SCC0173 – Mineração de Dados Biológicos

Agrupamento de Dados – Partes III & IV:
Métodos Particionais e Validação

Prof. Ricardo J. G. B. Campello

SCC / ICMC / USP

Créditos

- O material a seguir consiste de adaptações e extensões dos originais:
  - gentilmente cedidos pelo Prof. Eduardo R. Hruschka
  - de (Tan et al., 2006)
  - de E. Keogh (SBBD 2003)
  - de G. Piatetsky-Shapiro (KDNuggets)

Conteúdo

- Algoritmos Particionais
  - k-means
  - Variantes do k-means
  - Estimativa do número de grupos k
- Critérios relativos de validade de agrupamento
- Critérios externos de validade de agrupamento

Métodos Particionais (Sem Sobreposição)

- Métodos *particionais* sem sobreposição referem-se a algoritmos de agrupamento que buscam (explícita ou implicitamente) por uma partição rígida de um conjunto de objetos $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$
  - **Partição Rígida**: coleção $\mathbf{P} = \{\mathbf{C}_1, \mathbf{C}_2, ..., \mathbf{C}_k\}$ de k grupos não sobrepostos tal que:

$$\mathbf{C}_1 \cup \mathbf{C}_2 \cup ... \cup \mathbf{C}_k = \mathbf{X}$$

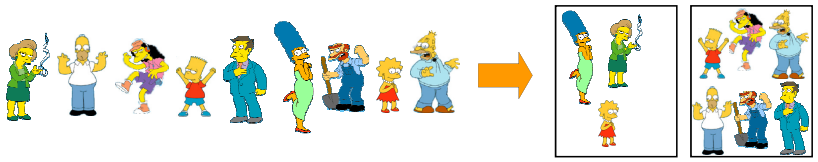
$$\mathbf{C}_i \neq \emptyset$$

$$\mathbf{C}_i \cap \mathbf{C}_j = \emptyset \quad \text{para} \quad i \neq j$$

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

Métodos Particionais (Sem Sobreposição)

- Cada exemplo pertence a um *cluster* dentre *k clusters* possíveis
- Usuário normalmente deve fornecer o número de *clusters (k)*
- Normalmente envolvem a **otimização** de algum índice (critério numérico) que reflete a **qualidade** de determinada partição



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

Particionamento como Problema Combinatório

- **Problema**: Assumindo que *k* seja conhecido, o no. de possíveis formas de agrupar *N* objetos em *k clusters* é dado por (Liu, 1968):

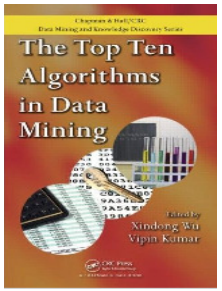
$$NM(N,k) = \frac{1}{k!}\sum_{i=0}^{k}(-1)^i \binom{k}{i}(k-i)^N$$

- Por exemplo, NM(100, 5) ≈ 56.6×10^{67}.
  - Em um computador com capacidade de avaliar 10^9 partições/s, precisaríamos ≈ 1.8×10^{50} séculos para processar todas as avaliações...
- Como k em geral é desconhecido, problema é ainda maior...
  - Avaliação computacional exaustiva é impraticável (problema NP-Hard)...
- **Solução**: formulações alternativas...

Baseado no original do Prof. Eduardo R. Hruschka

Algoritmo k-Means

- Começaremos nosso estudo com um dos algoritmos mais clássicos da área de **mineração de dados** em geral
  - algoritmo das **k-médias** ou ***k-means***
  - listado entre os **Top 10 Most Influential Algorithms in DM**



• Wu, X. and Kumar, V. (Editors), *The Top Ten Algorithms in Data Mining*, CRC Press, 2009

• X. Wu et al., "Top 10 Algorithms in Data Mining", *Knowledge and Info. Systems*, vol. 14, pp. 1-37, 2008

k-means

1) Escolher aleatoriamente um número *k* de protótipos (centros) para os clusters
2) Atribuir cada objeto para o cluster de centro mais *próximo* (segundo alguma distância, e.g. Euclidiana)
3) Mover cada centro para a média (centróide) dos objetos do cluster correspondente
4) Repetir os passos 2 e 3 até que algum critério de convergência seja obtido:

