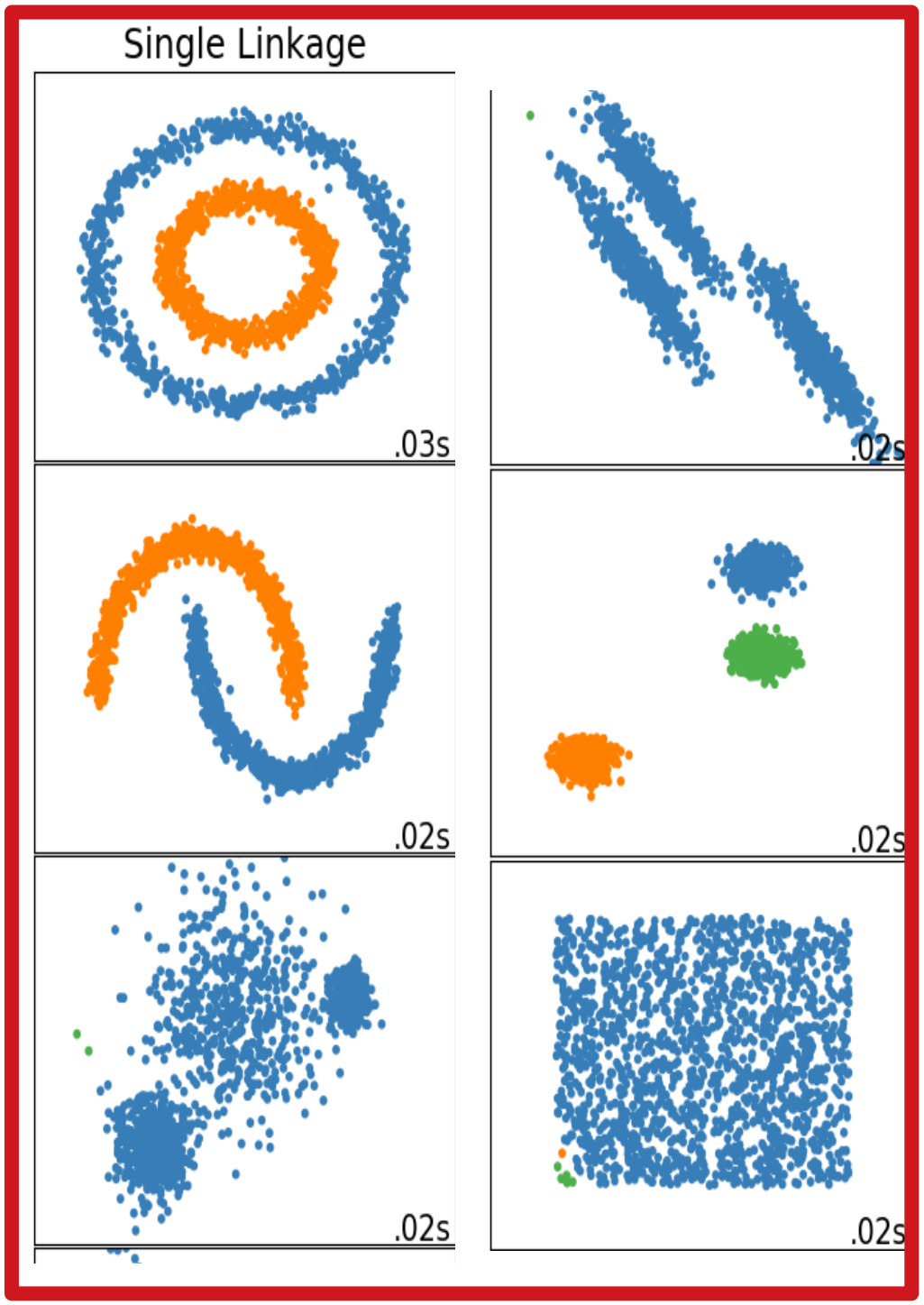


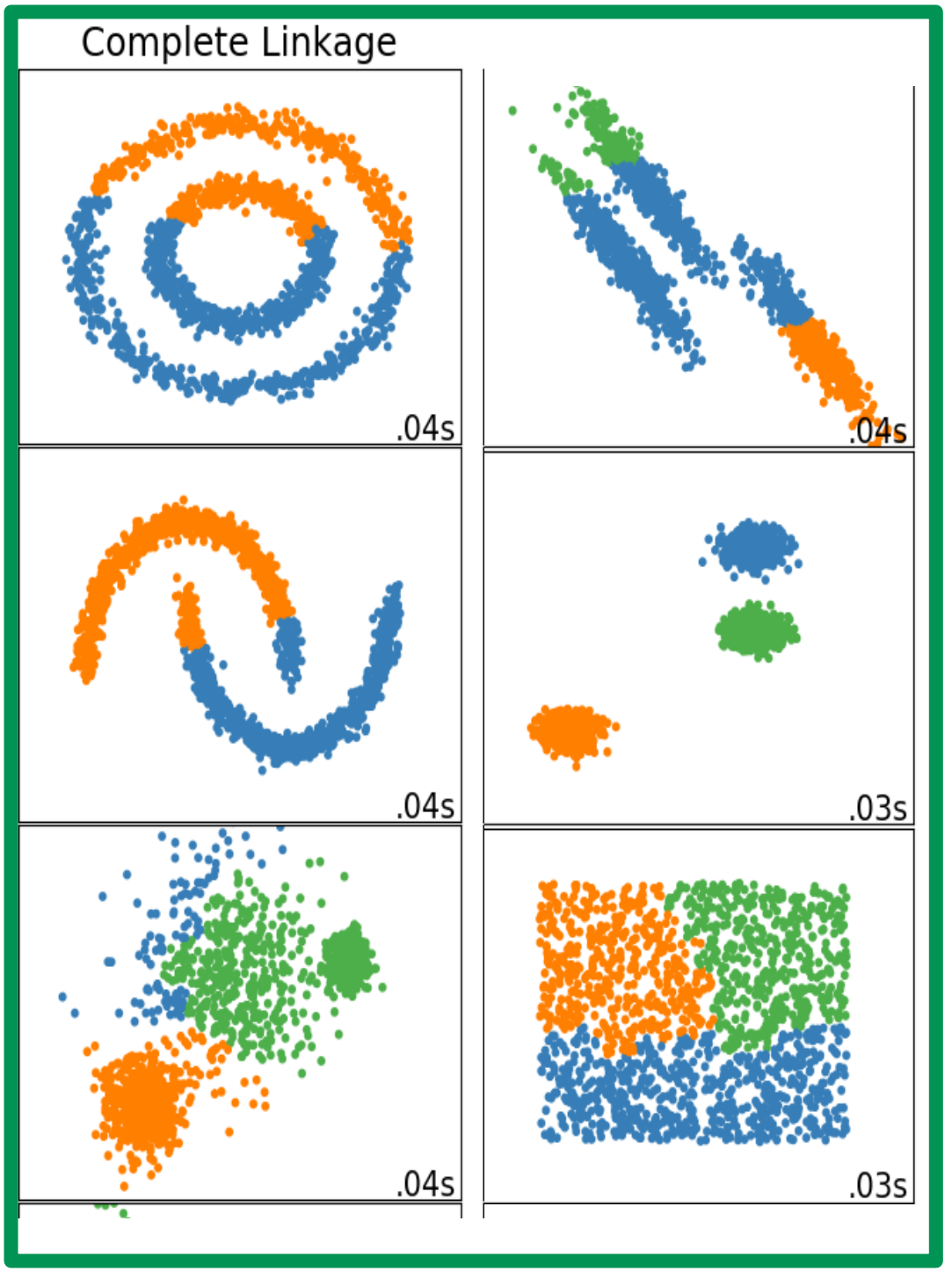
single-link



complete-link

I risultati possono essere molto diversi





Clustering gerarchico agglomerativo

Altri criteri di unione dei cluster

- UPGMA (Unweighted pair group method using arithmetic averages)
  - ⇨ la distanza tra cluster è definita come la media delle distanze di tutte le possibili coppie formate da un punto del primo e un punto del secondo
  - ⇨ utilizzato nel periodo iniziale della filogenesi

- Metodo di Ward
  - ⇨ fonde assieme i cluster che portano alla minima perdita di informazione
  - ⇨ informazione intesa in termini di varianza

Misture di Gaussiane (cenni)

