



STATISTICA E ANALISI DEI DATI

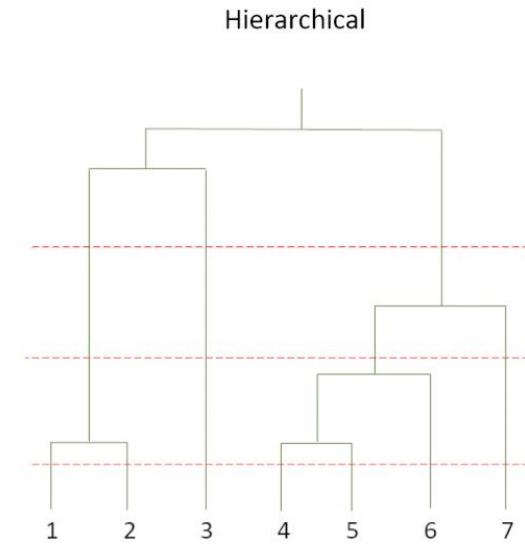
Capitolo 7 - Analisi dei cluster: parte 2

Dott. Stefano Cirillo
Dott. Luigi Di Biasi

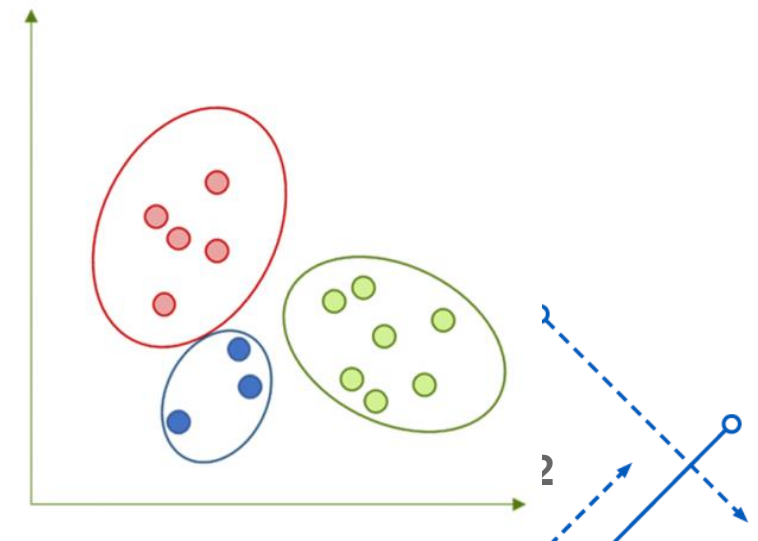
a.a. 2025-2026

Introduzione

- Il problema principale che si presenta utilizzando le tecniche di enumerazione completa è che esse sono computazionalmente onerose
 - Prevedono il calcolo della funzione obiettivo per ogni possibile partizione dell'insieme totale di n individui in m cluster
- Spesso si adottano i metodi di clustering gerarchici e non gerarchici che operano su una sottoclasse delle partizioni degli n individui in cluster

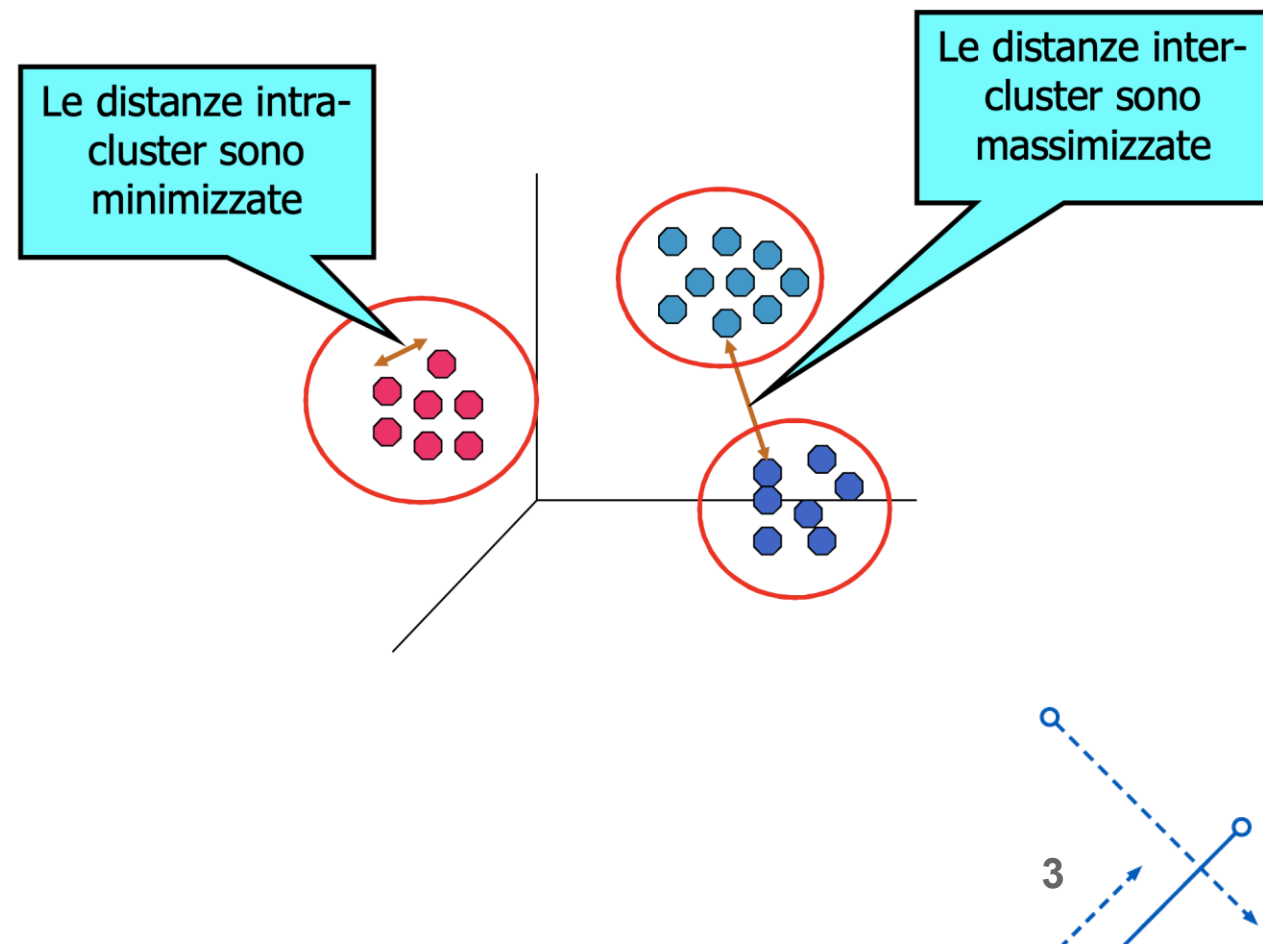


Non-hierarchical



Clustering Analysis

- Ricerca di gruppi di oggetti tali che gli oggetti appartenenti a un gruppo siano “simili” tra loro e differenti dagli oggetti negli altri gruppi
- Obiettivi del Clustering:
 - **Sintetizzare l'informazione:**
 - Ridurre la complessità dei dati trovando strutture sottostanti
 - **Segmentazione:**
 - Dividere un insieme di dati in gruppi significativi
 - **Identificare anomalie:**
 - Rilevare dati che non appartengono a nessun cluster (outliers)



Applicazioni della Clustering Analysis

- **Marketing e Segmentazione dei Clienti**

- Riconoscere gruppi di clienti con comportamenti o caratteristiche simili per creare campagne pubblicitarie mirate e personalizzate per ogni gruppo
- **Esempio:** Dividere clienti in cluster basati su frequenza di acquisti, preferenze di prodotto e budget di spesa

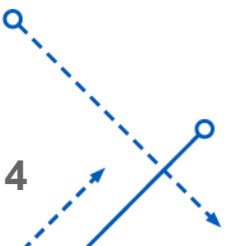
- **Biologia e Genetica**

- Raggruppare organismi o geni in base a somiglianze genetiche o fenotipiche per identificare specie simili, geni con funzioni correlate, e a costruire alberi evolutivi.
- **Esempio:** Clustering dei geni per studiare gruppi di malattie genetiche o classificazione delle specie tramite DNA

- **Motori di Ricerca e Analisi Web**

- Organizzare e categorizzare i contenuti in gruppi rilevanti per migliorare la rilevanza dei risultati di ricerca e personalizza l'esperienza dell'utente
- **Esempio:** Raggruppare i siti web per tema, parole chiave o comportamento di navigazione degli utenti

- ...



Cosa **non** è la Clustering analysis

- **Non è un'Analisi Supervisionata**

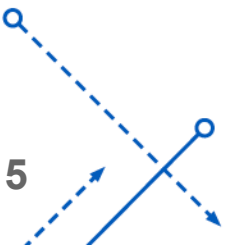
- La clustering analysis **non utilizza etichette predefinite** o classi target, come nelle tecniche di classificazione supervisionate
- Gli algoritmi di clustering cercano di individuare gruppi in modo autonomo, senza un'indicazione esterna su come dovrebbero essere formati i gruppi

- **Non è un Approccio di Classificazione**

- La classificazione assegna ogni istanza a una classe conosciuta, mentre il clustering identifica **gruppi naturali** senza conoscere in anticipo le classi
- Non può essere usata per prevedere categorie predefinite, ma solo per rilevare pattern di somiglianza all'interno dei dati

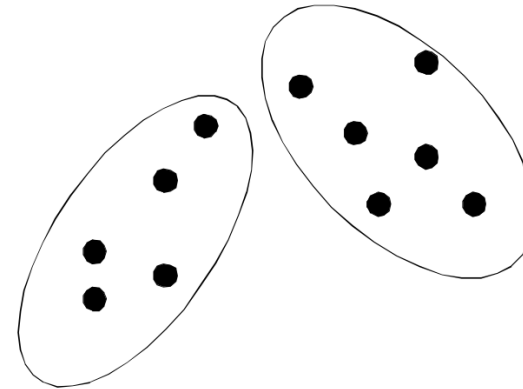
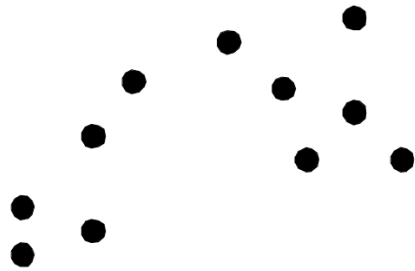
- **Non Fornisce un Unico Risultato "Corretto"**

- La clustering analysis può produrre **risultati diversi a seconda dell'algoritmo e dei parametri** scelti, come il numero di cluster
- Non esiste un'unica "risposta corretta" e la scelta dell'algoritmo e dei parametri deve essere guidata dagli obiettivi dell'analisi e dalla natura dei dati

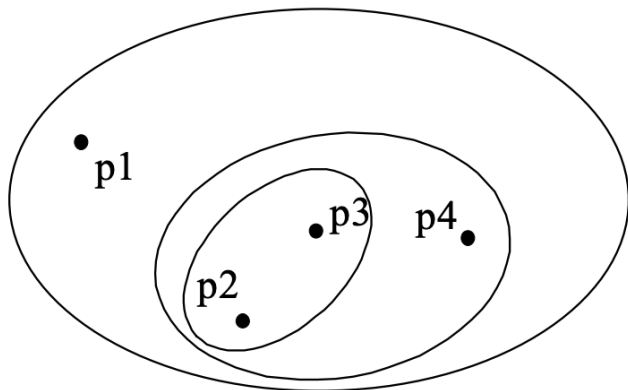


Tipi di Clustering

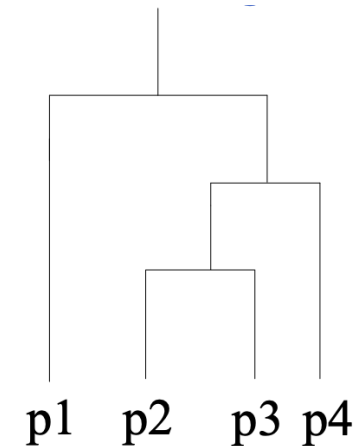
- Un clustering è un insieme di cluster che può essere:
 - **Clustering partizionante**: una divisione degli oggetti in sottoinsiemi (cluster) non sovrapposti. Ogni oggetto appartiene esattamente a un cluster



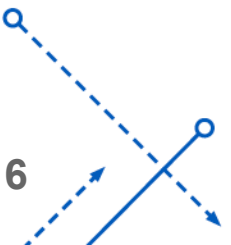
- **Clustering gerarchico**: un insieme di cluster annidati organizzati come un albero gerarchico



Clustering gerarchico tradizionale



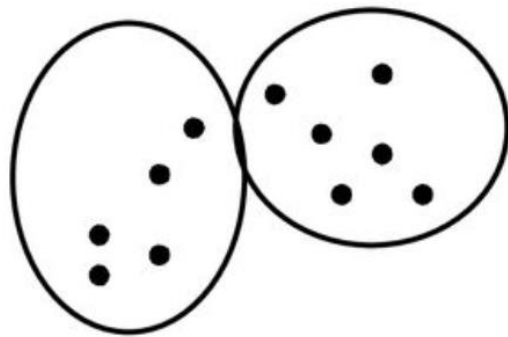
Dendrogramma



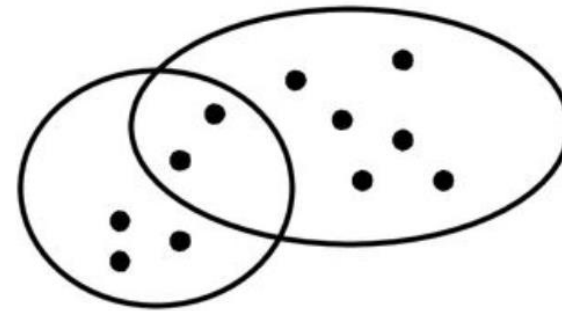
Altre Caratteristiche del Clustering

- **Esclusivo vs non esclusivo**

- In un clustering non esclusivo, i punti possono appartenere a più cluster
- Utile per rappresentare punti di confine o più tipi di classi



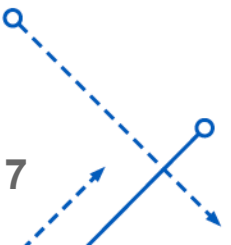
Exclusive
clustering



Non-exclusive
clustering

- **Esempio di Clustering Esclusivo:**

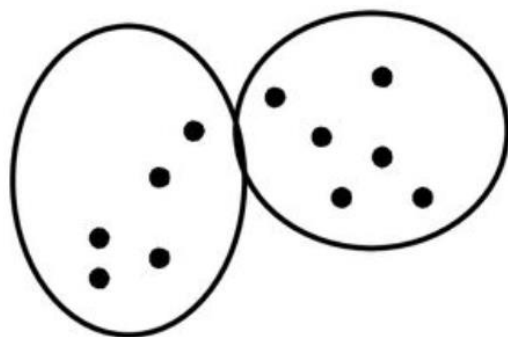
- Segmentazione di clienti in base all'età e al reddito per creare gruppi distinti (ad es., giovani adulti, famiglie, pensionati)
- Classificazione delle specie animali in gruppi distinti basati su tratti genetici



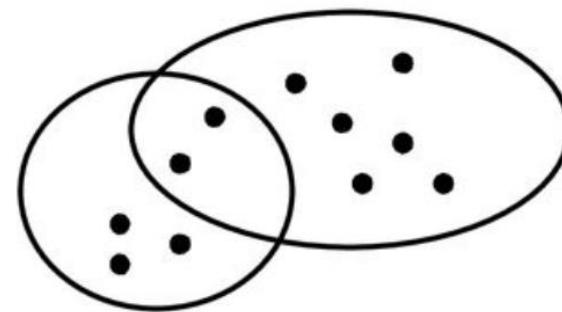
Altre Caratteristiche del Clustering

- **Esclusivo vs non esclusivo**

- In un clustering non esclusivo, i punti possono appartenere a più cluster
- Utile per rappresentare punti di confine o più tipi di classi



Exclusive
clustering



Non-exclusive
clustering

- **Esempio di Clustering Non-Esclusivo:**

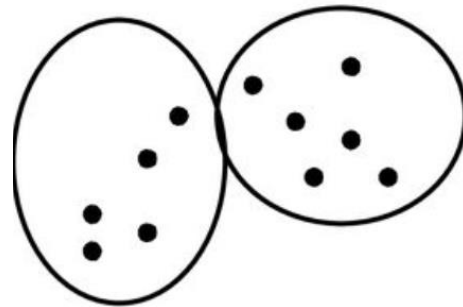
- Analisi dei profili di utenti di un social media, dove un utente può mostrare interessi in più categorie (es. tecnologia e sport)
- Segmentazione del mercato in base a comportamenti d'acquisto, in cui un cliente può appartenere parzialmente a più segmenti (es. shopping di moda e elettronica).