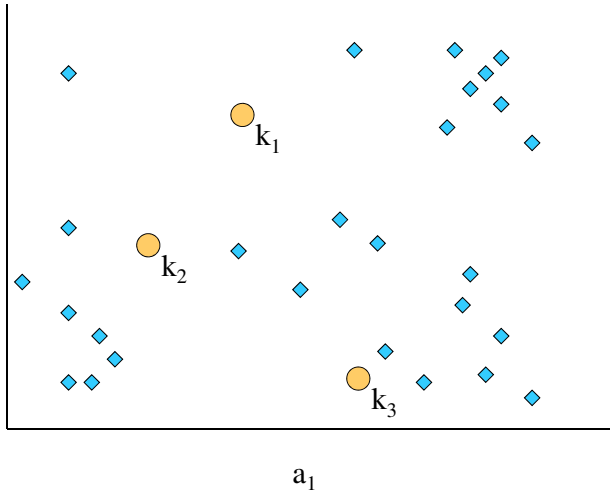
   - número máximo de iterações

   - limiar mínimo de mudanças nos centróides

Prof. Eduardo R. Hruschka

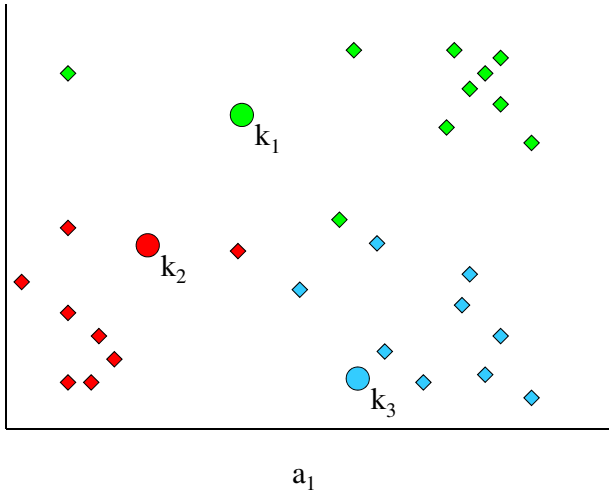
k-means - passo 1:

Escolher 3 centros iniciais



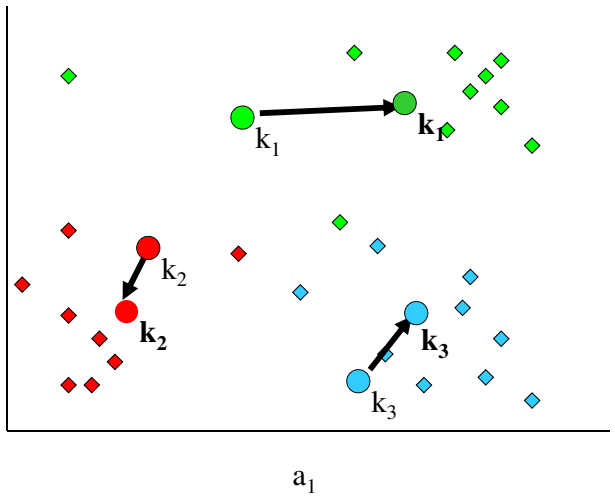
k-means - passo 2:

Atribuir cada objeto ao cluster de centro + próximo



k-means - passo 3:

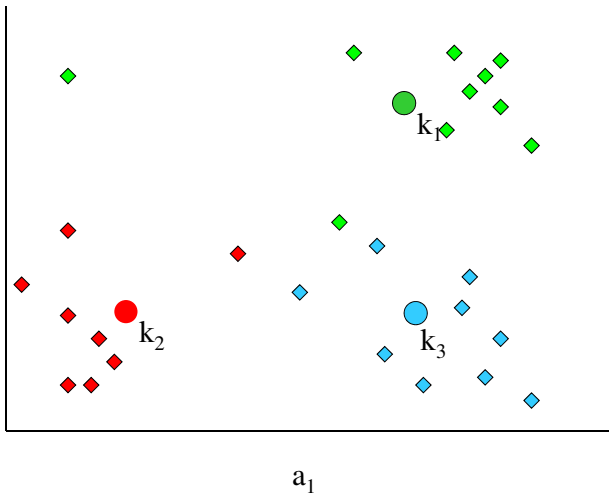
Mover cada centro para o vetor médio do cluster (centróide)



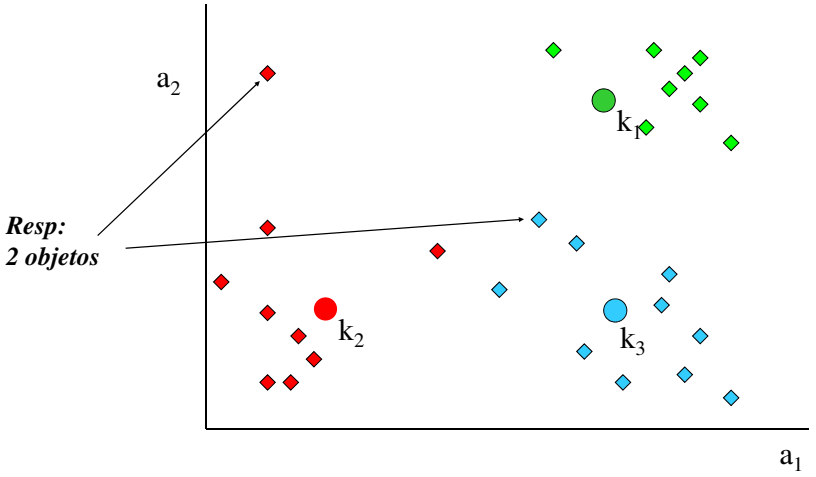
k-means:

Re-atribuir objetos aos clusters de centróides mais próximos

Quais objetos mudarão de cluster?

