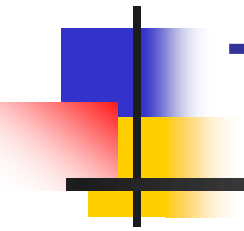
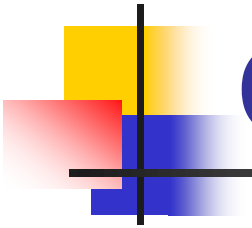


Tópicos Avançados em IA

Prof. Eduardo R. Hruschka

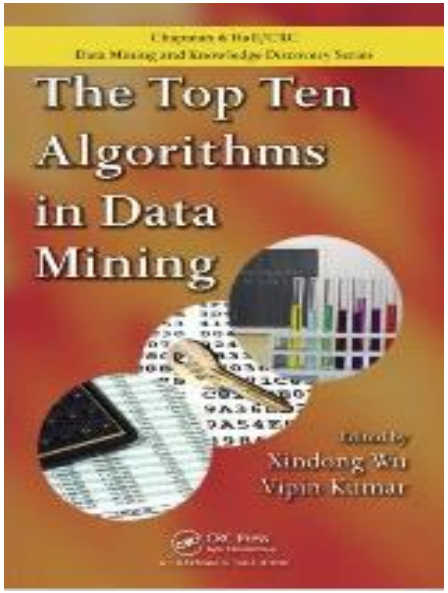


Créditos

- Este material consiste de adaptações dos originais:
  - Elaborados por Eduardo Hruschka e Ricardo Campello
  - de (Tan et al., 2006)
  - de E. Keogh (SBBD 2003)
  - de G. Piatetsky-Shapiro (KDNuggets)

Algoritmo k-Means – Breve Revisão

- 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

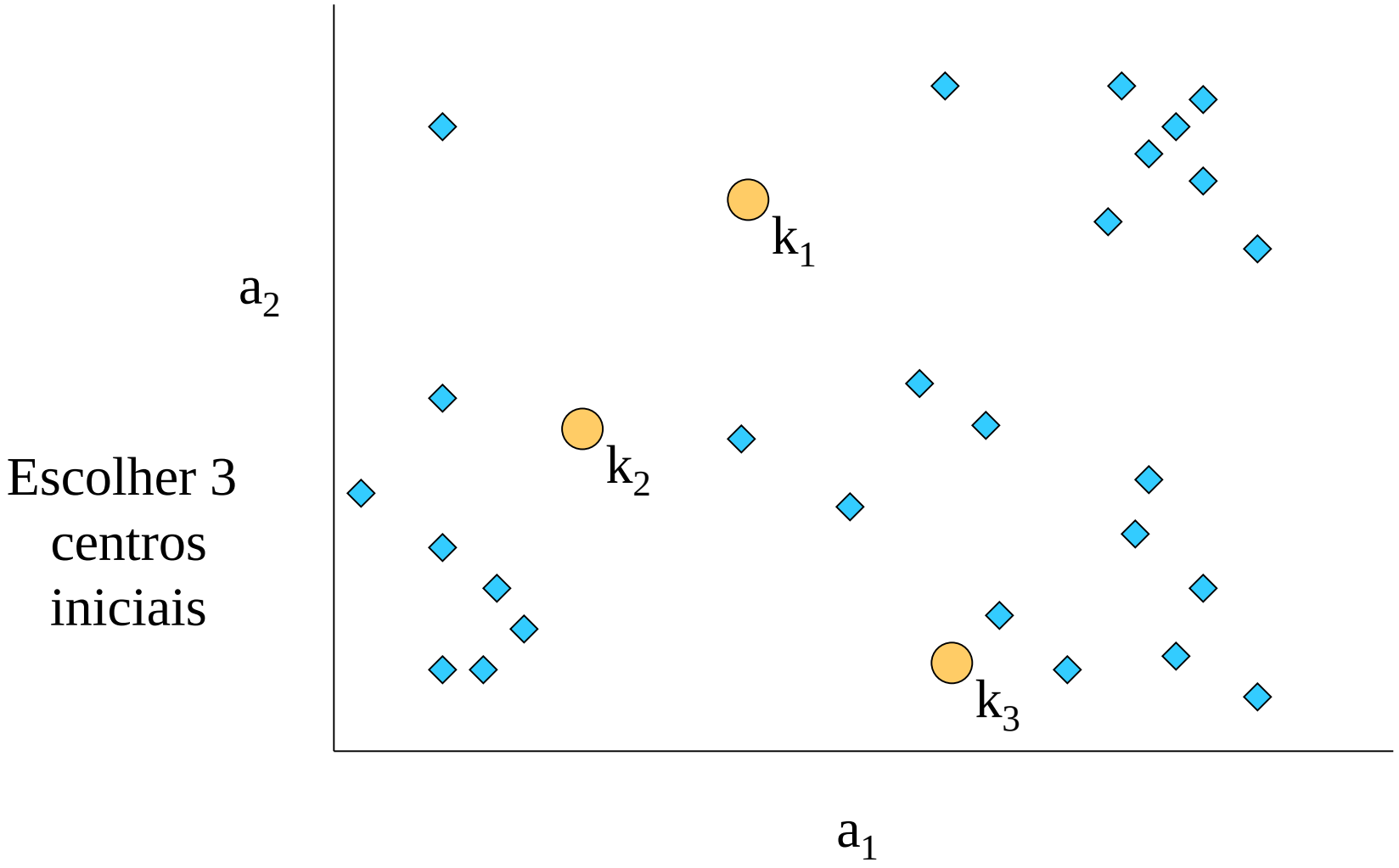
k-Means

1) Escolher aleatoriamente *k* 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

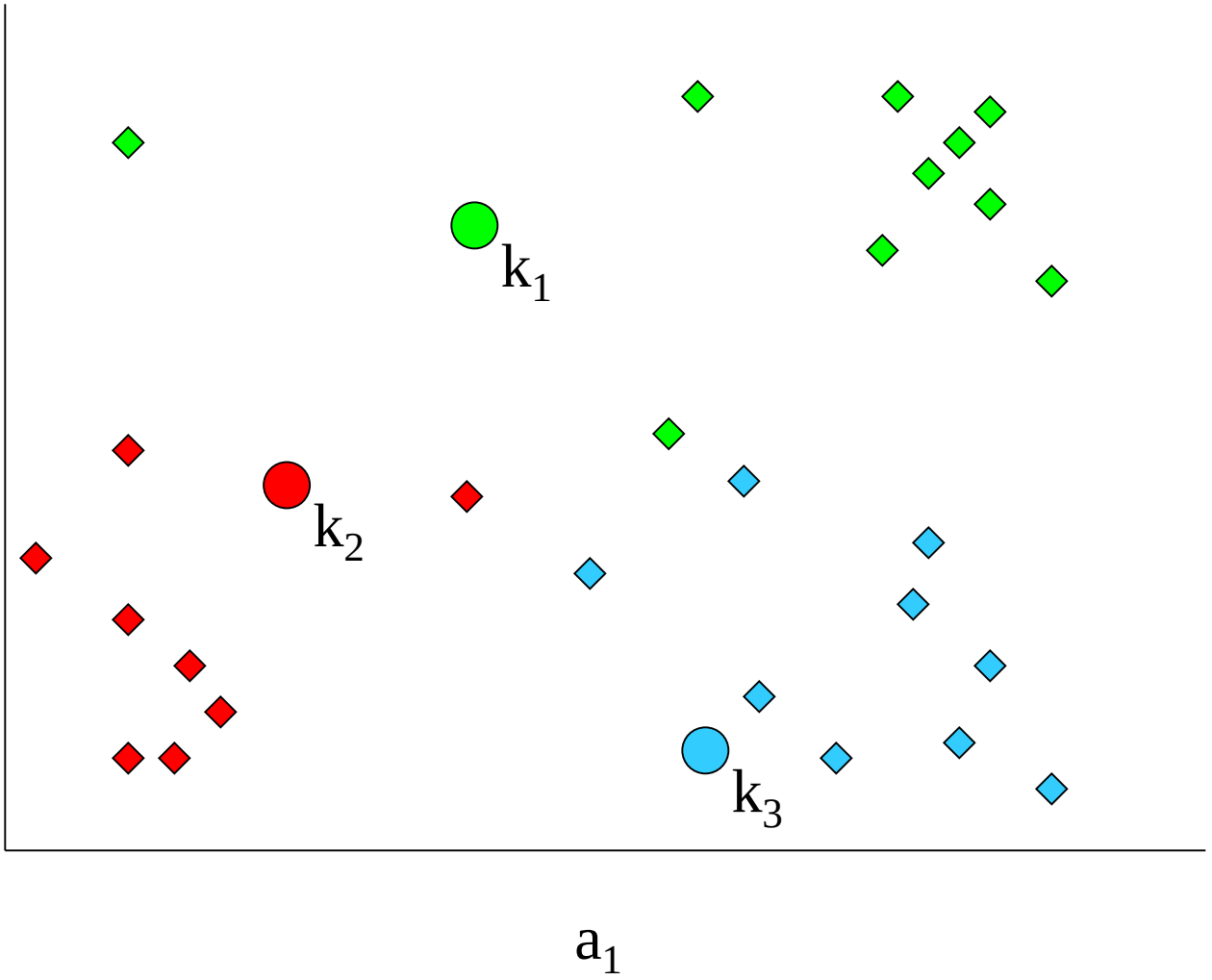
k-Means - passo 1:



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

k-Means - passo 2:

