



ANTICIPARE LA CRESCITA CON LE NUOVE COMPETENZE SUI BIG DATA – EDIZIONE 2

Operazione Rif. PA 2019-11596/RER “Anticipare la crescita con le nuove competenze sui Big Data”, approvata dalla Regione Emilia-Romagna con DGR n° 789 del 20 maggio 2019 e co-finanziata dal Fondo Sociale Europeo PO 2014-2020



**POLITECNICO
MILANO 1863**
POLO TERRITORIALE DI
PIACENZA



UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA



**UNIVERSITÀ
CATTOLICA
del Sacro Cuore**



**Università
degli Studi
di Ferrara**



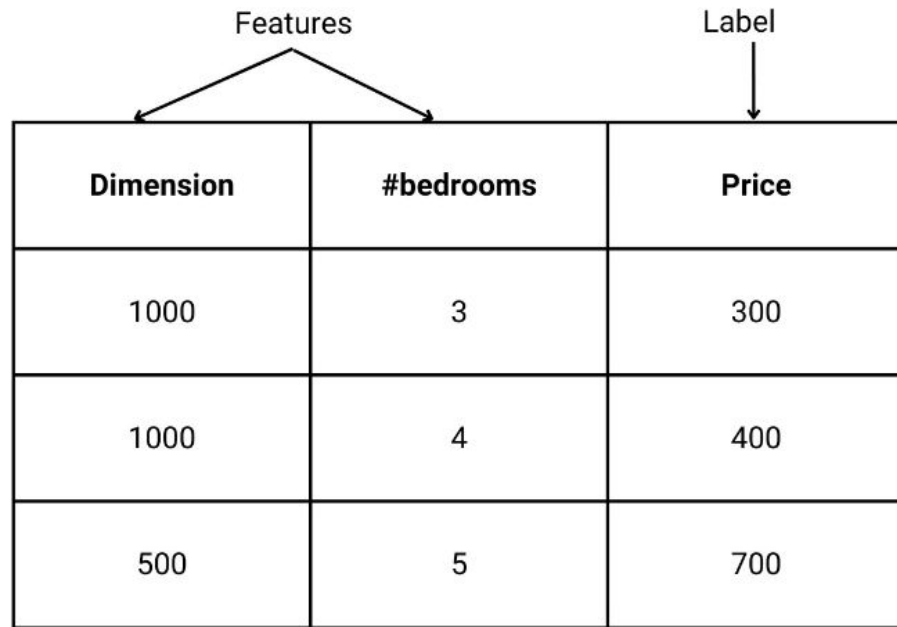
**UNIVERSITÀ
DI PARMA**

Programma della lezione

- Supervised Learning Vs Unsupervised Learning
- Clustering
- K-Means
- DBSCAN

Supervised Learning

- Finora abbiamo imparato modelli di ML che imparano prendendo in input **dati etichettati**
 - Esempio: prevedere il prezzo della casa data la dimensione e il numero di letti
- Questi tipi di modelli si chiamano **supervisionati** per questo motivo



Features		Label
Dimension	#bedrooms	Price
1000	3	300
1000	4	400
500	5	700

Unsupervised Learning

- Questo paradigma cambia con il unsupervised learning
- Un modello **non supervisionato** impara a estrarre informazioni dei dati
 - senza ricevere risposte da cui imparare
- Esempio: suddividere e dividere i clienti in gruppi in base a caratteristiche specifiche, per permettere all'azienda di lanciare campagne di marketing specializzate

Classical Machine Learning

Task Driven

Supervised Learning
(Pre Categorized Data)



Classification

(Divide the
socks by Color)

Eg. Identity
Fraud Detection



Regression

(Divide the
Ties by Length)

Eg. Market
Forecasting

Data Driven

Unsupervised Learning
(Unlabelled Data)



Clustering

(Divide by
Similarity)

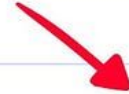
Eg. Targeted
Marketing



Association

(Identify
Sequences)

Eg. Customer
Recommendation



Dimensionality
Reduction

(Wider
Dependencies)

Eg. Big Data
Visualization

Obj: Predications & Predictive Models

Pattern/ Structure Recognition

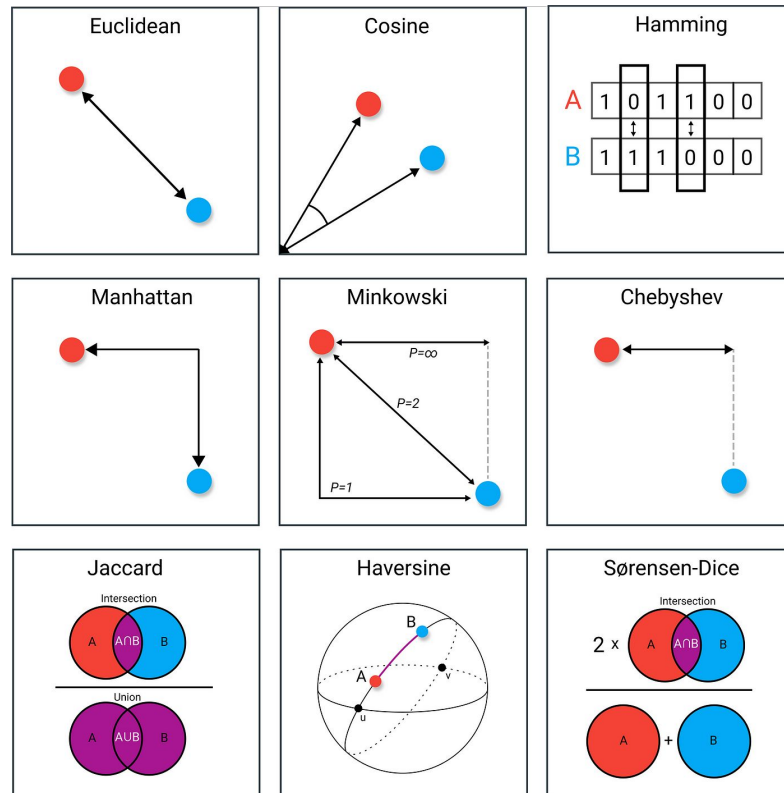


Clustering

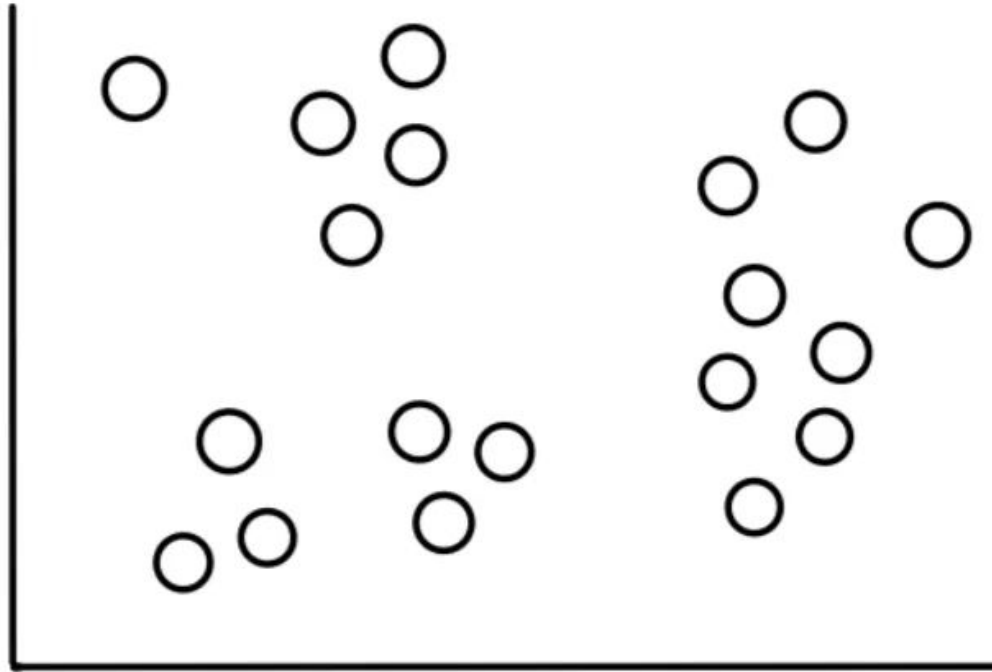
- Trovare similarità nei dati in base a caratteristiche presenti e raggruppare i dati simili in cluster
- Applicazioni
 - Biologia
 - Marketing
 - Clima
 - Economia
 - Geologia
- Tool di pre-processing
 - Sintesi di dati
 - Compressione
 - Outlier detection

