Unsupervised Classification

A Connectivity-Based Algorithm: Agglomerative Hierarchical Clustering

Data Analysis – juliette.chevallier@insa-toulouse.fr

INSA Toulouse, Applied Mathematics, 4th year

Hierarchical Classification

Introduction

We observe n individuals described by p variables: $x_i = \left(x_{i1}, x_{i2}, \dots, x_{ip}\right) \in \mathcal{X}$

$$X = \begin{pmatrix} x_{11} & \boxed{x_{12}} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}$$

$$\mathcal{X} = \mathbb{R}^p\,,\ \{0,1\}^p\,,\ ]-\pi,\pi]^p\,,\ \mathbb{R}^q \times \{0,1\}^{p-q}\,, \dots$$

- Initial measurements,
- Transformed measurements,
- Coordinates after dimension reduction.

- Let d be an adapted dissimilarity between individuals,
  $\rightsquigarrow$ Depends mainly on whether the data are *quantitative* or *qualitative*.

Introduction

We observe n individuals described by p variables: $x_i = \big(x_{i1}, x_{i2}, \ldots, x_{ip}\big) \in \mathcal{X}$

$$X = \begin{pmatrix} x_{11} & \boxed{x_{12}} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}$$

$$\mathcal{X} = \mathbb{R}^p\,,\ \{0,1\}^p\,,\ ]-\pi,\pi]^p\,,\ \mathbb{R}^q \times \{0,1\}^{p-q}\,, \ldots$$

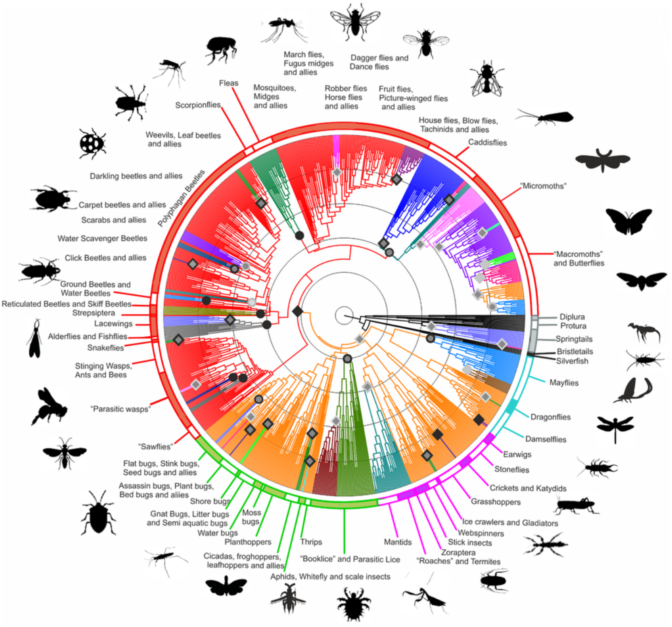
- Initial measurements,
- Transformed measurements,
- Coordinates after dimension reduction.

- Let d be an adapted dissimilarity between individuals,
  $\rightsquigarrow$ Depends mainly on whether the data are *quantitative* or *qualitative*.

- **Goal**: Prioritize the data, *i.e.* obtain a sequence of nested partitions.

  More precisely: Production of a structure (tree or dendrogram) allowing:
  - Identification of hierarchical links between individuals or groups of individuals,
  - Detection of a "natural" number of classes within the population.

Example of a Hierarchy: Phylogenetic Tree (Insects)



Hierarchy

Definition: Hierarchy of a set $\mathcal{X} = \{x_1, \dots, x_n\}$

A hierarchy $\mathcal{H}$ is a set of parts of $\mathcal{X}$ satisfying:

- $\forall i \in [\![1, n]\!]$, $\{x_i\} \in \mathcal{H}$,
- $\mathcal{X} \in \mathcal{H}$,
- $\forall A, B \in \mathcal{H}$, $A \cap B = \varnothing$ or $A \subset B$ or $B \subset A$.