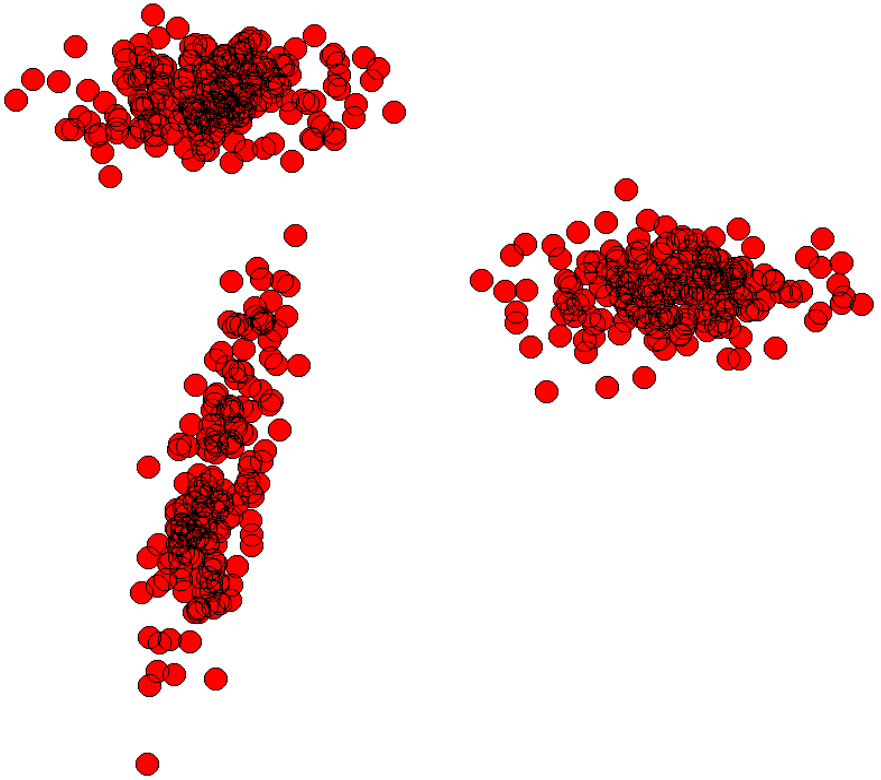
Misture di Gaussiane

Caratteristiche:

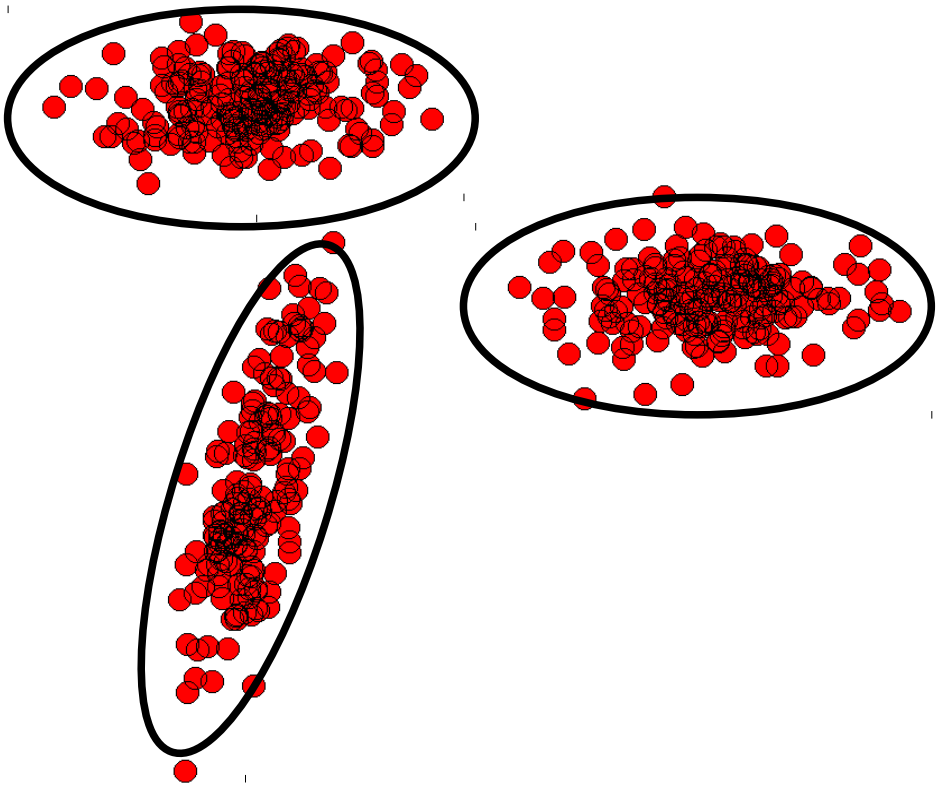
- Algoritmo più famoso di model-based clustering:
  - tecniche di clustering dove si creano dei modelli (tipicamente probabilistici) per i dati
  - l'obiettivo diventa quello di massimizzare il fit tra i modelli e i dati

Idea principale:

- Si assume che i dati siano generati da una mistura di Gaussiane, in cui ogni componente identifica un cluster
- Si cerca di stimare dai dati i parametri della mistura (attraverso una stima Maximum Likelihood)



Dati da clusterizzare



Mistura di Gaussiane (Gaussian Mixture Model)

Misture di Gaussiane

Dettagli:

- Una mistura (in generale) è descritta dalla seguente formula

$$p(x)=\sum_{j=1}^{K}\pi_j f_j(x|\Theta_j)$$

πj è la probabilità a priori della j-esima componente

(La probabilità è la somma pesata delle probabilità delle varie componenti)

- nel caso di mistura di Gaussiane, ogni componente è una Gaussiana

$$f_j(x|\Theta_j)=\frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}\exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$