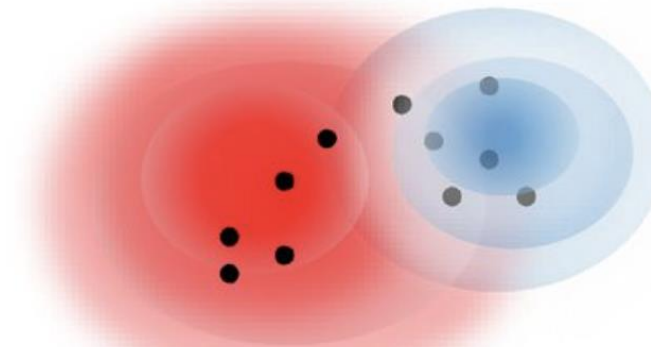
Altre Caratteristiche del Clustering

- **Fuzzy vs non-fuzzy**

- In un fuzzy clustering un punto appartiene a tutti i cluster con un peso tra 0 e 1.
- La somma dei pesi per ciascun punto deve essere 1



Non fuzzy
clustering



Fuzzy clustering

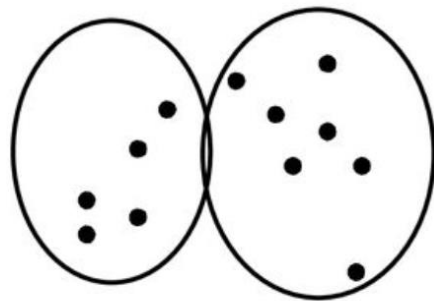
- **Esempio di Clustering Fuzzy:**

- Analisi dei profili di utenti di un social media, dove un utente può mostrare interessi in più categorie (es. tecnologia e sport)
- Classificazione delle recensioni di prodotti dove i commenti possono esprimere più sentimenti contemporaneamente (es. "soddisfatto ma con riserve").

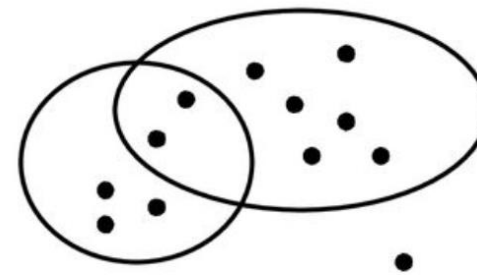
Altre Caratteristiche del Clustering

- **Parziale vs completo**

- In un clustering parziale alcuni punti potrebbero non appartenere a nessuno dei cluster
- Solo alcuni dati vengono inclusi nei cluster, lasciando fuori i dati che non si adattano bene a nessun gruppo (ad esempio, outlier o dati rumorosi).



Complete
clustering



Partial clustering

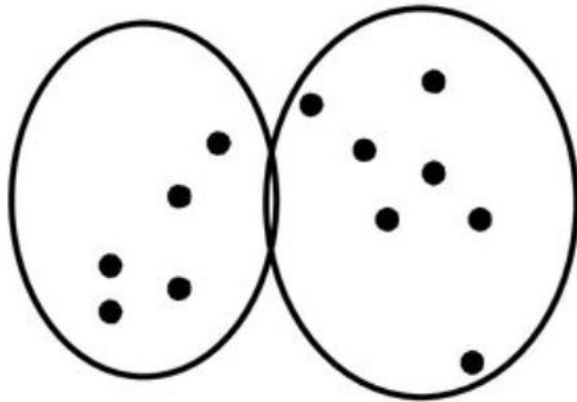
- **Esempio di Clustering Parziale:**

- Raggruppare utenti che interagiscono frequentemente su temi simili, escludendo gli utenti con poca attività o interessi troppo generici per essere inclusi in un cluster
- Identificare gruppi di utenti attivi e coerenti, utili per le campagne mirate o per lo studio delle dinamiche sociali

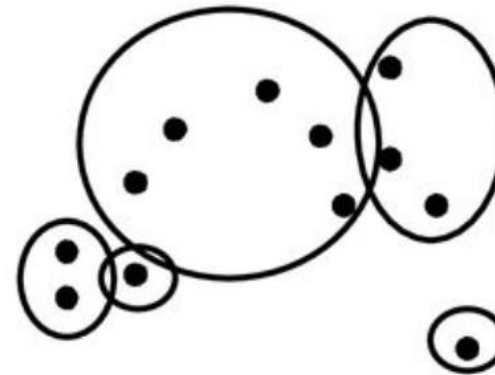
Altre Caratteristiche del Clustering

- **Eterogeneo vs omogeneo**

- In un cluster eterogeneo i cluster possono avere dimensioni, forme e densità molto diverse



Homogeneous



Heterogeneous
(in terms of size)

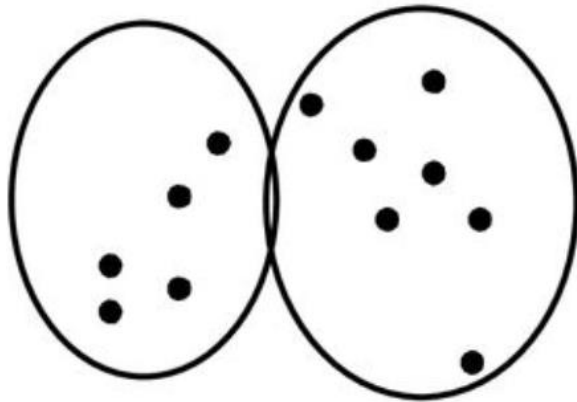
- **Esempio di Clustering Eterogeneo:**

- Raggruppare consumatori in cluster eterogenei basati su comportamenti d'acquisto variegati.
- **Esempio:** Suddividere i clienti in gruppi che comprano sia generi alimentari che prodotti di lusso, mostrando preferenze eterogenee.

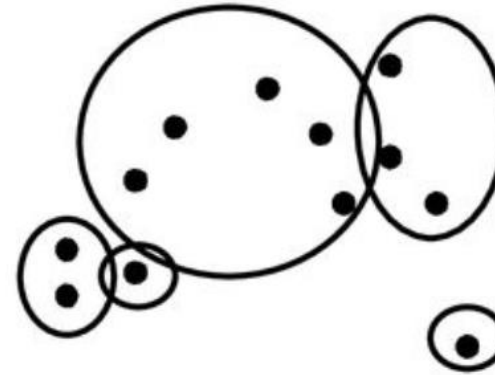
Altre Caratteristiche del Clustering

- **Eterogeneo vs omogeneo**

- In un cluster eterogeneo i cluster possono avere dimensioni, forme e densità molto diverse



Homogeneous



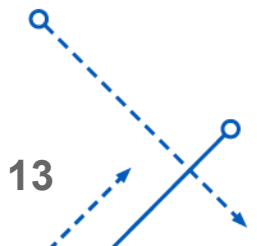
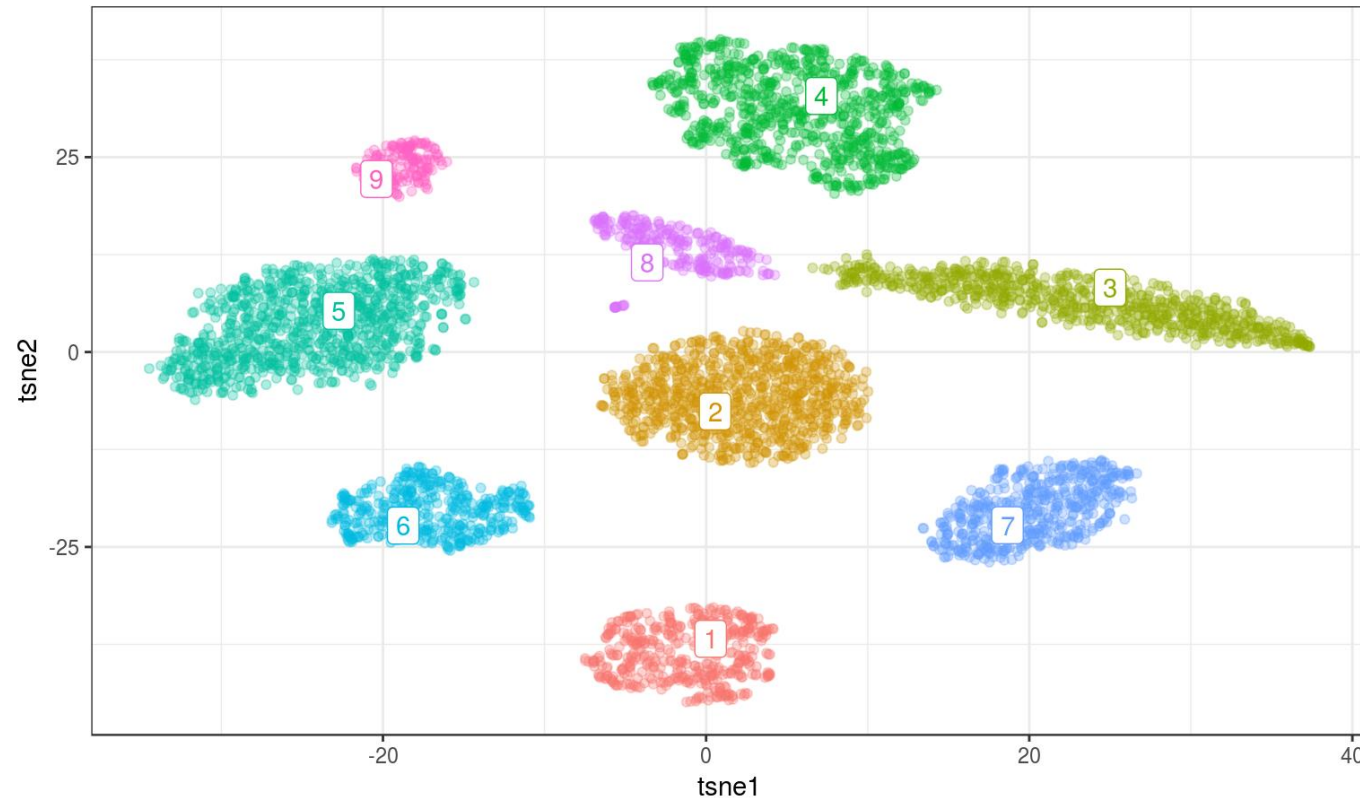
Heterogeneous
(in terms of size)

- **Esempio di Clustering Omogeneo:**

- **Clustering di Prodotti Simili:** Creazione di cluster di prodotti in una categoria omogenea (es. elettronica) in base a caratteristiche simili
- **Esempio:** Raggruppare modelli di smartphone in base a specifiche tecniche come memoria e processore

Altre Caratteristiche del Clustering

- La dimensione, la forma e la densità dei cluster sono caratteristiche fondamentali nell'analisi dei dati e nell'identificazione di gruppi omogenei all'interno di un dataset
- **Dimensioni dei cluster:**
 - La dimensione di un cluster è **determinata dal numero di punti che lo compongono**
 - Cluster grandi possono indicare gruppi di dati con molte osservazioni simili, mentre cluster piccoli possono rappresentare gruppi più specifici o anomalie



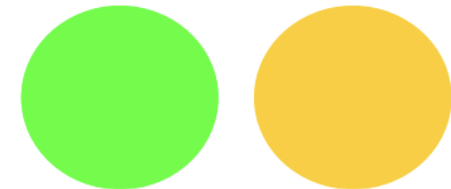
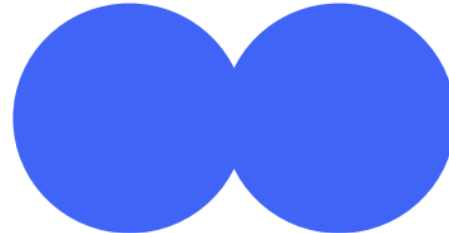
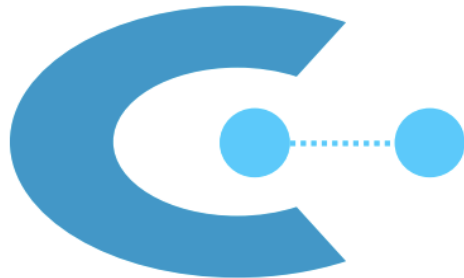
Altre Caratteristiche del Clustering

- **Forme dei cluster:**

- I cluster possono avere forme diverse a seconda della distribuzione dei dati e del metodo di clustering
 - I metodi come K-means tendono a generare cluster di forma **sferica**, poiché minimizzano la distanza euclidea tra i punti all'interno di ogni cluster.
- Algoritmi più avanzati, come DBSCAN o Mean Shift tendono ad individuare cluster di forma arbitraria, come quelli di forma allungata o irregolare, perché non si basano esclusivamente sulla distanza euclidea

- **Densità dei cluster:**

- La densità di un cluster si riferisce alla **concentrazione dei punti** all'interno del cluster stesso
 - Cluster densi indicano regioni del dataset in cui i punti sono più vicini tra loro, mentre cluster a bassa densità possono rappresentare aree con meno omogeneità.