Come si valuta la qualità del clustering

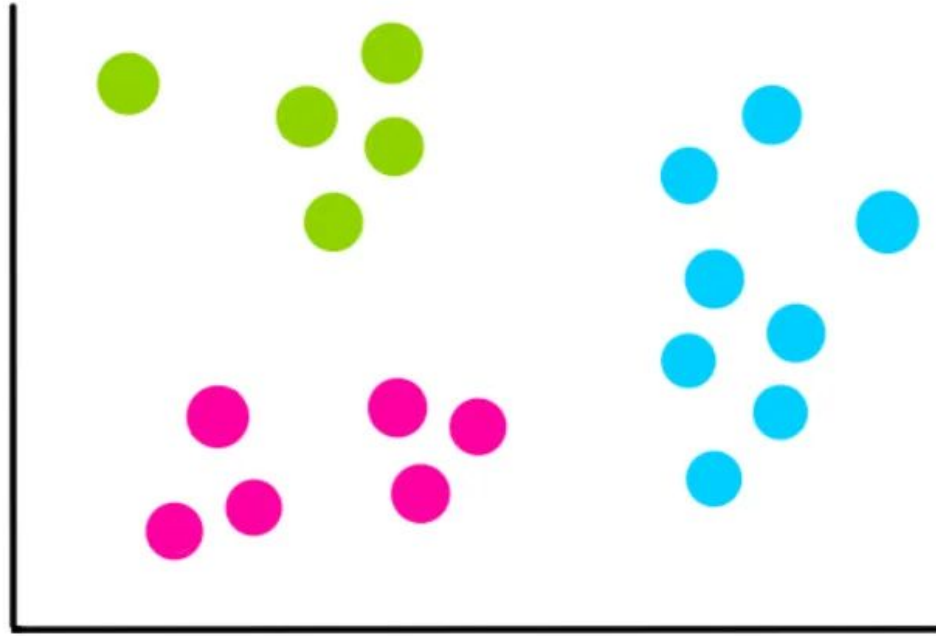
- Qualità del clustering
 - Alta similarità intra-custer
 - Alta dissimilarità inter-cluster
- La similarità viene espressa tramite una funzione di distanza
 - che cambia in base al tipo di dato



Esempio: vogliamo raggruppare 19 osservazioni



Esempio: vogliamo raggruppare 19 osservazioni



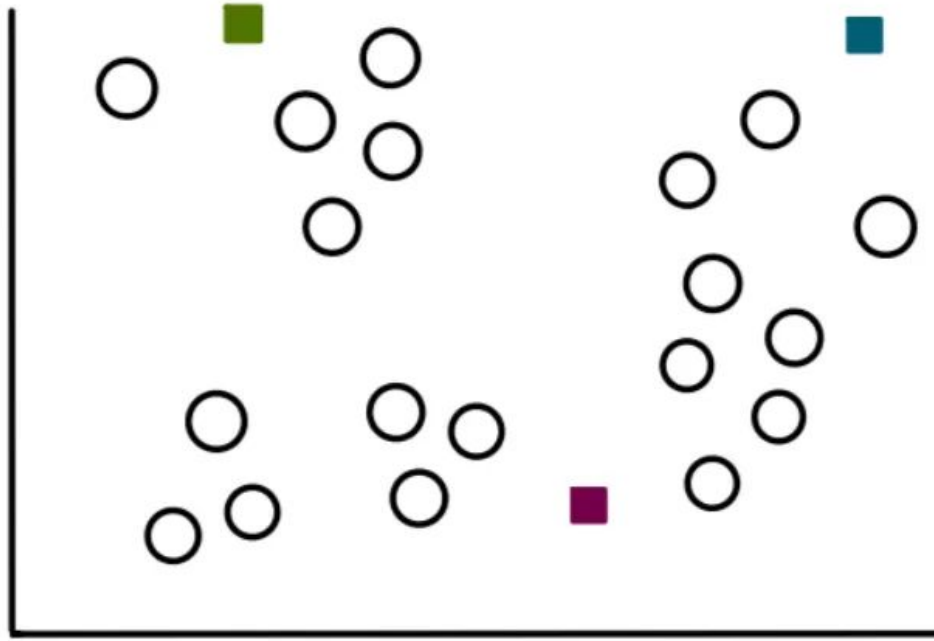
K-means Clustering

- Perché questo nome?
 - divide i dati in K gruppi, o clusters
 - Quindi k rappresenta il numero di clusters che si passano in input al modello
 - Means perché ogni cluster è rappresentato da un centroide, un punto centrale, ottenuto facendo la media aritmetica di tutti i punti all'interno del cluster

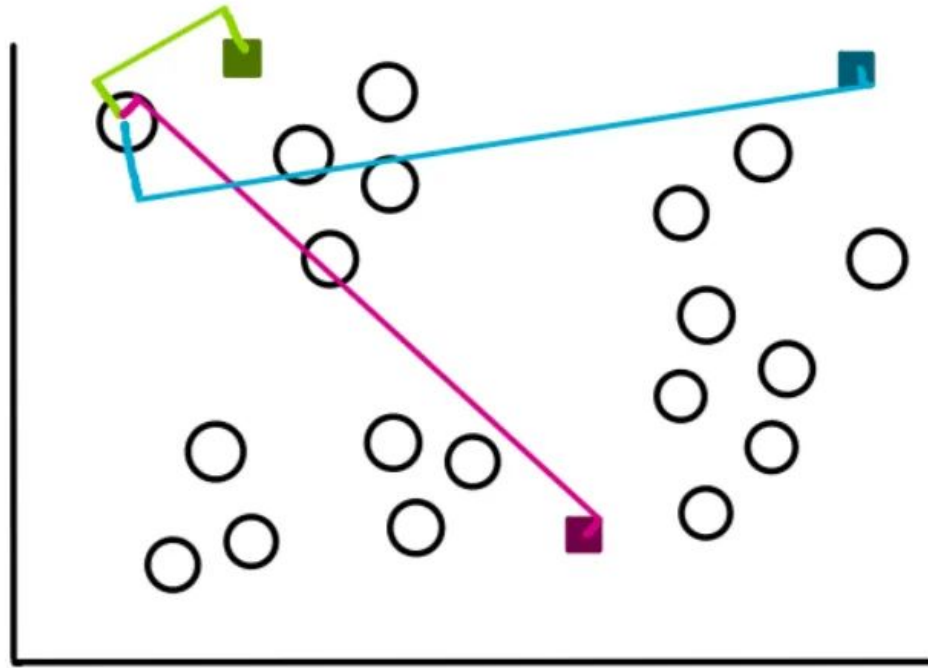
Step 1: Seleziono il numero di Clusters, k

- In questo caso, fissiamo $k=3$
 - Quindi vogliamo identificare 3 clusters

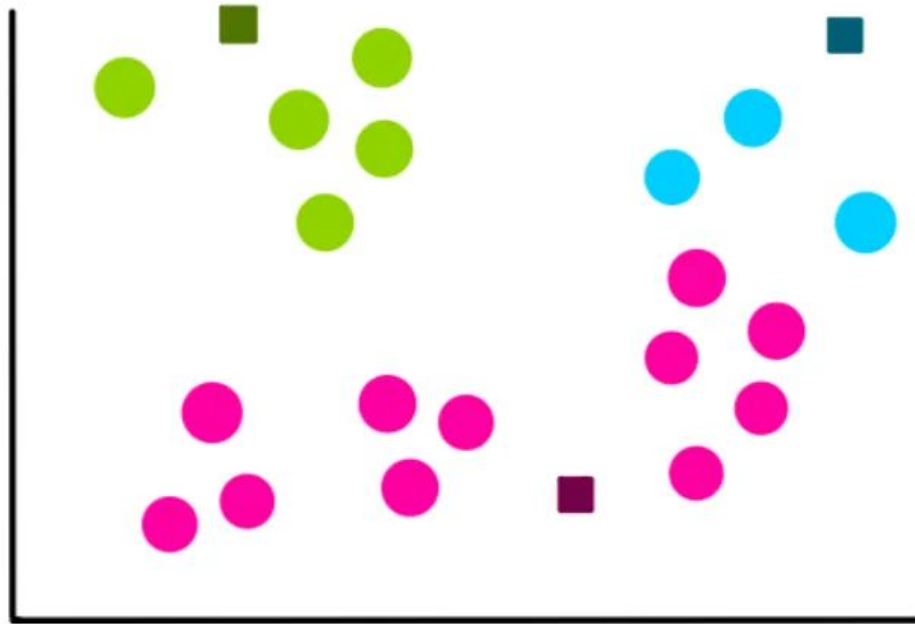
Step 2: Seleziono k punti in modo casuale



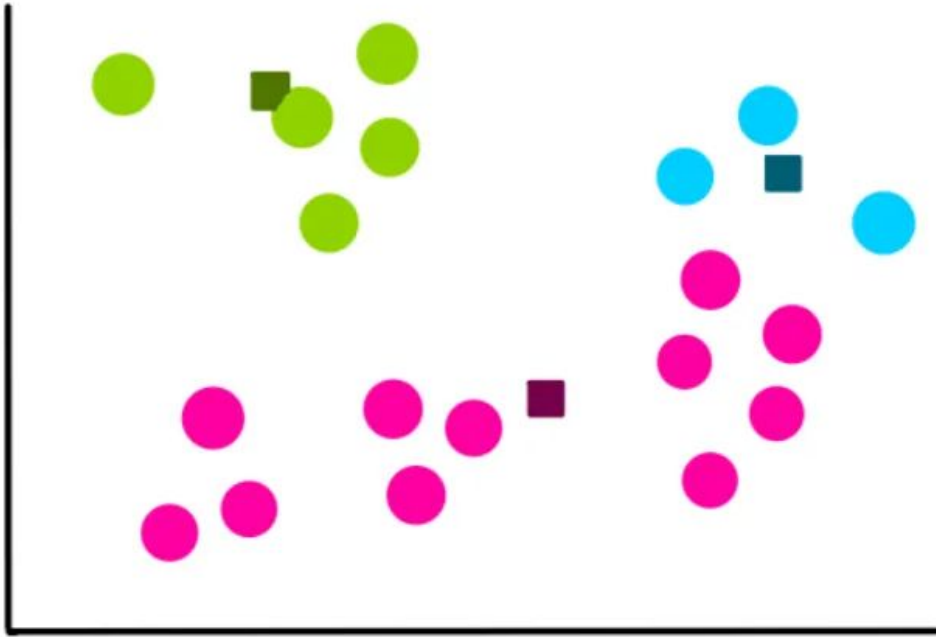
Step 3: Misuro la distanza tra ogni punto e il centroide



