SCC5895 – Análise de Agrupamento de Dados

Algoritmos Hierárquicos (Parte I)

Prof. Eduardo Raul Hruschka

PPG-CCMC / ICMC / USP

Créditos

- O material a seguir consiste de adaptações e extensões dos originais:
  - Elaborados por Eduardo R. Hruschka e Ricardo J. G. B. Campello
  - de (Tan et al., 2006)
  - de E. Keogh (SBBD 2003)
- Algumas figuras foram gentilmente cedidas por Lucas Vendramin

Agenda

- Algoritmos Hierárquicos
  - Conceitos e Definições
    - Dendrogramas
    - Grafos de Proximidade
    - *Cophenetic matrices*
  - Métodos Aglomerativos
    - *Single Linkage*
    - *Complete Linkage*
  - Relação com Teoria dos Grafos

Relembrando...

- **Matriz de Dados X:**
  - N linhas (objetos) e n colunas (atributos):

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nn} \end{bmatrix}$$

- Cada **objeto** (linha da matriz) é denotado por um vetor $\mathbf{x}_i$
  - Exemplo:

$$\mathbf{x}_1 = \begin{bmatrix} x_{11} & \cdots & x_{1n} \end{bmatrix}$$

Relembrando...

- **Matriz de Proximidade (**Dissimilaridade ou Similaridade**):**
  - N linhas e N colunas:

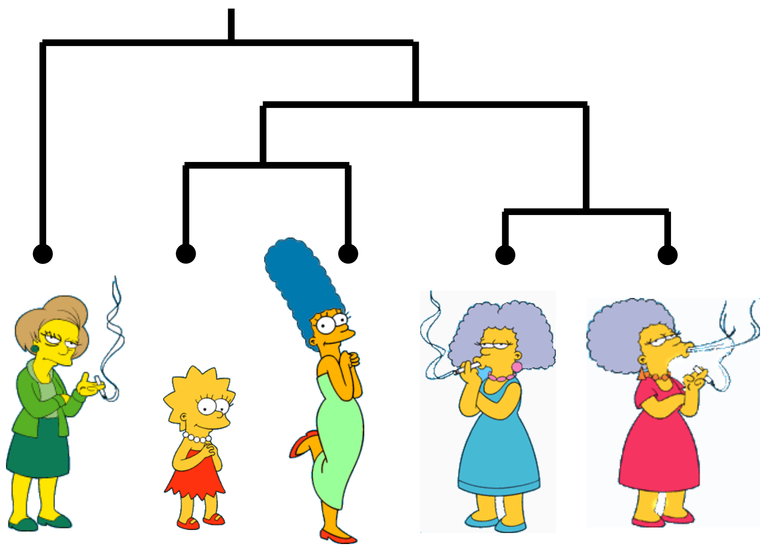
$$\mathbf{D} = \begin{bmatrix} d(\mathbf{x}_1, \mathbf{x}_1) & d(\mathbf{x}_1, \mathbf{x}_2) & \cdots & d(\mathbf{x}_1, \mathbf{x}_N) \\ d(\mathbf{x}_2, \mathbf{x}_1) & d(\mathbf{x}_2, \mathbf{x}_2) & \cdots & d(\mathbf{x}_2, \mathbf{x}_N) \\ \vdots & & \ddots & \vdots \\ d(\mathbf{x}_N, \mathbf{x}_1) & d(\mathbf{x}_N, \mathbf{x}_2) & \cdots & d(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}$$

- Simétrica se proximidade d apresentar propriedade de simetria

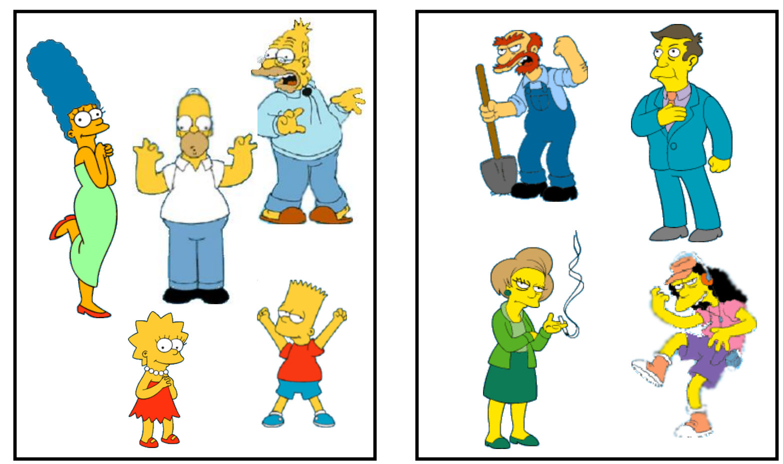
Relembrando...

- **Agrupamento Particional:** constrói uma *partição* dos dados
- **Agrupamento Hierárquico:** constrói uma *hierarquia de partições*

Hierárquicos



Particionais



Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Definição de Partição de Dados

- Consideremos um conjunto de N objetos a serem agrupados: $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$

- **Partição** (rígida): coleção de k grupos não sobrepostos $\mathbf{P} = \{\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_k\}$ tal que:

$$\boxed{\begin{gathered} \mathbf{C}_1 \cup \mathbf{C}_2 \cup \ldots \cup \mathbf{C}_k = \mathbf{X} \\ \mathbf{C}_i \neq \varnothing \\ \mathbf{C}_i \cap \mathbf{C}_j = \varnothing \quad \text{para} \quad i \neq j \end{gathered}}$$

  - Exemplo: $\mathbf{P} = \{\ (\mathbf{x}_1), (\mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_6), (\mathbf{x}_2, \mathbf{x}_5)\ \}$

Definição de Hierarquia

- **Hierarquia** (de partições de dados):
  - **Sequencia de partições aninhadas**
    - Uma partição $\mathbf{P}_1$ está *aninhada* em $\mathbf{P}_2$ se cada componente (grupo) de $\mathbf{P}_1$ é um subconjunto de um componente de $\mathbf{P}_2$
- **Exemplo:**

$$\mathbf{P}_1 = \{ (\mathbf{x}_1), (\mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_6), (\mathbf{x}_2, \mathbf{x}_5) \}$$

$$\mathbf{P}_2 = \{ (\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_6), (\mathbf{x}_2, \mathbf{x}_5) \}$$

- **Contra-Exemplo:**

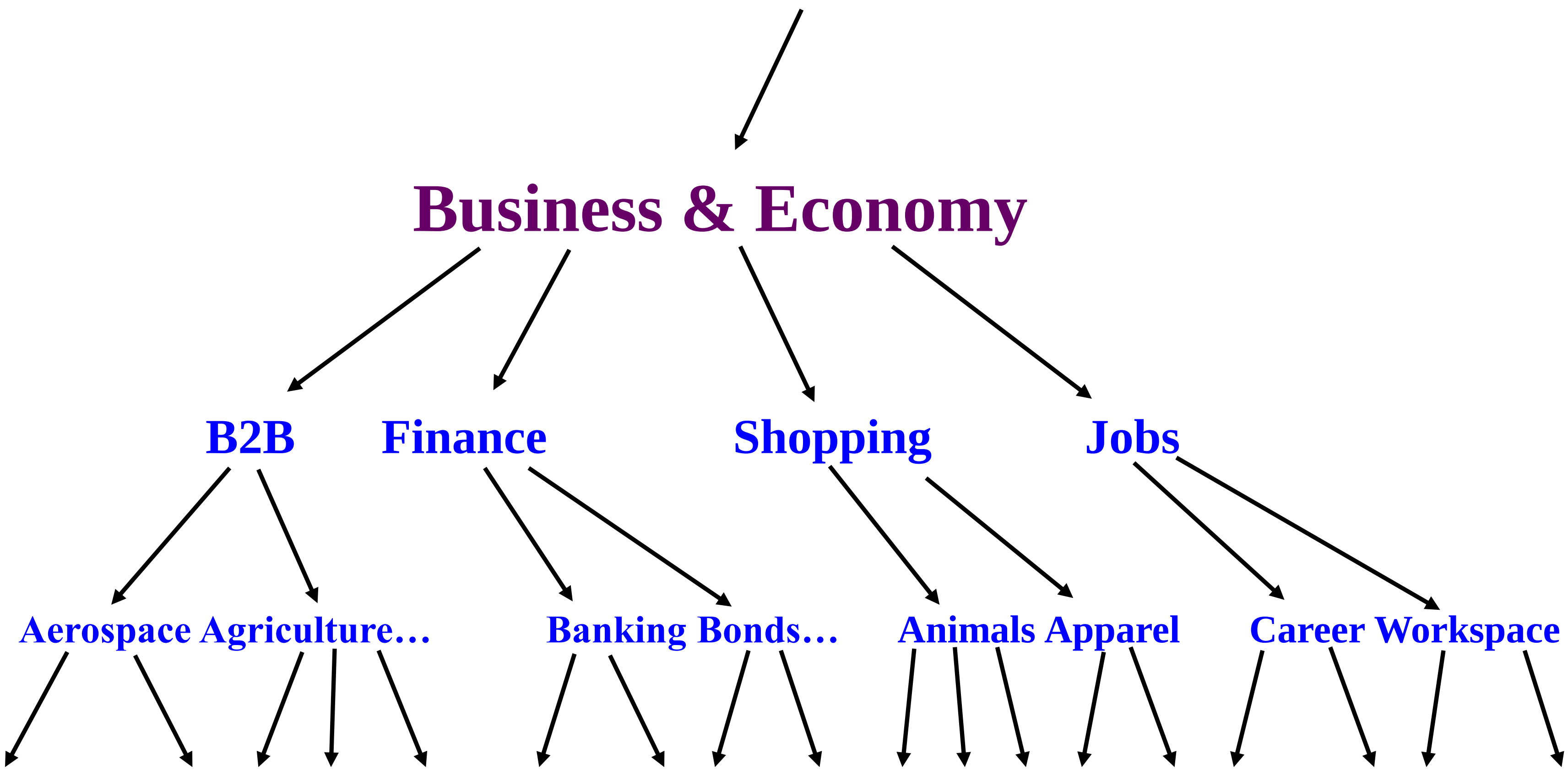
$$\mathbf{P}_3 = \{ (\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_6), (\mathbf{x}_2, \mathbf{x}_5) \}$$

$$\mathbf{P}_4 = \{ (\mathbf{x}_1, \mathbf{x}_2), (\mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_6), (\mathbf{x}_5) \}$$

Definição de Hierarquia

- Uma hierarquia completa:
  - Inicia ou termina com partição totalmente disjunta
    - ***Disjoint clustering***: apenas grupos **atômicos** (***singletons***)
    - Exemplo: $\mathbf{P} = \{ (\mathbf{x}_1), (\mathbf{x}_2), (\mathbf{x}_3), (\mathbf{x}_4), (\mathbf{x}_5), (\mathbf{x}_6) \}$
      - Também denominada "solução trivial"
  - Inicia ou termina com partição totalmente conjunta
    - ***Conjoint clustering***: grupo único com todos os objetos
    - Exemplo: $\mathbf{P} = \{ (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6) \}$
  - Geralmente possui $N - 2$ partições intermediárias