STATISTICA E ANALISI DEI DATI

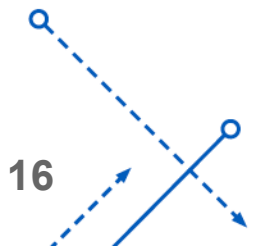
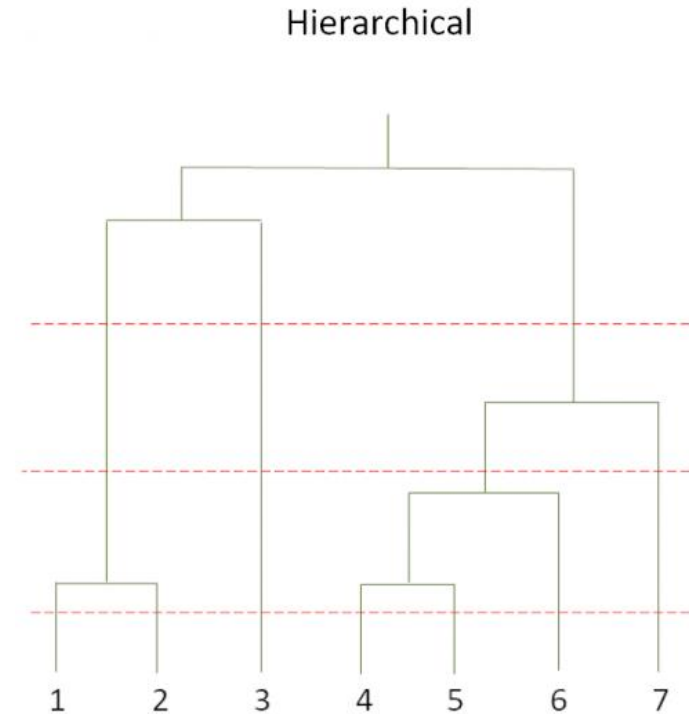
Capitolo 7 – Clustering Gerarchico

Dott. Stefano Cirillo
Dott. Luigi Di Biasi

a.a. 2025-2026

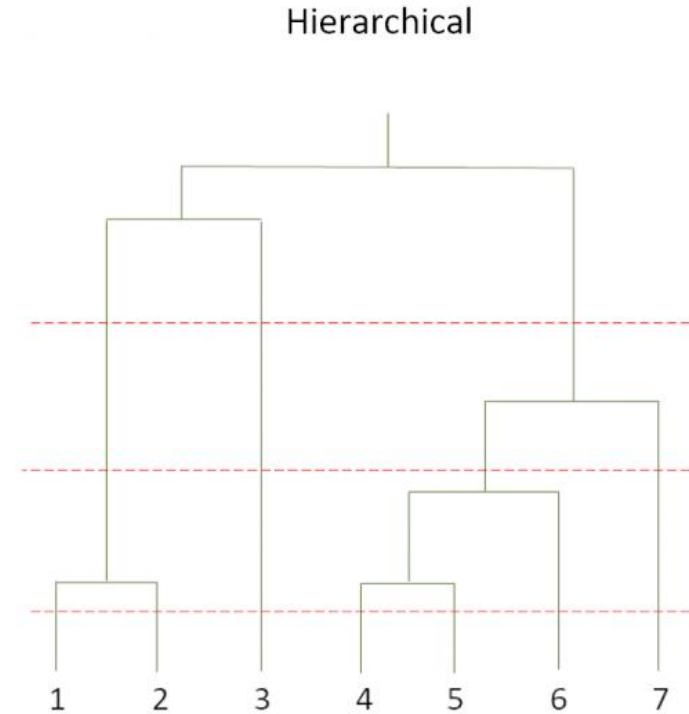
Clustering Gerarchica

- I **metodi gerarchici di clustering** eseguono una sequenza ordinata di operazioni della stessa natura
- **Vantaggi:**
 - quello di fornire una visione completa dell'insieme in termini di distanza o similarità
 - non comportare né la scelta a priori del numero di cluster e ne la scelta a priori di parametri per la determinazione automatica del loro numero
- **Svantaggio:**
 - non è possibile riallocare gli individui che sono stati già classificati ad un livello precedente dell'analisi



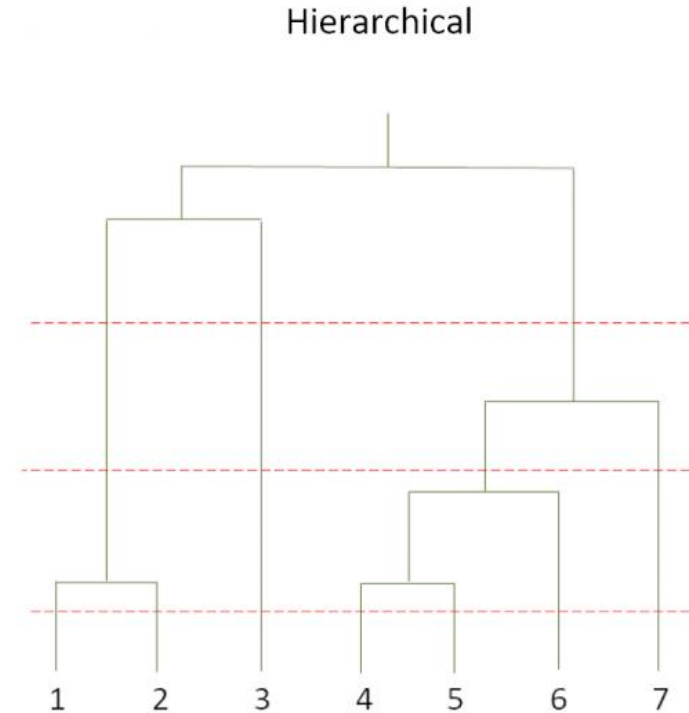
Dendrogramma

- L'obiettivo finale dei metodi gerarchici è quello di ottenere una sequenza di partizioni che possono essere rappresentate graficamente mediante una struttura ad albero: **Dendrogramma**
 - Sull'insieme delle ordinate sono riportati i livelli di distanza
 - Sull'asse delle ascisse sono riportati i singoli individui
- Il dendrogramma fornisce un quadro completo della struttura dell'insieme in termini delle misure di distanza tra gli individui



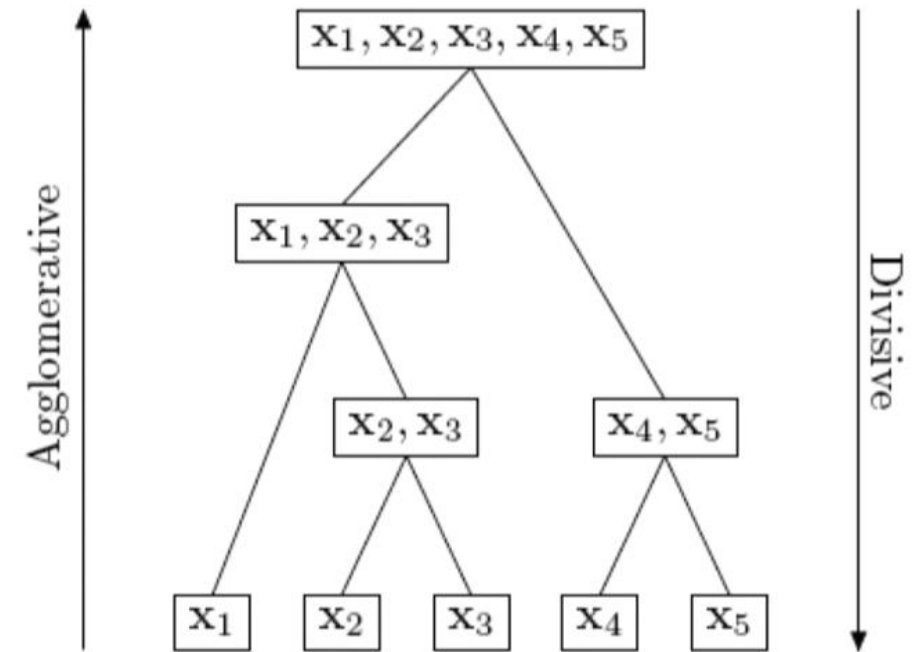
Dendrogramma

- L'obiettivo finale dei metodi gerarchici è quello di ottenere una sequenza di partizioni che possono essere rappresentate graficamente mediante una struttura ad albero: **Dendrogramma**
 - Sull'insieme delle ordinate sono riportati i livelli di distanza
 - Sull'asse delle ascisse sono riportati i singoli individui
- Ad ogni livello di distanza corrisponde una partizione,
- Ad ogni partizione corrispondono infiniti livelli di distanza compresi tra quelli che individuano due successive unioni o divisioni.



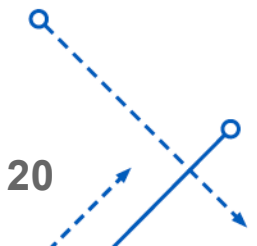
Clustering Gerarchica

- Possono essere
 - Agglomerativi: Fusioni a due a due di gruppi già formati
 - In fase iniziale ogni entità forma un gruppo e, come ultimo passo, tutte le unità sono in un gruppo unico
 - Divisivi: Divisioni a due a due di gruppi già formati
 - In fase iniziale tutte le entità sono in un unico gruppo e, come ultimo passo, le unità formano ciascuna un gruppo separato



Clustering Gerarchica Aggregativa

- Le tecniche gerarchiche di tipo **agglomerativo** si sono rivelate particolarmente utili in biologia e zoologia per raggruppare piante e animali rispetto a caratteristiche di tipo genetico
- Si fondono i due gruppi più prossimi in base all'indice di distanza o dissimilarità (detto **legame**) e si prosegue fino a formare un gruppo unico
 - Ad ogni passo della clustering gerarchica agglomerativa si fondono i due gruppi più prossimi secondo il legame che caratterizza l'algoritmo
 - Il legame stabilisce come giudicare la dissomiglianza tra i gruppi ovvero distanza cluster-to-cluster



Clustering Gerarchica Aggregativa

- Sia $I = \{I_1, I_2, \dots, I_n\}$ un insieme di n individui o entità appartenenti ad una popolazione
- Nei metodi gerarchici di clustering di tipo agglomerativo si valuta inizialmente la matrice delle distanze D tra gli individui
 - A partire da una di tali matrici, si considera inizialmente un insieme di n cluster $\{I_1\}, \{I_2\}, \dots, \{I_n\}$
 - I due cluster più vicini I_i e I_j sono **uniti** in un singolo cluster e quindi l'insieme di partenza sarà costituito da un cluster in meno:
$$\{I_1\}, \{I_2\}, \dots, \{I_i, I_j\}, \dots, \{I_n\}$$
 - Ripetendo tale procedura sequenzialmente si ottengono insiemi di $n - 2$ cluster, $n - 3$, etc. fino a che si ottiene un **unico** cluster di n individui (insieme originario di tutti gli I)

$$D = \begin{pmatrix} 0 & d_{12} & \dots & d_{1n} \\ d_{21} & 0 & \dots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \dots & 0 \end{pmatrix}$$



Clustering Gerarchica Aggregativa

- Algoritmo:

- **Step 1:** A partire dalla matrice X originaria dei dati, considerare la **matrice delle distanze** D (o la matrice di similarità S) tra gli individui (considerati come singoli cluster contenenti un solo individuo);
- **Step 2:** individuare la **coppia di cluster meno distanti** (o più somiglianti) e **raggruppare** in un unico cluster i **due cluster meno distanti** (o più somiglianti); inoltre, calcolare la distanza (o similarità) di questo nuovo cluster, originato dall'agglomerazione, da tutti gli altri gruppi già esistenti;
- **Step 3:** costruire una nuova matrice di distanza (o di similarità) che risulterà **ridotta di una riga e di una colonna** rispetto a quella che la precede; infatti, le due righe e due colonne dei gruppi agglomerati sono sostituite con una singola riga e una singola colonna contenenti le nuove distanze;



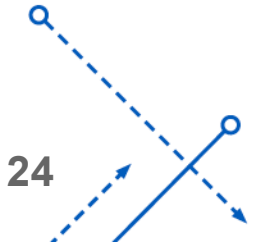
Clustering Gerarchica Aggregativa

- **Step 4:** operare sulla matrice così ottenuta a partire dagli step precedenti fino ad esaurire tutte le possibilità di raggruppamento, raggiungendo alla fine una matrice di cardinalità 2×2 . La procedura richiede $n - 1$ iterazioni
- **Step 5:** **rappresentare** graficamente il processo di agglomerazione attraverso un dendrogramma che riporta sull'asse verticale il livello di distanza a cui avviene l'agglomerazione e sull'asse orizzontale riporta gli individui. Ad ogni livello di distanza corrisponde una partizione.



Clustering Gerarchica Aggregativa

- **Step 1:** A partire dalla matrice X originaria dei dati, considerare la matrice delle distanze D (o la matrice di similarità S) tra gli individui (considerati come singoli cluster contenenti un solo individuo);
- **Step 2:** individuare la coppia di cluster meno distanti (o più somiglianti) e raggruppare in un unico cluster i due cluster meno distanti (o più somiglianti); inoltre, calcolare la distanza (o similarità) di questo nuovo cluster, originato dall'agglomerazione, da tutti gli altri gruppi già esistenti;
- Le differenze esistenti tra i vari metodi si riscontrano nel passo 1 e nel passo 2
 - Passo 1: la scelta della misura di distanza (o di misura di similarità) influenza il metodo richiedendo più o meno forti proprietà
 - Passo 2: è quello che caratterizza ciascun metodo il modo in cui si individuano i due cluster meno distanti (o più somiglianti) e per il modo in cui si determina la distanza (o similarità) che intercorre tra il nuovo cluster ottenuto e i rimanenti
- **Nota:** Il procedimento del passo 2 è specifico per ciascun metodo e influenza la sua denominazione

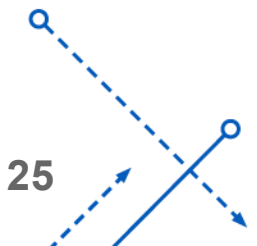


Clustering Gerarchica Aggregativa (R)

- L'analisi gerarchica di tipo agglomerativo viene effettuata in R attraverso la funzione:

hclust(d, method = "complete")

- La funzione hclust() produce come output una lista, i cui elementi sono:
 - Matrice di dimensione $(n-1) \times 2$ le cui righe descrivono le aggregazioni avvenute a ciascun passo dell'intero procedimento. Gli elementi negativi indicano singole unità, mentre quelli positivi indicano gruppi già formati (**\$merge**);
 - Un vettore, la cui lunghezza corrisponde al numero di iterazioni, che indica il livello di distanza alla quale è avvenuta l'unione tra due cluster (**\$height**);
 - Un vettore che fornisce la permutazione delle osservazioni originali da utilizzare per il grafico (**\$order**);
 - Un vettore delle etichette che contrassegnano le varie unità (**\$labels**)



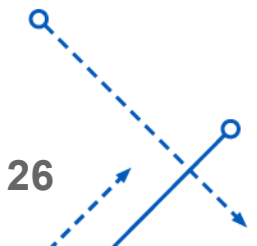
Clustering Gerarchica Aggregativa (R)

- L'analisi gerarchica di tipo agglomerativo viene effettuata in R attraverso la funzione:

hclust(d, method = "complete")

dove

- **d** rappresenta un oggetto (che individua una struttura di similarità o distanza) creato tramite la funzione `dist()` (Distance Matrix Computation)
- **method** seleziona il metodo gerarchico agglomerativo (di default è **complete**)
- Alcune delle opzioni disponibili per `method` sono:
 - (1) Metodo del legame singolo (single);
 - (2) Metodo del legame completo (complete);
 - (3) Metodo del legame medio (average);
 - (4) Metodo del centroide (centroid);
 - (5) Metodo della mediana (median).



