Cluster Analysis
Laboratorio di Data Science in Economia
CLEBA

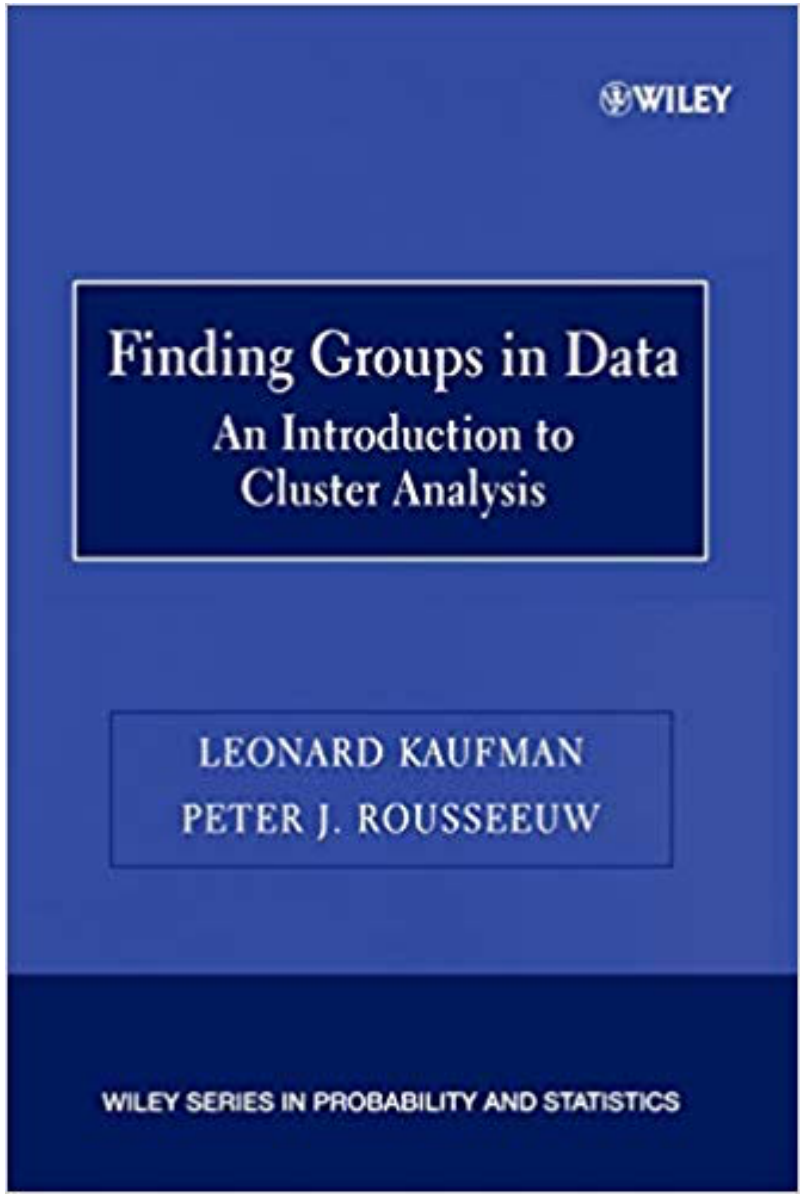
Roberto Benedetti

Dipartimento di Economia, email benedett@unich.it

Argomanti trattati

- Cluster Analysis: Concetti di Base
- Metodi scissori
- Metodi gerarchici
- Metodi basati su modelli delle densità
- Numero di Gruppi
- Valutazione dei gruppi
- Applicazioni

Riferimento principale



Introduction (Pages: 1-67)
Partitioning Around Medoids
(Program PAM) (Pages: 68-125)
Clustering Large Applications
(Program CLARA) (Pages: 126-163)
Fuzzy Analysis
(Program FANNY) (Pages: 164-198)
Agglomerative Nesting
(Program AGNES) (Pages: 199-252)
Divisive Analysis
(Program DIANA) (Pages: 253-279)
Monothetic Analysis
(Program MONA) (Pages: 280-311)
Appendix (Pages: 312-319)
References (Pages: 320-331)

Introduzione

- Cluster: una raccolta di oggetti (unità statistiche) simili (o correlati) tra loro all'interno dello stesso gruppo e dissimili (o non correlati) agli oggetti in altri gruppi
- Analisi del cluster (o clustering, segmentazione dei dati, ...)
- Trovare somiglianze tra i dati in base alle caratteristiche rilevate nei dati e raggruppare oggetti simili in cluster
- Apprendimento non supervisionato: non ci sono classi predefinite (ad es. Apprendimento per osservazioni vs. apprendimento per esempi: supervisione)
- Applicazioni tipiche
- Come strumento autonomo per ottenere informazioni dettagliate sulla distribuzione dei dati
- Come fase di pre-elaborazione per altri algoritmi

Ambiti di applicazione

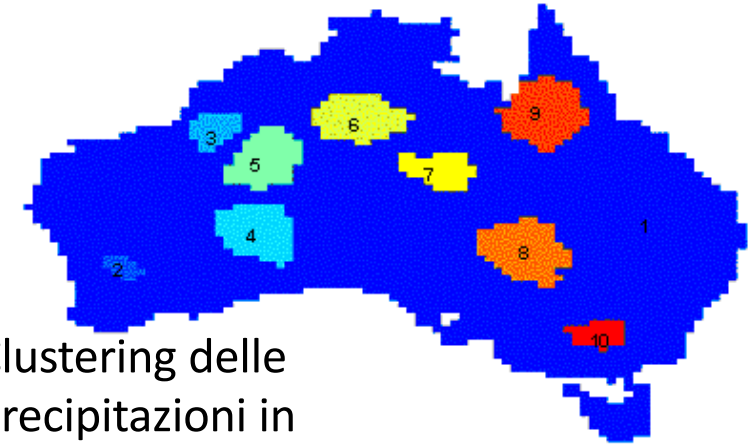
- Biologia: tassonomia degli esseri viventi: classe, ordine, famiglia, genere e specie
- Recupero di informazioni: raggruppamento di documenti
- Uso del suolo: identificazione di aree di utilizzo del suolo simile in un database di osservazione della terra
- Marketing: aiuta gli esperti di marketing a individuare gruppi distinti nelle loro basi-dati di clienti e quindi a utilizzare queste conoscenze per sviluppare programmi di marketing mirati
- Pianificazione urbana: identificazione di gruppi di case in base al tipo di casa, al valore e alla posizione geografica
- Studi sui terremoti: gli epicentri di terremoti osservati dovrebbero essere raggruppati lungo faglie continentali
- Clima: capire il clima terrestre, trovare modelli atmosferici e oceanici
- Scienza economica: ricerche di mercato

Ambiti di applicazione

- **Inferenza**
  - Cluster di documenti da ricerche web
  - Gruppi di geni e proteine che hanno funzioni simili,
  - Gruppi di azioni con fluttuazioni di prezzo simili

	Discovered Clusters	*Industry Group*
1	Applied-Matl-DOWN,Bay-Network-Down,3-COM-DOWN, Cabletron-Sys-DOWN,CISCO-DOWN,HP-DOWN, DSC-Comm-DOWN,INTEL-DOWN,LSI-Logic-DOWN, Micron-Tech-DOWN,Texas-Inst-Down,Tellabs-Inc-Down, Natl-Semiconduct-DOWN,Oracl-DOWN,SGI-DOWN, Sun-DOWN	Technology1-DOWN
2	Apple-Comp-DOWN,Autodesk-DOWN,DEC-DOWN, ADV-Micro-Device-DOWN,Andrew-Corp-DOWN, Computer-Assoc-DOWN,Circuit-City-DOWN, Compaq-DOWN, EMC-Corp-DOWN, Gen-Inst-DOWN, Motorola-DOWN,Microsoft-DOWN,Scientific-Atl-DOWN	Technology2-DOWN
3	Fannie-Mae-DOWN,Fed-Home-Loan-DOWN, MBNA-Corp-DOWN,Morgan-Stanley-DOWN	Financial-DOWN
4	Baker-Hughes-UP,Dresser-Inds-UP,Halliburton-HLD-UP, Louisiana-Land-UP,Phillips-Petro-UP,Unocal-UP, Schlumberger-UP	Oil-UP

- **Sintesi**
  - Ridurre la dimensione di dataset eccessivamente grandi



Clustering delle precipitazioni in Australia

Soluzioni metodologiche

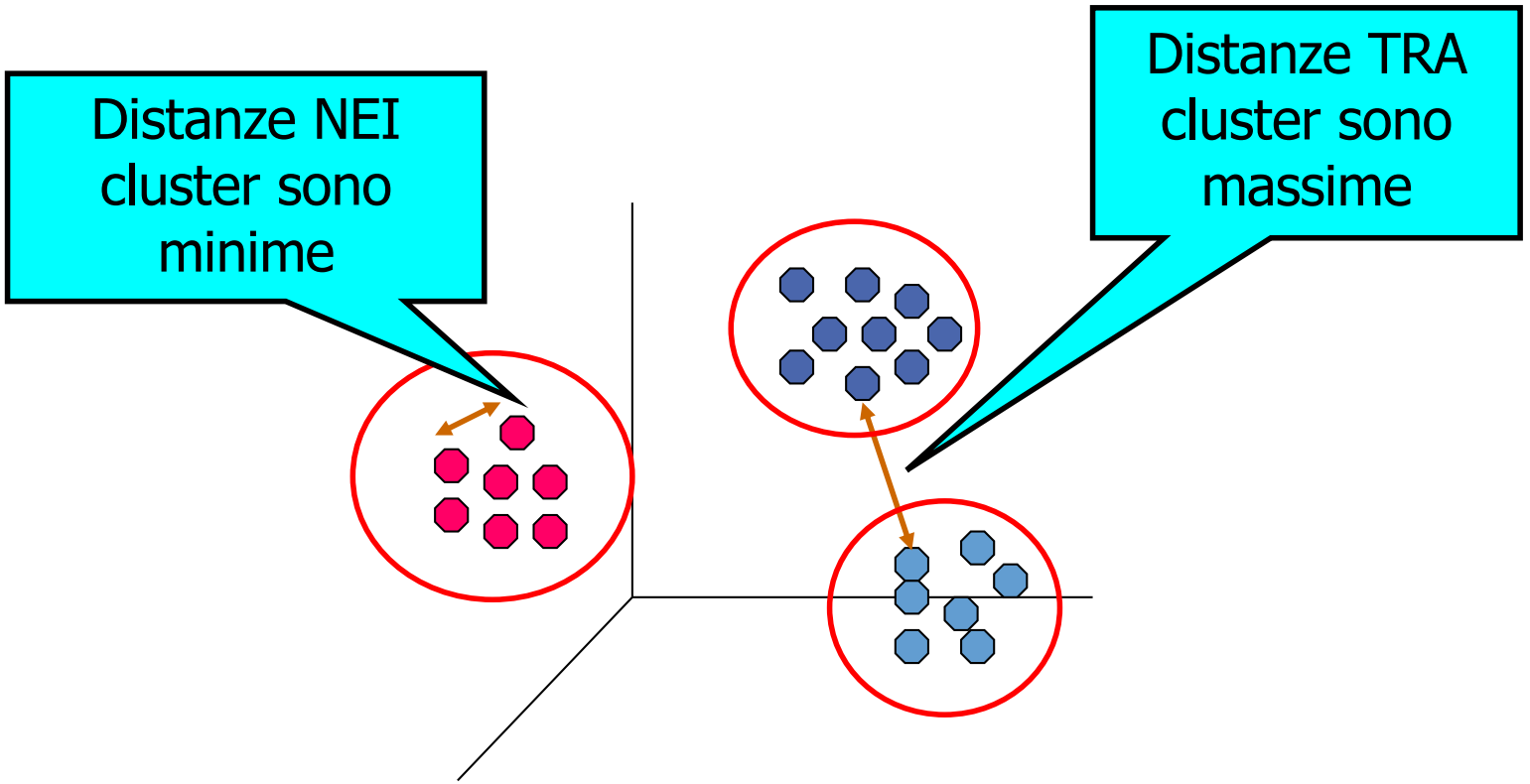
- Analisi esplorativa:
  - Pre-elaborazione per regressione, PCA, classificazione e analisi di associazione
- Compressione dati:
  - Elaborazione delle immagini: quantizzazione vettoriale
- Trovare i vicini reciproci a K
  - Finalizzare la ricerca su uno o un piccolo numero di gruppi
- Individuazione dei dati anomali
  - I valori anomali sono spesso visti come quelli "lontani" da qualsiasi gruppo

Soluzioni metodologiche

- Un buon metodo di clustering produce cluster di alta qualità
  - somiglianza intra-classe (within) alta: coesiva all'interno dei cluster
  - somiglianza tra classi (between) bassa: distintiva tra i cluster
- La qualità di un metodo di cluster dipende da;
  - la misura di somiglianza utilizzata dal metodo
  - la sua implementazione
  - la sua capacità di scoprire alcune o tutte le configurazioni nascoste

Cosa è la Cluster Analysis?

- Trovare gruppi di unità statistiche tali che le unità di un gruppo siano simili (o correlate) tra loro e diverse da (o estranee a) le unità in altri gruppi



Non è Cluster Analysis

- Classificazione supervisionata delle unità
  - Quando si ha già l'etichetta di appartenenza ad un gruppo
- Segmentazione semplice
  - Dividere gli studenti in gruppi in base al cognome
- Risultato di un'interrogazione del database
  - Raggruppare in base ad alcune caratteristiche predefinite
- Partizione grafica
  - Rilevanza parziale, ma non la stessa cosa

Distanze e Similarità

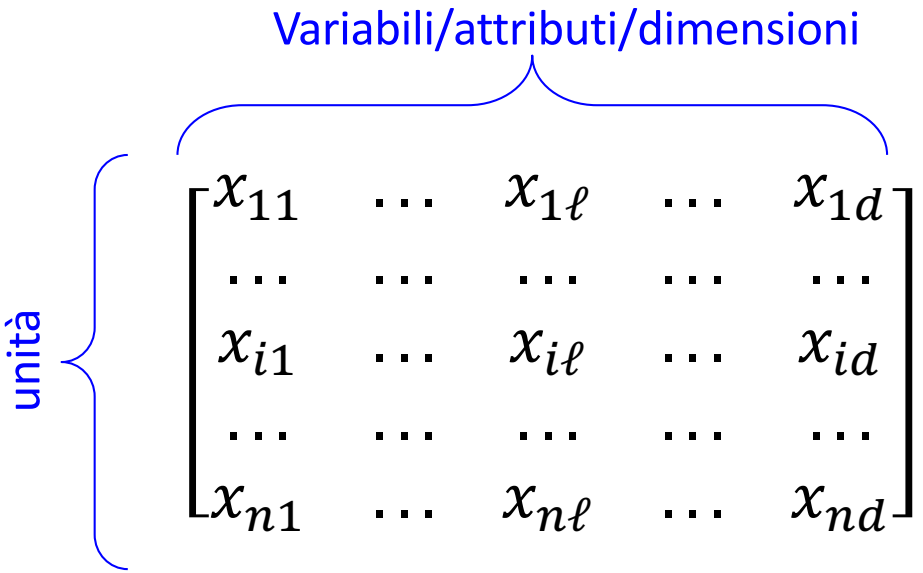
- Metrica dissimilarità / somiglianza
  - La somiglianza è espressa in termini di una funzione di distanza, in genere metrica: d(i, j) tra due unità i e j
  - Le definizioni delle funzioni di distanza sono in genere piuttosto diverse per il rapporto intervallo-scala, booleano, categorico, rapporto ordinale e variabili vettoriali
  - Problema definizione di distanza tra unità e gruppo oppure tra gruppo e gruppo
  - I pesi devono essere associati a diverse variabili basate su applicazioni e semantica dei dati
- Qualità del clustering:
  - Di solito c'è una funzione di "qualità" separata che misura la "bontà" di un cluster.
  - È difficile definire "abbastanza simile" o "abbastanza buono"
    - La soluzione è in genere molto soggettiva