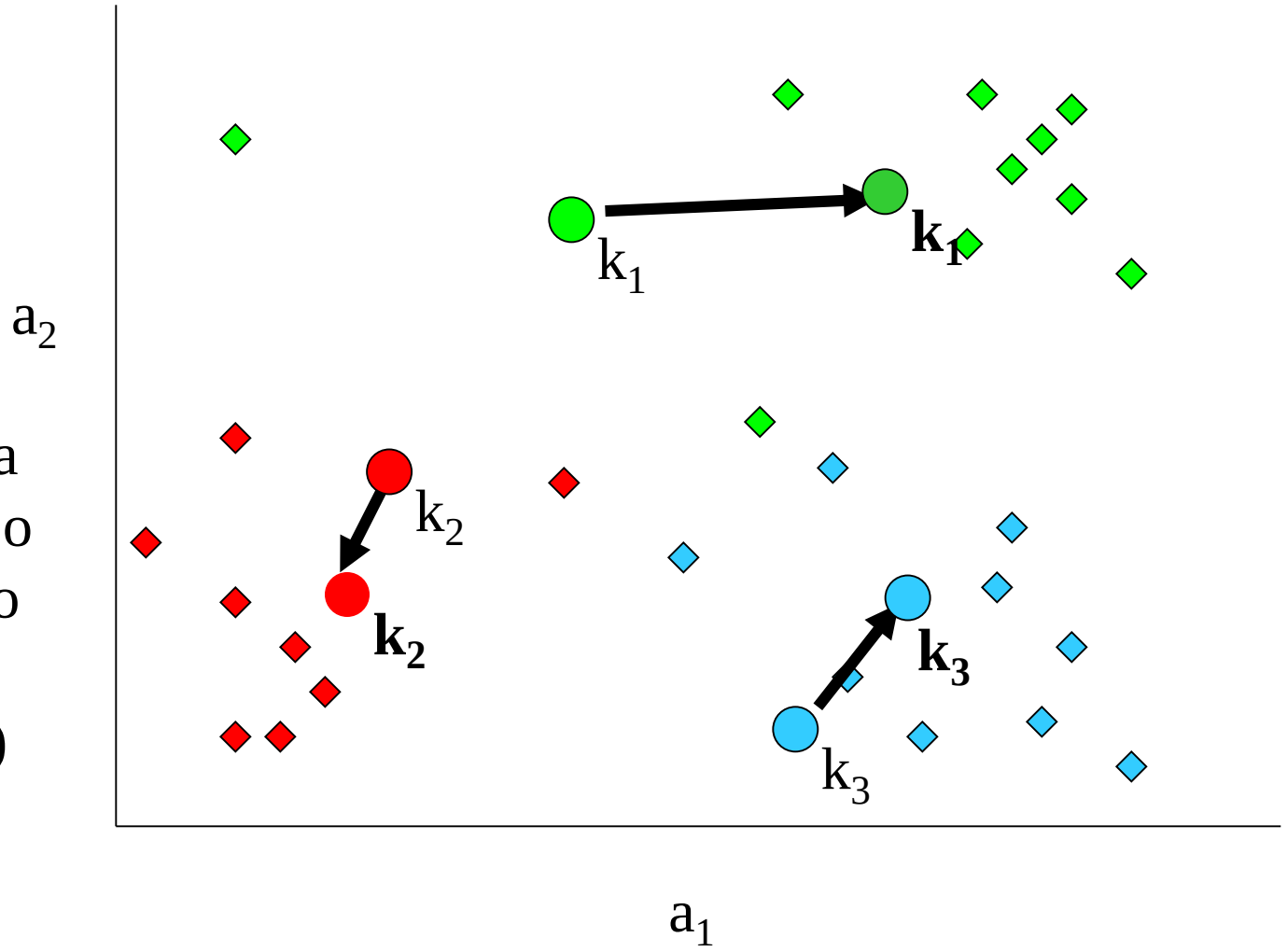
Atribuir cada objeto ao cluster de centro + próximo



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

k-Means - passo 3:

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

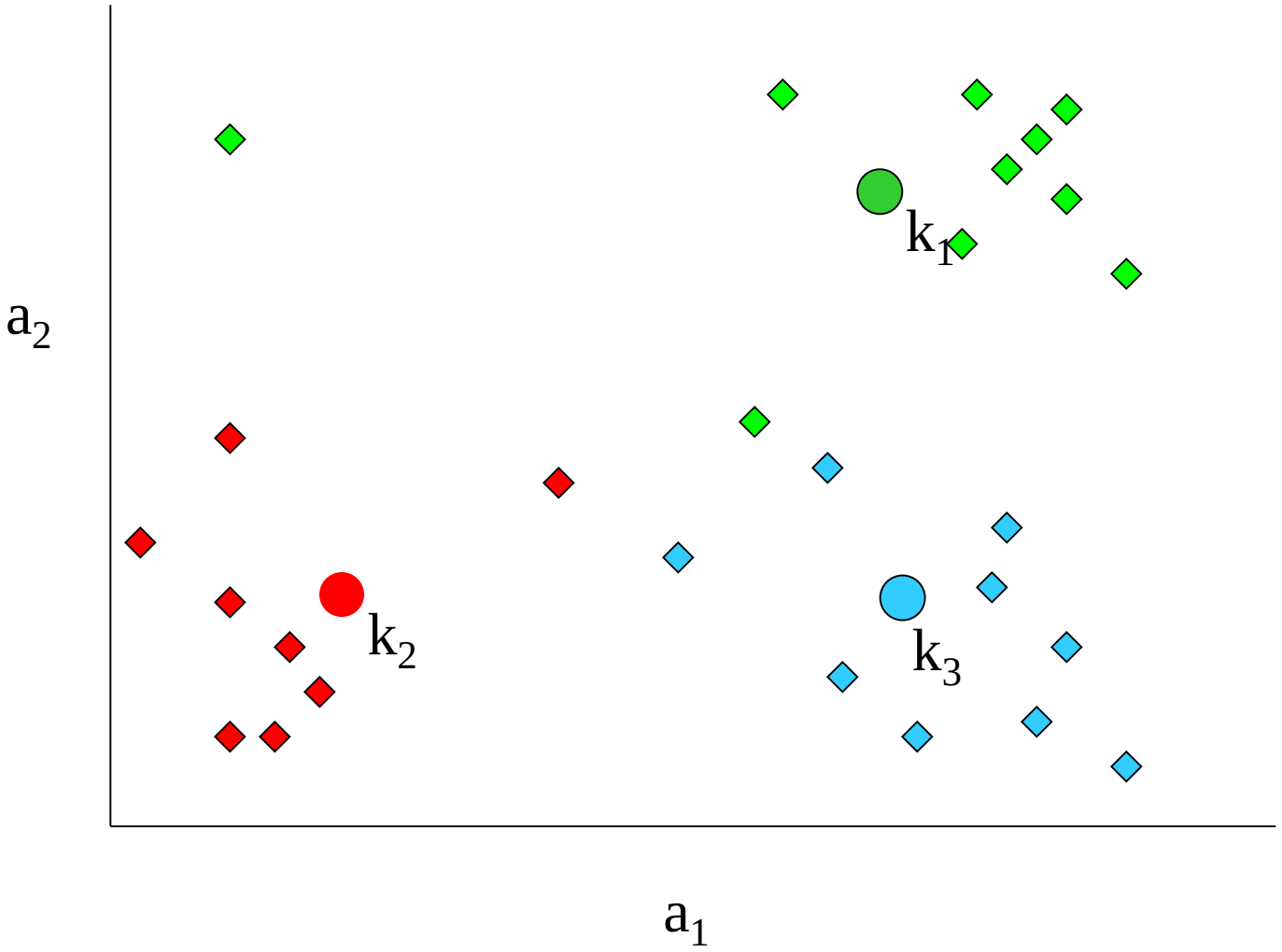


Slide baseado no curso de Gregory Piatetsky-Shapiro, disponível em http://www.kdnuggets.com

k-Means:

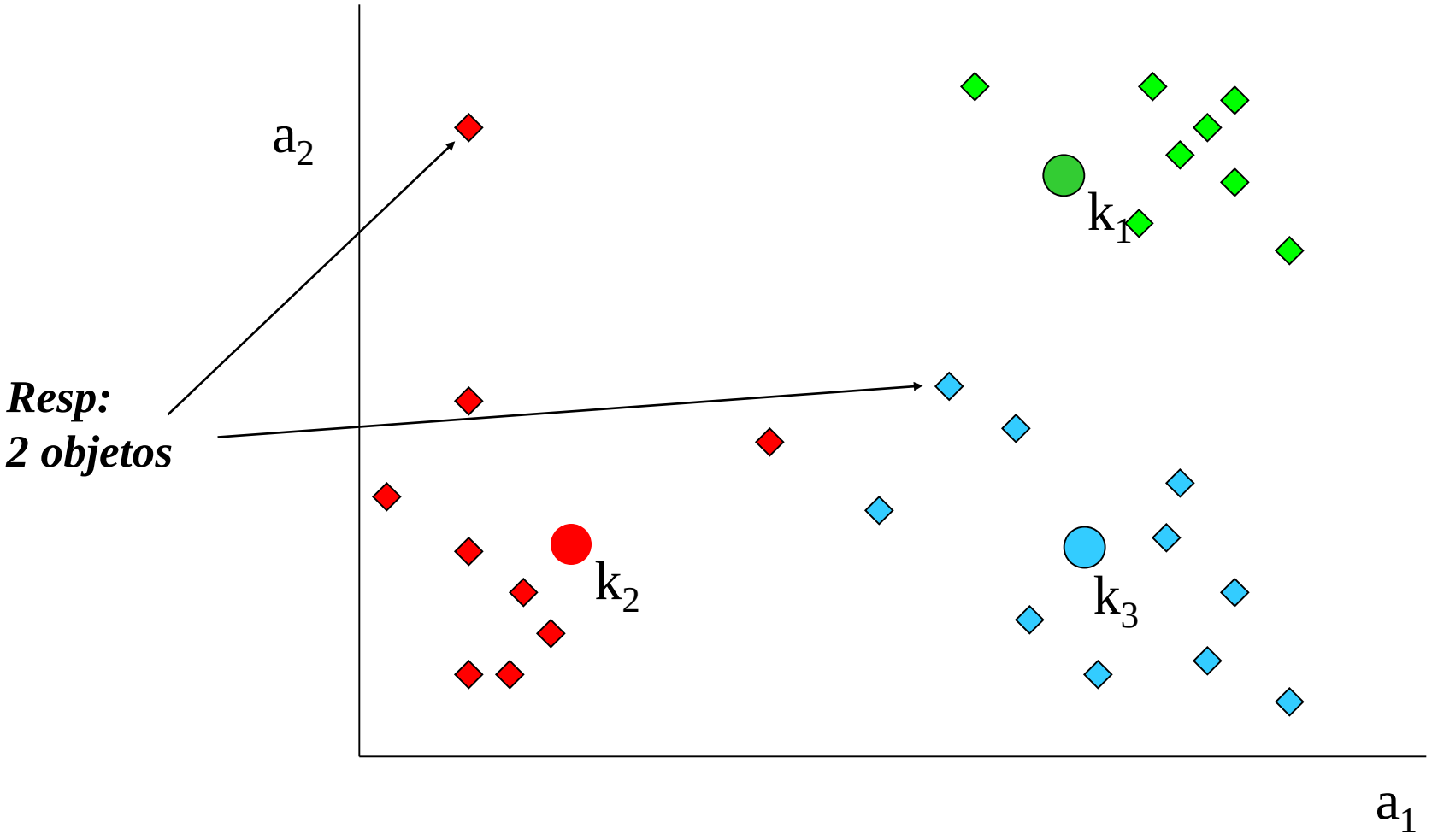
Re-atribuir objetos aos clusters de centróides mais próximos

Quais objetos mudarão de cluster?