Misture di Gaussiane

- Per stimare i parametri si utilizza un approccio “Maximum Likelihood”
- Dato un dataset D che contiene N punti D={x1..xN}, si trova la mistura che massimizza la likelihood (“quanto bene” il modello spiega i dati)
  - Likelihood: produttoria di tutti i p(xi)

$$Lik = \prod_{i=1}^{N} \sum_{j=1}^{K} \pi_j N(x_i | \mu_j, \Sigma_j)$$

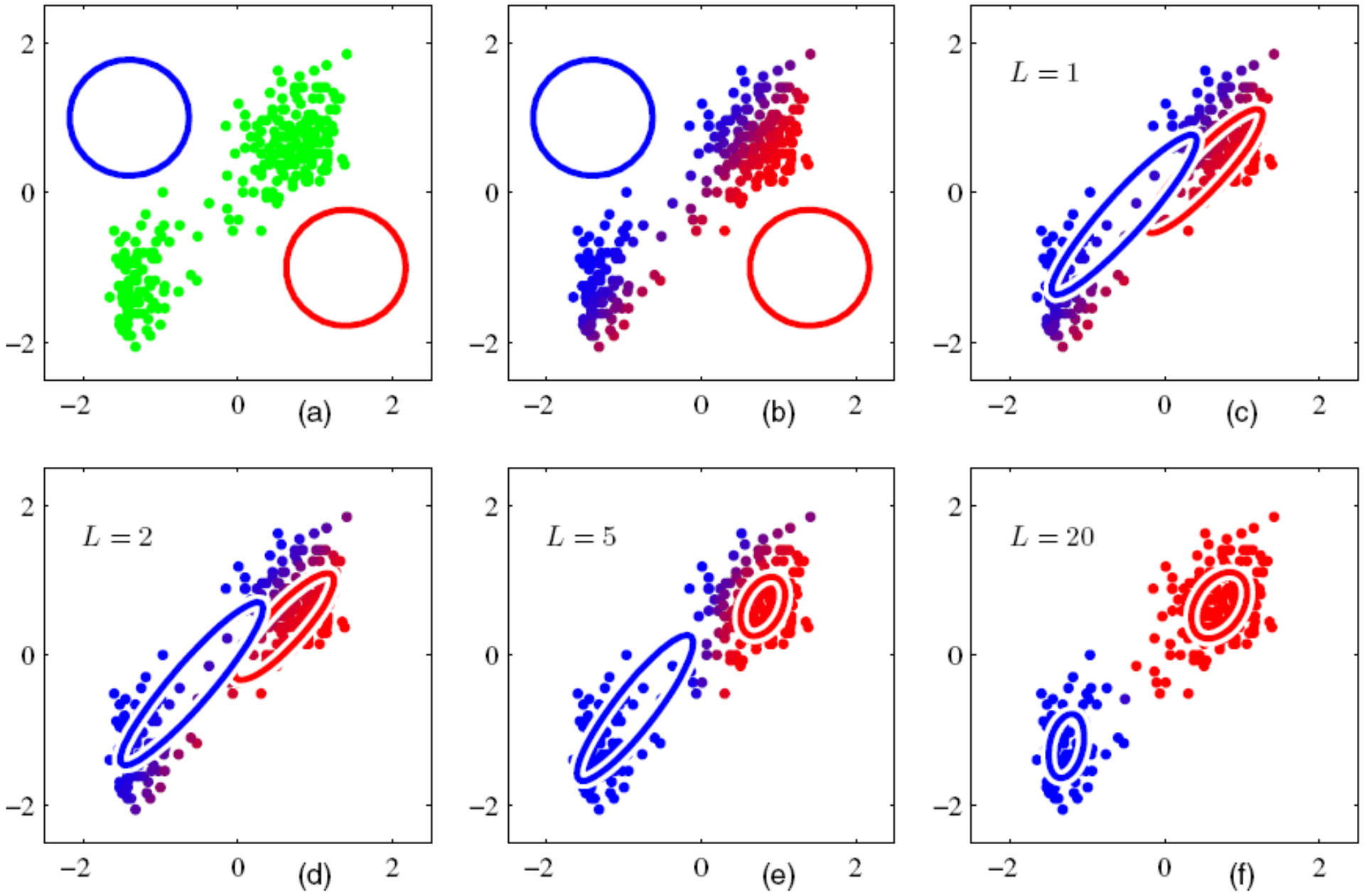
- Funzione molto difficile da ottimizzare, tipicamente non si può fare in modo analitico, di solito si utilizza l'EM (Expectation Maximization)

Misture di Gaussiane

IDEE: (Non vediamo nel dettaglio)

- Algoritmo iterativo, parte da un modello iniziale e lo migliora ad ogni passo
- L'algoritmo assomiglia al K-means, ma tiene conto del "grado di appartenenza" ad un cluster
- Cicla continuamente tra questi due passi.
  - E-step. Data la mistura, stima il grado di appartenenza di ogni punto alle diverse Gaussiane
  - M-step. Ristima i parametri delle Gaussiane utilizzando queste informazioni