Example:

$$\mathcal{H} = \Big\{\{A\}, \{B\}, \{C\}, \{D\}, \{E\}, \{A, B\}, \{C, D\}, \{C, D, E\}, \{A, B, C, D, E\}\Big\}$$

Hierarchy

Definition: Hierarchy of a set $\mathcal{X} = \{x_1, \ldots, x_n\}$

A hierarchy $\mathcal{H}$ is a set of parts of $\mathcal{X}$ satisfying:

- $\forall i \in [\![1, n]\!]$, $\{x_i\} \in \mathcal{H}$,
- $\mathcal{X} \in \mathcal{H}$,
- $\forall A, B \in \mathcal{H}$, $A \cap B = \varnothing$ or $A \subset B$ or $B \subset A$.

Example:

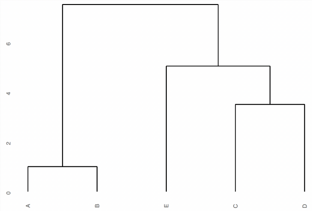
$$\mathcal{H} = \Big\{\{A\}, \{B\}, \{C\}, \{D\}, \{E\}, \{A, B\}, \{C, D\}, \{C, D, E\}, \{A, B, C, D, E\}\Big\}$$

Indexed hierarchy

An indexed hierarchy is a pair $(\mathcal{H}, h)$ where $\mathcal{H}$ is a hierarchy and $h \colon \mathcal{H} \to \mathbb{R}^+$ fulfills :

- $\forall A \in \mathcal{H}$, $h(A) = 0$ iff A is a *singleton*,
- $\forall A, B \in \mathcal{H}$ s.t $A \neq B$,
  If $A \subset B$ then $h(A) \leqslant h(B)$.

- $\forall x \in \{A, B, C, D, E\}, \ h(\{x\}) = 0$,
- $h(\{A, B\}) = 1$,
- $h(\{C, D\}) = 3.5$,
- $h(\{C, D, E\}) = 5.04$,
- $h(\{A, B, C, D, E\}) = 7.52$.



Hierarchy

Definition: Hierarchy of a set $\mathcal{X} = \{x_1, \dots, x_n\}$

A hierarchy $\mathcal{H}$ is a set of parts of $\mathcal{X}$ satisfying:

- $\forall i \in [\![1, n]\!]$, $\{x_i\} \in \mathcal{H}$,
- $\mathcal{X} \in \mathcal{H}$,
- $\forall A, B \in \mathcal{H}$, $A \cap B = \varnothing$ or $A \subset B$ or $B \subset A$.

Example:

$$\mathcal{H} = \Big\{\{A\}, \{B\}, \{C\}, \{D\}, \{E\}, \{A, B\}, \{C, D\}, \{C, D, E\}, \{A, B, C, D, E\}\Big\}$$

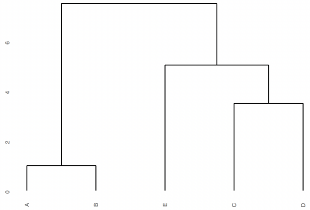
Indexed hierarchy

An indexed hierarchy is a pair $(\mathcal{H}, h)$ where $\mathcal{H}$ is a hierarchy and $h \colon \mathcal{H} \to \mathbb{R}^+$ fulfills :

- $\forall A \in \mathcal{H}$, $h(A) = 0$ iff A is a *singleton*,
- $\forall A, B \in \mathcal{H}$ s.t $A \neq B$,
  If $A \subset B$ then $h(A) \leqslant h(B)$.

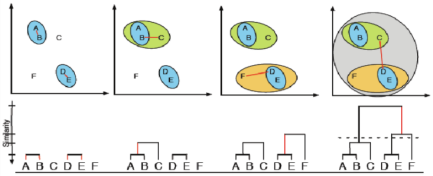
- $\forall x \in \{A, B, C, D, E\}$, $h(\{x\}) = 0$,
- $h(\{A, B\}) = 1$,
- $h(\{C, D\}) = 3.5$,
- $h(\{C, D, E\}) = 5.04$,
- $h(\{A, B, C, D, E\}) = 7.52$.

Dendrogram: Representation of the dendrogram not unique: if $\mathcal{X}$ is a set of n points, $2n - 1$ possibilities to order the leaves of the tree.