Hierarquias são comumente usadas para organizar informação, como, por exemplo, num portal



Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

- Outro Exemplo:
  - Árvores Filogenéticas em Biologia

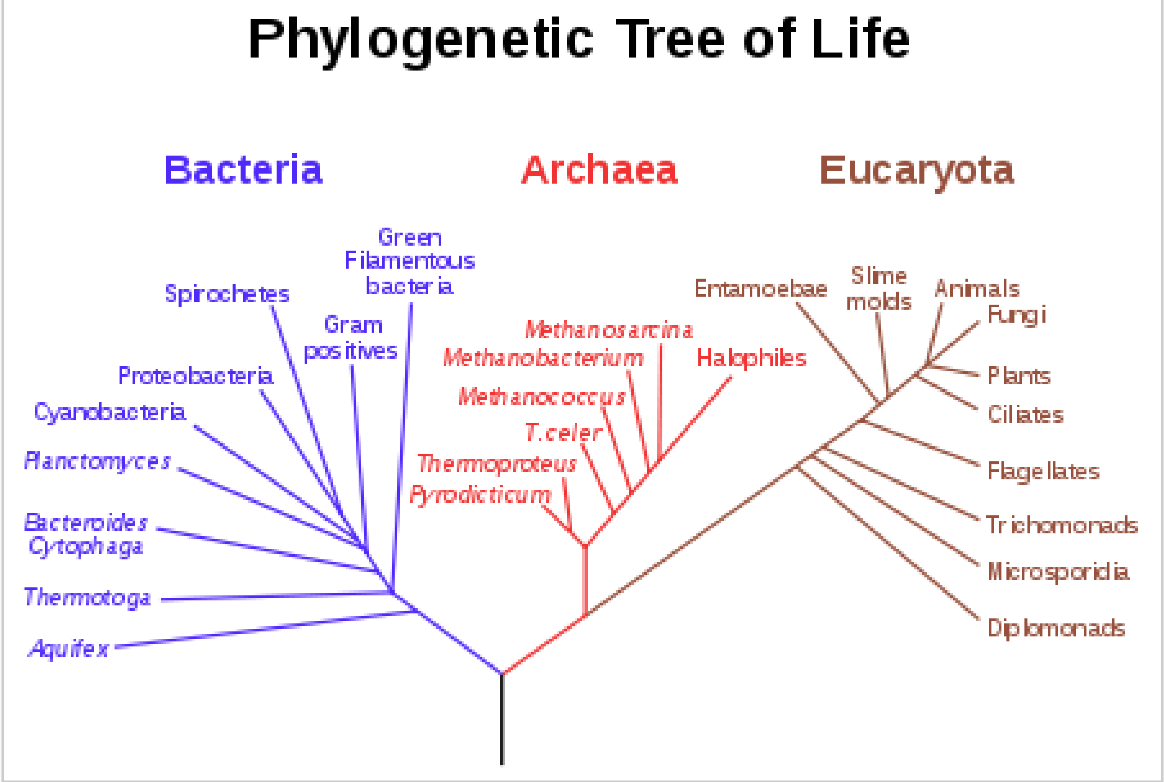
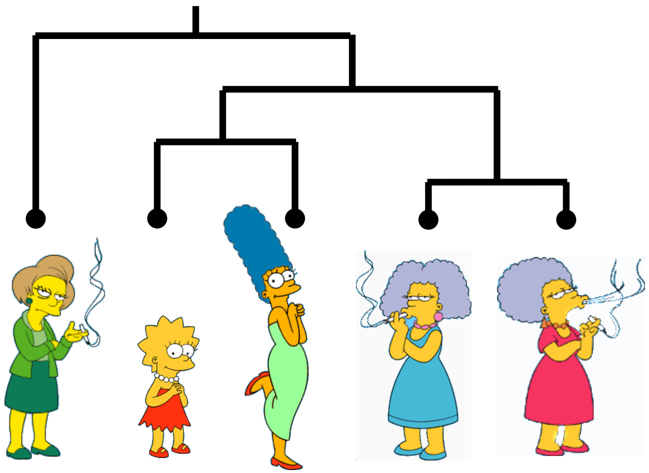


Fig. 1: A speculatively rooted tree for rRNA genes

http://en.wikipedia.org/wiki/Phylogenetic_tree

Métodos Clássicos para Agrupamento Hierárquico



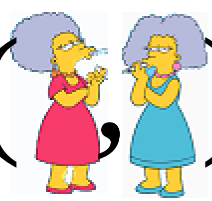
Bottom-Up (aglomerativos):
- Iniciar colocando cada objeto em um *cluster*
- Encontrar o melhor par de *clusters* para unir
- Unir o par de *clusters* escolhido
- Repetir até que todos os objetos estejam reunidos em um só *cluster*

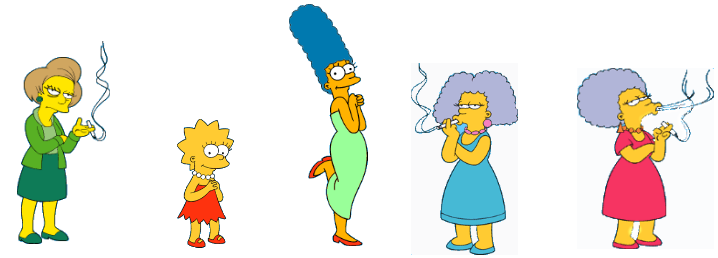
Top-Down (divisivos):
- Iniciar com todos objetos em um único *cluster*
- Sub-dividir o *cluster* em dois novos *clusters*
- Aplicar o algoritmo recursivamente em ambos, até que cada objeto forme um *cluster* por si só

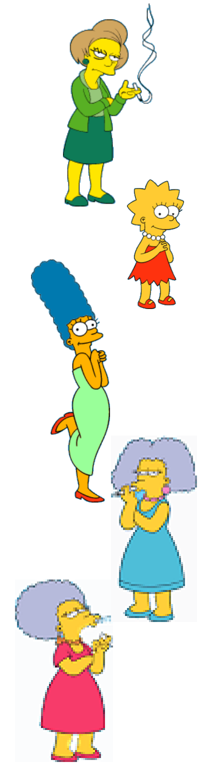
Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Algoritmos hierárquicos podem operar somente sobre uma matriz de distâncias: são (ou podem ser) **relacionais**.

D() = 8

D() = 1

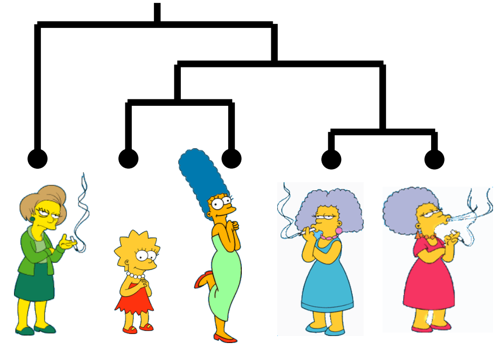




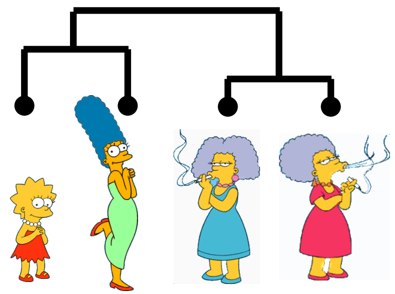
0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

Bottom-Up (aglomerativo):

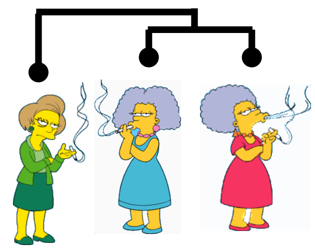
Iniciando com cada objeto em seu próprio cluster, encontrar o melhor par de *clusters* para unir em um novo *cluster*. Repetir até que todos os *clusters* sejam fundidos em um único *cluster*.



