Statistica II (750AA)

Lezione 2

Dario Trevisan – *https://web.dm.unipi.it/trevisan*

29/09/2025

Richiami dalla lezione 1

- Interpretazione variazionale (ERM) degli indicatori di centralità

Richiami dalla lezione 1

- Interpretazione variazionale (ERM) degli indicatori di centralità
- Distanze multidimensionali

Centroide e Medoide

$(x_i)_{i=1,\dots,n}$ osservazioni anche in $\mathbb{R}^d$

La media o centroide è $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$

- Medoide: $x \in \{x_i\}_{i=1,\dots,n}$ che minimizza

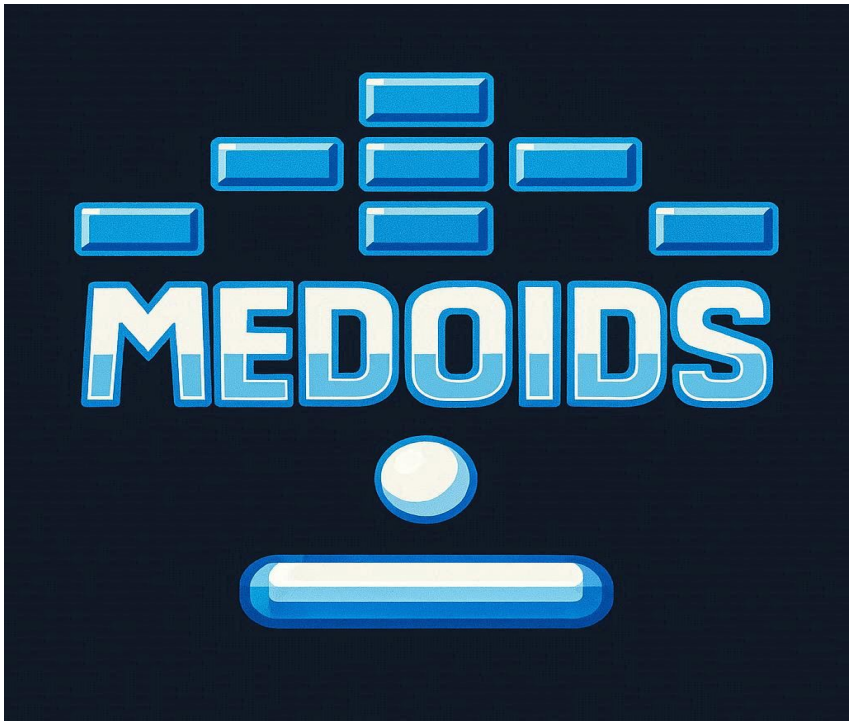
$$\sum_{i=1}^{n} \|x - x_i\|^2$$

$$x \in \operatorname*{argmin}_{h \in \{x_j\}_j} \sum_{i=1}^{n} \|h - x_i\|^2$$

Più in generale si può definire medoide rispetto a una funzione di loss $\ell(x_i; h)$ $\longrightarrow$ $x \in \operatorname*{argmin}_{h \in \{x_j\}} \sum_{i=1}^{n} \ell(x_i; h)$

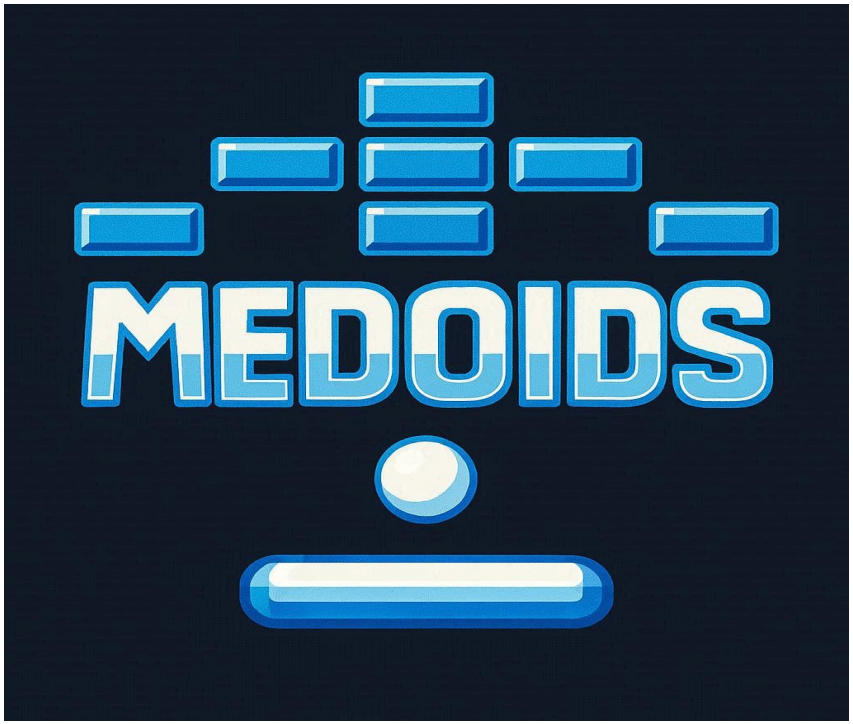
Medoids!

.



- Giocate voi! https://dario-trevisan.shinyapps.io/medoids/

Medoids!



- Giocate voi! https://dario-trevisan.shinyapps.io/medoids/
- Codice (R shiny) disponibile nel team e nella pagina del corso

Clustering

Introduzione al clustering

Non supervisionata / Esplorativa

Obiettivo del clustering

- Raggruppare i dati $\{x_i\}_{i=1\ldots,n}$ in gruppi simili:

Introduzione al clustering

Obiettivo del clustering

- Raggruppare i dati $\{x_i\}_{i=1\ldots,n}$ in gruppi simili:
  - in un certo numero k di gruppi disgiunti (cluster) $(C_j)_{j=1\ldots,k}$:

$$\{x_i\}_{i=1,\ldots,n} = \bigcup_{j=1}^{k} C_j$$

Introduzione al clustering

Obiettivo del clustering

- Raggruppare i dati $\{x_i\}_{i=1\ldots,n}$ in gruppi simili:
  - in un certo numero k di gruppi disgiunti (cluster) $(C_j)_{j=1\ldots,k}$:

$$\{x_i\}_{i=1,\ldots,n} = \bigcup_{j=1}^{k} C_j$$

  - minimizzando la **variabilità** all'interno di ciascun cluster.

