Multivariate Verfahren

7.1 Unsupervised Learning: Clustering

Hannah Schulz-Kümpel

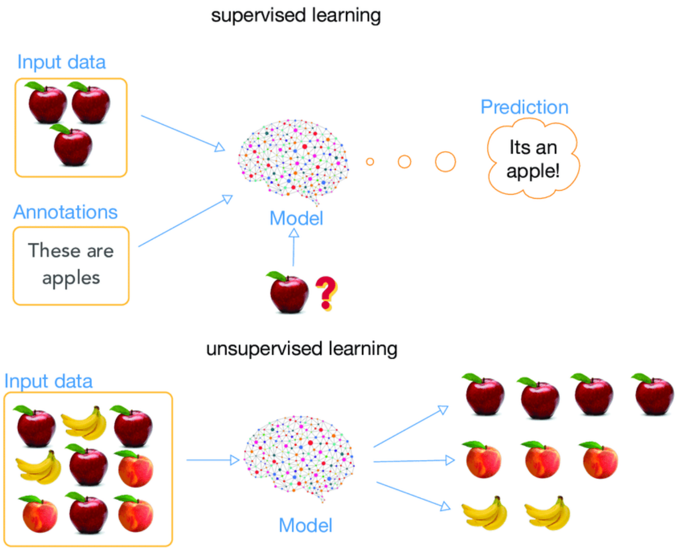
based on lecture slides by Sabine Hoffmann

Institut für Statistik, LMU München

Sommersemester 2024

Contents

What is unsupervised Learning? We recall:



What is unsupervised Learning? I

In contrast to supervised learning, *unsupervised learning methods*

- are applied to data that is **not labelled**:

$$\mathcal{D} = \{x_1, \ldots, x_n\} \in \mathcal{X}^n$$

  (i.e. no y_is)

- and aim at "*making inferences about the structure of $\mathcal{D}$* ".

What is unsupervised Learning? II

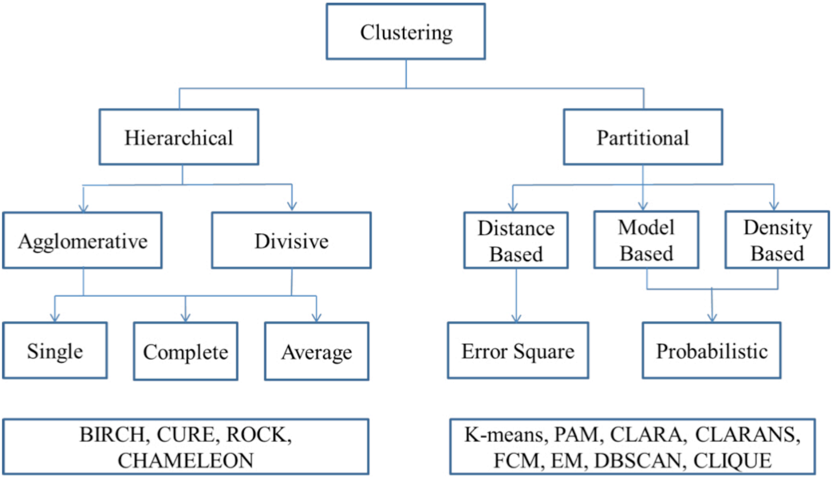
This goal is, admittedly, much vaguer than "finding the optimal parameter θ", but unsupervised learning methods have a lot of relevant applications, like

- data visualization,
- exploratory data analysis,
- grouping objects $\longrightarrow$ *this lecture*,
- dimensionality reduction $\longrightarrow$ *following lecture(s)*

Clustering

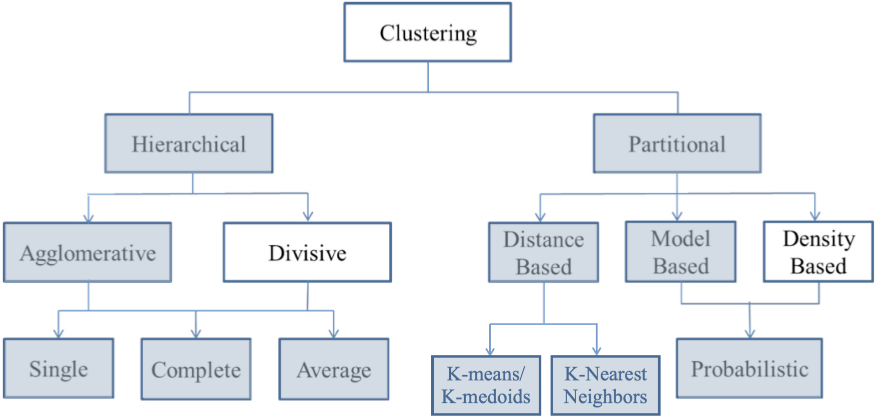
- *Clustering* refers to unsupervised learning methods aimed at grouping similar data points together.
- So we start with a sequence (or tupel) of data points $\mathcal{D} = \{x_1, \dots, x_n\}$ with the goal of assigning each point to one of K separate clusters.
- Applications of clustering methods include: *Image Processing*, *Genomics*, *Anomaly Detection*, *Document Categorization*, etc.
- Can clustering algorithms also be used for supervised learning? *Yes!* See the very end of these slides.

One way to categorize Clustering methods



Source: Saxena, Amit Kumar et al. "*A review of clustering techniques and developments*." Neurocomputing 267 (2017): 664-681.

Clustering methods covered in this class

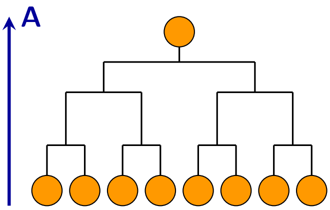


Hierarchical Clustering

Idea behind hierarchical clustering

Agglomerative methods

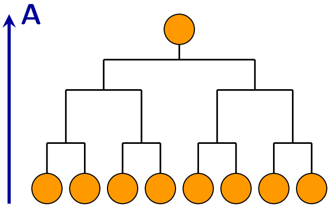
- Start with n objects as individual clusters.
- In each step, the two closest clusters are summarized.



Idea behind hierarchical clustering

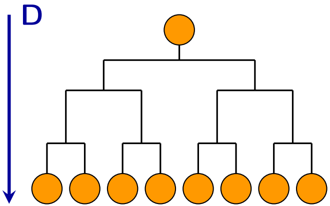
Agglomerative methods

- Start with n objects as individual clusters.
- In each step, the two closest clusters are summarized.



Divisive methods

- Start with all n objects in a cluster.
- A cluster is split in each step.