Distanze

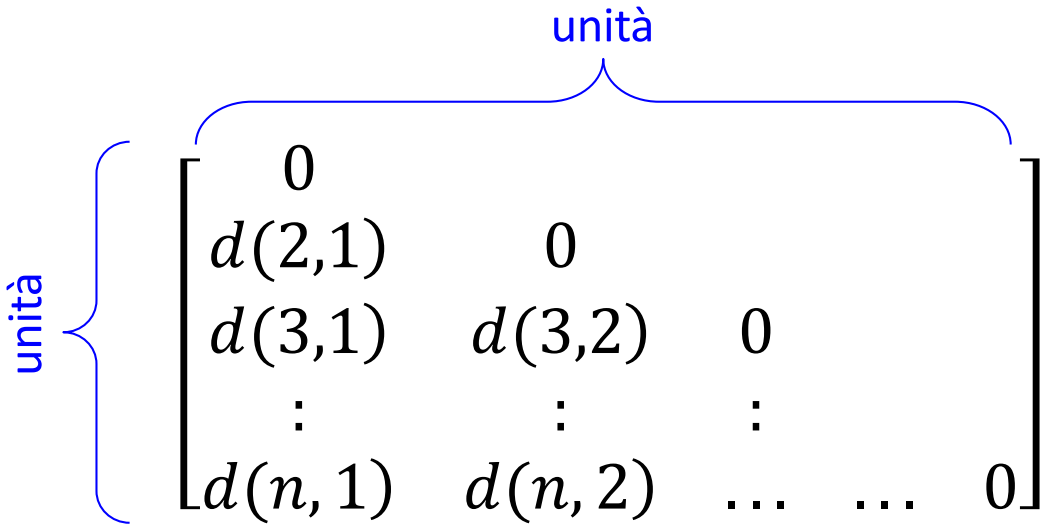
- La distanza **d(x, y)** tra due unità **x** ed **y** è una metrica se:
  - **d(i, j)≥0** (non-negatività)
  - **d(i, i)=0** (isolamento)
  - **d(i, j)= d(j, i)** (simmetria)
  - **d(i, j) ≤ d(i, h)+d(h, j)** (disuguaglianza triangolare) [**Perché è necessaria?**]
- Le definizioni di funzione di distanza sono solitamente molto diverse per variabili reali, booleane, categoriche, ed ordinali
- Si possono associare dei pesi a variabili differenti basate su realtà applicative.

Strutture Dati

- Matrice dei *dati*



- Matrice delle *distanze*



Distanze

- Norma **L_p** o distanza di *Minkowski* :

$$L_p(x,y) = (|x_1 - y_1|^p + |x_2 - y_2|^p + \ldots + |x_d - x_d|^p)^{1/p}$$

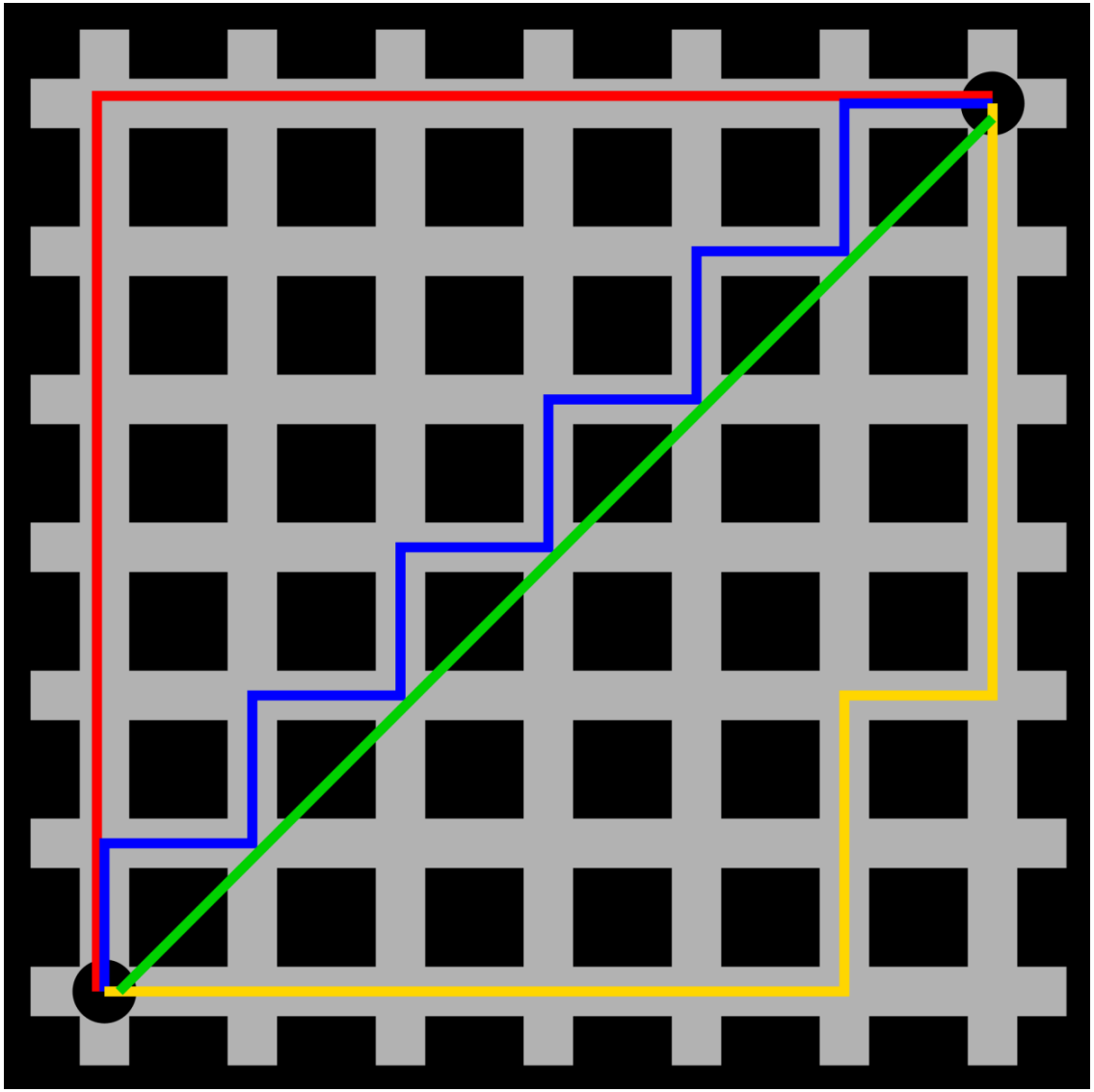
$$= \left(\sum_{i=1}^{d} (x_i - y_i) \right)^{1/p}$$

Dove ***p*** è un intero positivo

- Se **p = 1, L_1** è la distanza di *Manhattan (o della città a blocchi)*:

$$L_1(x,y) = |x_1 - y_1| + |x_2 - y_2| + \ldots + |x_d - y_d| = \sum_{i=1}^{d} |x_i - y_i|$$

Distanza di Minkowski: Esempi



Distanze

- Se **p = 2, L_2** è la distanza **Euclidea** :

$$d(x,y) = \sqrt{(|x_1 - y_1|^2 + |x_2 - y_2|^2 + \ldots + |x_d - y_d|^2)}$$

- Inoltre si può usare la distanza **pesata**:

$$d(x,y)$$
$$= \sqrt{(w_1|x_1 - x_1|^2 + w_2|x_2 - x_2|^2 + \ldots + w_d|x_d - y_d|^2)}$$
$$d(x,y)$$
$$= w_1|x_1 - y_1| + w_2|x_2 - y_2| + \ldots + w_d|x_d - y_d|$$

- Molto spesso **$L_p{}^p$** è utilizzata al posto di L_p (perché?)

Considerazioni sulla Cluster Analysis

- Criteri di partizionamento
  - Singolo livello contro il partizionamento gerarchico (spesso è preferibile il partizionamento gerarchico multilivello)
- Separazione dei cluster
  - Esclusivo (ad esempio, un cliente appartiene a una sola regione) o non esclusivo (ad esempio, un documento può appartenere a più di una classe)
- Misura di somiglianza
  - Basato sulla distanza (ad es., Euclidea, rete stradale, vettore) o basato sulla connettività (ad es. Densità o contiguità)
- Spazio dei cluster
  - Spazio pieno (spesso in caso di dimensioni ridotte) rispetto a sottospazi (spesso nel clustering ad alta dimensionalità)

Requisiti e Difficoltà

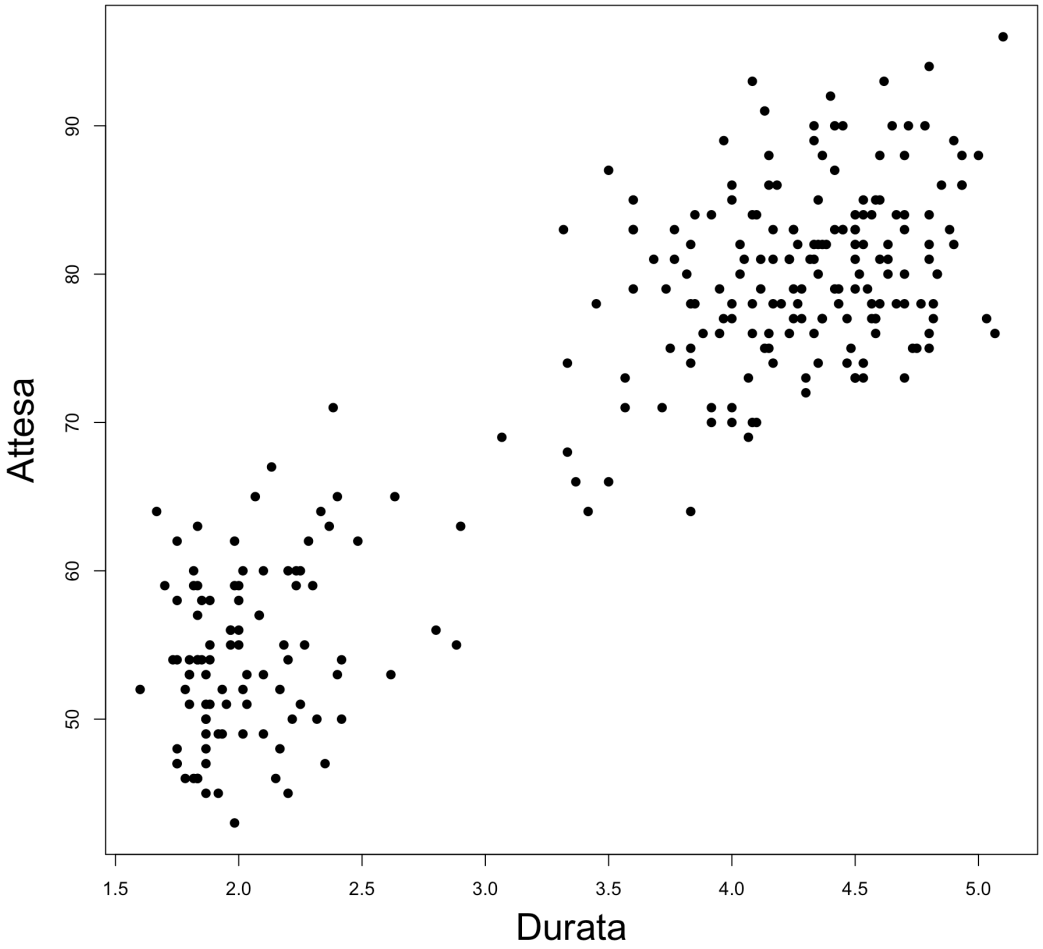
- Rappresentatività
  - Raggruppamento di tutti i dati anziché solo su campioni
- Capacità di gestire diversi tipi di attributi
  - Numerico, binario, categoriale, ordinale, collegato ed un misto di tipologie
- Cluster vincolato
  - L'utente può dare input sui vincoli
  - Utilizzare la conoscenza del dominio per determinare i parametri di input
- Interpretabilità e usabilità
- Altri
  - Identificazione di cluster con forma arbitraria
  - Capacità di gestire dati con errori di misura (noise)
  - Cluster incrementale e insensibilità all'ordine di input
  - Alta dimensionalità dei dati

Classi di metodi (algoritmi)

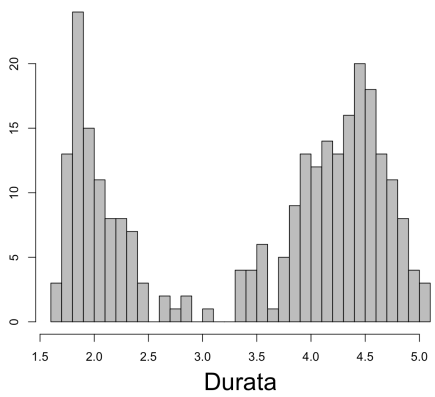
- Approccio di partizionamento:
  - Costruiscie varie partizioni e poi sceglie secondo uno o più criteri, ad esempio riducendo al minimo la somma degli errori quadrati
  - Metodi tipici: k-means, k-medoidi, CLARANS
- Approccio gerarchico:
  - Crea una scomposizione gerarchica dell'insieme delle unità (o oggetti) utilizzando alcuni criteri
  - Metodi tipici: Diana, Agnes, BIRCH, CAMELEON
- Approccio basato sulla densità:
  - Basato sulla densità (distribuzione, spesso adattata ad un modello) dei dati
  - Metodi tipici: DBSACN, OPTICS, DenClue

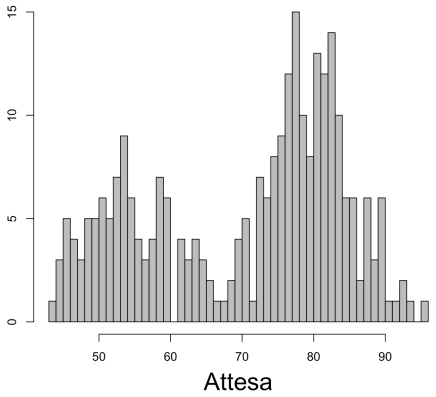
Esempio: Old Faithful

Wikipedia: Old Faithful è il nome con cui è noto uno dei geyser più famosi al mondo. Il nome gli fu assegnato nel 1870, le sue eruzioni durano dal minuto e mezzo fino ai cinque minuti a intervalli di 65-92 minuti. Nel 1938 Woodward per primo descrisse una relazione matematica tra la durata e l'intervallo temporale tra le eruzioni.




```
par(mar=c(4.5,4.5,1,1))
plot(faithful$eruptions,faithful$waiting,cex=1,pch=19,
xlab="Durata", ylab="Attesa",cex.lab=2)
par(mfrow=c(1,2),mar=c(4.5,4.5,1,1))
hist(faithful$eruptions,xlab="Durata",ylab="",main="",cex
.lab=2, col="gray",breaks = 40)
hist(faithful$waiting,xlab="Attesa",ylab="",main="",
cex.lab=2,col="gray",breaks = 40)
```





Metodi scissori

- Partizionamento di una matrice di dati **D** di *n* unità in un insieme di *k* gruppi, in modo tale che la somma delle distanze al quadrato sia ridotta al minimo (dove c_j è il centroide o il centro del gruppo C_i)

