Agrupamento por similaridade - Processos clássicos

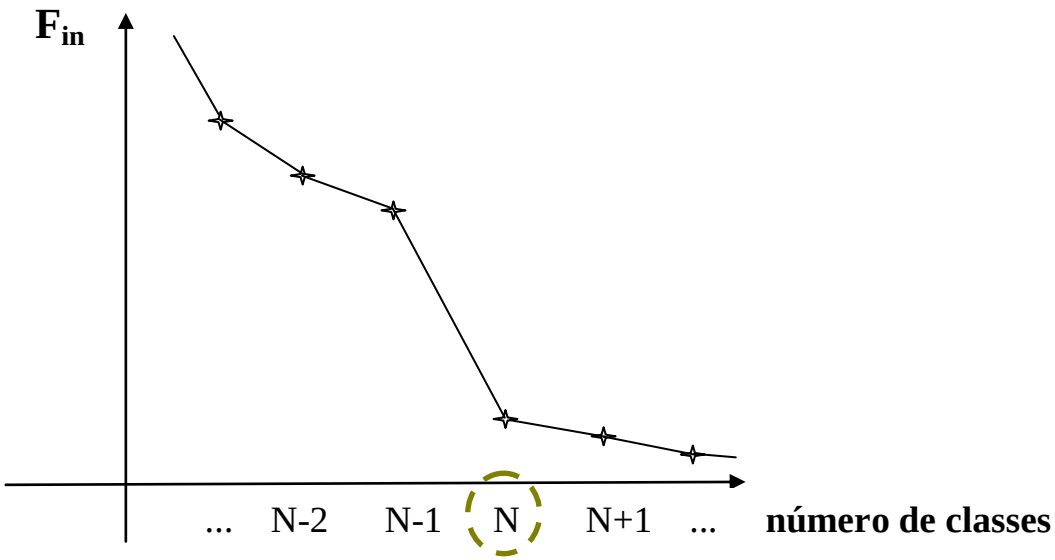
Agrupar, "de forma natural", conjuntos de dados com "similaridade interna"

determinar o número de classes à usar e

alocar os elementos às classes para minimizar F_{in}

1 - Determinação do número de classes a utilizar

Heurística: um agrupamento "natural" (um número N "bom" de classes) é alcançado na vizinhança imediata de uma grande variação em F_{in}.



2 - Alocação dos elementos às classes para minimizar a dispersão total intra classes F_{in}

função à minimizar

$$F_{in} = \sum_{\forall C_j} F_j \qquad \textbf{onde} \qquad F_j = n_j \sigma_j^2 = \sum_{\forall \vec{x}_i \in C_j} \left\| \vec{x}_i - \vec{m}_j \right\|^2$$

F_j é a dispersão intra classe da classe Cj

2,1 Localização dos padrões das classes

O padrão $\underline{p_j}$ da classe C_j deve minimizar a dispersão intra classe da classe C_j

$$F_j = \sum_{\forall \vec{x} \in C_j} \left| \vec{x} - \vec{p}_j \right|^2 = \sum_{\forall \vec{x} \in C_j} \sum_{\forall k} \left(x_k - p_{jk} \right)^2 = \sum_{\forall k} \sum_{\forall \vec{x} \in C_j} \left(x_k - p_{jk} \right)^2$$

após pequena álgebra

$$\frac{\partial}{\partial p_{jk}} \sum_{\forall \vec{x} \in C_j} \left(x_k - p_{jk} \right)^2 = 0 \quad \Rightarrow \quad p_{jk} = m_{jk} \quad \Rightarrow \quad \vec{p}_j = \vec{m}_j$$

o padrão que minimiza a dispersão intra classe (ou erro de representação) de uma classe é o seu baricentro (já visto anteriormente).