Step 4: Assegno ogni punto al cluster più vicino



Step 5: Calcolo il nuovo centroide di ogni cluster



$$(x'', y'') = \left(\frac{x_1 + \dots + x_n}{n}, \frac{y_1 + \dots + y_n}{n} \right)$$

$$(x'', y'') = \left(\frac{x_1 + \dots + x_5}{5}, \frac{y_1 + \dots + y_5}{5} \right)$$

Step 6: Valuto la qualità di ogni cluster

Calcoliamo la **Within-Cluster Sum of Squares**

$$WCSS = \sum_{C_k}^{C_n} \left(\sum_{d_i \text{ in } C_i}^{d_m} distance(d_i, C_k)^2 \right)$$

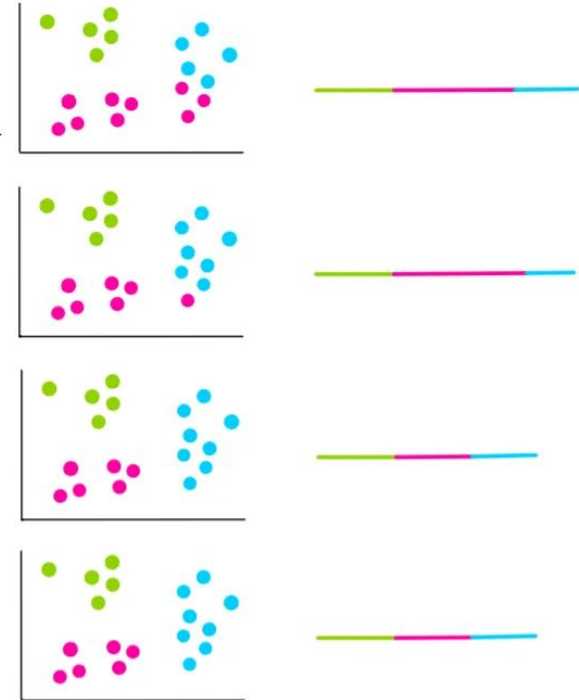
Where,

C is the cluster centroids and d is the data point in each Cluster.



Step 6: Ripeto Step 3 - 6

- L'algoritmo continuerà finché
 - I cluster con il minimo di WCSS non cambiano



Come scelgo k?

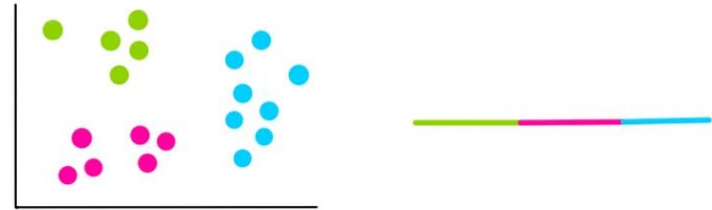
k=1:



k=2:



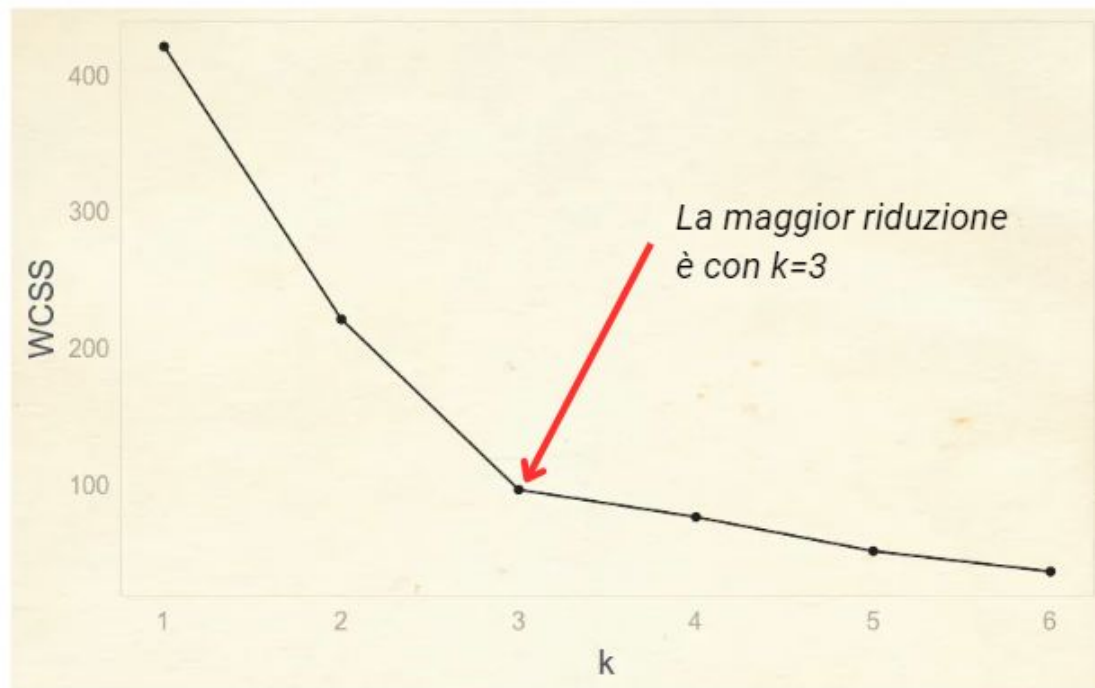
k=3:



k=4:



Elbow plot



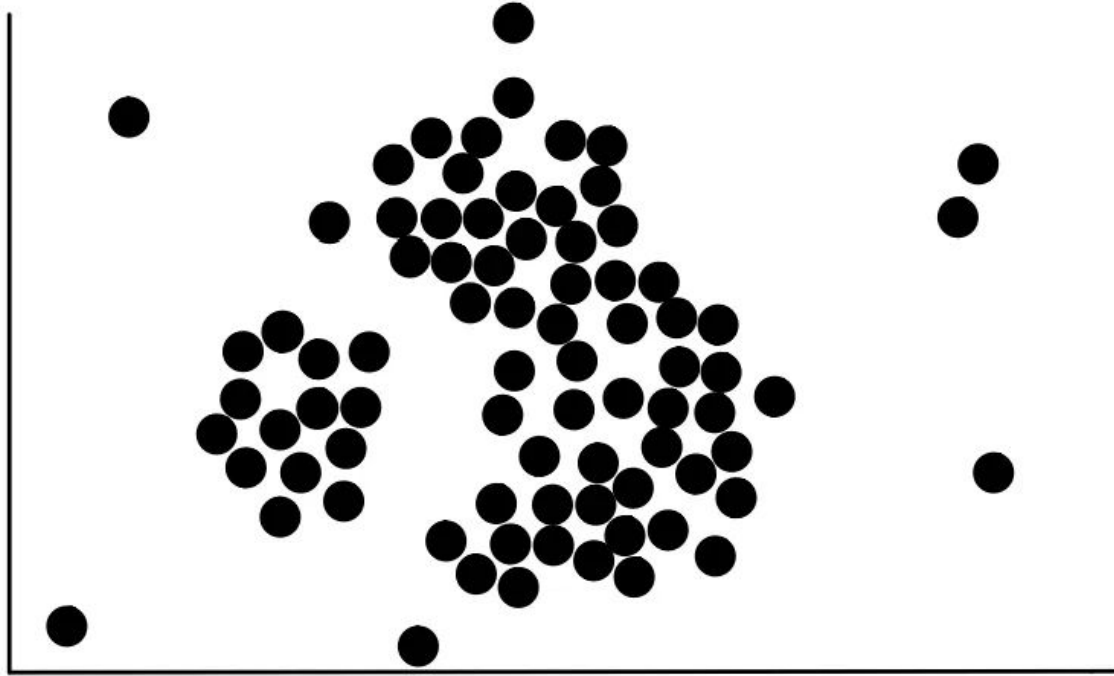
Vantaggi

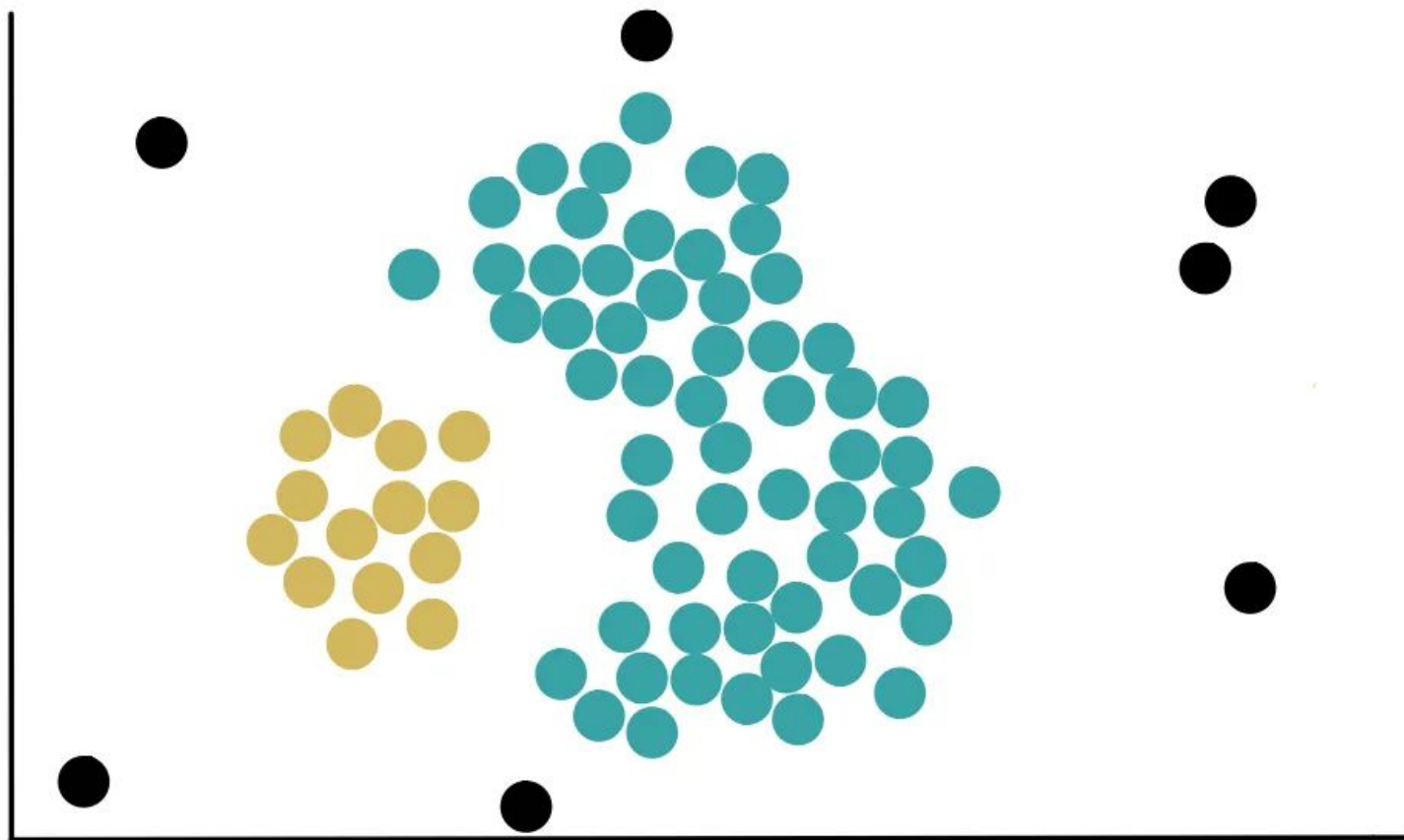
- Semplice
- Facile da applicare
- Ma ... ci sono anche svantaggi
 - Bisogna passare il numero di clusters k , non lo impara da solo
 - Assume che i cluster sono sferici, ma può capitare che i cluster abbiano diversa dimensione e densità

DBSCAN

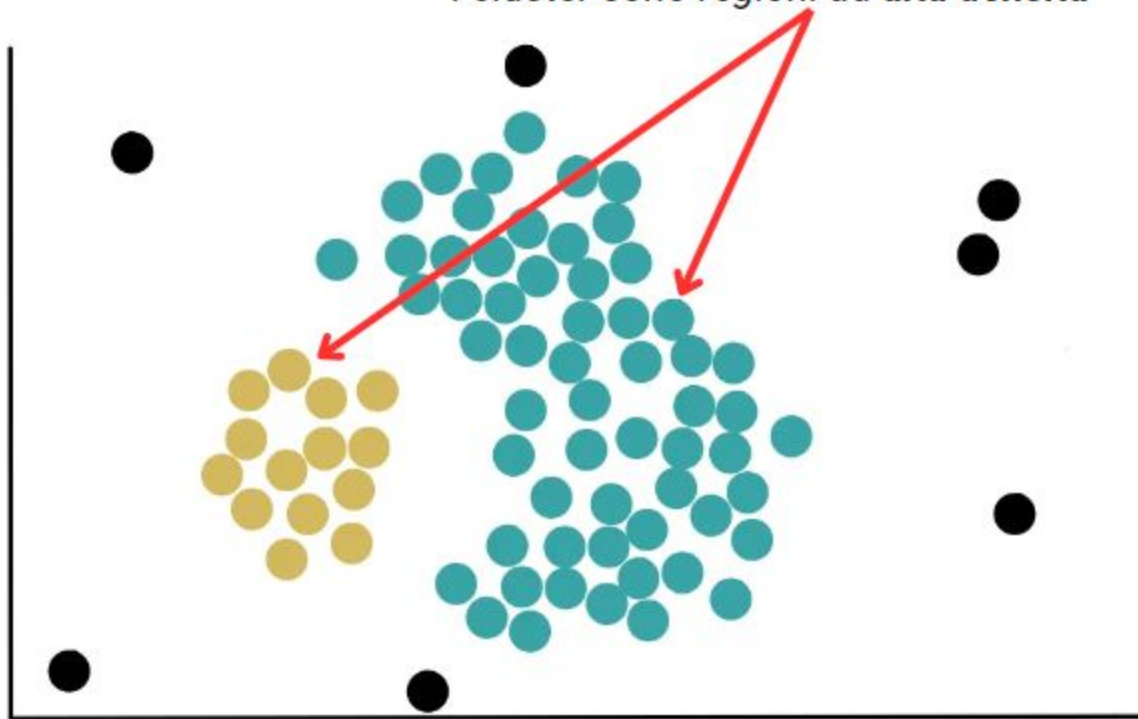
- DBSCAN è il diminutivo di **Density-based spatial clustering of applications with noise**
- Perché questo nome?
- Identifica i cluster basandosi sulla **densità** dei punti
- Assume che i cluster siano **regioni ad alta densità**
- Ci sono anche essere **outlier**, punti che non appartengono ad alcun cluster, che sono a bassa densità