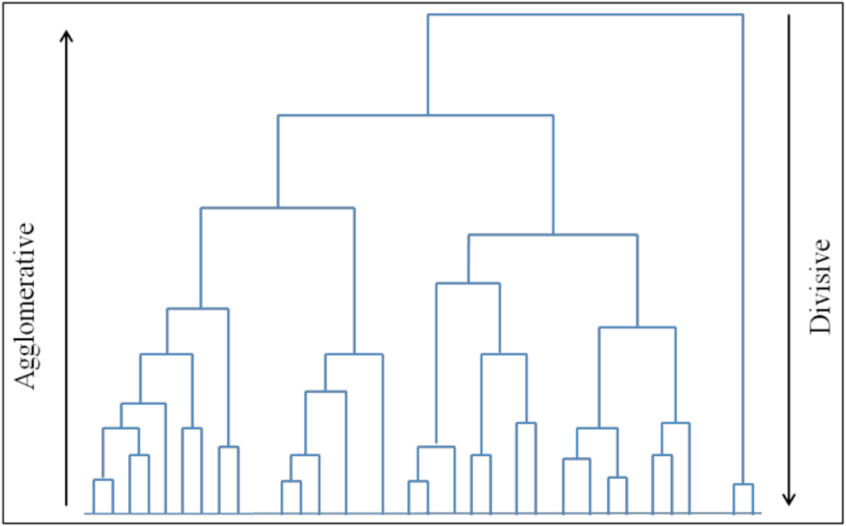
Hierarchical clustering

Form a hierarchy of partitions $\mathbb{C} = \{C_1, \ldots, C_g\}$ according to one of the following two principles:

- Agglomerative methods
  Start with the partition $\mathbb{C}^{(0)} = \{\{x_1\}, \ldots, \{x_n\}\}$, in which each observation forms its own cluster and successively *merge* the clusters.

- Divisive methods
  Start with the partition $\mathbb{C}^{(0)} = \{x_1, \ldots, x_n\}$, where all observations form a single cluster and successively *divide* the clusters.

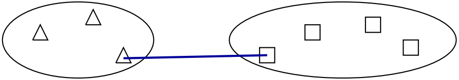
For both methods, the hierarchy of partitions $\mathbb{C}^{(0)}, \ldots, \mathbb{C}^{(n)}$ may be visualized as a *dendogram*:



Source: Saxena, Amit Kumar et al. "*A review of clustering techniques and developments*." Neurocomputing 267 (2017): 664-681.

Linkage: Quantifying distances between classes

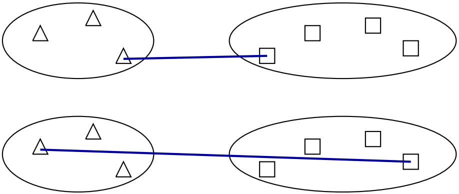
Linkage: Quantifying distances between classes



Single linkage:

- Minimal distance between clusters
- Nearest Neighbor

Linkage: Quantifying distances between classes



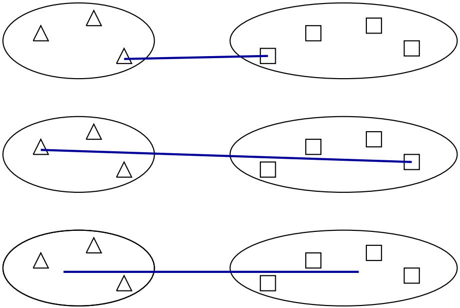
Single linkage:

- Minimal distance between clusters
- Nearest Neighbor

Complete linkage:

- Maximal distance between objects

Linkage: Quantifying distances between classes



Single linkage:

- Minimal distance between clusters
- Nearest Neighbor

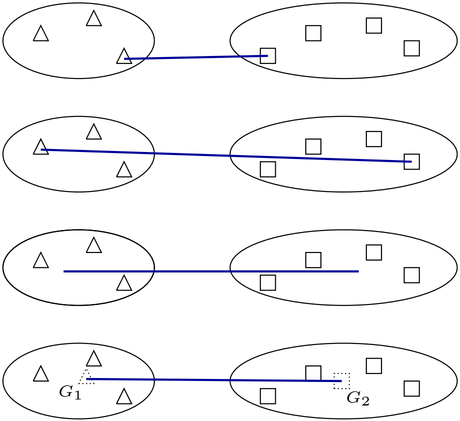
Complete linkage:

- Maximal distance between objects

Average linkage:

- Mean deviation of all pairwise distances

Linkage: Quantifying distances between classes



Single linkage:

- Minimal distance between clusters
- Nearest Neighbor

Complete linkage:

- Maximal distance between objects

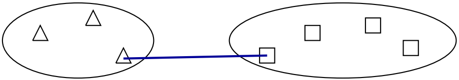
Average linkage:

- Mean deviation of all pairwise distances

Zentroid procedure:

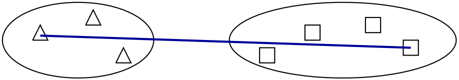
- Distance between cluster centroids

Linkage: Quantifying distances between classes



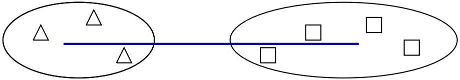
Single linkage:

- Minimal distance between clusters
- Nearest Neighbor



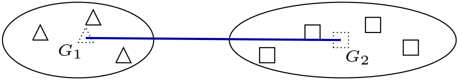
Complete linkage:

- Maximal distance between objects



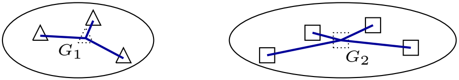
Average linkage:

- Mean deviation of all pairwise distances



Zentroid procedure:

- Distance between cluster centroids



Ward method:

- Consider the inertia within the clusters
- Analog to k-means (but not optimal)

Agglomerative hierarchical Clustering I

- In this class, we focus on **agglomerative hierarchical clustering methods**.
- Here, we require:
  - A distance measure to quantify the distance between objects
  - A distance measure to quantify the distance between classes
- The objects (classes) with the shortest distance are grouped together.
- This procedure is carried out until all objects are combined into one class.