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

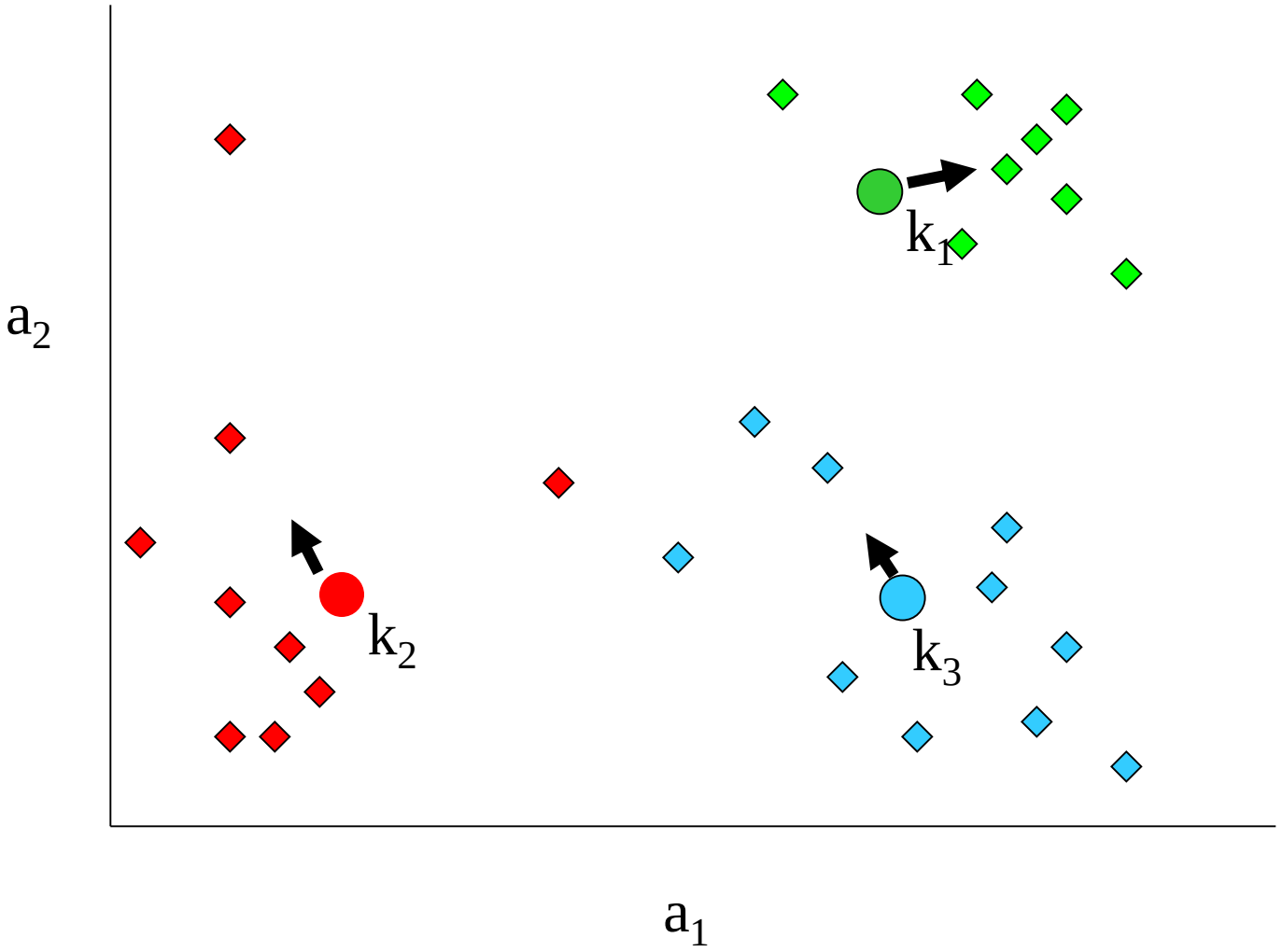
k-Means:



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

k-Means:

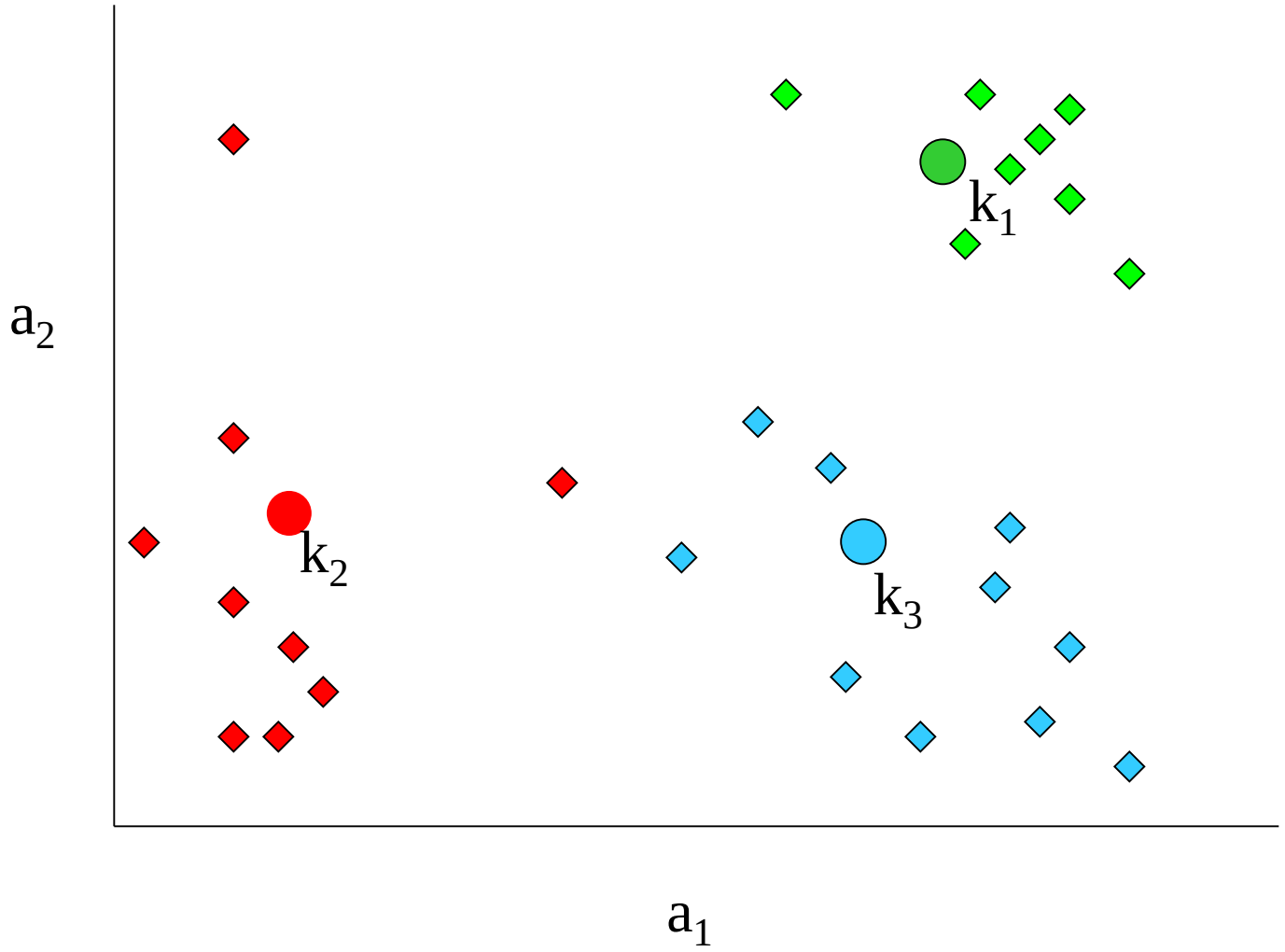
re-calcular vetores médios



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

k-Means:

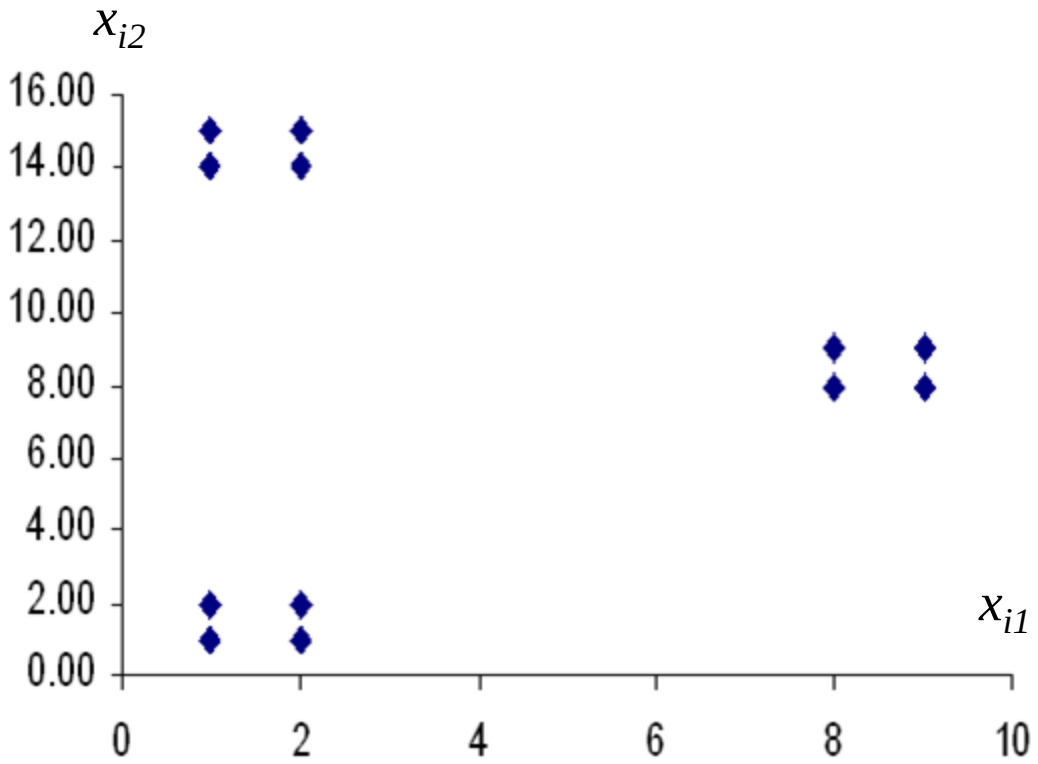
mover centros dos clusters…



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

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



- Executar k-means com k=3 a partir dos protótipos [6 6], [4 6] e [5 10]

K-Means sob Perspectiva de Otimização

- Algoritmo minimiza a seguinte função objetivo:
  - **SSE** = *Sum of Squared Erros* (**variâncias intra-cluster)**

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

onde d = Euclidiana e $\bar{\mathbf{x}}_c$ é o centróide do c-ésimo grupo:

$$\bar{\mathbf{x}}_c = \frac{1}{\left|\mathbf{C}_c\right|} \sum_{\mathbf{x}_j \in \mathbf{C}_c} \mathbf{x}_j$$

K-Means sob a Perspectiva de Otimização:

- Assumamos:
  - conjunto de objetos $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$
  - conjunto de k centróides quaisquer $\{\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, ..., \bar{\mathbf{x}}_k\}$
- Podemos reescrever o critério SSE de forma equivalente como:

$$J = \sum_{j=1}^{N}\sum_{c=1}^{k} \mu_{cj} \left\| \mathbf{x}_j - \bar{\mathbf{x}}_c \right\|^2 \; ; \; \sum_{c=1}^{k} \mu_{cj} = 1 \; \forall j \; ; \; \mu_{cj} \in \{0,1\}$$

- Desejamos minimizar J com respeito a $\{\bar{\mathbf{x}}_c\}$ e $\{\mu_{cj}\}$
- Pode-se fazer isso via um procedimento iterativo (2 passos):

a) Fixar $\{\bar{\mathbf{x}}_c\}$ e minimizar J com respeito a $\{\mu_{cj}\}$ (**E**)

b) Minimizar J com respeito a $\{\bar{\mathbf{x}}_c\}$, fixando-se $\{\mu_{cj}\}$ (**M**)

K-Means sob a Perspectiva de Otimização:

$$J = \sum_{j=1}^{N}\sum_{c=1}^{k} \mu_{cj} \left\| \mathbf{x}_j - \bar{\mathbf{x}}_c \right\|^2 \; ; \; \sum_{c=1}^{k} \mu_{cj} = 1 \;\; \forall j \; ; \; \mu_{cj} \in \{0,1\}$$

a) Fixar $\{\bar{\mathbf{x}}_c\}$ e minimizar J com respeito a $\{\mu_{cj}\}$ (**Passo E**)

- Termos envolvendo diferentes j são independentes
- Logo, pode-se otimizá-los separadamente
- $\mu_{cj} = 1$ para c que fornece o menor valor do erro quadrático

* **Atribuir $\mu_{cj} = 1$ para o grupo mais próximo**.

b) Minimizar J com respeito a $\{\bar{\mathbf{x}}_c\}$, fixando-se $\{\mu_{cj}\}$ (**Passo M**)

- Derivar J com respeito a cada $\bar{\mathbf{x}}_c$ e igualar a zero:

$$\nabla_{\bar{\mathbf{x}}_c} J = \sum_{j=1}^{N} \mu_{cj} \nabla_{\bar{\mathbf{x}}_c} \left[\left(\mathbf{x}_j - \bar{\mathbf{x}}_c \right)^T \left(\mathbf{x}_j - \bar{\mathbf{x}}_c \right) \right] = 2 \sum_{j=1}^{N} \mu_{cj} \left(\bar{\mathbf{x}}_c - \mathbf{x}_j \right) = \mathbf{0} \;\; \rightarrow \;\; \bar{\mathbf{x}}_c = \frac{\sum_{j=1}^{N} \mu_{cj} \mathbf{x}_j}{\sum_{j=1}^{N} \mu_{cj}}$$

Alternativas para Inicialização

- Múltiplas Execuções (inicializações aleatórias):
  - funciona bem em muitos problemas.
  - em bases de dados complexas, pode demandar um no. enorme de execuções.
  - em particular para no. de grupos grande.
    - especialmente porque k é, em geral, desconhecido
- Agrupamento Hierárquico:
  - agrupa-se uma amostra dos dados
  - tomam-se os centros da partição com k grupos

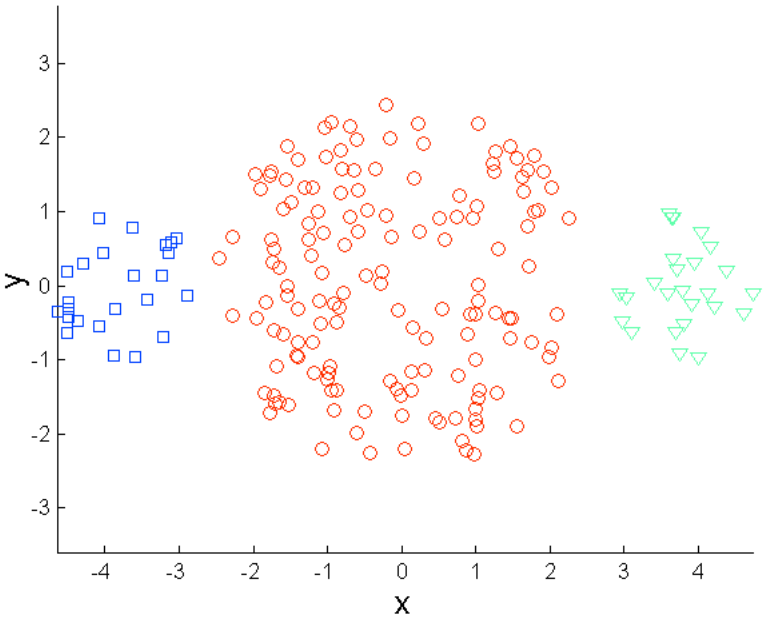
Alternativas para Inicialização

- Seleção "Informada" :
  - toma-se o 1º protótipo como um objeto aleatório
    - ou como o centro dos dados (*grand mean*)
  - sucessivamente escolhe-se o próximo protótipo
    - como o objeto mais distante dos protótipos correntes
  - **Nota**: para reduzir o esforço computacional e minimizar a probabilidade de seleção de outliers
    - processa-se apenas uma amostra dos dados
- Busca Guiada:
  - **X-means**, **k-means evolutivo**, ...

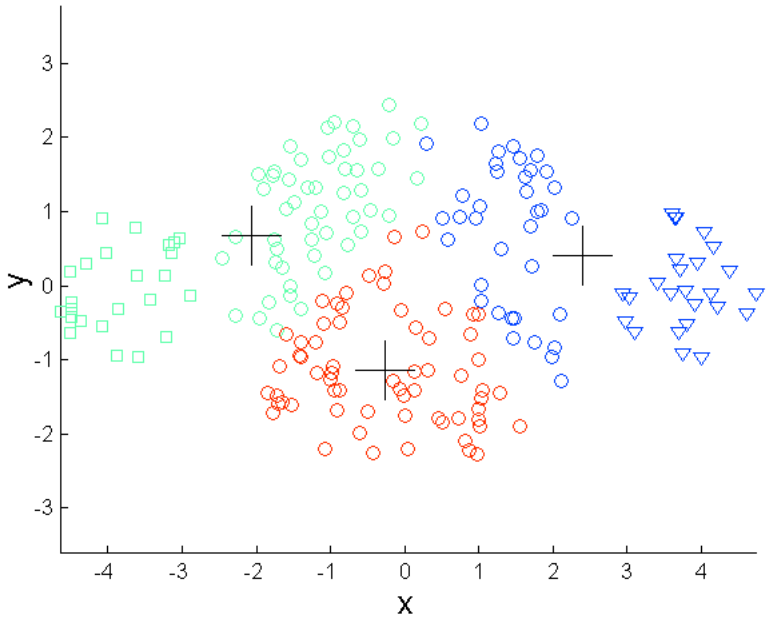
Discussão

- k-means é mais susceptível a problemas quando clusters são de diferentes
  - Tamanhos
  - Densidades
  - Formas não-globulares