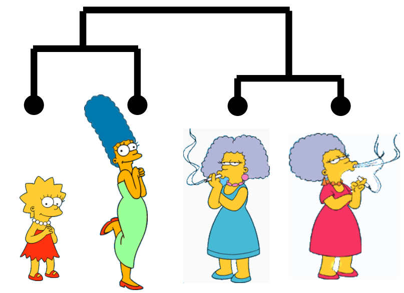
Considerar todas as uniões possíveis …



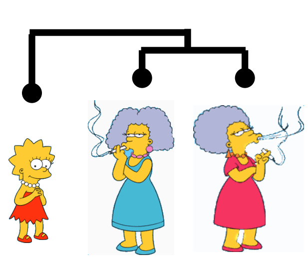
…

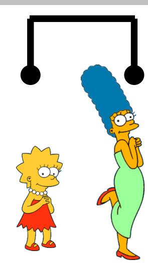


Escolher a melhor



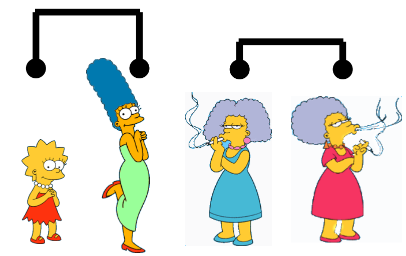
Considerar todas as uniões possíveis …



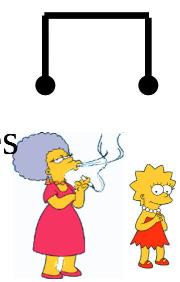


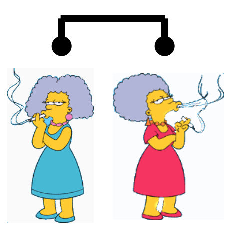
…

Escolher a melhor



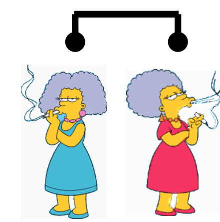
Considerar todas as uniões possíveis …



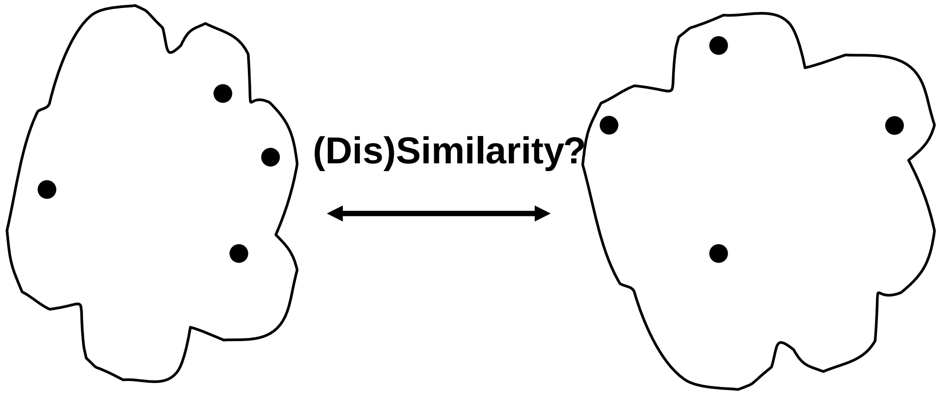


…

Escolher a melhor



How to Define Inter-Cluster (Dis)Similarity

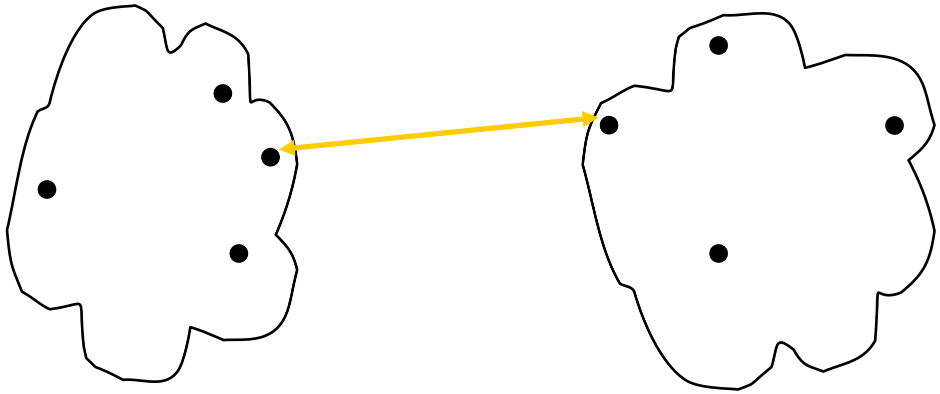


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods
  - Ward's
  - …

How to Define Inter-Cluster (Dis)Similarity

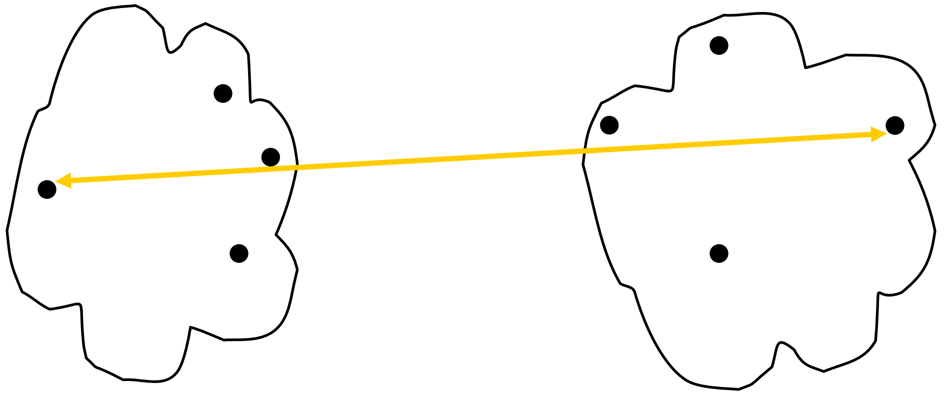


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods
  - Ward's
  - …

How to Define Inter-Cluster (Dis)Similarity

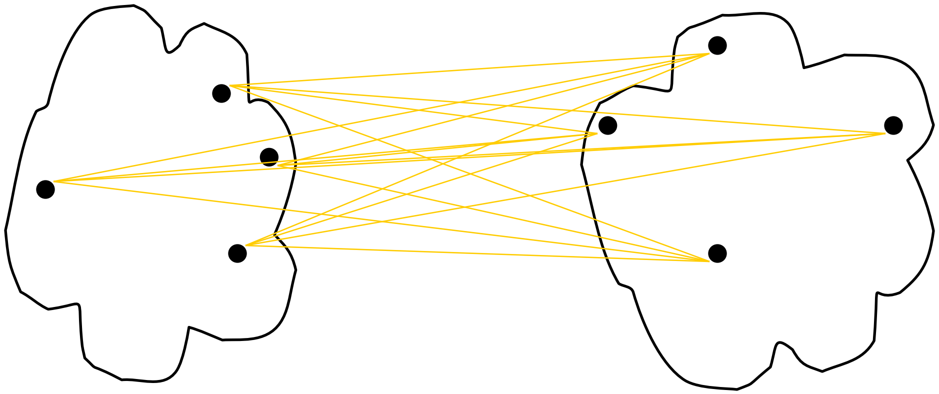


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods
  - Ward's
  - …

How to Define Inter-Cluster (Dis)Similarity

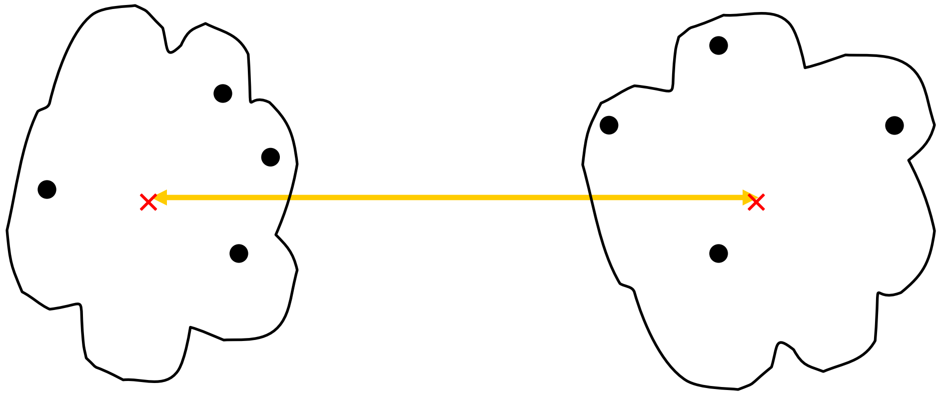


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods
  - Ward's
  - …

How to Define Inter-Cluster (Dis)Similarity



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods
  - Ward's
  - . . .

Como Comparar os Clusters?

- ***Single Linkage*** , Min, ou Vizinho mais Próximo :
  - Dissimilaridade entre *clusters* é dada pela menor dissimilaridade entre 2 objetos (um de cada *cluster*)

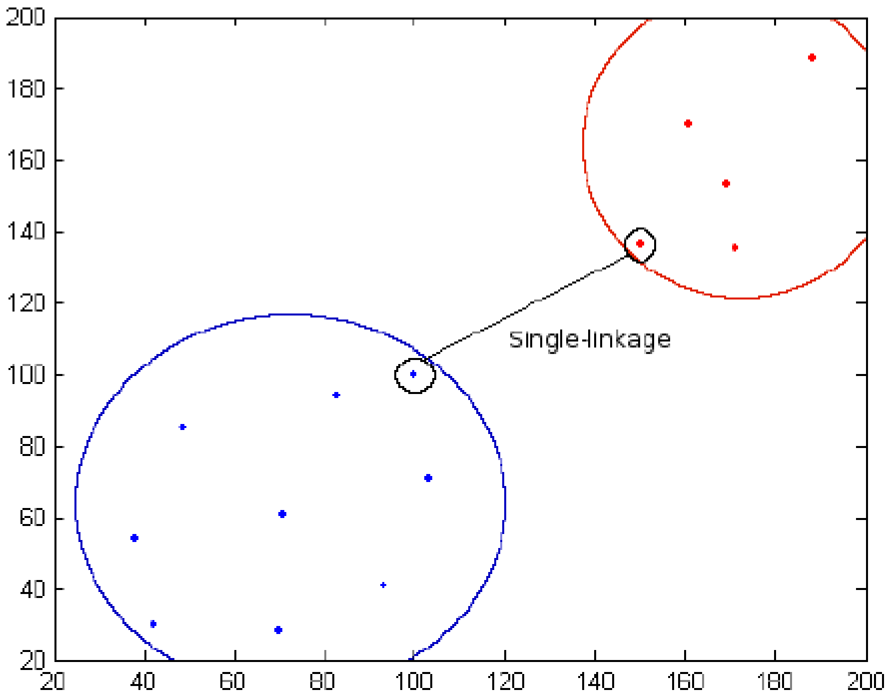


Figura por Lucas Vendramin

single link (Florek, 1951; Sneath, 1957)

<u>Originalmente baseado em **Grafos**</u>:
menor aresta entre dois vértices
de subconjuntos distintos

Propriedade Útil

- Propriedade da Função Mínimo (min):
  - $\min\{\mathbf{D}\} = \min\{ \min\{\mathbf{D}_1\} , \min\{\mathbf{D}_2\} \}$
    - $\mathbf{D}$, $\mathbf{D}_1$ e $\mathbf{D}_2$ são conjuntos de valores reais tais que $\mathbf{D}_1 \cup \mathbf{D}_2 = \mathbf{D}$
  - Exemplo:
    - $\min\{10, -3, 0, 100\} = \min \{ \min\{10, -3\}, \min\{0, 100\} \} = -3$
  - Propriedade vale recursivamente (para $\min\{\mathbf{D}_1\}$ e $\min\{\mathbf{D}_2\}$)
- Utilidade para Single-Linkage
  - Dada a distância entre os grupos **A** e **B** e entre **A** e **C**
    - É trivial calcular a distância entre **A** e (**B** $\cup$ **C**) !

Exemplo de Single Linkage: Método de Johnson (1967)

- Consideremos a seguinte matriz de distâncias iniciais ($\mathbf{D}_1$) entre 5 objetos {1,2,3,4,5}. Qual par de objetos será escolhido para formar o 1º *cluster* ?

$$\mathbf{D}_1 = \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} 0 & & & & \\ \boxed{2} & 0 & & & \\ 6 & 5 & 0 & & \\ 10 & 9 & 4 & 0 & \\ 9 & 8 & 5 & 3 & 0 \end{bmatrix}$$

- A menor distância entre objetos é $d_{12}=d_{21}=2$, indicando que estes dois objetos serão unidos em um *cluster.* Na seqüência, calcula-se:

$d_{(12)3}=min\{d_{13},d_{23}\}=d_{23}=5;$

$d_{(12)4}=min\{d_{14},d_{24}\}=d_{24}=9;$

$d_{(12)5}=min\{d_{15},d_{25}\}=d_{25}=8;$

- Desta forma, obtém-se uma nova matriz de distâncias ($\mathbf{D}_2$), que será usada na próxima etapa do agrupamento hierárquico:

Exemplo de (Everitt et al., 2001)

$$\mathbf{D}_2 = \begin{array}{c} 12 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} 0 & & & \\ 5 & 0 & & \\ 9 & 4 & 0 & \\ 8 & 5 & \boxed{3} & 0 \end{bmatrix}$$

- Qual o novo *cluster* a ser formado?
- Unindo os objetos **4** e **5** obtemos três clusters: {1,2}, {4,5}, {3}
- Como $d_{(12)3}$ já está calculada, calculamos na sequência:

$d_{(12)(45)} = min\{d_{(12)(4)}, d_{(12)(5)}\} = d_{(12)(5)} = 8$

$d_{(45)3} = min\{d_{43}, d_{53}\} = d_{43} = 4$

obtendo a seguinte matriz:

$$\mathbf{D}_3 = \begin{array}{c} 12 \\ 3 \\ 45 \end{array} \begin{bmatrix} 0 & & \\ 5 & 0 & \\ 8 & \boxed{4} & 0 \end{bmatrix}$$

* Unir *cluster* {3} com {4,5};

* Finalmente, unir todos os *clusters* em um único *cluster*

- A sequência de partições obtidas neste exemplo é, portanto:

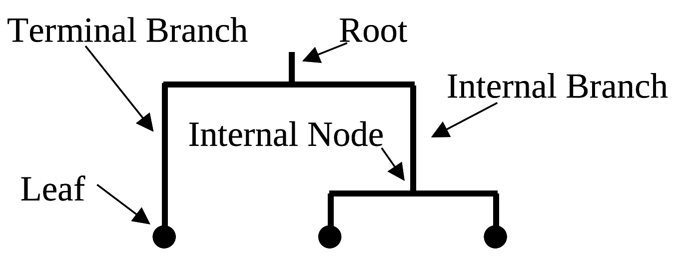
$\{ (1), (2), (3), (4), (5) \} \rightarrow \{ (1, 2), (3), (4), (5) \} \rightarrow$

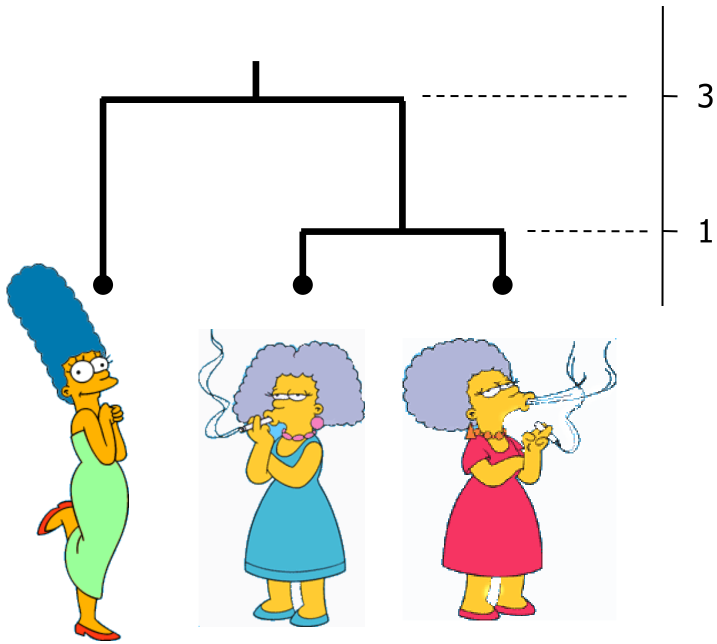
$\{ (1, 2), (3), (4, 5) \} \rightarrow \{ (1, 2), (3, 4, 5) \} \rightarrow \{ (1, 2, 3, 4, 5) \}$

- **Nota**: Para *single link*, a dissimilaridade entre 2 clusters pode ser computada naturalmente a partir da matriz atualizada na iteração anterior, sem necessidade da matriz original
  - Isso vale devido à propriedade da função *min* vista anteriormente
  - No nosso exemplo, simplificamos o cálculo de $d_{(12)(45)}$ como $\min\{d_{(12)(4)}, d_{(12)(5)}\}$ fazendo uso daquela propriedade:
    - $\min\{d_{(12)(4)}, d_{(12)(5)}\} = \min\{9, 8\} = \min\{d_{14}, d_{24}, d_{15}, d_{25}\}$

Dendrograma

Dendrograma: Hierarquia + Dissimilaridades entre Clusters





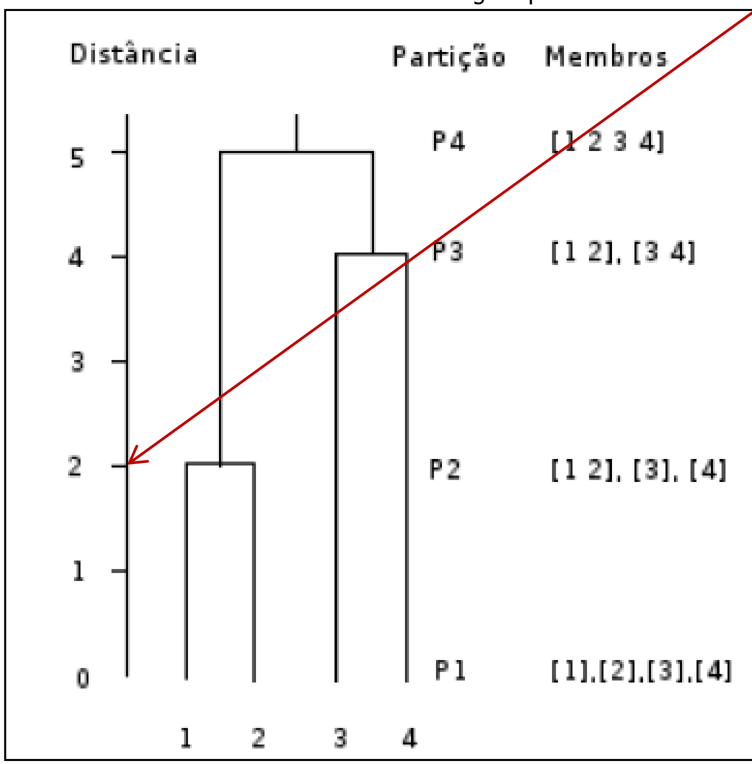
* A dissimilaridade entre dois clusters (possivelmente **singletons**) é representada como a altura do nó interno mais baixo compartilhado

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

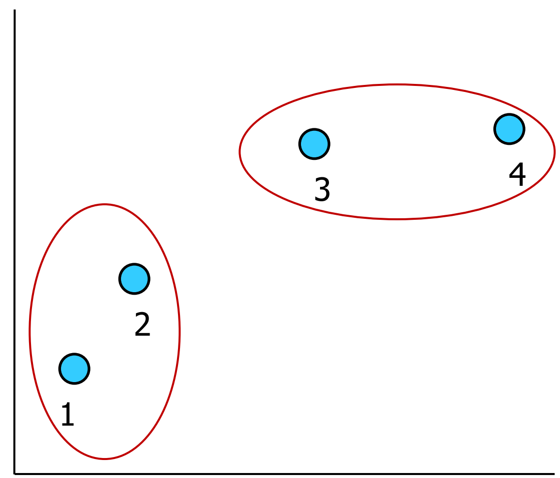
Exemplo de Dendrograma

$$\mathbf{D} = \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{bmatrix} 0 & 2 & 7 & 13 \\ 2 & 0 & 5 & 10 \\ 7 & 5 & 0 & 4 \\ 13 & 10 & 4 & 0 \end{bmatrix}$$

Figura por Lucas Vendramin

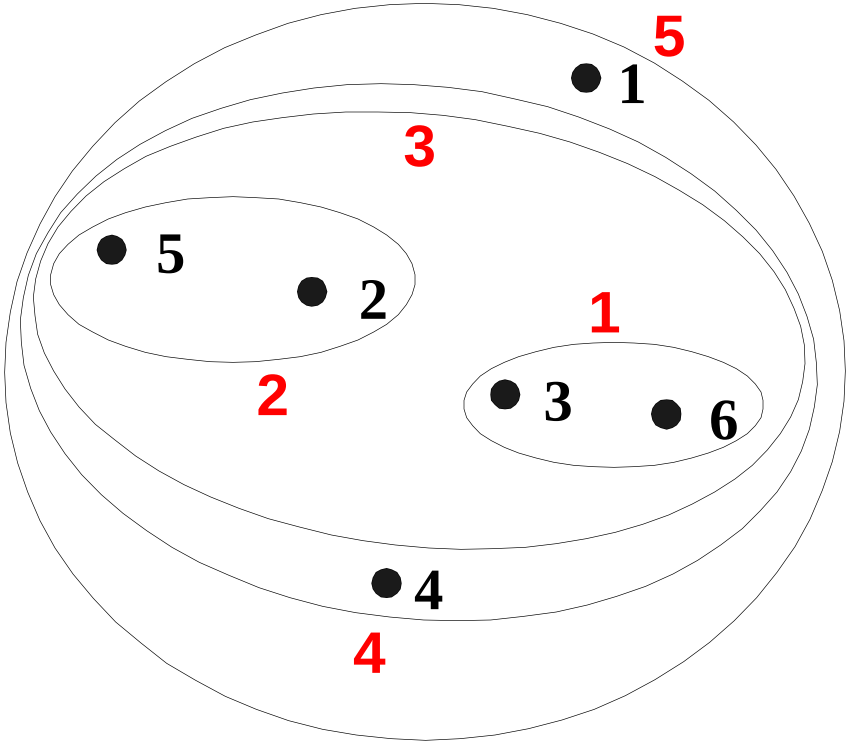


Dendrograma

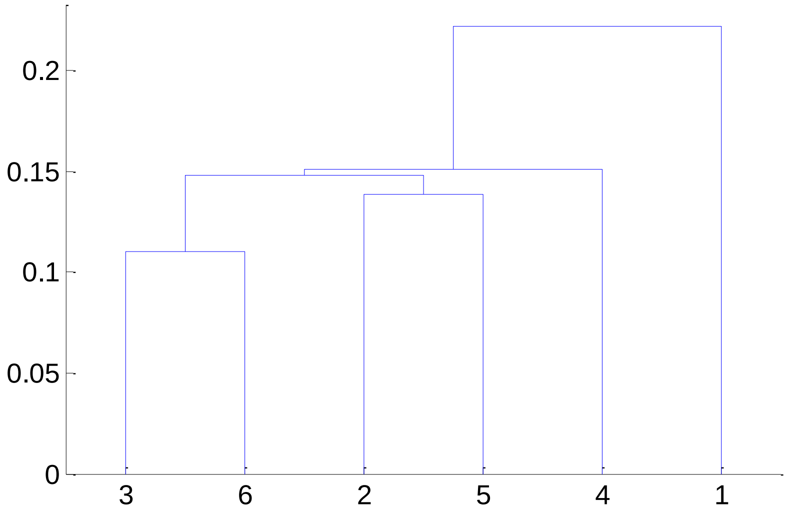


uma das partições aninhadas

Outro Exemplo de Dendrograma



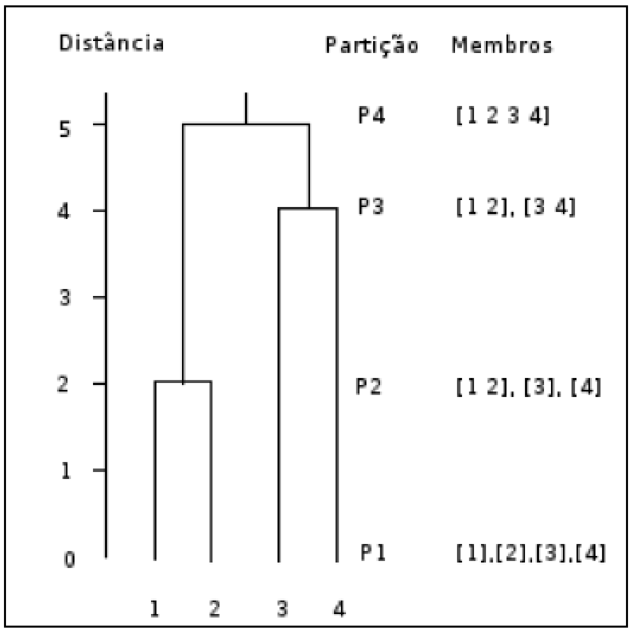
Nested Clusters



Dendrogram

Cophenetic Matrix

- Matriz com as dissimilaridades que levaram à união de cada par de objetos na base de dados. Exemplo:



$$\mathbf{Cp} = \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{bmatrix} 0 & 2 & 5 & 5 \\ 2 & 0 & 5 & 5 \\ 5 & 5 & 0 & 4 \\ 5 & 5 & 4 & 0 \end{bmatrix}$$