Agglomerative hierarchical Clustering II

- Start partition: $\mathbb{C}^{(0)} = \{C_1^{(0)} = \{x_1\}, \ldots, C_n^{(0)} = \{x_n\}\}$

- In the νth step, merge those clusters

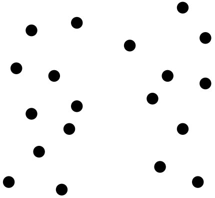
$$C_r^{(\nu)},\ C_s^{(\nu)}, \quad r \neq s\,,$$

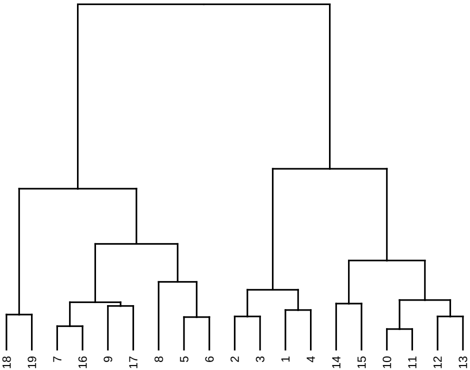
  which have the smallest distance D.

- The distance between the objects is determined by a distance measure, the distance between the clusters by the Linkage.

Agglomerative hierarcical clustering: Algorithm

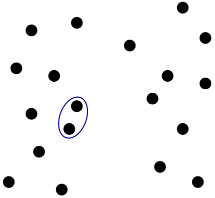
Step 1: Find the two elements that are closest to each other and combine them.

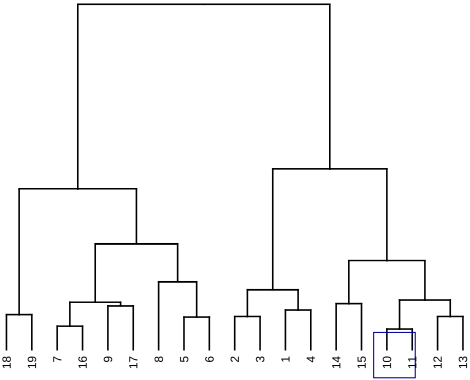




Agglomerative hierarcical clustering: Algorithm

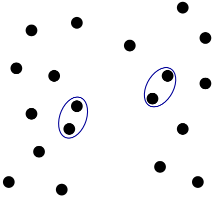
Step 1: Find the two elements that are closest to each other and combine them.

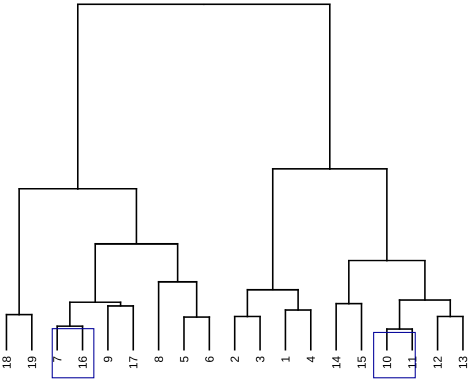




Agglomerative hierarcical clustering: Algorithm

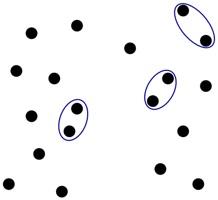
Step 1: Find the two elements that are closest to each other and combine them.

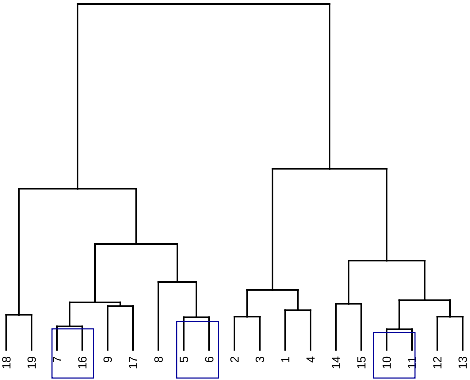




Agglomerative hierarcical clustering: Algorithm

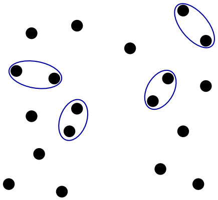
Step 1: Find the two elements that are closest to each other and combine them.

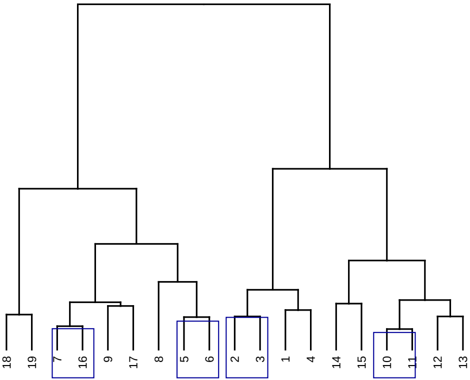




Agglomerative hierarcical clustering: Algorithm

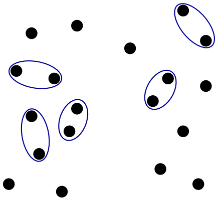
Step 1: Find the two elements that are closest to each other and combine them.

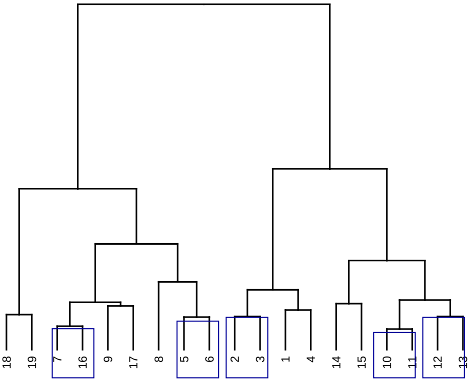




Agglomerative hierarcical clustering: Algorithm

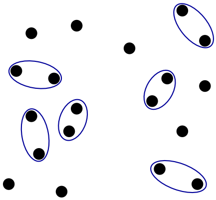
Step 1: Find the two elements that are closest to each other and combine them.

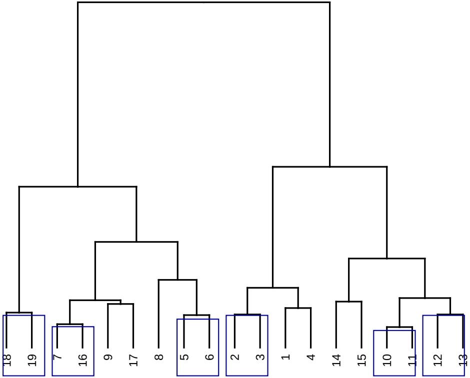




Agglomerative hierarcical clustering: Algorithm

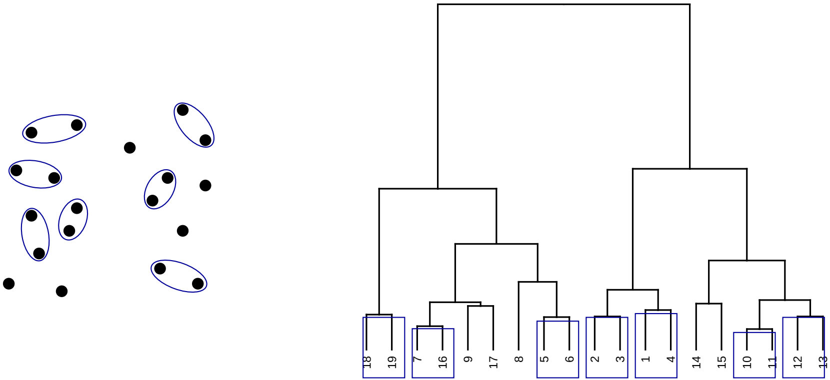
Step 1: Find the two elements that are closest to each other and combine them.





