Foundations of Machine Learning
CentraleSupélec — Fall 2017

12. Clustering

Chloé-Agathe Azencott
Centre for Computational Biology, Mines ParisTech
chloe-agathe.azencott@mines-paristech.fr

Inserm
Institut national
de la santé et de la recherche médicale

institutCurie

PSL
RESEARCH UNIVERSITY PARIS

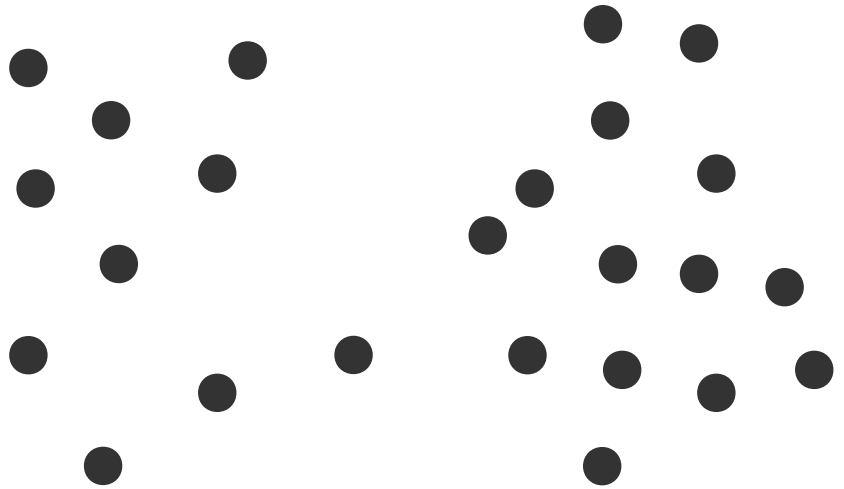
MINES
ParisTech

Learning objectives

- Explain what **clustering algorithms** can be used for.
- Explain and implement three different ways to **evaluate clustering algorithms.**
- Implement **hierarchical clustering,** discuss its various flavors.
- Implement **k-means clustering,** discuss its advantages and drawbacks.
- Sketch out a **density-based clustering** algorithm.

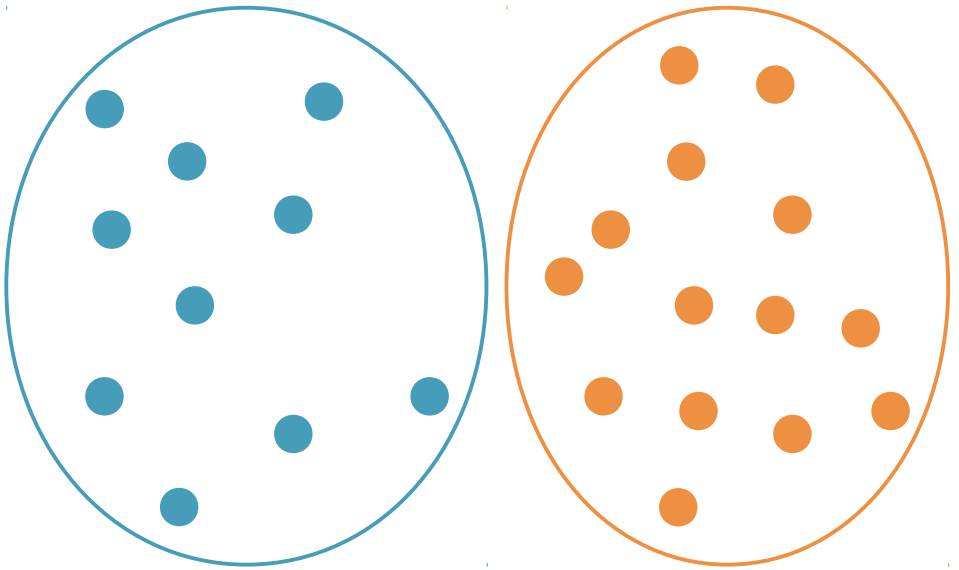
Goals of clustering

Group objects that are similar into **clusters:** classes that are unknown beforehand.



Goals of clustering

Group objects that are similar into **clusters:** classes that are unknown beforehand.



Goals of clustering

Group objects that are similar into **clusters:** classes that are unknown beforehand.

E.g.

- group genes that are similarly affected by a disease
- group patients whose genes respond similarly to a disease
- group pixels in an image that belong to the same object (image segmentation).

Applications of clustering

- **Understand** general characteristics of the data
- **Visualize** the data
- **Infer** some properties of a data point based on how it relates to other data points

  E.g.
  - find subtypes of diseases
  - visualize protein families
  - find categories among images
  - find patterns in financial transactions
  - detect communities in social networks

Distances and similarities

Distances & similarities

- Assess how **close / far**
  - data points are from each other
  - a data point is from a cluster
  - two clusters are from each other
- **Distance metric**

Distances & similarities

- Assess how **close / far**
  - data points are from each other
  - a data point is from a cluster
  - two clusters are from each other
- **Distance metric**

$d : \mathcal{X} \to \mathbb{R}$

$d(x, x) = 0$

$d(x^1, x^2) = d(x^2, x^1)$ **symmetry**

$d(x^1, x^2) \leq d(x^1, x^3) + d(x^3, x^2)$ **triangle inequality**

- **E.g. Lq distances**

$$d(x^1, x^2) = ||x^1 - x^2||_q = \left(\sum_{j=1}^{p} |x_j^1 - x_j^2|^q \right)^{1/q}$$

Distance & similarities

- **How do we get similarities?**

Distance & similarities

- Transform **distances into similarities?**

$$\mathrm{sim}(x, x') = \frac{1}{1 + d(x, x')} \qquad \mathrm{sim}(x, x') = \exp(-\gamma d(x, x')^2)$$

- **Kernels** define similarities

  For a given mapping

$$\Phi : \mathcal{X} \mapsto \mathcal{H}$$

  from the space of objects X to some Hilbert space H, the **kernel** between two objects x and x' is the inner product of their images in the feature spaces.

$$\forall x, x' \in \mathcal{X}, K(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{H}}$$

Pearson's correlation

- Measure of the **linear correlation** between two variables

$$\rho(x, z) = \frac{\sum_{j=1}^{p}(x_j - \bar{x})(z_j - \bar{z})}{\sqrt{\sum_{j=1}^{p}(x_j - \bar{x})^2}\sqrt{\sum_{j=1}^{p}(z_j - \bar{z})^2}}$$

- If the features are centered: 

$$\bar{x} = \frac{1}{p}\sum_{j=1}^{p} x_j$$

Pearson's correlation

- Measure of the **linear correlation** between two variables

$$\rho(x,z) = \frac{\sum_{j=1}^{p}(x_j - \bar{x})(z_j - \bar{z})}{\sqrt{\sum_{j=1}^{p}(x_j - \bar{x})^2}\sqrt{\sum_{j=1}^{p}(z_j - \bar{z})^2}}$$

$$\bar{x} = \frac{1}{p}\sum_{j=1}^{p} x_j$$

- If the features are centered:

$$\rho(x,z) = \frac{\sum_{j=1}^{p} x_j z_j}{\sqrt{\sum_{j=1}^{p} {x_j}^2}\sqrt{\sum_{j=1}^{p} {z_j}^2}} = \frac{\langle x, z \rangle}{||x||.||z||}$$

- **Normalized dot product = cosine**

Pearson vs Euclide

- **Pearson's coefficient**

$$\rho(x,z) = \frac{\sum_{j=1}^{p}(x_j - \bar{x})(z_j - \bar{z})}{\sqrt{\sum_{j=1}^{p}(x_j - \bar{x})^2}\sqrt{\sum_{j=1}^{p}(z_j - \bar{z})^2}}$$

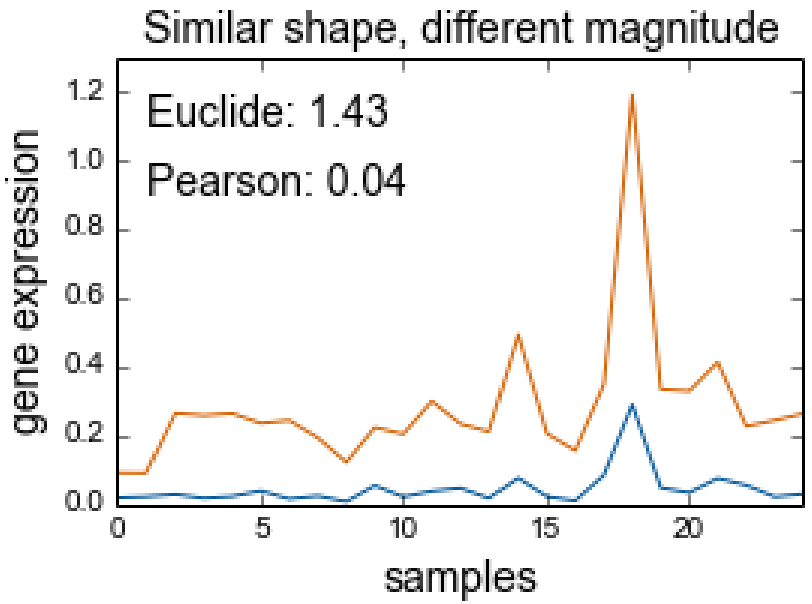
  Profiles of similar **shapes** will be close to each other, even if they differ in magnitude.

