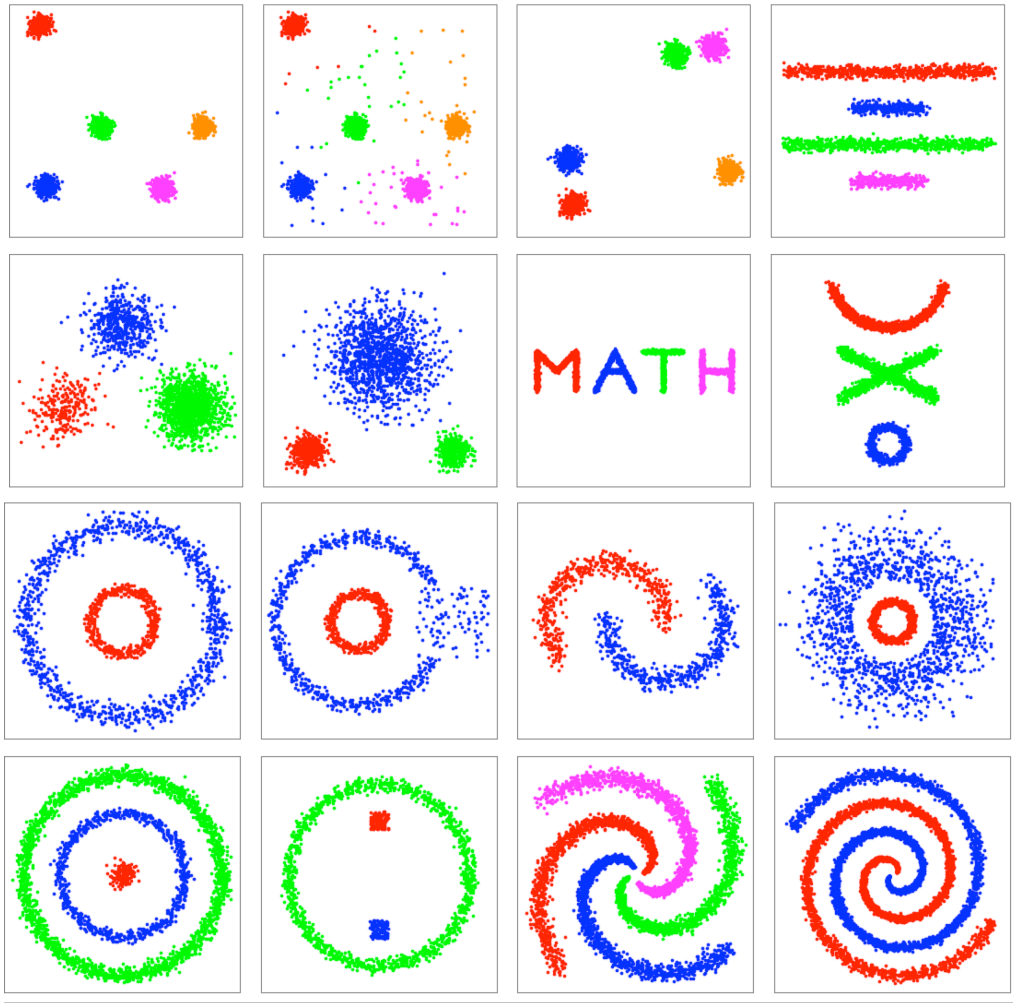
Algoritmi adattivi per il clustering convesso e non convesso basati sulla NMF simmetrica



Il problema del clustering

*Il **clustering** è un insieme di tecniche volte alla selezione e raggruppamento di elementi omogenei in un insieme di dati. Le tecniche di clustering si basano su misure di somiglianza tra gli elementi. In molti approcci questa similarità è concepita in termini di distanza in uno spazio multidimensionale. Gli algoritmi di clustering raggruppano gli elementi sulla base della loro distanza reciproca, e quindi l'appartenenza o meno a un insieme dipende da quanto l'elemento preso in esame è distante dall'insieme stesso.*

In questo lavoro ci restringiamo a insiemi in $\Re^m$ ($\Re^2$ negli esempi).

La strada che seguiamo è quella della **Fattorizzazione Non negativa di Matrici** (**NMF**)

Il problema della NMF

Sia $\Re_+^m$ lo spazio dei vettori a m dimensioni con componenti non negative e X una matrice di n colonne