Esempio



Misture di Gaussiane

Nota:

- Assunzioni diverse sulla forma della matrice di covarianza portano a diverse forme delle misture
  - Sferica / Diagonale / Full
  - Diversa / uguale per ogni cluster

- La scelta ha anche un impatto sulla stima del modello (flessibilità vs accuratezza della stima)

Misture di Gaussiane

VANTAGGI:

- molto utilizzato in svariati contesti per la sua flessibilità (spesso come alternativa al K-means)
- E' una tecnica di **soft clustering**: ritorna anche la probabilità con cui un punto appartiene ai vari cluster

SVANTAGGI:

- l'inizializzazione è un problema (tuttavia è meno rilevante rispetto al caso del K-means)
- Stimare il numero di cluster è un problema
- Il metodo di clustering funziona bene se i cluster hanno una forma Gaussiana

La validazione del clustering

Definizione

- Validazione del clustering: insieme di procedure che valutano il risultato di un'analisi di clustering in modo **quantitativo** e **oggettivo**
  - ⇨ Differente dalla validazione "soggettiva": data dal particolare contesto applicativo, con l'utilizzo della conoscenza a priori sul problema (intesa anche come "interpretazione dei risultati")
  - ⇨ In questa parte: validazione "oggettiva": misura quantitativa della capacità della struttura trovata di spiegare i dati (indipendentemente dal contesto)

Indici di validità

Tipicamente si utilizzano degli indici, che possono essere diversi a seconda di cosa si va a validare

- Gerarchie: risultato degli algoritmi gerarchici
  - ⇨ Possiamo anche voler valutare una gerarchia esistente, ad esempio un modello teorico
- Partizioni: risultato degli algoritmi partizionali
  - ⇨ Si può valutare una partizione esistente derivante da informazioni di categoria
- Clusters: sottoinsiemi di patterns
  - ⇨ Derivanti da cluster analysis, informazione di categorie, …

Indici di validità

Tipi di indici:

- Esterni:
  - ⇨ misurano le performance di un clustering andando a confrontare il risultato con le etichette già note a priori
- Interni:
  - ⇨ Misurano le performance di un clustering utilizzando solo i dati (completamente non supervisionato)
- Relativi:
  - ⇨ Confronta due risultati di clustering

Indici di validità per partizioni

- Rispondono alle seguenti domande:
  - ⇨ La partizione ha un buon match con le categorie?
  - ⇨ Quanti cluster ci sono nel dataset?
  - ⇨ Dove deve essere tagliato il dendrogramma?
  - ⇨ Quale tra due partizioni date fitta meglio il dataset?

Indici di validità per partizioni

Criteri esterni:

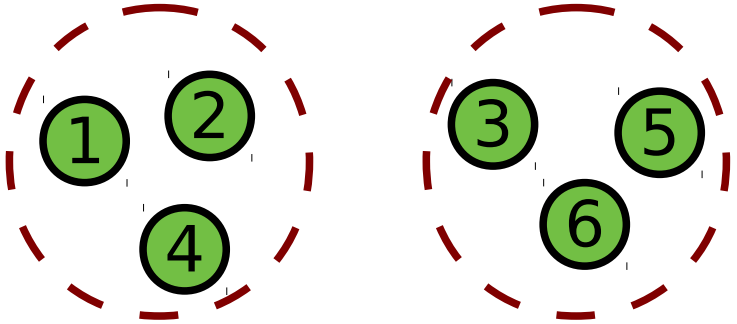
- Tipicamente si va a confrontare due partizioni:
  - ⇨ Una deriva dal clustering
  - ⇨ Una deriva dall'informazione a priori (etichette)

- Diversi indici Rand, Jaccard, Fowlkes and Mallows, Γ statistic

Indici di validità per partizioni

- Punto di partenza: una funzione indicatrice $I_U(i,j)$
  - $I_U(i,j)$ vale 1 se gli oggetti i e j sono nello stesso cluster secondo il clustering U

Partizione U



Funzione Indicatrice

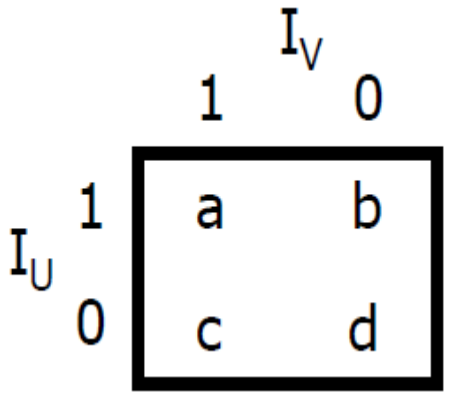
I_U	1	2	3	4	5	6
1	1	1	0	1	0	0
2	1	1	0	1	0	0
3	0	0	1	0	1	1
4	1	1	0	1	0	0
5	0	0	1	0	1	1
6	0	0	1	0	1	1