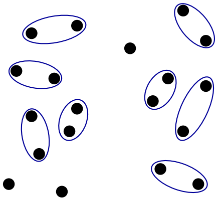
Agglomerative hierarcical clustering: Algorithm

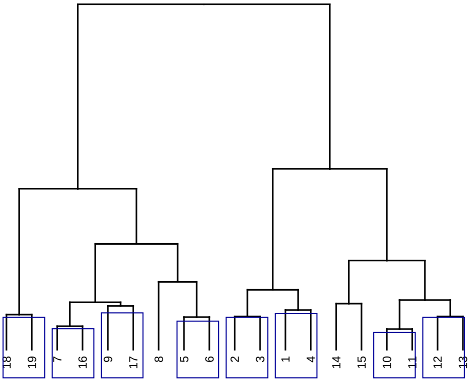
Step 1: Find the two elements that are closest to each other and combine them.



Agglomerative hierarcical clustering: Algorithm

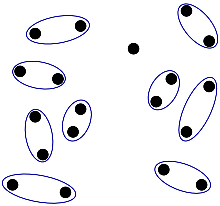
Step 1: Find the two elements that are closest to each other and combine them.

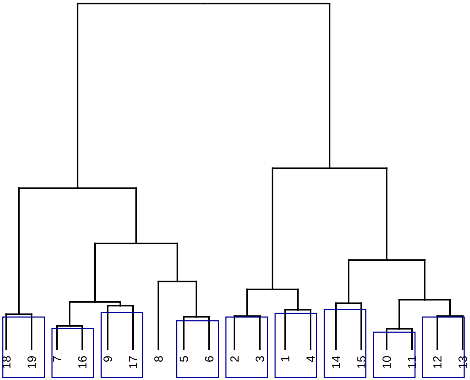




Agglomerative hierarcical clustering: Algorithm

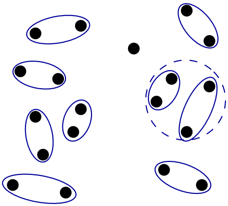
Step 1: Find the two elements that are closest to each other and combine them.

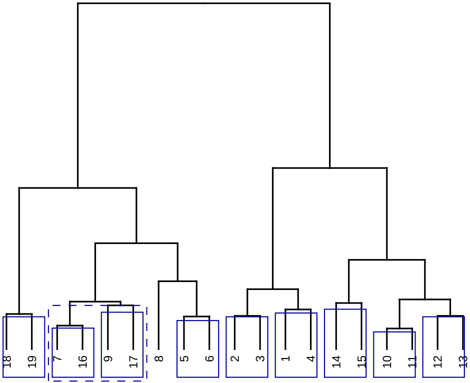




Agglomerative hierarcical clustering: Algorithm

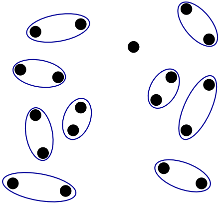
Step 2: Using some type of Linkage, determine the distance between the clusters.

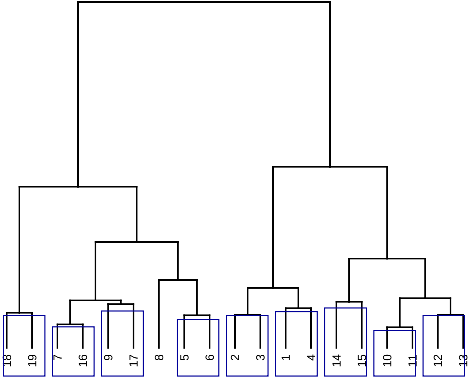




Agglomerative hierarcical clustering: Algorithm

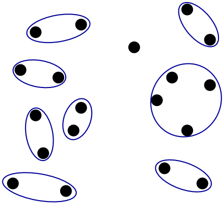
Step 2: Find the two clusters that are closest to each other and merge them.

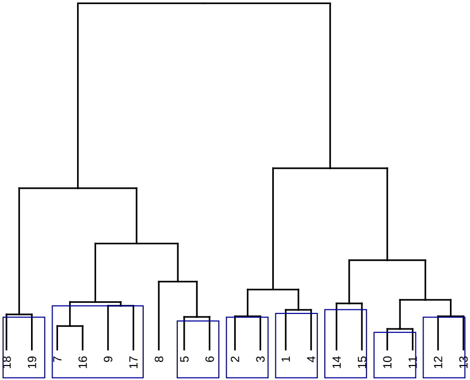




Agglomerative hierarcical clustering: Algorithm

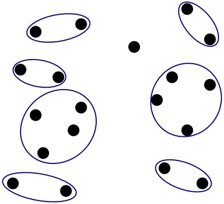
Step 2: Find the two clusters that are closest to each other and merge them.

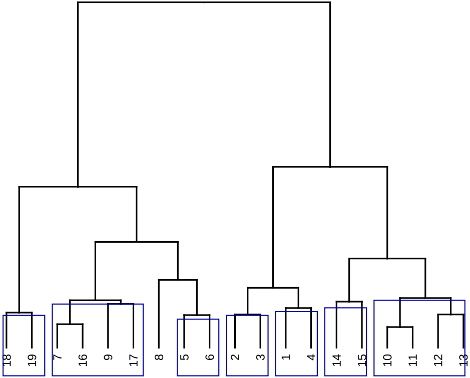




Agglomerative hierarcical clustering: Algorithm

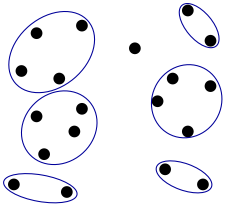
Step 2: Find the two clusters that are closest to each other and merge them.