$$x_i \in \Re_+^m, \quad i = 1,2,...n.$$

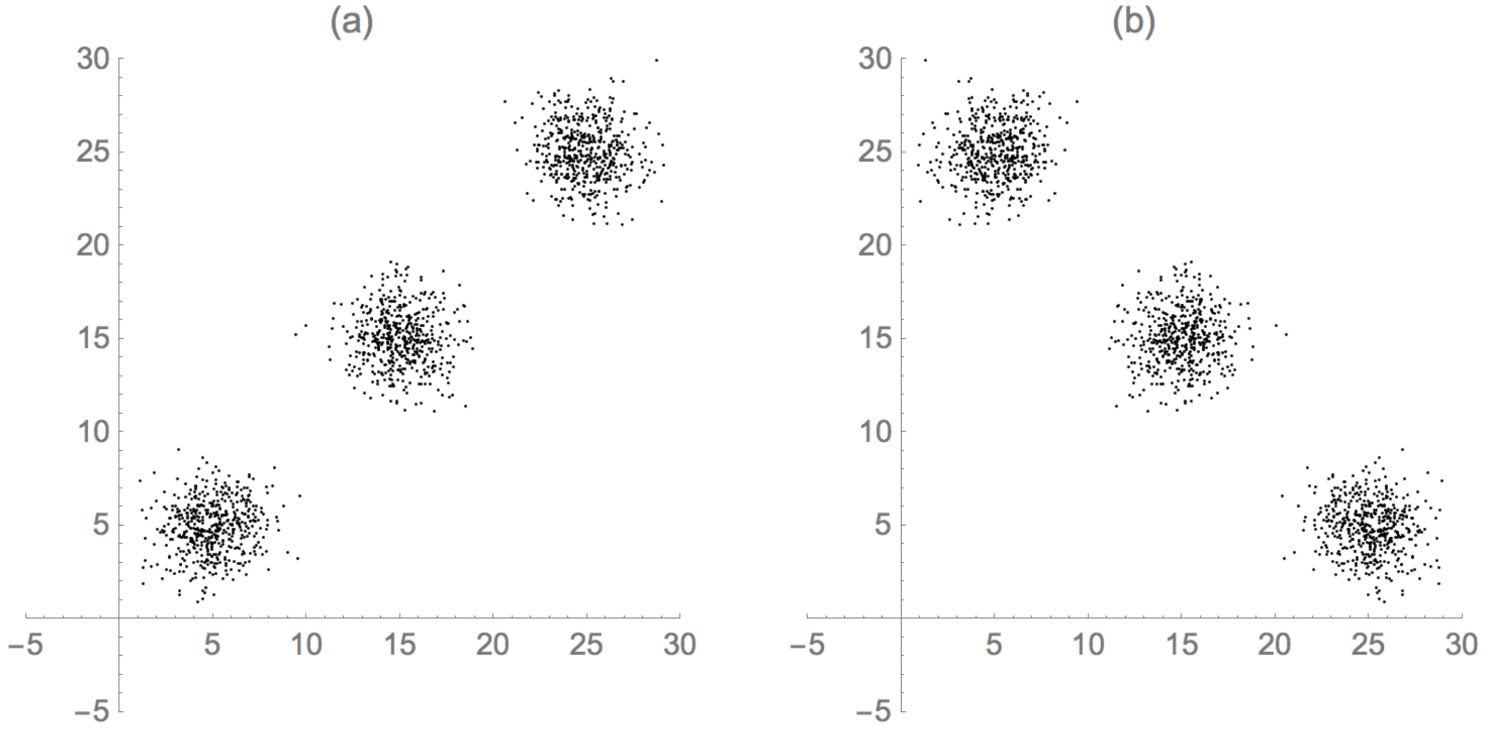
Sia $k \ll n$, ricerchiamo due matrici

$$C \in \Re_+^{m \times k}, \quad H \in \Re_+^{n \times k},$$

tali che il prodotto CH^T approssimi X, ovvero si cerca di risolvere un problema con vincoli di non negatività

$$\min_{C,H \geq O} \left\| X - CH^T \right\|_F^2 \tag{1}$$

Questo approccio permette già di affrontare il problema del clustering ma i risultati dipendono pesantemente dalla distribuzione dei punti di X. Per esempio si dimostra che per l'insieme (a) si ottengono risultati peggiori rispetto all'insieme (b) e questo è chiaramente inaccettabile.



Nota Bene: se i punti di X non sono vettori ma altre "cose" (per esempio documenti di testo) questo approccio è semplicemente IMPOSSIBILE

È possibile invece usare una matrice di **similarità** A, **simmetrica** e **non negativa**, legata a X attraverso una nozione di **distanza** e cercarne una fattorizzazione simmetrica:

$$\min_{W \in \Re_+^{n\times k}} \left\| A - WW^T \right\|_F^2$$

La matrice di similarità

Nel caso di clustering su $\Re^m$ è molto usata una matrice di similarità basata su un Kernel gaussiano.

$$e_{ij} = \exp\left(- \frac{\left\| x_i - x_j \right\|^2}{\mu\sigma} \right), \quad \mu = \max_{ij} \left\| x_i - x_j \right\|^2,$$

$$a_{ij} = d_i^{-1/2} e_{ij}\, d_j^{-1/2}, \quad d_i = \sum_{r=1}^{n} e_{ij}.$$

Si noti che A è invariante rispetto a roto-traslazioni e variazioni di scala.

La scelta del parametro σ ha importanza fondamentale ai fini del clustering:

- σ molto grande tende a produrre una matrice piena di rango 1,
- σ molto piccolo tende a produrre una matrice diagonale,
- in genere più grande è σ minore è il numero dei cluster che l'algoritmo tende a trovare.

Quest'approccio vale per qualunque tipo di raccolte di "oggetti" semplicemente usando la corretta misura di "similarità".

Algoritmi per la NMF

Il problema (1), ovvero minimizzare con vincoli di non negatività la funzione di quarto grado

$$\left\|X - CH^T\right\|_F^2$$

non è convesso e la funzione obiettivo presenta numerosi (possibilmente infiniti) minimi locali.

Invece i problemi locali

$$C = \underset{C \geq O}{\arg\min} \left\|X - CH^T\right\|_F^2, \quad H = \underset{H \geq O}{\arg\min} \left\|X - CH^T\right\|_F^2$$

sono convessi e possono essere affrontati con algoritmi ben noti.

L'algoritmo **ANLS** (*Alternating Nonnegative Least Squares*), applicato alla NMF da [Kim, Park (2014)], prende in input una matrice iniziale $C^{(0)}$ e calcola una sequenza di matrici

$$H^{(0)}, \quad C^{(1)}, H^{(1)}, \quad C^{(2)}, H^{(2)} \quad ..., \quad C^{(l)}, H^{(l)} ...$$

risolvendo alternativamente in C e H i problemi locali fino a che non è soddisfatta una qualche condizione di terminazione.

Algoritmi per i problemi locali

Il problema

$$H = \arg\min \left\| X - CH^T \right\|_F^2 \quad H \geq O$$

può essere scritto come

$$C^T C H^T = C^T X, \quad H \geq O$$

e visto come *n* sistemi lineari indipendenti con *k* equazioni e *k* incognite con il vincolo della soluzione positiva.

Si possono usare per esempio:

- un metodo del tipo *Active-Set*, [vedi *Algorithm* 2 in Kim, Park (2011)],
- il metodo *Greedy Coordinate Descent*, [Hsieh, Dillon (2011)] che ha dato risultati nettamente superiori (tempi dimezzati a parità di precisione) e che è stato adottato nei calcoli successivi.