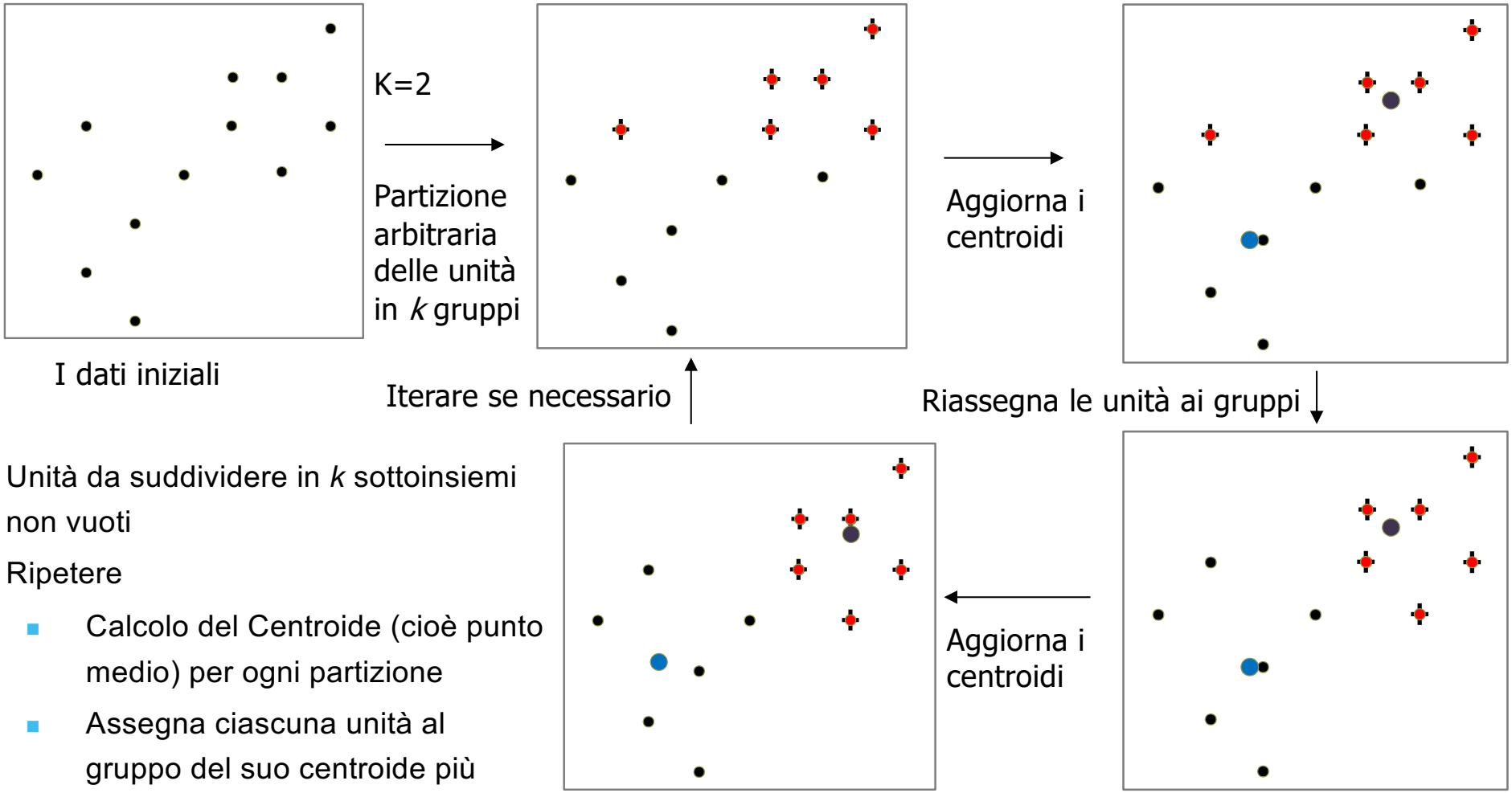
$$E = \Sigma_{i=1}^{k} \Sigma_{p \in C_i} (p - c_i)^2$$

- Dato *k*, trova una partizione di *k* gruppiche ottimizzi il criterio di partizionamento scelto
- Globale ottimale: enumerare esaurientemente tutte le partizioni
- Metodi euristici: algoritmi *k*-means e *k*-medoids
- *k*-means (MacQueen'67, Lloyd'57 / '82): ogni gruppo è rappresentato dal centro del cluster
- *k*-medoidi o PAM (Partition around medoids) (Kaufman & Rousseeuw'87): ogni gruppo è rappresentato da uno degli oggetti nel cluster

Il metodo del *k-means*

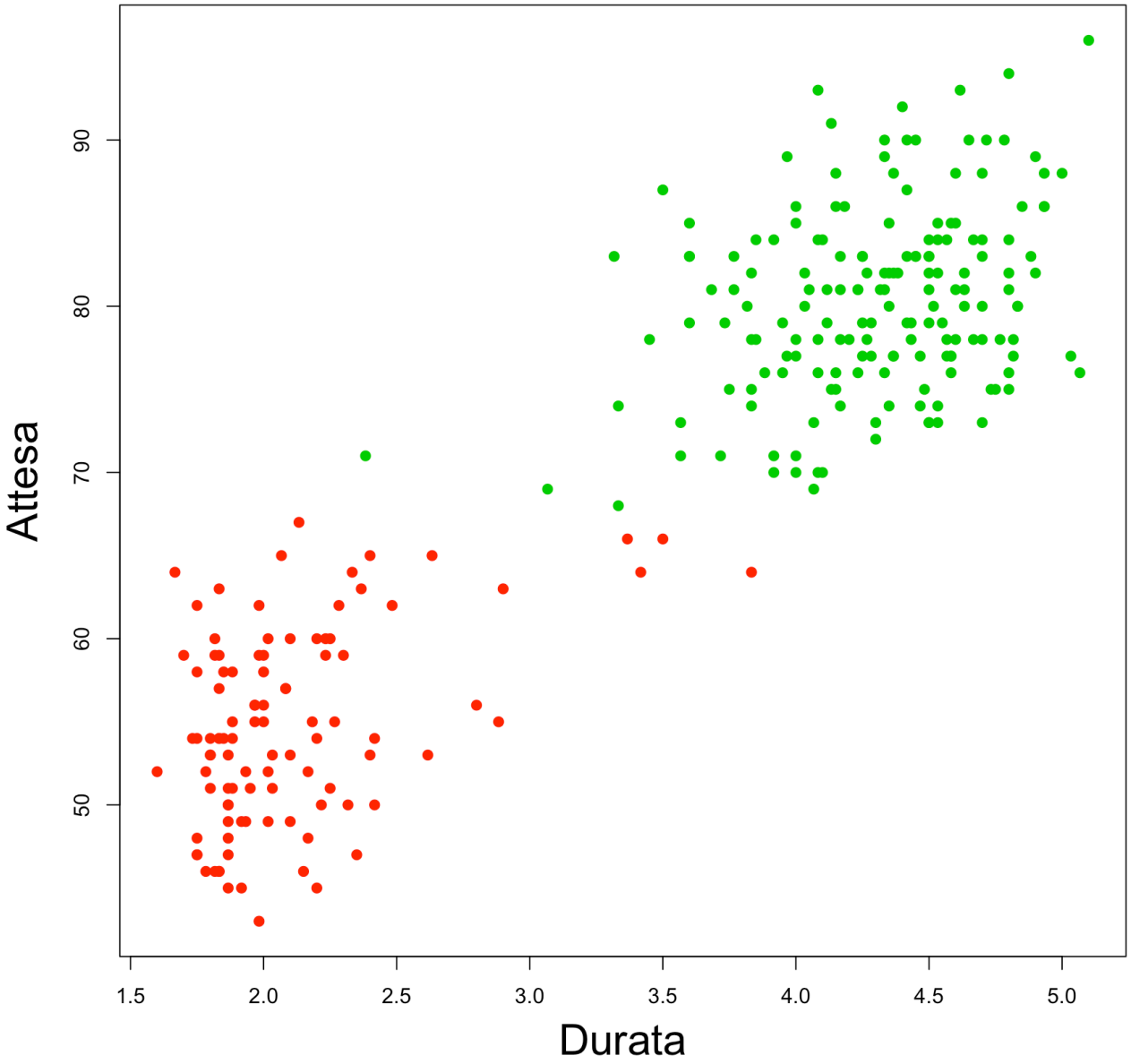
- Dato *k*, l'algoritmo *k-means* è implementato in quattro fasi:
  1. Partizionare le unità in *k* sottoinsiemi non vuoti
  2. Calcolare i punti seme (seed) come il centroide dei cluster del gruppo corrente (il centroide è il centro, cioè il punto medio del gruppo)
  3. Assegna ciascuna unità al gruppo con il punto di seme più vicino
  4. Torna al punto 2, fermati quando il risultato non cambia

Il metodo del *k-means*



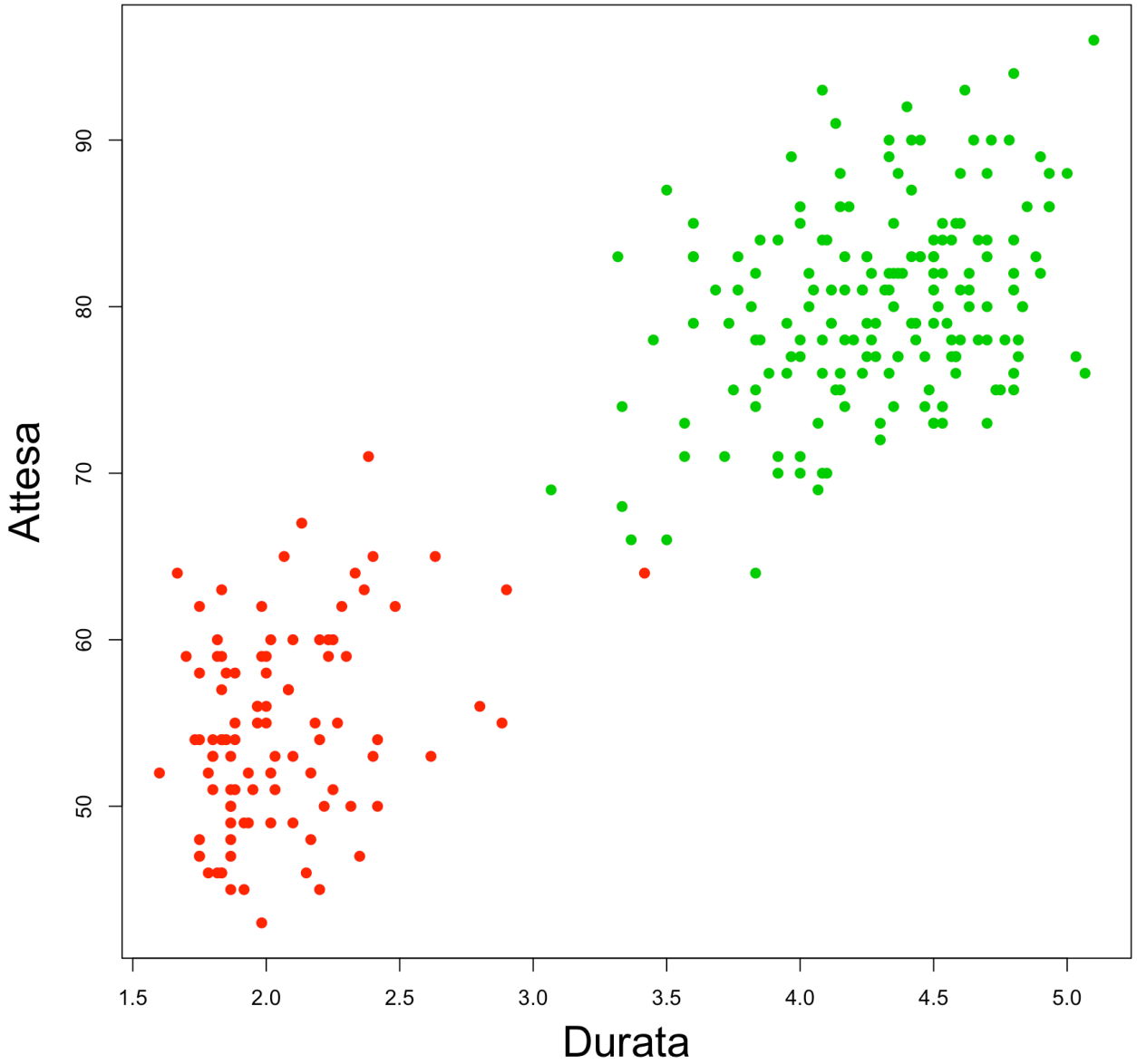
- Unità da suddividere in *k* sottoinsiemi non vuoti
- Ripetere
  - Calcolo del Centroide (cioè punto medio) per ogni partizione
  - Assegna ciascuna unità al gruppo del suo centroide più vicino
- Ripetere fino a nessun cambiamento

Esempio: Old Faithful



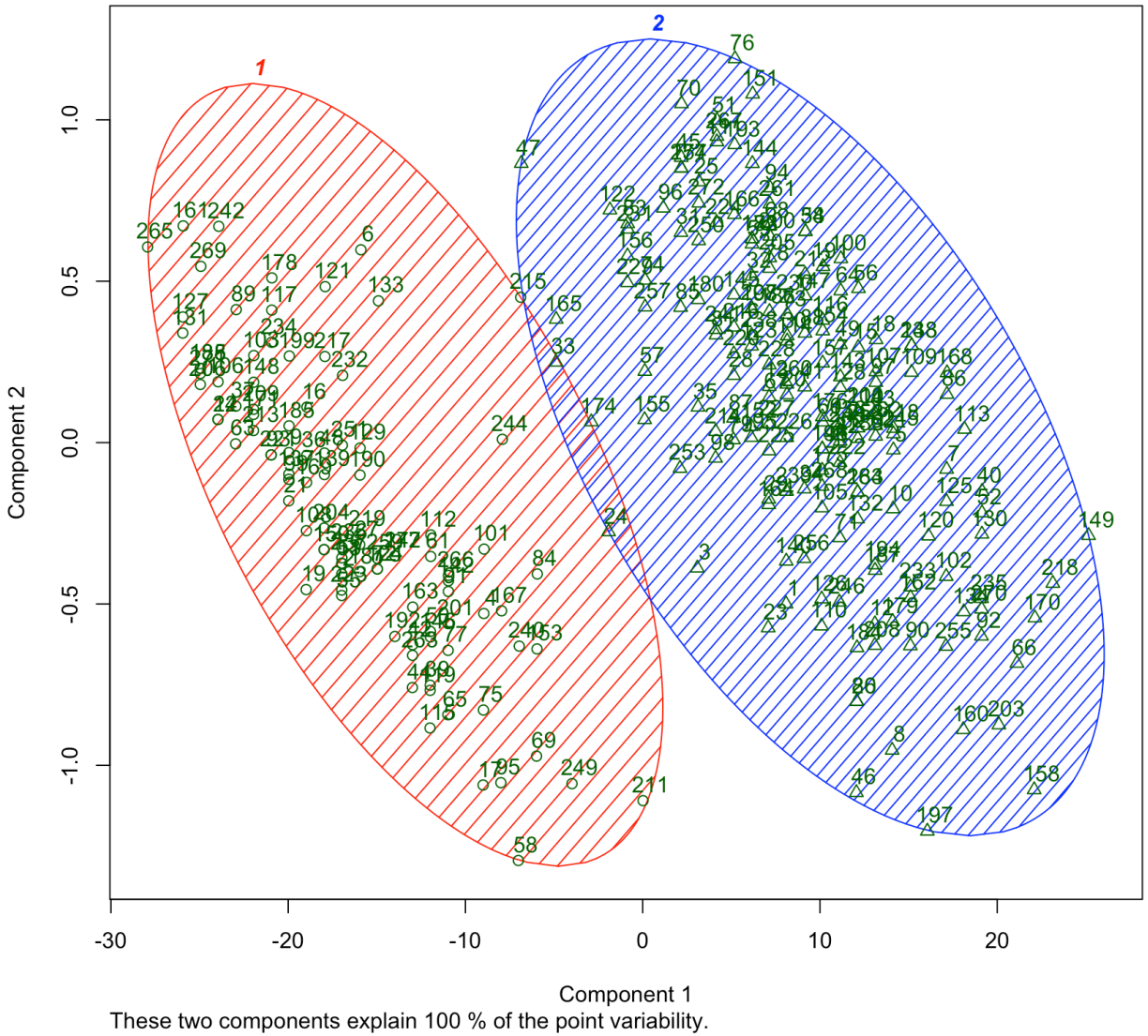

```
clus <- kmeans(faithful,2)
table(clus$cluster)
  1   2
100 172
par(mfrow=c(1,1),mar=c(4.5,4.5,1,1))
plot(faithful$eruptions,faithful$waiting,cex
=1,pch=19,xlab="Durata",ylab="Attesa",ce
x.lab=2,col=clus$cluster+1)
```