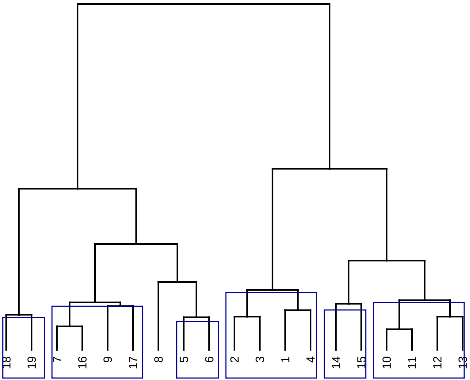




Agglomerative hierarcical clustering: Algorithm

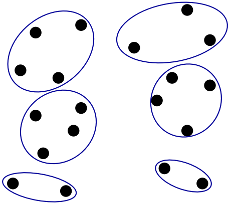
Step 2: Find the two clusters that are closest to each other and merge them.

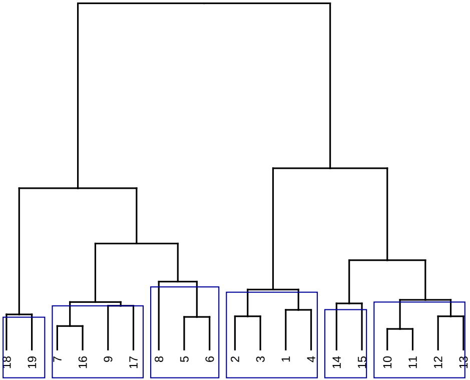




Agglomerative hierarcical clustering: Algorithm

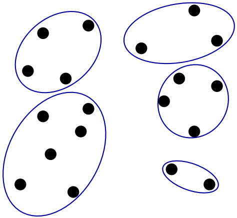
Step 2: Find the two clusters that are closest to each other and merge them.

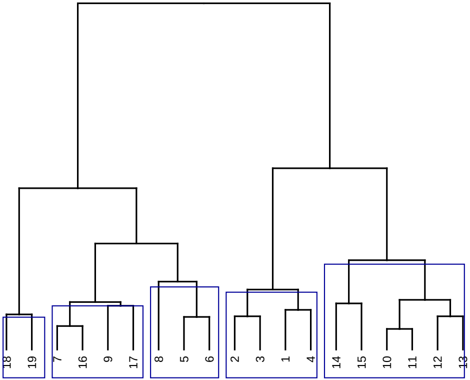




Agglomerative hierarcical clustering: Algorithm

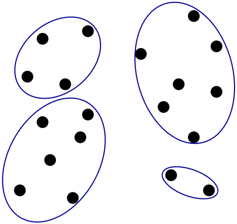
Step 2: Find the two clusters that are closest to each other and merge them.

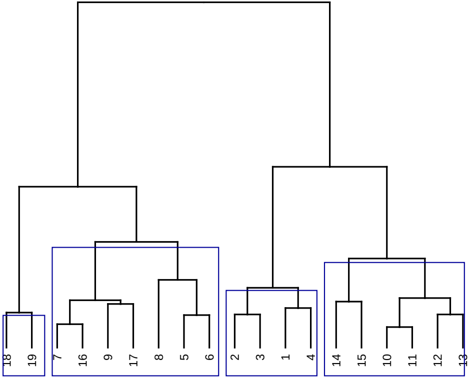




Agglomerative hierarcical clustering: Algorithm

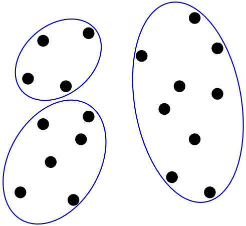
Step 2: Find the two clusters that are closest to each other and merge them.

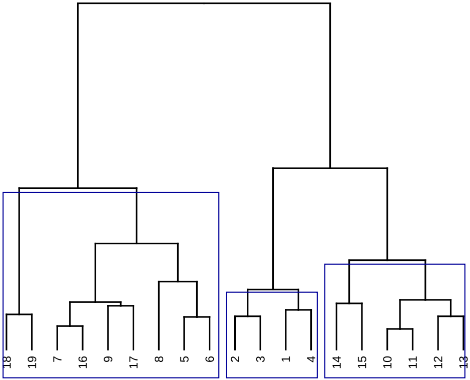




Agglomerative hierarcical clustering: Algorithm

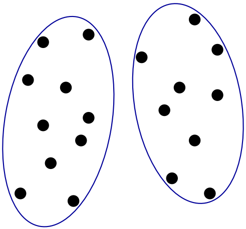
Step 2: Find the two clusters that are closest to each other and merge them.

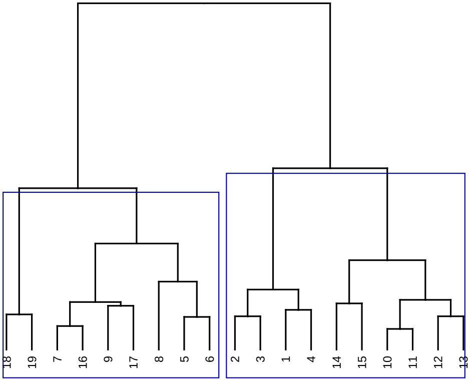




Agglomerative hierarcical clustering: Algorithm

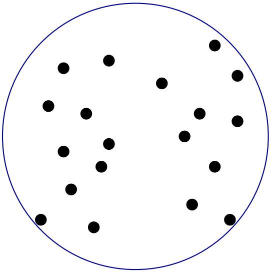
Step 2: Find the two clusters that are closest to each other and merge them.

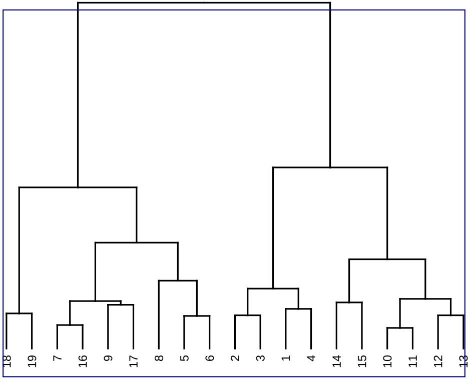




Agglomerative hierarcical clustering: Algorithm

Step 3: Stop when all objects are combined.





Example: agglomerative procedure

- Consider the age of 6 persons: 43, 38, 6, 47, 37, 9

- Determine the Euclidean distance between 2 people:

$$\mathbf{D} = \begin{pmatrix} 0 & 5 & 37 & 4 & 6 & 34 \\ 5 & 0 & 32 & 9 & 1 & 29 \\ 37 & 32 & 0 & 41 & 31 & 3 \\ 4 & 9 & 41 & 0 & 10 & 38 \\ 6 & 1 & 31 & 10 & 0 & 28 \\ 34 & 29 & 3 & 38 & 28 & 0 \end{pmatrix}$$

Merge classes with the smallest distance $\Rightarrow$ Merge classes 2 and 5 (ages 37 and 38)

Example: agglomerative procedure

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3\}$	$\{4\}$	$\{6\}$
$\{2,5\}$					
$\{1\}$					
$\{3\}$		37			
$\{4\}$		4	41		
$\{6\}$		34	3	38	

How is the distance between {2, 5} and the other classes determined?

