



# Machine Learning

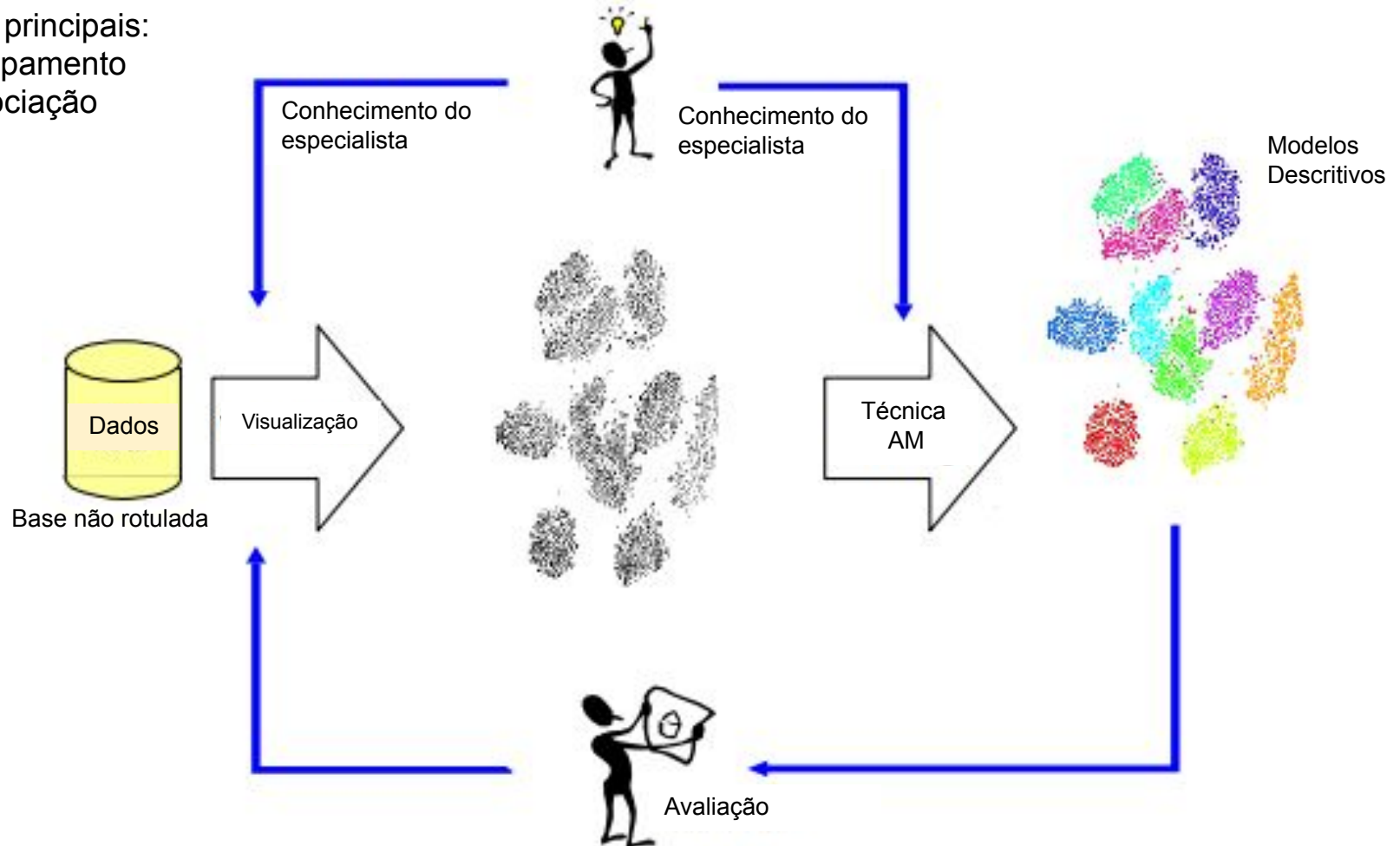
## Aprendizagem não Supervisionada

Tarefa: agrupamento (clustering)

## Aprendizagem de Máquina (AM) - Abordagem Não Supervisionada

Tarefa principais:

- Agrupamento
- Associação





# Aprendizagem Não-Supervisionada

- Quando os dados ou amostras de um problema não estão rotulados.
- Busca descobrir padrões nos dados.
- Tarefa principal desta abordagem:
  - **Agrupamento:** consiste em descobrir grupos (clusters) compostos por instâncias (exemplos) similares segundo algum critério.