Algoritmi per la NMF Simmetrica

Il problema simmetrico

$$\min_{W \geq O} \left\| A - WW^T \right\|_F^2$$

può essere risolto con una tecnica di penalizzazione

$$\min_{W,H \geq O} \left\| A - WH^T \right\|_F^2 + \lambda \left\| W - H \right\|_F^2,$$

che a sua volta si può affrontare con gli algoritmi per la NMF non simmetrica, scegliendo una matrice iniziale $W^{(0)}$ e risolvendo i problemi locali

$$H^{(\nu)} = \arg\min_{H^{(\nu)} \geq O} \left\| \begin{bmatrix} A \\ \sqrt{\lambda} W^{(\nu-1)T} \end{bmatrix} - \begin{bmatrix} W^{(\nu-1)} \\ \sqrt{\lambda} I_k \end{bmatrix} H^T \right\|_F^2,$$

$$W^{(\nu)} = \arg\min_{W^{(\nu)} \geq O} \left\| \begin{bmatrix} A \\ \sqrt{\lambda} H^{(\nu)T} \end{bmatrix} - \begin{bmatrix} H^{(\nu)} \\ \sqrt{\lambda} I_k \end{bmatrix} W^T \right\|_F^2.$$

Si noti che tali formule possono essere calcolate senza costruire esplicitamente le matrici a blocchi.

La scelta di λ

La tecnica di penalizzazione richiede la scelta del parametro λ:

- un valore di λ troppo grande fa tendere al minimo locale banale $H = W = \mathrm{O}$,
- un valore di λ troppo piccolo rende difficile la convergenza ad un minimo locale simmetrico.

Dato che le soluzioni del problema simmetrico sono minimi locali anche per il problema penalizzato, nella vicinanza di un minimo locale simmetrico si ha convergenza anche per λ piccolo.

Questo suggerisce una strategia adattiva per il valore di λ. Definiamo le quantità

$$es^{(\nu)} = \left\|A - WW^T\right\|_F^2 / \|A\|_F^2, \quad ens^{(\nu)} = \left\|A - WH^T\right\|_F^2 / \|A\|_F^2, \quad \delta^{(\nu)} = \|W - H\|_F^2 / \|A\|_F^2,$$

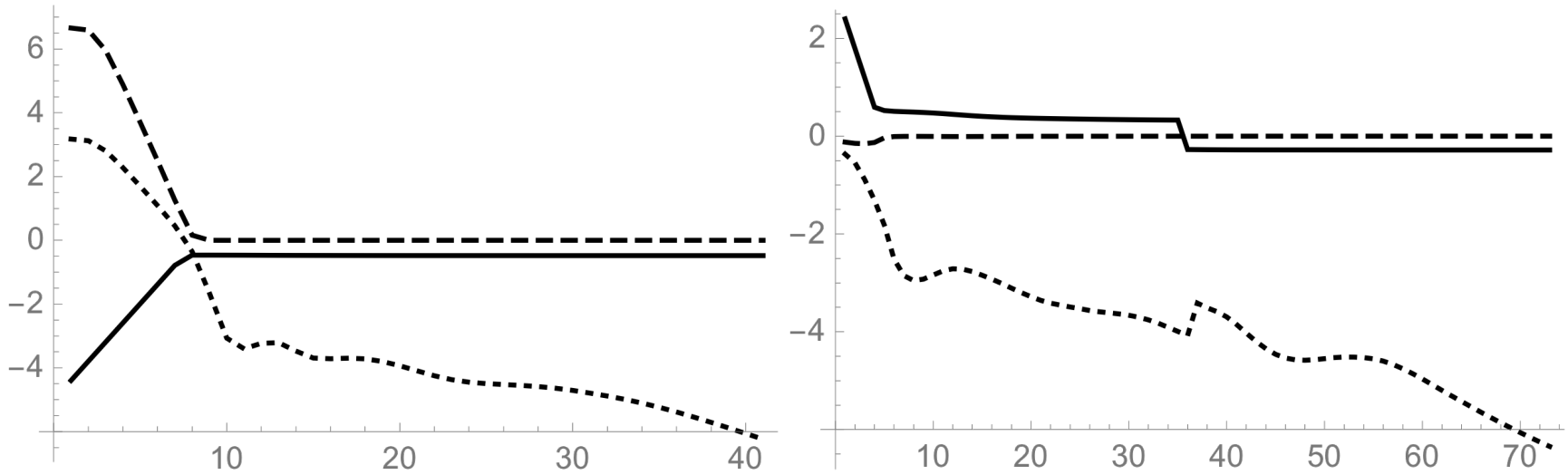
che misurano rispettivamente l'errore relativo del problema simmetrico, l'errore relativo del problema non simmetrico e il grado di simmetria raggiunto dal metodo di penalizzazione, alla ν-esima iterazione.

Si noti che la sostituzione di H con W può essere vista come una sorta di estrapolazione che **può** migliorare l'approssimazione nelle vicinanze di un minimo locale simmetrico.

- migliora l'approssimazione nelle vicinanze di un minimo locale simmetrico,
- peggiora l'approssimazione se invece si è lontani da una soluzione simmetrica.

L'algoritmo adattivo prevede di aumentare o diminuire il valore di λ a seconda del valore del rapporto $es^{(\nu)} / ens^{(\nu)}$. Gli esperimenti numerici mostrano che con questa tecnica, partendo da valori di λ di vari ordini di grandezza si raggiunge comunque un minimo locale del problema simmetrico.

Nella figura sono mostrati due esempi di convergenza.



Ascissa: numero di iterazioni. Ordinata (in scala logaritmica): λ tratto continuo, $es^{(v)}$ / $ens^{(v)}$ tratteggiato, $\delta^{(v)}$ punteggiato.

Nella pratica si è scelto di partire con λ = 1.

Estrazione del cluster

Una volta trovata una matrice W tale che WW^T approssimi "bene" A, bisogna ricavarne il clustering associato.

Vediamo dapprima un caso limite ideale (cluster infinitamente compatti e infinitamente lontani tra loro). Se A è diagonale a blocchi con blocchi di valore costante (per esempio tutti 1) esiste una fattorizzazione esatta con k uguale al numero dei blocchi.