Introduzione al clustering

Obiettivo del clustering

- Raggruppare i dati $\{x_i\}_{i=1\ldots,n}$ in gruppi simili:
  - in un certo numero k di gruppi disgiunti (cluster) $(C_j)_{j=1\ldots,k}$:

$$\{x_i\}_{i=1,\ldots,n} = \bigcup_{j=1}^{k} C_j$$

  - minimizzando la **variabilità** all'interno di ciascun cluster.
- **Applicazioni**: marketing, medicina, logistica... $\rightarrow$ classificazione (imputation)

Problemi principali

- Quale nozione di similarità? (scelta del costo)

Problemi principali

- Quale nozione di similarità? (scelta del costo)
- Quanti clusters? (*iperparametro* k)

Problemi principali

- Quale nozione di similarità? (scelta del costo)
- Quanti clusters? (*iperparametro* k)
- Come calcolare i cluster? (algoritmi)

Metodi di clustering

- Non-gerarchici → K fissato

Metodi di clustering

- Non-gerarchici
  - K-means

Metodi di clustering

- Non-gerarchici
  - K-means
  - PAM

Metodi di clustering

- Non-gerarchici
  - K-means
  - PAM
- Gerarchici $\longrightarrow$ K non fissato si crea una gerarchia di possibili cluster

Metodi di clustering

- Non-gerarchici
  - K-means
  - PAM
- Gerarchici
  - Agglomerativo (AGNES)

Metodi di clustering

- Non-gerarchici
  - K-means
  - PAM
- Gerarchici
  - Agglomerativo (AGNES)
  - Divisivo (DIANA)

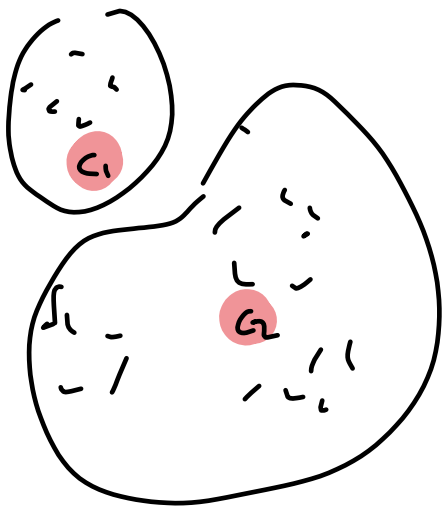
K-means

- K fissato
- similarità è misurata con distanza Euclidea

ERM: minimizzare il *Within Clusters Sum of Squares* (inerzia):

$$WCSS((C_j)_{j=1}^k) = \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - c_j\|^2$$

dove c_j è il centroide (media) del cluster C_j:



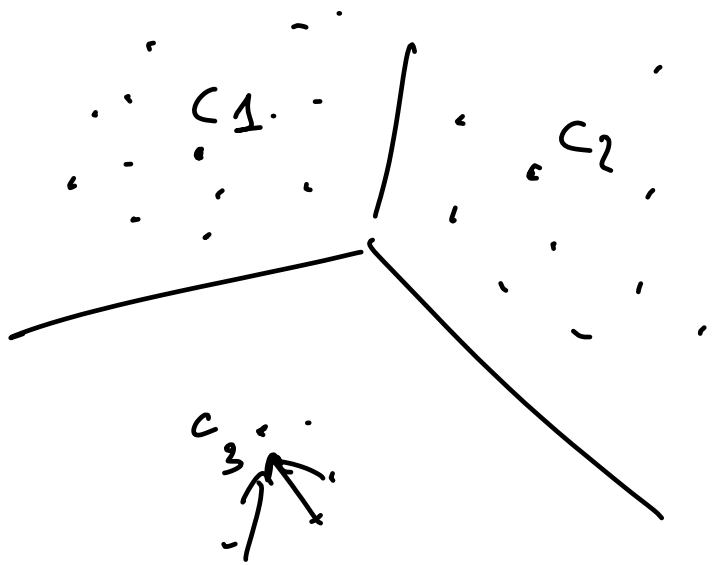
$$c_j := \frac{1}{\sharp C_j} \sum_{x_i \in C_j} x_i.$$

↑ numero di elementi nel cluster C_j

Algoritmo di Lloyd

Il problema è molto difficile da risolvere esattamente (NP-hard)

- I cluster ottimali sono tali che ogni osservazione ~~osservazione~~ x_i viene assegnata al centroide c_j più vicino.