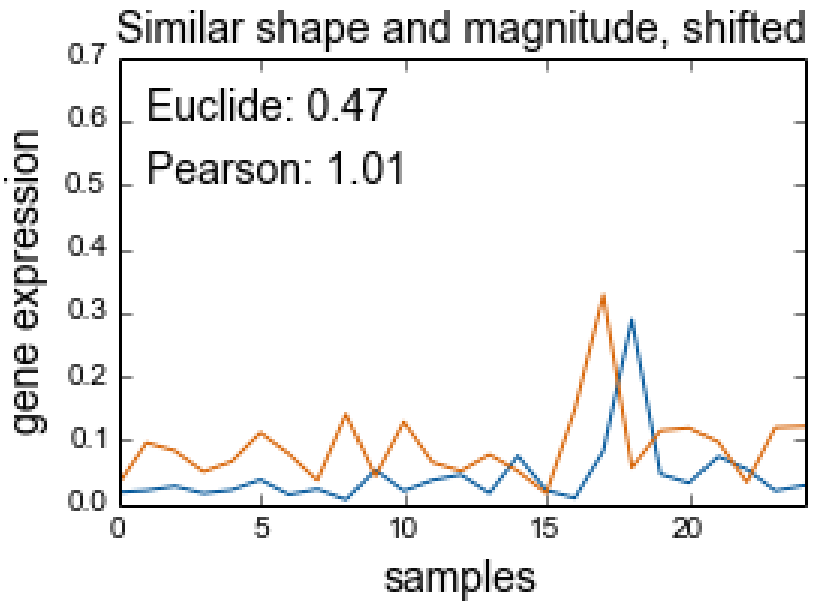
- **Euclidean distance**

$$d(x,z) = \sqrt{\sum_{j=1}^{p}(x_j - z_j)^2}$$

  Magnitude is taken into account.

Pearson vs Euclide





Evaluating clusters

Evaluating clusters

- Clustering is **unsupervised.**
- There is no ground truth. How do we evaluate the quality of a clustering algorithm?

Evaluating clusters

- Clustering is **unsupervised.**
- There is no ground truth. How do we evaluate the quality of a clustering algorithm?
- 1) Based on the **shape** of the clusters:

  Points within the same cluster should be nearby/similar and points far from each other should belong to different clusters.

Evaluating clusters

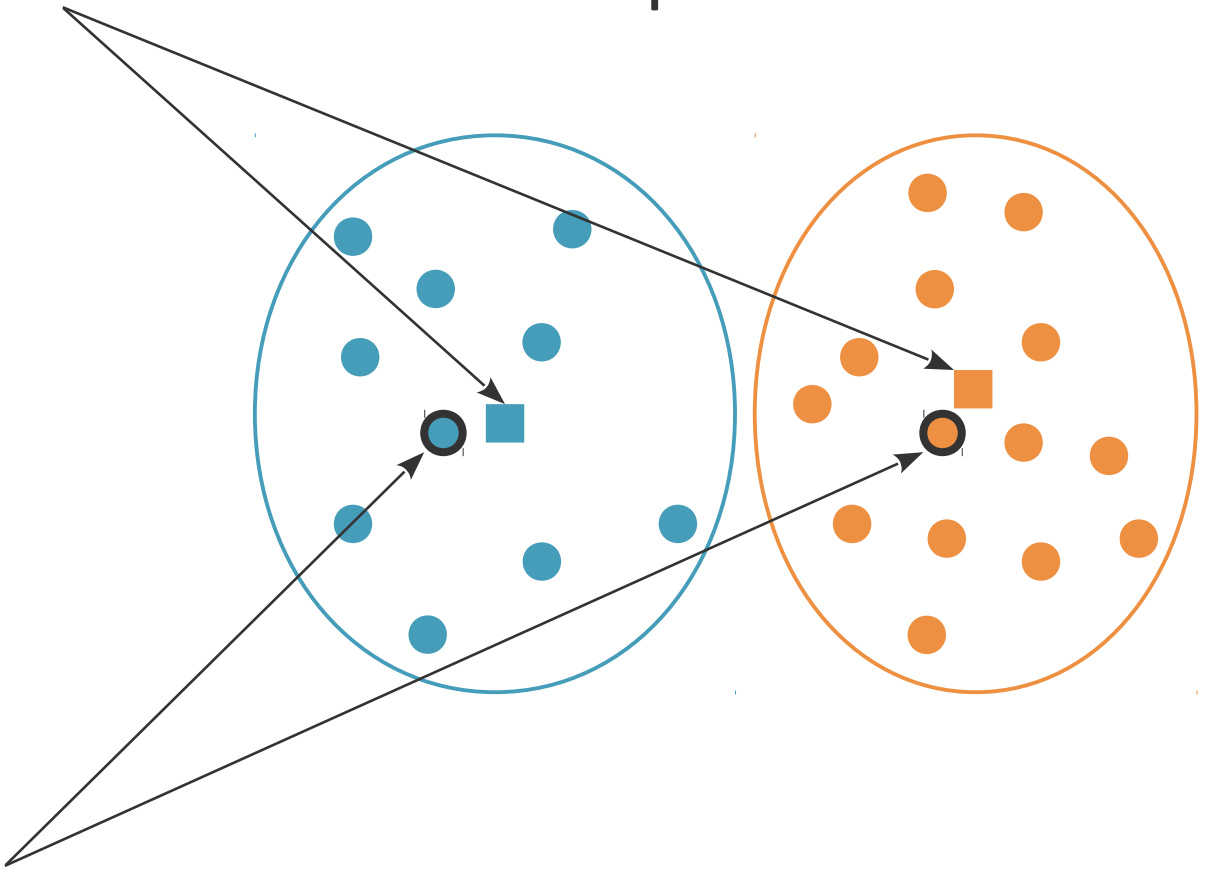
- Clustering is **unsupervised.**
- There is no ground truth. How do we evaluate the quality of a clustering algorithm?
- 1) Based on the **shape** of the clusters:

  Points within the same cluster should be nearby/similar and points far from each other should belong to different clusters.

Centroids and medoids

- **Centroid:** mean of the points in the cluster.

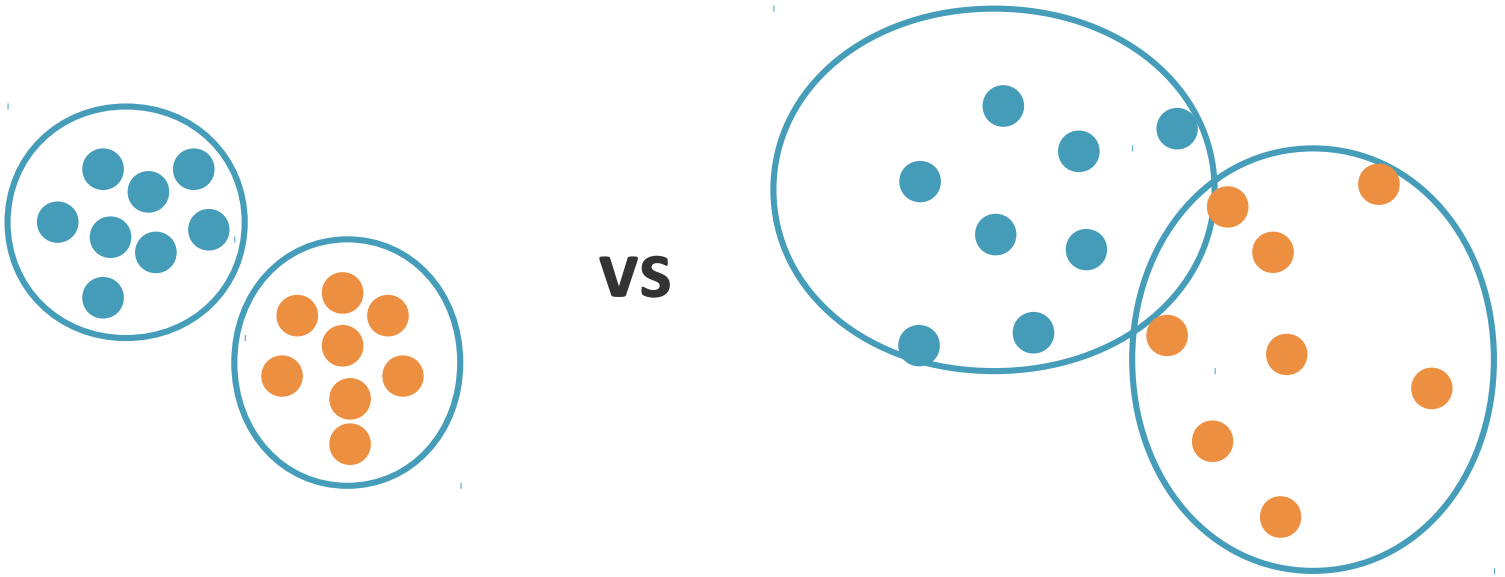
$$\boldsymbol{\mu} = \frac{1}{|C|} \sum_{\boldsymbol{x} \in C} \boldsymbol{x}$$