# Aprendizagem Não-Supervisionada

## ■ Motivação:

### □ Rotulação automática

- rotular bases de dados é uma tarefa de alto custo, demanda um especialista no problema.
- é comum não se ter conhecimento das classes do problema quando do planejamento de modelos preditivos.

### □ Descoberta de padrões: utilizada na mineração de dados que busca transformar dados brutos em conhecimento.

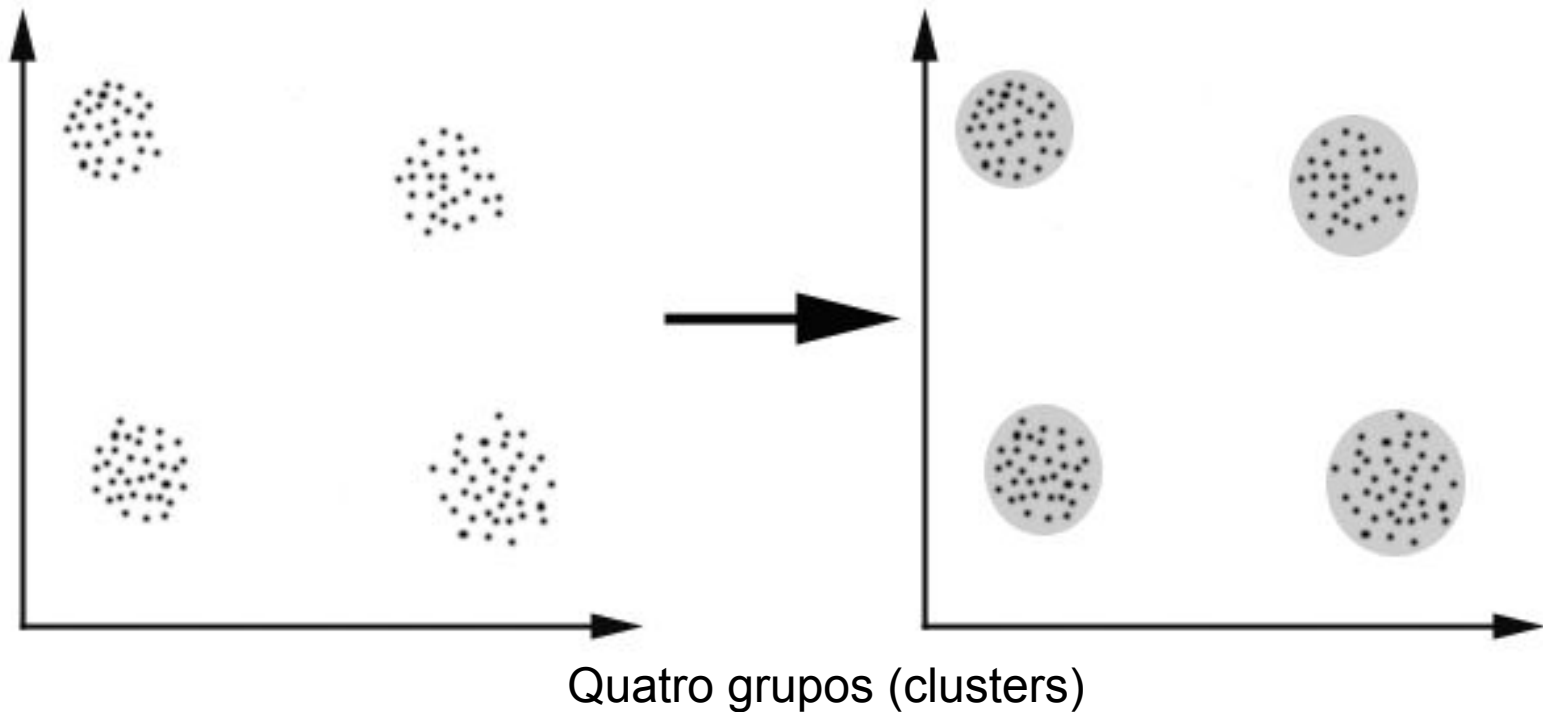


# Aprendizagem Não-Supervisionada

- Um agrupamento pode ser utilizado como um passo antes da criação de um classificador:
  - Dada uma base de dados não rotulada, pode-se utilizar a aprendizagem não-supervisionada para fazer uma pré-classificação, e então treinar um classificador de maneira supervisionada.