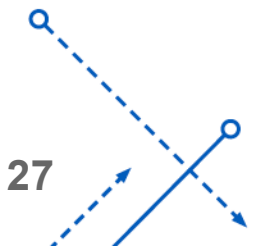
Disegnare il Dendrogramma (R)

- Per ottenere il dendrogramma si impiega la funzione:

plot(z, labels = NULL, hang = -1, main = "Dendrogramma", sub = NULL, xlab = NULL)

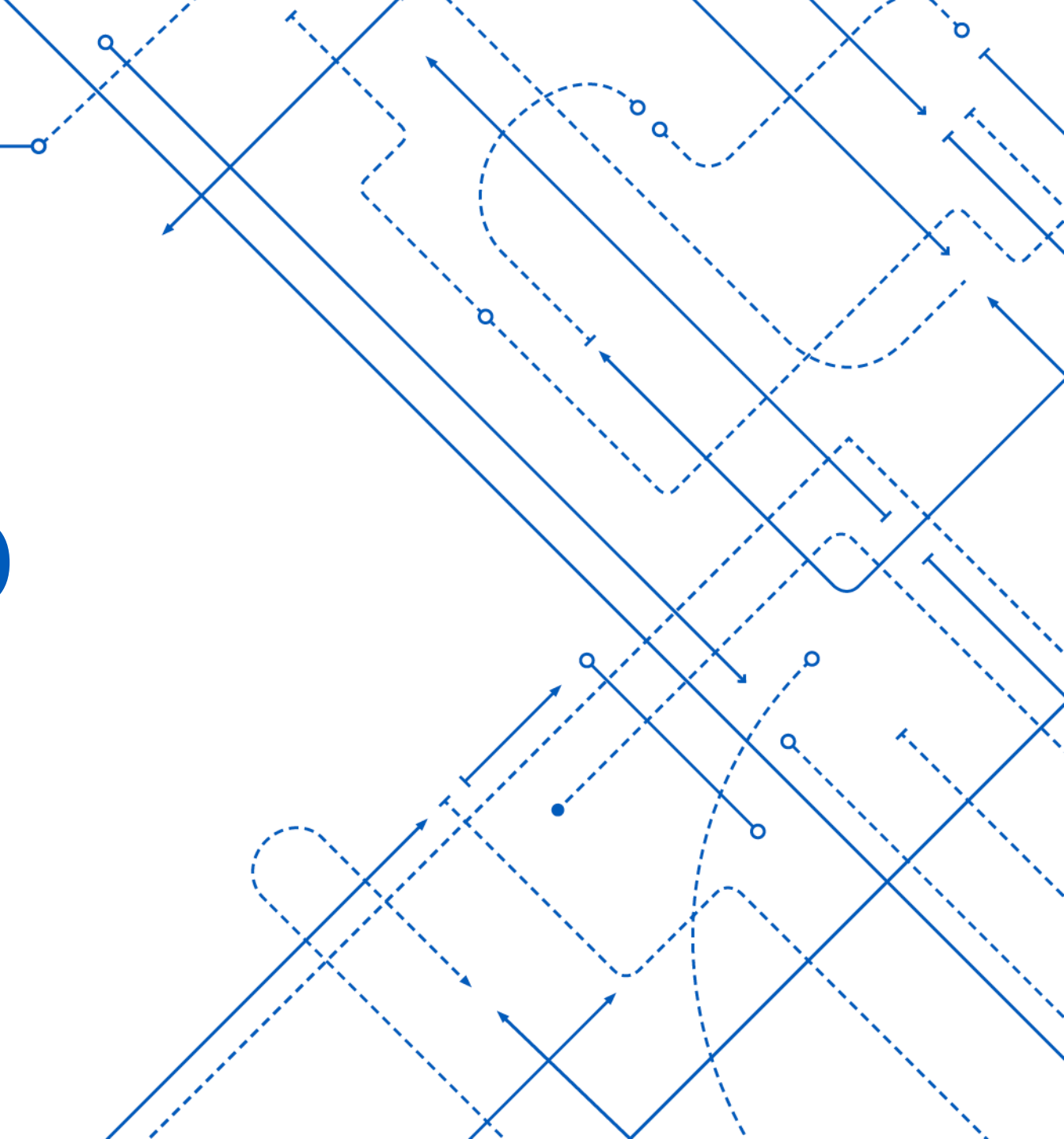
dove

- **z** è l'oggetto creato (output) dalla funzione **hclust()**;
- **labels** è un vettore di etichette per i rami del dendrogramma (di default impiega i nomi delle righe del data frame);
- **hang** determina l'altezza alla quale le etichette vengono visualizzate al di sotto del dendrogramma (un valore negativo pone le etichette al di sotto dell'ordinata nulla);
- **main**, **sub**, **xlab** sono comandi per la finestra grafica



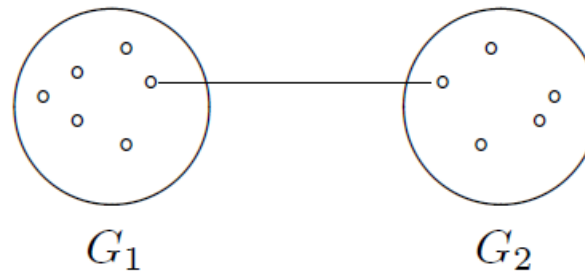
CLUSTERING GERARCHICO

Metodo del Legame Singolo



Metodo del Legame Singolo

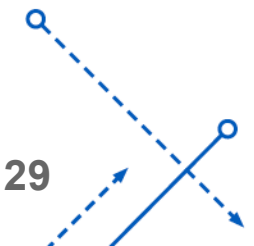
- Nel **Metodo del legame singolo** o Nearest neighbour method la distanza tra i gruppi G_1 (contenente n_1 individui) e G_2 (contenente n_2 individui) è definita come la **minima** tra tutte le distanze tra $n_1 n_2$ che si possono calcolare tra ogni individuo di G_1 e ogni individuo di G_2



- Nella procedura gerarchica si considera inizialmente, ossia al livello 0, un insieme di n cluster

$$\{I_1\}, \{I_2\}, \dots, \{I_n\}$$

- Al passo successivo si cerca nella matrice D delle distanze il **coefficiente di distanza minima** e si raggruppano nello stesso cluster G_{ij} i due individui I_i e I_j associati secondo tale coefficiente
- Nel caso i coefficienti di distanza minima siano più di uno, si attua una scelta arbitraria tra di essi.



Metodo del Legame Singolo

- Successivamente al livello 1 quindi si modifica la matrice delle distanze valutando le distanze di G_{ij} da ogni altro individuo I_k non appartenente a G_{ij} mediante la seguente relazione:

$$d_{(ij),k} = \min(d_{ij}, d_{jk})$$

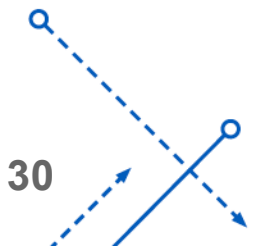
Cioè la distanza dell'individuo I_k dal cluster G_{ij} si ottiene scegliendo la più **piccola distanza** tra d_{ij} e d_{jk}

- Quindi, al livello 1 si costruisce una nuova matrice D_1 di cardinalità $(n-1) \times (n-1)$ costituita da G_{ij} (che viene considerato come un unico elemento) e dai restanti $(n-2)$ individui fuori dal cluster G_{ij}
- Ad ogni passo successivo, dopo che i cluster G_u e G_v sono stati uniti scegliendo dalla precedente matrice delle distanze i due cluster più vicini, la distanza tra il nuovo cluster, denotato con G_{uv} , e un altro cluster G_z è così definita:

$$d_{(uv),z} = \min(d_{uz}, d_{vz})$$

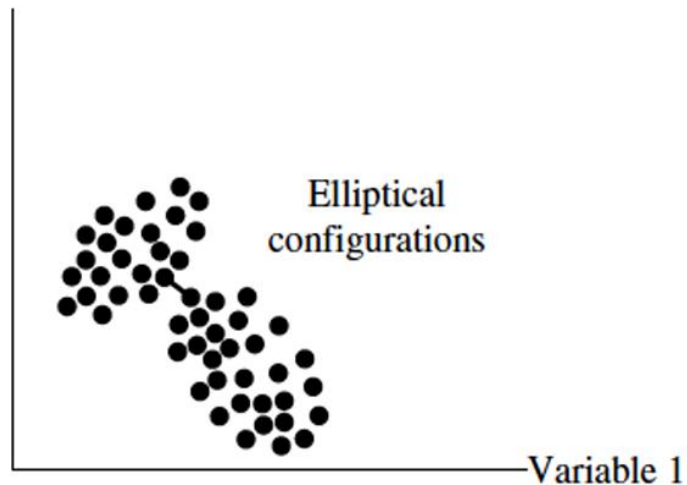
Cioè $d_{(uv),z}$ rappresenta la misura di distanza tra gli elementi meno distanti dei cluster G_{uv} e G_z

- La procedura si ripete fino ad ottenere un unico cluster formato da tutti gli individui

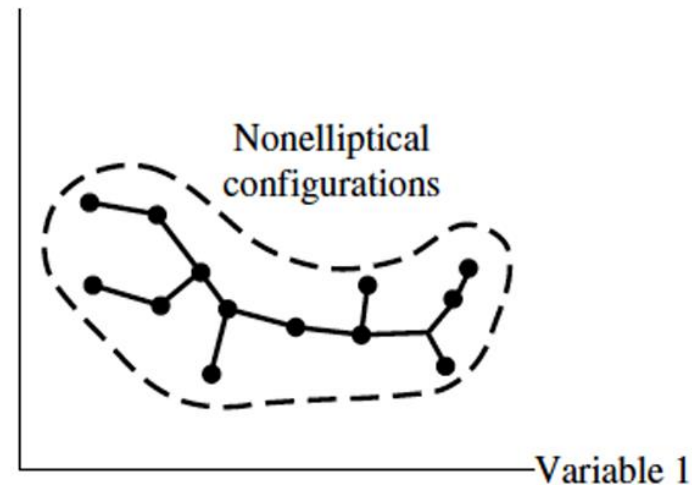


Effetto Catena (Chaining)

- Una peculiarità del legame singolo è l'**effetto catena (chaining)**
 - Da un lato consente di cogliere gruppi di forma particolare (Figura(b))
 - dall'altro rischia di legare osservazioni che non appartengono a uno stesso gruppo (Figura(a))
- L'**effetto chaining** (o **effetto catena**) nel clustering gerarchico con **legame singolo** si riferisce alla tendenza di collegare punti in cluster allungati attraverso una "catena" di collegamenti ravvicinati
 - Questo effetto crea un **cluster unico e allungato** invece di mantenere gruppi ben separati, anche se i dati potrebbero effettivamente contenere cluster distinti



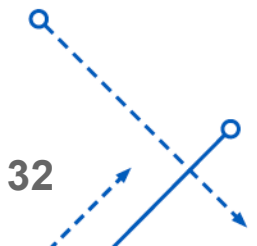
(a) Single linkage confused by near overlap



(b) Chaining effect

Effetto Catena (Chaining)

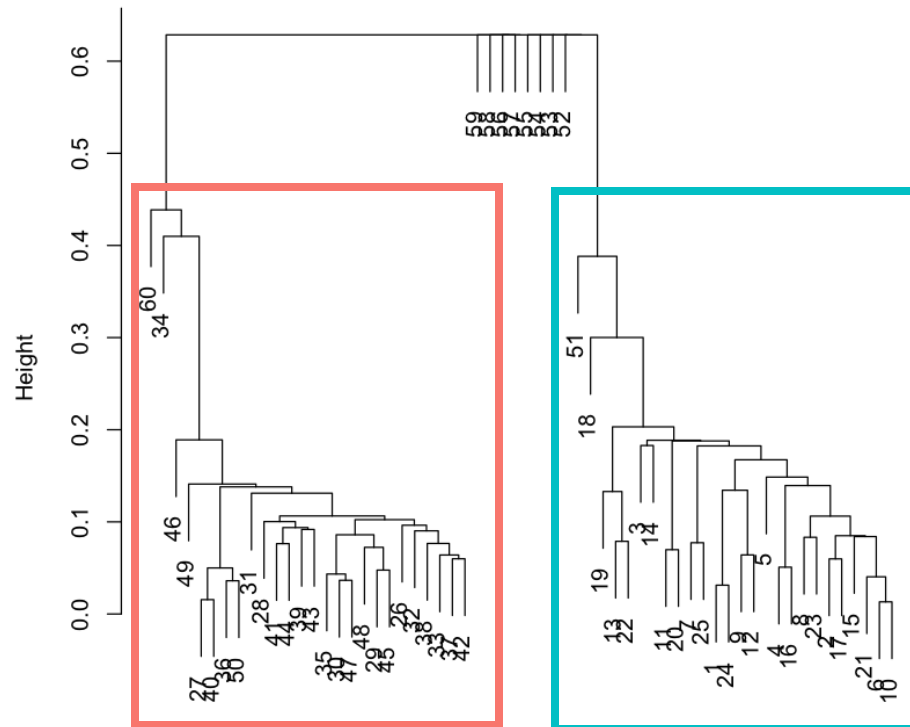
- Quando utilizziamo il metodo del legame singolo, il criterio per unire due cluster è la **distanza minima** tra qualsiasi coppia di punti appartenenti ai due cluster
 - Questo significa che basta trovare un singolo punto in ciascun cluster che sia sufficientemente vicino all'altro per unire i cluster
- **Esempio:**
 - Immaginiamo di avere due gruppi di persone (**cluster**) che si trovano su due lati di un fiume, ma c'è una serie di ponti molto piccoli (**i punti intermedi**) che li collega.
 - Anche se le due aree principali sono lontane tra loro, i ponti creano una connessione.
 - Nel clustering con legame singolo, i due gruppi saranno considerati parte dello stesso "cluster" a causa di quei ponti