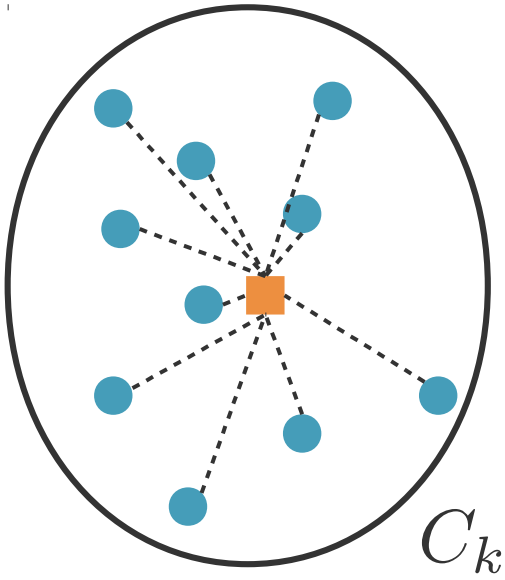
- **Medoid:** point in the cluster that is closest to the centroid.

$$\boldsymbol{m} = \arg\min_{\boldsymbol{x} \in C} d(\boldsymbol{x}, \boldsymbol{\mu})$$

Cluster shape: Tightness



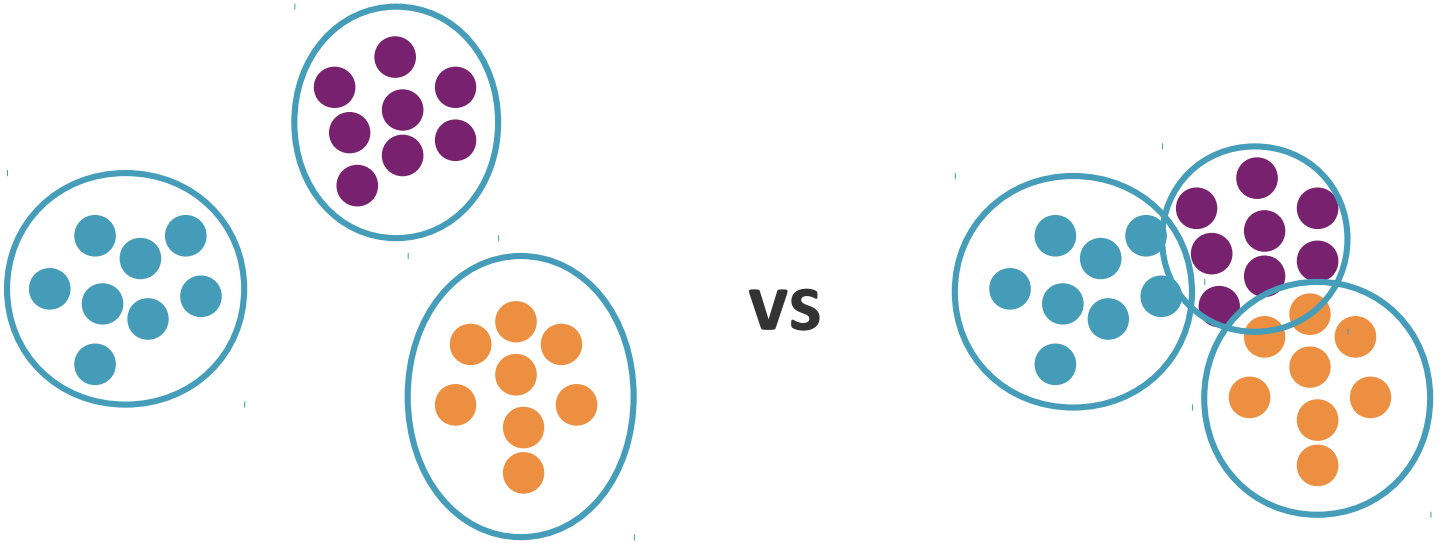
Cluster shape: Tightness



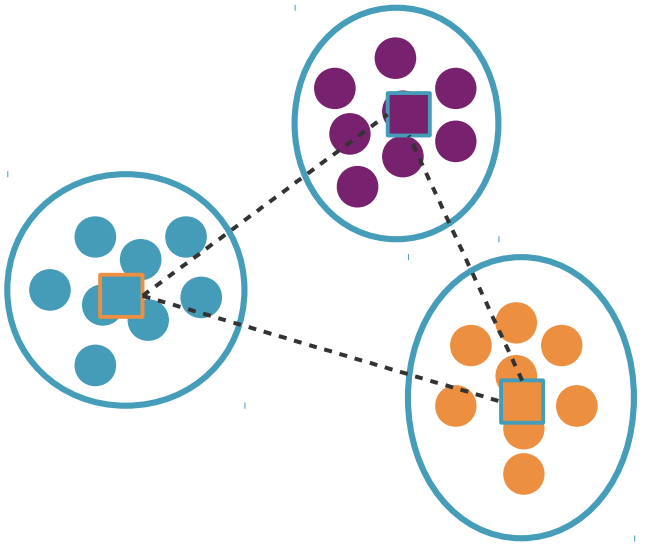
$$\sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{\boldsymbol{x} \in C_k} d(\boldsymbol{x}, \boldsymbol{\mu}_k)$$

T_k

Cluster shape: Separability



Cluster shape: Separability



$$\sum_{k=1}^{K} \sum_{l=k+1}^{K} \boxed{d(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l)} \leftarrow S_{kl}$$

Clusters shape: Davies-Bouldin

- Cluster **tightness** (**homogeneity**)

$$\sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{\boldsymbol{x} \in C_k} d(\boldsymbol{x}, \boldsymbol{\mu}_k)$$

T_k

- Cluster **separation**

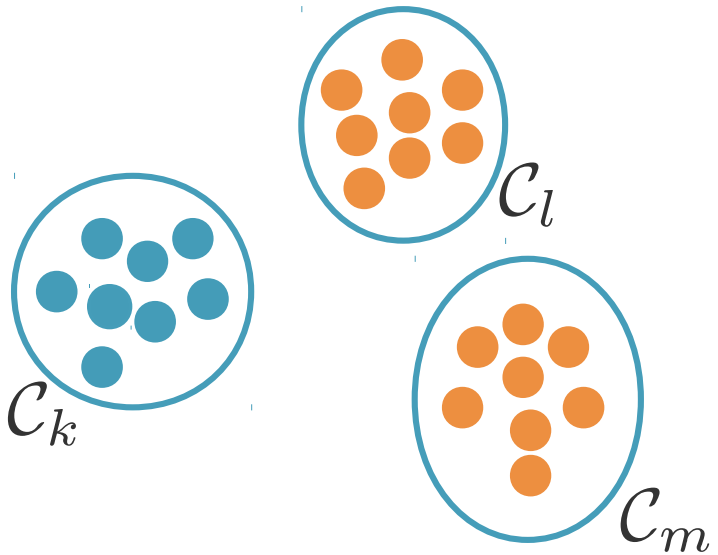
$$\sum_{k=1}^{K} \sum_{l=k+1}^{K} d(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l)$$

S_{kl}

- **Davies-Bouldin index**

$$D_k = \max_{l:l \neq k} \frac{T_k + T_l}{S_{kl}} \qquad DB = \frac{1}{K} \sum_{k=1}^{K} D_k$$

Clusters shape: Silhouette coefficient



- how well **x** fits in its cluster:

$$a(\boldsymbol{x}) = \frac{1}{n_k - 1} \sum_{\boldsymbol{u} \in \mathcal{C}_k, \boldsymbol{u} \neq \boldsymbol{x}} d(\boldsymbol{x}, \boldsymbol{u})$$

- how well **x** would fit in another cluster:

$$b(\boldsymbol{x}) = \min_{l \neq k} \frac{1}{n_l - 1} \sum_{\boldsymbol{u} \in \mathcal{C}_l} d(\boldsymbol{x}, \boldsymbol{u})$$

$$s(\boldsymbol{x}) = \frac{b(\boldsymbol{x}) - a(\boldsymbol{x})}{\max(a(\boldsymbol{x}), b(\boldsymbol{x}))}$$

$$s = \sum_{\boldsymbol{x}} s(\boldsymbol{x})$$

- if **x** is very close to the other points of its cluster: s(**x**) = 1
- if **x** is very close to the points in another cluster: s(**x**) = -1

Evaluating clusters

- Clustering is **unsupervised.**
- There is no ground truth. How do we evaluate the quality of a clustering algorithm?
- 1) Based on the **shape** of the clusters:

  Points within the same cluster should be nearby/similar and points far from each other should belong to different clusters.