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad \mathbf{W} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

Si nota che le colonne di W rappresentano l'appartenenza dei punti ai cluster.

In pratica, nel caso reale, si normalizzano le righe di W usando la norma infinito e si pongono a 0 gli elementi minori di 1 ottenendo ancora una matrice B a valori in $\{0, 1\}$.

Teoricamente si possono verificare i seguenti casi:

1. La presenza di una riga di *W* con elementi numericamente nulli significa che il punto corrispondente non può essere assegnato ad alcun cluster (evento molto raro).

2. La presenza di una riga di *B* con più valori non nulli significa che quel punto può essere assegnato a più cluster (evento raro).

3. L'assenza di colonne di *B* hanno almeno un elemento non nullo significa che l'algoritmo ha prodotto $kf = k$ cluster (evento frequente).

4. La presenza di colonne di *B* con tutti elementi nulli significa che l'algoritmo ha prodotto *kf* cluster con $kf < k$ (evento frequente).

Dopo aver trattato a parte i casi 1 e 2 i cluster possono essere ricavati come

$$\Pi = \left\{\pi_1, \pi_2, \ldots \pi_{kf}\right\}, \quad \pi_i = \{j : w_{ji} = 1\}.$$

Le misure di valutazione dei risultati

In letteratura sono presenti decine di misure per valutare la "bontà" di un clustering. In genere si richiede che i vari cluster siano compatti e ben separati e una "buona" misura è un compromesso tra le due proprietà.

Per esempio il DB-Index [Davis, Bouldin (1979)] è definito come segue. Siano

$$c_i = mean\{x_j : j \in \pi_i\}, \quad i = 1,2,...,kf$$

i centroidi del clustering. I valori

$$\gamma_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \left\| x_j - c_j \right\|_2^2, \quad n_i = \#\pi_i, \quad i = 1,2,...,kf$$

misurano la compattezza dei singoli cluster. Supponendo $kf > 1$, l'indice globale è dato da:

$$DB(\Pi) = \frac{1}{kf} \sum_{i=1}^{kf} \max_{r \neq i} \frac{\gamma_r + \gamma_i}{\left\| c_r - c_i \right\|_2^2}$$

La scelta dei parametri iniziali σ e $W^{(0)}$

La scelta dei parametri iniziali σ e $W^{(0)}$ è **molto critica**,

- il valore di $W^{(0)}$ determina il minimo locale a cui si converge,
- il valore di σ influenza il numero di cluster che l'algoritmo tende a trovare.

L'approccio seguito è stato quello di provare numerose istanze del problema con valori diversi, portando avanti il calcolo di quelle più promettenti e scartando le altre in base al valore del DB-index.

Si noti che l'obiettivo della elaborazione non è tanto trovare un minimo globale per la funzione obiettivo quanto di proporre uno o più clustering con un buon valore del DB-index

L'algoritmo è organizzato come segue.

Algoritmo euristico per la ricerca del cluster “migliore”

- Si genera un numero elevato (qualche decina) di istanze del problema con valori diversi di σ e $W^{(0)}$ e si pongono in una coda a priorità con associato il valore 0.

- Per ogni elemento della coda (scelto in base al DB-index più basso) si effettuano 10 iterazioni, si calcola il clustering associato e l’indice DB e se l’elemento risulta attivo lo si rimette in coda.

- Un elemento è considerato non attivo ed eliminato se vale almeno una delle seguenti condizioni:

  - ha raggiunto la terminazione numerica,
  - per due volte si è ottenuto lo stesso clustering,
  - il valore del DB-index è troppo alto rispetto agli altri già calcolati.

- La computazione termina se vale almeno una delle seguenti condizioni:

  - la coda è vuota,
  - si continuano ad ottenere valori “buoni” (uguali al minimo),
  - si continuano ad ottenere valori “cattivi” (maggiori del minimo).

- Alla fine del calcolo si restituisce l’elemento con il miglior valore del DB-index.

Si noti che gli elementi della coda possono essere trattati indipendentemente e di conseguenza

l’algoritmo è facilmente parallelizzabile

La scelta della dimensione *k*

Il parametro fondamentale ancora da trattare è la dimensione *k* con cui si imposta il problema di minimo.

Sono possibili varie alternative:

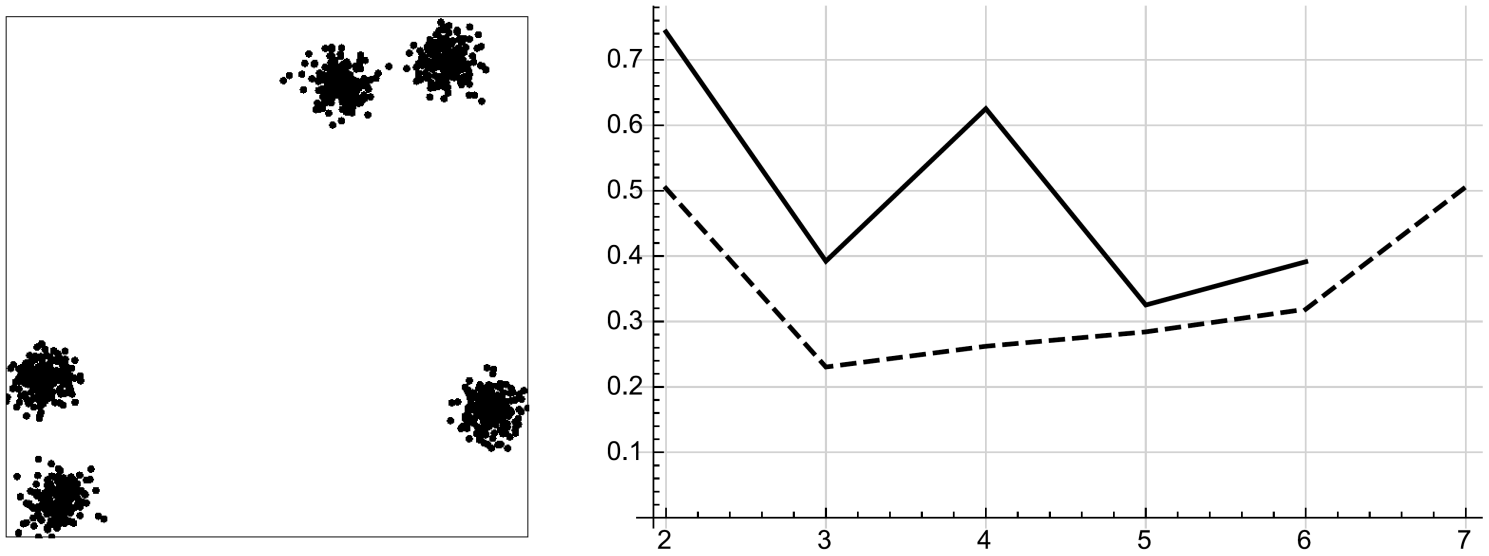
- si possono impostare uno o più valori di *k* lasciando il programma libero di trovare soluzioni con $kf < k$,
- si può imporre che la soluzione abbia $kf = k$ se questo è richiesto dal problema a monte del clustering,
- si può usare l'algoritmo più volte con diversi valori di *k* ottenendo un insieme di soluzioni con diverso numero di cluster.