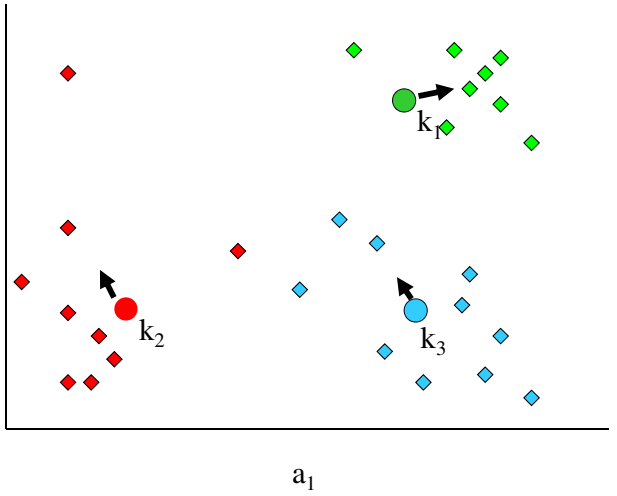


k-means:



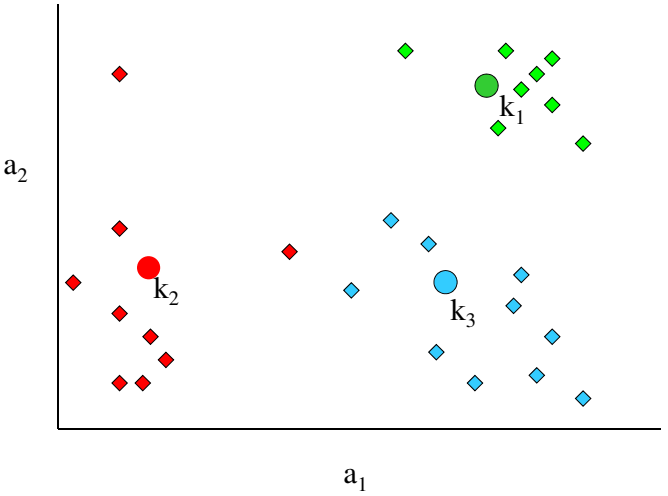
k-means:

re-calcular
vetores
médios



k-means:

mover
centros dos
clusters…



Critério de Qualidade das Partições

- Pode-se demonstrar que o algoritmo minimiza a seguinte função objetivo (variâncias intra-cluster):

$$J = \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in \mathbf{C}_i} d\left(\mathbf{x}_j, \bar{\mathbf{x}}_i\right)^2$$

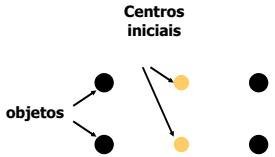
onde $\bar{\mathbf{x}}_i$ é o centróide do i-ésimo cluster:

$$\bar{\mathbf{x}}_i = \frac{1}{|\mathbf{C}_i|} \sum_{\mathbf{x}_i \in \mathbf{C}_i} \mathbf{x}_i$$

- Exemplo de cálculo de J: no quadro...

Discussão

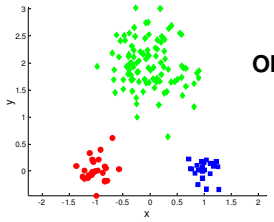
- Resultado pode variar significativamente dependendo da escolha das sementes (protótipos) iniciais
- *k*-means pode "ficar preso" em ótimos locais
  - Exemplo:



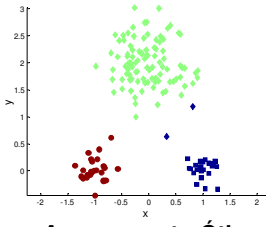
- Como evitar ... ?

Slide baseado no curso de Gregory Piatetsky-Shapiro, disponível em http://www.kdnuggets.com