- 2) Based on the **stability** of the clusters:

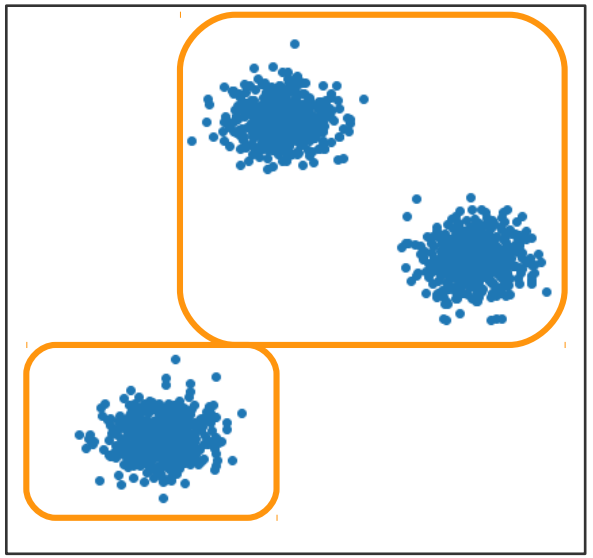
  We should get the same results if we remove some data points, add noise, etc.

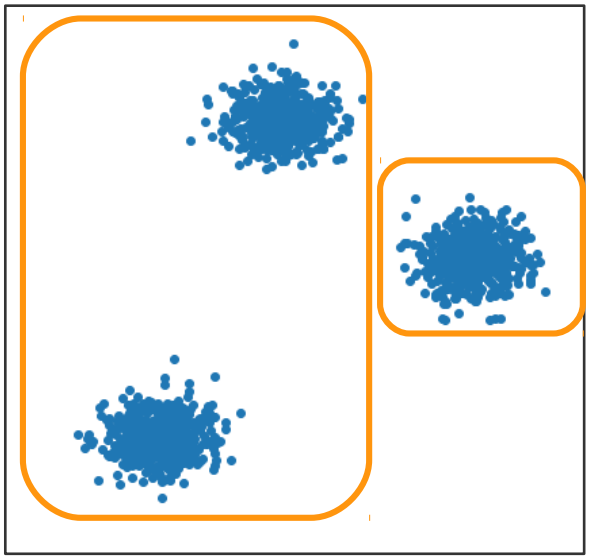
Cluster stability

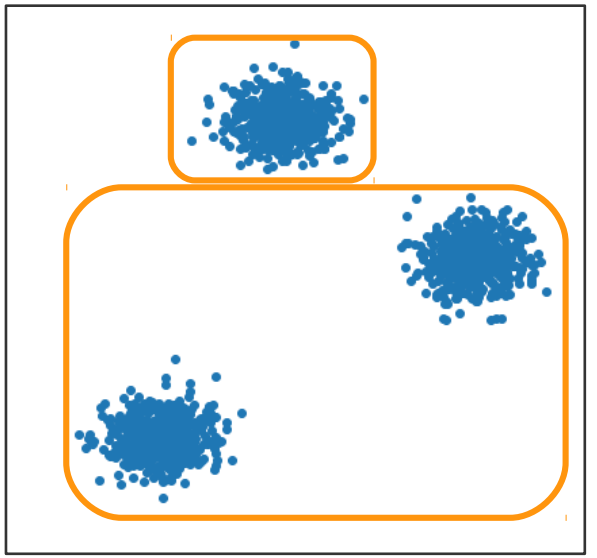
- **How many clusters?**

Cluster stability

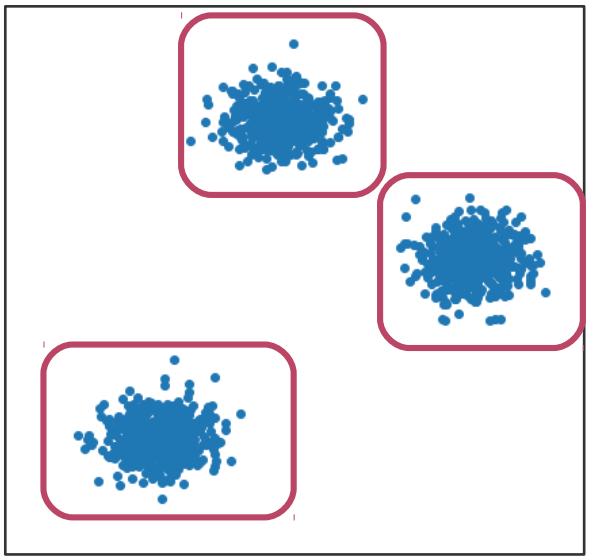
- K=2





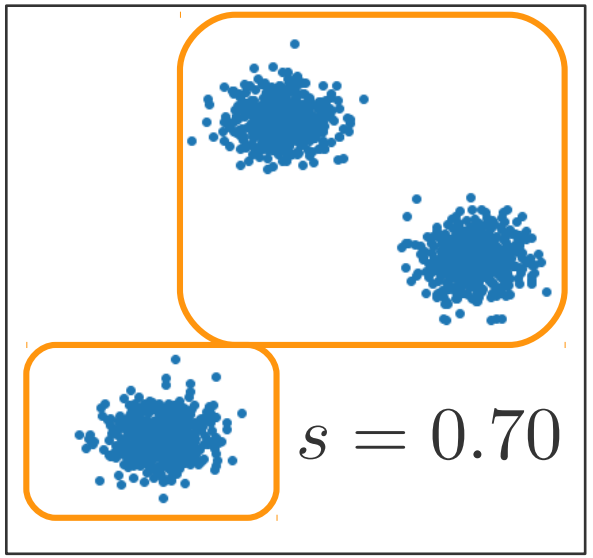


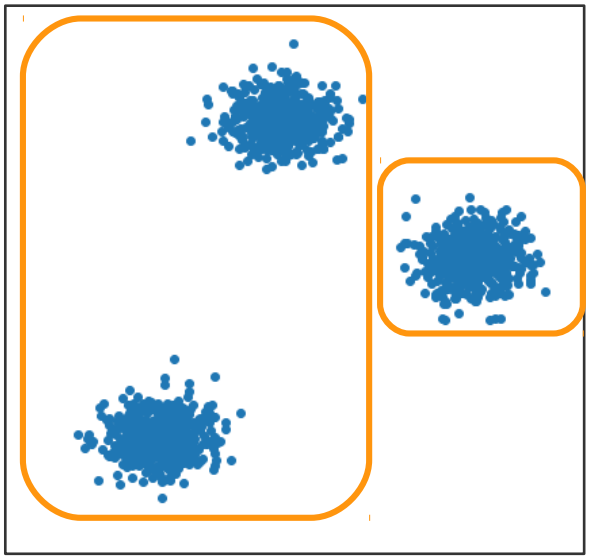
- K=3

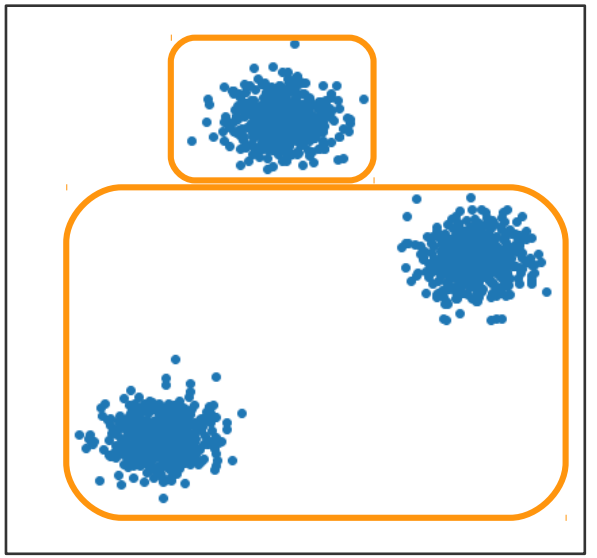


Cluster stability

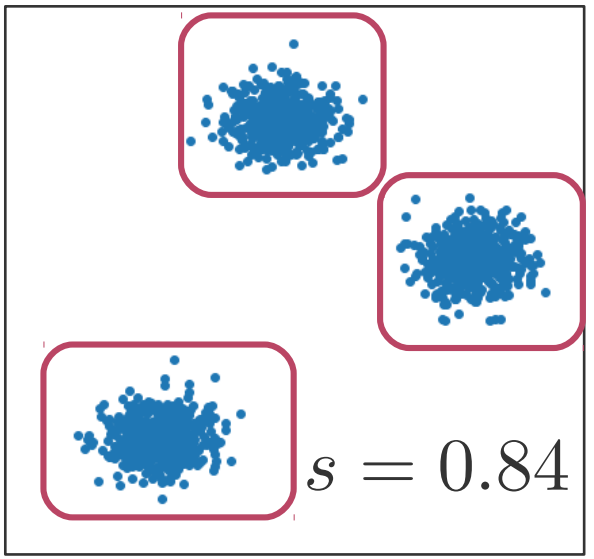
- K=2







- K=3



Evaluating clusters

- Clustering is **unsupervised.**
- There is no ground truth. How do we evaluate the quality of a clustering algorithm?
- 1) Based on the **shape** of the clusters:

  Points within the same cluster should be nearby/similar and points far from each other should belong to different clusters.
- 2) Based on the **stability** of the clusters:

  We should get the same results if we remove some data points, add noise, etc.
- 3) Based on **domain knowledge:**

  The clusters should “make sense”.