Indici di validità per partizioni

Tipicamente si hanno due partizioni U e V

- U: risultato del clustering
- V: clustering “vero” (deriva dalle etichette note a priori)

Posso calcolare la matrice di contingenza



a = numero di coppie di oggetti che sono messi nello stesso cluster in tutte e due le partizioni

b = numero di coppie di oggetti che sono messi nello stesso cluster da U ma non da V

c = numero di coppie di oggetti che sono messi nello stesso cluster da V ma non da U

d = numero di coppie di oggetti messi in cluster diversi sia da U che da V

Indici di validità per partizioni

Matematicamente

$$a = \sum_{i,j} I_U(i,j)\, I_V(i,j)$$

È uguale a 1 se sia U che V sono 1, cioè se sia U che V mettono gli oggetti xi e xj nello stesso cluster

$$b = \sum_{i,j} I_U(i,j)\,(1 - I_V(i,j))$$

È uguale a 1 se U è 1 e V è 0, quindi se U mette xi e xj nello stesso cluster ma V no

Indici di validità per partizioni

$$c = \sum_{i,j} \left(1 - I_U(i,j)\right) I_V(i,j)$$

$$d = \sum_{i,j} \left(1 - I_U(i,j)\right)\left(1 - I_V(i,j)\right)$$

Si possono anche calcolare le seguenti quantità

- m_1 = numero di coppie nello stesso gruppo in U
  - ⇨ $m_1 = a+b$
- m_2 = numero di coppie nello stesso gruppo in V
  - ⇨ $m_2 = a+c$
- M = numero totale di coppie
  - M = a+b+c+d

Indici di validità per partizioni

I diversi indici sono definiti a partire da queste quantità: l'idea generale è quella di misurare quanto vanno d'accordo le due partizioni

$$\frac{a+d}{\binom{n}{2}}$$

Indice RAND

$$\frac{a}{(a+b+c)}$$

Indice Jaccard

$$\frac{Ma-m_1 m_2}{\left[m_1 m_2 (M-m_1)(M-m_2)\right]^{1/2}}$$

Γ statistic

$$\frac{a}{(m_1 m_2)^{1/2}}$$

Fowlkes & Mallows

Indici di validità per partizioni

Criteri interni:

- Difficili da stimare: devono misurare il fitting tra una partizione data e il dataset
- Problema fondamentale: stimare il numero di clusters
- Molti metodi (esempio metodi di model selection per modelli probabilistici)
- Ma molte difficoltà:
  - ⇨ Stima della baseline (campionamento di molti dataset + stima di un indice interno --- ma quale modello per campionare i dati?)
  - ⇨ Gli indici interni dipendono strettamente dai parametri del problema:
    - ⇨ Numero di features, numero di patterns, numero di clusters
    - …

Un particolare indice

L’indice di Davies-Bouldin (1979)

- Inizialmente utilizzato per decidere quando fermare un clustering sequenziale
- L’indice viene calcolato al variare del numero di clusters
- Il miglior clustering corrisponde al valore minimo

L'indice di Davies Bouldin

DEFINIZIONI

- $\{\mathbf{x}_1, \ldots \mathbf{x}_N\}$ punti da clusterizzare
- $C_1..C_K$: partizione da valutare (insieme dei K clusters, ognuno di cardinalità n_j)

Si possono calcolare il centroide, la variazione intracluster e la variazione tra clusters

$$m_j = \frac{1}{n_j} \sum_{x_i \in C_j} x_i$$ centroide

$$e_j^2 = \sum_{x_i \in C_j} (x_i - m_j)^T (x_i - m_j)$$ within cluster variation