2.2 – Processos de Agrupamento

Como realizar a

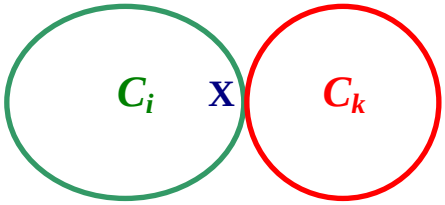
Minimização interativa de F_{in}

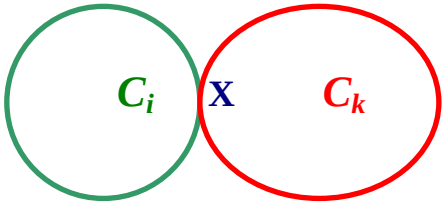
(e simultaneamente a maximização de F_{out}) ?

$$F_{in} = \sum_j F_j$$

Rearranjando os elementos entre as classes para reduzir F_{in}

Deslocando um elemento $\hat{\vec{x}}$ da classe C_i para a classe C_k





Após alguma álgebra:

$$\vec{m}_k = \frac{1}{n_k}\sum_{\forall \vec{x}\in C_k}\vec{x} \quad \Rightarrow \quad \vec{m}'_k = \frac{1}{n'_k}\sum_{\forall \vec{x}\in C'_k}\vec{x} = \frac{1}{n_k+1}\left(\hat{\vec{x}} + n_k\vec{m}_k\right) = \vec{m}_k + \frac{\hat{\vec{x}} - \vec{m}_k}{n_k+1}$$

$$F_k = \sum_{\forall \vec{x}\in C_k}\left|\vec{x}-\vec{m}_k\right|^2 \quad \Rightarrow \quad F'_k = \sum_{\forall \vec{x}\in C'_k}\left|\vec{x}-\vec{m}'_k\right|^2 = \ldots = F_k + \frac{n_k}{n_k+1}\left|\hat{\vec{x}}-\vec{m}_k\right|^2$$

$\hat{\vec{x}}$

$$\vec{m}_i = \frac{1}{n_i}\sum_{\forall \vec{x}\in C_i}\vec{x} \quad \Rightarrow \quad \vec{m}'_i = \frac{1}{n'_i}\sum_{\forall \vec{x}\in C'_i}\vec{x} = \frac{1}{n_i-1}\left(-\hat{\vec{x}} + n_i\vec{m}_i\right) = \vec{m}_i - \frac{\hat{\vec{x}} - \vec{m}_i}{n_i-1}$$

$$F_i = \sum_{\forall \vec{x}\in C_i}\left|\vec{x}-\vec{m}_i\right|^2 \quad \Rightarrow \quad F'_i = \sum_{\forall \vec{x}\in C'_i}\left|\vec{x}-\vec{m}'_i\right|^2 = \ldots = F_i - \frac{n_i}{n_i-1}\left|\hat{\vec{x}}-\vec{m}_i\right|^2$$

$$\Delta F_{in} = \left(F'_k - F_k\right) + \left(F'_i - F_i\right) = \frac{n_k}{n_k+1}\left|\hat{\vec{x}}-\vec{m}_k\right|^2 - \frac{n_i}{n_i-1}\left|\hat{\vec{x}}-\vec{m}_i\right|^2$$

F_{in} é reduzido quando deslocamos um elemento $\hat{\vec{x}}$ da classe C_i para a classe C_k *se* $\Delta F_{in} < 0$, i.e. se

$$\frac{n_i}{n_i - 1}\left|\hat{\vec{x}} - \vec{m}_i\right| > \frac{n_k}{n_k + 1}\left|\hat{\vec{x}} - \vec{m}_k\right|^2$$

como $\frac{n_i}{n_i - 1} > 1 > \frac{n_k}{n_k + 1}$ **podemos simplificar a condição, basta que**

$$\left|\hat{\vec{x}} - \vec{m}_i\right| > \left|\hat{\vec{x}} - \vec{m}_k\right|^2$$

i.e. que a distância do elemento ao baricentro da nova classe seja menor que ao da antiga.