Domain knowledge

- **Do the cluster match natural categories?**
  - Check with human expertise

Ontology enrichment analysis

- **Ontology:**

  Entities may be grouped, related within a hierarchy, and subdivided according to similarities and differences.

  Build by human experts

- E.g.: **The Gene Ontology**

  `http://geneontology.org/`

  – Describe genes with a common vocabulary, organized in categories

    E.g. cellular process > cell death > programmed cell death > apoptotic process > execution phase of apoptosis

Ontology enrichment analysis

- **Enrichment analysis:**

  Are there more data points from ontology category G in cluster C than expected by chance?

- **TANGO** [Tanay et al., 2003]
  - Assume data points sampled from a hypergeometric distribution
  - The probability for the intersection of G and C to contain more than t points is:

$$Pr(|G \cap C| \geq t) = 1 - \sum_{i=1}^{t} \frac{\binom{|G|}{i}\binom{n-|G|}{|C|-i}}{\binom{n}{|C|}}$$

Ontology enrichment analysis

- **Enrichment analysis:**

  Are there more data points from ontology category G in cluster C than expected by chance?

- **TANGO** [Tanay et al., 2003]
  - Assume data points sampled from a hypergeometric distribution
  - The probability for the intersection of G and C to contain more than t points is:

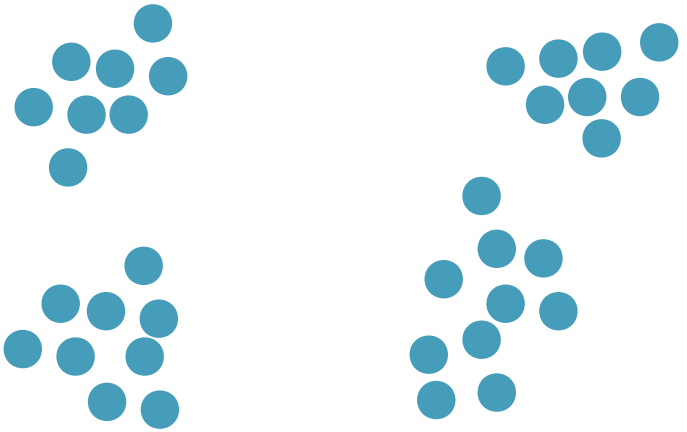
$$Pr(|G \cap C| \geq t) = 1 - \sum_{i=1}^{t} \frac{\binom{|G|}{i}\binom{n-|G|}{|C|-i}}{\binom{n}{|C|}}$$

Probability of getting i points from G when drawing |C| points from a total of n samples.

Hierarchical clustering

Hierachical clustering

Group data over a variety of possible scales, in a multi-level hierarchy.



Construction

- **Agglomerative** approach (**bottom-up**)

  Start with each element in its own cluster

  Iteratively **join** neighboring clusters.

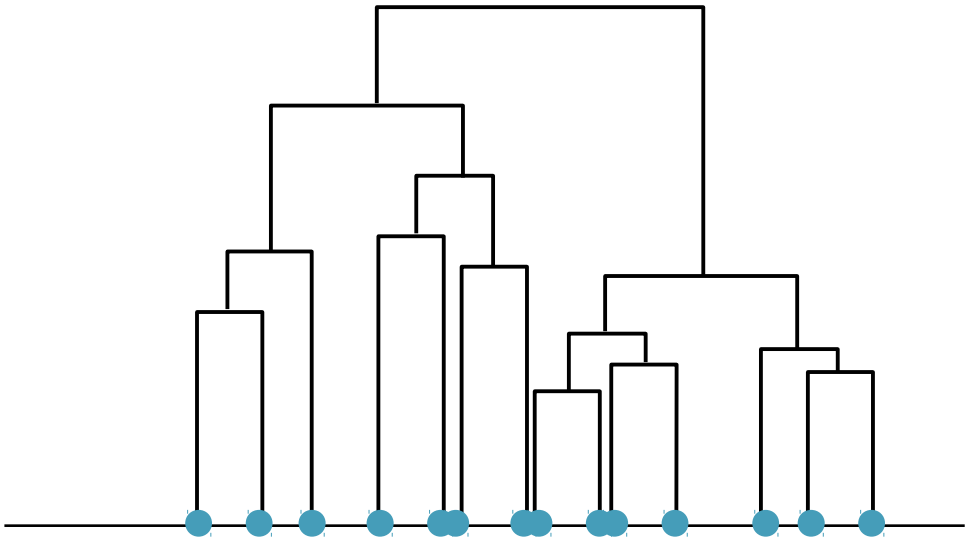
- **Divisive** approach (**top-down**)

  Start with all elements in the same cluster

  Iteratively **separate** into smaller clusters.

Dendogram

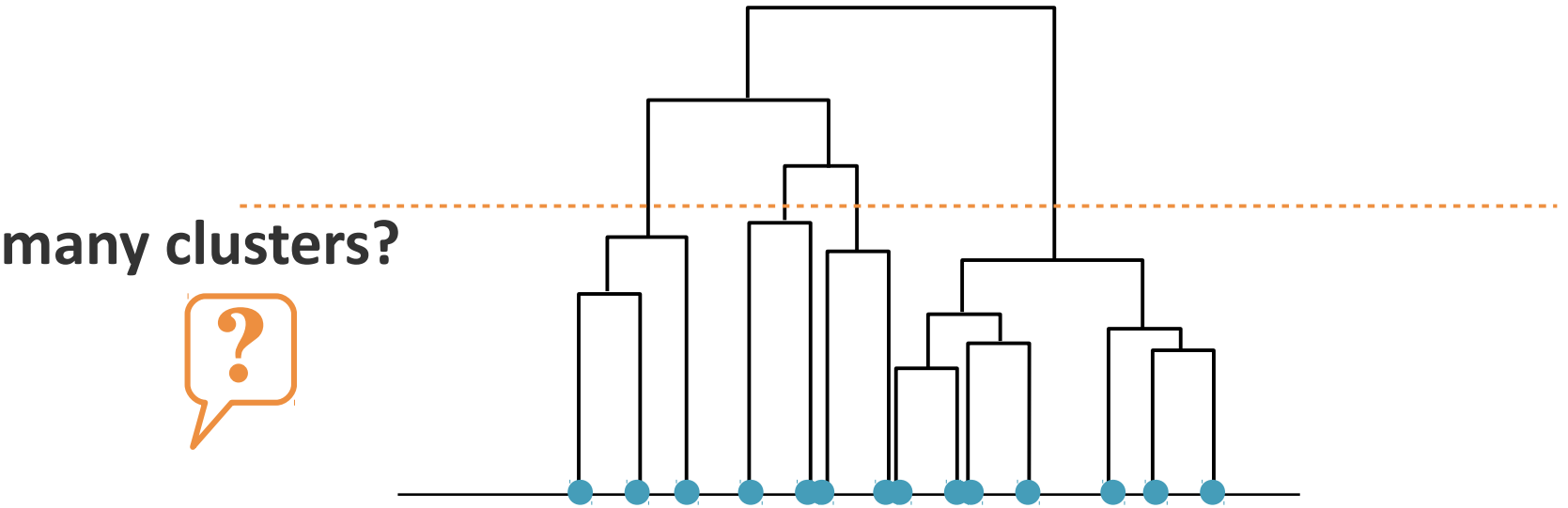
- The results of a hierarchical clustering algorithm are presented in a **dendogram.**
- Branch length = cluster distance.



Dendogram

- The results of a hierarchical clustering algorithm are presented in a **dendogram.**
- U height = distance.

How many clusters?