$$dm(j,h) = d(m_j, m_h)$$ between cluster variation

L'indice di Davies Bouldin

Passi per calcolare l'indice

- Per ogni coppia di cluster (j,h) si calcola

$$R_{jh} = \frac{e_j + e_h}{dm(j,h)}$$

- Per ogni cluster si calcola

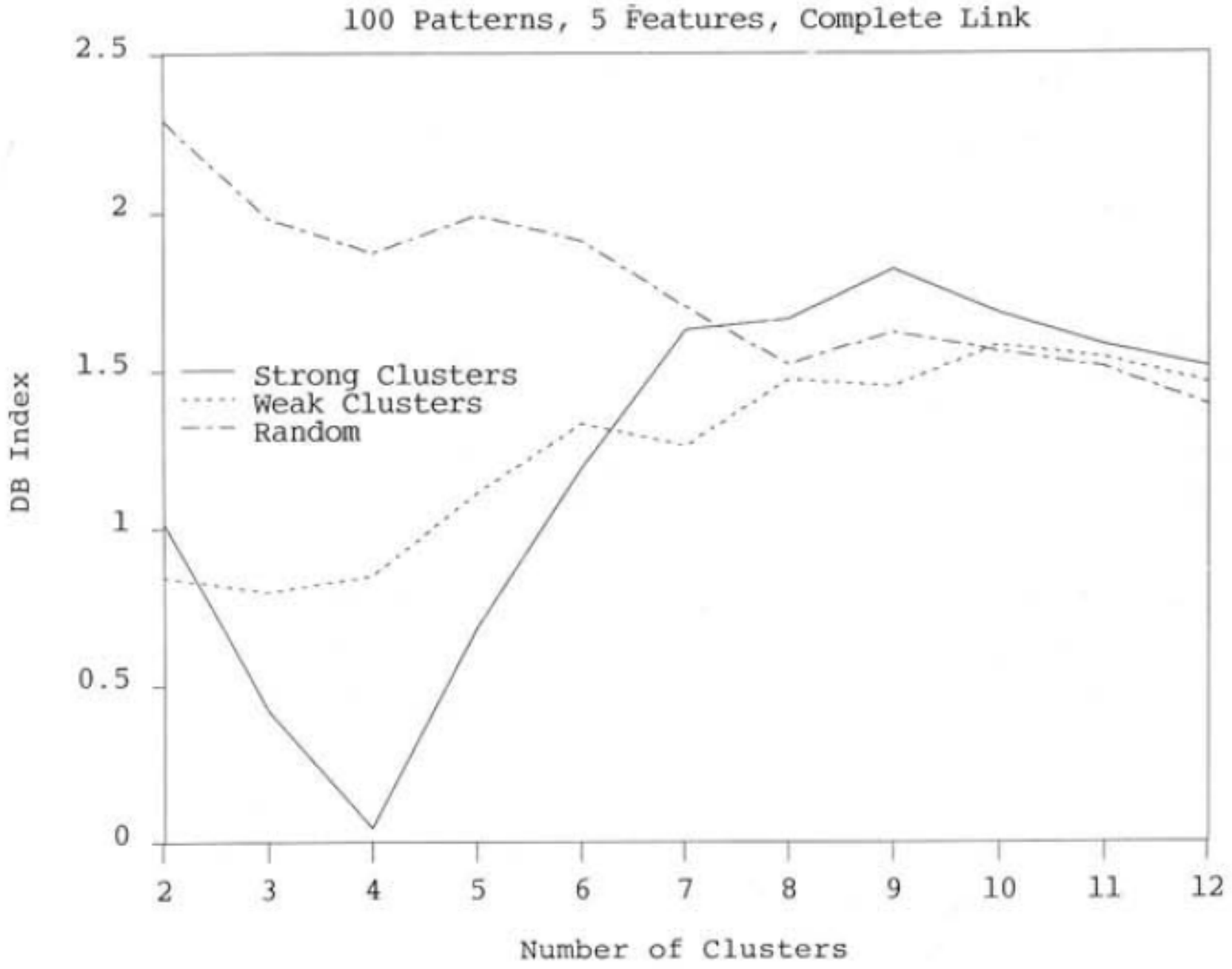
$$R_j = \max_{j \neq h} R_{jh}$$

- L'indice di Davies Bouldin viene determinato come

$$DB(\{C_{1,\ldots}, C_K\}) = \frac{1}{K} \sum_{j=1}^{K} R_j$$

Più piccolo è il valore dell'indice migliore è il clustering!

Può anche essere utilizzato per determinare la presenza di una struttura di clustering



Cluster tendency

- Problema: gli algoritmi di clustering producono sempre un output, indipendentemente dal dataset
- Definizione di cluster tendency: identificare, senza effettuare il clustering, se i dati hanno una predisposizione ad aggregarsi in gruppi naturali

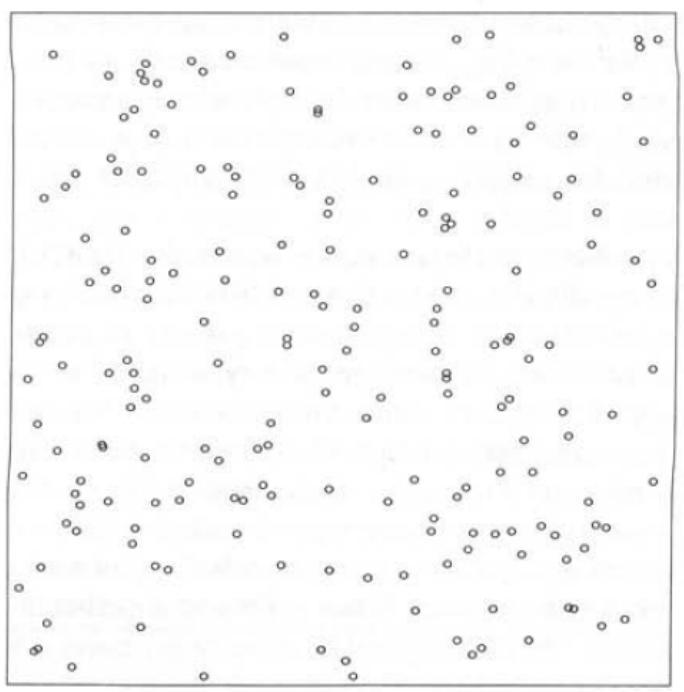
- Operazione preliminare cruciale:
  - ⇨ Previene dall'applicare elaborate metodologie di clustering e di validazione a dati in cui i cluster sono sicuramente degli artefatti degli algoritmi di clustering