Algoritmo di Lloyd

Il problema è molto difficile da risolvere esattamente (NP-hard)

- I cluster ottimali sono tali che ogni osservazione osservazione x_i viene assegnata al centroide c_j più vicino.
- Questo suggerisce un **algoritmo iterativo**:

Algoritmo di Lloyd

Il problema è molto difficile da risolvere esattamente (NP-hard)

- I cluster ottimali sono tali che ogni osservazione osservazione x_i viene assegnata al centroide c_j più vicino.
- Questo suggerisce un **algoritmo iterativo**:

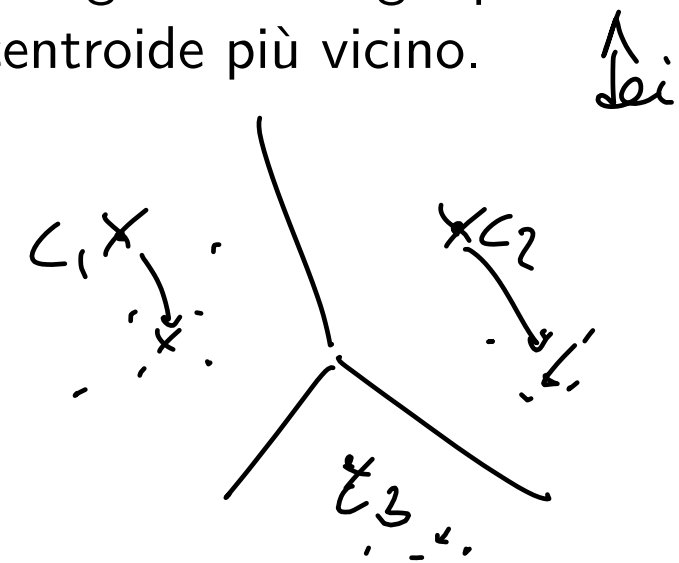
0. Inizializzazione: scegliere k centroidi casualmente.

Algoritmo di Lloyd

Il problema è molto difficile da risolvere esattamente (NP-hard)

- I cluster ottimali sono tali che ogni osservazione osservazione x_i viene assegnata al centroide c_j più vicino.
- Questo suggerisce un **algoritmo iterativo**:

0. Inizializzazione: scegliere k centroidi casualmente.
1. Assegnazione: ogni punto dati viene assegnato al cluster con il centroide più vicino.



Algoritmo di Lloyd

Il problema è molto difficile da risolvere esattamente (NP-hard)

- I cluster ottimali sono tali che ogni osservazione osservazione x_i viene assegnata al centroide c_j più vicino.
- Questo suggerisce un **algoritmo iterativo**:

0. Inizializzazione: scegliere k centroidi casualmente.
1. Assegnazione: ogni punto dati viene assegnato al cluster con il centroide più vicino.
2. Aggiornamento: i centroidi vengono ricalcolati come la media dei punti assegnati. Ripetere 1-2 fino alla convergenza.

- **Vantaggi**:

- **Vantaggi**:
  - Semplice da implementare

- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.

- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.
  - Si aggiorna facilmente per nuove osservazioni.

- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.
  - Si aggiorna facilmente per nuove osservazioni.
- **Svantaggi**:

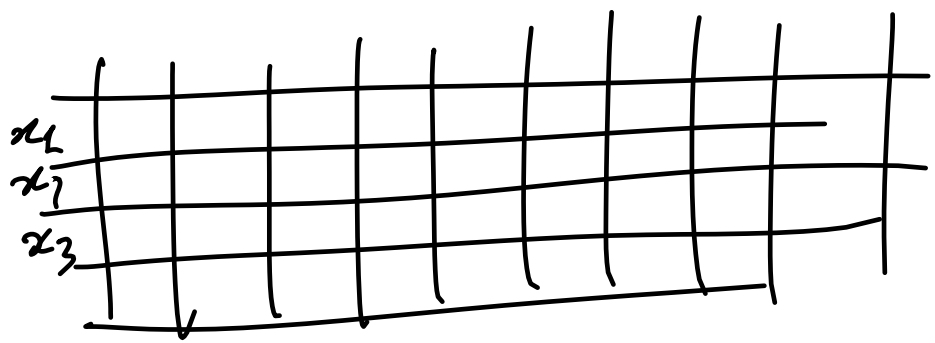
- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.
  - Si aggiorna facilmente per nuove osservazioni.
- **Svantaggi**:
  - Necessità di specificare k.

- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.
  - Si aggiorna facilmente per nuove osservazioni.
- **Svantaggi**:
  - Necessità di specificare k.
  - Sensibile agli outlier.

- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.
  - Si aggiorna facilmente per nuove osservazioni.
- **Svantaggi**:
  - Necessità di specificare k.
  - Sensibile agli outlier.
  - Risultato dipende dall'inizializzazione.

- **Vantaggi**:
  - Semplice da implementare
  - Veloce anche per dataset di grandi dimensioni.
  - Si aggiorna facilmente per nuove osservazioni.
- **Svantaggi**:
  - Necessità di specificare k.
  - Sensibile agli outlier.
  - Risultato dipende dall'inizializzazione.
  - **Curse of dimensionality**

Se $d >> 4$



K-medoids PAM (Partitioning Around Medoids)

Simile a K-means, ma utilizza medoidi invece della media:

$$WCR((C_j)_{j=1}^{k}) = \sum_{j=1}^{k} \sum_{x_i \in C_j} \ell(x_i, c_j)$$

dove c_j è il medoide del cluster C_j e $\ell(x, c)$ è una funzione di costo.

- Vantaggi:

- Vantaggi:
  - Utile se la media non ha significato

- Vantaggi:
  - Utile se la media non ha significato
  - Maggiore robustezza agli outlier rispetto a k-means

- Vantaggi:
  - Utile se la media non ha significato
  - Maggiore robustezza agli outlier rispetto a k-means
  - Distanze/costi più generali (anche categorici, k-modes)

- Vantaggi:
  - Utile se la media non ha significato
  - Maggiore robustezza agli outlier rispetto a k-means
  - Distanze/costi più generali (anche categorici, k-modes)
- Svantaggi:

- Vantaggi:
  - Utile se la media non ha significato
  - Maggiore robustezza agli outlier rispetto a k-means
  - Distanze/costi più generali (anche categorici, k-modes)
- Svantaggi:
  - Algoritmi generalmente più lenti

- Vantaggi:
  - Utile se la media non ha significato
  - Maggiore robustezza agli outlier rispetto a k-means
  - Distanze/costi più generali (anche categorici, k-modes)
- Svantaggi:
  - Algoritmi generalmente più lenti
  - **Curse of dimensionality**