- Esta matriz é importante para a validação de agrupamentos hierárquicos (tópico a ser discutido posteriormente no curso)

Exercício:

- Obtenha o dendrograma completo para o exemplo visto de execução do *single linkage* (matriz de distâncias abaixo)

$$\mathbf{D}_1 = \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} 0 & & & & \\ 2 & 0 & & & \\ 6 & 5 & 0 & & \\ 10 & 9 & 4 & 0 & \\ 9 & 8 & 5 & 3 & 0 \end{bmatrix}$$

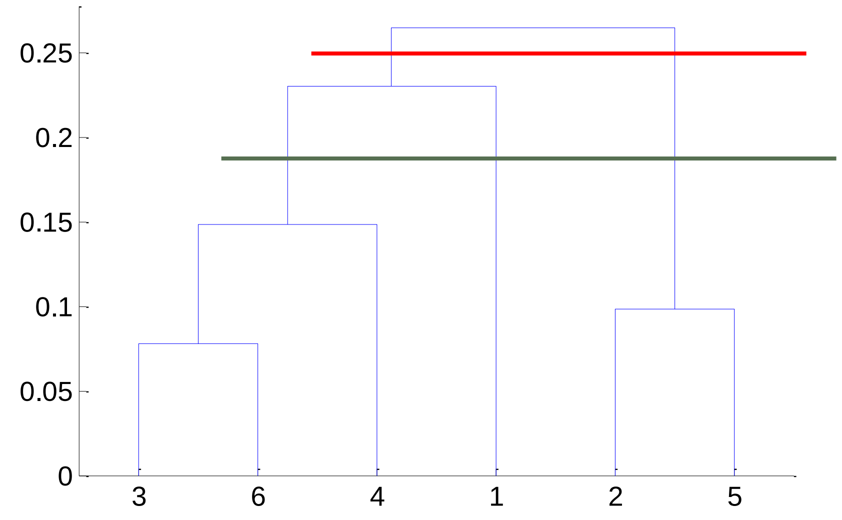
- Apresente também a *cophenetic matrix* correspondente

Dendrogramas e Partições

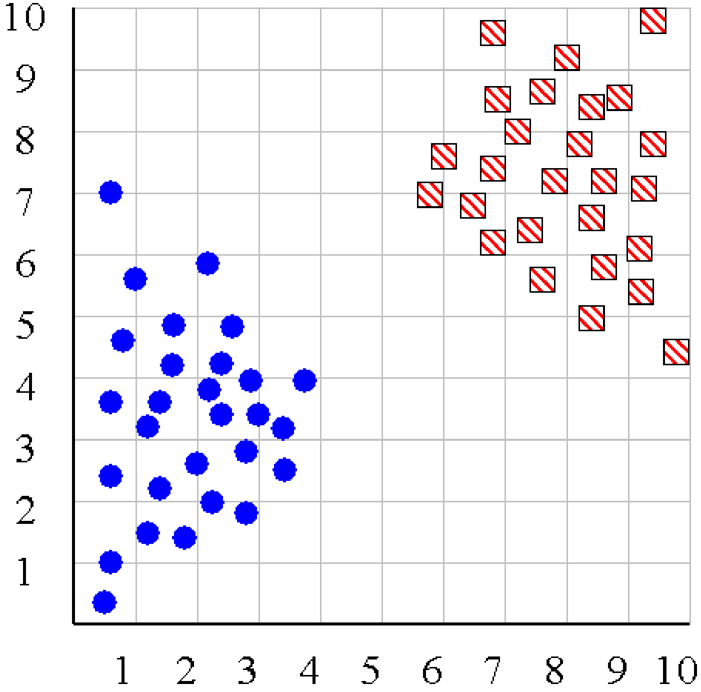
- Partições são obtidas via **cortes** no dendrograma
  - cortes horizontais
  - no. de grupos da partição = no. de interseções
- **Exemplos:**

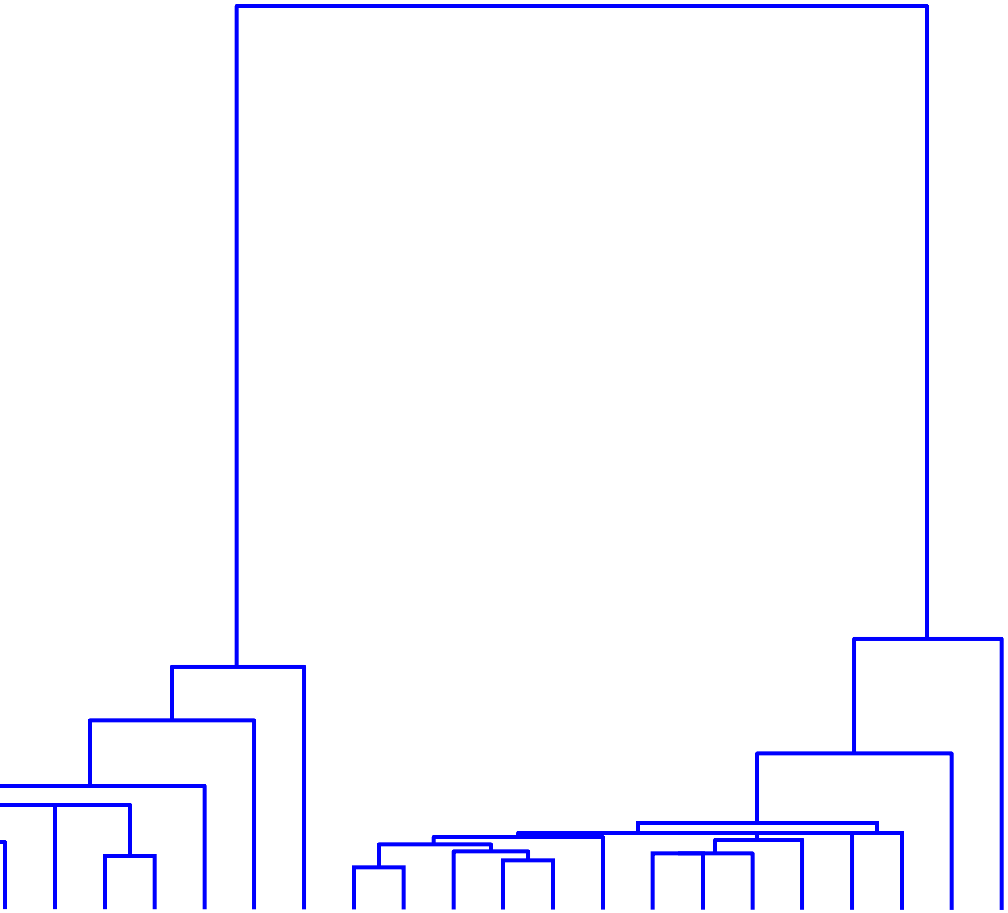
$P_2 = \{ (\mathbf{x}_1,\mathbf{x}_3,\mathbf{x}_4,\mathbf{x}_6), (\mathbf{x}_2,\mathbf{x}_5) \}$

$P_1 = \{ (\mathbf{x}_1), (\mathbf{x}_3,\mathbf{x}_4,\mathbf{x}_6), (\mathbf{x}_2,\mathbf{x}_5) \}$



Pode-se examinar o dendrograma para tentar estimar o número *mais natural* de clusters. No caso abaixo, existem duas sub-árvores bem separadas, sugerindo dois grupos de dados. Infelizmente, na prática, as distinções não são tão simples…

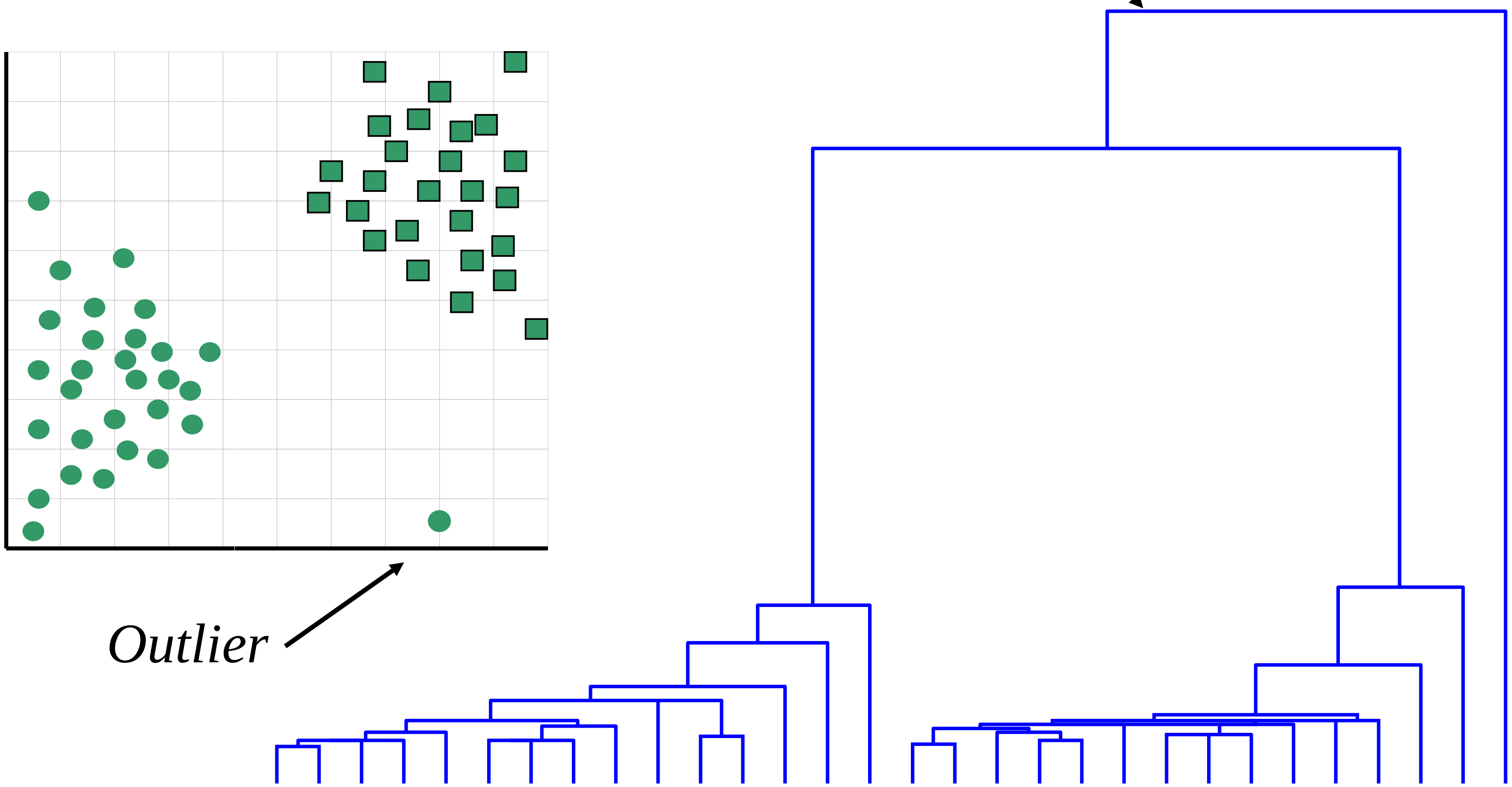




Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Pode-se usar o dendrograma para tentar detectar *outliers*:

Ramo isolado sugere que o objeto é muito diferente dos demais.