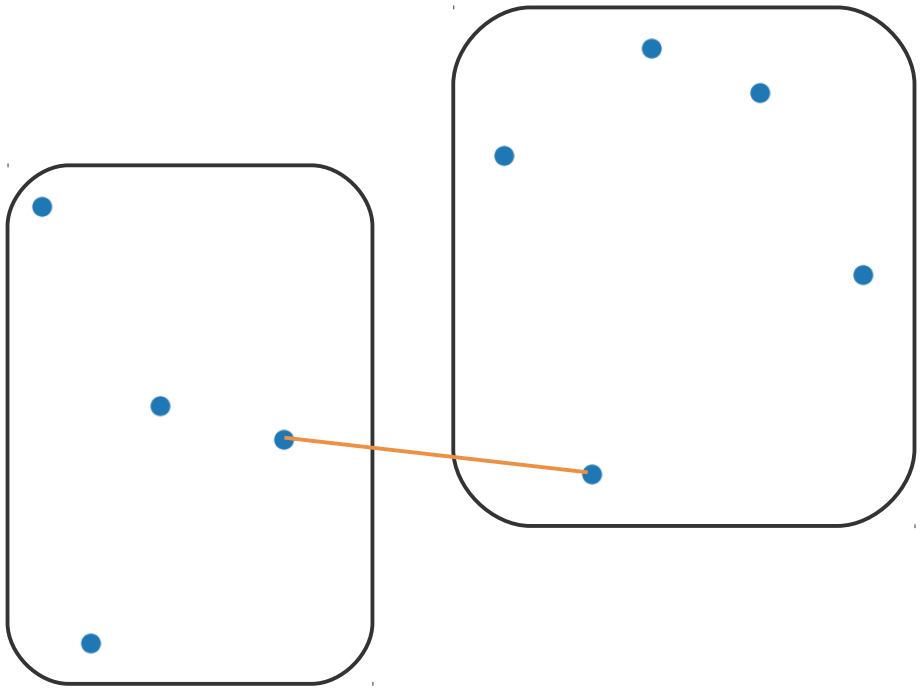


Dendogram

- The results of a hierarchical clustering algorithm are presented in a **dendogram.**
- U height = distance.

Linkage: connecting two clusters

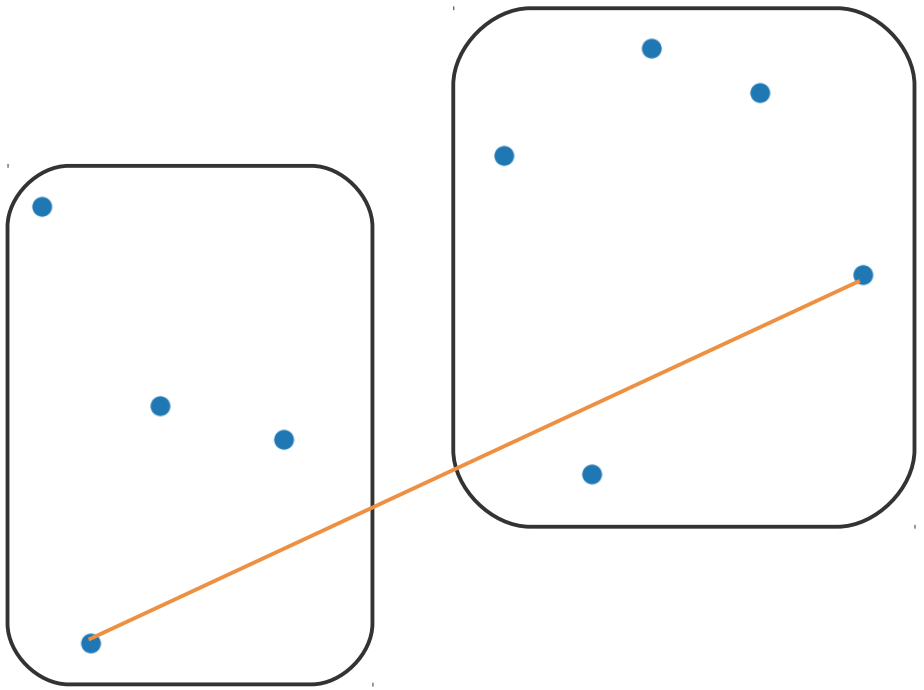
- **Single linkage**



$$d(\mathcal{C}_1, \mathcal{C}_2) = \min_{\boldsymbol{x} \in \mathcal{C}_1, \boldsymbol{z} \in \mathcal{C}_2} d(\boldsymbol{x}, \boldsymbol{z})$$

Linkage: connecting two clusters

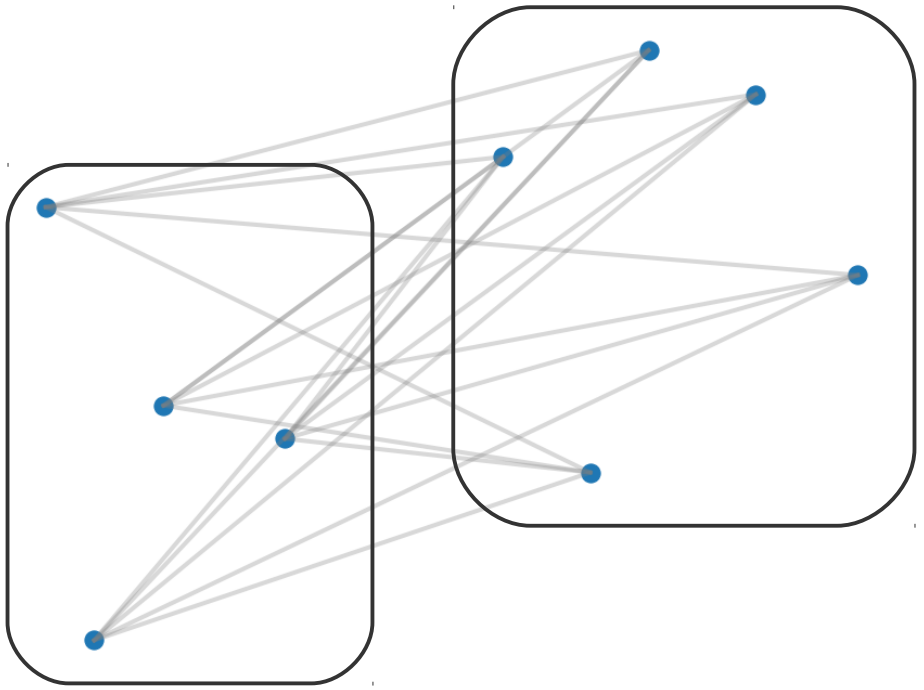
- **Complete linkage**



$$d(\mathcal{C}_1, \mathcal{C}_2) = \max_{\boldsymbol{x} \in \mathcal{C}_1, \boldsymbol{z} \in \mathcal{C}_2} d(\boldsymbol{x}, \boldsymbol{z})$$

Linkage: connecting two clusters

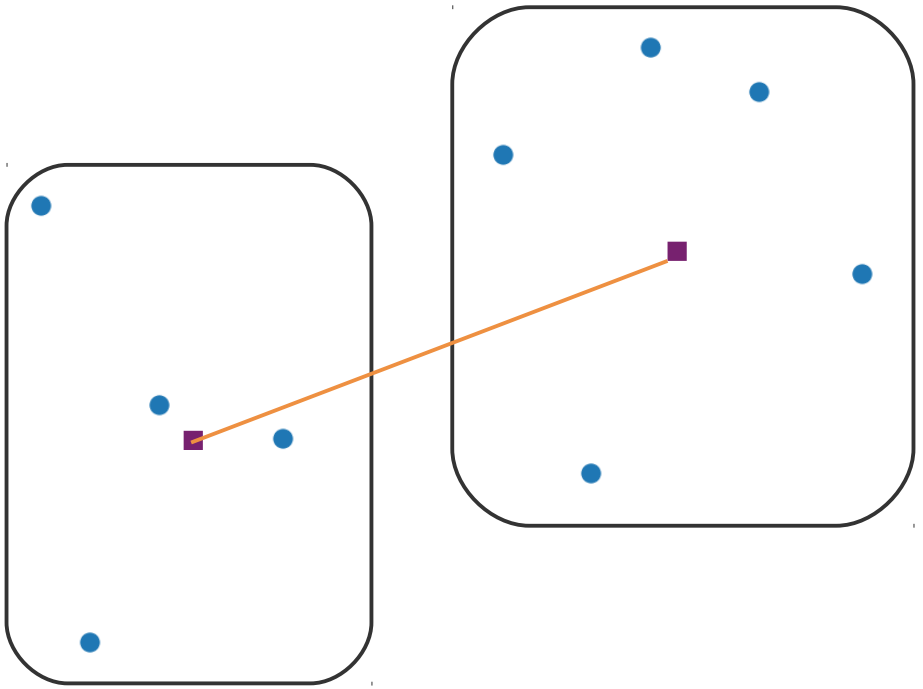
- **Average linkage**



$$d(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{|\mathcal{C}_1||\mathcal{C}_2|} \sum_{\boldsymbol{x} \in \mathcal{C}_1} \sum_{\boldsymbol{z} \in \mathcal{C}_2} d(\boldsymbol{x}, \boldsymbol{z})$$

Linkage: connecting two clusters

- **Centroid linkage**



$$d(\mathcal{C}_1, \mathcal{C}_2) = d\left(\sum_{\boldsymbol{x} \in \mathcal{C}_1} \frac{\boldsymbol{x}}{|C_1|}, \sum_{\in \mathcal{C}_2} \frac{\boldsymbol{z}}{|\mathcal{C}_2|}\right)$$