Hierarchical Classification

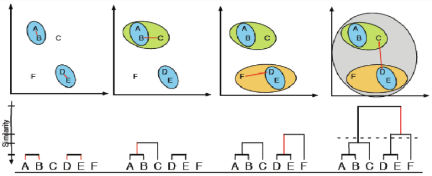
Hierarchy *vs.* Clustering



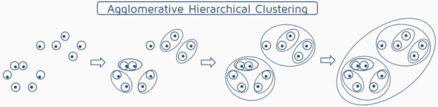
Janssen (2012)

- **Aim**: Build an indexed hierarchy

Hierarchy *vs.* Clustering



Janssen (2012)

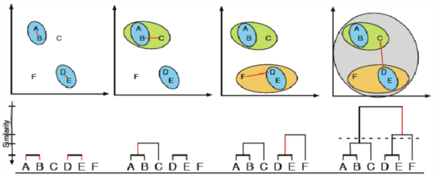


- **Aim**: Build an indexed hierarchy

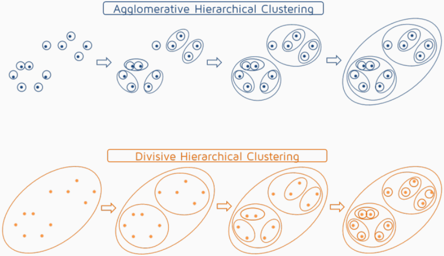
- **1st strategy**: *Agglomerative Hierarchical Classification (AHC)*
  - Sart from the bottom of the dendrogram (the singletons),
  - Aggregate two by two the closest parts until we obtain only one class.

    $\rightsquigarrow$ *How to choose the classes to aggregate?*

Hierarchy *vs.* Clustering



Janssen (2012)



- **Aim**: Build an indexed hierarchy

- **1st strategy**: *Agglomerative Hierarchical Classification (AHC)*
  - Sart from the bottom of the dendrogram (the singletons),
  - Aggregate two by two the closest parts until we obtain only one class.

  $\rightsquigarrow$ *How to choose the classes to aggregate?*

- **2nd strategy**: *Divise Hierarchical Classification (DHC)*
  - Start from the top of the dendrogram,
  - Successive divisions until we obtain classes reduced to singletons.

  $\rightsquigarrow$ *How to choose the classes to divide?*

Agglomerative Hierarchical Classification [Johnson, 1967, Ward, 1963]

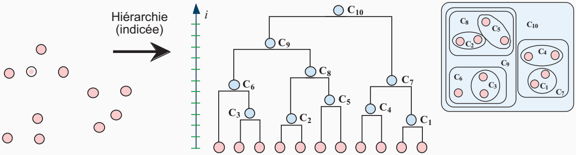
Initialization:

- Let an aggregation measure $\mathcal{D}$.
- Let $\mathcal{P}_n^{(0)} = \big\{\{x_1\}, \ldots, \{x_n\}\big\}$ be the singleton partition.

Iteration t: From the partition $\mathcal{P}_K^{(t)} = \{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$ into K classes,

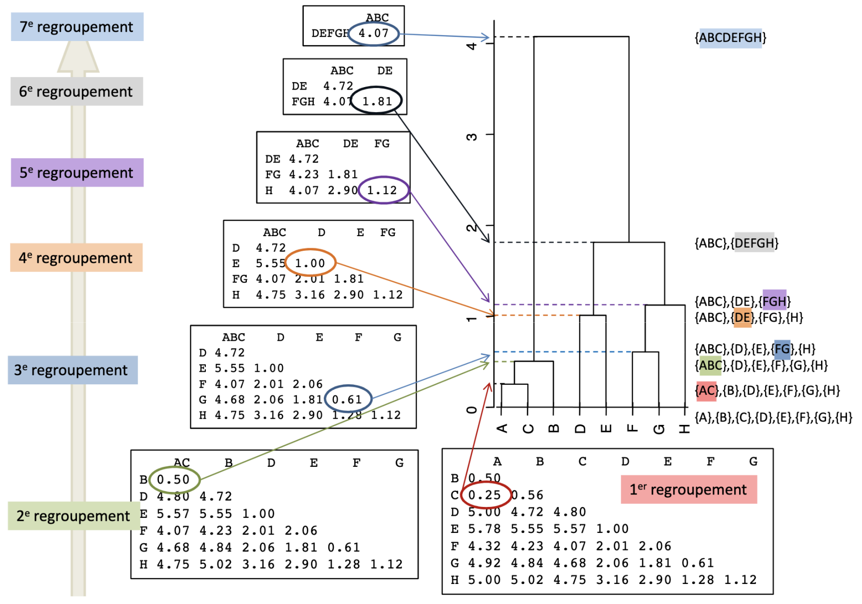
- Aggregate the two classes $\mathcal{C}_k$ and $\mathcal{C}_{k'}$ that minimize the aggregation measure $\mathcal{D}$: $\boxed{\mathcal{C}_{k\cup k'} = \mathcal{C}_k \cup \mathcal{C}_{k'}}$
- Form a partition into $K-1$ classes: $\mathcal{P}_{K-1}^{(t+1)} = \{\mathcal{C}_1, \ldots, \mathcal{C}_{k\cup k'}, \ldots, \mathcal{C}_K\}$