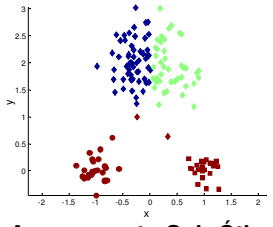
Exemplo: Duas Inicializações Diferentes



Objetos Originais

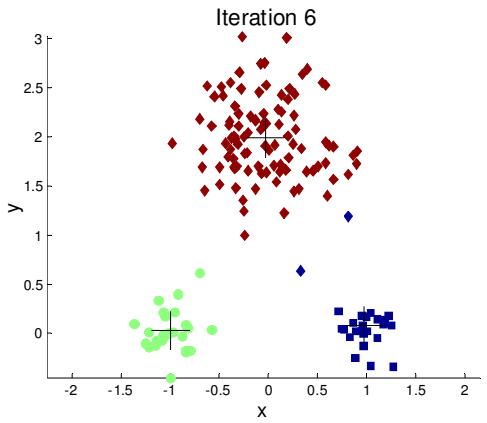


Agrupamento Ótimo

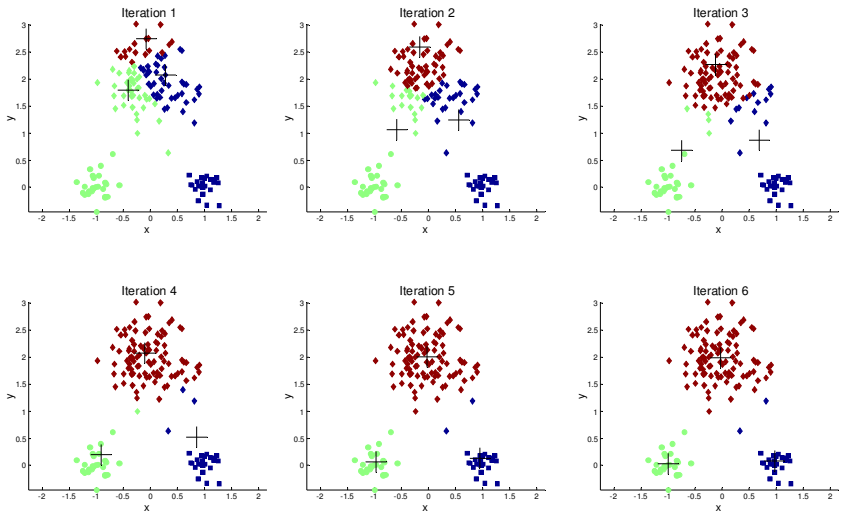


Agrupamento Sub-Ótimo

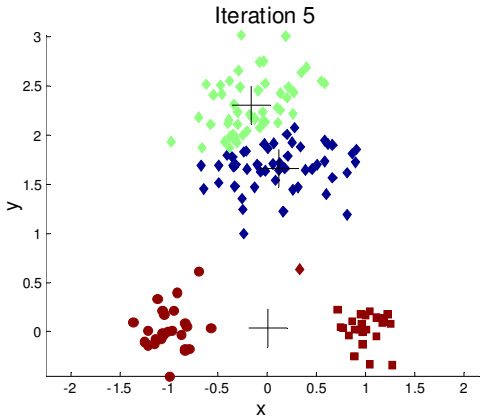
Exemplo: Execução com a 1ª Inicialização



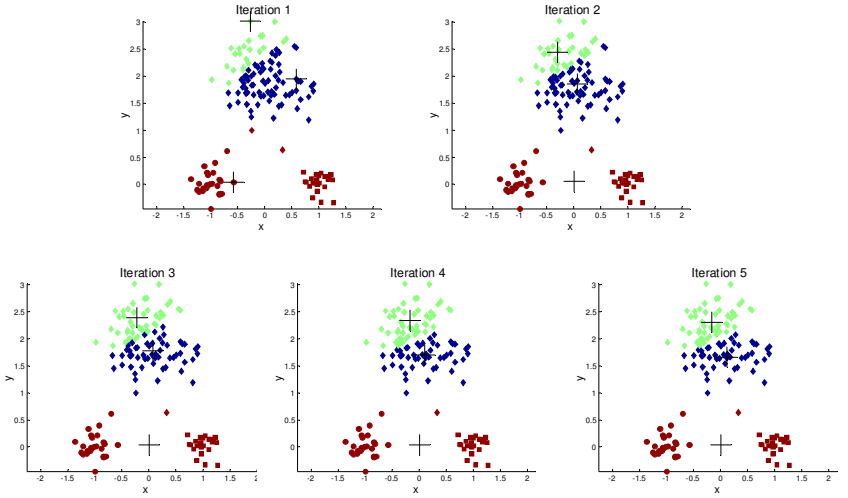
Exemplo: Execução com a 1ª Inicialização



Exemplo: Execução com a 2ª Inicialização



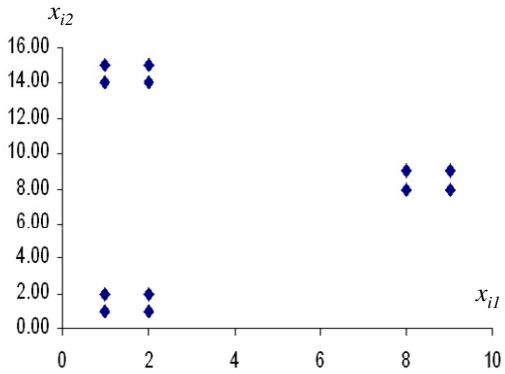
Exemplo: Execução com a 2ª Inicialização