Esempio: Old Faithful (standardizzare le variabili)




```
clus <- kmeans(scale(faithful),2)
table(clus$cluster)
  1   2
 98 174
par(mfrow=c(1,1),mar=c(4.5,4.5,1,1))
plot(faithful$eruptions,faithful$waiting,cex
=1,pch=19,xlab="Durata",ylab="Attesa",ce
x.lab=2,col=clus$cluster+1)
```

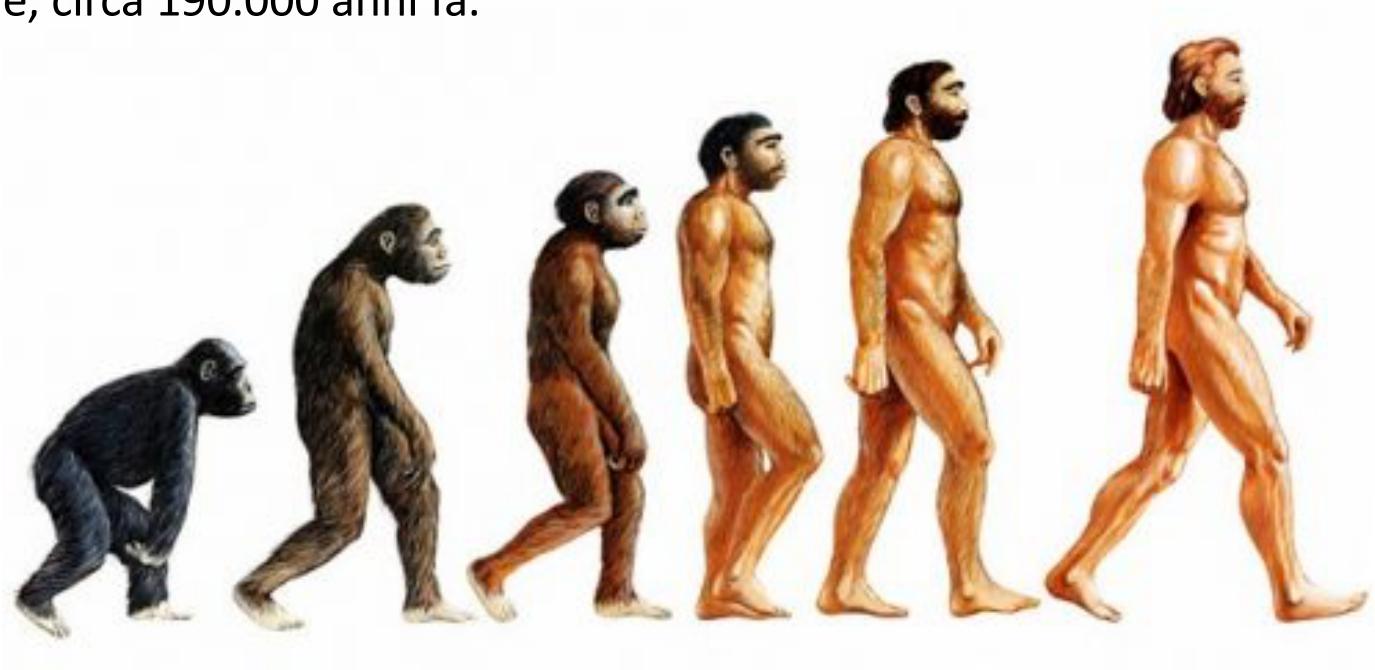
Esempio: Old Faithful (Uso delle PCA)



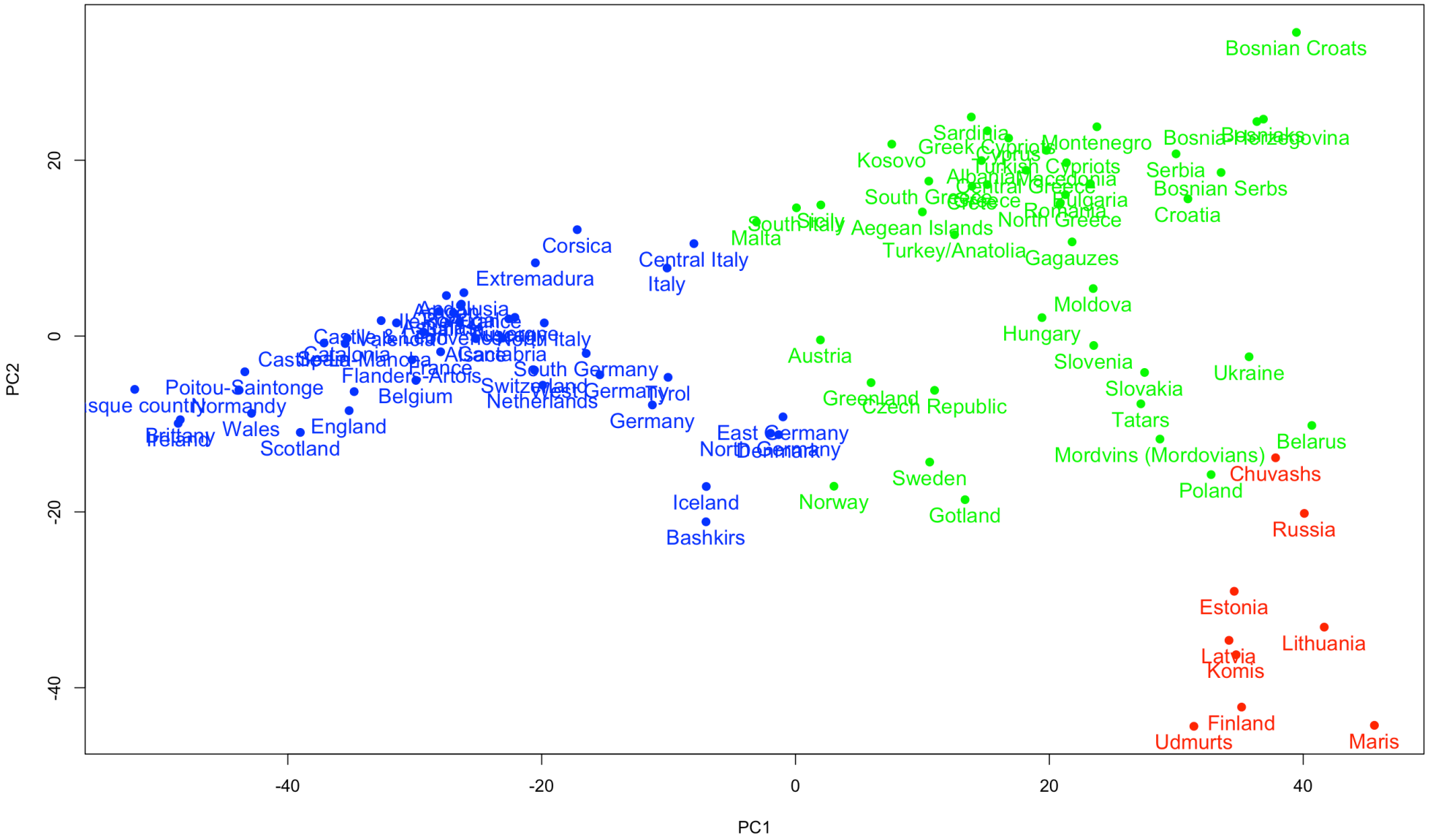

```
library(cluster)
par(mar=c(5,5,1,1))
clusplot(faithful, clus$cluster, color=TRUE,
shade=TRUE, labels=2, main="",lines=0)
clusplot(pam(faithful,2), color=TRUE,
shade=TRUE, labels=2, main="",lines=0)
```

Esempio: Aplogruppi

Nel campo della genetica umana, gli aplogruppi del DNA mitocondriale sono raggruppamenti di mutazioni (aplotipi) definiti dalle differenze tra un totale di 16569 paia di basi nel DNA del mitocondrio umano e questi gruppi rappresentano geneticamente l'eredità per relazione di parentela matrilineare di tutte le popolazioni umane, le loro origini ed i processi migratori.
Lo studio degli aplogruppi ha fornito risultati particolarmente significativi: le migrazioni dell'uomo parlano di un'origine nell'Africa orientale[1] e in base all'assunto che un individuo erediti i mitocondri solo dalla propria madre, tutti gli esseri umani hanno una linea di discendenza femminile che deriva da una donna che i ricercatori hanno soprannominato Eva mitocondriale, circa 190.000 anni fa.



Esempio: Aplogruppi

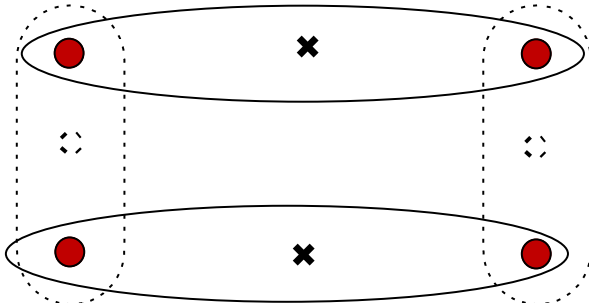


Commenti sul metodo *k-means*

- Forza: Efficiente: $O(tkn)$, dove n è # unità, k è # gruppi, e t è # iterazioni. Normalmente, k, $t << n$.
    - Confronto: PAM: $O(k(n-k)^2)$, CLARA: : $O(ks^2 + k(n-k))$
- Commento: spesso termina in un ottimo locale.
- Debolezza
    - Applicabile solo agli oggetti in uno spazio n-dimensionale continuo
        - Utilizzo del metodo *k*-modes per dati categoriali
        - In confronto, i *k*-medoidi possono essere applicati a un'ampia gamma di dati
    - È necessario specificare *k*, il *numero* di cluster, in anticipo (ci sono modi per determinare automaticamente il migliore k (vedi Hastie et al., 2009)
    - Influenzato da errori di misura ed *outlier*
    - Non adatto per individuare gruppi con *forme non convesse*

Variazioni del *k-means*

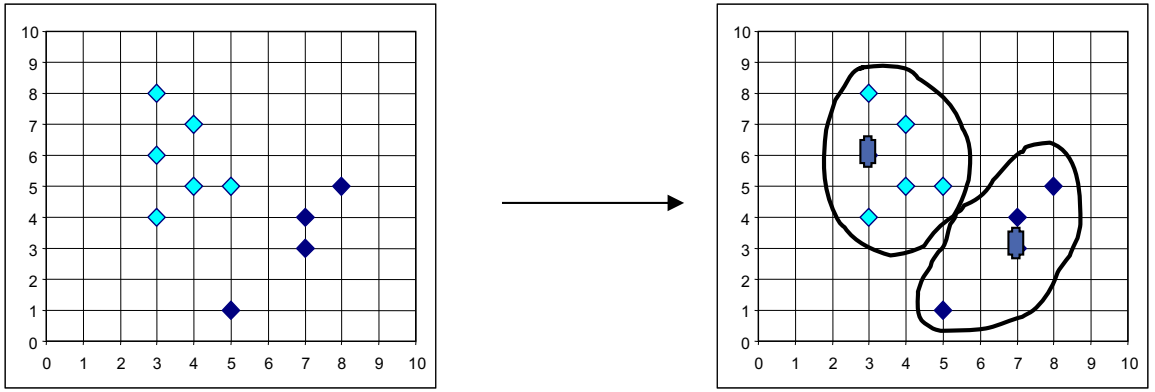
- La maggior parte delle varianti *k-means* differiscono in
  - Selezione delle *k* medie iniziali
  - Calcoli di dissimilarità
  - Strategie per calcolare le medie dei grappoli



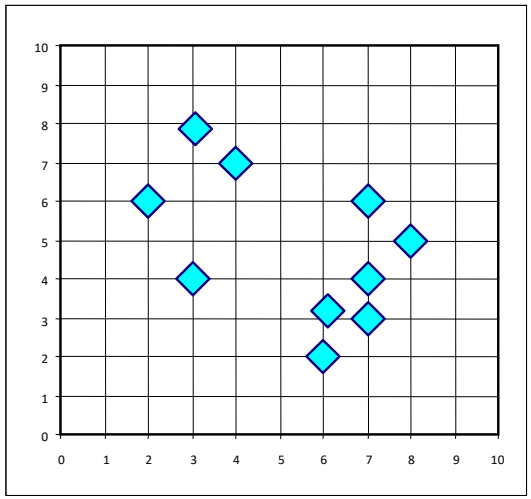
- Gestione dei dati categoriali*: k-modes*
  - Sostituzione le medie dei grappoli con modalità di freq. max.
  - Utilizzando nuove misure di dissomiglianza per gestire variabili categoriali
  - Utilizzo di un metodo basato sulla frequenza per aggiornare le mode dei gruppi
  - Una mistura di dati categoriali e numerici: metodo : *k-prototype*

Quale è il problema con il *k-means*

- L'algoritmo *k-means* è sensibile ai valori anomali!
  - Poiché una unità con un valore estremamente grande può sostanzialmente distorcere la distribuzione dei dati
- *K-Medoids*: invece di prendere il valore medio in un gruppo come punto di riferimento, è possibile utilizzare i medoidi, che è l'unità situata più centralmente in un gruppo