Outlier

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

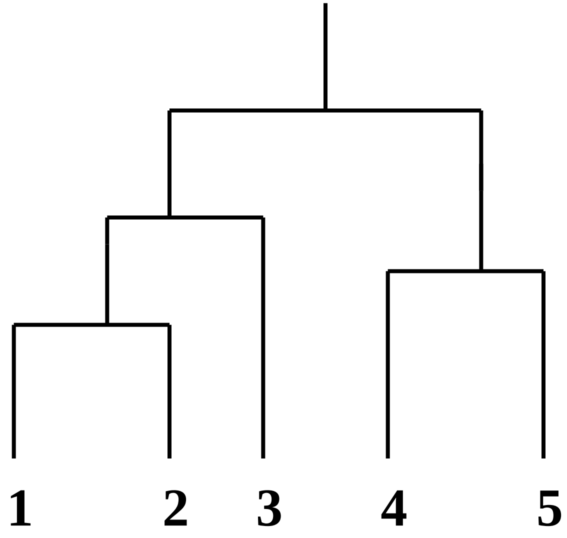
Como Plotar o Dendrograma ?

- Para muitos objetos, ilustração manual é inviável
- Gerar o gráfico automaticamente demanda ordenar de forma apropriada os objetos no eixo horizontal
- Algoritmo Recursivo Simples:
  - Iniciar com o topo da hierarquia (grupo único)
  - Dividir o eixo horizontal em 2 subintervalos e colocar em cada um os objetos de cada um dos 2 grupos que derivam do grupo único
  - Executar recursivamente o passo anterior para cada subintervalo

Voltando ao Single Linkage (Min)...

- Similarity of two clusters is based on the two most similar (closest) points in the clusters
  - Determined by **one pair of points**
  - i.e., by **one link** in the **proximity graph**

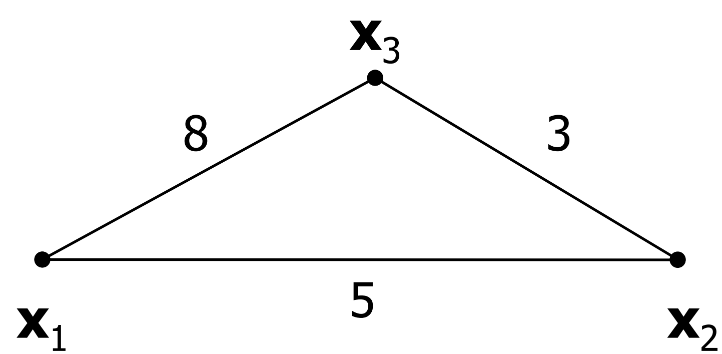
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



Grafo de Proximidades

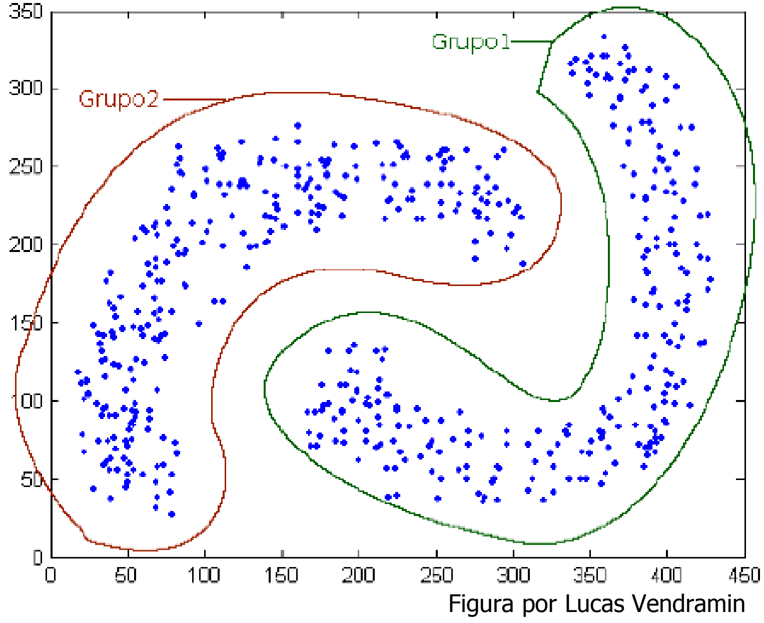
- Grafo (**ponderado**, sem laços e sem múltiplas arestas) no qual:
  - vértices representam os objetos da base de dados
  - arestas representam as (dis)similaridades entre pares de objetos
- Exemplo (3 objetos):

$$\mathbf{D} = \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{bmatrix} 0 & 5 & 8 \\ 5 & 0 & 3 \\ 8 & 3 & 0 \end{bmatrix} \Rightarrow$$



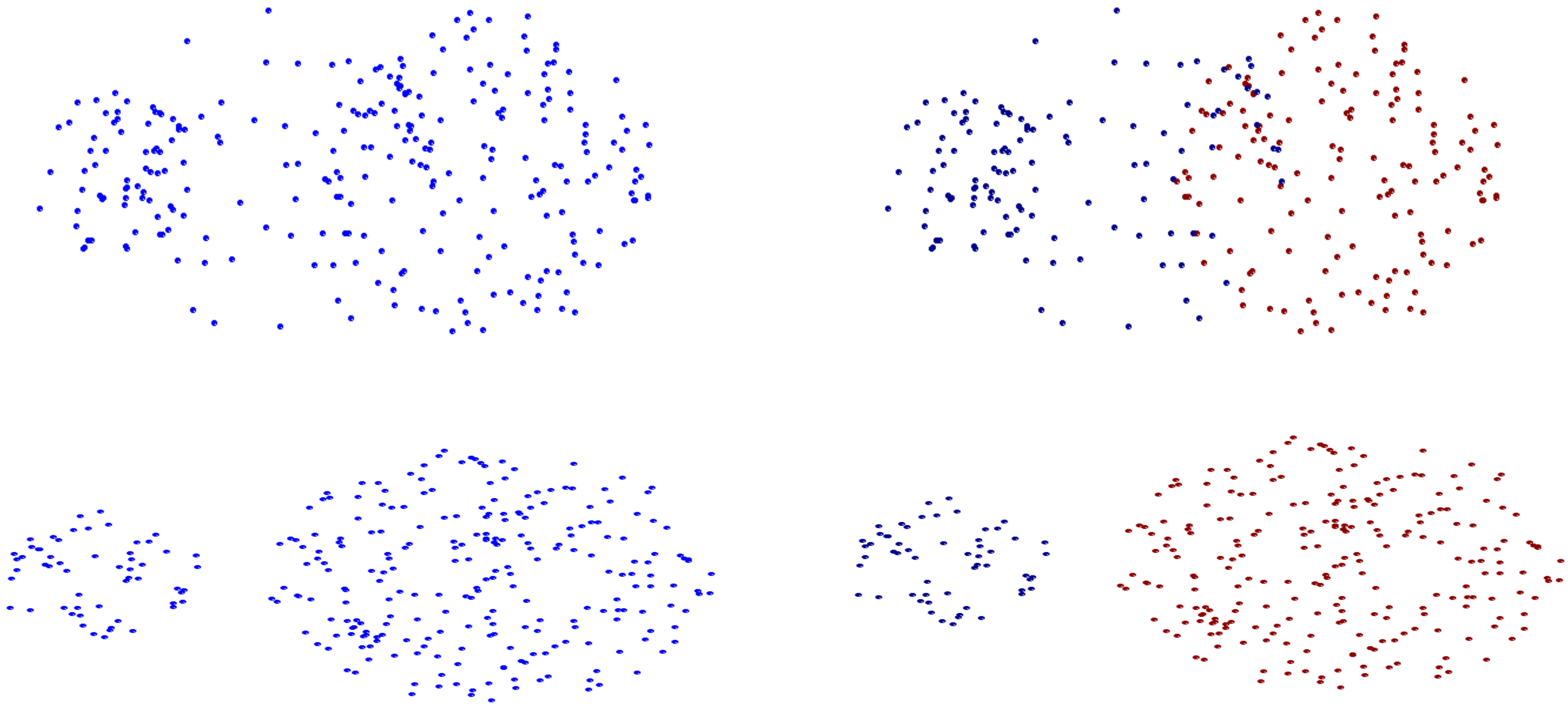
Strength of MIN

- **Can handle non-elliptical shapes**



Main Limitations of MIN

- **Sensitive to noise and outliers**



Original Points **Two Clusters**

Como Comparar os Clusters?

- ***Complete Linkage*** , Max, ou Vizinho mais Distante:
  - Dissimilaridade entre *clusters* é dada pela maior dissimilaridade entre dois objetos (um de cada *cluster*)

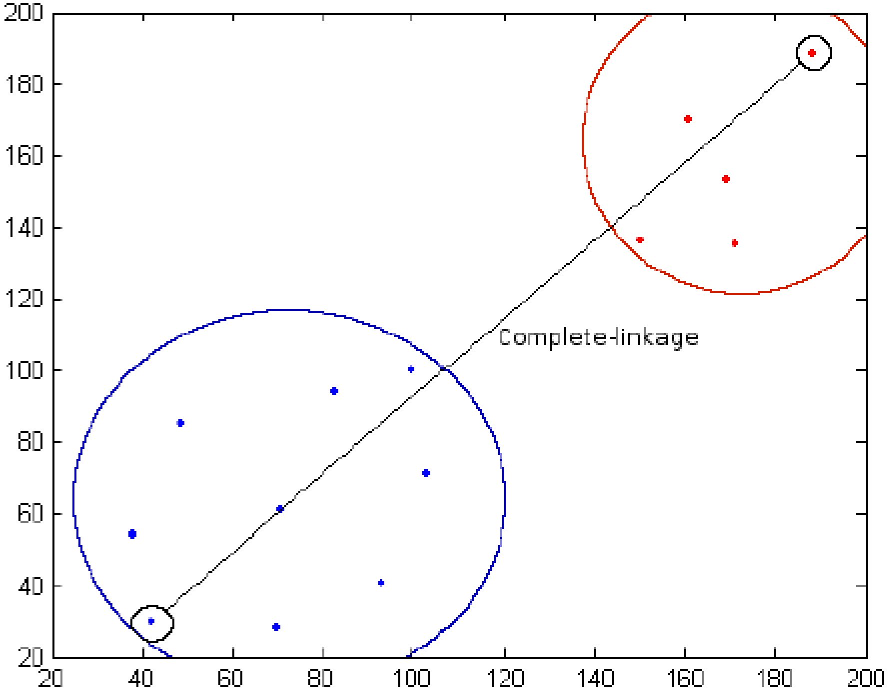


Figura por Lucas Vendramin

complete link (Sorensen, 1948)

<u>Originalmente baseado em **Grafos**</u>: maior aresta entre dois vértices de subconjuntos distintos

Propriedade Útil

- Propriedade da Função Máximo (max):
  - $\max\{\mathbf{D}\} = \max\{ \max\{\mathbf{D}_1\} , \max\{\mathbf{D}_2\} \}$
    - $\mathbf{D}$, $\mathbf{D}_1$ e $\mathbf{D}_2$ são conjuntos de valores reais tais que $\mathbf{D}_1 \cup \mathbf{D}_2 = \mathbf{D}$
  - Exemplo:
    - $\max\{10, -3, 0, 100\} = \max \{ \max\{10, -3\}, \max\{0, 100\} \} = 100$
  - Propriedade vale recursivamente (para $\max\{\mathbf{D}_1\}$ e $\max\{\mathbf{D}_2\}$)
- Utilidade para *Complete-Linkage*
  - Dada a distância entre os grupos **A** e **B** e entre **A** e **C**
    - É trivial calcular a distância entre **A** e (**B** ∪ **C**) !