Formas diferentes

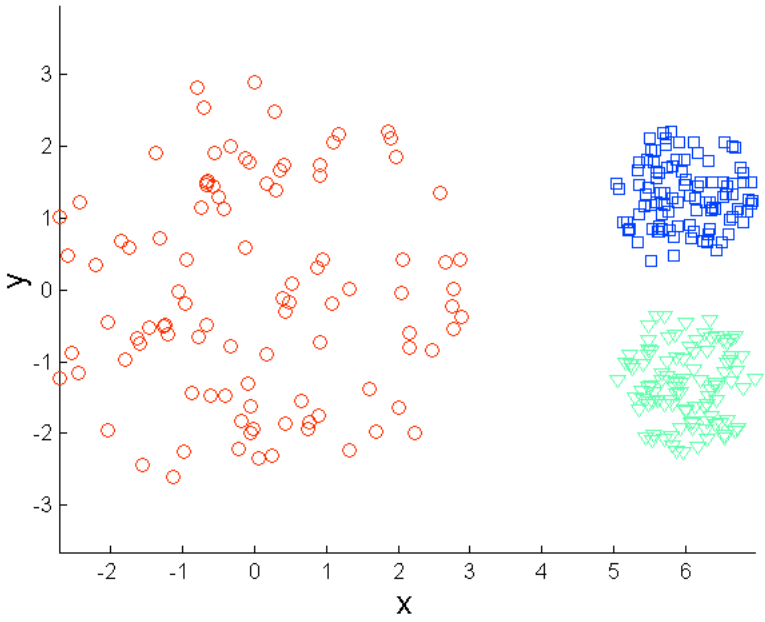


Original Points

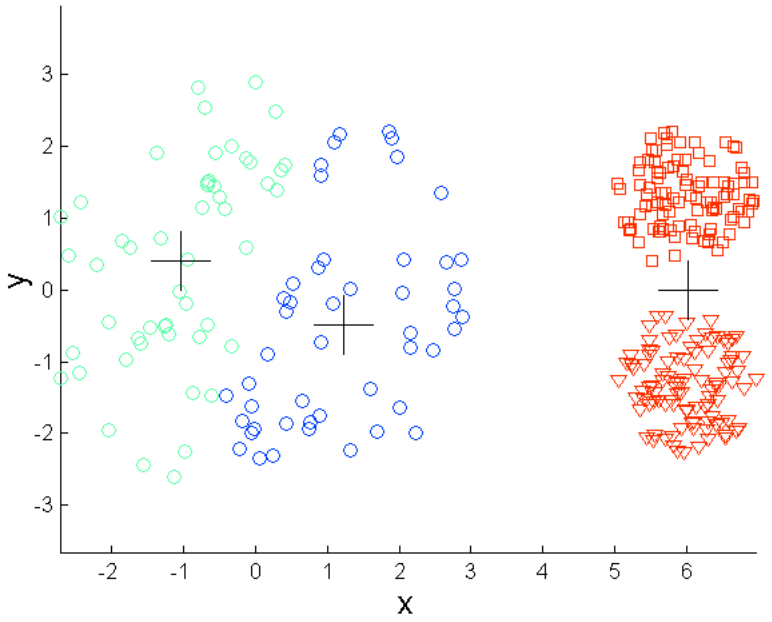


K-means (3 Clusters)

Densidades diferentes



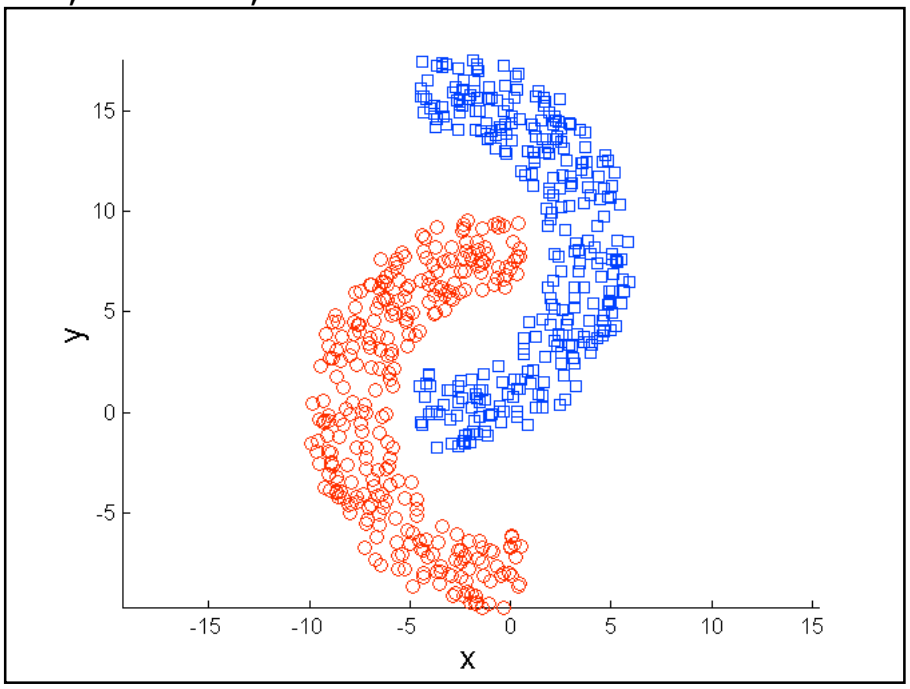
Original Points

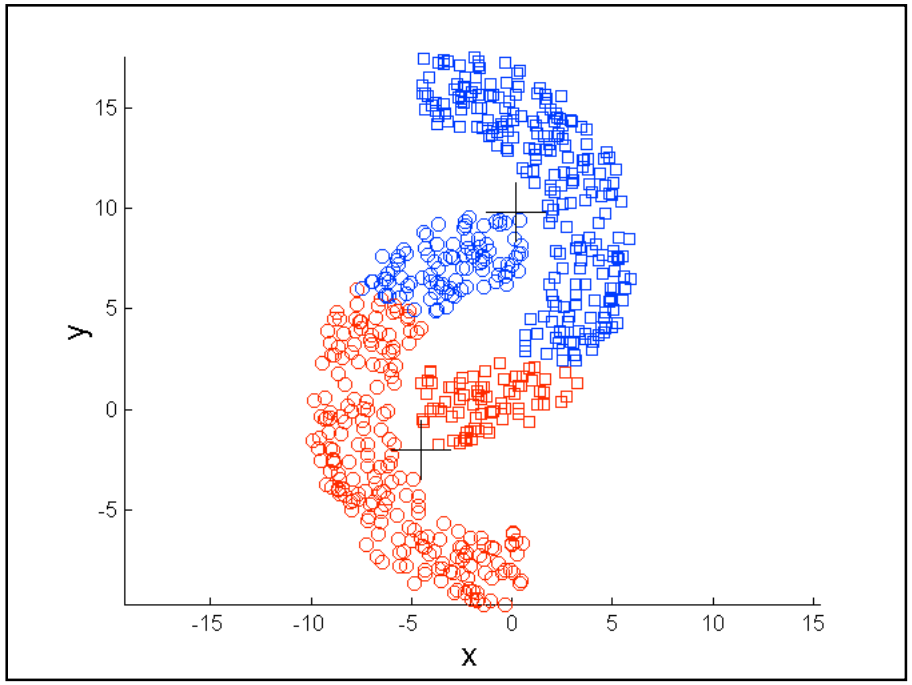


K-means (3 Clusters)

Formas Não-Globulares

Tan, Steinbach, Kumar

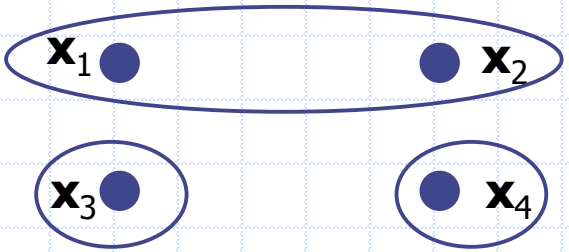




- **Nota**: na prática, esse problema em geral não é crítico, i.e., há pouco interesse na maioria das aplicações de mundo real
  - Grandes BDs (muitos objetos & atributos) e necessidade de interpretação dos resultados (e.g. segmentação de mercado...)

Manipulando Grupos Vazios

- k-means pode gerar **grupos vazios**
  - Por inicialização em pontos "dominados" do espaço
    - protótipos não representativos: nenhum objeto mais próximo
    - inicialização como objetos ao invés de pontos aleatórios resolve
  - Pela inicialização de grupos
    - cujos protótipos são não representativos; por exemplo:



Grupos iniciais

***k* =3**

  - Ao longo das iterações

Manipulando Grupos Vazios

- Estratégias para contornar o problema:
  - Eliminar os protótipos não representativos (reduz k)
    - viável se o número inicial de grupos, k, puder ser reduzido
    - pode ser útil para ajustar valores superestimados de k
  - Substituir cada protótipo não representativo (mantém k)
    - pelo objeto que mais contribui para o SSE da partição
    - por um dos objetos do grupo com maior MSE
      - visa dividir o grupo com maior erro quadrático médio
    - **Nota**: a execução do algoritmo prossegue após a substituição

Resumo do k-means

Vantagens

- Simples e intuitivo
- Complexidade computacional **linear** em todas as variáveis críticas: O(N n k)
  - quadrático se n ≈ N ...
- Eficaz em muitos cenários de aplicação e produz resultados de interpretação 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 rígida**, ou seja, sem sobreposição)
- Limitado a atributos numéricos
- Sensível a *outliers*