Exercício

Objeto $\mathbf{x}_i$	x_{i1}	x_{i2}
1	1	2
2	2	1
3	1	1
4	2	2
5	**8**	**9**
6	**9**	**8**
7	**9**	**9**
8	**8**	**8**
9	1	15
10	2	15
11	1	14
12	2	14

Prof. Eduardo R. Hruschka



- Executar k-means com k=3 nos dados acima a partir dos protótipos [6 6], [4 6] e [5 10] e outros a sua escolha

Implementações Eficientes

- Desempenho pode ser "turbinado" de diferentes formas:
  - **Heurísticas de Inicialização**
  - **Estruturas de Dados**
  - **Algoritmos**, por exemplo:
    - **Atualização recursiva dos centróides**:
      - Cálculo de cada centróide só depende do número de objetos do cluster em questão, dos novos objetos atribuídos ao cluster, dos objetos que deixaram o cluster, e do valor anterior do centróide
      - Não demanda recalcular tudo novamente
      - **Exercício:** a partir da equação do cálculo do centróide, escrever a equação de atualização recursiva descrita acima !

Resumo do k-Means

Vantagens

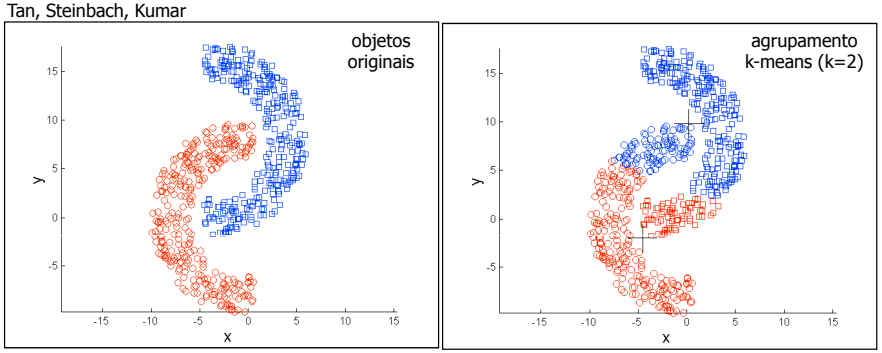
- Simples e intuitivo
- Possui complexidade computacional linear em todas as variáveis críticas (N, n, k)
- Eficaz em muitos cenários de aplicação e produz resultados de interpretação relativamente simples
- Considerado um dos 10 mais influentes algoritmos em Data Mining (Wu & Kumar, 2009) !

Desvantagens

- k = ?
- Sensível à inicialização dos protótipos (mínimos locais de J)
- Limita-se a encontrar clusters **volumétricos / globulares**
- Cada item deve pertencer a um único cluster (partição é rígida, ou seja, sem sobreposição)
- Limitado a atributos numéricos
- Sensível a *outliers*

Nota (Formas Não-Globulares)

Tan, Steinbach, Kumar



- Na prática, esse problema em geral não é crítico, i.e., há pouco interesse na maioria das aplicações de mundo real
  - Grandes BDs (muitos objetos & atributos) e necessidade de interpretação dos resultados (e.g. segmentação de mercado...)

Algumas Variantes do k-Means

- **K-Medóides:** Substituir cada centróide por um objeto representativo do cluster, denominado **medóide**
  - Medóide = objeto mais próximo aos demais objetos do cluster
    - Mais próximo em média (empates resolvidos aleatoriamente)
  - **Vantagens:**
    - menos sensível a outliers
    - permite cálculo relacional (apenas matriz de distâncias)
      - logo, pode ser aplicado a bases com atributos categóricos
    - convergência assegurada com qualquer medida de (dis)similaridade !
  - **Desvantagem:** Complexidade quadrática com o no. de objetos N

Exemplo de k-Medóides

- Execução de k-medóides com k = 2 na base de dados relacional abaixo, com medóides iniciais $\mathbf{x}_1$ e $\mathbf{x}_2$:

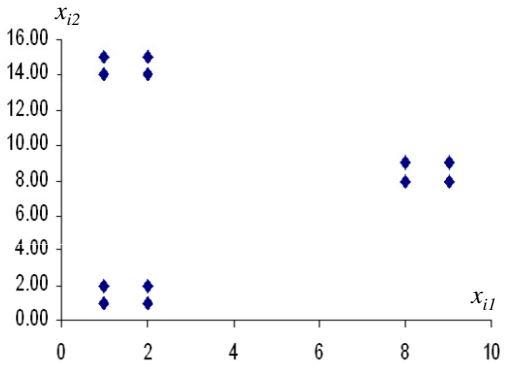
$$\mathbf{D} = \begin{array}{c} \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \end{array} \begin{bmatrix} 0 & 2 & 7 & 13 \\ 2 & 0 & 5 & 10 \\ 7 & 5 & 0 & 4 \\ 13 & 10 & 4 & 0 \end{bmatrix}$$

No quadro...