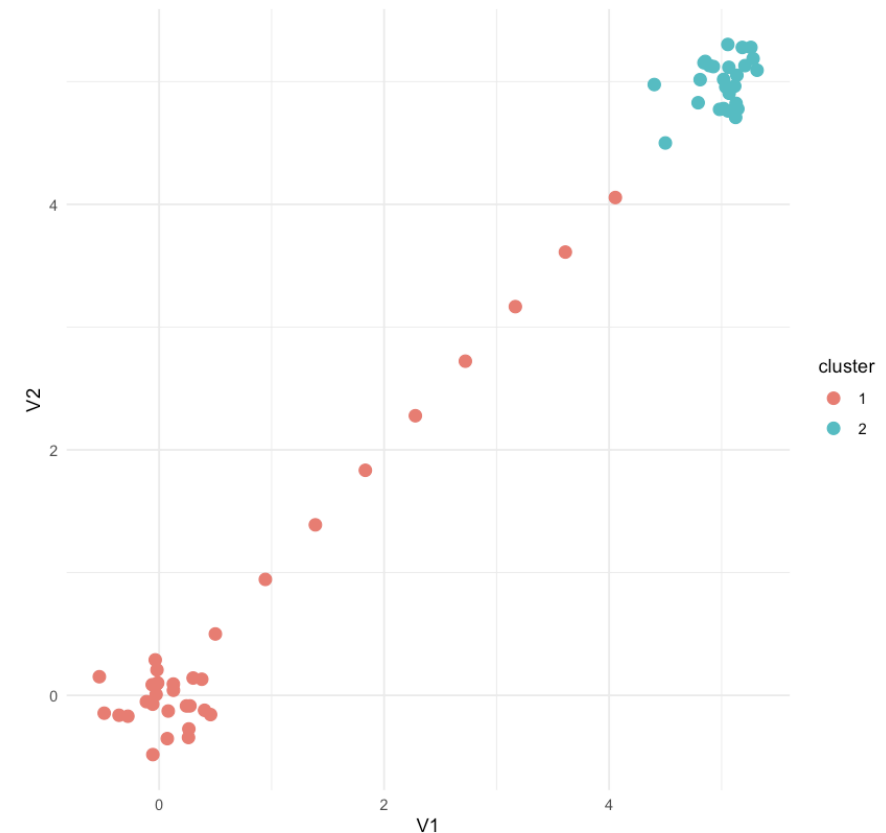
Effetto Catena (Chaining)

- Quando utilizziamo il metodo del legame singolo, il criterio per unire due cluster è la **distanza minima** tra qualsiasi coppia di punti appartenenti ai due cluster
 - Questo significa che basta trovare un singolo punto in ciascun cluster che sia sufficientemente vicino all'altro per unire i cluster

Dendrogramma - Effetto Chaining con Legame Singolo



Effetto Chaining nel Clustering con Legame Singolo

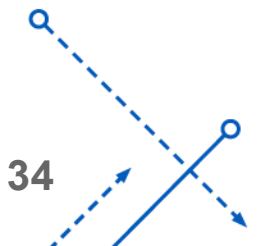


Esempio in R – Legame Singolo

- Consideriamo la seguente matrice contenente due caratteristiche C_1 e C_2 osservate per 5 individui I_1, I_2, I_3, I_4, I_5

```
> X<-data.frame(c1=c(1,1,6,8,8),c2=c(1,2,3,2,0))
> row.names(X)<-c("I1","I2","I3","I4","I5")
> X # visualizza il data frame X
  c1 c2
I1  1  1
I2  1  2
I3  6  3
I4  8  2
I5  8  0
>
> Z<-scale(X)
> Z # visualizza la matrice scalata
      c1      c2
I1 -1.0663057 -0.5262348
I2 -1.0663057  0.3508232
I3  0.3367281  1.2278812
I4  0.8979417  0.3508232
I5  0.8979417 -1.4032928
attr(,"scaled:center")
  c1  c2
4.8 1.6
attr(,"scaled:scale")
  c1      c2
3.563706 1.140175
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$



Esempio in R – Legame Singolo

- Consideriamo la seguente matrice contenente due caratteristiche C_1 e C_2 osservate per 5 individui I_1, I_2, I_3, I_4, I_5

```
> X<-data.frame(c1=c(1,1,6,8,8),c2=c(1,2,3,2,0))
> row.names(X)<-c("I1","I2","I3","I4","I5")
> X # visualizza il data frame X
```

	c1	c2
I1	1	1
I2	1	2
I3	6	3
I4	8	2
I5	8	0

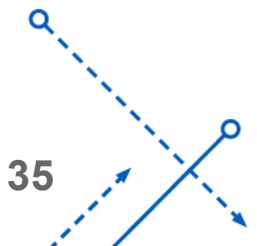
```
>
> Z<-scale(X)
> Z # visualizza la matrice scalata
```

	c1	c2
I1	-1.0663057	-0.5262348
I2	-1.0663057	0.3508232
I3	0.3367281	1.2278812
I4	0.8979417	0.3508232
I5	0.8979417	-1.4032928

```
attr(,"scaled:center")
  c1  c2
4.8 1.6
attr(,"scaled:scale")
  c1  c2
3.563706 1.140175
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

La funzione `scale()` calcola la media e la deviazione standard dell'intero vettore e 'scala' ogni elemento sottraendo ad ogni valore la media e dividendo per la deviazione standard



Esempio in R – Legame Singolo

- Consideriamo la seguente matrice contenente due caratteristiche C_1 e C_2 osservate per 5 individui I_1, I_2, I_3, I_4, I_5

```
> X<-data.frame(c1=c(1,1,6,8,8),c2=c(1,2,3,2,0))
> row.names(X)<-c("I1","I2","I3","I4","I5")
> X # visualizza il data frame X
```

	c1	c2
I1	1	1
I2	1	2
I3	6	3
I4	8	2
I5	8	0

```
>
> Z<-scale(X)
> Z # visualizza la matrice scalata
```

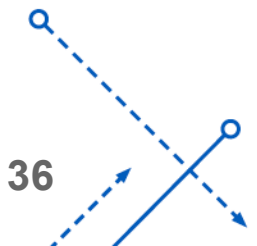
	c1	c2
I1	-1.0663057	-0.5262348
I2	-1.0663057	0.3508232
I3	0.3367281	1.2278812
I4	0.8979417	0.3508232
I5	0.8979417	-1.4032928

```
attr(,"scaled:center")
  c1  c2
4.8 1.6
attr(,"scaled:scale")
  c1  c2
3.563706 1.140175
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

La funzione `scale()` calcola la media e la deviazione standard dell'intero vettore e 'scala' ogni elemento sottraendo ad ogni valore la media e dividendo per la deviazione standard

`center` fornisce le **medie campionarie** relative alle due caratteristiche della matrice iniziale dei dati



Esempio in R – Legame Singolo

- Consideriamo la seguente matrice contenente due caratteristiche C_1 e C_2 osservate per 5 individui I_1, I_2, I_3, I_4, I_5

```
> X<-data.frame(c1=c(1,1,6,8,8),c2=c(1,2,3,2,0))
> row.names(X)<-c("I1","I2","I3","I4","I5")
> X # visualizza il data frame X
```

	c1	c2
I1	1	1
I2	1	2
I3	6	3
I4	8	2
I5	8	0

```
>
> Z<-scale(X)
> Z # visualizza la matrice scalata
```

	c1	c2
I1	-1.0663057	-0.5262348
I2	-1.0663057	0.3508232
I3	0.3367281	1.2278812
I4	0.8979417	0.3508232
I5	0.8979417	-1.4032928

```
attr(,"scaled:center")
```

	c1	c2
4.8	1.6	

```
attr(,"scaled:scale")
```

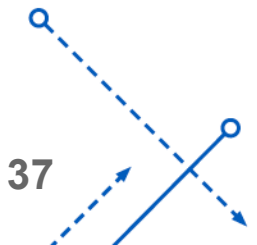
	c1	c2
3.563706	1.140175	

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

La funzione *scale()* calcola la media e la deviazione standard dell'intero vettore e 'scala' ogni elemento sottraendo ad ogni valore la media e dividendo per la deviazione standard

center fornisce le **medie campionarie** relative alle due caratteristiche della matrice iniziale dei dati

scale fornisce le **deviazioni standard campionarie**

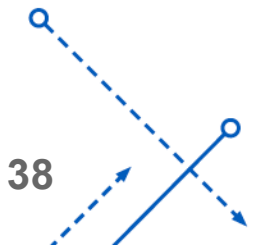


Esempio in R – Legame Singolo

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame singolo

```
> d<-dist(Z,method="euclidean",diag=TRUE,upper=TRUE)
> d # visualizza la matrice delle distanze
      I1      I2      I3      I4      I5
I1 0.000000 0.877058 2.246203 2.151162 2.151162
I2 0.877058 0.000000 1.654610 1.964247 2.633475
I3 2.246203 1.654610 0.000000 1.041245 2.690360
I4 2.151162 1.964247 1.041245 0.000000 1.754116
I5 2.151162 2.633475 2.690360 1.754116 0.000000
>
> hls<-hclust(d,method="single")
>
> str(hls) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int  [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num  [1:4] 0.877 1.041 1.655 1.754
 $ order      : int  [1:5] 5 1 2 3 4
 $ labels     : chr  [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr  "single"
 $ call       : language hclust(d = d, method = "single")
 $ dist.method: chr  "euclidean"
- attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$



Esempio in R – Legame Singolo

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame singolo

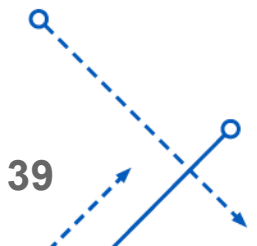
```
> d<-dist(Z,method="euclidean",diag=TRUE,upper=TRUE)
> d # visualizza la matrice delle distanze
      I1      I2      I3      I4      I5
I1 0.000000 0.877058 2.246203 2.151162 2.151162
I2 0.877058 0.000000 1.654610 1.964247 2.633475
I3 2.246203 1.654610 0.000000 1.041245 2.690360
I4 2.151162 1.964247 1.041245 0.000000 1.754116
I5 2.151162 2.633475 2.690360 1.754116 0.000000
>
> hls<-hclust(d,method="single")
>
> str(hls) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num [1:4] 0.877 1.041 1.655 1.754
 $ order      : int [1:5] 5 1 2 3 4
 $ labels     : chr [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr "single"
 $ call       : language hclust(d = d, method = "single")
 $ dist.method: chr "euclidean"
- attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

I risultati di **\$merge** sono stati disposti su due colonne:

- i numeri con il segno negativo indicano i singoli individui
- i numeri positivi indicano i cluster che si formano

\$height indica la distanza in cui è avvenuta l'agglomerazione tra i cluster



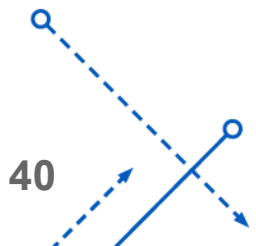
Esempio in R – Legame Singolo

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame singolo

```
> d<-dist(Z,method="euclidean",diag=TRUE,upper=TRUE)
> d # visualizza la matrice delle distanze
      I1      I2      I3      I4      I5
I1 0.000000 0.877058 2.246203 2.151162 2.151162
I2 0.877058 0.000000 1.654610 1.964247 2.633475
I3 2.246203 1.654610 0.000000 1.041245 2.690360
I4 2.151162 1.964247 1.041245 0.000000 1.754116
I5 2.151162 2.633475 2.690360 1.754116 0.000000
>
> hls<-hclust(d,method="single")
>
> str(hls) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int  [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num  [1:4] 0.877 1.041 1.655 1.754
 $ order      : int  [1:5] 5 1 2 3 4
 $ labels     : chr  [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr  "single"
 $ call       : language hclust(d = d, method = "single")
 $ dist.method: chr  "euclidean"
- attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

	Agglomerazione	Distanza
-1 -2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877



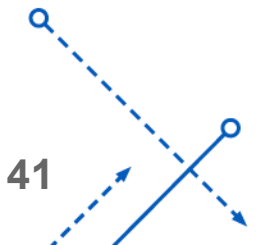
Esempio in R – Legame Singolo

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame singolo

```
> d<-dist(Z,method="euclidean",diag=TRUE,upper=TRUE)
> d # visualizza la matrice delle distanze
      I1      I2      I3      I4      I5
I1 0.000000 0.877058 2.246203 2.151162 2.151162
I2 0.877058 0.000000 1.654610 1.964247 2.633475
I3 2.246203 1.654610 0.000000 1.041245 2.690360
I4 2.151162 1.964247 1.041245 0.000000 1.754116
I5 2.151162 2.633475 2.690360 1.754116 0.000000
>
> hls<-hclust(d,method="single")
>
> str(hls) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int  [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num  [1:4] 0.877 1.041 1.655 1.754
 $ order      : int  [1:5] 5 1 2 3 4
 $ labels     : chr  [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr  "single"
 $ call       : language hclust(d = d, method = "single")
 $ dist.method: chr  "euclidean"
- attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

	Agglomerazione	Distanza
-1 -2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877
-3 -4	Al livello 2 si uniscono gli individui I_3 e I_4	1.041



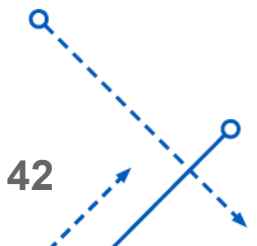
Esempio in R – Legame Singolo

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame singolo

```
> d<-dist(Z,method="euclidean",diag=TRUE,upper=TRUE)
> d # visualizza la matrice delle distanze
      I1      I2      I3      I4      I5
I1 0.000000 0.877058 2.246203 2.151162 2.151162
I2 0.877058 0.000000 1.654610 1.964247 2.633475
I3 2.246203 1.654610 0.000000 1.041245 2.690360
I4 2.151162 1.964247 1.041245 0.000000 1.754116
I5 2.151162 2.633475 2.690360 1.754116 0.000000
>
> hls<-hclust(d,method="single")
>
> str(hls) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int  [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num  [1:4] 0.877 1.041 1.655 1.754
 $ order      : int  [1:5] 5 1 2 3 4
 $ labels     : chr  [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr  "single"
 $ call       : language hclust(d = d, method = "single")
 $ dist.method: chr  "euclidean"
- attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

		Agglomerazione	Distanza
-1	-2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877
-3	-4	Al livello 2 si uniscono gli individui I_3 e I_4	1.041
1	2	Al livello 3 si uniscono il primo cluster (formato dagli individui I_1 e I_2) con il secondo cluster (formato dagli individui I_3 e I_4)	1.655