Esempio: vogliamo raggruppare questi punti

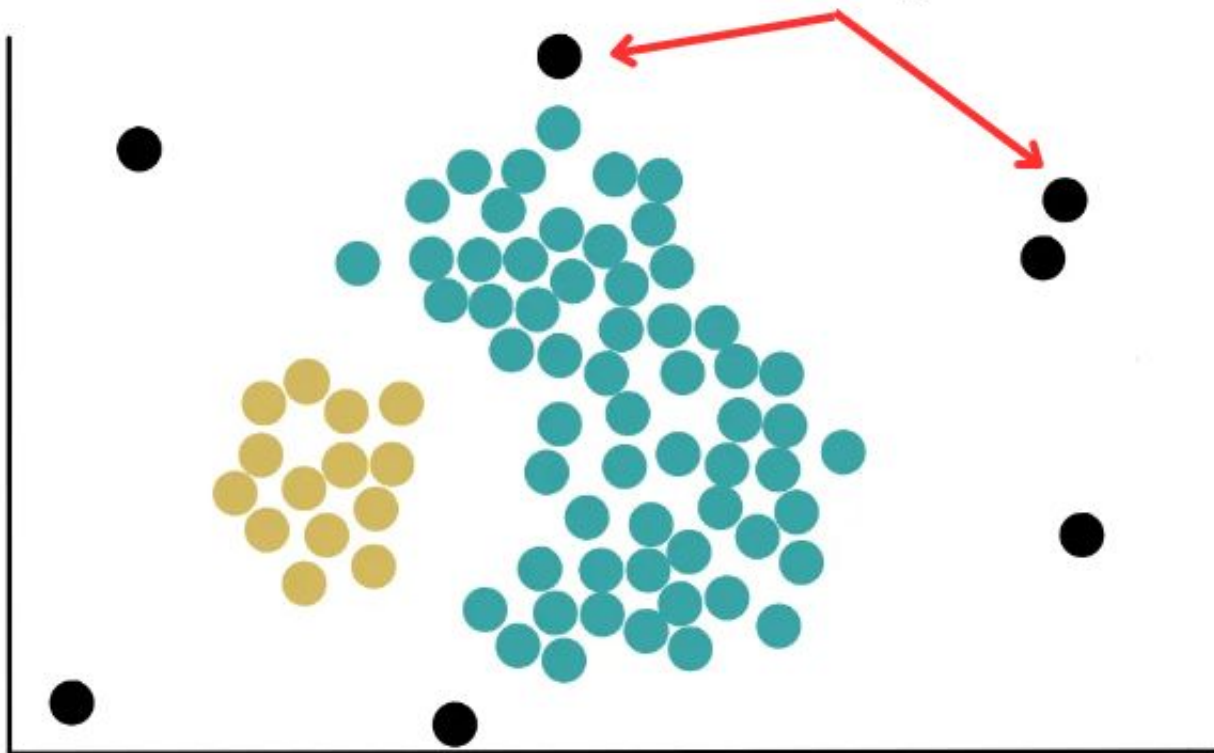




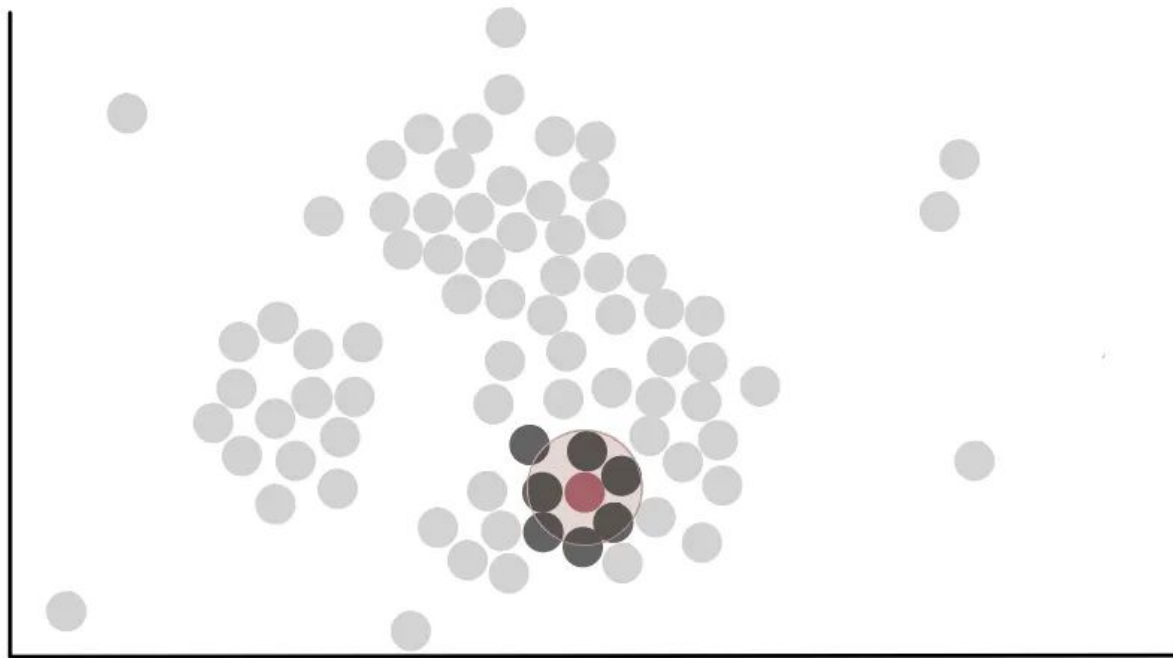
*I cluster sono regioni ad **alta densità***



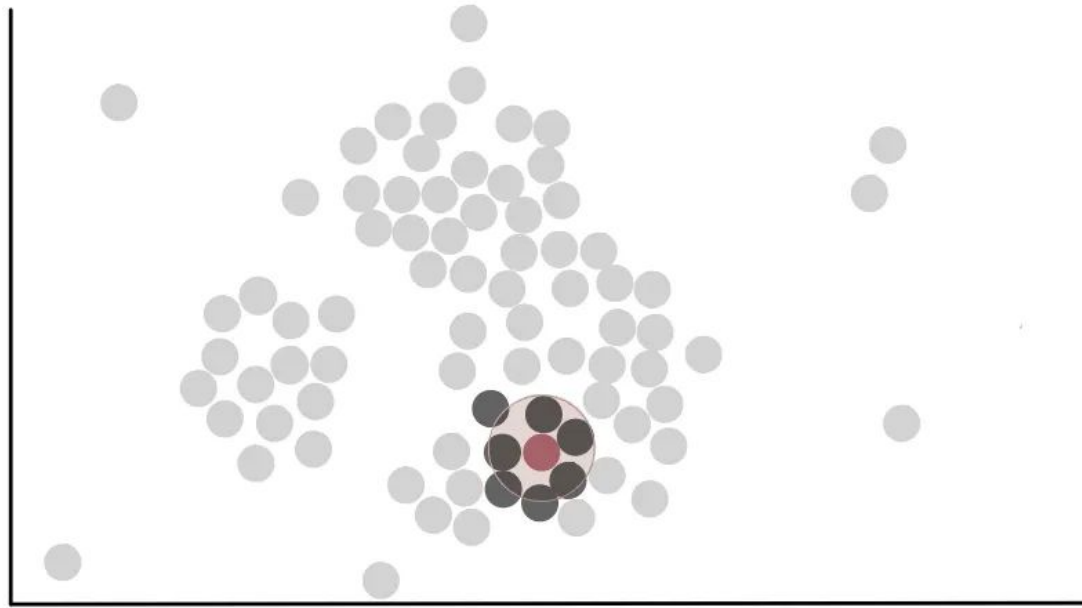
Gli outlier tendono a essere regioni ad **bassa densità**



Contare il numero di punti vicini ad ogni punto



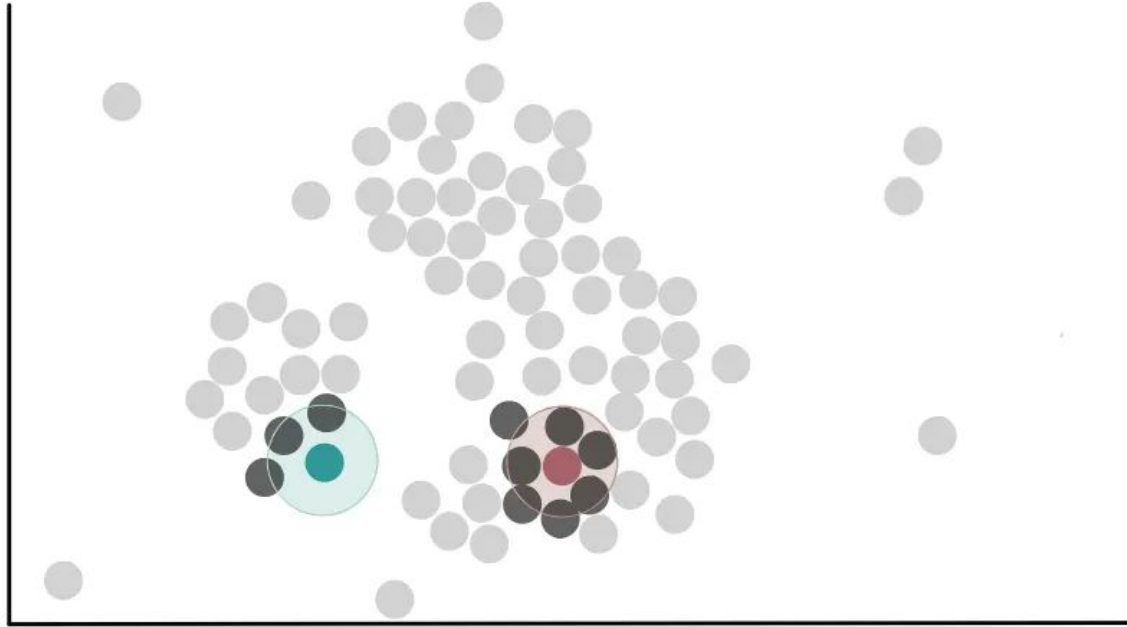
Contare il numero di punti vicini ad ogni punto



Nota:

- Il raggio del cerchio rosso, chiamato **eps**, è definito dall'utente

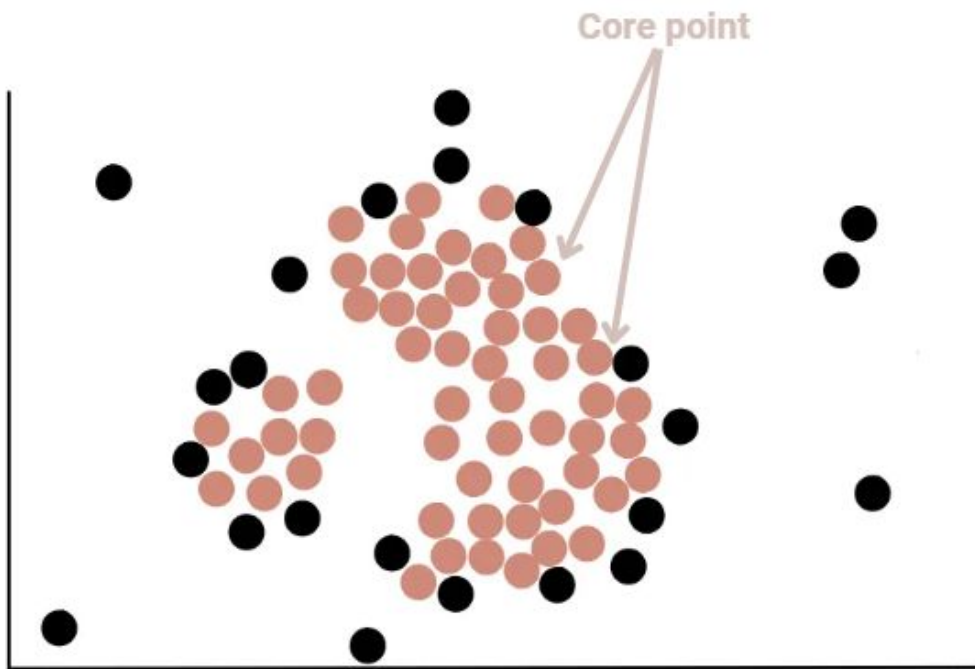
Contare il numero di punti vicini ad ogni punto