Exercício

Objeto $\mathbf{x}_i$	x_{i1}	x_{i2}
1	1	2
2	2	1
3	1	1
4	2	2
5	**8**	**9**
6	**9**	**8**
7	**9**	**9**
8	**8**	**8**
9	1	15
10	2	15
11	1	14
12	2	14

Prof. Eduardo R. Hruschka



- Executar k-medóides com k=3 nos dados acima, com medóides iniciais dados pelos objetos 5, 6 e 8

Algumas Variantes do k-Means

- **Métodos de Múltiplas Execuções de k-Means:**
  - Executam k-means repetidas vezes a partir de diferentes valores de k e de posições iniciais dos protótipos
    - Ordenado: n_p inicializações de protótipos para cada $k \in [k_{min}, k_{max}]$
    - Aleatório: n_T inicializações de protótipos, com k sorteado em $[k_{min}, k_{max}]$
  - Tomam a melhor partição resultante de acordo com algum critério de qualidade (critério de validade de agrupamento)
  - **Vantagens:** Estimam k e são menos sensíveis a mínimos locais
  - **Desvantagem:** Custo computacional pode ser elevado

Questão...

- Á própria função objetivo J do k-means não poderia ser utilizada como medida de qualidade para escolher a melhor partição dentre um conjunto de candidatas ???
  - Resposta é sim se todas têm o mesmo no. k de clusters (fixo)
  - Mas e se k for desconhecido, portanto variável...?
- Para responder, considere que as partições são geradas por múltiplas execuções de k-means com $k \in [k_{min}, k_{max}]$

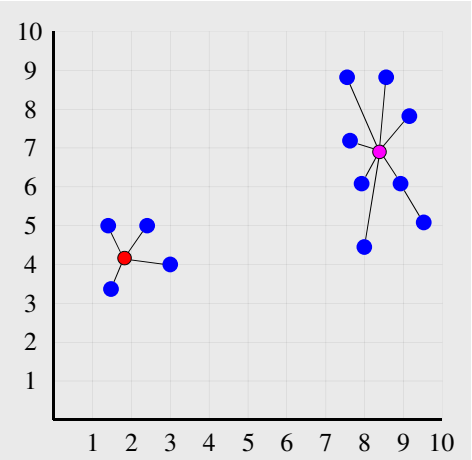
Questão...

- Para tentar responder a questão anterior, vamos considerar o método de múltiplas execuções ordenadas de k-means, com uso da função objetivo J

Erro Quadrático:

$$J = \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in \mathbf{C}_i} d\left(\mathbf{x}_j, \bar{\mathbf{x}}_i\right)^2$$

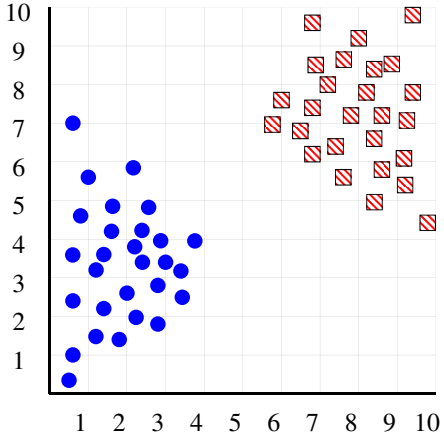
Função Objetivo



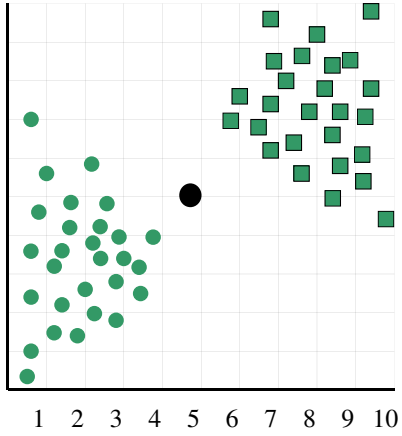
Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Questão...

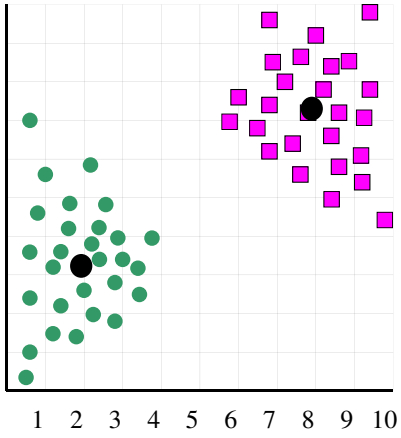
- Considere o seguinte exemplo:



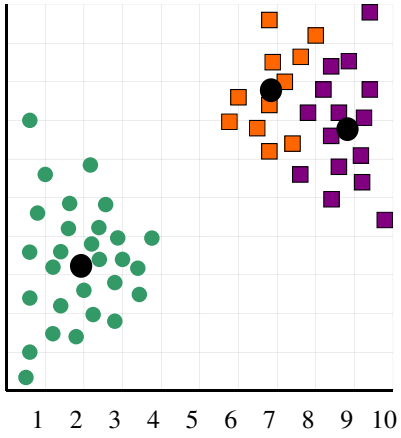
Para $k = 1$, o valor da função objetivo é 873,0.