2.3 Um processo de clusterização interativo simples e eficaz: K-means, C-means, Isodata ou Lloyd

entradas: $\vec{x}_i \quad i = 1,2,...,N$

centros de classe: $\vec{m}_j \quad j = 1,2,...,K$

Inicialização: Arbitre K centros $\vec{m}_j \quad j = 1,2,...,K$

sugestão: use as K primeiras entradas $\vec{x}_i \quad i = 1,2,...,K$
não muito próximas entre si (à discutir mais tarde)

Loop

Reclassificação das entradas

$$\forall \ \vec{x}_i \quad i = 1,2,...,N$$

Se $\vec{m}_j$ é a melhor representação de $\vec{x}_i$, i.e. se

$$\left|\vec{x}_i - \vec{m}_j\right| < \left|\vec{x}_i - \vec{m}_k\right| \quad \forall k \neq j \quad \textbf{então}$$

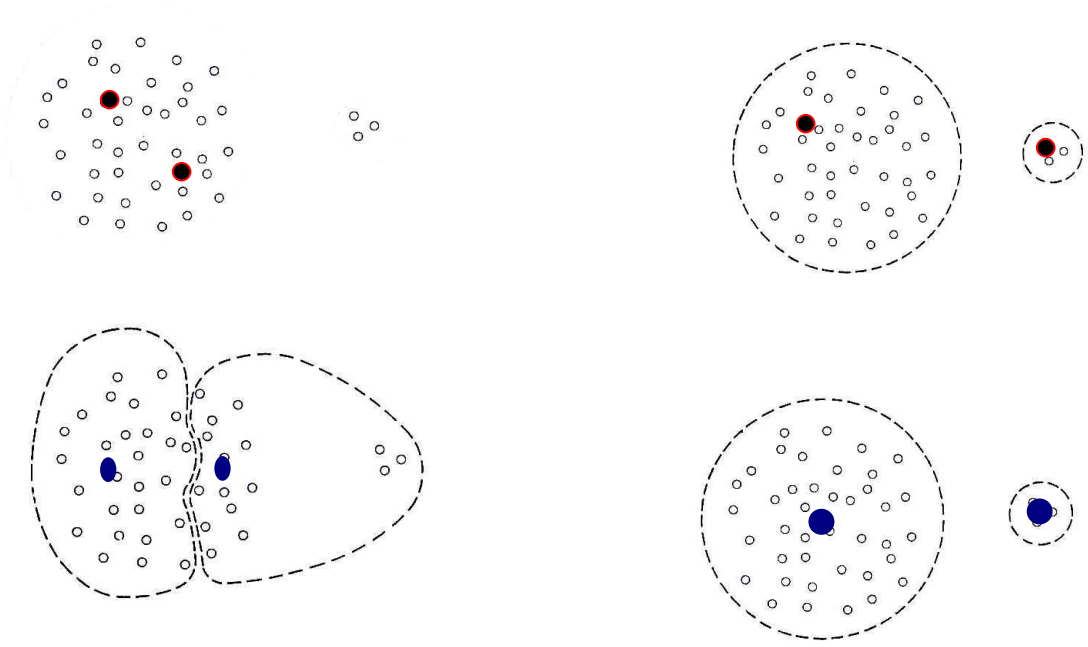
$$\vec{x}_i \in C_j$$

Atualização dos centros de classe

$$\forall \quad j = 1,2,...,K$$

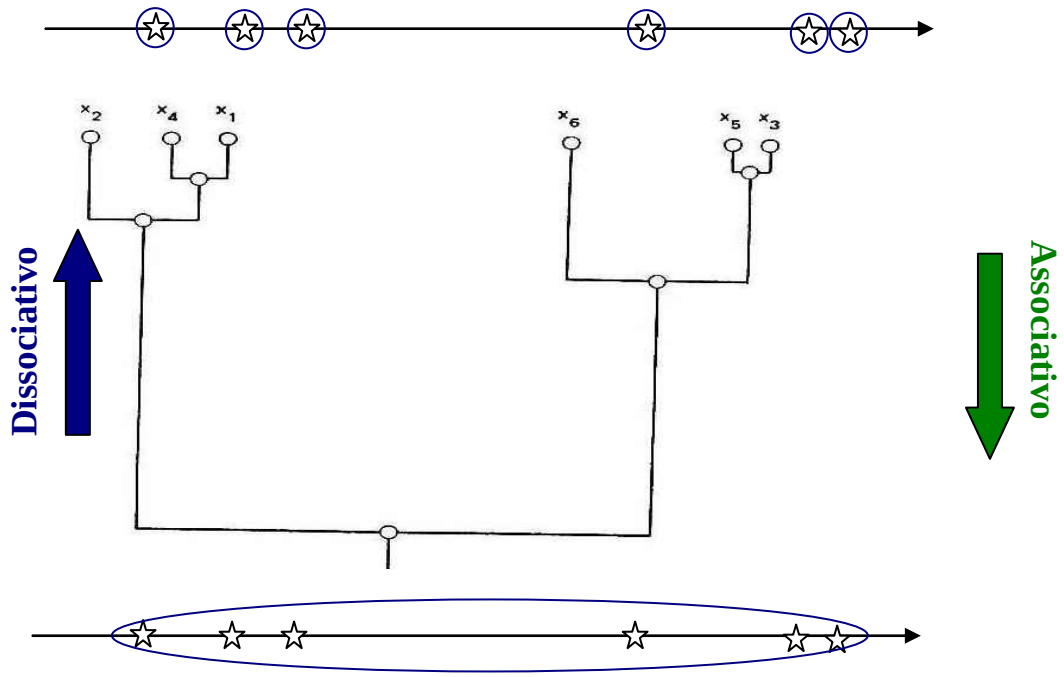
$$\vec{m}_j = \frac{1}{n_j} \sum_{\forall \vec{x} \in C_j} \vec{x}$$

Quando nenhum elemento trocar de classe: Fim.



inicializar com entradas não muito próximas

2.4 - Algorítmos Hierárquicos: dendogramas, árvores

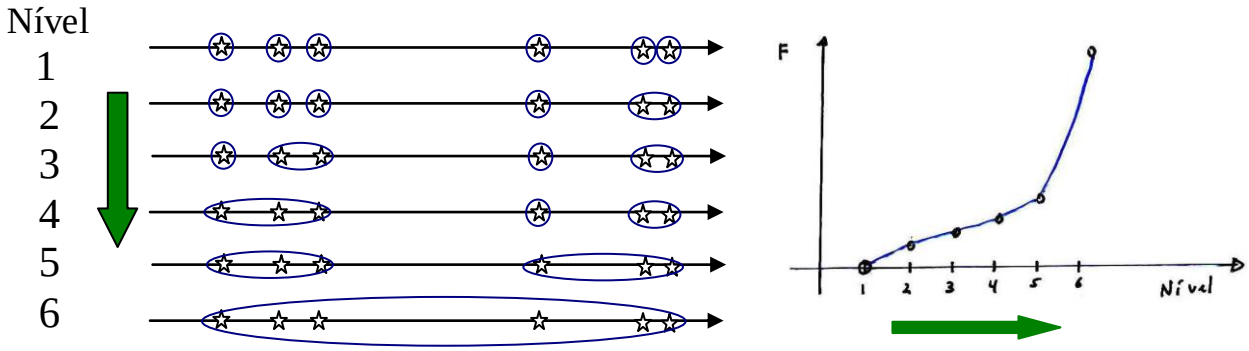


2.4.1 Algorítmo Aglomerativo, Associativo:

n elementos c classes

c = n n-1 n-2 ... 2 1

F_{in} = F_n=0 < F_{n-1} < F_{n-2} < F_2 < F_1



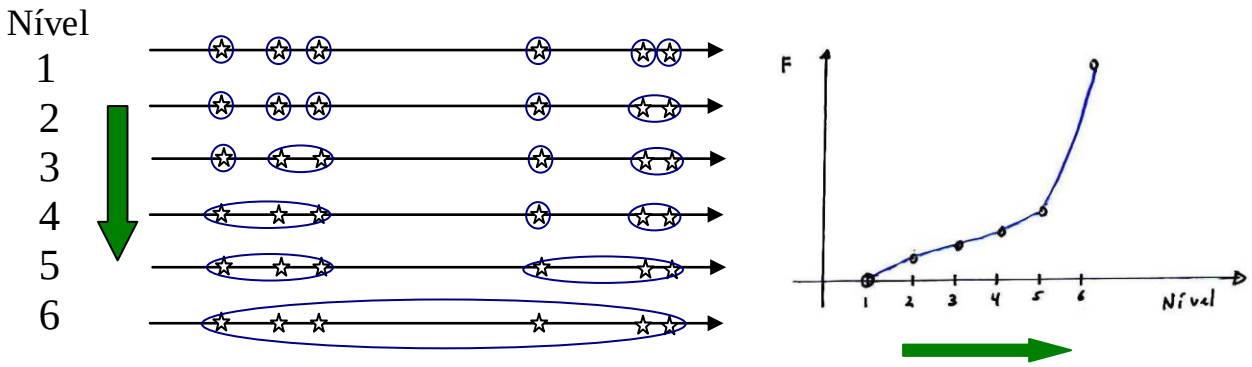
Algoritmo Hierárquico Associativo:

Inicialização: **n elementos** ⟹ **n clusters**

Loop:

Junte os dois clusters que aumentam F_{in} o mínimo possível

Repetir até chegar ao número de clusters desejado ou F_{in} crescer muito



Obs:

se $\underline{x}_i$ e $\underline{x}_j$ estão em um mesmo cluster, permanecerão juntos.

É dito um algorítmo hierárquico.

Que clusters juntar ?

$$C_j \qquad \vec{m}_j = \frac{1}{n_j} \sum_{\forall \vec{x} \in C_j} \vec{x} \qquad F_j = \sum_{\forall \vec{x} \in C_j} |\vec{x}|^2 - n_j |\vec{m}_j|^2$$

$$C_k \qquad \vec{m}_k = \frac{1}{n_k} \sum_{\forall \vec{x} \in C_k} \vec{x} \qquad F_k = \sum_{\forall \vec{x} \in C_k} |\vec{x}|^2 - n_k |\vec{m}_k|^2$$

$$C_{jk} = C_j \cup C_k \qquad \vec{m}_{jk} = \frac{n_j \vec{m}_j + n_k \vec{m}_k}{n_j + n_k} \qquad F_{jk} = \sum_{\forall \vec{x} \in C_j, C_k} |\vec{x}|^2 - (n_j + n_k) |\vec{m}_{jk}|^2$$

$$\Delta F = F_{jk} - \left(F_j + F_k\right) = \frac{n_j \; n_k}{n_j + n_k} |\vec{m}_j - \vec{m}_k|^2$$

$$\Delta F = \frac{n_j \; n_k}{n_j + n_k} |\vec{m}_j - \vec{m}_k|^2 \qquad \textbf{deve ser pequeno}$$

escolher centros próximos:

$$\vec{m}_j \approx \vec{m}_k$$

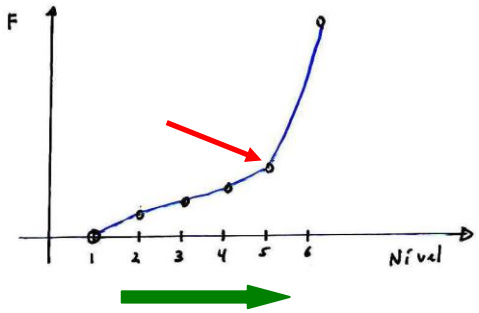
de preferência com populações desiguais:

$$\frac{n_j n_k}{n_j + n_k} \approx \begin{cases} n_j & se \quad n_j << n_k \\ \dfrac{n_j}{2} & se \quad n_j \approx n_k \end{cases}$$

Quando parar o processo ?

Heurística: um agrupamento "natural" (um número "bom" de classes) é alcançado no algorítmo associativo imediatamente antes de F_{in} apresentar um grande acréscimo em um único passo.

Por este critério o ponto de parada está indicado pela seta vermelha no exemplo anterior.