In quest'ultimo caso, adottando una misura più sofisticata (l'indice DB**), si può ottenere un clustering totalmente adattivo.

L'indice DB**

L'indice DB** [Kim, Ramakrishma (2005)] prende in considerazione un insieme di clustering con valori di *k* consecutivi e tenta di stimare il migliore di questi.

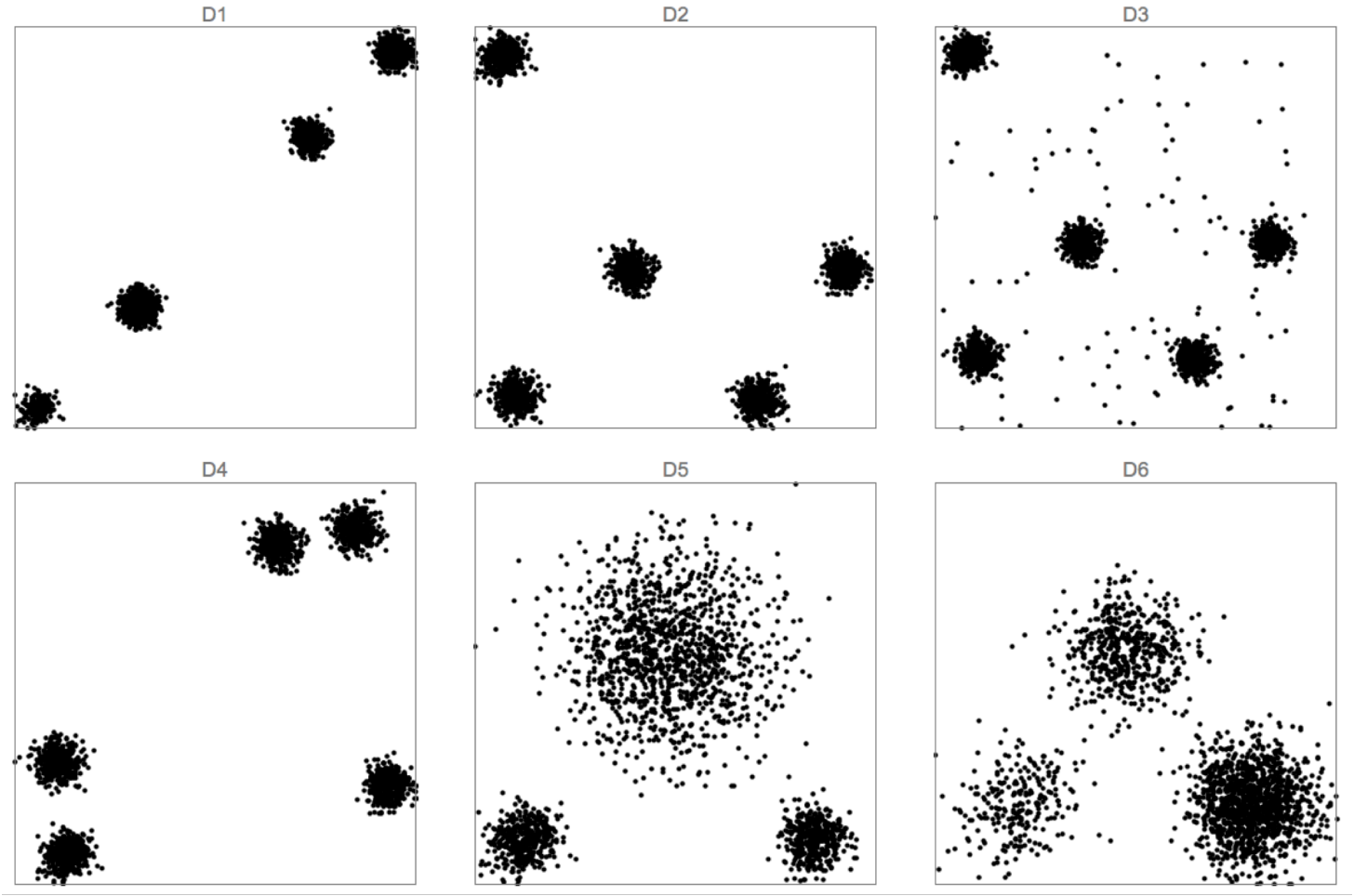
Piuttosto che presentarne la complicata definizione, vediamo un esempio di funzionamento.



Sono presenti due clustering "naturali" con *k*=3 e *k*=5, l'indice DB (linea tratteggiata) non se ne accorge mentre l'indice DB** (linea continua) segnala correttamente la cosa con due minimi locali.

Risultati della sperimentazione: *k* fisso

Sono state provate 6 configurazioni con numero di punti che varia da 1000 a 16000 con i clustering “naturali” generati con distribuzione gaussiana e il valore del DB-index è stato confrontato con quello del generatore.