End: Repeat the aggregation step until a single-class partition is obtained.



Bisson (2001)

Agglomerative Hierarchical Classification



Data analysis MOOC of François Husson (in French).

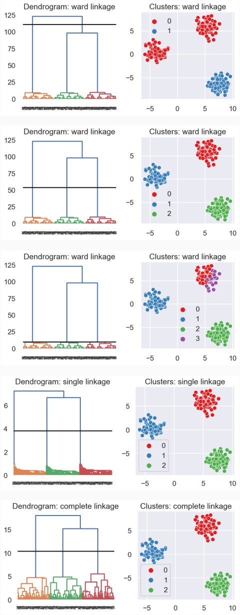
Missing bricks to implement classification

1. Choice of a dissimilarity d between points,
   To be made according to the type of data:
   Qualitative, Quantitative, etc.

2. Choice of an aggregation measure $\mathcal{D}$ between classes.

3. Construction of a dendrogram (not unique!).

4. Criterion for the cut of the dendrogram to deduce a classification of the data.



Package `scipy.cluster.hierarchy` $\rightsquigarrow$ See attached python notebook.

- `linkage`: `method='single','complete','average','ward'`, *etc.*
- `dendrogram` to draw the dendrogram,
- `cut_tree` to cut the dendrogram so that there are K clusters,
- `fcluster` to obtain a clustering from a dendrogram, at a given level

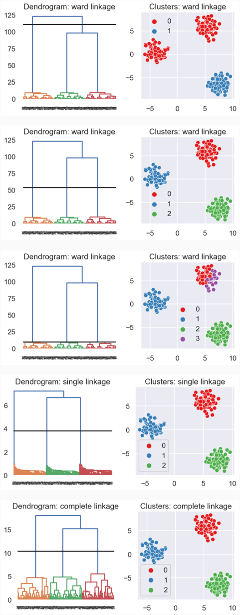
Missing bricks to implement classification

1. Choice of a dissimilarity d between points,
   To be made according to the type of data:
   Qualitative, Quantitative, etc.

2. Choice of an aggregation measure $\mathcal{D}$ between classes.

3. Construction of a dendrogram (not unique!).

4. Criterion for the cut of the dendrogram to deduce a classification of the data.



Package `scipy.cluster.hierarchy` $\rightsquigarrow$ See attached python notebook.

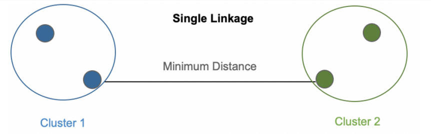
- `linkage`: `method='single','complete','average','ward'`, *etc.*
- `dendrogram` to draw the dendrogram,
- `cut_tree` to cut the dendrogram so that there are K clusters,
- `fcluster` to obtain a clustering from a dendrogram, at a given level

Dendrogram Construction

Single *vs.* Complete Linkage

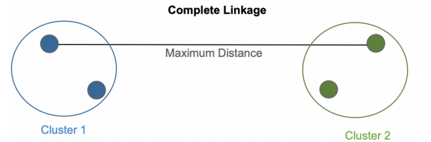
Single linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \min_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$



Complete linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \max_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

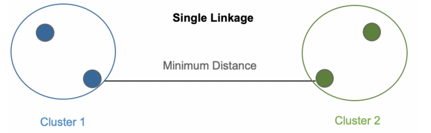


Single *vs.* Complete Linkage

Single linkage:

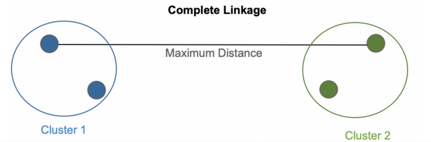
$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \min_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

+ Minimal spanning tree,



Complete linkage:

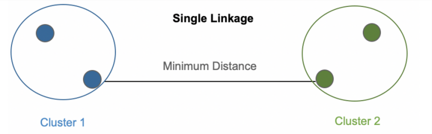
$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \max_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$



Single linkage:

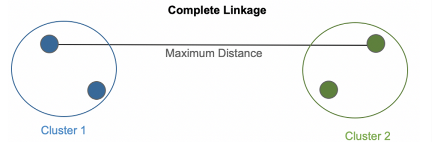
$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \min_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

+ Minimal spanning tree,
– Classes with very different diameters,
– Chaining effect: tendency to aggregate rather than create new classes
– Sensitivity to noisy individuals.



Complete linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \max_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

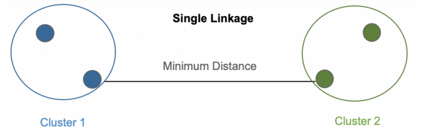


Single *vs.* Complete Linkage

Single linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \min_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

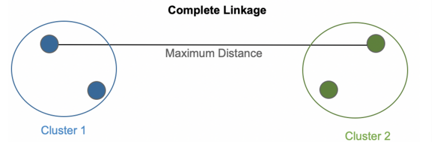
+ Minimal spanning tree,
– Classes with very different diameters,
– Chaining effect: tendency to aggregate rather than create new classes
– Sensitivity to noisy individuals.



Complete linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \max_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

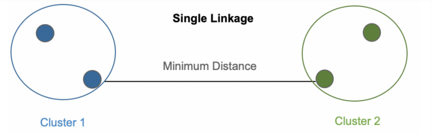
+ Creates compact classes (diameter control): this fusion generates the smallest increase in diameters,



Single linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \min_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

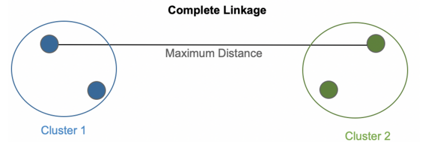
+ Minimal spanning tree,
– Classes with very different diameters,
– Chaining effect: tendency to aggregate rather than create new classes
– Sensitivity to noisy individuals.



Complete linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \max_{i \in \mathcal{C}_k,\, i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

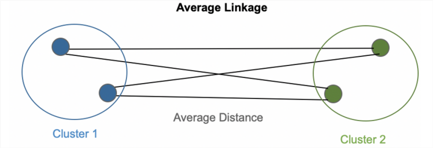
+ Creates compact classes (diameter control): this fusion generates the smallest increase in diameters,
– No separation control: arbitrarily close classes,
– Sensitivity to noisy individuals.



Average *vs.* Ward's Linkage

Average linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \frac{1}{|\mathcal{C}_k||\mathcal{C}_{k'}|} \sum_{i \in \mathcal{C}_k} \sum_{i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$



Ward's linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \frac{|\mathcal{C}_k||\mathcal{C}_{k'}|}{|\mathcal{C}_k| + |\mathcal{C}_{k'}|} d(\mu_k, \mu_{k'})^2}$$

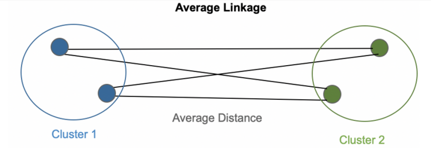
where $\mu_k/\mu_{k'}$ gravity centers of $\mathcal{C}_k/\mathcal{C}_{k'}$.

Average *vs.* Ward's Linkage

Average linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \frac{1}{|\mathcal{C}_k||\mathcal{C}_{k'}|} \sum_{i \in \mathcal{C}_k} \sum_{i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})}$$

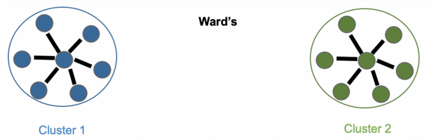
+ Compromise between single and complete linkages: good balance between class separation and class diameter,
+ Tendency to produce classes of close variance.