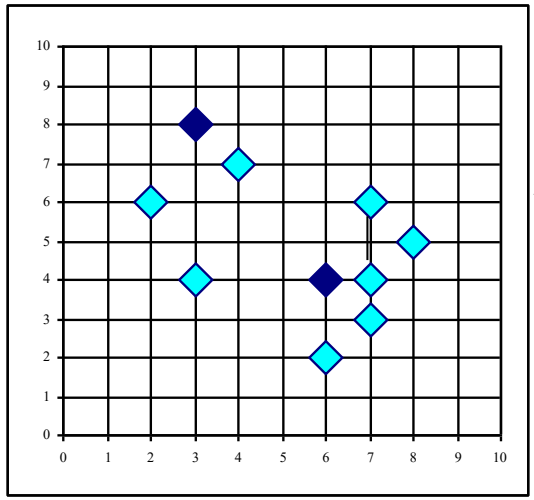


PAM: un tipico algoritmo *k-medoid*



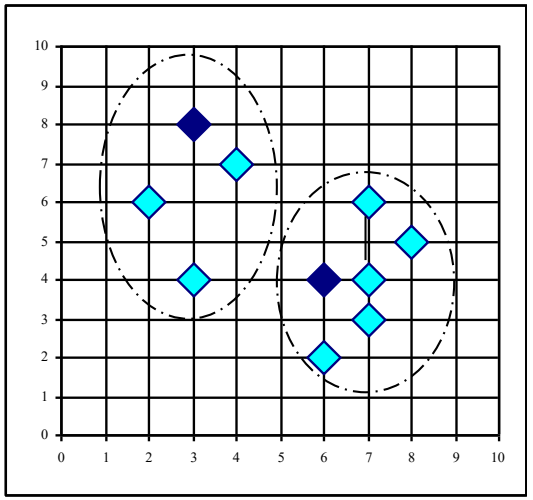
K=2

Scegliere arbitrariamente *k* medoidi

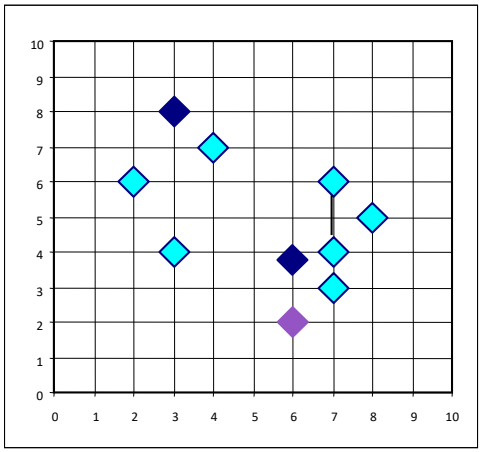


Assegnare ciascuna unità al medoide più vicino

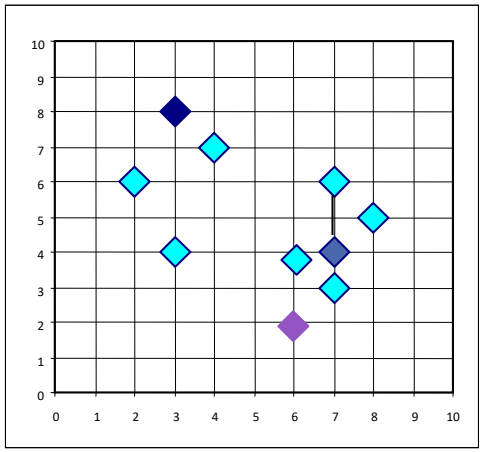
Total Cost = 20



Selezionare casualmente una unità non medoide, O_{ramdom}



Calcolare il costo totale di scambiare il medoide



Total Cost = 26

Scegliere O oppure O_{ramdom} se il costo si riduce

Ripetere finche non ci sono più cambiamenti

Algoritmi dei *k-medoid*

- *K-Medoids*: Usa una unità *reppresentativa* (medoide) in ciascun gruppo
  - *PAM* (Partitioning Around Medoids, Kaufmann & Rousseeuw 1987)
    - Inizia da un set iniziale di medoidi e sostituisce iterativamente uno dei medoidi con uno dei non-medoidi se migliora la distanza totale del clustering risultante
    - *PAM* funziona efficacemente per piccoli data-set, ma non si adatta bene ai grandi insiemi di dati (a causa della complessità computazionale)
    - Miglioramento dell'efficienza di *PAM*
  - *CLARA* (Kaufmann & Rousseeuw, 1990): *PAM* su campioni
  - *CLARANS* (Ng & Han, 1994): Ricampionamento casuale

PAM (Partitioning Around Medoids) (1987)

- PAM (Kaufman e Rousseeuw, 1987), sviluppato in R
- Usa unità reali per rappresentare il gruppo
  - Seleziona k unitàrappresentative arbitrariamente
  - Per ogni coppia di unità non selezionati ***h*** e selezionata l'unità i, calcola il costo totale di scambio TC_{ih}
  - Per ogni coppia ***i*** e ***h***,
    - Se $TC_{ih}<0$, ***i*** viene sostituita da ***h***
    - Quindi assegna ciascuna unità non selezionato all'unità rappresentativa più simile
- ripetere i passaggi 2-3 fino a quando non vi è alcun cambiamento

Quale è il problema con PAM ?

- PAM è più robusto di k-means in presenza di errori di misura e valori anomali perché un medeide è meno influenzato da valori anomali o altri valori estremi di quanto lo sia la media
- PAM funziona in modo efficiente per piccoli set di dati, ma non si adatta bene ai grandi insiemi di dati.
  - $O(k(n-k)^2)$ per ogni iterazione

    dove n è # di dati, k è # di gruppi
- Variante basata sul campionamento: CLARA (Clustering LARge Applications)

CLARA (Clustering Large Applications)

- *CLARA (Kaufmann e Rousseeuw nel 1990): sviluppato in pacchetti di analisi statistica, come R*
  - *Seleziona più campioni del set di dati, applica PAM su ciascun campione e fornisce il miglior clustering come output*
- *Forza:*
  - *si occupa di set di dati più grandi di PAM*
- *Debolezza:*
  - *L'efficienza dipende dalla dimensione del campione*
  - *Un buon clustering basato su campioni non rappresenta necessariamente un buon clustering dell'intero set di dati se il campione è distorto*

CLARANS ("Randomized" CLARA)

- *CLARANS (Algoritmo di clustering basato sulla ricerca casuale*) (Ng e Han'94)
  - Seleziona dinamicamente un campione di vicini
  - Il processo di clustering può essere presentato come ricerca di un grafo in cui ogni nodo è una soluzione potenziale, ovvero un insieme di k medoidi
  - Se viene trovato l'ottimo locale, inizia con un nuovo nodo selezionato casualmente nella ricerca di un nuovo ottimo locale
- Vantaggi: più efficiente e scalabile rispetto a PAM e CLARA
- Ulteriore miglioramento: tecniche di convergenza e strutture spaziali (Ester et al.'95)

Ricchi e Poveri

Indagine panel biennale sui bilanci delle famiglie italiane, particolarmente efficace nella rilevazione di redditi, ricchezza e risparmi. Disponibili I microdati nel web per le 7421 famiglie intervistate nel 2016. Ne utilizziamo un estratto dedicato alla ricchezza

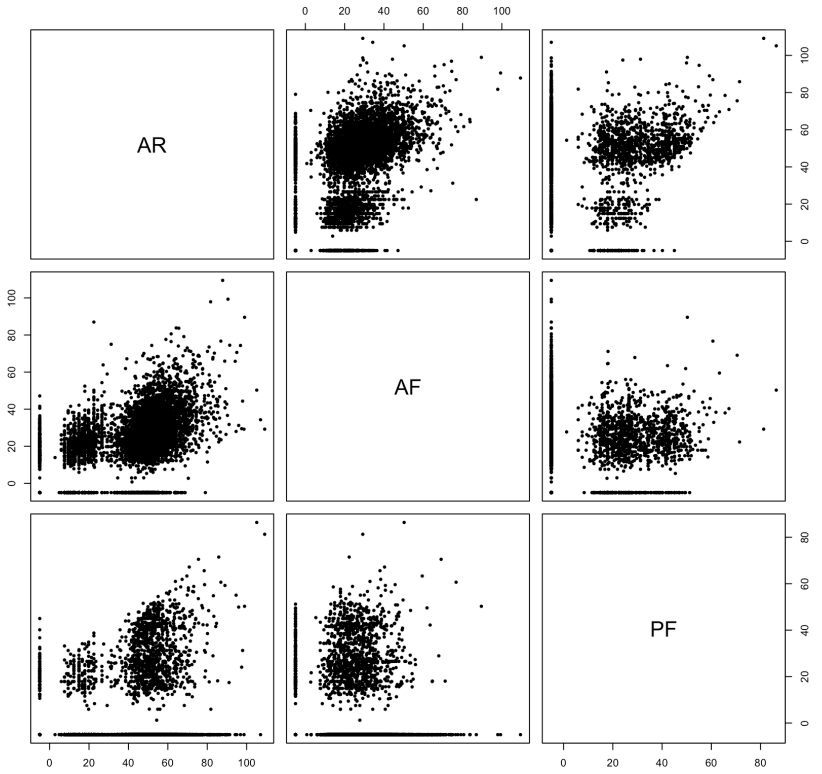
Trasformata di Box-Cox delle variabili

con $\lambda = 1/5$ $\quad y_i^{(\lambda)} = \frac{y_i^{\lambda} - 1}{\lambda}$

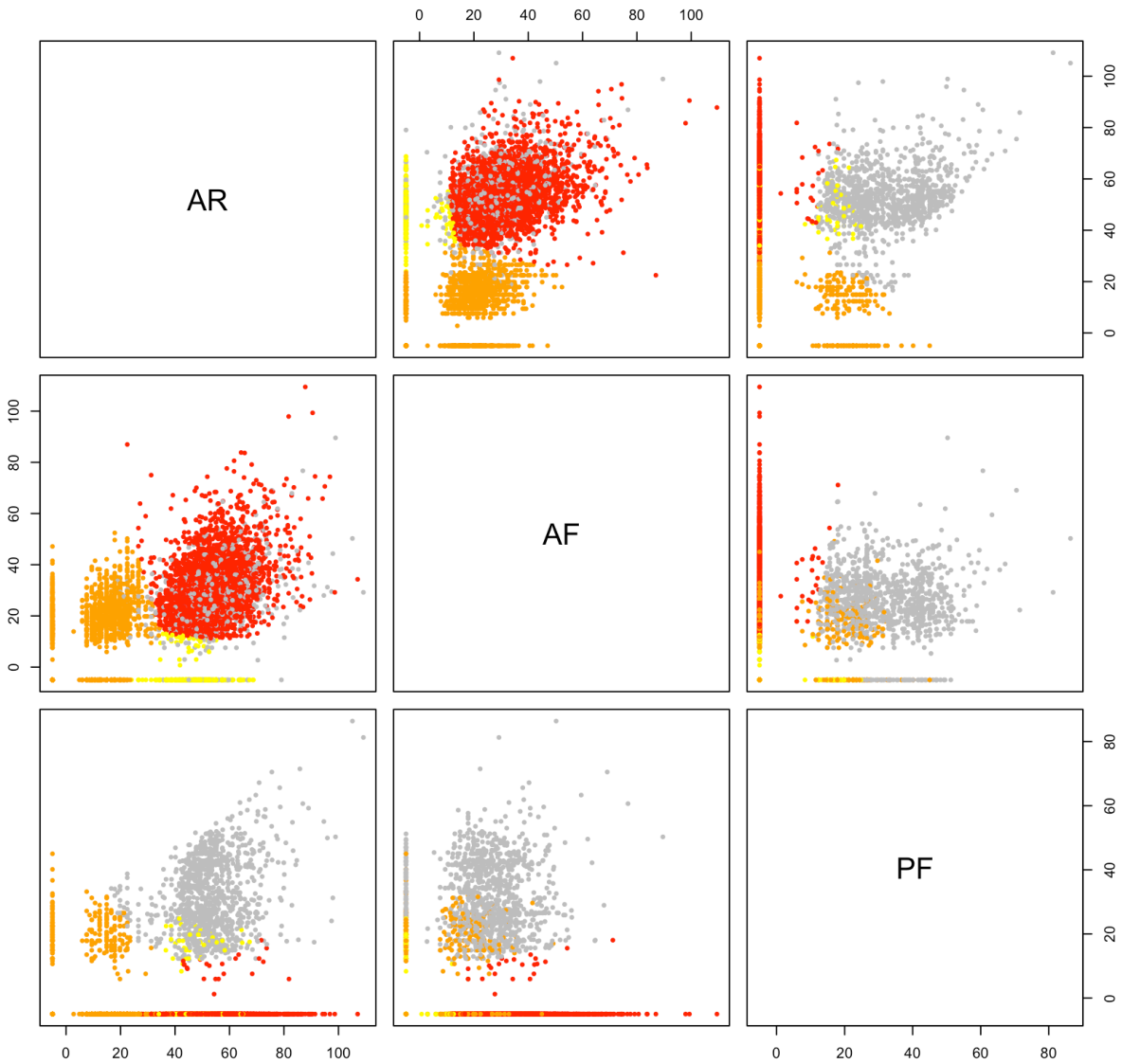
AR = Attività Reali

AF = Attività finanziarie

PF = Passività finanziarie



Ricchi e Poveri




```
datibi16tr <- (datibi16^(1/5)-1)*5
pairs(datibi16tr[,c(2,6,11)],cex=0.5,pch=19)
clbi16 <-
clara(datibi16tr[,c(2,6,11)],k=4,samples = 50,
sampsize = 500)
table(clbi16$clustering)
pairs(datibi16tr[,c(2,6,11)],cex=0.5,pch=19,col
=c("red","yellow","orange","gray","blue")[clbi
16$clustering])
tt <- table(clbi16$clustering,
datibi16$NASCAREA)
tt / rowSums(tt)
```

	1	2	3
1	0.4037412	0.2179787	0.3782801
2	0.1380952	0.1555556	0.7063492
3	0.3169062	0.1409993	0.5420945
4	0.3642931	0.2291022	0.4066047

Estraterrestri (esistono pianeti con vita?)

Spazio, ultima frontiera. Eccovi i viaggi dell'astronave Enterprise durante la sua missione quinquennale, diretta all'esplorazione di nuovi mondi, alla ricerca di altre forme di vita e di civiltà, fino ad arrivare laddove nessun uomo è mai giunto prima. (James Tiberius Kirk)

Missione Keplero, individuati più di 5000 esopianeti su una piccola porzione di spazio.