- We use some type of *Linkage* to calculate the distance $D(C_r, C_s)$ with $C_r, C_s \subset \mathbb{C}^{(i)}$ for step i.

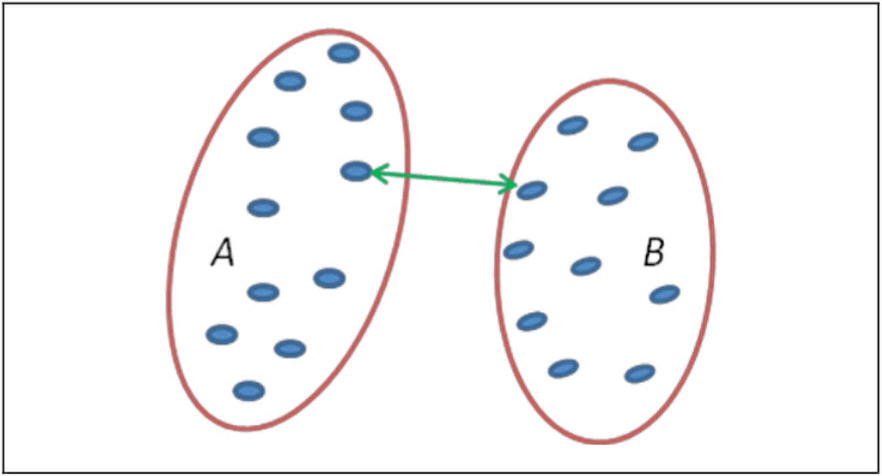
Single Linkage

For Single Linkage, we have

$$D_{SL}(C_r, C_s) = \min_{\substack{x_i \in C_r \\ x_j \in C_s}} \{d(x_i, x_j)\}$$

- "Nearest Neighbor"
- Robust against small changes to individual data points
- Risk of chain formation or bridge formation
- Application in taxonomy

$$\min\left\{d(a,b) : a \in A, b \in B\right\}$$



Example: Single Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9
- $D_{SL}(\{2,5\},\{1\}) = \min\{d_{21}, d_{51}\} = \min\{5,6\} = 5$
- $D_{SL}(\{2,5\},\{3\}) = \min\{d_{23}, d_{53}\} = \min\{32,31\} = 31$
- $D_{SL}(\{2,5\},\{4\}) = \min\{d_{24}, d_{54}\} = \min\{9,10\} = 9$
- $D_{SL}(\{2,5\},\{6\}) = \min\{d_{26}, d_{56}\} = \min\{29,28\} = 28$

Example: Single Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3\}$	$\{4\}$	$\{6\}$
$\{2,5\}$					
$\{1\}$	5				
$\{3\}$	31	37			
$\{4\}$	9	4	41		
$\{6\}$	28	34	3	38	

$\Rightarrow$ Merge classes 3 and 6 (age 6 and 9)

Example: Single Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3,6\}$	$\{4\}$
$\{2,5\}$				
$\{1\}$	5			
$\{3,6\}$	28	34		
$\{4\}$	9	4	38	

$\Rightarrow$ Merge classes 1 and 4 (age 43 and 47)

Example: Single Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1,4\}$	$\{3,6\}$
$\{2,5\}$			
$\{1,4\}$	5		
$\{3,6\}$	28	34	

$\Rightarrow$ Merge {1,4} and {2,5}

Example: Single Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{1,2,4,5\}$	$\{3,6\}$
$\{1,2,4,5\}$		
$\{3,6\}$	28	

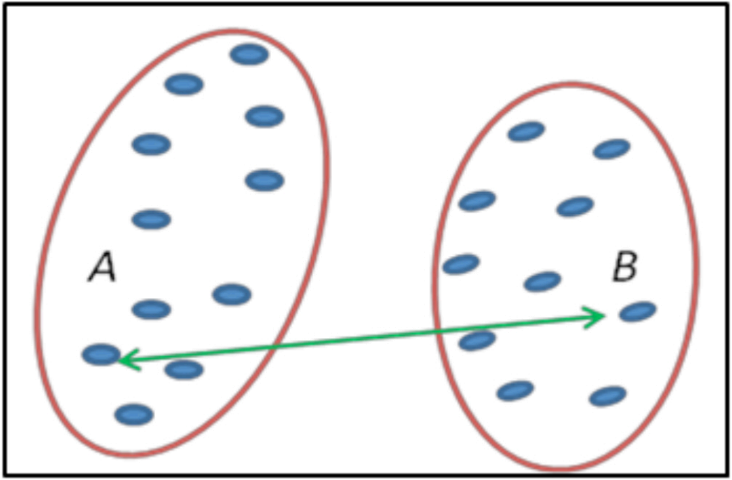
Complete Linkage

For Complete Linkage, we have

$$D_{CL}(C_r, C_s) = \max_{\substack{x_i \in C_r \\ x_j \in C_s}} d(x_i, x_j)$$

- "Furthest Neighbor"
- Large clusters grow slowly
- Instability with regard to small changes
- Suitable for splitting data without a clear structure

$$\max\left\{d(a,b) : a \in A, b \in B\right\}$$



Example: Complete Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9
- $D_{SL}(\{2,5\},\{1\}) = \max\{d_{21}, d_{51}\} = \max\{5,6\} = 6$
- $D_{SL}(\{2,5\},\{3\}) = \max\{d_{23}, d_{53}\} = \max\{32,31\} = 32$
- $D_{SL}(\{2,5\},\{4\}) = \max\{d_{24}, d_{54}\} = \max\{9,10\} = 10$
- $D_{SL}(\{2,5\},\{6\}) = \max\{d_{26}, d_{56}\} = \max\{29,28\} = 29$

Example: Average Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3\}$	$\{4\}$	$\{6\}$
$\{2,5\}$					
$\{1\}$	6				
$\{3\}$	32	37			
$\{4\}$	10	4	41		
$\{6\}$	29	34	3	38	

$\Rightarrow$ Merge classes 3 and 6 (age 6 and 9)

Example: Average Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3,6\}$	$\{4\}$
$\{2,5\}$				
$\{1\}$	6			
$\{3,6\}$	32	37		
$\{4\}$	10	4	41	

$\Rightarrow$ Merge classes 1 and 4 (age 43 and 47)