Linkage: connecting two clusters

- **Ward**

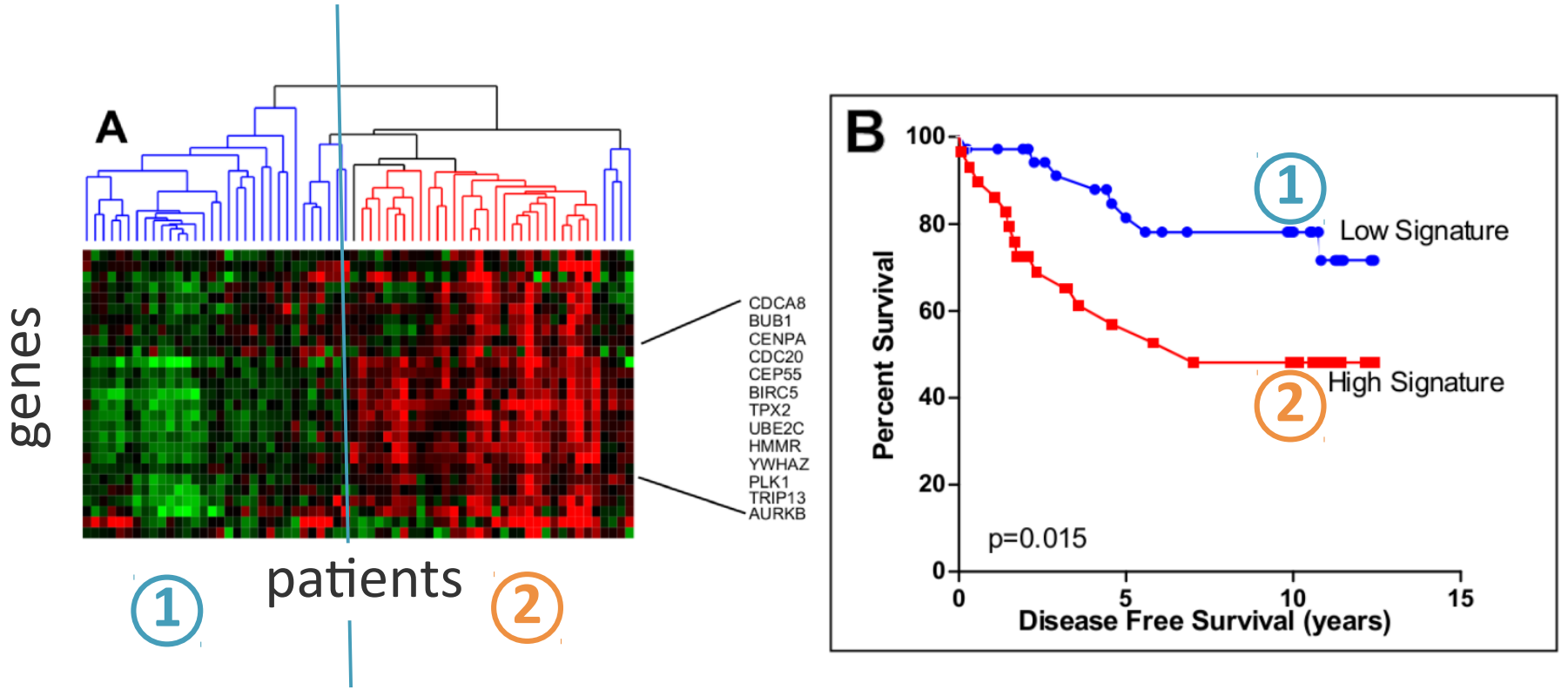
  Join clusters so as to minimize within-cluster variance

$$\mathrm{Var}_{\mathrm{in}}(C) = \frac{1}{|C|} \sum_{\boldsymbol{x} \in C} ||\boldsymbol{x} - \boldsymbol{\mu}_C||^2$$

Example: Gene expression clustering

Breast cancer survival signature

[Bergamashi et al. 2011]



Hierarchical clustering

- **Advantages**
  - No need to pre-define the number of clusters
  - Interpretability
- **Drawbacks**
  - Computational complexity 

Hierarchical clustering

- **Advantages**
  - No need to pre-define the number of clusters
  - Interpretability
- **Drawbacks**
  - Computational complexity

    E.g. Single/complete linkage (naive):

    At least $O(pn^2)$ to compute all pairwise distances.
  - Must decide at which level of the hierarchy to split
  - Lack of robustness (unstable)

K-means

K-means clustering

- Minimize the **intra-cluster variance**

$$\mathrm{Var}_{\mathrm{in}}(C) = \frac{1}{|C|} \sum_{\boldsymbol{x} \in C} ||\boldsymbol{x} - \boldsymbol{\mu}_C||^2$$

$$V = \sum_{k=1}^{K} \sum_{\boldsymbol{x} \in C_k} \frac{1}{|C_k|} ||\boldsymbol{x} - \boldsymbol{\mu}_{C_k}||^2$$

- **What will this partition of the space look like?**

K-means clustering

- Minimize the **intra-cluster variance**

$$\mathrm{Var}_{\mathrm{in}}(C) = \frac{1}{|C|} \sum_{\boldsymbol{x} \in C} ||\boldsymbol{x} - \boldsymbol{\mu}_C||^2$$

$$V = \sum_{k=1}^{K} \sum_{\boldsymbol{x} \in C_k} \frac{1}{|C_k|} ||\boldsymbol{x} - \boldsymbol{\mu}_{C_k}||^2$$

- For each cluster, the points in that cluster are those that are closest to its centroid than to any other centroid

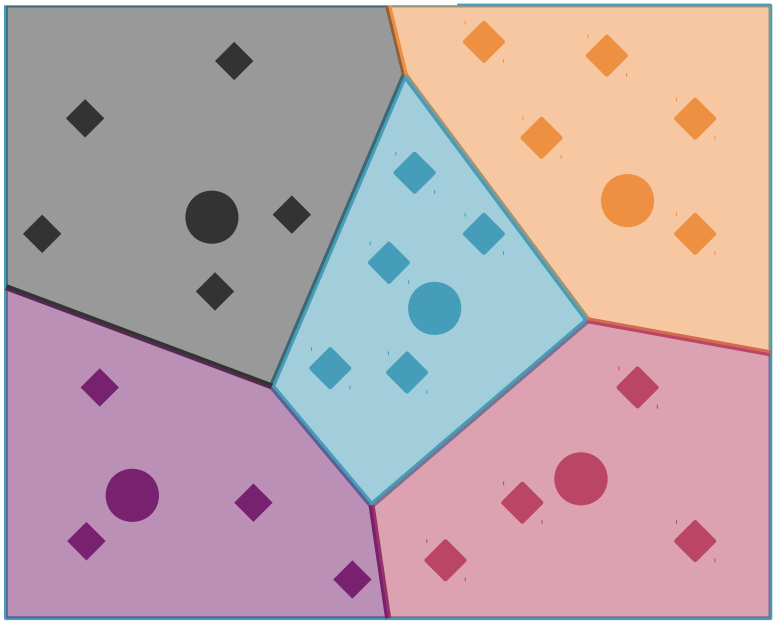
K-means clustering

- Minimize the **intra-cluster variance**

$$\mathrm{Var}_{\mathrm{in}}(C) = \frac{1}{|C|} \sum_{\boldsymbol{x} \in C} ||\boldsymbol{x} - \boldsymbol{\mu}_C||^2$$

$$V = \sum_{k=1}^{K} \sum_{\boldsymbol{x} \in C_k} \frac{1}{|C_k|} ||\boldsymbol{x} - \boldsymbol{\mu}_{C_k}||^2$$

- **Voronoi tesselation**



Lloyd's algorithm

- K-means cannot be easily optimized
- We adopt a **greedy strategy.**
  - Partition the data into K clusters at random
  - Compute the centroid of each cluster
  - Assign each point to the cluster whose centroid it is closest to
  - Repeat until cluster membership converges.