Metodi gerarchici (HCA)

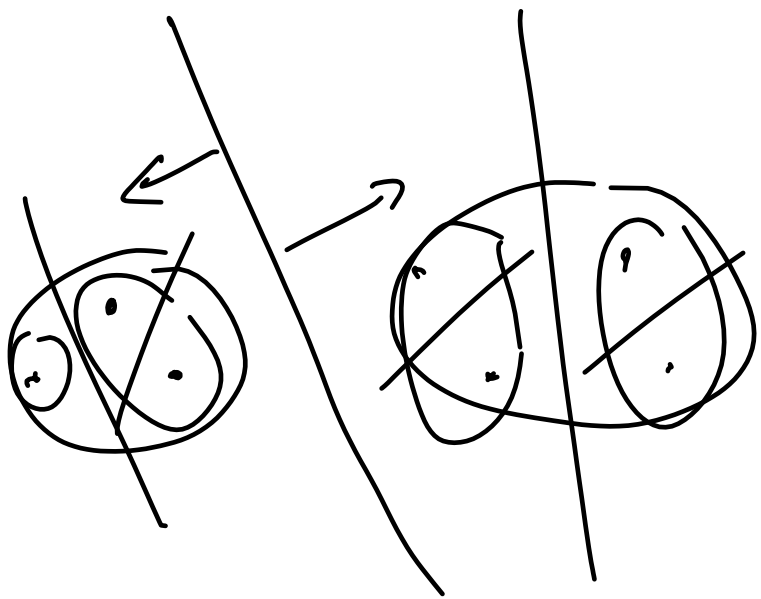
- **Definizione**: tecniche di clustering che costruiscono una *gerarchia di cluster*.

Metodi gerarchici (HCA)

- **Definizione**: tecniche di clustering che costruiscono una *gerarchia di cluster*.
  - agglomerativi (bottom-up)

Metodi gerarchici (HCA)

- **Definizione**: tecniche di clustering che costruiscono una *gerarchia di cluster*.
  - agglomerativi (bottom-up)
  - divisivi (top-down).

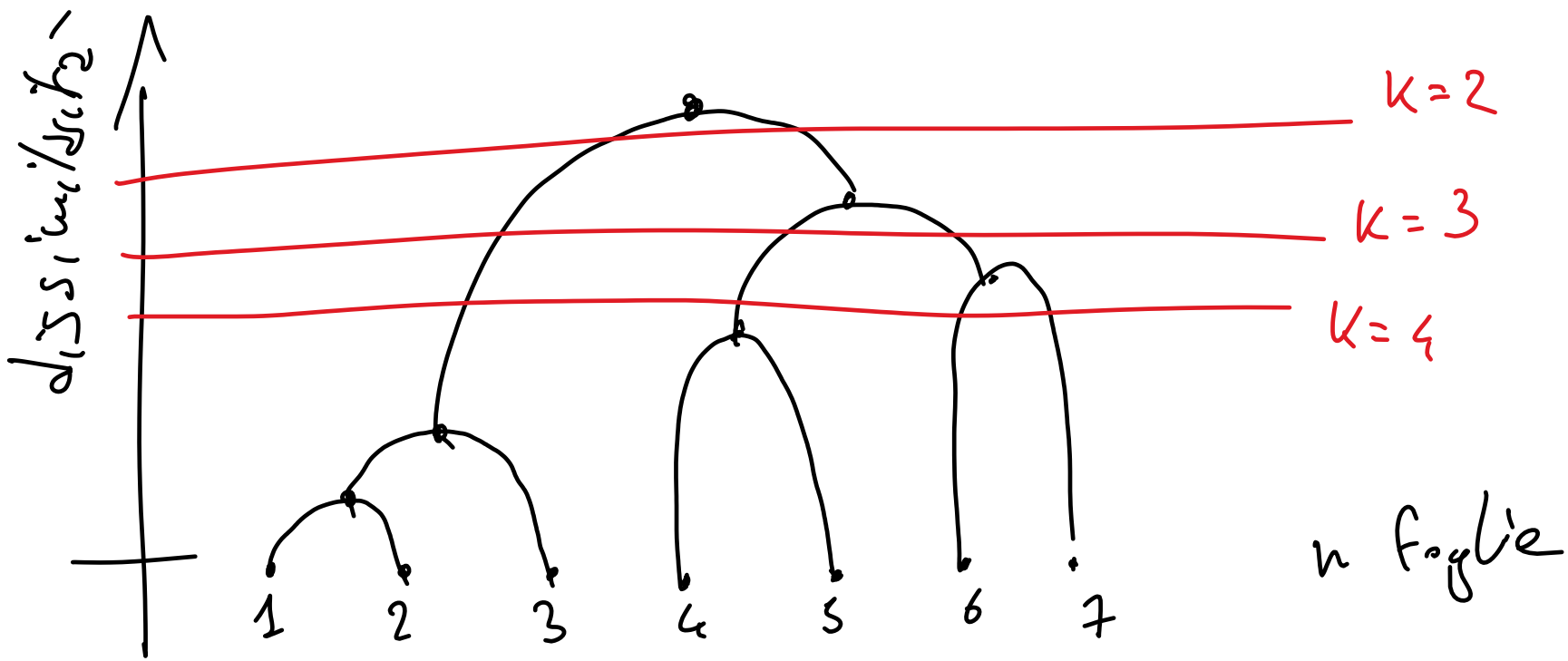


Metodi gerarchici (HCA)

- **Definizione**: tecniche di clustering che costruiscono una *gerarchia di cluster*.
  - agglomerativi (bottom-up)
  - divisivi (top-down).
- **Vantaggi**: Non richiede il numero di cluster in anticipo e fornisce una gerarchia completa.