- Seja a seguinte matriz de distâncias iniciais ($\mathbf{D}_1$) entre 5 objetos :

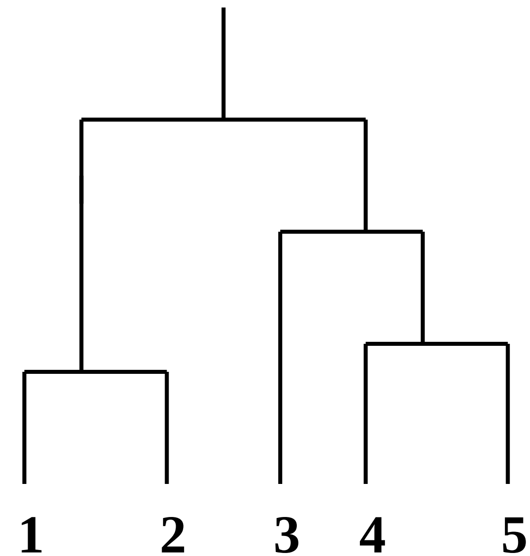
$$\mathbf{D}_1 = \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} 0 & & & & \\ 2 & 0 & & & \\ 6 & 5 & 0 & & \\ 10 & 9 & 4 & 0 & \\ 9 & 8 & 5 & 3 & 0 \end{bmatrix}$$

- Exercício: executar o *complete linkage* através de sucessivas atualizações da matriz de distâncias (método de Johnson).

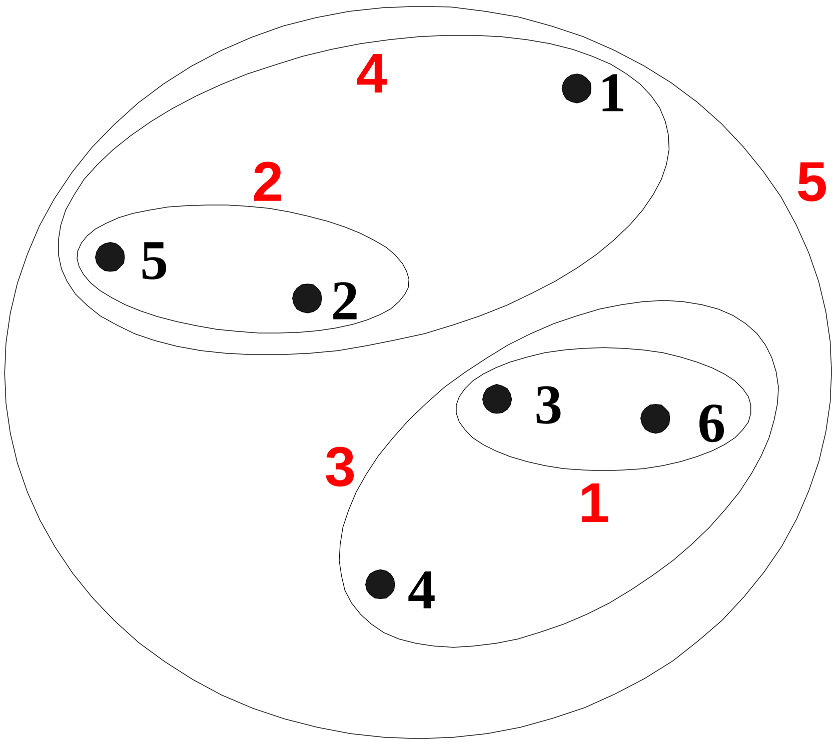
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on the two least similar (most distant) points in the clusters
  - Determined by **one pair of points**

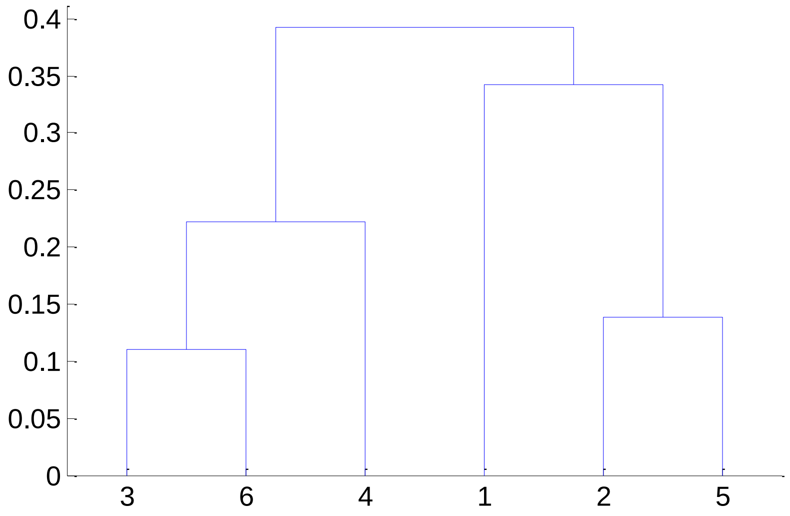
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



Hierarchical Clustering: MAX

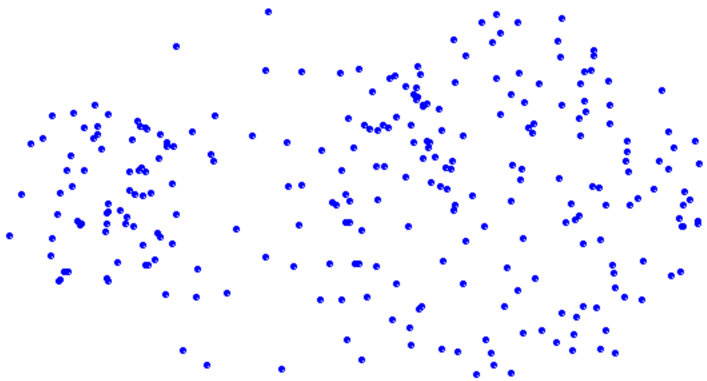


Nested Clusters

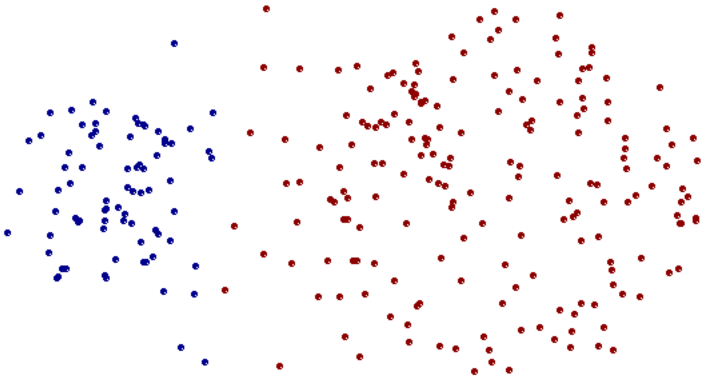


Dendrogram

Strength of MAX



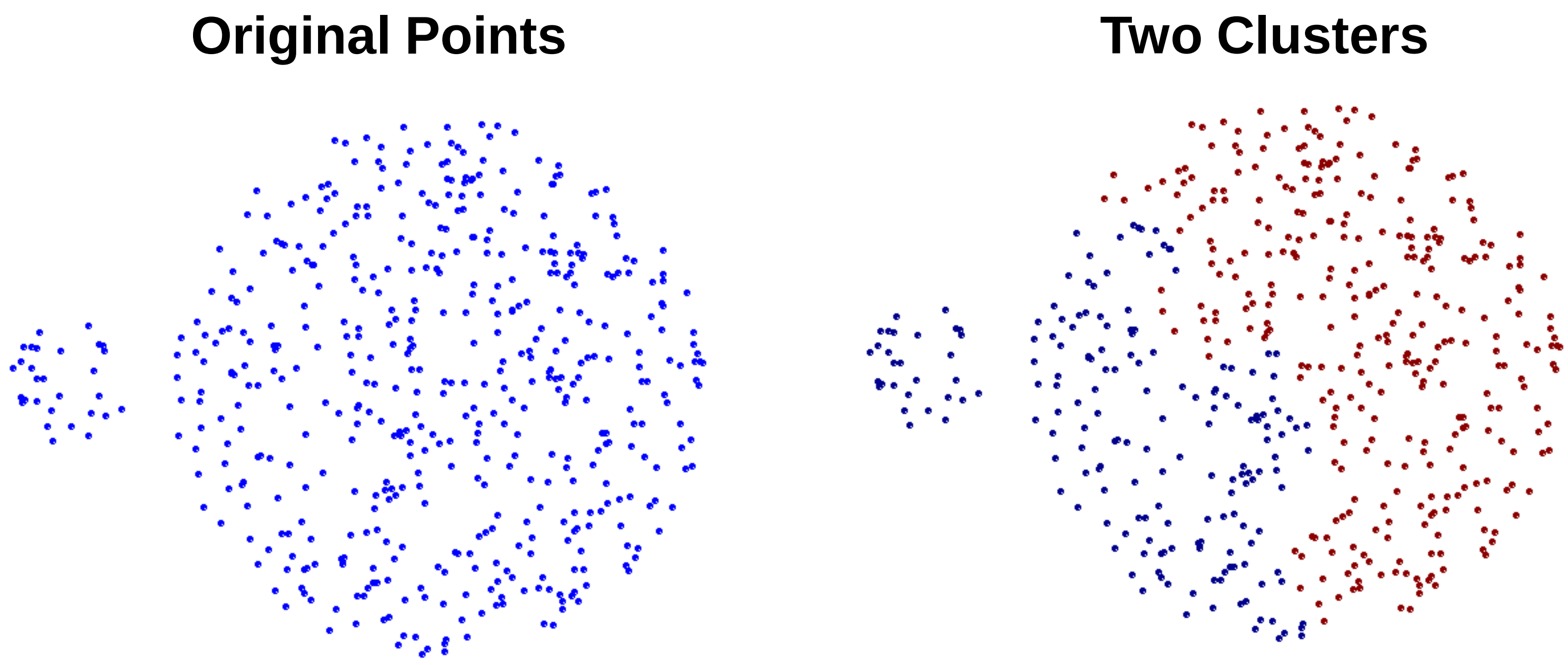
Original Points



Two Clusters

- **Less susceptible to noise and outliers**

Main Limitations of MAX



- Tends to break large clusters
- Biased towards globular clusters

Vinculação Simples/Completa sob Perspectiva de Teoria dos Grafos

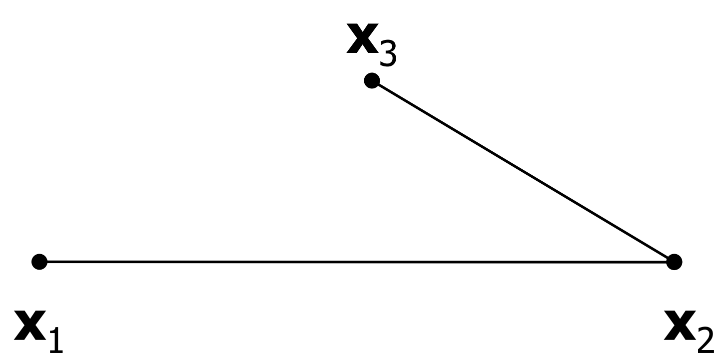
- Por simplicidade, assumamos uma matriz de distâncias em escala ordinal (sem empates)
- Exemplo:

$$\mathbf{D} = \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{array} \begin{bmatrix} 0 & 6 & 8 & 2 & 7 \\ 6 & 0 & 1 & 5 & 3 \\ 8 & 1 & 0 & 10 & 9 \\ 2 & 5 & 10 & 0 & 4 \\ 7 & 3 & 9 & 4 & 0 \end{bmatrix}$$