Esempio in R – Legame Singolo

- Calcoliamo ora la matrice delle distanze utilizzando la **metrica euclidea** e applichiamo il metodo gerarchico del legame singolo

```
> d<-dist(Z,method="euclidean",diag=TRUE,upper=TRUE)
> d # visualizza la matrice delle distanze
      I1      I2      I3      I4      I5
I1 0.000000 0.877058 2.246203 2.151162 2.151162
I2 0.877058 0.000000 1.654610 1.964247 2.633475
I3 2.246203 1.654610 0.000000 1.041245 2.690360
I4 2.151162 1.964247 1.041245 0.000000 1.754116
I5 2.151162 2.633475 2.690360 1.754116 0.000000
>
> hls<-hclust(d,method="single")
>
> str(hls) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int  [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num  [1:4] 0.877 1.041 1.655 1.754
 $ order      : int  [1:5] 5 1 2 3 4
 $ labels     : chr  [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr  "single"
 $ call       : language hclust(d = d, method = "single")
 $ dist.method: chr  "euclidean"
- attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

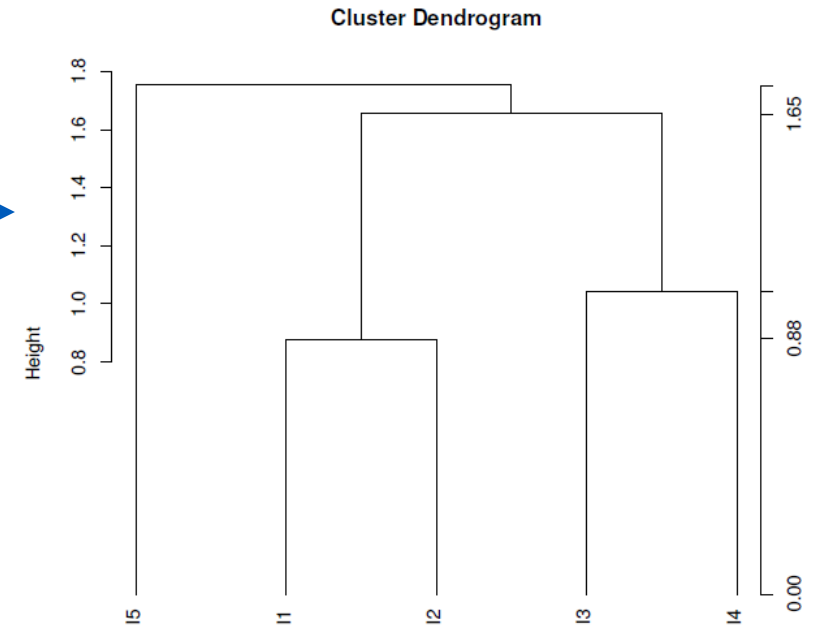
		Agglomerazione	Distanza
-1	-2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877
-3	-4	Al livello 2 si uniscono gli individui I_3 e I_4	1.041
1	2	Al livello 3 si uniscono il primo cluster (formato dagli individui I_1 e I_2) con il secondo cluster (formato dagli individui I_3 e I_4)	1.655
-5	3	Al livello 4 si unisce il terzo cluster (formato dagli individui I_1, I_2, I_3, I_4) con l'individuo I_5	1.754

Esempio in R – Legame Singolo

- Costruiamo ora il dendrogramma utilizzando le seguenti linee di codice

```
> plot(hls, hang=-1, xlab="Metodo gerarchico agglomerativo",  
+ sub="del legame singolo")  
> axis(side=4, at=round(c(0, hls$height), 2))
```

L'istruzione permette di costruire l'asse delle altezze alla destra del grafico arrotondando i numeri alla seconda cifra decimale



Metodo gerarchico agglomerativo
del legame singolo

Esempio in R – Legame Singolo

- Costruiamo ora il dendrogramma utilizzando le seguenti linee di codice

```
> plot(hls, hang=-1, xlab="Metodo gerarchico agglomerativo",  
+ sub="del legame singolo")  
> axis(side=4, at=round(c(0, hls$height), 2))
```

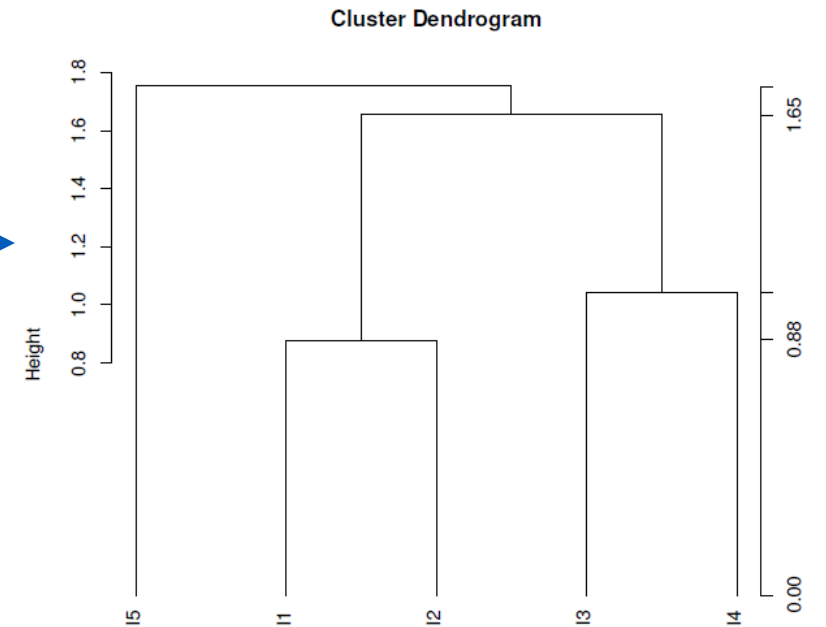
L'istruzione permette di costruire l'asse delle altezze alla destra del grafico arrotondando i numeri alla seconda cifra decimale

- Per visualizzare il taglio del dendrogramma in corrispondenza di un salto nelle distanze si utilizza la funzione

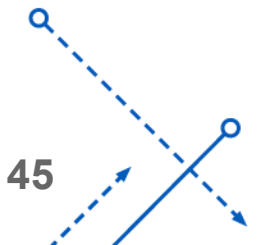
abline(h = NULL, lty = NULL)

dopo il comando plot() dove:

- **h** è l'altezza alla quale si inserisce il taglio;
- **lty** definisce lo stile della linea del taglio (1 per una linea continua, 2 per una linea tratteggiata, . . .)



Metodo gerarchico agglomerativo
del legame singolo



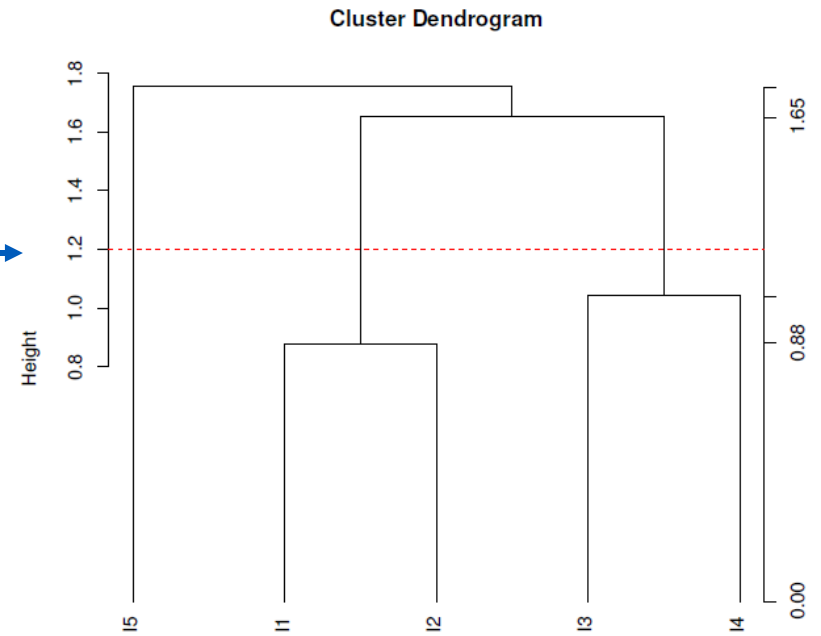
Esempio in R – Legame Singolo

- Costruiamo ora il dendrogramma utilizzando le seguenti linee di codice

```
> plot(hls, hang=-1, xlab="Metodo gerarchico agglomerativo",  
+ sub="del legame singolo")  
> axis(side=4, at=round(c(0, hls$height), 2))
```

L'istruzione permette di costruire l'asse delle altezze alla destra del grafico arrotondando i numeri alla seconda cifra decimale