Algumas Variantes do k-means

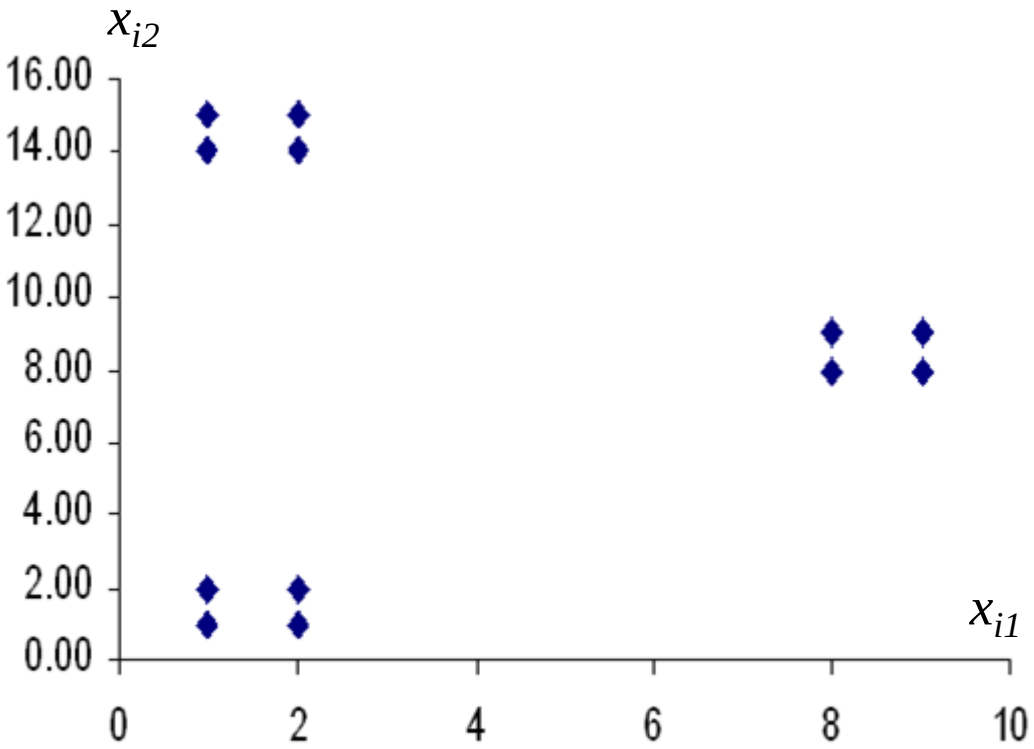
- **K-medianas:** Substituir as médias pelas medianas
  - Média de 1, 3, 5, 7, 9 é **5**
  - Média de 1, 3, 5, 7, 1009 é **205**
  - Mediana de 1, 3, 5, 7, 1009 é **5**
  - **Vantagem:** menos sensível a outliers*
  - **Desvantagem:** implementação mais complexa
    - cálculo da mediana em cada atributo...
- Pode-se mostrar que minimiza a soma das **distâncias de Manhattan** dos objetos aos centros (medianas) dos grupos

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 nº. de objetos (N)

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



- Executar k-medóides para k=3 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...

- J poderia ser utilizada 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 e, portanto, variável ?
- Considere, por exemplo, que as partições são geradas a partir de múltiplas execuções do algoritmo:
  - com protótipos iniciais aleatórios
  - com no. variável de grupos $k \in [k_{min}, k_{max}]$

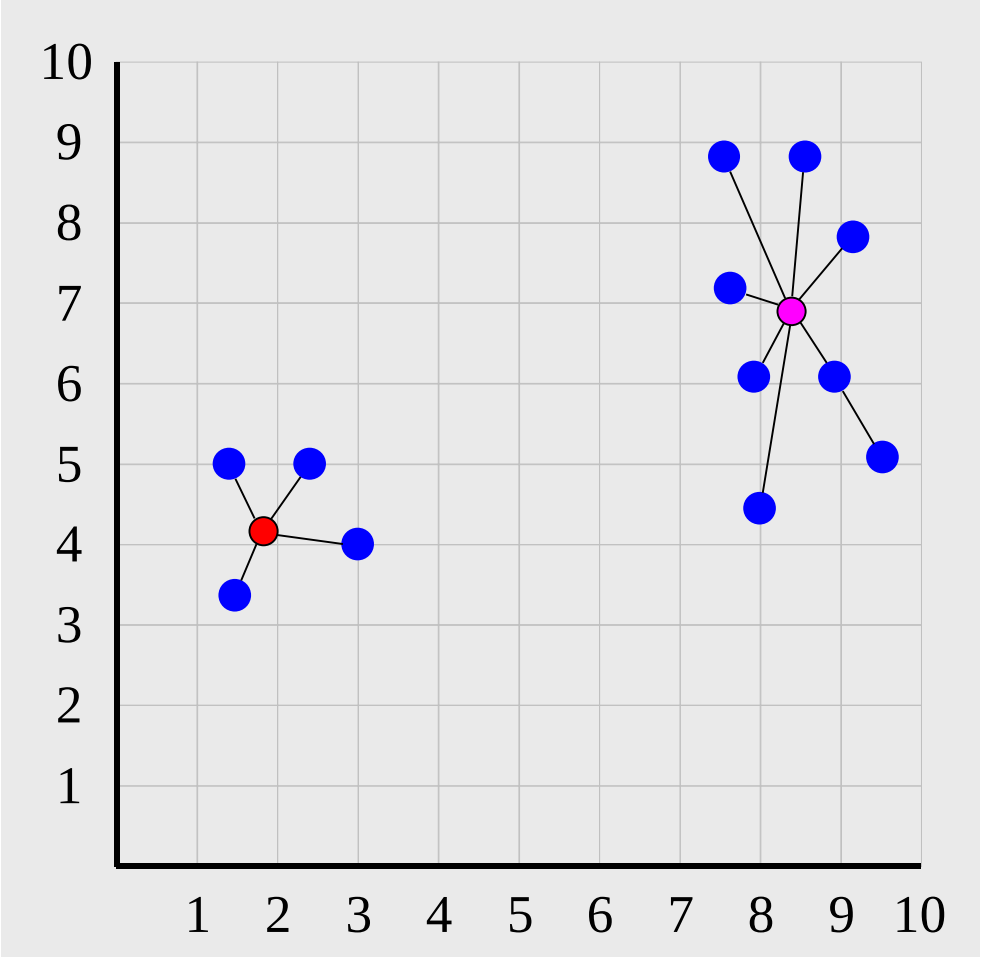
Questão...

Considere múltiplas execuções ordenadas de k-means, com uso de J:

Erro Quadrático:

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

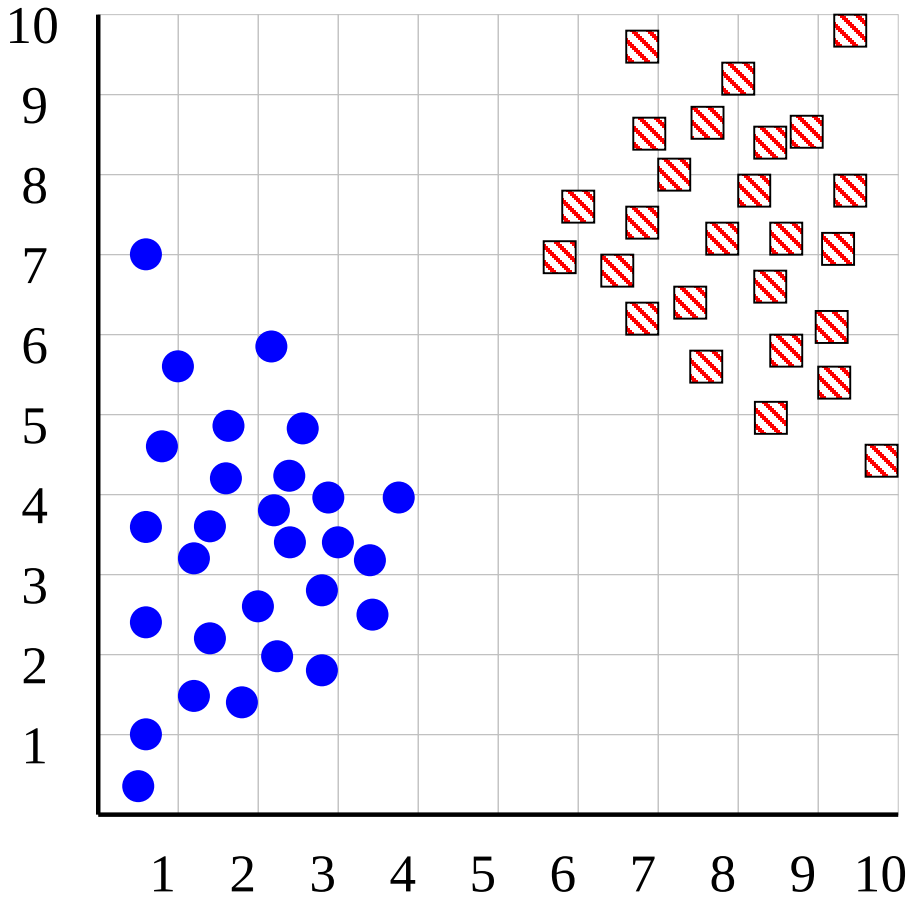
Função Objetivo



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

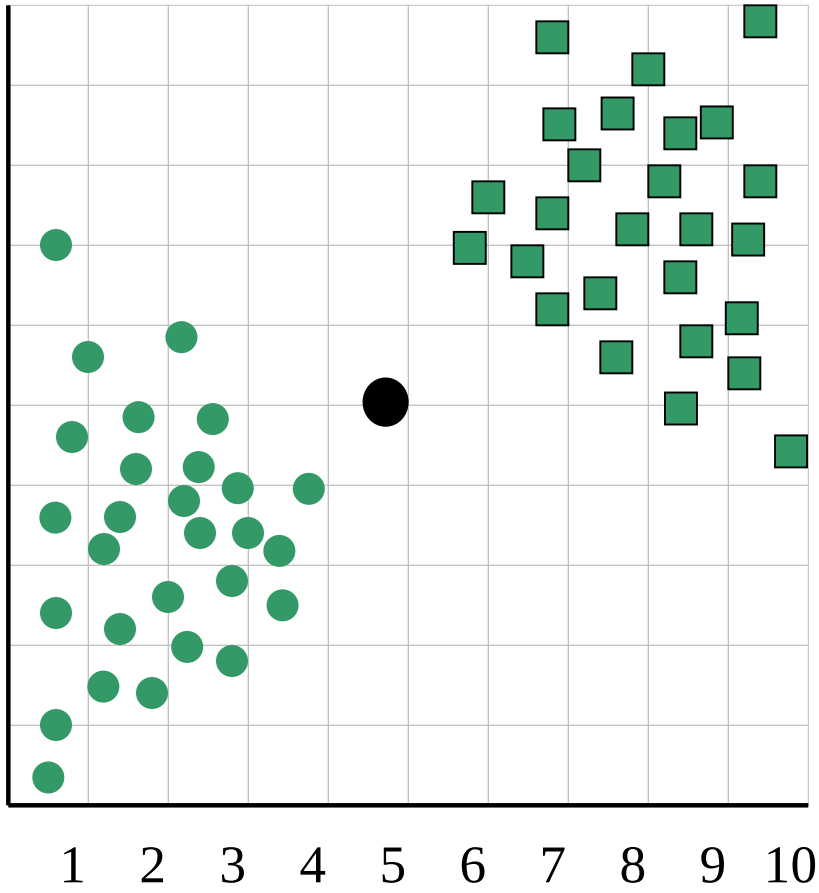
Questão...