Example: Complete Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1,4\}$	$\{3,6\}$
$\{2,5\}$			
$\{1,4\}$	10		
$\{3,6\}$	32	41	

$\Rightarrow$ Merge {1,4} and {2,5}

Example: Complete Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{1,2,4,5\}$	$\{3,6\}$
$\{1,2,4,5\}$		
$\{3,6\}$	41	

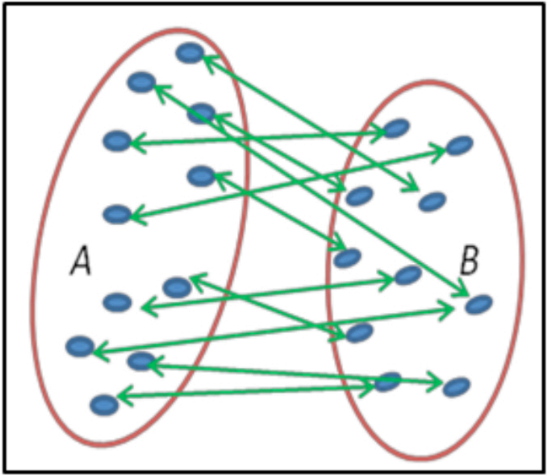
Average Linkage

For Average Linkage, we have

$$
\begin{aligned}
D_{AL}(C_r, C_s) &= \frac{1}{n_r n_s} \sum_{x_i \in C_r} \sum_{x_j \in C_s} d(x_i, x_j) \\
\text{with} \quad n_i &= | \, C_i \, |
\end{aligned}
$$

- Compromise between complete linkage and single linkage
- Averaging should make sense

$$\frac{1}{|A||B|}\sum_{a\in A}\sum_{b\in B} d(a,b)$$



Example: Average Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9
- $D_{AL}(\{2,5\},\{1\}) = \frac{d_{21}+d_{51}}{2} = \frac{5+6}{2} = 5.5$
- $D_{AL}(\{2,5\},\{3\}) = \frac{d_{23}+d_{53}}{2} = \frac{32+31}{2} = 31.5$
- $D_{AL}(\{2,5\},\{4\}) = \frac{d_{24}+d_{54}}{2} = \frac{9+10}{2} = 9.5$
- $D_{AL}(\{2,5\},\{6\}) = \frac{d_{26}+d_{56}}{2} = \frac{29+28}{2} = 28.5$

Example: Average Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3\}$	$\{4\}$	$\{6\}$
$\{2,5\}$					
$\{1\}$	5.5				
$\{3\}$	31.5	37			
$\{4\}$	9.5	4	41		
$\{6\}$	28.5	34	3	38	

Example: Average Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9
- $D_{AL}(\{3,6\},\{2,5\}) = \frac{d_{32}+d_{35}+d_{62}+d_{65}}{5} = \frac{32+31+29+28}{2} = 30$
- $D_{AL}(\{3,6\},\{1\}) = \frac{d_{31}+d_{61}}{2} = \frac{37+34}{2} = 35.5$
- $D_{AL}(\{3,6\},\{4\}) = \frac{d_{34}+d_{64}}{2} = \frac{41+38}{2} = 39.5$

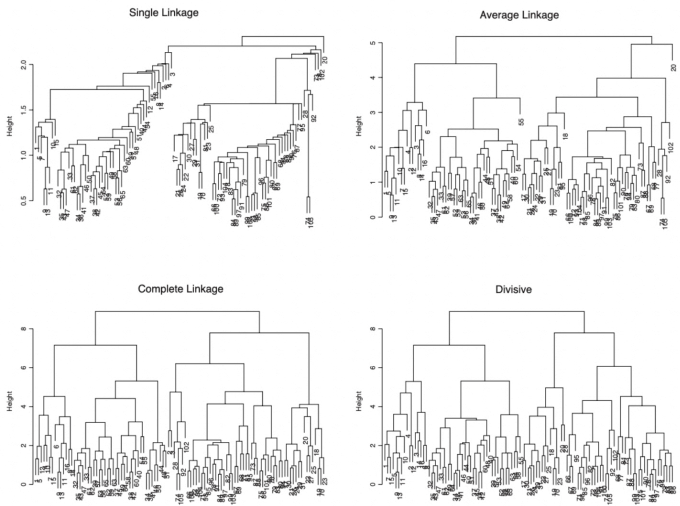
Example: Average Linkage

- We are still considering the age of 6 persons: 43, 38, 6, 47, 37, 9

	$\{2,5\}$	$\{1\}$	$\{3,6\}$	$\{4\}$
$\{2,5\}$				
$\{1\}$	5.5			
$\{3,6\}$	30	35.5		
$\{4\}$	9.5	4	39.5	

- And so on...

Comparison of Single, Average, and Complete Linkage



Zentroid-Procedure

For the Zentroid-Procedure, we have

$$\begin{aligned} D_Z(C_r, Cs) &= \| \bar{\mathbf{x}}_r - \bar{\mathbf{x}}_s \|^2 \\ \text{with } \bar{\mathbf{x}}_i &= \frac{1}{n_i} \sum_{j \in C_i} \mathbf{x_j} \end{aligned}$$

- Note: This procedure is only suited for metric data points.

Example: Zentroid-Procedure

- Next, let's consider the age of 4 persons: 19, 25, 20, 23

	{1}	{2}	{3}	{4}
{1}				
{2}	36			
{3}	1	25		
{4}	16	4	9	

Example: Zentroid-Procedure

- Again considering the age of 4 persons: 19, 25, 20, 23
- Mean of class $\{1,3\} = (19+20)/2 = 19.5$
- $D_Z(\{1,3\},\{2\}) = (19.5-25)^2 = 30.25$
- $D_Z(\{1,3\},\{4\}) = (19.5-23)^2 = 12.25$

Example: Zentroid-Procedure

- Again considering the age of 4 persons: 19, 25, 20, 23

	$\{1,3\}$	$\{2\}$	$\{4\}$
$\{1,3\}$			
$\{2\}$	30.25		
$\{4\}$	12.25	4	

Example: Zentroid-Procedure

- Again considering the age of 4 persons: 19, 25, 20, 23
- Mean of class $\{2, 4\} = (25 + 23)/2 = 24$
- $D_Z(\{1, 3\}, \{2, 4\}) = (19.5 - 24)^2 = 20.25$

Comparison of Zentroid and Average-Linkage when using the squared euclidean distance

$$
\begin{aligned}
D_{AL}(C_r, C_s) &= \frac{1}{n_r n_s} \sum_{x_i \in C_r} \sum_{x_\ell \in C_s} \{d(x_i, x_\ell)\} \\
&= \frac{1}{n_r n_s} \sum_{\mathbf{x}_i \in C_r} \sum_{\mathbf{x}_\ell \in C_s} \|\mathbf{x}_i - \mathbf{x}_\ell\|^2 \\
&= \|\bar{\mathbf{x}}_r - \bar{\mathbf{x}}_s\|^2 + \frac{1}{n_r} \sum_{\mathbf{x}_i \in C_r} \|\mathbf{x}_i - \bar{\mathbf{x}}_r\|^2 + \frac{1}{n_s} \sum_{\mathbf{x}_\ell \in C_s} \|\mathbf{x}_\ell - \bar{\mathbf{x}}_s\|^2 \\
&= D_Z(C_r, C_s) + s_r^2 + s_s^2
\end{aligned}
$$

$\rightarrow$ Average linkage takes into account the distance between the centers of gravity and the spread around it.