```
> abline(h=1.2, lty=2, col="red")
```



Metodo gerarchico agglomerativo
del legame singolo

- Sulla base del livello di distanza è possibile identificare i cluster degli inconsiderati
 - nel grafico precedente un livello di distanza pari a 1.2 l'insieme di individui resta suddiviso nei tre cluster:
 $\{I_1, I_2\}, \{I_3, I_4\}, \{I_5\}$
- Nota: La scelta del punto di taglio dipende spesso dall'obiettivo dell'analisi e dalla comprensione dei dati

Esempio in R – Sveliamo l'arcano

- Come è avvenuto il processo di agglomerazione con il metodo del legame singolo?

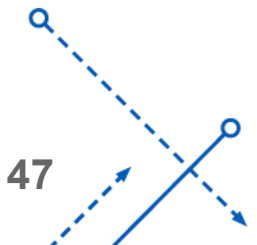
- Partiamo dalla matrice delle distanze ottenuta con R

- **Livello 1:** $d_{12} = 0.877058$ è il più piccolo valore della matrice delle distanze e pertanto I_1 e I_2 sono uniti formando un unico cluster

- Le distanze tra questo nuovo gruppo e $\{I_3\}, \{I_4\}, \{I_5\}$ sono (dalla matrice):

$$D = \begin{matrix} & I_1 & I_2 & I_3 & I_4 & I_5 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{0.877058} & 2.246203 & 2.151162 & 2.151162 \\ \boxed{0.877058} & 0.000000 & 1.654610 & 1.964247 & 2.633475 \\ 2.246203 & 1.654610 & 0.000000 & 1.041245 & 2.690360 \\ 2.151162 & 1.964247 & 1.041245 & 0.000000 & 1.754116 \\ 2.151162 & 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

$$\begin{aligned} d_{(1,2),3} &= \min(d_{13}, d_{23}) = \min(2.246203, 1.654610) = 1.654610 \\ d_{(1,2),4} &= \min(d_{14}, d_{24}) = \min(2.151162, 1.964247) = 1.964247 \\ d_{(1,2),5} &= \min(d_{15}, d_{25}) = \min(2.151162, 2.633475) = 2.151162. \end{aligned}$$



Esempio in R – Sveliamo l'arcano

- Come è avvenuto il processo di agglomerazione con il metodo del legame singolo?

- Partiamo dalla matrice delle distanze ottenuta con R

- **Livello 1:** $d_{12} = 0.877058$ è il più piccolo valore della matrice delle distanze e pertanto I_1 e I_2 sono uniti formando un unico cluster

- Le distanze tra questo nuovo gruppo e $\{I_3\}, \{I_4\}, \{I_5\}$ sono (dalla matrice):

$$D = \begin{matrix} & I_1 & I_2 & I_3 & I_4 & I_5 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{0.877058} & 2.246203 & 2.151162 & 2.151162 \\ \boxed{0.877058} & 0.000000 & 1.654610 & 1.964247 & 2.633475 \\ 2.246203 & 1.654610 & 0.000000 & 1.041245 & 2.690360 \\ 2.151162 & 1.964247 & 1.041245 & 0.000000 & 1.754116 \\ 2.151162 & 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

$$\begin{aligned} d_{(1,2),3} &= \min(d_{13}, d_{23}) = \min(2.246203, 1.654610) = 1.654610 \\ d_{(1,2),4} &= \min(d_{14}, d_{24}) = \min(2.151162, 1.964247) = 1.964247 \\ d_{(1,2),5} &= \min(d_{15}, d_{25}) = \min(2.151162, 2.633475) = 2.151162. \end{aligned}$$

- È quindi possibile costruire una nuova matrice delle distanze D_1 di ordine 4 (considerando un individuo in meno):

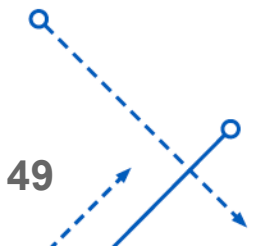
$$D_1 = \begin{matrix} & I_{1,2} & I_3 & I_4 & I_5 \\ \begin{matrix} I_{1,2} \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 1.654610 & 1.964247 & 2.151162 \\ 1.654610 & 0.000000 & \boxed{1.041245} & 2.690360 \\ 1.964247 & \boxed{1.041245} & 0.000000 & 1.754116 \\ 2.151162 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

Esempio in R – Sveliamo l'arcano

- **Livello 2:** $d_{34} = 1,041245$ è il più piccolo valore della matrice delle distanze e pertanto I_3 e I_4 sono uniti formando un unico cluster
 - Le distanze tra questo nuovo gruppo e $\{I_1, I_2\}, \{I_5\}$ sono (dalla matrice):

$$D_1 = \begin{matrix} & I_{1,2} & I_3 & I_4 & I_5 \\ \begin{matrix} I_{1,2} \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 1.654610 & 1.964247 & 2.151162 \\ 1.654610 & 0.000000 & \boxed{1.041245} & 2.690360 \\ 1.964247 & \boxed{1.041245} & 0.000000 & 1.754116 \\ 2.151162 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

$$d_{(3,4),(1,2)} = \min(d_{3,(1,2)}, d_{4,(1,2)}) = \min(1.654610, 1.964247) = 1.654610$$
$$d_{(3,4),5} = \min(d_{35}, d_{45}) = \min(2.690360, 1.754116) = 1.754116.$$



Esempio in R – Sveliamo l'arcano

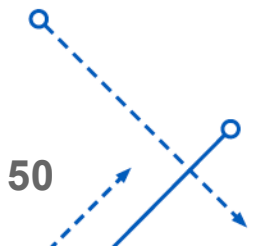
- **Livello 2:** $d_{34} = 1,041245$ è il più piccolo valore della matrice delle distanze e pertanto I_3 e I_4 sono uniti formando un unico cluster
 - Le distanze tra questo nuovo gruppo e $\{I_1, I_2\}, \{I_5\}$ sono (dalla matrice):
- È quindi possibile costruire una nuova matrice delle distanze D_2 di ordine 3 (considerando due individui in meno):

$$D_1 = \begin{matrix} & I_{1,2} & I_3 & I_4 & I_5 \\ \begin{matrix} I_{1,2} \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 1.654610 & 1.964247 & 2.151162 \\ 1.654610 & 0.000000 & \boxed{1.041245} & 2.690360 \\ 1.964247 & \boxed{1.041245} & 0.000000 & 1.754116 \\ 2.151162 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

$$d_{(3,4),(1,2)} = \min(d_{3,(1,2)}, d_{4,(1,2)}) = \min(1.654610, 1.964247) = 1.654610$$

$$d_{(3,4),5} = \min(d_{35}, d_{45}) = \min(2.690360, 1.754116) = 1.754116.$$

$$D_2 = \begin{matrix} & I_{1,2} & I_{3,4} & I_5 \\ \begin{matrix} I_{1,2} \\ I_{3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{1.654610} & 2.151162 \\ \boxed{1.654610} & 0.000000 & 1.754116 \\ 2.151162 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}.$$



Esempio in R – Sveliamo l'arcano

- **Livello 3:** $d_{(12),(34)} = 1,654610$ è il più piccolo valore della matrice delle distanze e pertanto $I_{(1,2)}$ e $I_{(3,4)}$ sono uniti formando un unico cluster

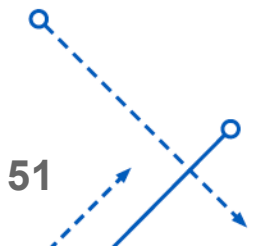
- La distanza tra questo nuovo gruppo e $\{I_5\}$ è (dalla matrice):

$$d_{(1,2,3,4),5} = \min(d_{(1,2),5}, d_{(3,4),5}) = \min(2.151162, 1.754116) = 1.754116$$

- È quindi possibile costruire una nuova matrice delle distanze D_3 di ordine 2 (considerando due individui in meno):

$$D_2 = \begin{matrix} & I_{1,2} & I_{3,4} & I_5 \\ \begin{matrix} I_{1,2} \\ I_{3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{1.654610} & 2.151162 \\ \boxed{1.654610} & 0.000000 & 1.754116 \\ 2.151162 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}.$$

$$D_3 = \begin{matrix} & I_{1,2,3,4} & I_5 \\ \begin{matrix} I_{1,2,3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{1.754116} \\ \boxed{1.754116} & 0.000000 \end{pmatrix} \end{matrix}$$



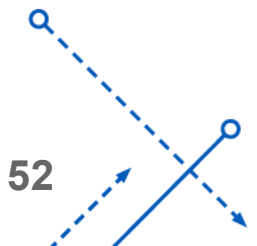
Esempio in R – Sveliamo l'arcano

- **Livello 4:** $d_{(1234),5} = 1,754116$ è il più piccolo valore della matrice delle distanze e pertanto $I_{(1,2,3,4)}$ e I_5 sono uniti formando un unico cluster

$$D_3 = \begin{matrix} & I_{1,2,3,4} & I_5 \\ \begin{matrix} I_{1,2,3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{1.754116} \\ \boxed{1.754116} & 0.000000 \end{pmatrix} \end{matrix}$$

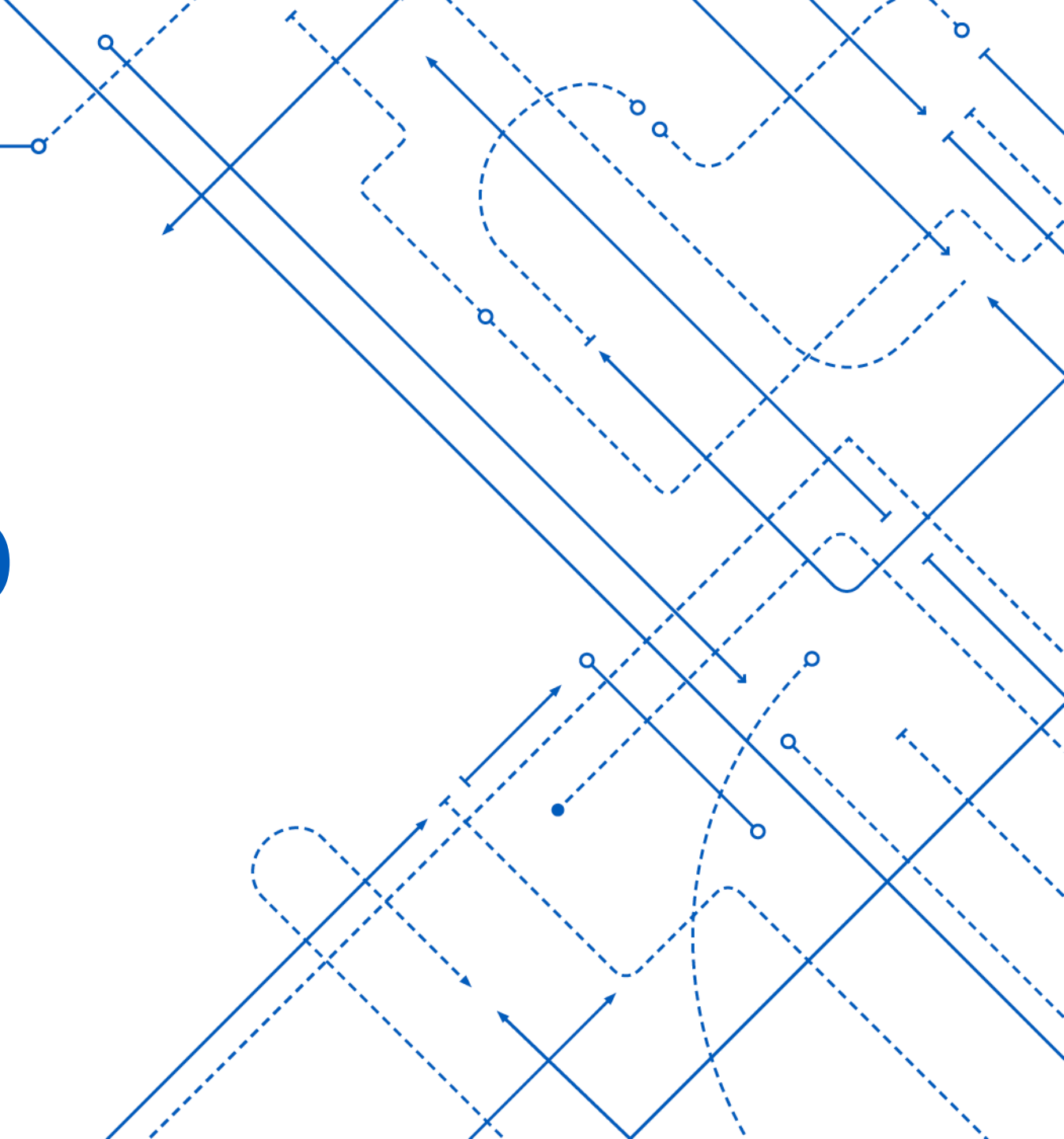
- La sequenza delle agglomerazioni del metodo del legame singolo è stata:

Numero di cluster	Cluster	Livello di distanza
5	$\{I_1\}, \{I_2\}, \{I_3\}, \{I_4\}, \{I_5\}$	
4	$\{I_1, I_2\}, \{I_3\}, \{I_4\}, \{I_5\}$	0.877058
3	$\{I_1, I_2\}, \{I_3, I_4\}, \{I_5\}$	1.041245
2	$\{I_1, I_2, I_3, I_4\}, \{I_5\}$	1.654610
1	$\{I_1, I_2, I_3, I_4, I_5\}$	1.754116



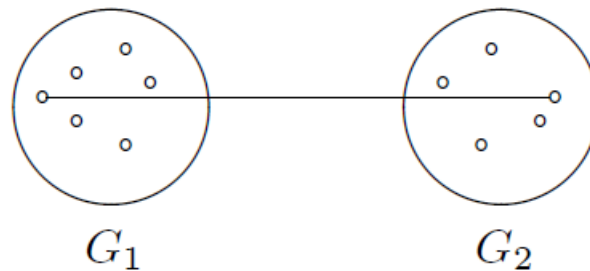
CLUSTERING GERARCHICO

Metodo del Legame Completo



Metodo del Legame Completo

- Nel **Metodo del legame completo** o Furthest neighbour method la distanza tra i gruppi G_1 (contenente n_1 individui) e G_2 (contenente n_2 individui) è definita come la **massima** tra tutte le distanze tra $n_1 n_2$ che si possono calcolare tra ogni individuo di G_1 e ogni individuo di G_2



- Nella procedura gerarchica si considera inizialmente, ossia al livello 0, un insieme di n cluster $\{I_1\}, \{I_2\}, \dots, \{I_n\}$
- Al passo successivo si cerca nella matrice D delle distanze il coefficiente di distanza **minima** e si raggruppano nello stesso cluster G_{ij} i due individui I_i e I_j associati secondo tale coefficiente
- Nel caso i coefficienti di distanza minima siano più di uno, si attua una scelta arbitraria tra di essi.



Metodo del Legame Completo

- Al livello 1 quindi si modifica la matrice delle distanze valutando le distanze di G_{ij} da ogni altro individuo I_k non appartenente a G_{ij} mediante la seguente relazione:

$$d_{(ij),k} = \max(d_{ij}, d_{jk})$$

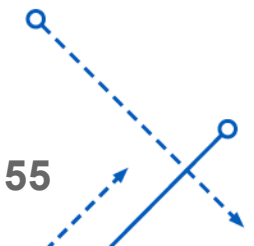
Cioè la distanza dell'individuo I_k dal cluster G_{ij} si ottiene scegliendo la più grande distanza tra d_{ij} e d_{jk}

- Quindi, al livello 1 si costruisce una nuova matrice D_1 di cardinalità $(n-1) \times (n-1)$ costituita da G_{ij} (che viene considerato come un unico elemento) e dai restanti $(n-2)$ individui fuori dal cluster G_{ij}
- Ad ogni passo successivo, dopo che i cluster G_u e G_v vengono uniti scegliendo dalla precedente matrice delle distanze i due cluster più vicini, la distanza tra il nuovo cluster, denotato con G_{uv} , e un altro cluster G_z è così definita:

$$d_{(uv),z} = \max(d_{uz}, d_{vz})$$

Cioè $d_{(uv),z}$ rappresenta la misura di distanza tra gli elementi meno distanti dei cluster G_{uv} e G_z

- La procedura si ripete fino ad ottenere un unico cluster formato da tutti gli individui



Esempio in R

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame completo

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

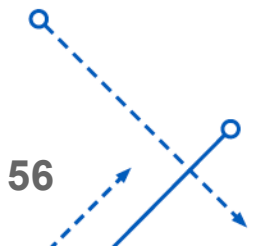
```
> hlc<-hclust(d,method="complete")
>
> str(hlc) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num [1:4] 0.877 1.041 2.246 2.69
 $ order      : int [1:5] 5 1 2 3 4
 $ labels     : chr [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr "complete"
 $ call       : language hclust(d = d, method = "complete")
 $ dist.method: chr "euclidean"
 - attr(*, "class")= chr "hclust"
```

I risultati di **\$merge** sono stati disposti su due colonne:

- i numeri con il segno negativo indicano i singoli individui
- i numeri positivi indicano i cluster che si formano

\$height indica la distanza in cui è avvenuta l'agglomerazione tra i cluster

	Agglomerazione	Distanza
-1 -2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877



Esempio in R

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame completo

```
> hlc<-hclust(d,method="complete")
>
> str(hlc) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num [1:4] 0.877 1.041 2.246 2.69
 $ order      : int [1:5] 5 1 2 3 4
 $ labels     : chr [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr "complete"
 $ call       : language hclust(d = d, method = "complete")
 $ dist.method: chr "euclidean"
 - attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

	Agglomerazione	Distanza
-1 -2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877
-3 -4	Al livello 2 si uniscono gli individui I_3 e I_4	1.041



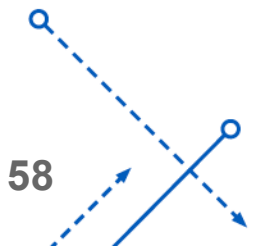
Esempio in R

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame completo

```
> hlc<-hclust(d,method="complete")
>
> str(hlc) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num [1:4] 0.877 1.041 2.246 2.69
 $ order      : int [1:5] 5 1 2 3 4
 $ labels     : chr [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr "complete"
 $ call       : language hclust(d = d, method = "complete")
 $ dist.method: chr "euclidean"
 - attr(*, "class")= chr "hclust"
```

$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$

	Agglomerazione	Distanza
-1 -2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877
-3 -4	Al livello 2 si uniscono gli individui I_3 e I_4	1.041
1 2	Al livello 3 si uniscono il primo cluster (formato dagli individui I_1 e I_2) con il secondo cluster (formato dagli individui I_3 e I_4)	2.246



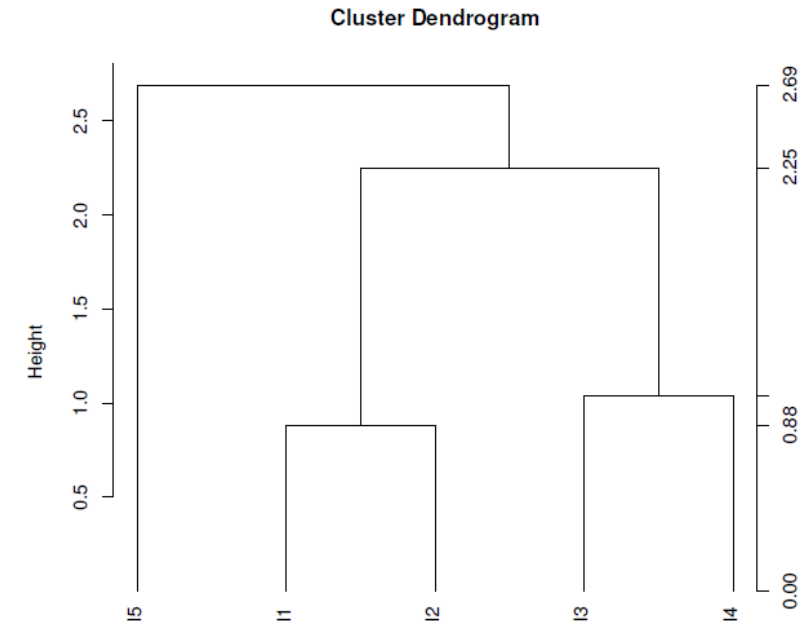
Esempio in R

- Calcoliamo ora la matrice delle distanze utilizzando la metrica euclidea e applichiamo il metodo gerarchico del legame completo

```
> hlc<-hclust(d,method="complete")
>
> str(hlc) # visualizza informazioni sull'oggetto cluster
List of 7
 $ merge      : int [1:4, 1:2] -1 -3 1 -5 -2 -4 2 3
 $ height     : num [1:4] 0.877 1.041 2.246 2.69
 $ order      : int [1:5] 5 1 2 3 4
 $ labels     : chr [1:5] "I1" "I2" "I3" "I4" ...
 $ method     : chr "complete"
 $ call       : language hclust(d = d, method = "complete")
 $ dist.method: chr "euclidean"
 - attr(*, "class")= chr "hclust"
```

		Agglomerazione	Distanza
-1	-2	Al livello 1 si uniscono gli individui I_1 e I_2	0.877
-3	-4	Al livello 2 si uniscono gli individui I_3 e I_4	1.041
1	2	Al livello 3 si uniscono il primo cluster (formato dagli individui I_1 e I_2) con il secondo cluster (formato dagli individui I_3 e I_4)	2.246
-5	3	Al livello 4 si unisce il terzo cluster (formato dagli individui I_1, I_2, I_3, I_4) con l'individuo I_5	2.690