Dendrogramma

- **Descrizione**: rappresentazione della gerarchia dei cluster mediante un *grafo ad albero*. Ogni ramo rappresenta un cluster e la lunghezza del ramo indica la distanza tra i cluster uniti.



Dendrogramma

- **Descrizione**: rappresentazione della gerarchia dei cluster mediante un *grafo ad albero*. Ogni ramo rappresenta un cluster e la lunghezza del ramo indica la distanza tra i cluster uniti.
- **Interpretazione**:

Dendrogramma

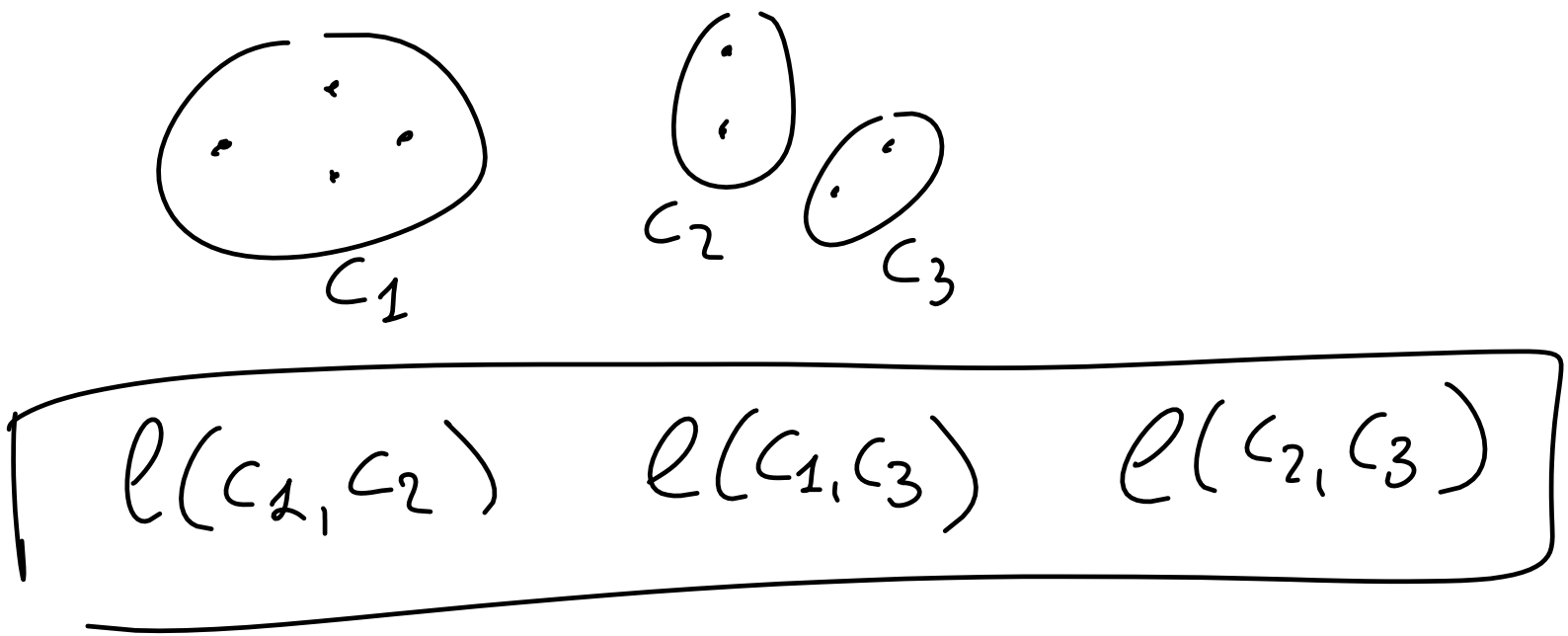
- **Descrizione**: rappresentazione della gerarchia dei cluster mediante un *grafo ad albero*. Ogni ramo rappresenta un cluster e la lunghezza del ramo indica la distanza tra i cluster uniti.
- **Interpretazione**:
  - I punti di fusione mostrano a quale distanza i cluster vengono uniti.

Dendrogramma

- **Descrizione**: rappresentazione della gerarchia dei cluster mediante un *grafo ad albero*. Ogni ramo rappresenta un cluster e la lunghezza del ramo indica la distanza tra i cluster uniti.
- **Interpretazione**:
  - I punti di fusione mostrano a quale distanza i cluster vengono uniti.
  - Consente di visualizzare il numero di cluster a un certo livello di similarità.

Metodi agglomerativi (AGNES)

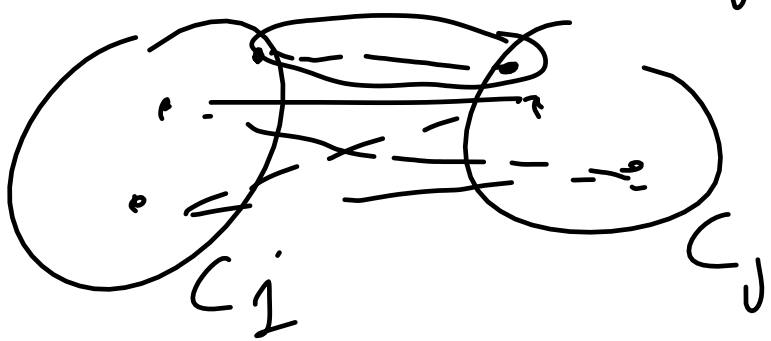
- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):



Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum

$$\ell(C_i, C_j) = \min_{\substack{x \in C_i \\ y \in C_j}} \|x - y\|$$



Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete

Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete
  - average

$$\ell_{avg}(C_i, C_j) = \frac{1}{\#C_i \# C_j} \sum_{\substack{x \in C_i \\ y \in C_j}} \|x - y\|$$

Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete
  - average
- **Algoritmo**:

Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete
  - average
- **Algoritmo**:
  1. Calcola la matrice delle similarità tra tutti i cluster

Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete
  - average
- **Algoritmo**:
  1. Calcola la matrice delle similarità tra tutti i cluster
  2. Unisci i due cluster più vicini.

Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete
  - average
- **Algoritmo**:
  1. Calcola la matrice delle similarità tra tutti i cluster
  2. Unisci i due cluster più vicini.
  3. Aggiorna la matrice delle distanze.

Metodi agglomerativi (AGNES)

- **Descrizione**: "AGglomerative NESting" (AGNES) inizia con ogni punto come un cluster e fonde iterativamente i cluster più simili tramite un *criterio di collegamento* (linkage criterion):
  - minimum
  - maximum/complete
  - average
- **Algoritmo**:
  1. Calcola la matrice delle similarità tra tutti i cluster
  2. Unisci i due cluster più vicini.
  3. Aggiorna la matrice delle distanze.
  4. Ripeti fino a ottenere un singolo cluster che contiene tutti i punti.

Metodi divisivi (DIANA)

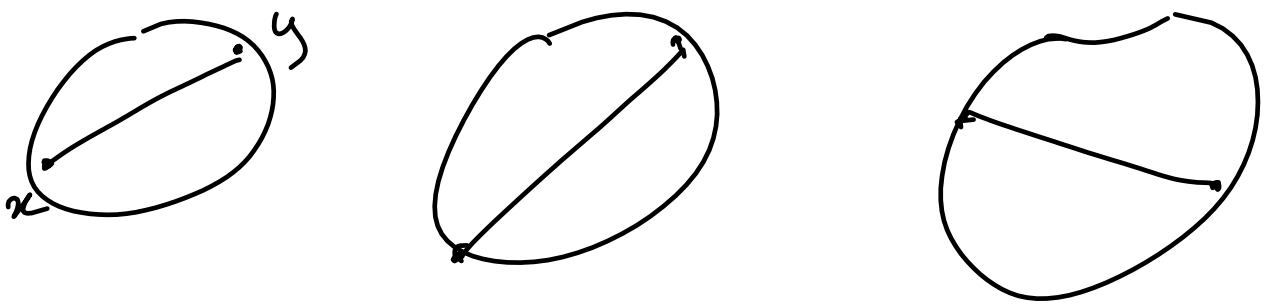
- **Descrizione**: “DIvisive ANAlysis” (DIANA) inizia con un unico cluster che contiene tutti i punti e divide iterativamente in sottogruppi.

Metodi divisivi (DIANA)

- **Descrizione**: “DIvisive ANAlysis” (DIANA) inizia con un unico cluster che contiene tutti i punti e divide iterativamente in sottogruppi.
- **Procedure**:

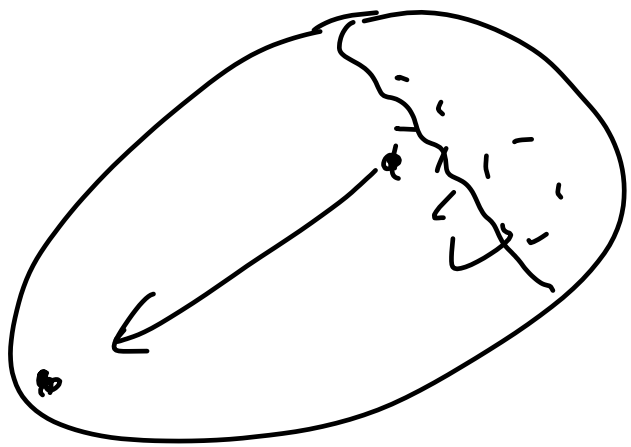
Metodi divisivi (DIANA)

- **Descrizione**: "DIvisive ANAlysis" (DIANA) inizia con un unico cluster che contiene tutti i punti e divide iterativamente in sottogruppi.
- **Procedure**:
  1. Identifica il cluster da dividere: quello con *diametro* maggiore



Metodi divisivi (DIANA)

- **Descrizione**: "DIvisive ANAlysis" (DIANA) inizia con un unico cluster che contiene tutti i punti e divide iterativamente in sottogruppi.
- **Procedure**:
  1. Identifica il cluster da dividere: quello con *diametro* maggiore
  2. Divide il cluster in due cluster distinti partendo dal punto più *dissimile* dagli altri e associandogli i punti più *simili* ad esso che ai rimanenti.



Metodi divisivi (DIANA)

- **Descrizione**: "DIvisive ANAlysis" (DIANA) inizia con un unico cluster che contiene tutti i punti e divide iterativamente in sottogruppi.
- **Procedure**:
  1. Identifica il cluster da dividere: quello con *diametro* maggiore
  2. Divide il cluster in due cluster distinti partendo dal punto più *dissimile* dagli altri e associandogli i punti più *simili* ad esso che ai rimanenti.
  3. Ripete fino a quando ogni punto è in un cluster individuale o fino a un numero desiderato di cluster.

Selezione del numero di clusters

In tutti i metodi rimane la domanda: come determinare k?

- In breve: *Non c'è un k "giusto", dipende dall'utilizzo che si farà dei cluster trovati!*

Selezione del numero di clusters

In tutti i metodi rimane la domanda: come determinare k?

- In breve: *Non c'è un k "giusto", dipende dall'utilizzo che si farà dei cluster trovati!*
- Questo porta all'esistenza di svariati *criteri* per la selezione di k. Discutiamo:

Selezione del numero di clusters

In tutti i metodi rimane la domanda: come determinare k?

- In breve: *Non c'è un k "giusto", dipende dall'utilizzo che si farà dei cluster trovati!*
- Questo porta all'esistenza di svariati *criteri* per la selezione di k. Discutiamo:
  1. Elbow (gomito)

Selezione del numero di clusters

In tutti i metodi rimane la domanda: come determinare k?