- Considere o seguinte exemplo:

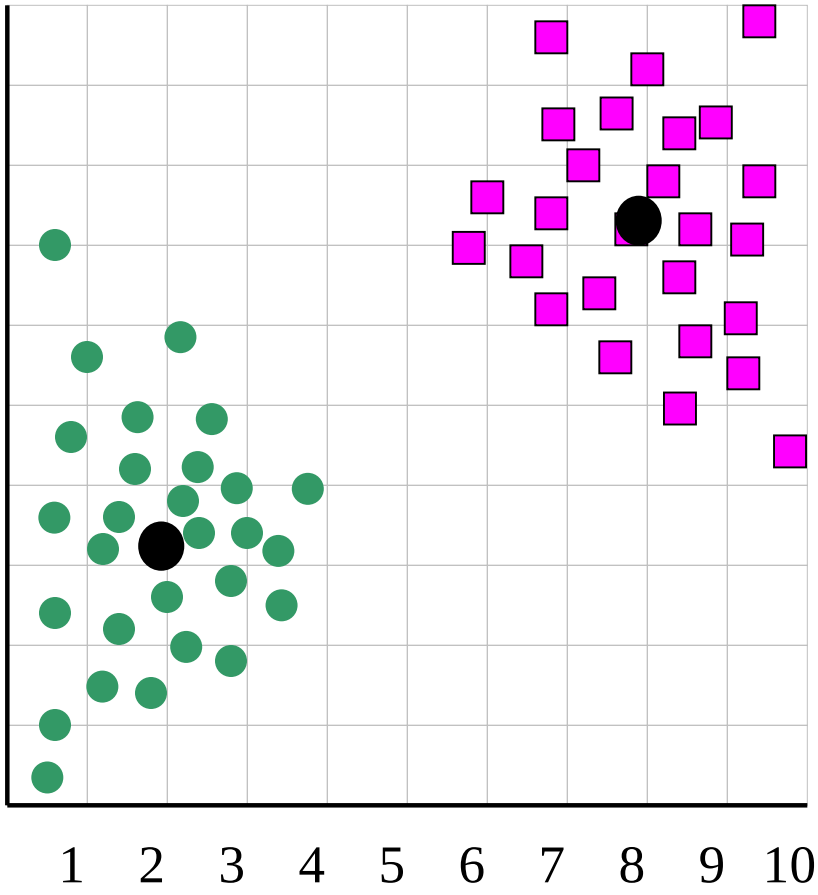


Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

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

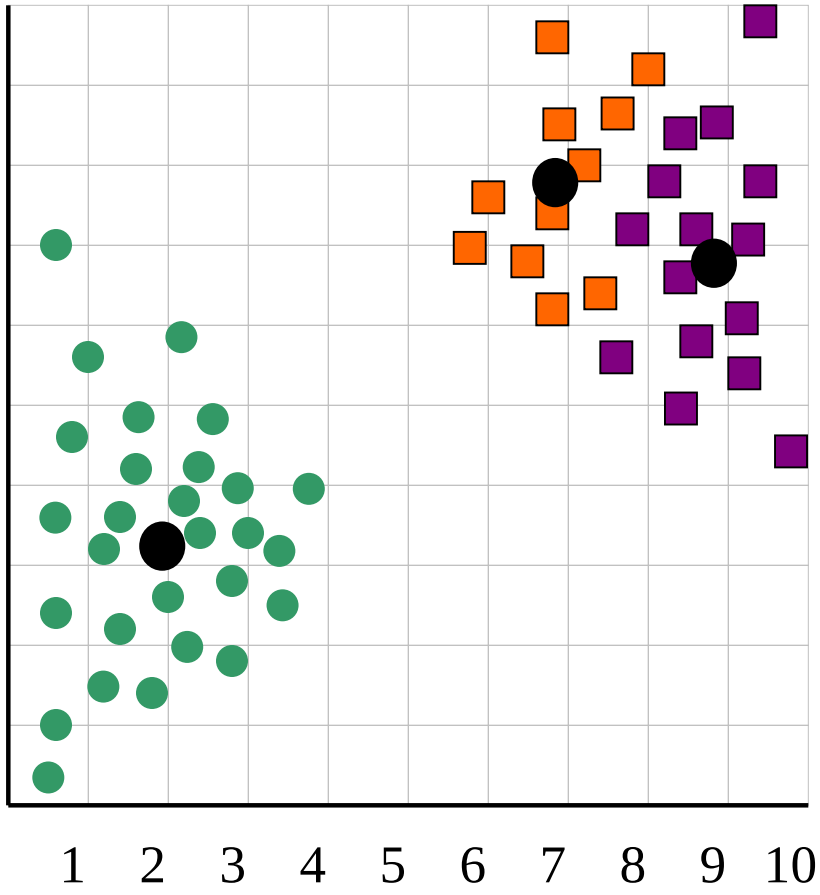


Para $k = 2$, o valor da função objetivo é 173,1



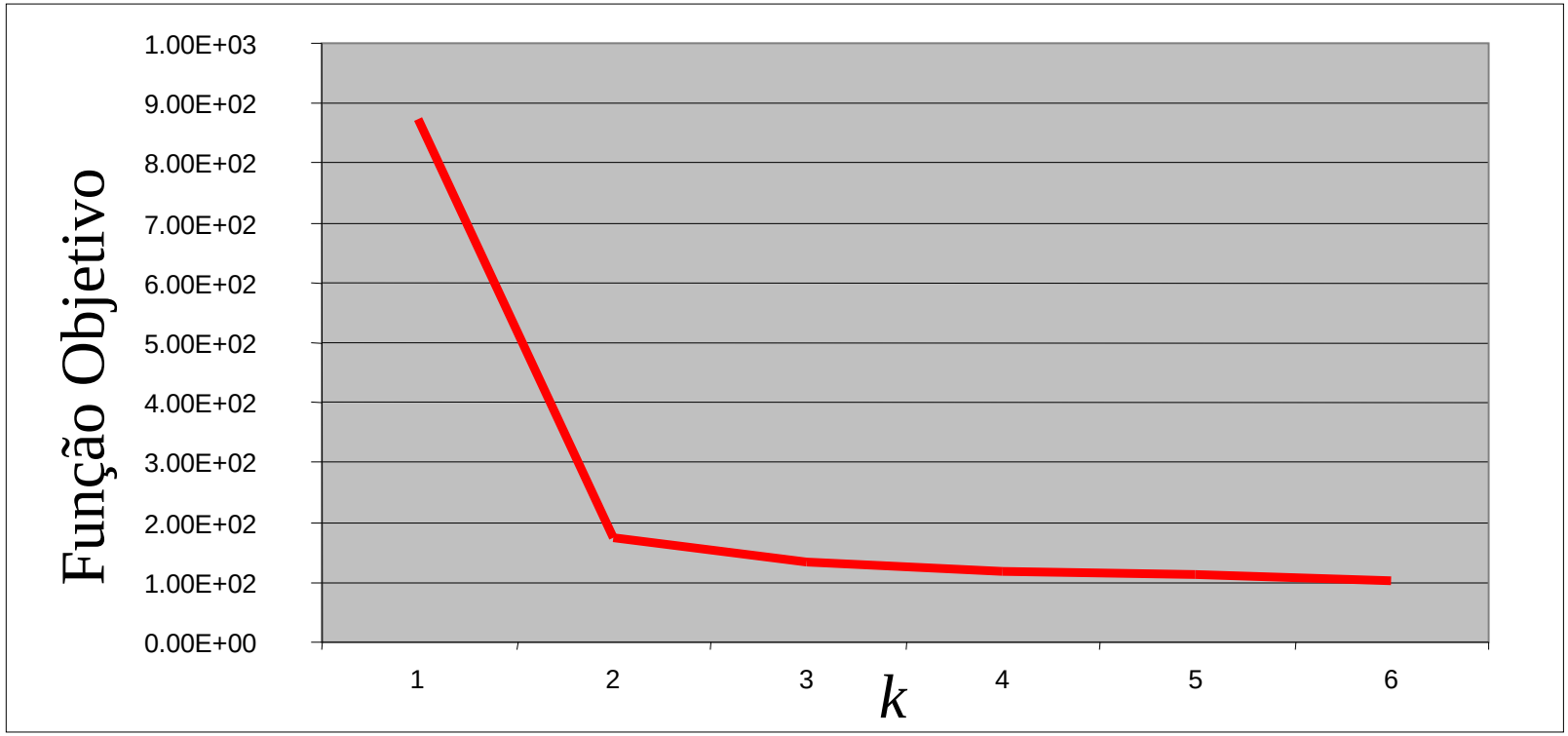
Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