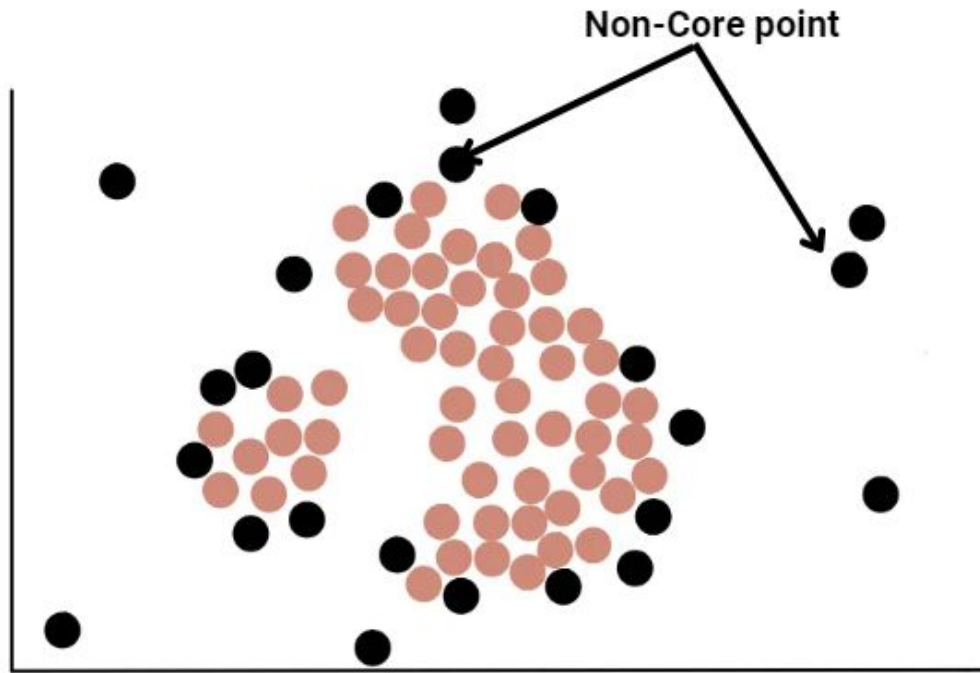
Nota:

- In questo esempio definiamo un core point come punto **vicino ad almeno 4 punti**.
- Come prima, questo numero di punti vicino al core point è definito dall'utente

Seguendo la definizione di prima ...



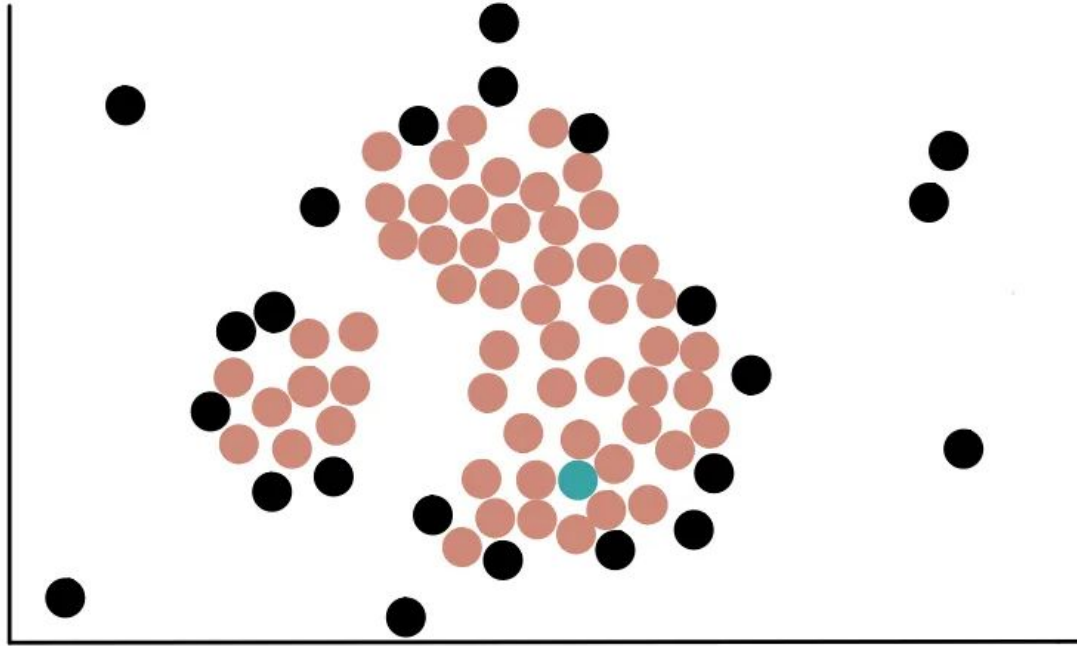
Seguendo la definizione di prima ...



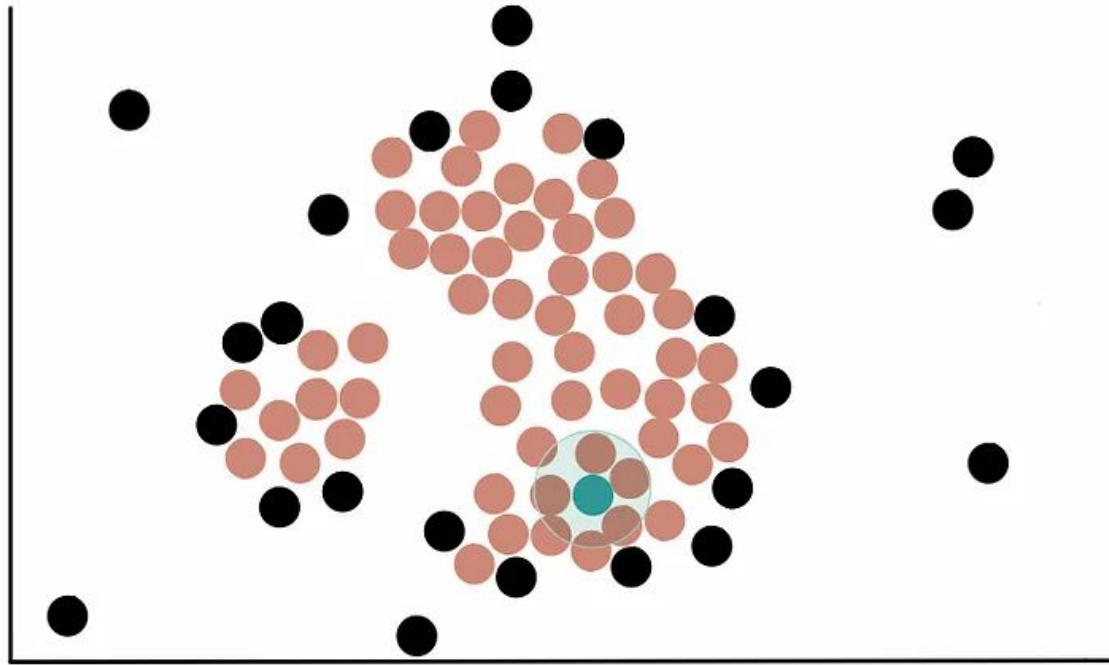
Notazione

- Eps è il raggio del cerchio rosso
- Core point è un punto che vicino ad almeno un numero definito di punti
- Non-Core points sono i punti rimanenti

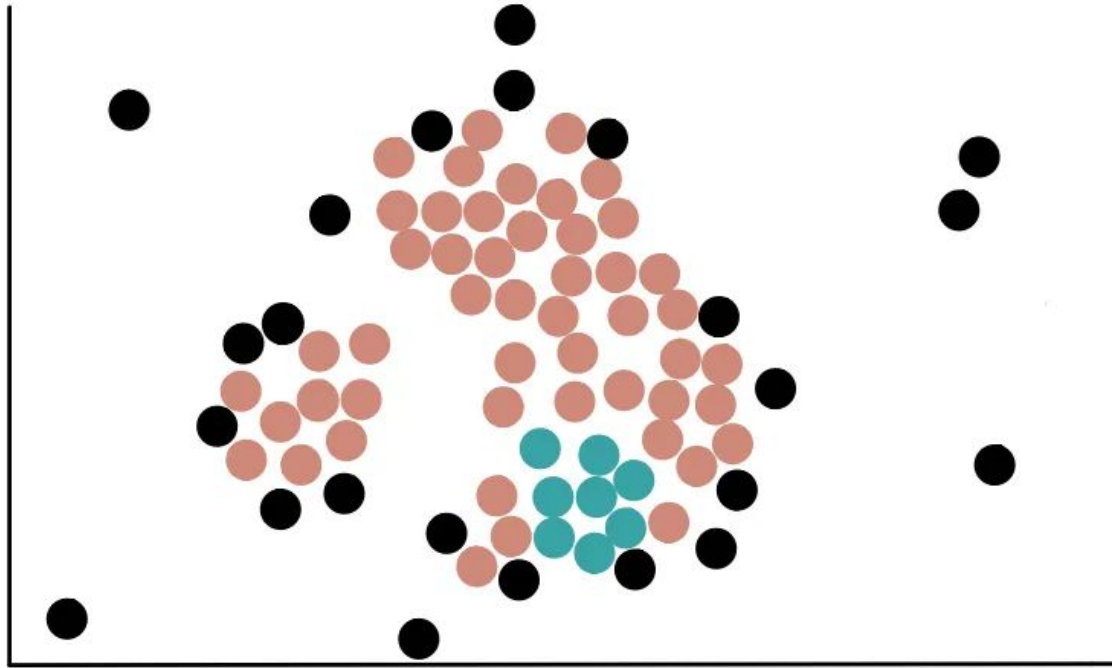
Step 1: Prendo un core point in modo casuale e lo assegno al primo cluster