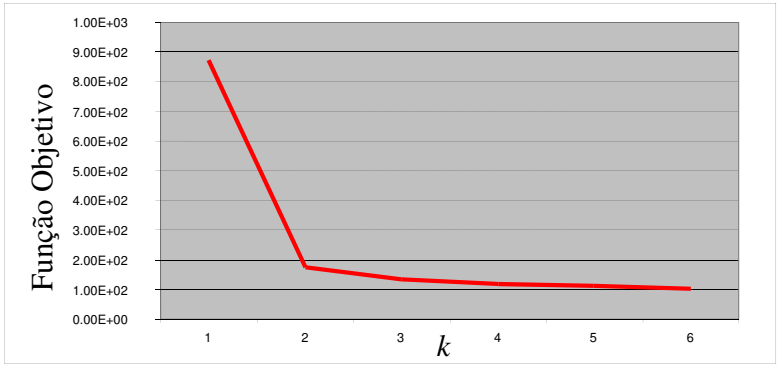
Para $k = 2$, o valor da função objetivo é 173,1.



Para $k = 3$, o valor da função objetivo é 133,6.



Podemos então repetir este procedimento e plotar os valores da função objetivo J para k =1,…,6, ... e tentar identificar um "joelho" :

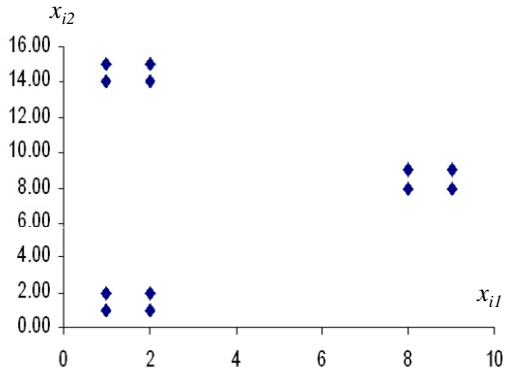


Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Exercício

Objeto $\mathbf{x}_i$	x_{i1}	x_{i2}
1	1	2
2	2	1
3	1	1
4	2	2
5	**8**	**9**
6	**9**	**8**
7	**9**	**9**
8	**8**	**8**
9	1	15
10	2	15
11	1	14
12	2	14

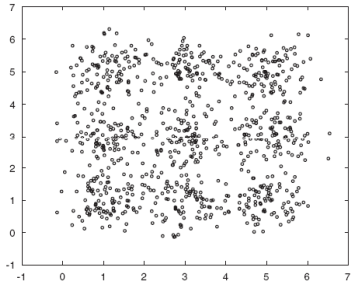
Prof. Eduardo R. Hruschka

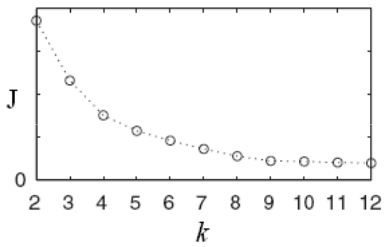


- Executar k-means com k=2 até k=5 nos dados acima e representar graficamente a f. objetivo J em função de k

Questão...

- Infelizmente os resultados não são sempre tão claros quanto no exemplo anterior… Vide exemplo abaixo...





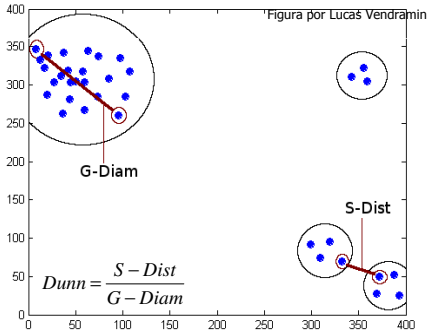
- Além disso, como utilizar essa metodologia em variantes baseadas em busca guiada, que otimizam k ?