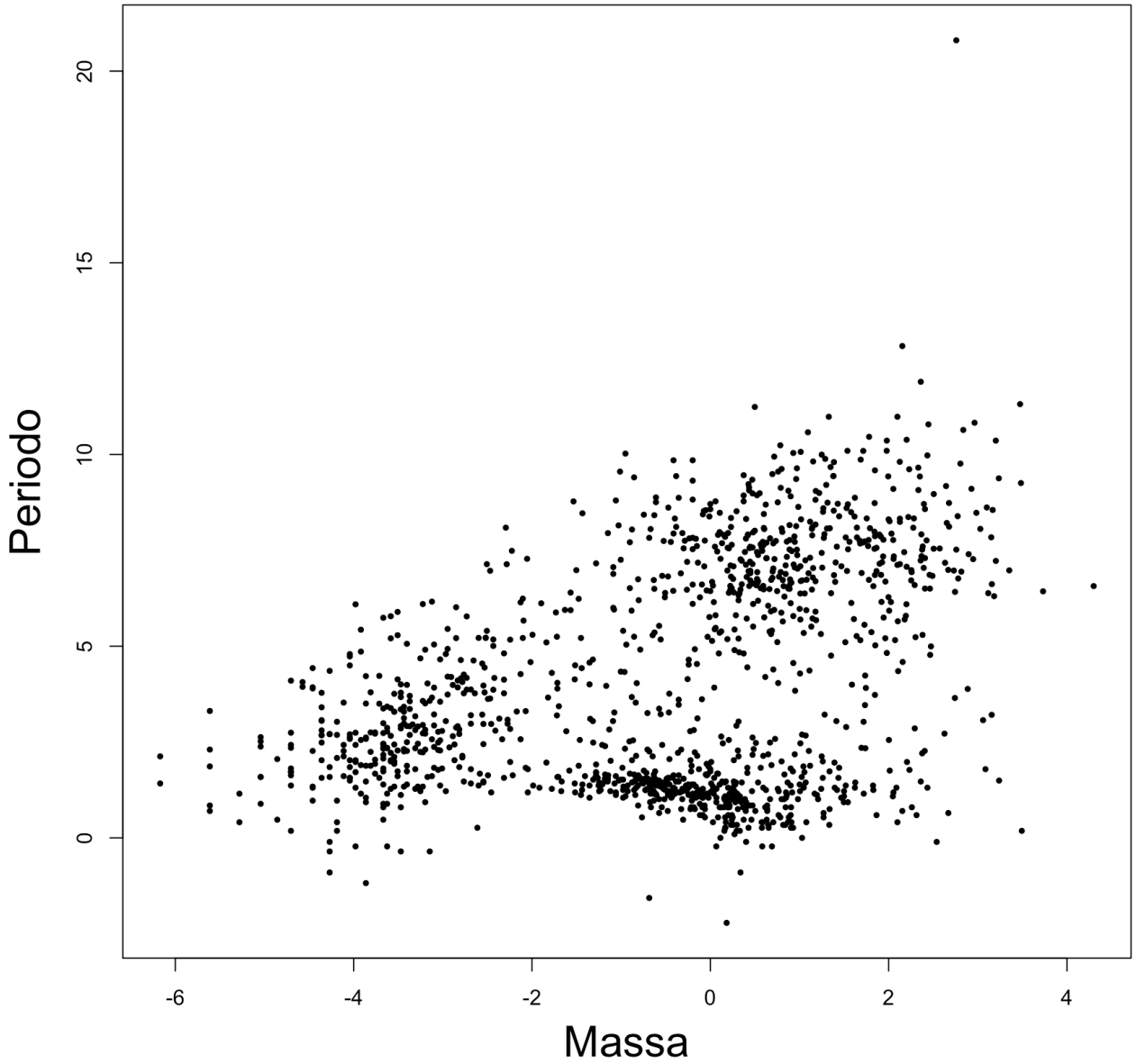
Rilevata distanza dalla stella, massa, raggio, densità etc

Sono stati definiti degli indici di vivibilità

Paradosso di Fermi ed equazione di Drake

l paradosso di Fermi, attribuito al fisico Enrico Fermi, sorge nel contesto di una valutazione della probabilità di entrare in contatto con forme di vita intelligente extraterrestre.
Il punto di partenza è il seguente ragionamento: dato l'enorme numero di stelle nell'universo osservabile, è naturale pensare che la vita possa essersi sviluppata in un grande numero di pianeti e che moltissime civiltà extraterrestri evolute siano apparse durante la vita dell'universo.
Formalmente ne segue l'equazione di Drake:
$N=R^* \times f_p \times n_e \times f_l \times f_i \times f_c \times L$ – stima pari a 10 (originale, elaborazioni NASA = 23,1)
dove: N è il numero di civiltà extraterrestri presenti oggi nella nostra Galassia con le quali si può pensare di stabilire una comunicazione; R^* è il tasso medio annuo con cui si formano nuove stelle nella Via Lattea (10); f_p è la frazione di stelle che possiedono pianeti (0,5); n_e è il numero medio di pianeti per sistema planetario in condizione di ospitare forme di vita (2); f_l è la frazione dei pianeti n_e su cui si è effettivamente sviluppata la vita (1); f_i è la frazione dei pianeti f_l su cui si sono evoluti esseri intelligenti (0.01); f_c è la frazione di civiltà extraterrestri in grado di comunicare (0,01); L è la stima della durata di queste civiltà evolute (10000). Da tale relazione nasce però la domanda-paradosso:
«*Se ci sono così tante civiltà evolute, perché non ne abbiamo ancora ricevuto le prove, come trasmissioni radio, sonde o navi spaziali?*»
Il "paradosso" è il contrasto tra l'affermazione che non siamo soli nell'Universo e i dati osservativi che contrastano con questa ipotesi. Ne deriva che la nostra osservazione/comprensione dei dati è incompleta.

Esopianeti

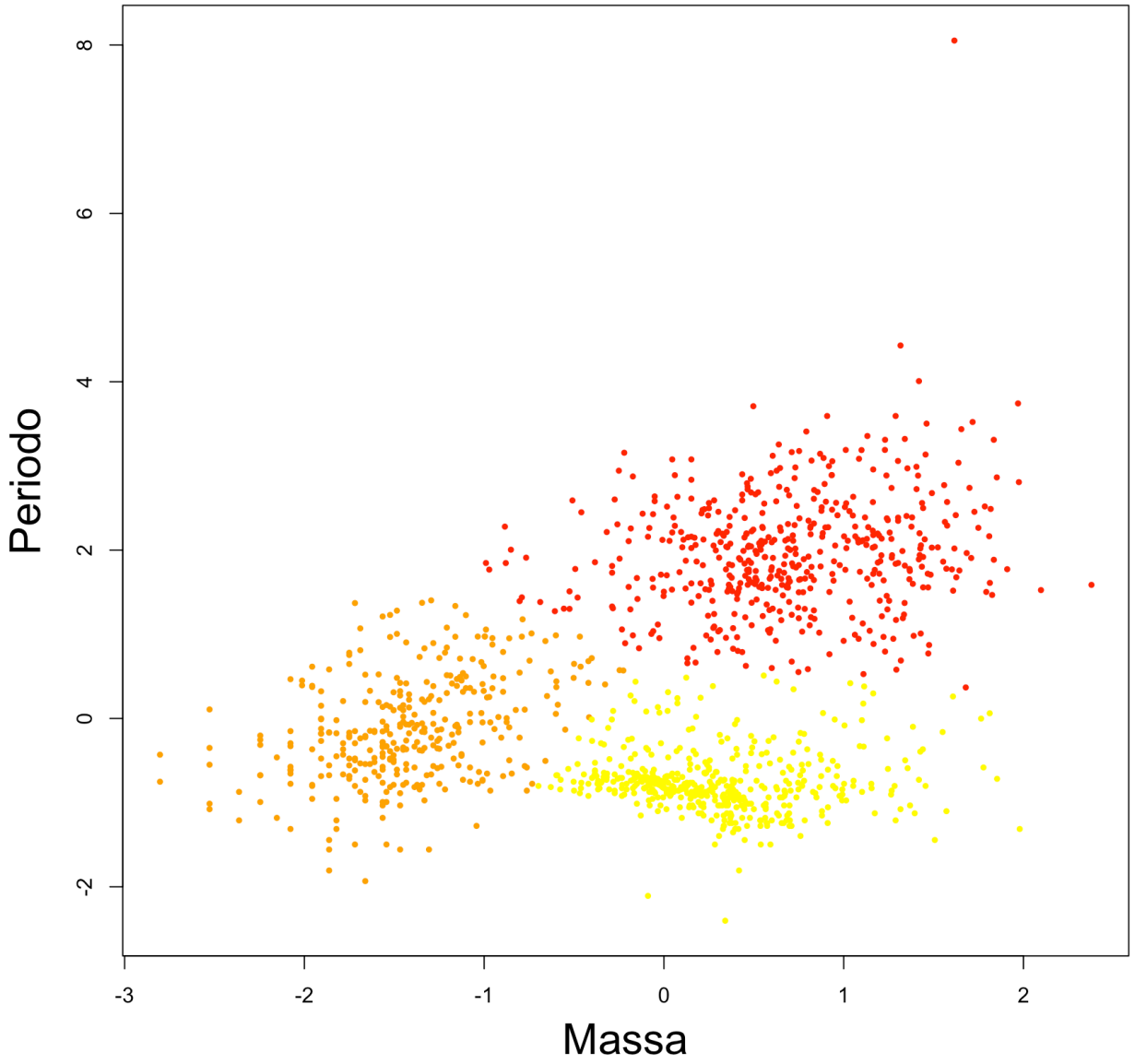


Trasformata di Box-Cox delle variabili con $\lambda = 1/30$

Su 3679 pianeti identificati ed accettati

Di cui solo 1316 hanno massa stimata, ossia non dato mancante

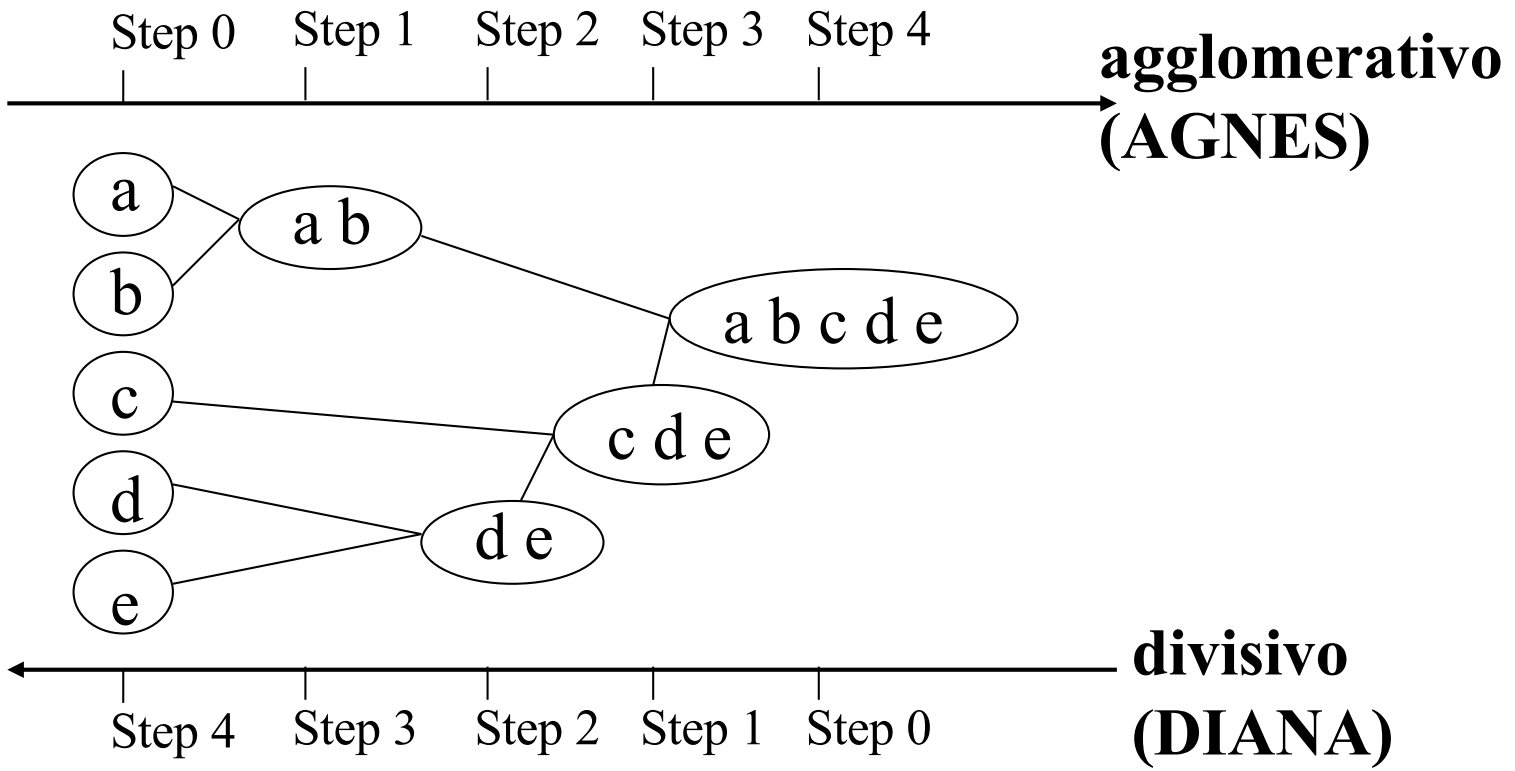
Esopianeti




```
massa <- (dati$MASS^(1/30)-1)*30
perio <- (dati$PERIOD^(1/30)-1)*30
x <-
scale(cbind(massa,perio),center=T,scale=T)
x <- x[complete.cases(x),]
par(mar=c(4.5,4.5,1,1))
plot(x,cex=0.5,pch=19,xlab="Massa”,
ylab="Periodo",cex.lab=2)
pampl <- pam(x, 3)
plot(x,cex=0.5,pch=19,xlab="Massa",
ylab="Periodo",cex.lab=2,col=c("red","yellow",
"orange")[pampl$clustering])
```

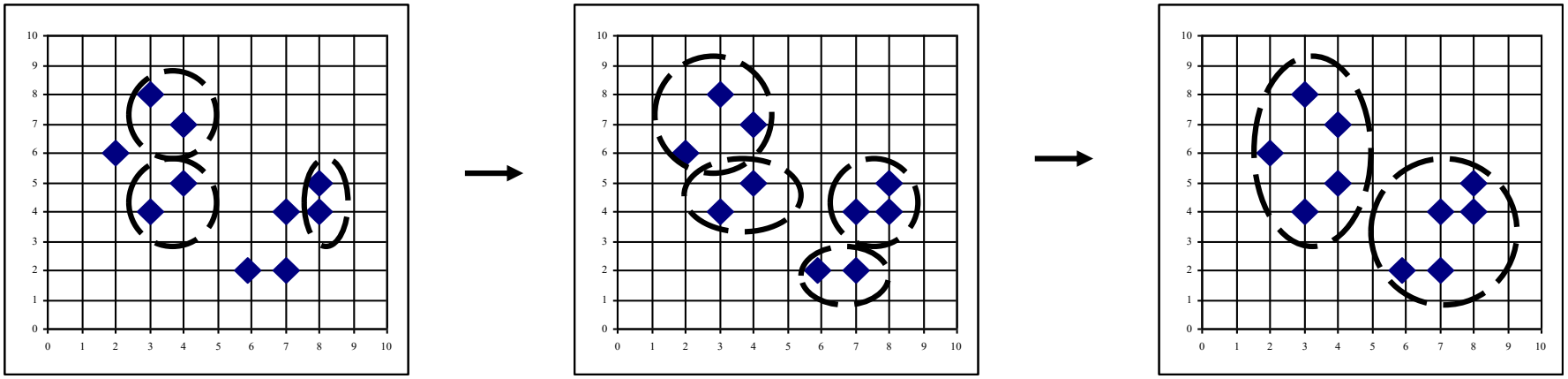
Metodi gerarchici

Utilizza la matrice delle distanze come criterio di raggruppamento. Questa classe di metodi non richiede il numero di cluster ***k*** come input, ma richiede una condizione di arresto



AGNES (Agglomerative Nesting)