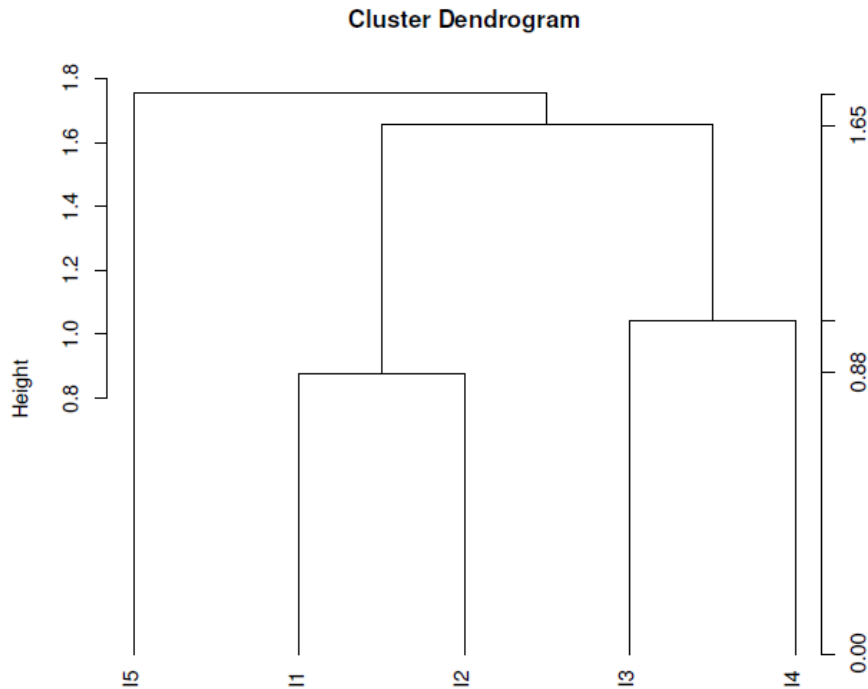
$$X = \begin{matrix} & C_1 & C_2 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 6 & 3 \\ 8 & 2 \\ 8 & 0 \end{pmatrix} \end{matrix}$$



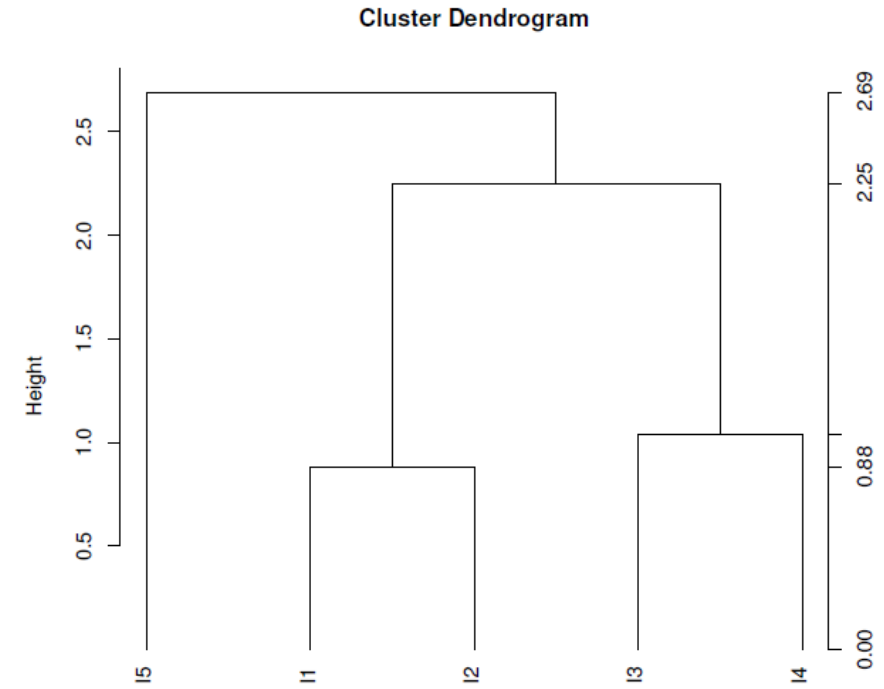
Metodo gerarchico agglomerativo
del legame completo

Singolo VS Completo

- Si nota che con il metodo del legame completo la suddivisione in cluster non cambia rispetto al caso del legame singolo, ma variano alcuni livelli di distanza relativi alle aggregazioni
- Confrontando i dendrogrammi si nota che risulta più accentuato il salto tra i livelli di distanza corrispondenti alle ultime due aggregazioni



Metodo gerarchico agglomerativo
del legame singolo



Metodo gerarchico agglomerativo
del legame completo

Esempio in R – Sveliamo l'arcano

- Come è avvenuto il processo di agglomerazione con il metodo del legame completo?

- Partiamo dalla matrice delle distanze ottenuta con R

- **Livello 1:** $d_{12} = 0.877058$ è il più piccolo valore della matrice delle distanze e pertanto I_1 e I_2 sono uniti formando un unico cluster

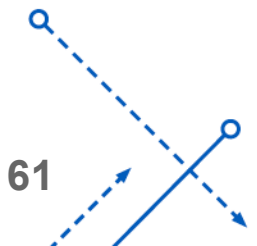
- Le distanze tra questo nuovo gruppo e $\{I_3\}, \{I_4\}, \{I_5\}$ sono (dalla matrice):

$$D = \begin{matrix} & I_1 & I_2 & I_3 & I_4 & I_5 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 0.877058 & 2.246203 & 2.151162 & 2.151162 \\ 0.877058 & 0.000000 & 1.654610 & 1.964247 & 2.633475 \\ 2.246203 & 1.654610 & 0.000000 & 1.041245 & 2.690360 \\ 2.151162 & 1.964247 & 1.041245 & 0.000000 & 1.754116 \\ 2.151162 & 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

$$d_{(1,2),3} = \max(d_{13}, d_{23}) = \max(2.246203, 1.654610) = 2.246203$$

$$d_{(1,2),4} = \max(d_{14}, d_{24}) = \max(2.151162, 1.964247) = 2.151162$$

$$d_{(1,2),5} = \max(d_{15}, d_{25}) = \max(2.151162, 2.633475) = 2.633475$$



Esempio in R – Sveliamo l'arcano

- Come è avvenuto il processo di agglomerazione con il metodo del legame completo?

- Partiamo dalla matrice delle distanze ottenuta con R

- Livello 1:** $d_{12} = 0.877058$ è il più piccolo valore della matrice delle distanze e pertanto I_1 e I_2 sono uniti formando un unico cluster

- Le distanze tra questo nuovo gruppo e $\{I_3\}, \{I_4\}, \{I_5\}$ sono (dalla matrice):

$$D = \begin{matrix} & I_1 & I_2 & I_3 & I_4 & I_5 \\ \begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{0.877058} & 2.246203 & 2.151162 & 2.151162 \\ \boxed{0.877058} & 0.000000 & 1.654610 & 1.964247 & 2.633475 \\ 2.246203 & 1.654610 & 0.000000 & 1.041245 & 2.690360 \\ 2.151162 & 1.964247 & 1.041245 & 0.000000 & 1.754116 \\ 2.151162 & 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

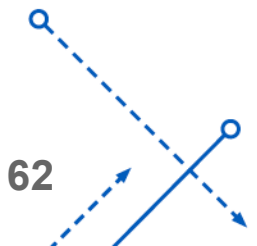
$$d_{(1,2),3} = \max(d_{13}, d_{23}) = \max(2.246203, 1.654610) = 2.246203$$

$$d_{(1,2),4} = \max(d_{14}, d_{24}) = \max(2.151162, 1.964247) = 2.151162$$

$$d_{(1,2),5} = \max(d_{15}, d_{25}) = \max(2.151162, 2.633475) = 2.633475$$

- È quindi possibile costruire una nuova matrice delle distanze D_1 di ordine 4 (considerando un individuo in meno):

$$D_1 = \begin{matrix} & I_{1,2} & I_3 & I_4 & I_5 \\ \begin{matrix} I_{1,2} \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 2.246203 & 2.151162 & 2.633475 \\ 2.246203 & 0.000000 & \boxed{1.041245} & 2.690360 \\ 2.151162 & \boxed{1.041245} & 0.000000 & 1.754116 \\ 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$



Esempio in R – Sveliamo l'arcano

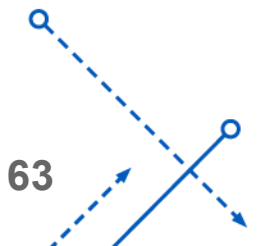
- **Livello 2:** $d_{34} = 1,041245$ è il più piccolo valore della matrice delle distanze e pertanto I_3 e I_4 sono uniti formando un unico cluster

$$D_1 = \begin{matrix} & I_{1,2} & I_3 & I_4 & I_5 \\ \begin{matrix} I_{1,2} \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 2.246203 & 2.151162 & 2.633475 \\ 2.246203 & 0.000000 & \boxed{1.041245} & 2.690360 \\ 2.151162 & \boxed{1.041245} & 0.000000 & 1.754116 \\ 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

- Le distanze tra questo nuovo gruppo e $\{I_1, I_2\}, \{I_5\}$ sono (dalla matrice):

$$d_{(3,4),(1,2)} = \max(d_{3,(1,2)}, d_{4,(1,2)}) = \max(2.246203, 2.151162) = 2.246203$$

$$d_{(3,4),5} = \max(d_{35}, d_{45}) = \max(2.690360, 1.754116) = 2.690360$$



Esempio in R – Sveliamo l'arcano

- **Livello 2:** $d_{34} = 1,041245$ è il più piccolo valore della matrice delle distanze e pertanto I_3 e I_4 sono uniti formando un unico cluster

$$D_1 = \begin{matrix} & I_{1,2} & I_3 & I_4 & I_5 \\ \begin{matrix} I_{1,2} \\ I_3 \\ I_4 \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & 2.246203 & 2.151162 & 2.633475 \\ 2.246203 & 0.000000 & \boxed{1.041245} & 2.690360 \\ 2.151162 & \boxed{1.041245} & 0.000000 & 1.754116 \\ 2.633475 & 2.690360 & 1.754116 & 0.000000 \end{pmatrix} \end{matrix}$$

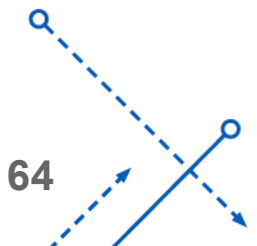
- Le distanze tra questo nuovo gruppo e $\{I_1, I_2\}, \{I_5\}$ sono (dalla matrice):

$$d_{(3,4),(1,2)} = \max(d_{3,(1,2)}, d_{4,(1,2)}) = \max(2.246203, 2.151162) = 2.246203$$

$$d_{(3,4),5} = \max(d_{35}, d_{45}) = \max(2.690360, 1.754116) = 2.690360$$

- È quindi possibile costruire una nuova matrice delle distanze D_2 di ordine 3 (considerando due individui in meno):

$$D_2 = \begin{matrix} & I_{1,2} & I_{3,4} & I_5 \\ \begin{matrix} I_{1,2} \\ I_{3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{2.246203} & 2.633475 \\ \boxed{2.246203} & 0.000000 & 2.690360 \\ 2.633475 & 2.690360 & 0.000000 \end{pmatrix} \end{matrix}.$$



Esempio in R – Sveliamo l'arcano

- **Livello 3:** $d_{(12),(34)} = 2.246203$ è il più piccolo valore della matrice delle distanze e pertanto $I_{(1,2)}$ e $I_{(3,4)}$ sono uniti formando un unico cluster

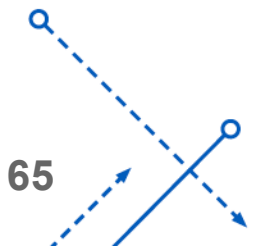
$$D_2 = \begin{matrix} & I_{1,2} & I_{3,4} & I_5 \\ \begin{matrix} I_{1,2} \\ I_{3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{2.246203} & 2.633475 \\ \boxed{2.246203} & 0.000000 & 2.690360 \\ 2.633475 & 2.690360 & 0.000000 \end{pmatrix} \end{matrix}.$$

- La distanza tra questo nuovo gruppo e $\{I_5\}$ è (dalla matrice):

$$d_{(1,2,3,4),5} = \max(d_{(1,2),5}, d_{(3,4),5}) = \max(2.633475, 2.690360) = 2.690360$$

- È quindi possibile costruire una nuova matrice delle distanze D_3 di ordine 2 (considerando due individui in meno):

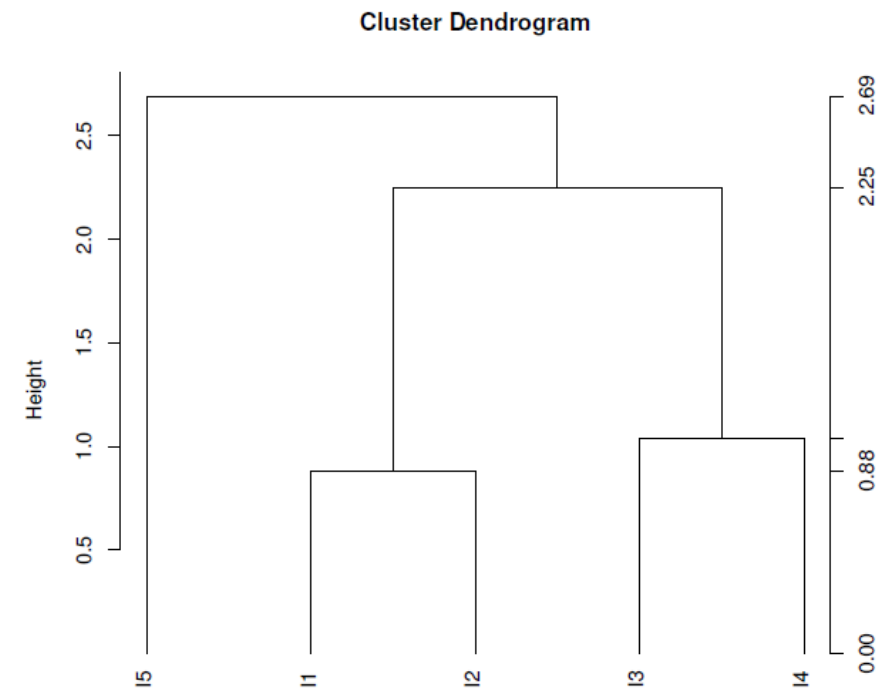
$$D_3 = \begin{matrix} & I_{1,2,3,4} & I_5 \\ \begin{matrix} I_{1,2,3,4} \\ I_5 \end{matrix} & \begin{pmatrix} 0.000000 & \boxed{2.690360} \\ \boxed{2.690360} & 0.000000 \end{pmatrix} \end{matrix}$$



Esempio in R – Sveliamo l'arcano

- **Livello 4:** Unendo i gruppi $\{I_1, I_2, I_3, I_4\}$ e $\{I_5\}$ si ottiene un unico cluster contenente tutti e 5 gli individui
- La sequenza delle agglomerazioni del metodo del legame completo è stata:

Numero di cluster	Cluster	Livello di distanza
5	$\{I_1\}, \{I_2\}, \{I_3\}, \{I_4\}, \{I_5\}$	
4	$\{I_1, I_2\}, \{I_3\}, \{I_4\}, \{I_5\}$	0.877058
3	$\{I_1, I_2\}, \{I_3, I_4\}, \{I_5\}$	1.041245
2	$\{I_1, I_2, I_3, I_4\}, \{I_5\}$	2.246203
1	$\{I_1, I_2, I_3, I_4, I_5\}$	2.690360



Metodo gerarchico agglomerativo
del legame completo