Ward's linkage:

$$\boxed{\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \frac{|\mathcal{C}_k||\mathcal{C}_{k'}|}{|\mathcal{C}_k| + |\mathcal{C}_{k'}|} d(\mu_k, \mu_{k'})^2}$$

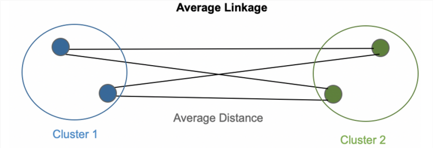
where $\mu_k/\mu_{k'}$ gravity centers of $\mathcal{C}_k/\mathcal{C}_{k'}$.



Average linkage:

$$\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \frac{1}{|\mathcal{C}_k||\mathcal{C}_{k'}|} \sum_{i \in \mathcal{C}_k} \sum_{i' \in \mathcal{C}_{k'}} d(x_i, x_{i'})$$

+ Compromise between single and complete linkages: good balance between class separation and class diameter,
+ Tendency to produce classes of close variance.

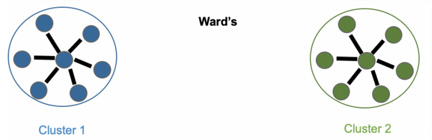


Ward's linkage:

$$\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) = \frac{|\mathcal{C}_k||\mathcal{C}_{k'}|}{|\mathcal{C}_k| + |\mathcal{C}_{k'}|} d(\mu_k, \mu_{k'})^2$$

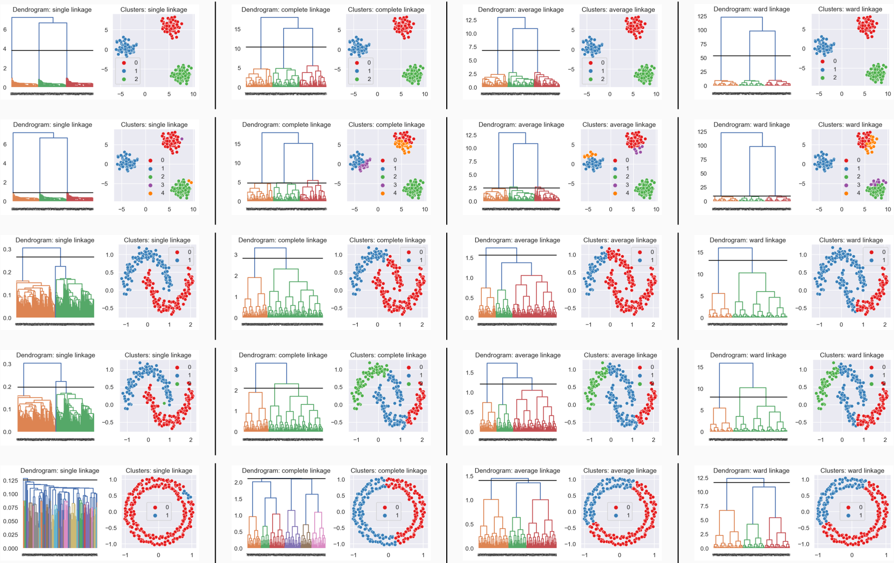
where $\mu_k/\mu_{k'}$ gravity centers of $\mathcal{C}_k/\mathcal{C}_{k'}$.

+ Tendency to build classes of equal size for a given level of hierarchy,
+ Groups together classes with close gravity centers,
+ Favors spherical classes.



Linkage Criteria

Single linkage	**Complete** linkage	**Average** linkage	**Ward** linkage



$K = 3$

$K = 5$

$K = 2$

$K = 3$

$K = 2$

Ward's Method [Ward, 1963]

Proposition

Let $\mathcal{P}_K = \{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$ a partition of the data and $k \neq \ell$.

$$\boxed{I_{Intra}(\mathcal{C}_{k \cup k'}) = I_{Intra}(\mathcal{C}_k) + I_{Intra}(\mathcal{C}_{k'}) + \frac{|\mathcal{C}_k||\mathcal{C}_{k'}|}{|\mathcal{C}_k| + |\mathcal{C}_{k'}|} d(\mu_k, \mu_{k'})^2}$$

where $\mu_k/\mu_{k'}$ gravity centers of $\mathcal{C}_k/\mathcal{C}_{k'}$, and d Euclidean distance.