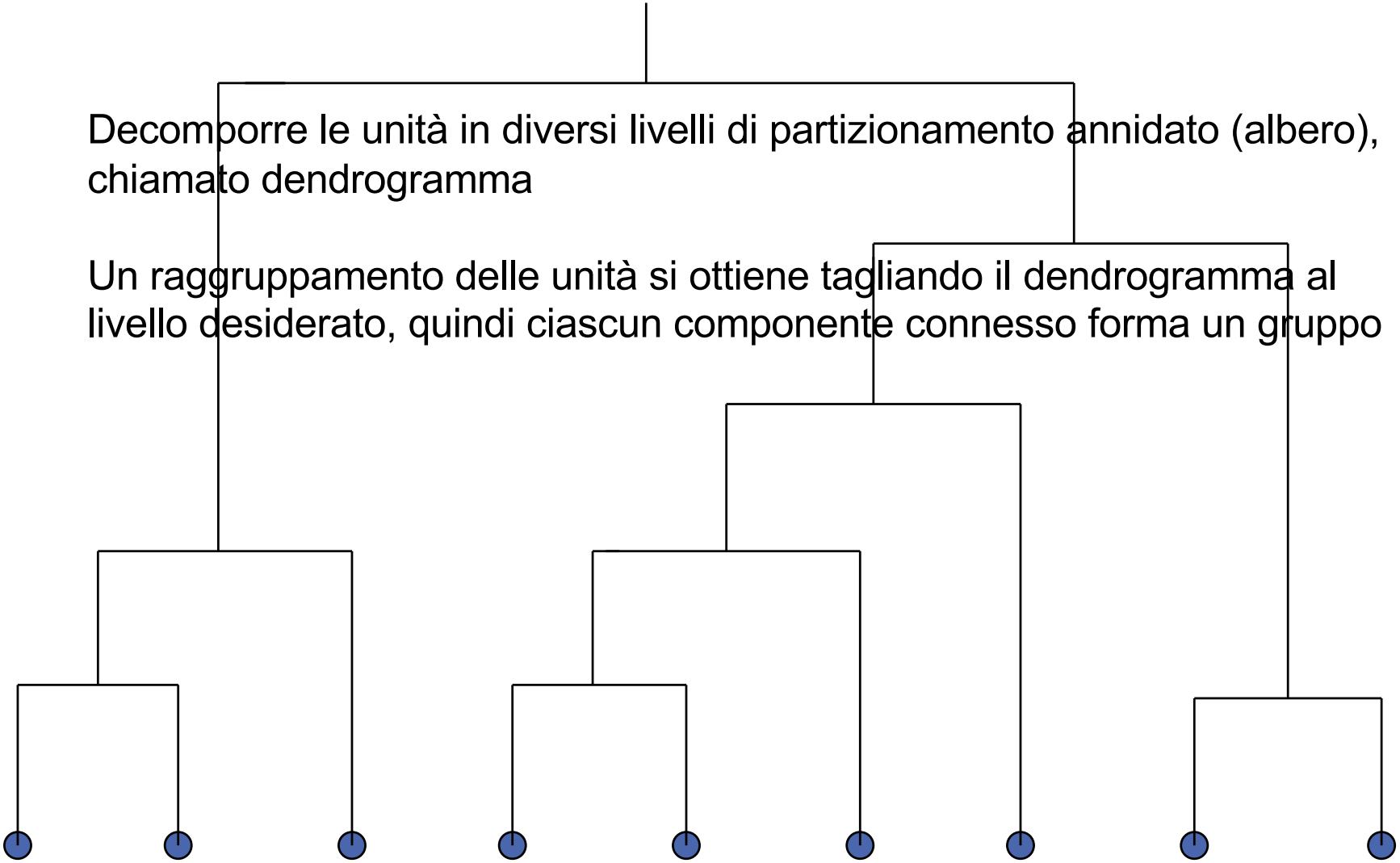
- Introdotto da Kaufmann e Rousseeuw (1990)
- Implementato in pacchetti statistici, ad es. R
- Utilizza il metodo ***Single-Linkage*** e la matrice di dissomiglianza
- Unisce nodi che hanno la minor differenza
- Continua in modo non discendente
- Alla fine tutti i nodi appartengono allo stesso gruppo



Dendrogramma: visualizza come I gruppi sono uniti

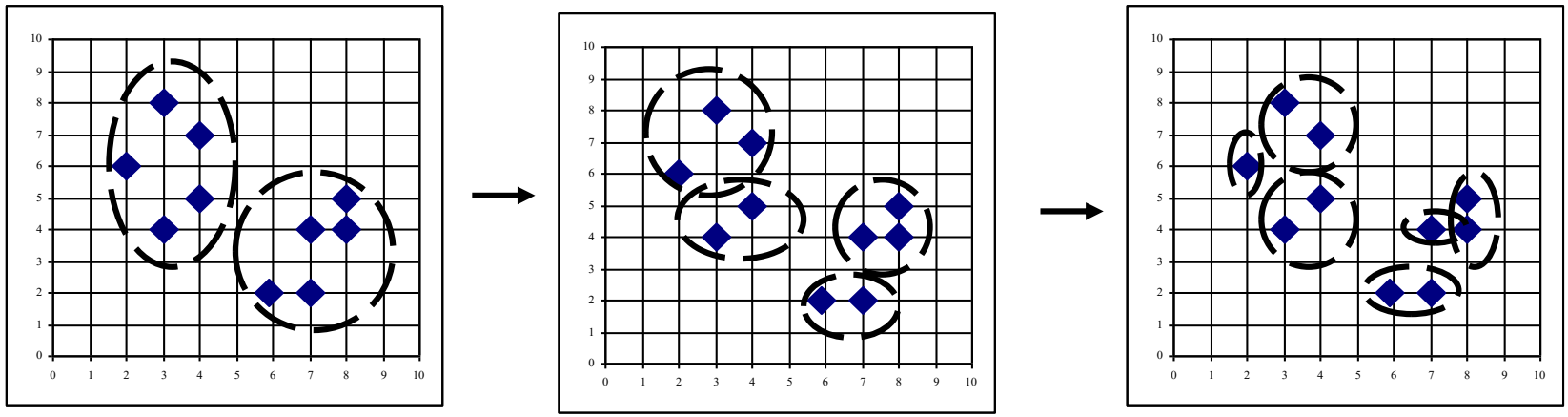
Decomporre le unità in diversi livelli di partizionamento annidato (albero), chiamato dendrogramma

Un raggruppamento delle unità si ottiene tagliando il dendrogramma al livello desiderato, quindi ciascun componente connesso forma un gruppo



DIANA (Divisive Analysis)

- Introdotto da Kaufmann e Rousseeuw (1990)
- Implementato in pacchetti di analisi statistiche, ad es. R
- Ordine inverso di AGNES
- Alla fine ogni nodo forma un cluster da solo



Distanze tra gruppi

- Single link: distanza minima tra un elemento in un gruppo e un elemento nell'altro, cioè dist(K_i, K_j) = min(t_{ip}, t_{jq})
- Complete link: maggiore distanza tra un elemento in un gruppo e un elemento nell'altro, cioè dist(K_i, K_j) = max(t_{ip}, t_{jq})
- Media: distanza media tra un elemento in un cluster e un elemento nell'altro, cioè dist(K_i, K_j) = avg(t_{ip}, t_{jq})
- Centroide: distanza tra i centroidi di due gruppi, cioè , dist(K_i, K_j) = dist(C_i, C_j)
- Medoide: distanza tra i medoidi di due gruppi, cioè , dist(K_i, K_j) = dist(M_i, M_j)
  - Medoide: un oggetto scelto, posizionato centralmente nel cluster

Alcune definizioni

- Centroide: il "centro" di un cluster

$$C_m = \frac{\Sigma_{i=1}^{N}(t_{ip})}{N}$$

- Raggio: radice quadrata della distanza media da qualsiasi punto del cluster al suo centroide

$$R_m = \sqrt{\frac{\Sigma_{i=1}^{N}(t_{ip} - c_m)^2}{N}}$$

- Diametro: radice quadrata della distanza quadratica media tra tutte le coppie di punti nel gruppo

$$D_m = \sqrt{\frac{\Sigma_{i=1}^{N}\Sigma_{i=1}^{N}(t_{ip} - t_{iq})^2}{N(N-1)}}$$

Metodo di Ward

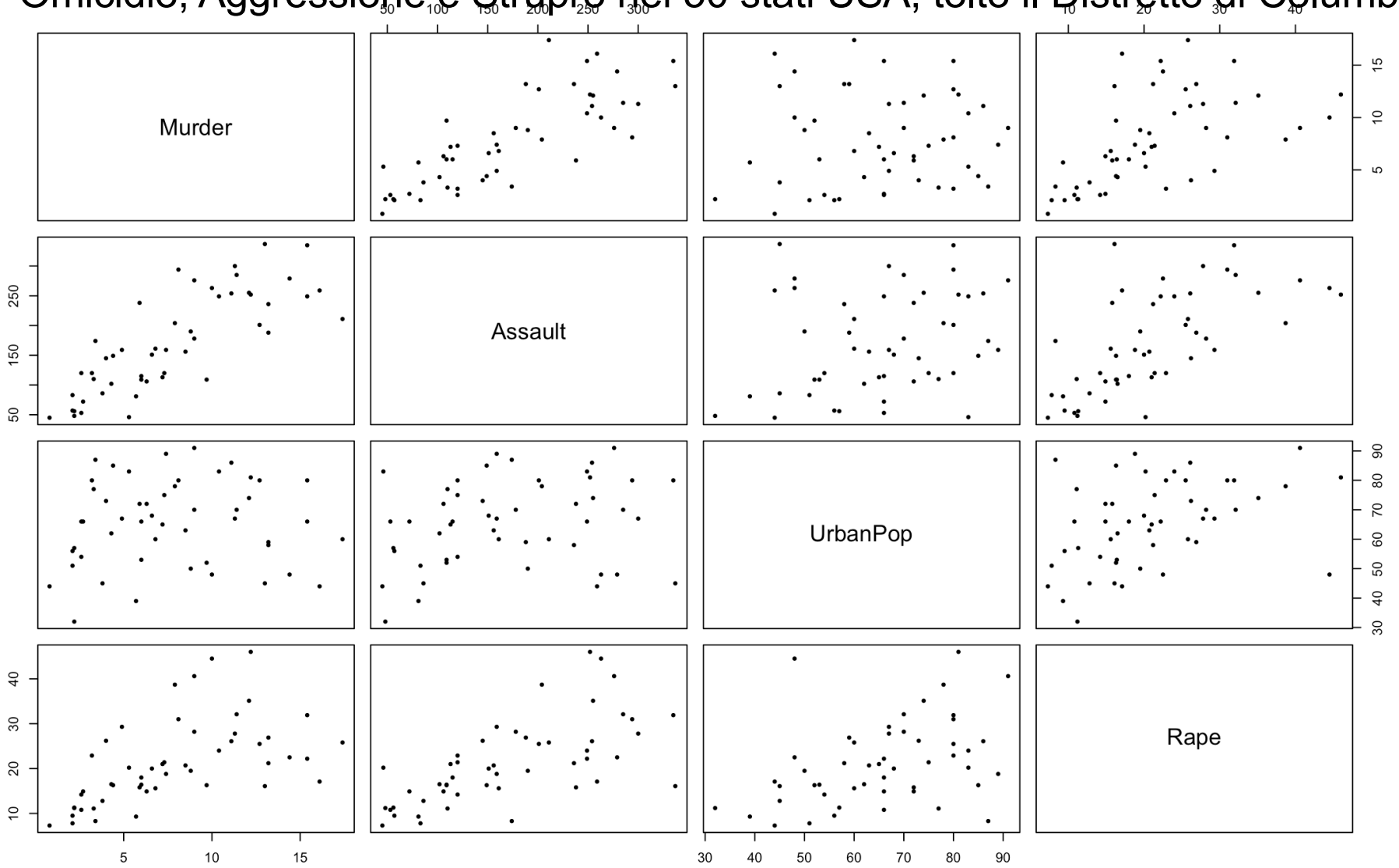
- La similarità di due gruppi si basa sull'incremento di SSE quando i due gruppi vengono uniti: tanto minore l'incremento tanto più elevata la similarità
- Simile al criterio della media se la distanza tra punti è la distanza Euclidea al quadrato.
- Meno sensibile a rumore e outliers e sbilanciato verso cluster globulari
- Analogo gerarchico di K-means
  - Può essere usato per scegliere I centroidi in K-means

Alcune considerazioni

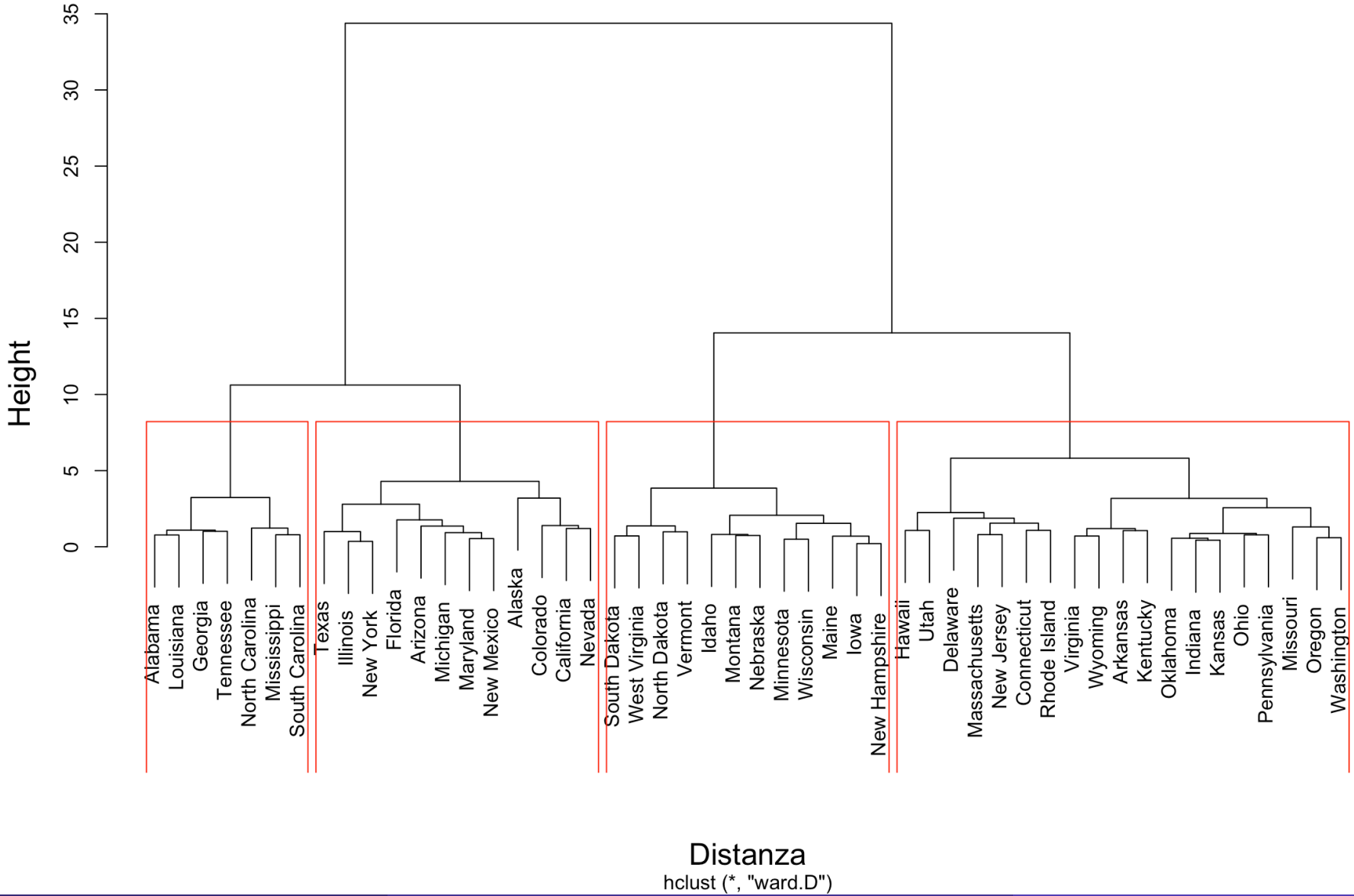
- Principale debolezza dei metodi di clustering agglomerativi
  - Non possono mai annullare ciò che è stato fatto nei passi precedenti
  - Non scalano bene: la complessità temporale di almeno O (n^2), dove n è il numero di unità
- Integrazione di clustering gerarchico e basato sulla distanza
  - BIRCH (1996): utilizza CF-tree e regola in modo incrementale la qualità dei sottogruppi
  - CHAMELEON (1999): clustering gerarchico mediante modellazione dinamica