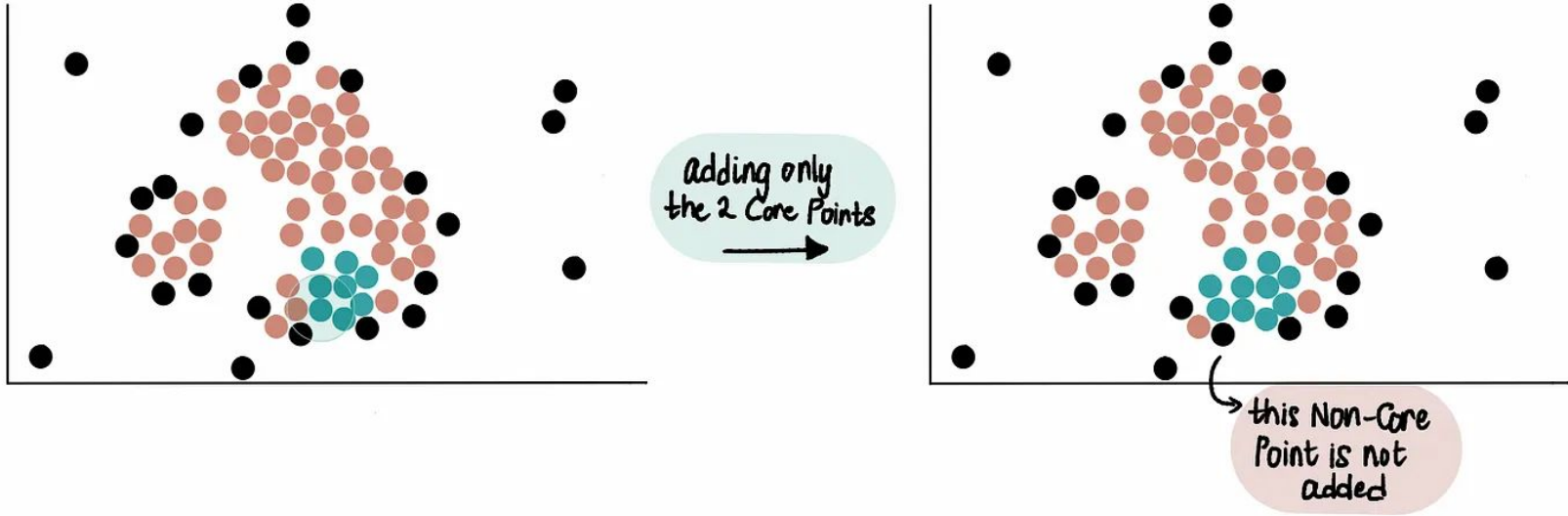
Step 2: Assegno al primo cluster i punti vicini al core point



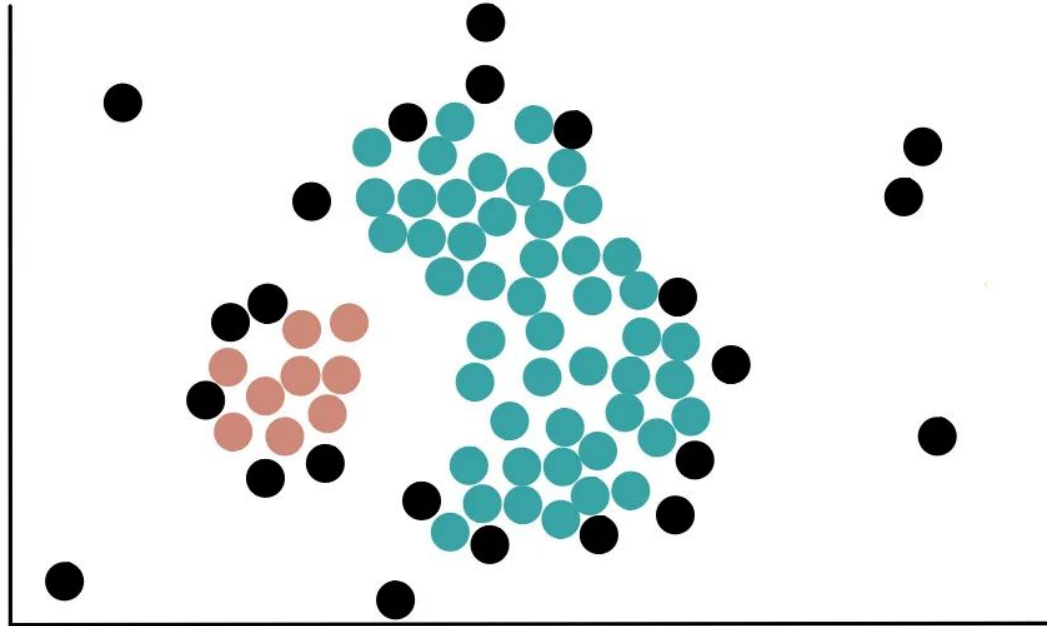
Step 2: Assegno al primo cluster i punti vicini al core point



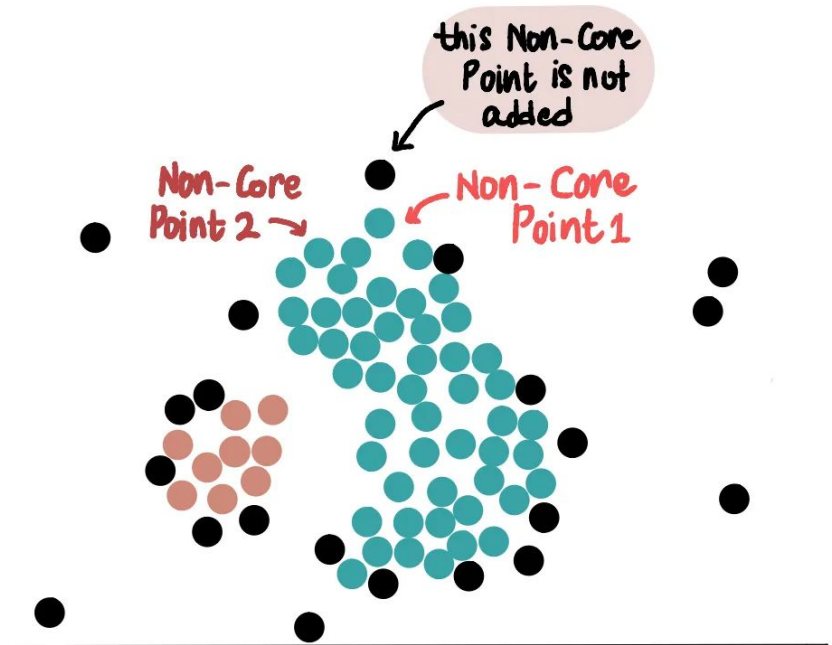
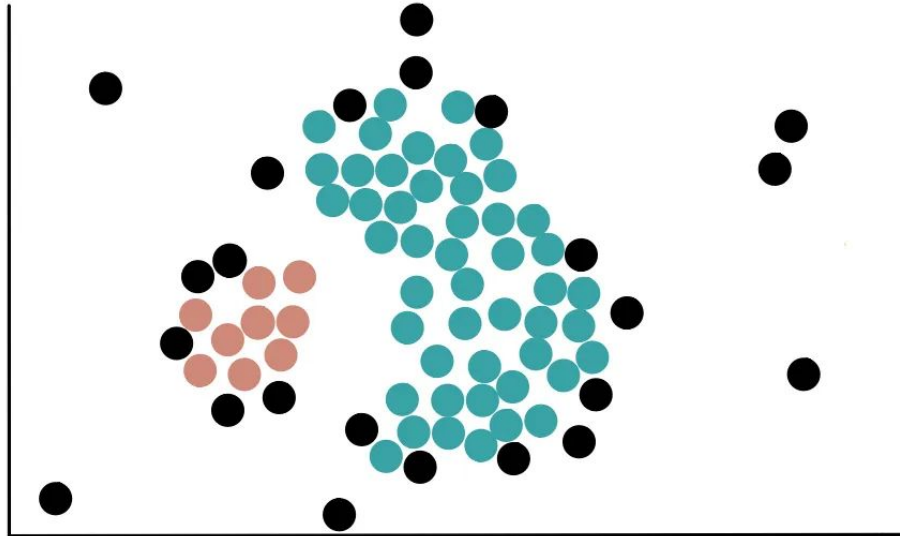
Step 3: Si aggiungono i core point vicini al primo cluster