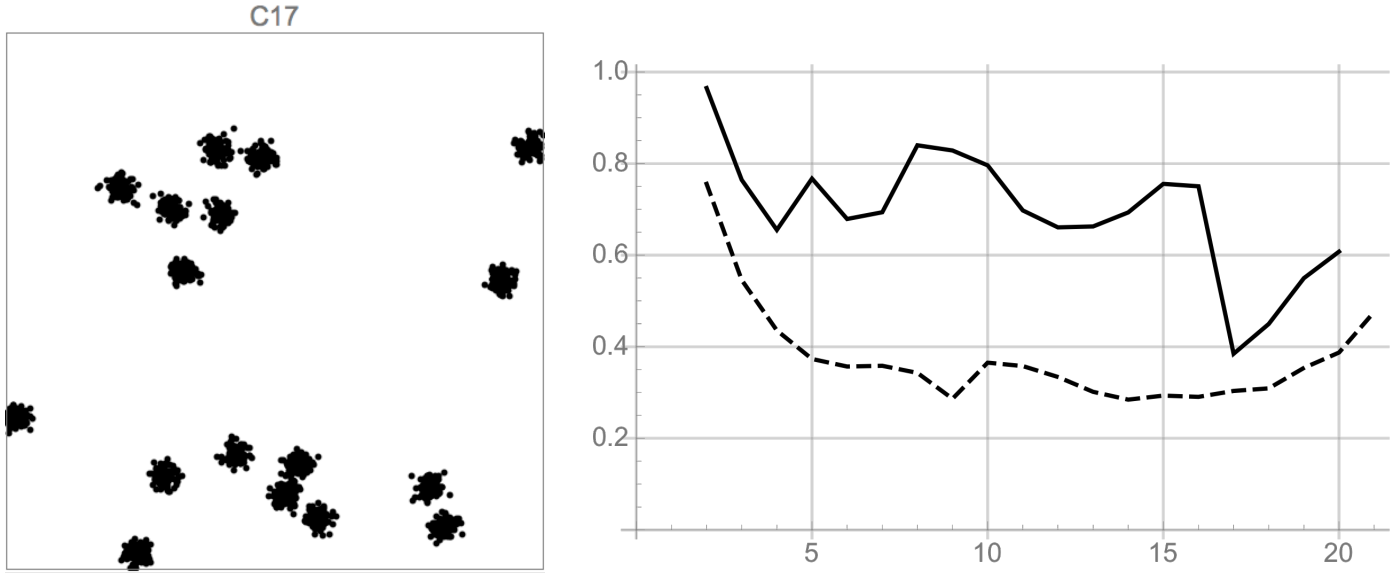
I risultati della prove sono evidenziati nella seguente tabella.

Problem	1000	2000	4000	8000	16000
D1	*	*	*	*	*
D2	*	*	*	*	*
D3	**	*	**	**	**
D4	*	*	*	*	**
D5	**	**	**	**	**
D6	*	**	—	—	**

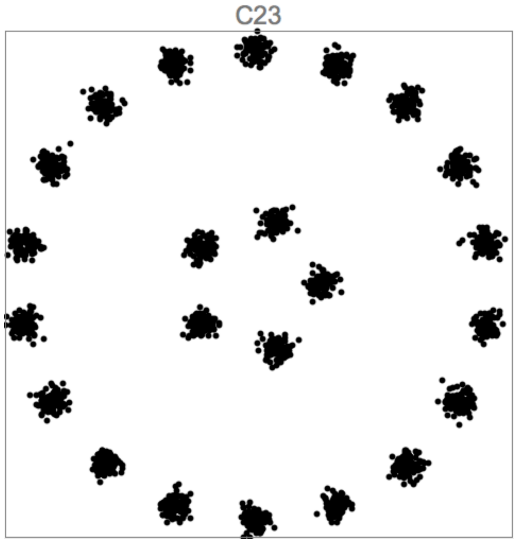
* indica che il DB-index è uguale a quello di generazione,
** indica che il DB-index è minore di quello di generazione,
— indica che il DB-index è maggiore di quello di generazione.

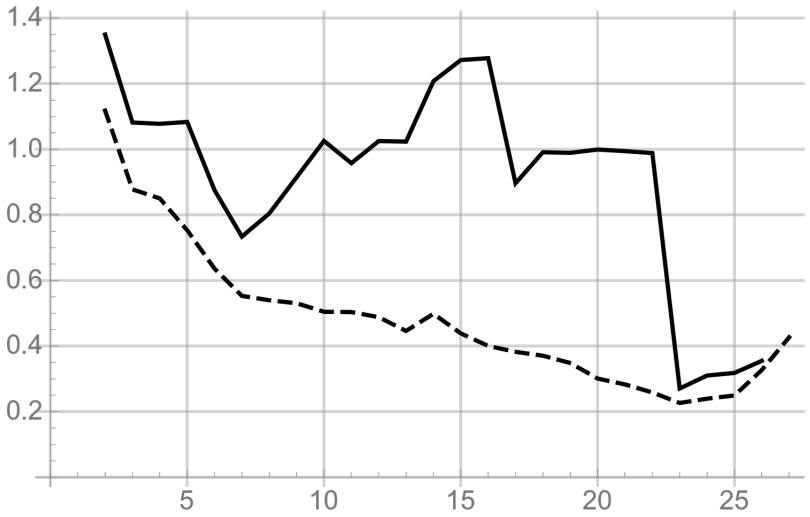
Risultati della sperimentazione: *k* variabile

Nel caso di *k* variabile sono state scelte figure per cui il clustering "naturale" non è immediatamente evidente:

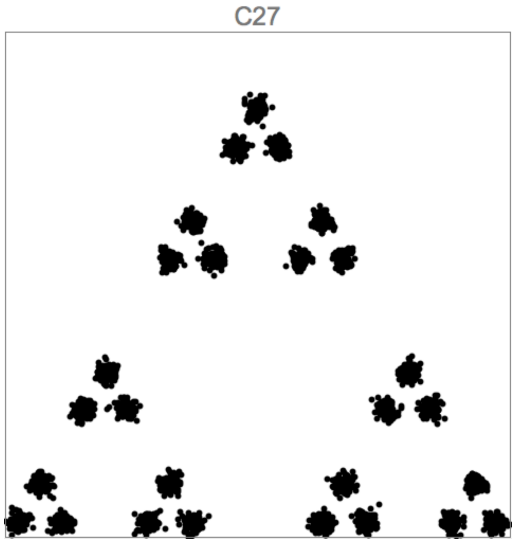


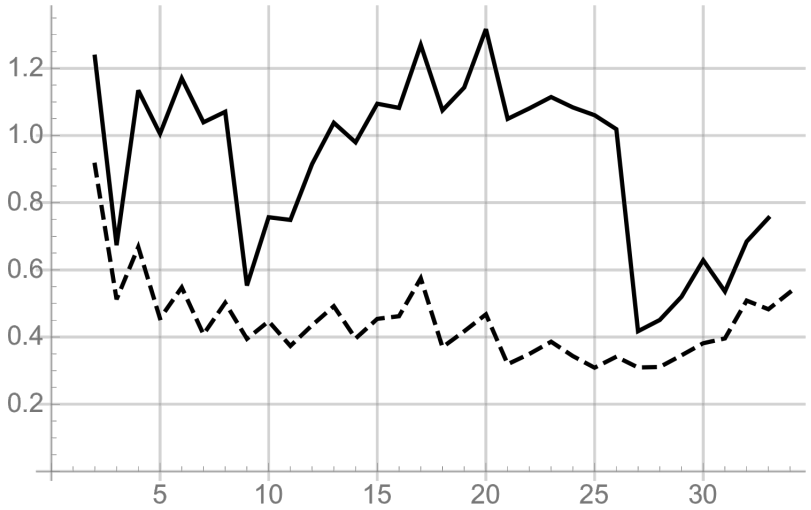
Indice DB (linea tratteggiata), l'indice DB** (linea continua) segnala tre clustering (*k*=4, *k*=6 e *k*=17)





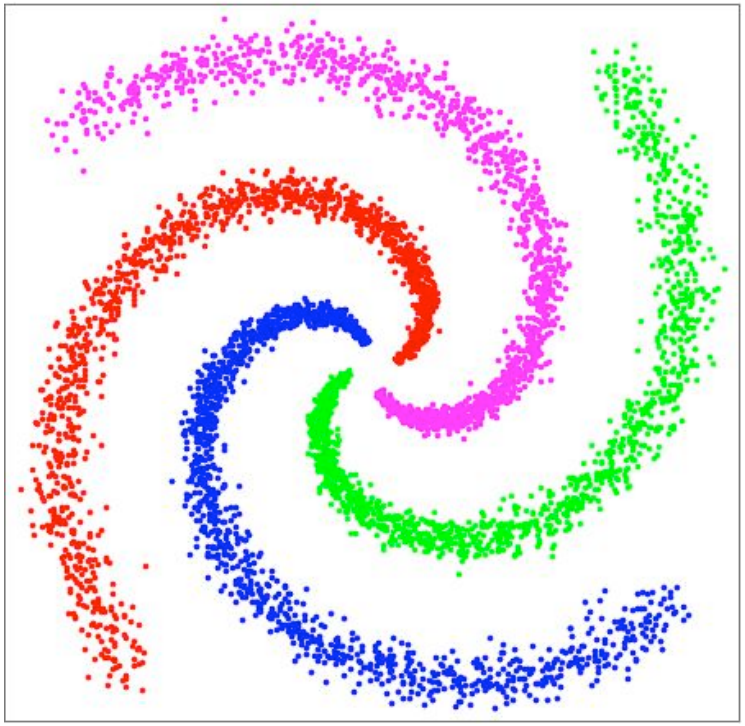
Indice DB (linea tratteggiata), l'indice DB** (linea continua) segnala 4 clustering (k=7, k=11, k=17 e k=23)





Indice DB (linea tratteggiata), l'indice DB** (linea continua) segnala tre clustering (k=9, k=9 e k=27)

Il caso non convesso



È evidente che una figura come questa non può essere trattata con le tecniche viste sopra. L'approccio nel caso di cluster di forma arbitraria è il seguente.

- Si effettua un clustering convesso con valori di *k* elevati e con un indice di valutazione che tiene conto solo della separazione
- Si mettono da parte tutte le soluzioni "ragionevoli" con il *k* cercato.
- Si effettua una fase di merging dei cluster.
- Si confrontano i risultati scegliendone uno con una tecnica di votazione.
- In caso di incertezza si raddoppia il *k* e si riprova.

Algoritmo di merging

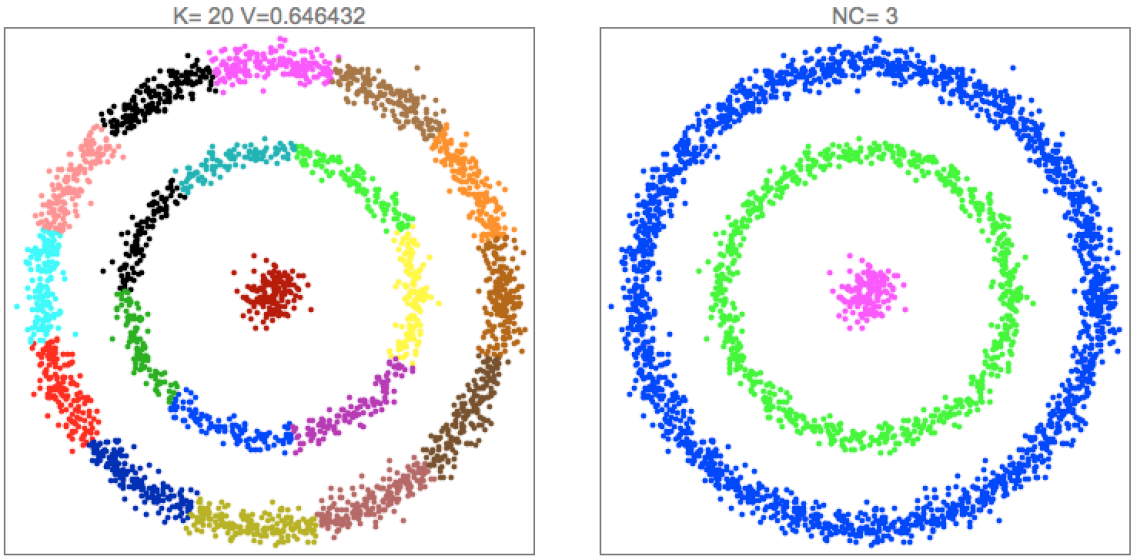
- Si scelgono due costanti α e β (nei nostri esempi $\alpha = 0.8$, $\beta = 0.0002$),
- si normalizzano le righe di W in norma infinito,
- si costruisce la matrice B nel modo seguente

$$B = \{b_{ij}\}, \quad b_{ij} = \begin{cases} 1 & \text{se } w_{ij} > \alpha \\ 0 & \text{altrimenti} \end{cases}$$