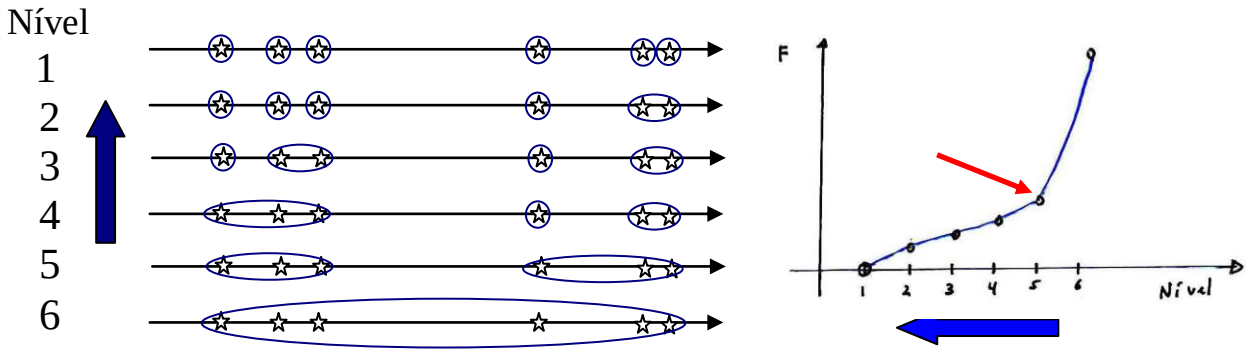


2.4.2 Algorítmo Divisivo, Dissociativo:

n elementos c classes

c = 1 2 3 ... n-1 n

F_{in} = F_1 < $F_{2<}$ F_3 < F_{n-1} < F_n=0



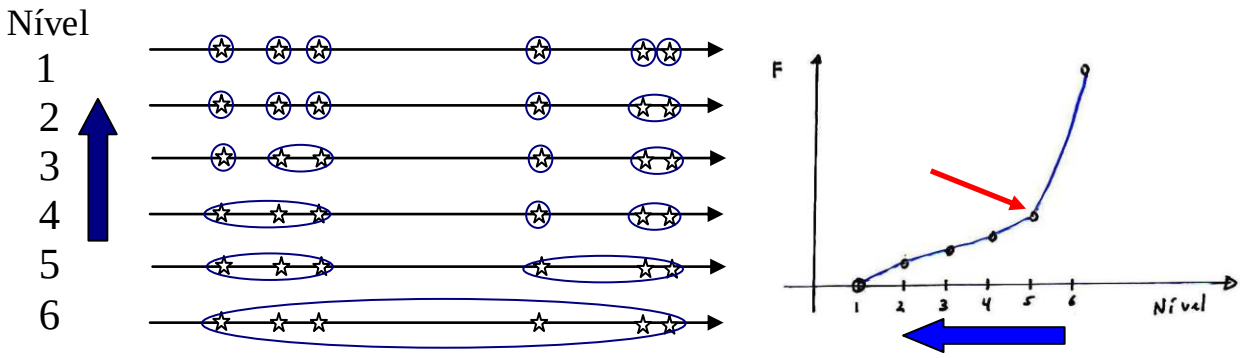
Algoritmo Hierárquico Dissociativo:

Inicialização: **n elementos** ⇨ **1 cluster**

Loop:

Divida o cluster que reduz F_{in} o máximo possível

Repetir até chegar ao número de clusters desejado ou F_{in} quase não decair



Que cluster dividir ?

Critério: maior F_j , por exemplo

Método:

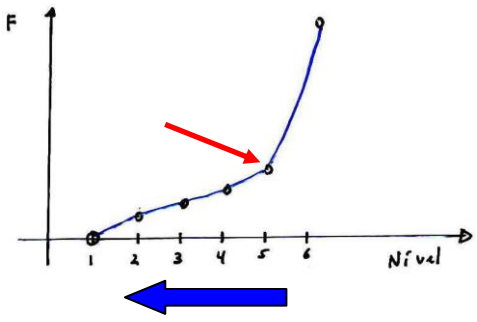
$$\vec{m}_j \rightarrow \begin{cases} \vec{m}_j + \vec{\delta} \\ \vec{m}_j - \vec{\delta} \end{cases}$$

⇨ **K-means**

Quando parar o processo ?

Heurística: um agrupamento "natural" (um número "bom" de classes) é obtido no algorítimo dissociativo imediatamente após F_{in} apresentar um grande decréscimo em um único passo.

Por este critério o ponto de parada esta indicado pela seta vermelha no exemplo anterior.



2.5 - Algorítmos Mistos

Iniciar com K-means

Associar / Dissociar verificando a variação de F_{in}