Ward's method: *Choose at each step to group the two classes whose merging implies a minimal increase of the intraclass inertia.*

Reminder: $I_{Tot} = I_{Inter} + I_{Intra}$

where $I_{Intra} = \sum_{k=1}^{K} I_{Intra}(\mathcal{C}_k)$ and $I_{Intra}(\mathcal{C}_k) = \sum_{i \in \mathcal{C}_k} d(\mu_k, x_i)^2$.

Lance-Williams Algorithms

- **Naive implementation** of hierarchical clustering: Compute the distance matrix between each cluster at each step.

Lance-Williams Algorithms

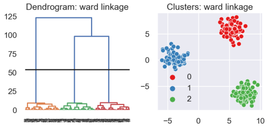
- **Naive implementation** of hierarchical clustering: Compute the distance matrix between each cluster at each step.
- **Lance-Williams algorithms**: Recursive formula for computing cluster distances at each step.

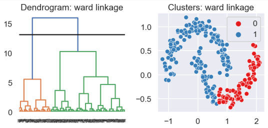
$$\boxed{\begin{aligned}\mathcal{D}(\mathcal{C}_\ell, \mathcal{C}_{k\cup k'}) \;=\; & \alpha\,\mathcal{D}(\mathcal{C}_\ell, \mathcal{C}_k) + \beta\,\mathcal{D}(\mathcal{C}_\ell, \mathcal{C}_{k'}) + \gamma\,\mathcal{D}(\mathcal{C}_k, \mathcal{C}_{k'}) \\ & + \delta\,|\mathcal{D}(\mathcal{C}_\ell, \mathcal{C}_k) - \mathcal{D}(\mathcal{C}_\ell, \mathcal{C}_{k'})|\end{aligned}}$$

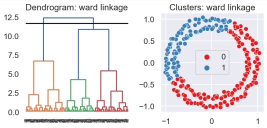
Linkage	α	β	γ	δ
Single	0.5	0.5	0	-0.5
Complete	0.5	0.5	0	0.5
Average	$\frac{\lvert\mathcal{C}_k\rvert}{\lvert\mathcal{C}_k\rvert + \lvert\mathcal{C}_{k'}\rvert}$	$\frac{\lvert\mathcal{C}_{k'}\rvert}{\lvert\mathcal{C}_k\rvert + \lvert\mathcal{C}_{k'}\rvert}$	0	0
Ward	$\frac{\lvert\mathcal{C}_k\rvert + \lvert\mathcal{C}_\ell\rvert}{\lvert\mathcal{C}_k\rvert + \lvert\mathcal{C}_{k'}\rvert + \lvert\mathcal{C}_\ell\rvert}$	$\frac{\lvert\mathcal{C}_{k'}\rvert + \lvert\mathcal{C}_\ell\rvert}{\lvert\mathcal{C}_k\rvert + \lvert\mathcal{C}_{k'}\rvert + \lvert\mathcal{C}_\ell\rvert}$	$-\frac{\lvert\mathcal{C}_\ell\rvert}{\lvert\mathcal{C}_k\rvert + \lvert\mathcal{C}_{k'}\rvert + \lvert\mathcal{C}_\ell\rvert}$	0

- In general, $\forall A, B \in \mathcal{H}$, $\boxed{h(A \cup B) = \mathcal{D}(A, B)}$
- If $(\mathcal{H}, h)$ defined in this way does not verify the properties of an indexed hierarchy, we can use the following relation:

$$\forall A, B \in \mathcal{H}\,, \quad h(A \cup B) = \max\{\mathcal{D}(A, B),\, h(A),\, h(B)\}\,.$$



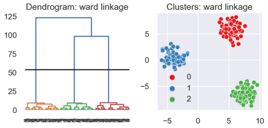


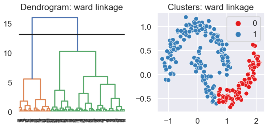


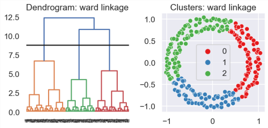
Dendrogram Construction

- Cutting the dendrogram at a given index level $\Longrightarrow$ **Partition**.
  i.e. cut-off level determines the nb of classes and these classes are then unique.