Criminalità stati USA

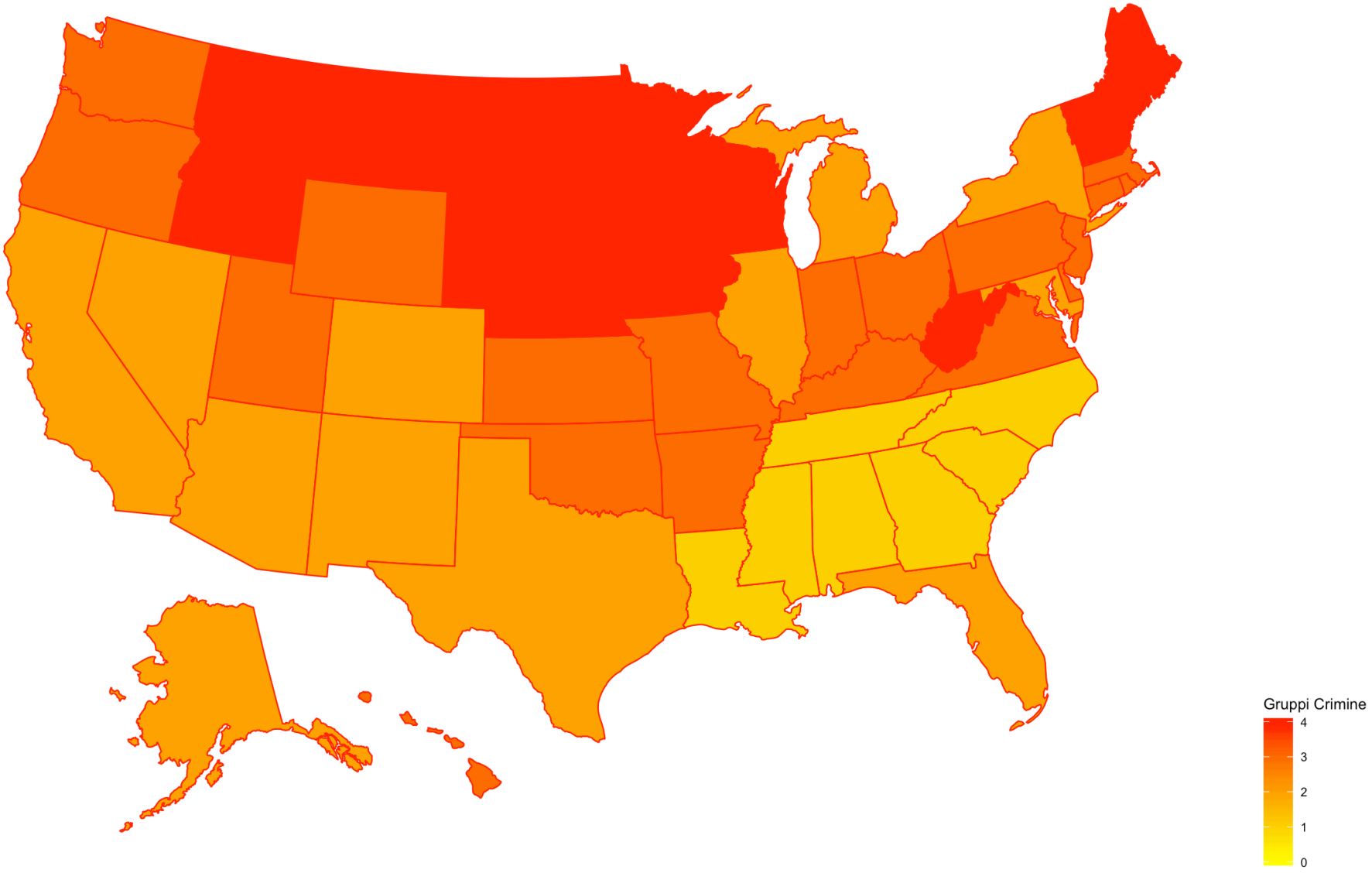
- Omicidio, Aggressione e Strupro nei 50 stati USA, tolto il Distretto di Columbia



Criminalità stati USA



Criminalità stati USA



Criminalità stati USA

```
pairs(USArrests,cex=0.5,pch=19)
prusa <- prcomp(USArrests,4,scale=T)
Distanza <- dist(prusa$x, method = "euclidean")
fit <- hclust(Distanza, method="ward.D")
par(mar=c(5,5,1,1))
plot(fit,main="",cex.lab=1.5)
groups <- cutree(fit, k=4)
rect.hclust(fit, k=4, border="red")
library(usmap)
library(ggplot2)
newdat <- as.data.frame(statepop)
newdat$clust <- rep(0,51)
newdat$clust[1:8] <- groups[1:8]
newdat$clust[10:51] <- groups[9:50]
plot_usmap(data = newdat, values = "clust", lines = "red") +
  scale_fill_continuous(low = "yellow", high = "red", name = "Gruppi Crimine", label =
scales::comma) +  theme(legend.position = "right")
```

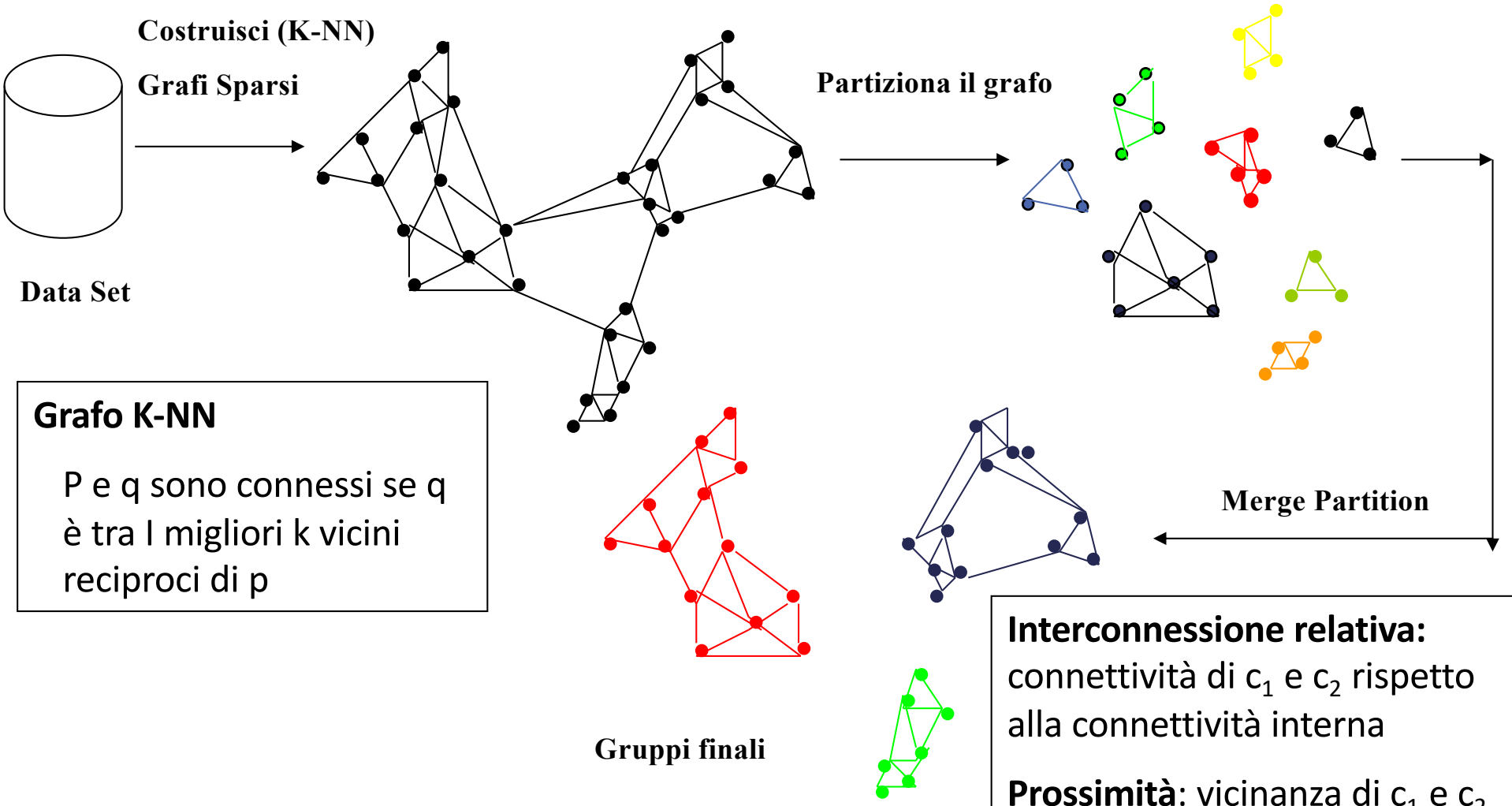
BIRCH (Balanced Iterative Reducing and Clustering Using Hierarchies)

- Zhang, Ramakrishnan e Livny, SIGMOD'96
- Costruire in modo incrementale un albero CF (Clustering Feature), una struttura gerarchica dei dati per il clustering multifase
  - Fase 1: analizzare l dati per creare un albero CF iniziale in memoria (una compressione multilivello dei dati che tenta di preservare la struttura di clustering intrinseca dei dati)
  - Fase 2: utilizzare un algoritmo di clustering arbitrario per raggruppare i nodi terminali dell'albero CF
- Scala linearmente: trova un buon raggruppamento con una singola scansione e migliora la qualità con alcune scansioni aggiuntive
- Debolezza: gestisce solo i dati numerici e sensibili all'ordine delle unità

CHAMELEON

- G. Karypis, E. H. Han, and V. Kumar, 1999
- Misura la somiglianza basata su un modello dinamico
  - Due cluster vengono uniti solo se la *interconnectivity* e *closeness (proximity)* tra due gruppi sono elevati rispetto all'interconnettività interna dei gruppi e la vicinanza degli elementi all'interno dei gruppi
- Basato su grafico ed è un algoritmo a due fasi
  1. Usa un algoritmo di partizionamento grafico: oggetti cluster in un numero elevato di sottogruppi relativamente piccoli
  2. Usa un algoritmo di clustering gerarchico agglomeratativo: trovare i gruppi reali combinando ripetutamente questi sottogruppi

CHAMELEON



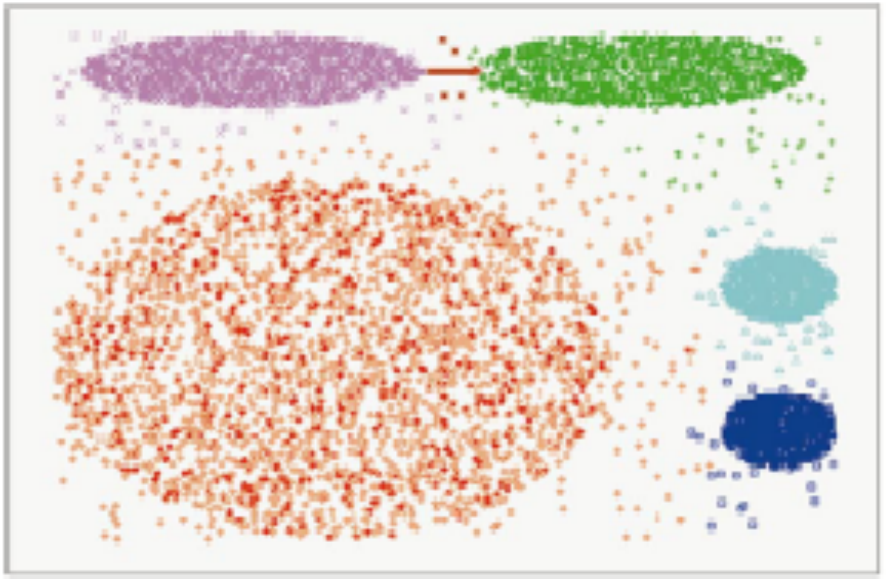
Grafo K-NN

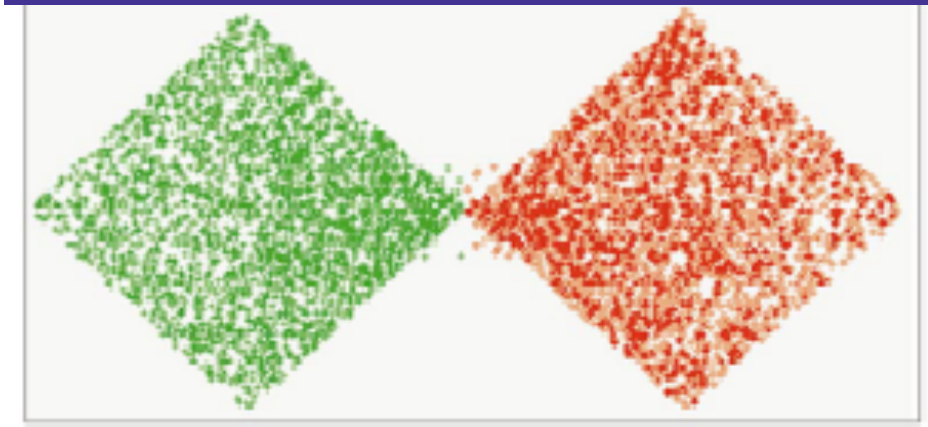
P e q sono connessi se q è tra I migliori k vicini reciproci di p

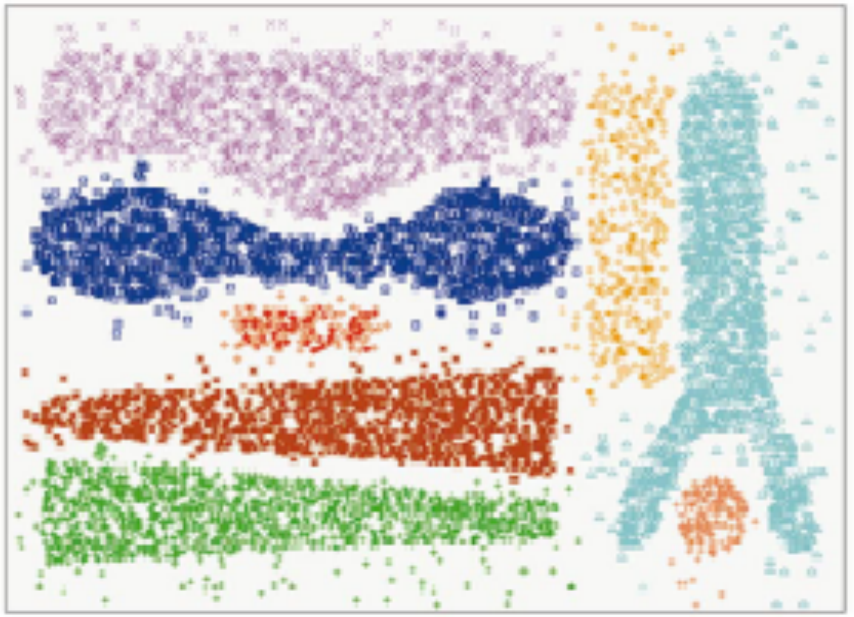
Interconnessione relativa: connettività di c_1 e c_2 rispetto alla connettività interna

Prossimità: vicinanza di c_1 e c_2 rispetto alla vicinanza interna

CHAMELEON







Clustering probabilistico gerarchico

- Algoritmo di clustering gerarchico
  - Non banale scegliere una buona misura della distanza
  - Difficile gestire i dati mancanti
  - Obiettivo di ottimizzazione non chiaro: euristica, ricerca locale
- Clustering gerarchico probabilistico
  - Utilizzare modelli probabilistici per misurare le distanze tra i gruppi
  - Modello generativo: considera l'insieme di unità da raggruppare come un campione del meccanismo di generazione dei dati sottostanti da analizzare
  - Facile da capire, stessa efficienza del metodo clustering agglomerativo algoritmico, in grado di gestire dati parzialmente osservati
- In pratica, supponiamo che i modelli generatori dei dati seguano funzioni di distribuzione comuni, ad esempio distribuzione gaussiana o distribuzione di Bernoulli, dipendenti parametri