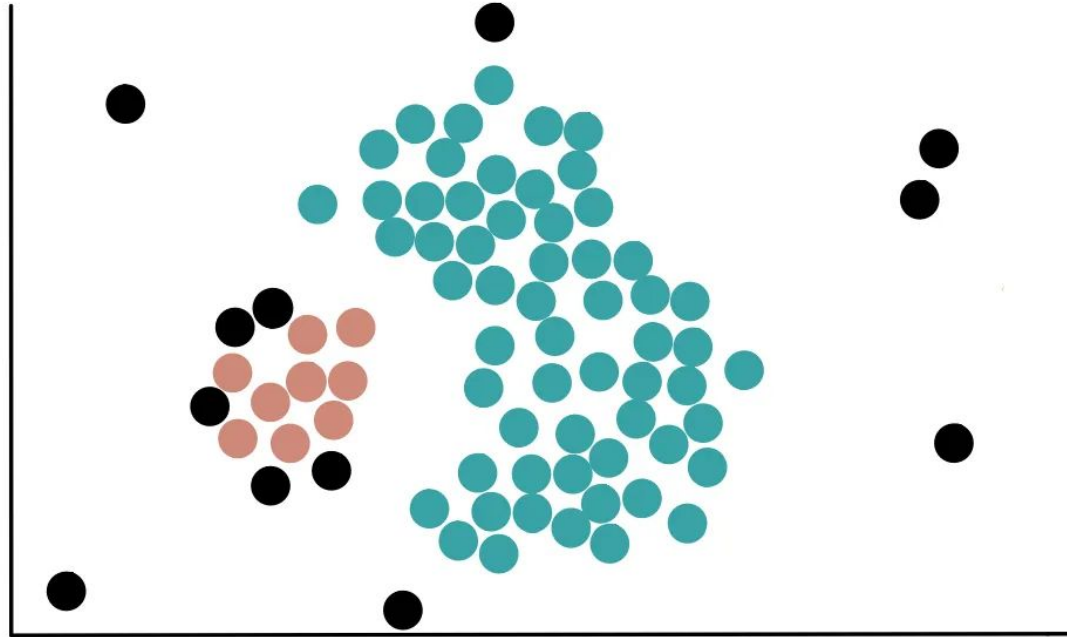
Step 4: Si continua ad aggiungere i **core points** vicini al primo cluster



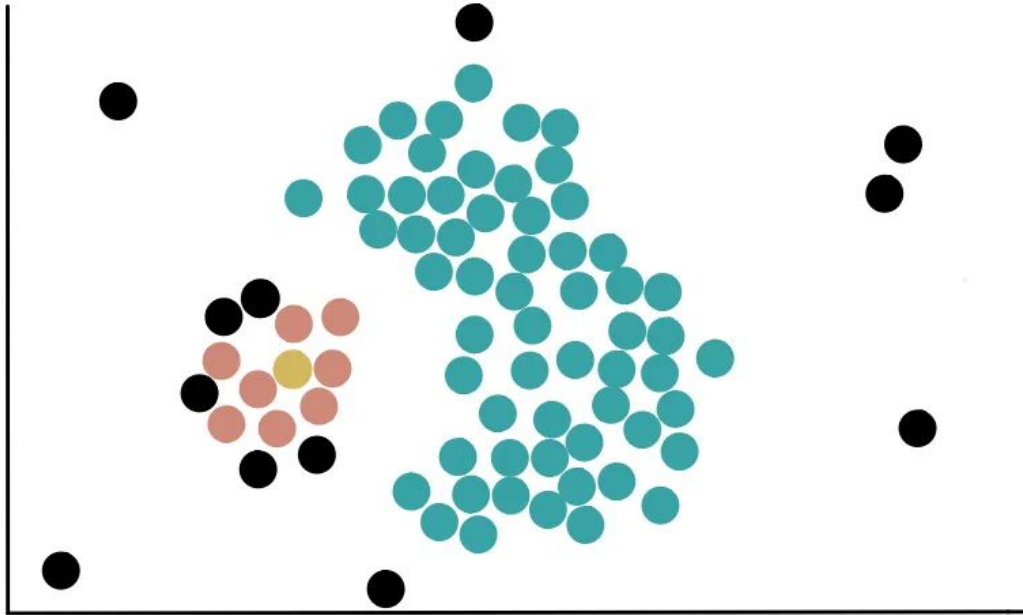
Step 5: Si aggiungono al cluster i **non core points** vicini a un core point



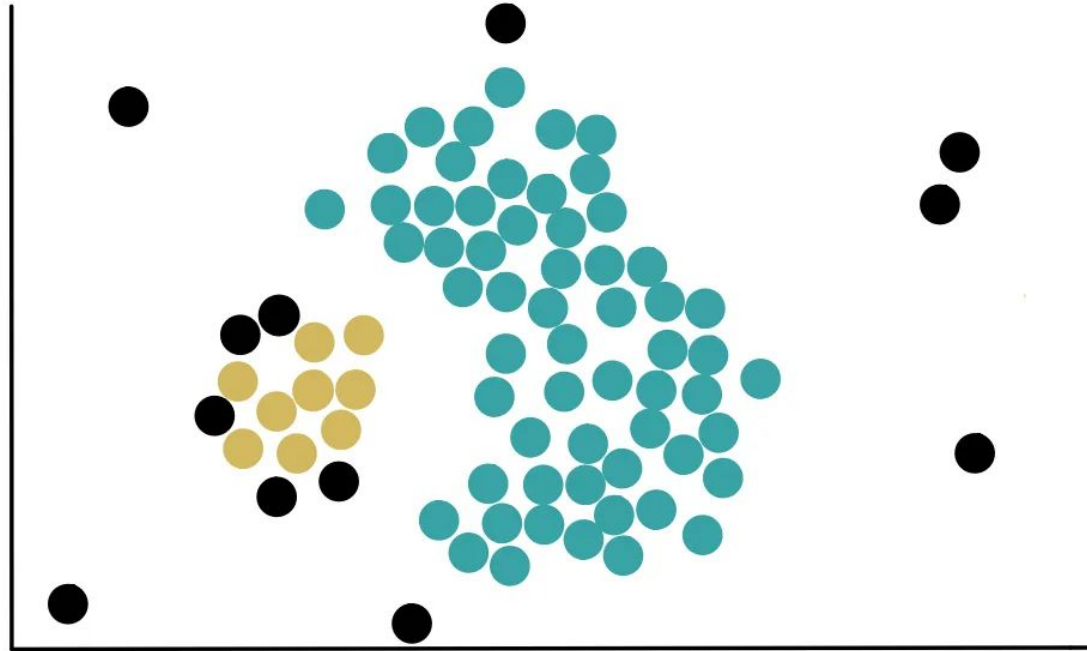
Step 5: Si aggiungono al cluster i **non core points** vicini a un core point



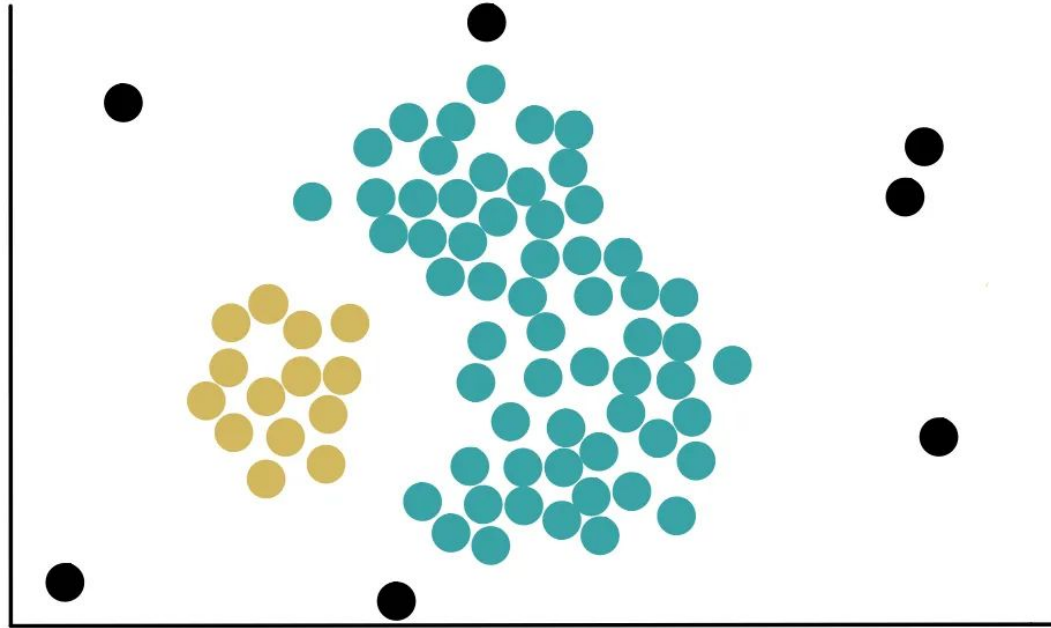
Step 6: Prendo un core point in modo casuale e lo assegno al secondo cluster



Step 7: Assegno al cluster i core points vicini al secondo cluster



Step 8: Assegno al cluster i non core points vicini a un core point del secondo cluster



Risorse:

- Statistical Learning di Trevor Hastie
- [k-Means Clustering: Explain It to Me Like I'm 10](#)
- [DBSCAN Clustering: Break it Down for Me](#)
- [Hierarchical Clustering: Explain It to Me Like I'm 10](#)
- [Distanze per Clustering](#)