Critérios de Validade Relativos

- Para avaliar relativamente a qualidade de diferentes partições, possivelmente com números distintos de grupos, faz-se necessário um tipo de índice:
  - **Critério Relativo de Validade de Agrupamento**
- Existem vários critérios na literatura
- Veremos a seguir dois exemplos
  - Critério de Dunn
  - Critério da Silhueta

Critério de Dunn

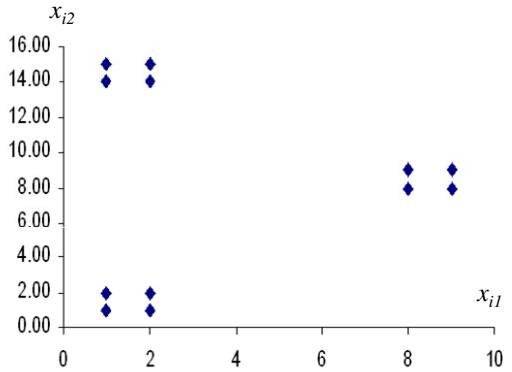
- Razão entre a menor distância inter-grupos (S-Dist) e a maior distância intra-grupo (G-Diam)
- Exemplo:



Exercício

Objeto $\mathbf{x}_i$	x_{i1}	x_{i2}
1	1	2
2	2	1
3	1	1
4	2	2
5	**8**	**9**
6	**9**	**8**
7	**9**	**9**
8	**8**	**8**
9	1	15
10	2	15
11	1	14
12	2	14

Prof. Eduardo R. Hruschka



- Executar k-means nos dados acima e calcular os valores do índice de Dunn das partições obtidas para vários k

Critério da Largura de Silhueta

SWC = Silhueta média sobre todos os objetos: $SWC = \frac{1}{N}\sum_{i=1}^{N} s(i)$

Silhueta (i-ésimo objeto): $s(i) - \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}$

$a(i)$: dissimilaridade média do i-ésimo objeto ao seu cluster

$b(i)$: dissimilaridade média do i-ésimo objeto ao cluster vizinho mais próximo

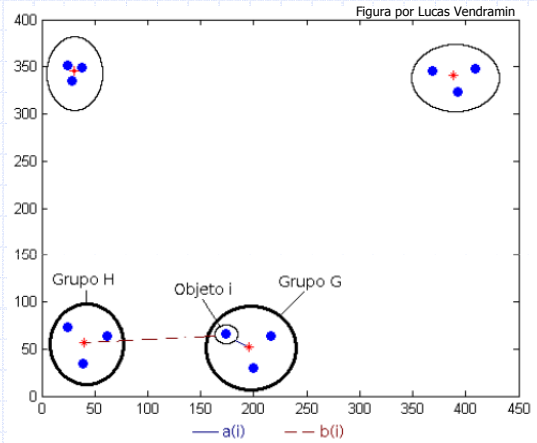
Silhueta Original: $a(i)$ e $b(i)$ são calculados como a distância média (Euclidiana, Mahalanobis, etc) do i-ésimo objeto a todos os demais objetos do cluster em questão. Complexidade quadrática

* Obs: SWC ∈ [–1 ,+1] ; s(i) := 0 para singletons

Critério da Largura de Silhueta

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

$$SWC = \frac{1}{N}\sum_{i-1}^{N} s(i)$$

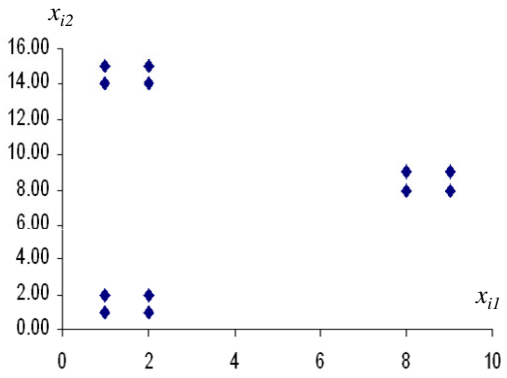


Silhueta Simplificada: $a(i)$ e $b(i)$ são calculados como a distância do i-ésimo objeto ao centróide do cluster em questão. Complexidade linear

Exercício

Objeto $\mathbf{x}_i$	x_{i1}	x_{i2}
1	1	2
2	2	1
3	1	1
4	2	2
5	**8**	**9**
6	**9**	**8**
7	**9**	**9**
8	**8**	**8**
9	1	15
10	2	15
11	1	14
12	2	14

Prof. Eduardo R. Hruschka



- Executar k-means nos dados acima e calcular as silhuetas (original e simplificada) das partições obtidas para vários k

Nota (Normalização dos Dados)

- O problema de normalização / padronização dos dados é mais complexo em clustering do que em outras tarefas de DM (p. ex. classificação)
- As técnicas disponíveis são essencialmente as mesmas, bem como o objetivo da aplicação destas
  - Por exemplo, evitar que atributos com escalas muito maiores do que outros dominem os cálculos de dissimilaridade e, portanto, induzam sozinhos a estrutura de clusters
- No entanto, a aplicação dessas técnicas pode distorcer totalmente a estrutura original dos dados em clusters !!!
  - É preciso mais cautela, experimentação e conhecimento de domínio para realizar pré-processamento de dados em clustering !