- si calcola la matrice quadrata $T = B^{\mathrm{T}}B$ (di dimensione k), t_{ij} conta il numero di punti attribuibili ai due cluster π_i e π_j in base alla soglia α
- i due cluster π_i e π_j vengono fusi se

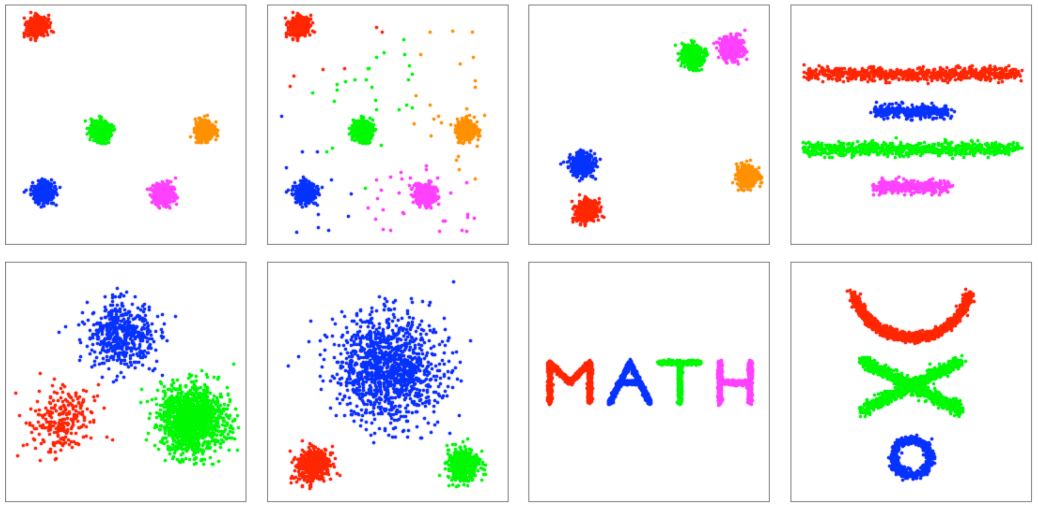
$$t_{ij} > \frac{\beta}{t_{ii}\, t_{jj}}$$

Esempio

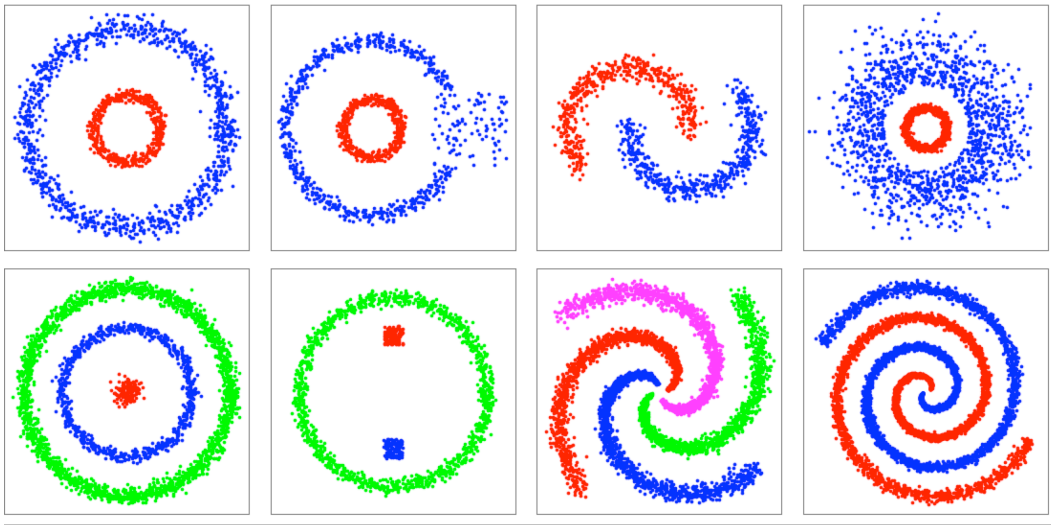


Risultati della sperimentazione

Sono stati provati sia cluster con inviluppi convessi separabili



che cluster con inviluppi non separabili



ottenendo in ogni caso la soluzione corretta.

Bibliografia

- **Favati, Lotti, Menchi, Romani (2015). Adaptive symmetric NMF for graph clustering (submitted to Numerical Linear Algebra with Applications).**
- **Favati, Lotti, Menchi, Romani (2016). Arbitrary shape clustering via NMF factorization (submitted to SIMAX).**

- Davis, Bouldin (1979). A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI 1979; 1: 224-227.
- Kim, Park (2011) Fast nonnegative matrix factorization: an active-set-like method and comparisons. SIAM J. on Scientific Computing 2011; 33: 3261-3281
- Kim, Park (2014) Algorithms for nonnegative matrix and tensor factorization: an unified view based on block coordinate descent framework. J. Glob Optim 2014; 58: 285-319.
- Kim, Ramakrishma (2005) New indices for cluster validity assessment. Pattern Recognition Lett. 2005; 26: 2353-2363.
- Hsieh, Dillon (2011) Fast coordinate descent methods with variable selection for non-negative matrix factorization, Proceedings of the 17th ACM SIGKDD, 2011; 1064-1072.