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



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

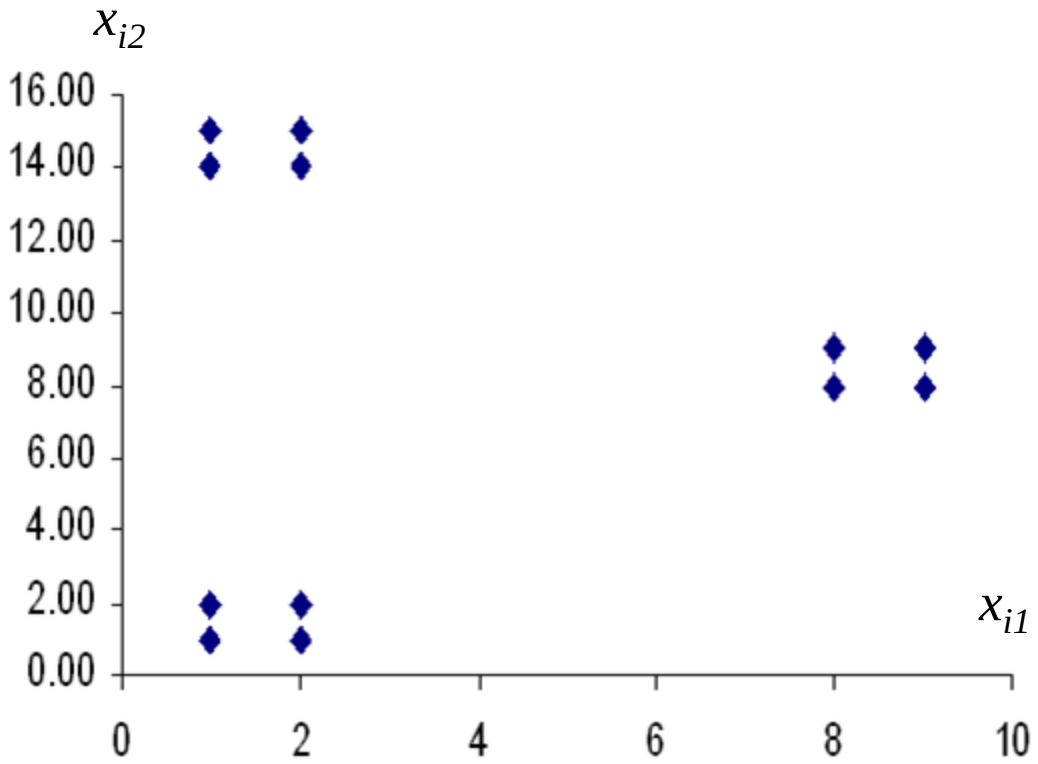
Podemos então repetir este procedimento e plotar os valores da função objetivo J para k =1,…,6, ... e tentar identificar um "joelho" :



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

Exercício

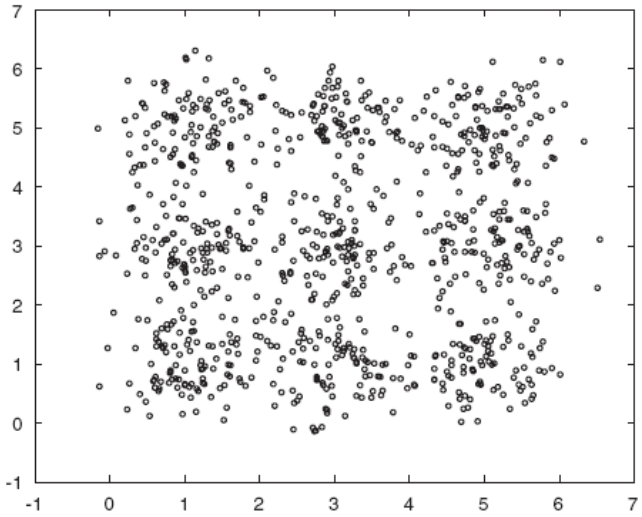
Objeto $\mathbf{x}_i$	x_{i1}	x_{i2}
1	1	2
2	2	1
3	1	1
4	2	2
5	**8**	**9**
6	**9**	**8**
7	**9**	**9**
8	**8**	**8**
9	1	15
10	2	15
11	1	14
12	2	14

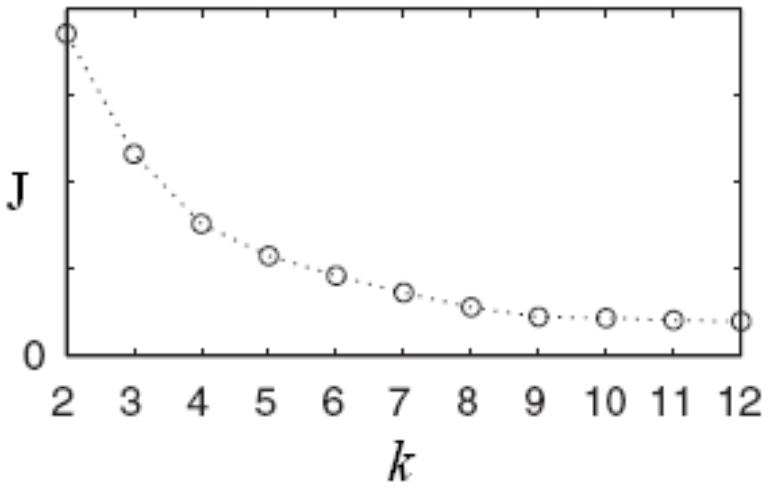


- Executar k-means com k=2 até k=5 e representar graficamente J=f(k)

Questão...

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





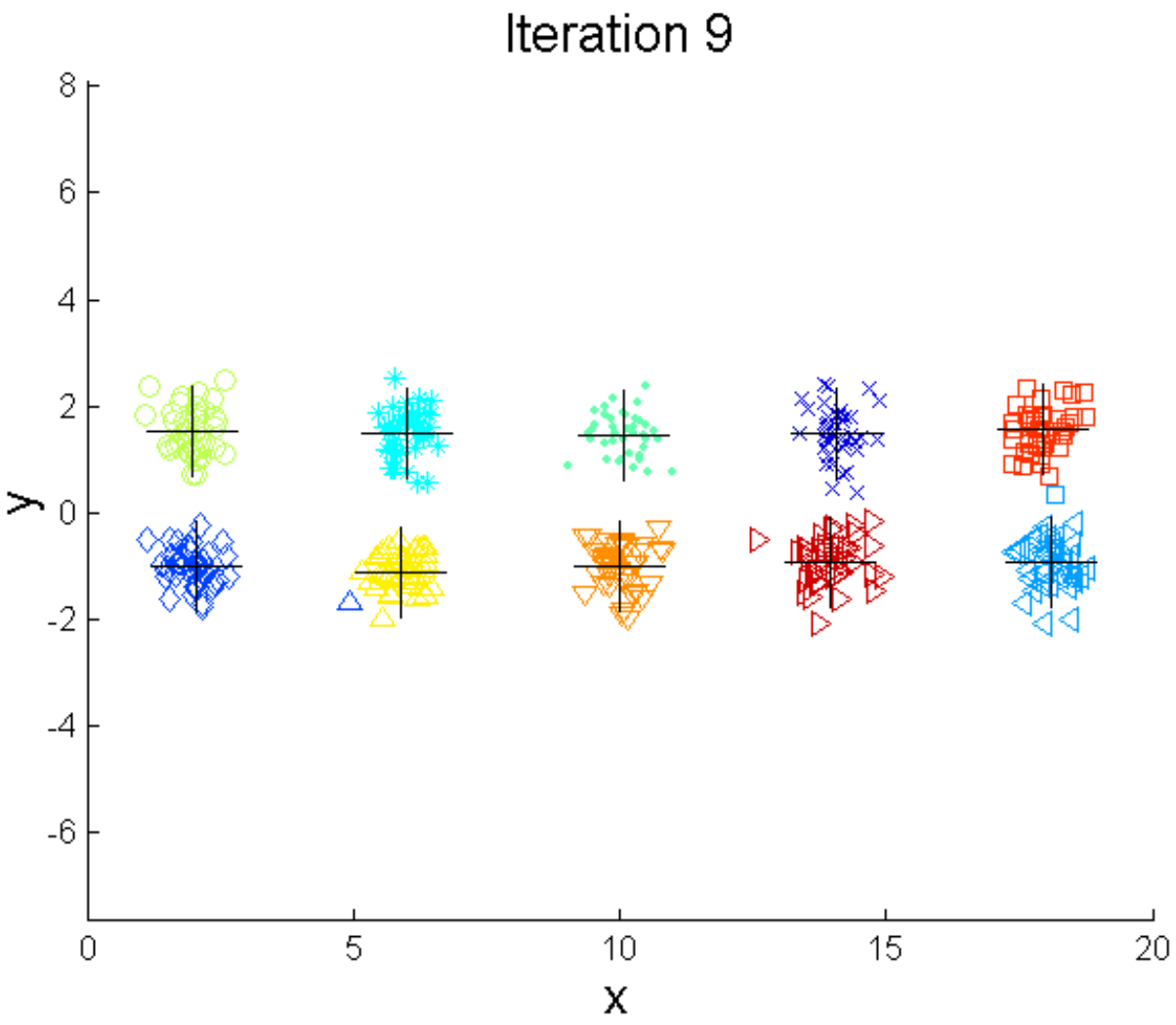
Bisecting K-means (BKM)

- Variante que pode produzir uma partição ou uma hierarquia

1: Initialize the list of clusters to contain the cluster containing all points.
2: **repeat**
3: Select a cluster from the list of clusters
4: **for** $i = 1$ to *number_of_iterations* **do**
5: Bisect the selected cluster using basic K-means
6: **end for**
7: Add the two clusters from the bisection with the lowest SSE to the list of clusters.
8: **until** Until the list of clusters contains K clusters

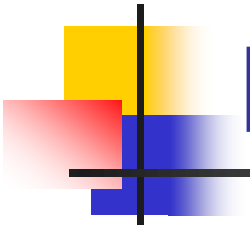
$$SSE(\mathbf{C}_i) = \sum_{\mathbf{x}_j \in \mathbf{C}_i} d(\mathbf{x}_j, \bar{\mathbf{x}}_i)^2$$

Bisecting K-means Example



Bisecting K-Means

- Fazendo K = N (no. total de objetos) no passo 8 do algoritmo, obtemos uma hierarquia completa
- No passo 3, a seleção do grupo a ser bi-seccionado pode ser feita de diferentes maneiras
  - Utiliza-se algum critério de avaliação de qualidade dos grupos, para eleger o "pior". Por exemplo:
    - Diâmetro máximo (sensível a outliers)
    - SSE normalizado pelo no. de objetos do grupo (mais robusto)
    - Critérios de avaliação de grupos individuais que consideram os objetos nos demais grupos (veremos posteriormente no curso)



Referências

- Jain, A. K. and Dubes, R. C., Algorithms for Clustering Data, Prentice Hall, 1988
- Kaufman, L., Rousseeuw, P. J., Finding Groups in Data – An Introduction to Cluster Analysis, Wiley, 2005.
- Tan, P.-N., Steinbach, M., and Kumar, V., *Introduction to Data Mining*, Addison-Wesley, 2006
- Wu, X. and Kumar, V., *The Top Ten Algorithms in Data Mining*, Chapman & Hall/CRC, 2009
- D. Steinley, *K-Means Clustering: A Half-Century Synthesis*, British J. of Mathematical and Stat. Psychology, V. 59, 2006