- In breve: *Non c'è un k "giusto", dipende dall'utilizzo che si farà dei cluster trovati!*
- Questo porta all'esistenza di svariati *criteri* per la selezione di k. Discutiamo:
  1. Elbow (gomito)
  2. Indice di Dunn

Selezione del numero di clusters

In tutti i metodi rimane la domanda: come determinare k?

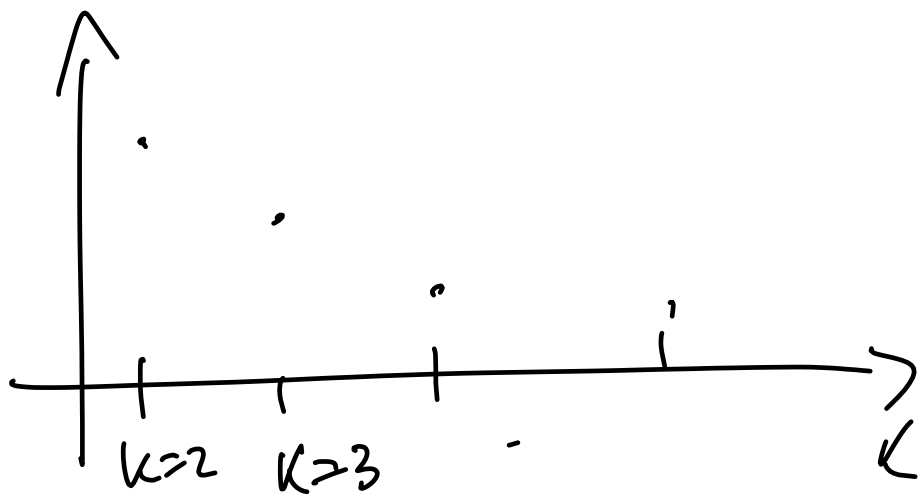
- In breve: *Non c'è un k "giusto", dipende dall'utilizzo che si farà dei cluster trovati!*
- Questo porta all'esistenza di svariati *criteri* per la selezione di k. Discutiamo:
  1. Elbow (gomito)
  2. Indice di Dunn
  3. Silhouette

Elbow "non criterio"

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:

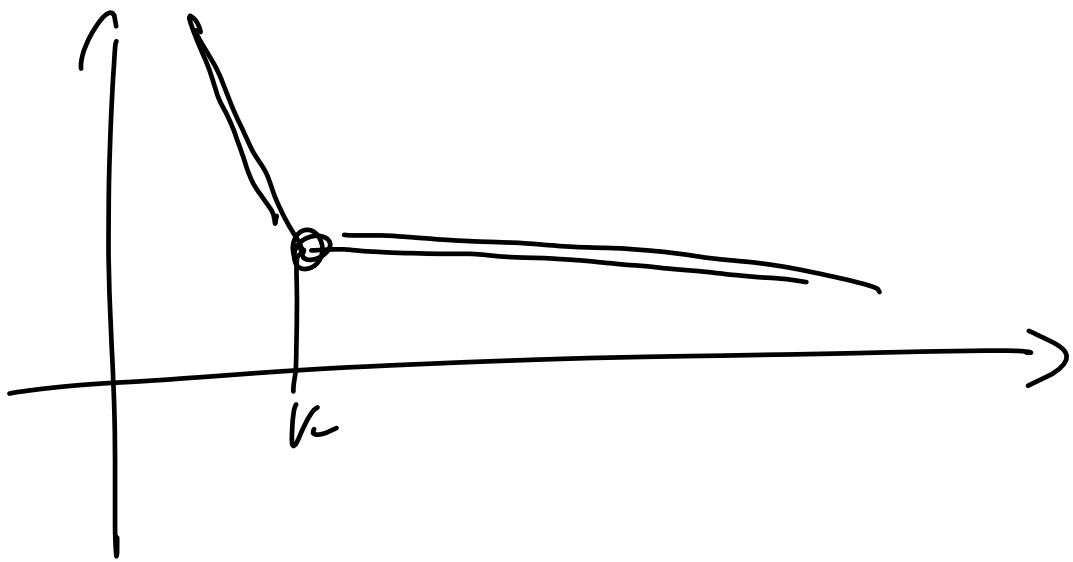
Elbow

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:
  - al crescere di k la $WCSS_k$ diminuisce



Elbow

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:
  - al crescere di k la $WCSS_k$ diminuisce
  - si cerca il punto k_{elbow} dove l'incremento $WCSS_k - WCSS_{k+1}$ *comincia a diminuire*.



Elbow

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:
  - al crescere di k la $WCSS_k$ diminuisce
  - si cerca il punto k_{elbow} dove l'incremento $WCSS_k - WCSS_{k+1}$ *comincia a diminuire*.
- **Intuizione**: separando *veri* cluster si ottiene un maggior incremento nella *WCSS*!

Elbow

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:
  - al crescere di k la $WCSS_k$ diminuisce
  - si cerca il punto k_{elbow} dove l'incremento $WCSS_k - WCSS_{k+1}$ *comincia a diminuire*.
- **Intuizione**: separando *veri* cluster si ottiene un maggior incremento nella *WCSS*!
- **Problemi**:

Elbow

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:
  - al crescere di k la $WCSS_k$ diminuisce
  - si cerca il punto k_{elbow} dove l'incremento $WCSS_k - WCSS_{k+1}$ *comincia a diminuire*.
- **Intuizione**: separando *veri* cluster si ottiene un maggior incremento nella *WCSS*!
- **Problemi**:
  - non è ben definito il punto di gomito

Elbow

- Il metodo del *gomito* consiste nel confrontare le inerzie $WCSS_k$ ottimali per diversi valori di k:
  - al crescere di k la $WCSS_k$ diminuisce
  - si cerca il punto k_{elbow} dove l'incremento $WCSS_k - WCSS_{k+1}$ *comincia a diminuire*.
- **Intuizione**: separando *veri* cluster si ottiene un maggior incremento nella *WCSS*!
- **Problemi**:
  - non è ben definito il punto di gomito
  - dataset molto diversi danno grafici simili

Richiede di calcolare $WCSS_K$ per vari K

Indice di Dunn

- Misura la qualità del clustering confrontando la *distanza minima tra i cluster* con la distanza massima (*diametro*) all'interno dello stesso cluster.