Cluster tendency

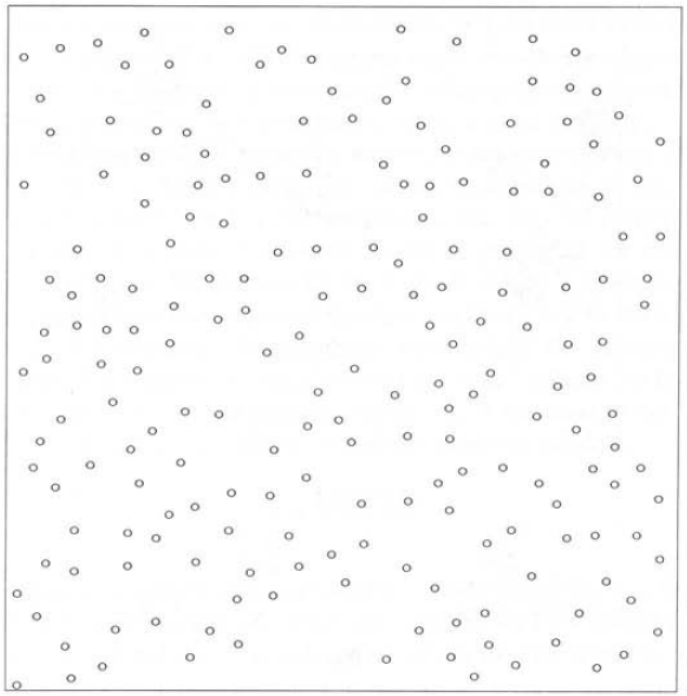
- IDEA: studio dello spazio delle features in modo da identificare tre possibili situazioni:
  1. I pattern sono sistemati in modo casuale (spatial randomness)
  2. I pattern sono aggregati, cioè esibiscono una mutua attrazione
  3. I pattern sono spaziati regolarmente, cioè esibiscono una mutua repulsione

- Nei casi 1 e 3 non ha senso effettuare il clustering

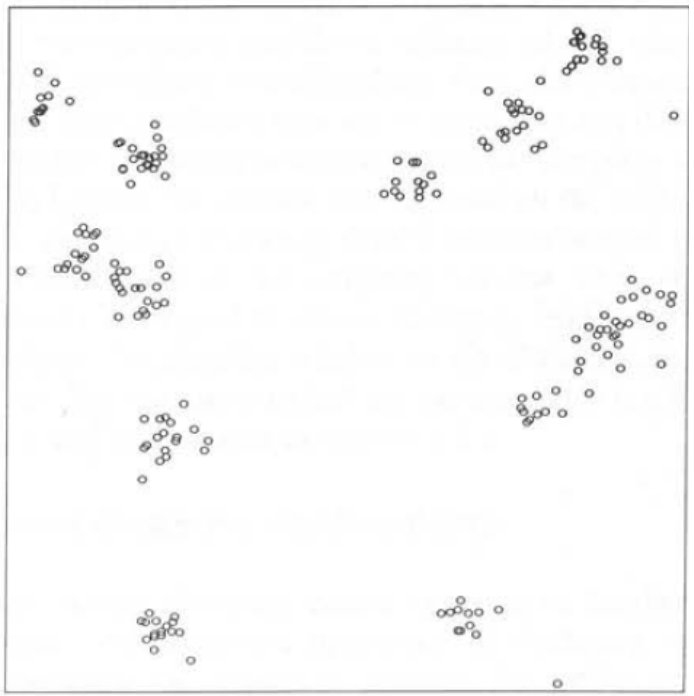
Cluster tendency



random



regular



cluster

Cluster tendency

IDEA: effettuare alcuni test in modo da determinare se esiste o meno una struttura (e.g. test per una distribuzione uniforme in una finestra detta sampling window)

ESEMPI:

- Scan tests:
  - ⇨ Contare il numero di pattern presenti nella sottoregione più popolosa
  - ⇨ Se il numero è inusualmente grande allora esiste un clustering
  - ⇨ PROBLEMI: come definire le sottoregioni, cosa vuol dire “inusualmente grande”