Grafo de Limiar $G(v)$

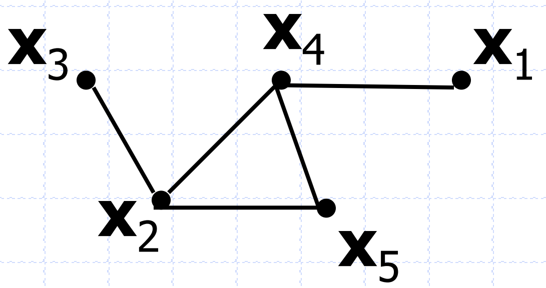
- Grafo (**não ponderado**, sem laços e sem múltiplas arestas) no qual:
  - vértices de $G(v)$ representam os objetos da base de dados
  - arestas (links) de $G(v)$ conectam pares de objetos que possuem dissimilaridade menor ou igual a um limiar v
    - ou similaridade maior ou igual a um limiar v
- Exemplo Simples: $G(5)$

$$\mathbf{D} = \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{bmatrix} 0 & 5 & 8 \\ 5 & 0 & 3 \\ 8 & 3 & 0 \end{bmatrix}$$



- $G(v)$ define uma **relação binária** para qualquer nº real v
  - subconjunto do Produto Cartesiano $\mathbf{X} \times \mathbf{X}$, onde $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$
- Exemplo para $v = 5$:

$$\mathbf{D} = \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{array} \begin{bmatrix} 0 & 6 & 8 & 2 & 7 \\ 6 & 0 & 1 & 5 & 3 \\ 8 & 1 & 0 & 10 & 9 \\ 2 & 5 & 10 & 0 & 4 \\ 7 & 3 & 9 & 4 & 0 \end{bmatrix}$$



$$\mathbf{R} = \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{array} \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Algoritmo Vinculação Simples (*single link*):

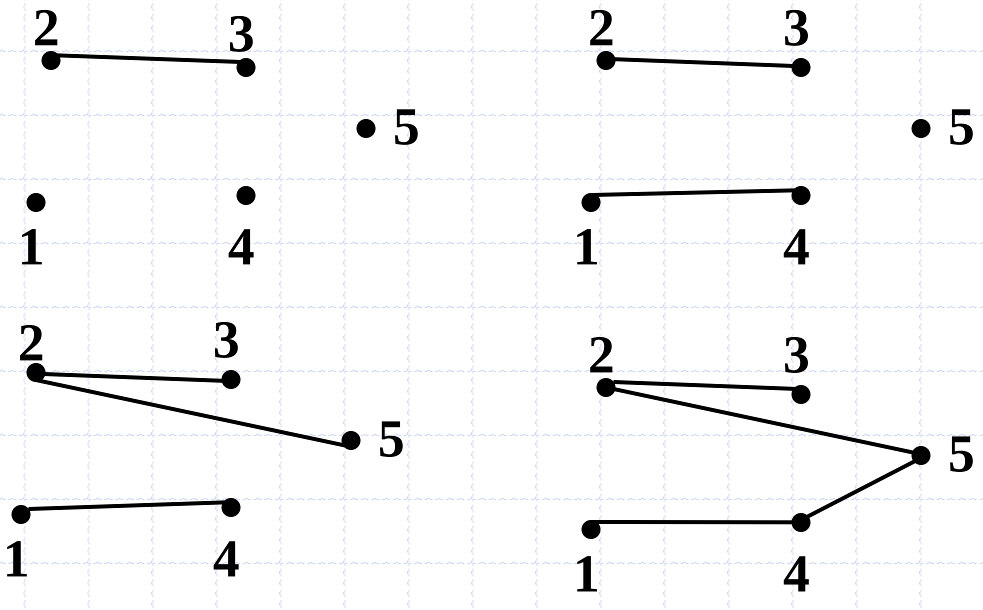
1) Iniciar com $G(0)$: cada objeto formando um grupo. $k \leftarrow 1$

2) Formar o grafo de limiar $G(k)$. Se o nº de componentes conexos em $G(k)$ for menor do que o nº de grupos corrente, re-nomear cada um dos componentes como sendo um grupo

3) Se $G(k)$ formar um único componente conexo, parar. Senão, fazer $k \leftarrow k + 1$ e voltar ao passo 2

Exemplo:

$$\mathbf{D} = \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{array} \begin{bmatrix} 0 & 6 & 8 & 2 & 7 \\ 6 & 0 & 1 & 5 & 3 \\ 8 & 1 & 0 & 10 & 9 \\ 2 & 5 & 10 & 0 & 4 \\ 7 & 3 & 9 & 4 & 0 \end{bmatrix}$$



Algoritmo Vinculação Completa:

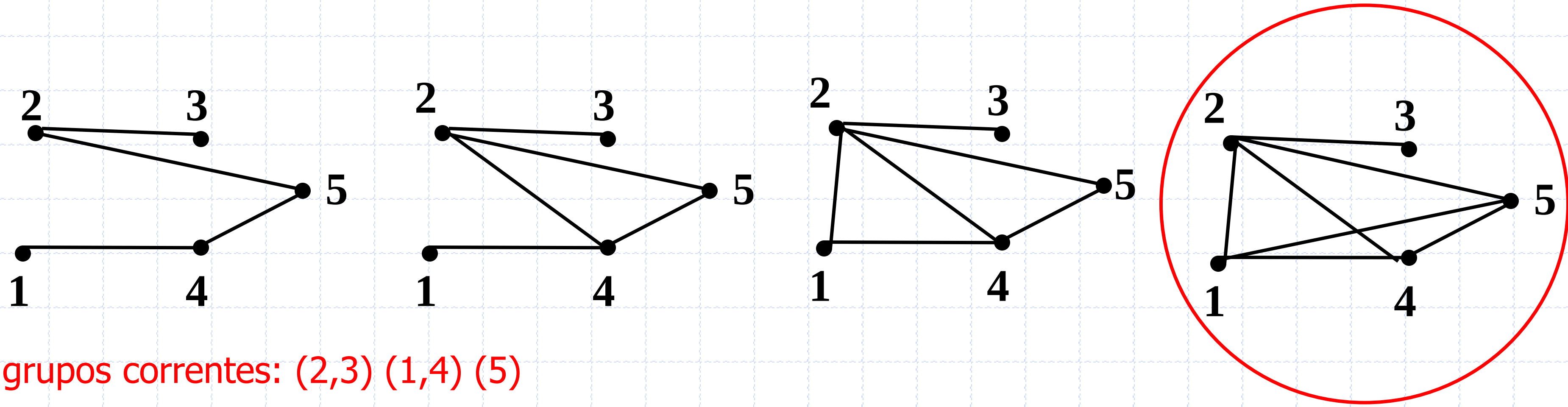
1) Iniciar com $G(0)$: cada objeto formando um grupo. $k \leftarrow 1$

2) Formar o grafo de limiar $G(k)$. Se 2 dos grupos correntes formam um *clique* (subgrafo completo) em $G(k)$, unir tais grupos

3) Se $k = N(N-1)/2$, o que implica que $G(k)$ é um grafo completo, parar. Caso contrário, $k \leftarrow k + 1$ e voltar ao passo 2

Exemplo (continuação):

clique de grupos correntes $\rightarrow$ (1, 4, 5)



grupos correntes: (2,3) (1,4) (5)

* Dois grupos: pode-se interromper o processo...

Hierarquias Resultantes:

Vinculação **simples**:

{(2,3),1,4,5}
{(2,3),(1,4),5}
{(2,3,5),(1,4)}

Vinculação **completa**:

{(2,3),1,4,5}
{(2,3),(1,4),5}
{(2,3),(1,4,5)}

Notem que $\mathbf{x}_5$ está em grupos diferentes

- Validação (a ser estudada posteriormente no curso) poderá nos auxiliar nesses casos

Observações

- **Matrizes não ordinais**:
  - ordena-se os valores de dissimilaridade e faz-se k assumir esses valores (ao invés de inteiros)
  - inicia em $k = 0$ e depois assume os valores de dissimilaridade do menor para o maior
  - os valores de k para os quais ocorre a união de dois grupos são armazenados para depois traçar o dendrograma

Observações

- **Empates** na matriz de proximidades:
  - devem ser resolvidos arbitrariamente
    - é simples verificar que **single linkage é invariante**!
      - 2 uniões candidatas empatadas serão sempre feitas (e em seguida)
    - mas **complete link** pode ser fortemente afetado pela decisão...
      - **Exercício:** mostrar isso através de um exemplo

Observações

- **Propriedades**:
  - O algoritmo de grafo visto anteriormente deixa evidente que *single* e *complete link* são **monótonos**
    - dissimilaridade das uniões é não decrescente
      - é crescente ao longo da hierarquia se não houver empates
    - dendrograma não possui “reversões”.
  - Além disso, eles são **invariantes** a qualquer **transformação monótona** da matriz de proximidades
    - ou seja, que não altera a ordem relativa dos elementos

Observações

- **Outras Motivações para o Estudo das Relações com Grafos (cont.)**:
  - À parte de questões computacionais e outras questões de cunho prático, os princípios fundamentais por trás das relações entre *clustering* e grafos são importantes
    - medidas de conectividade, p. ex., podem definir inter-relacionamentos indiretos entre objetos de um grupo, ao invés de similaridade explícita
  - Alguns algoritmos modernos de agrupamento de dados são baseados em grafos.

Referências

- Jain, A. K. and Dubes, R. C., Algorithms for Clustering Data, Prentice Hall, 1988
- Everitt, B. S., Landau, S., and Leese, M., *Cluster Analysis*, Arnold, 4th Edition, 2001.
- Tan, P.-N., Steinbach, M., and Kumar, V., *Introduction to Data Mining*, Addison-Wesley, 2006