Indice di Dunn

- Misura la qualità del clustering confrontando la *distanza minima tra i cluster* con la distanza massima (*diametro*) all'interno dello stesso cluster.
- **Definizioni**: dato un cluster $(C_j)_{j=1,\dots,k}$, poste

$$\text{diametro}(C_j) := \max_{x,y \in C_j} \|x-y\|, \quad \text{distanza}(C_i, C_j) := \min_{x \in C_i, y \in C_j} \|x-y\|$$

definiamo

$$\text{DunnIndex}((C_j)_{j=1,\dots,k}) := \frac{\min_{i \neq j} \text{distanza}(C_i, C_j)}{\max_i \text{diametro}(C_k)}$$

Indice di Dunn

- Misura la qualità del clustering confrontando la *distanza minima tra i cluster* con la distanza massima (*diametro*) all'interno dello stesso cluster.
- **Definizioni**: dato un cluster $(C_j)_{j=1,\dots,k}$, poste

  $$\text{diametro}(C_j) := \max_{x,y \in C_j} \|x-y\|, \quad \text{distanza}(C_i, C_j) := \min_{x \in C_i, y \in C_j} \|x-y\|$$

  definiamo

  $$\text{DunnIndex}((C_j)_{j=1,\dots,k}) := \frac{\min_{i \neq j} \text{distanza}(C_i, C_j)}{\max_i \text{diametro}(C_k)}$$

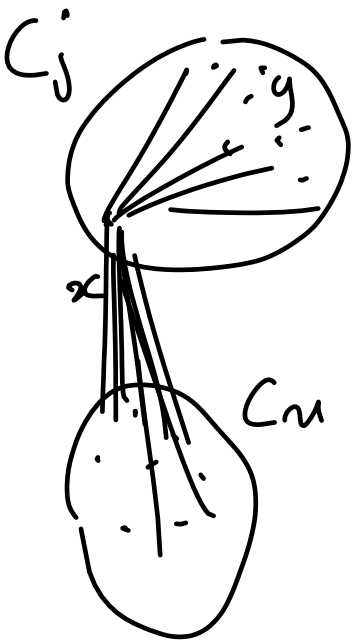
- **Interpretazione**: Un indice di Dunn più alto indica cluster più separati e ben definiti. Valori più alti sono preferibili.

Silhouette

- Misura quanto un punto x è simile a quelli del proprio cluster C_j rispetto ai cluster vicini. Viene calcolata per ogni punto e poi viene fatta una media.

Silhouette

- Misura quanto un punto x è simile a quelli del proprio cluster C_j rispetto ai cluster vicini. Viene calcolata per ogni punto e poi viene fatta una media.
- **Definitioni**: dato $x \in C_j$,



$$\textit{(adesione)} \quad a(x) := \frac{1}{\sharp C_j - 1} \sum_{y \in C_j, y \neq x} \|x - y\|$$

$$\textit{(separazione)} \quad b(x) := \min_{u \neq j} \frac{1}{\sharp C_u} \sum_{y \in C_u} \|x - y\|$$

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- Silhouette media per il cluster C_i:

$$s(C_i) := \frac{1}{\sharp C_i} \sum_{x \in C_i} s(x)$$

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- Silhouette media per il cluster C_i:

$$s(C_i) := \frac{1}{\sharp C_i} \sum_{x \in C_i} s(x)$$

- **Silhouette media** (score):

$$s((C_i)_{i=1,\dots,k}) = \frac{1}{k} \sum_{i=1}^{k} s(C_i)$$

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.
  - $s(x) < 0 \rightarrow x$ potrebbe essere assegnato al cluster sbagliato.

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.
  - $s(x) < 0 \rightarrow x$ potrebbe essere assegnato al cluster sbagliato.
- Valori di Silhouette media (score) :

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.
  - $s(x) < 0 \rightarrow x$ potrebbe essere assegnato al cluster sbagliato.
- Valori di Silhouette media (score) :
  - $s > 0.7$ clustering *forte*

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.
  - $s(x) < 0 \rightarrow x$ potrebbe essere assegnato al cluster sbagliato.
- Valori di Silhouette media (score) :
  - $s > 0.7$ clustering *forte*
  - $0.5 < s < 0.7$ clustering *buono*

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.
  - $s(x) < 0 \rightarrow x$ potrebbe essere assegnato al cluster sbagliato.
- Valori di Silhouette media (score) :
  - $s > 0.7$ clustering *forte*
  - $0.5 < s < 0.7$ clustering *buono*
  - $s > 0.25$ clustering *debole*

$$\textit{(silhouette)} \quad s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

- **Interpretazione**:
  - $s(x) \approx 1 \rightarrow x$ è ben assegnati al suo cluster.
  - $s(x) \approx 0 \rightarrow x$ è vicino al *bordo* dei cluster.
  - $s(x) < 0 \rightarrow x$ potrebbe essere assegnato al cluster sbagliato.
- Valori di Silhouette media (score) :
  - $s > 0.7$ clustering *forte*
  - $0.5 < s < 0.7$ clustering *buono*
  - $s > 0.25$ clustering *debole*
- Attenzione alla **curse of dimensionality**!

Final Clustering

- Giocate voi!
  https://dario-trevisan.shinyapps.io/FinalClustering/

Final Clustering

- Giocate voi!
  https://dario-trevisan.shinyapps.io/FinalClustering/
- Codice (R shiny) disponibile nel team e nella pagina del corso