Ward method

Motivation: Merge the two clusters that generate the minimum increase in variance (heterogeneity) in the new cluster:

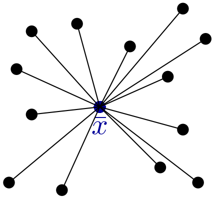
$$H(\mathbb{C}) = \sum_{r=1}^{g} \sum_{\mathbf{x}_i \in C_r} \left\| \mathbf{x}_i - \bar{\mathbf{x}}_r \right\|^2$$

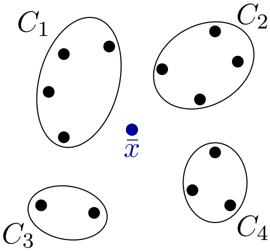
with

$$\bar{\mathbf{x}}_r = \frac{1}{n_r} \sum_{x_i \in C_r} \mathbf{x}_i$$

Ward method: properties of Inertia I

The overall inertia may be devided in:

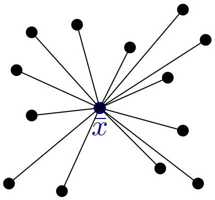


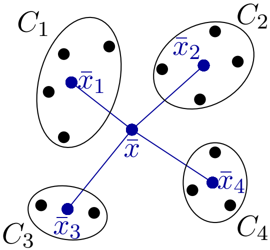


Ward method: properties of Inertia I

The overall inertia may be devided in:

- Inertia **between the K clusters** $C_k,\ k = 1, \ldots, K$

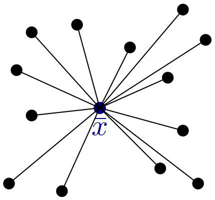


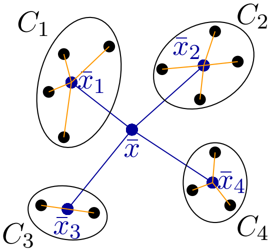


Ward method: properties of Inertia I

The overall inertia may be devided in:

- Inertia **between the K clusters** $C_k,\ k = 1, \ldots, K$
- Inertia **within the clusters** (sum of inertia in the K clusters)





Ward method: properties of Inertia II

- total inertia = inertia between the cluaters + inertia within the clusters

$$I_G = \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{x}_i - \bar{\boldsymbol{x}}\|^2 = \sum_{k=1}^{K} \frac{n_k}{n} \|\bar{\boldsymbol{x}}_k - \bar{\boldsymbol{x}}\|^2 + \frac{1}{n}\sum_{k=1}^{K}\sum_{i\in C_k} \|\boldsymbol{x}_i - \bar{\boldsymbol{x}}_k\|^2$$

- The total inertia is constant, we aim **to minimize the within cluster inertia** (similar to maximizing the between cluster inertia)

Example: Ward method

- Again considering the age of 4 persons: 19, 25, 20, 23
- Consider all possibilities $\mathbb{C} = \{\{1\},\{2\},\{3\},\{4\}\}$ for merging

$\mathbb{C} = \{\{1,2\},\{3\},\{4\}\}$:

$$H(\mathbb{C}) = (19-22)^2 + (25-22)^2 + (20-20)^2 + (23-23)^2 = 18$$

$\mathbb{C} = \{\{1,3\},\{2\},\{4\}\}$:

$$H(\mathbb{C}) = (19-19.5)^2 + (25-25)^2 + (20-19.5)^2 + (23-23)^2 = 0.5$$

$\mathbb{C} = \{\{1,4\},\{2\},\{3\}\}$:

$$H(\mathbb{C}) = (19-21)^2 + (25-25)^2 + (20-20)^2 + (23-21)^2 = 8$$

$\mathbb{C} = \{\{2,3\},\{1\},\{4\}\}$:

$$H(\mathbb{C}) = (19-19)^2 + (25-22.5)^2 + (20-22.5)^2 + (23-23)^2 = 12.5$$

$\mathbb{C} = \{\{2,4\},\{1\},\{3\}\}$:

$$H(\mathbb{C}) = (19-19)^2 + (25-24)^2 + (20-20)^2 + (23-24)^2 = 2$$

$\mathbb{C} = \{\{3,4\},\{1\},\{2\}\}$:

$$H(\mathbb{C}) = (19-19)^2 + (25-25)^2 + (20-21.5)^2 + (23-21.5)^2 = 4.5$$

Example: Ward method

- Again considering the age of 4 persons: 19, 25, 20, 23
- Now, consider all possibilities for merging on $\mathbb{C} = \{\{1,3\},\{2\},\{4\}\}$

$\mathbb{C} = \{\{1,2,3\},\{4\}\}$:

$$H(\mathbb{C}) = (19-21.33)^2+(25-21.33)^2+(20-21.33)^2+(23-23)^2 = 20.67$$

$\mathbb{C} = \{\{1,3,4\},\{2\}\}$:

$$H(\mathbb{C}) = (19-20.67)^2+(25-25)^2+(20-20.67)^2+(23-20.67)^2 = 8.67$$

$\mathbb{C} = \{\{1,3\},\{2,4\}\}$:

$$H(\mathbb{C}) = (19-19.5)^2+(25-24)^2+(20-19.5)^2+(23-24)^2 = 2.5$$

Eigenschaften agglomerativer Clusterverfahren

- Single-Linkage tends to form chains (suitable for identifying outliers).
- Average-Linkage and Ward lead to very homogeneous clusters.
- Complete-Linkage is more sensitive to small changes in the data than Single-Linkage.
- Centroid and Ward are only applicable for metric features.
- Centroid and Ward can lead to Inversion (distance measure decreases compared to previous step)

Comparison of different Linkage-types