K-means

- **Advantages**
  - **What is the computational time of k-means?**

K-means

- **Advantages**
  - **What is the computational time of k-means?**

compute kn distances in p dimensions

number of iterations

Can be small if there's indeed a cluster structure in the data

K-means

- **Advantages**
  - Computational time is linear

$$\mathcal{O}(npkt)$$

  - Easily implementable
- **Drawbacks**
  - Need to set up K ahead of time
  - **What happens when there are outliers?**

K-means

- **Advantages**
  - Computational time is linear

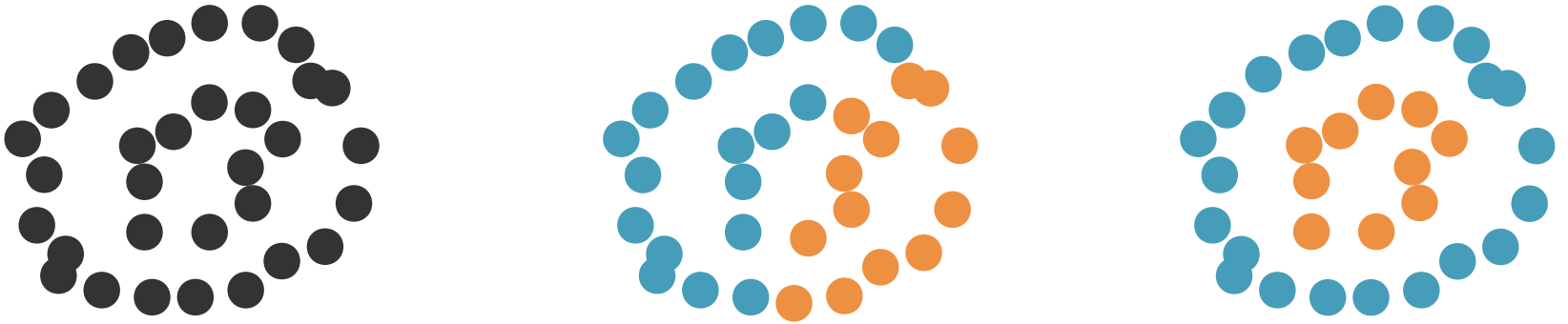
$$\mathcal{O}(npkt)$$

  - Easily implementable
- **Drawbacks**
  - Need to set up K ahead of time
  - Sensitive to noise and outliers
  - Stochastic (different solutions with each iteration)
  - The clusters are forced to have convex shapes

K-means variants

- **K-means++**
  - Seeding algorithm to initialize clusters with centroids "spread-out" throughout the data.
  - Deterministic
- **K-medoids**
- **Kernel k-means**

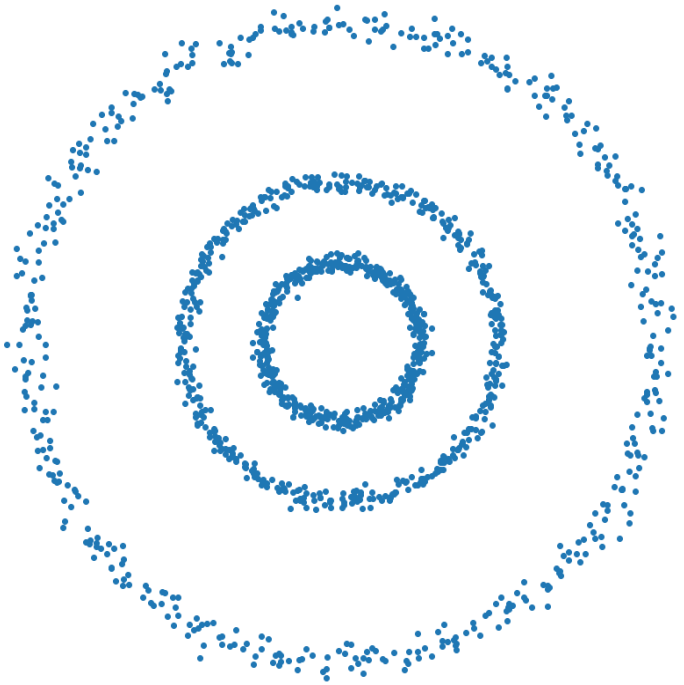
  Find clusters in feature space



k-means kernel k-means

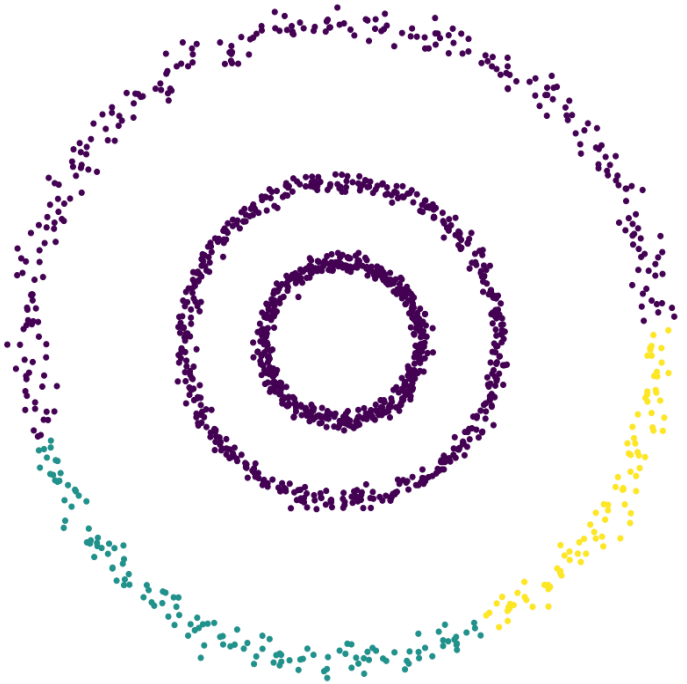
Density-based clustering

Density-based clustering



Hierarchical clustering:

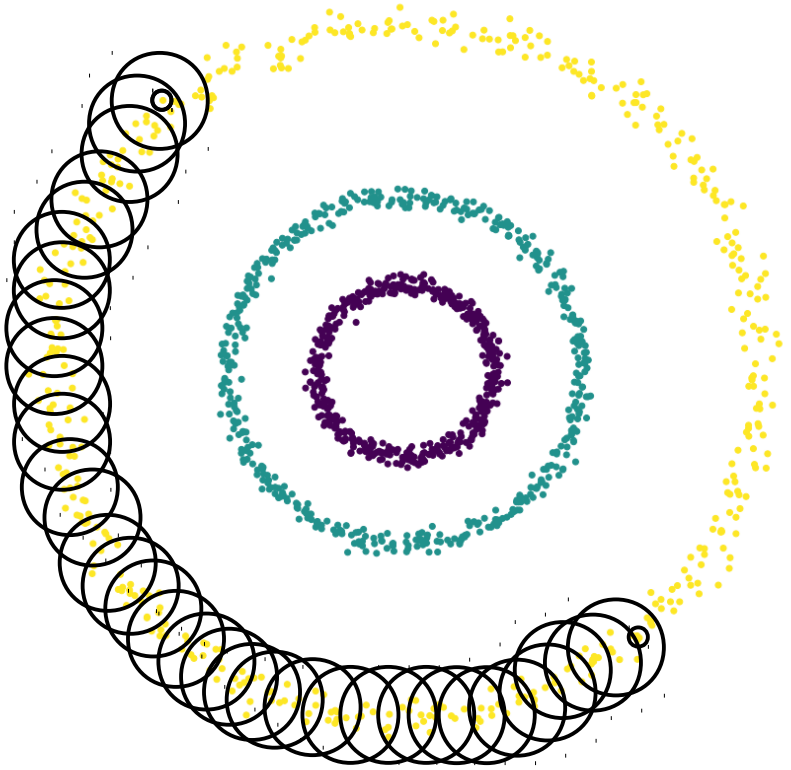
```
cluster.AgglomerativeClustering(linkage='average',
                                n_clusters=3)
```



k-means clustering

```
cluster.KMeans(n_clusters=3)
```

DBSCAN



- Density-based clustering: clusters are made of dense neighborhoods of points

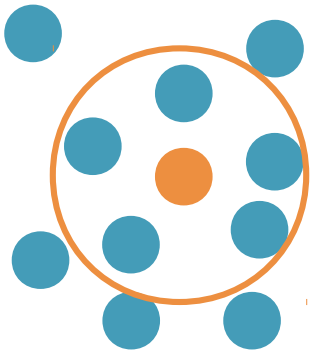
DBSCAN

- **ε-neighborhood:**

$$\mathcal{N}_\epsilon(\boldsymbol{x}) = \{\boldsymbol{z} | d(\boldsymbol{x}, \boldsymbol{z}) < \epsilon\}$$

- **core points:**

$$\boldsymbol{x} : |\mathcal{N}_\epsilon(\boldsymbol{x})| \geq n_{\min}$$



- **x** and **z** are **density-connected:**

  $\exists\ \boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^m$ core points such that

  - $\boldsymbol{x}^1 \in \mathcal{N}_\epsilon(\boldsymbol{x})$
  - $\boldsymbol{x}^i \in \mathcal{N}_\epsilon(\boldsymbol{x}^{i-1})$
  - $\boldsymbol{z} \in \mathcal{N}_\epsilon(\boldsymbol{x}^m)$

Summary

- **Clustering:** unsupervised approach to group similar data points together.
- **Evaluate** clustering algorithms based on
  - the **shape** of the cluster
  - the **stability** of the results
  - the consistency with **domain knowledge.**
- **Hierarchical clustering**
  - top-down / bottom-up
  - various **linkage** functions.
- **k-means clustering** tries to minimize intra-cluster variance
- **density-based clustering** clusters dense neighborhoods together.

References

- *Introduction to Data Mining*

  P. Tang, M. Steinbach, V. Kumar

  Chap. 8: Cluster analysis

  `https://www-users.cs.umn.edu/~kumar001/dmbook/ch8.pdf`