- The cut-off should be done :
  - After aggregations corresponding to low values of the index,
  - Before aggregations corresponding to high levels of the index, which dissociate the well-distinct groups of the population.







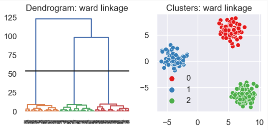
Cutting the Dendrogram

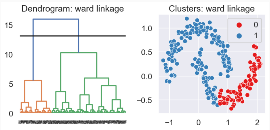
- Cutting the dendrogram at a given index level $\Longrightarrow$ **Partition**.
  i.e. cut-off level determines the nb of classes and these classes are then unique.

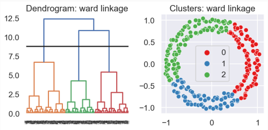
- The cut-off should be done :
  - After aggregations corresponding to low values of the index,
  - Before aggregations corresponding to high levels of the index, which dissociate the well-distinct groups of the population.

- **Empirical rule**: Selection of a cut when there is a significant jump in the index by visual inspection of the tree.
  This jump reflects the sudden passage from classes of a certain homogeneity to much less homogeneous classes.

- In most cases, several thresholds and therefore several possible choices of partitions.







Cutting the Dendrogram: Some Criteria

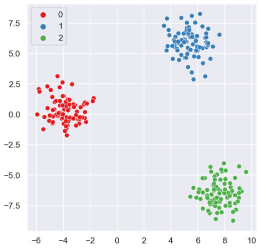
- The dendrogram cut-off can be defined by determining a priori the number of classes into which we want to divide the data set.

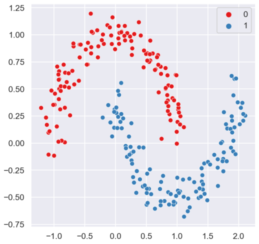
  For this, we can use the **usual criteria**:

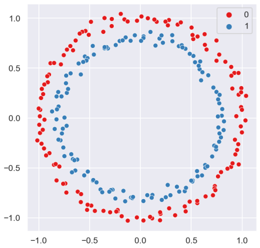
  - R-square (RSQ): *Elbow on the curve* $K \mapsto RSQ(K)$*,*
  - Semi-partial R-square ($SPRSQ$): *Stronger reduction of the* $SPRSQ$*,*
  - Calinski-Harabasz: *Peak on the curve*
  - Silhouette criterion,
  - *etc.*

Pros:

- Easy consideration of distances and similarities of any type,
- No assumption of a particular number of clusters,
- May correspond to meaningful taxonomies.



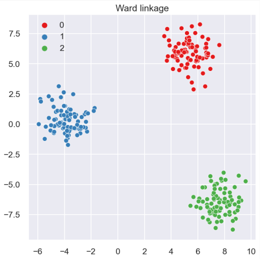


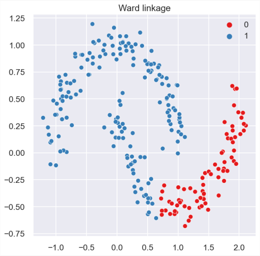


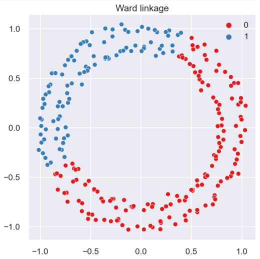
Strengths and Weaknesses

Pros:
- Easy consideration of distances and similarities of any type,
- No assumption of a particular number of clusters,
- May correspond to meaningful taxonomies.

Cons:
- Choice of the dendrogram cut-off,
- The partition obtained at a step depends on the one at the previous step,
- Once a decision is made to combine two clusters, it can't be undone,
- Too slow for large data sets.







References i

Bisson, G. (2000). **La similarité: une notion symbolique/numérique.** Apprentissage symbolique-numérique, 2:169–201.

Janssen, P. (2012). **Cluster analysis to understand socio-ecological systems: a guideline.**

Johnson, S. C. (1967). **Hierarchical clustering schemes.** Psychometrika, 32(3):241–254.

Ward, J. H. (1963). **Hierarchical grouping to optimize an objective function.** Journal of the American statistical association, 58(301):236–244.