# Agrupamento (*Clustering*)

- Organização de objetos em grupos (clusters) segundo algum critério de similaridade.



# O que é um *Cluster*?

- Uma coleção de objetos que são similares entre si e diferentes dos objetos pertencentes a outros clusters.
- Encontrar clusters demanda uma medida de similaridade.
- Usualmente utiliza-se uma *distância*, o que caracteriza a forma mais comum de agrupamento:
  - *Distance-based Clustering*

# *k-Means Clustering*

- É a técnica mais simples de aprendizagem não supervisionada, e consiste, basicamente, nos seguintes passos:
  - Fixar  $k$  centróides (de maneira aleatória), um para cada grupo (ou *cluster*).
  - Associar cada indivíduo ao centróide mais próximo.
  - Recalcular os centróides com base nos indivíduos classificados.
- Critério de Parada: até que não ocorram mais mudanças nos centroids.

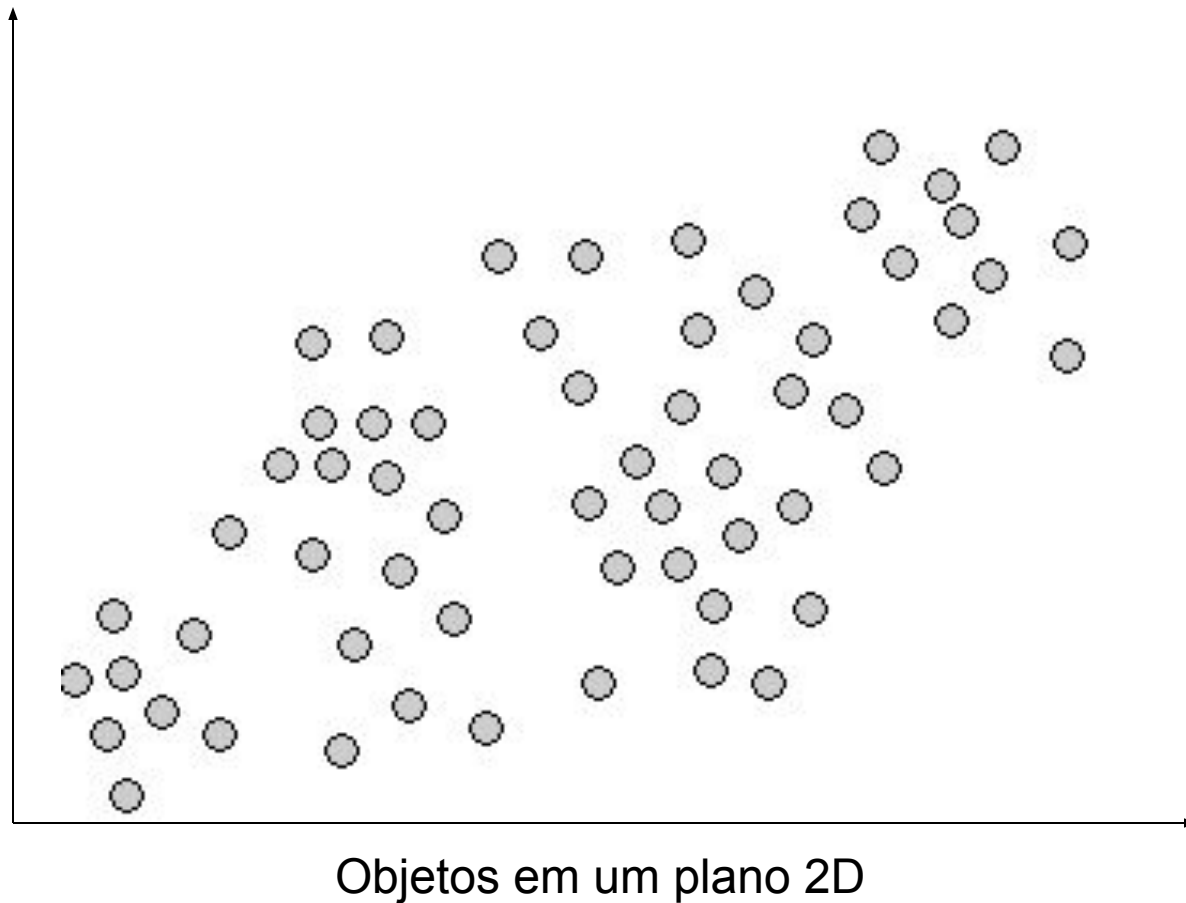




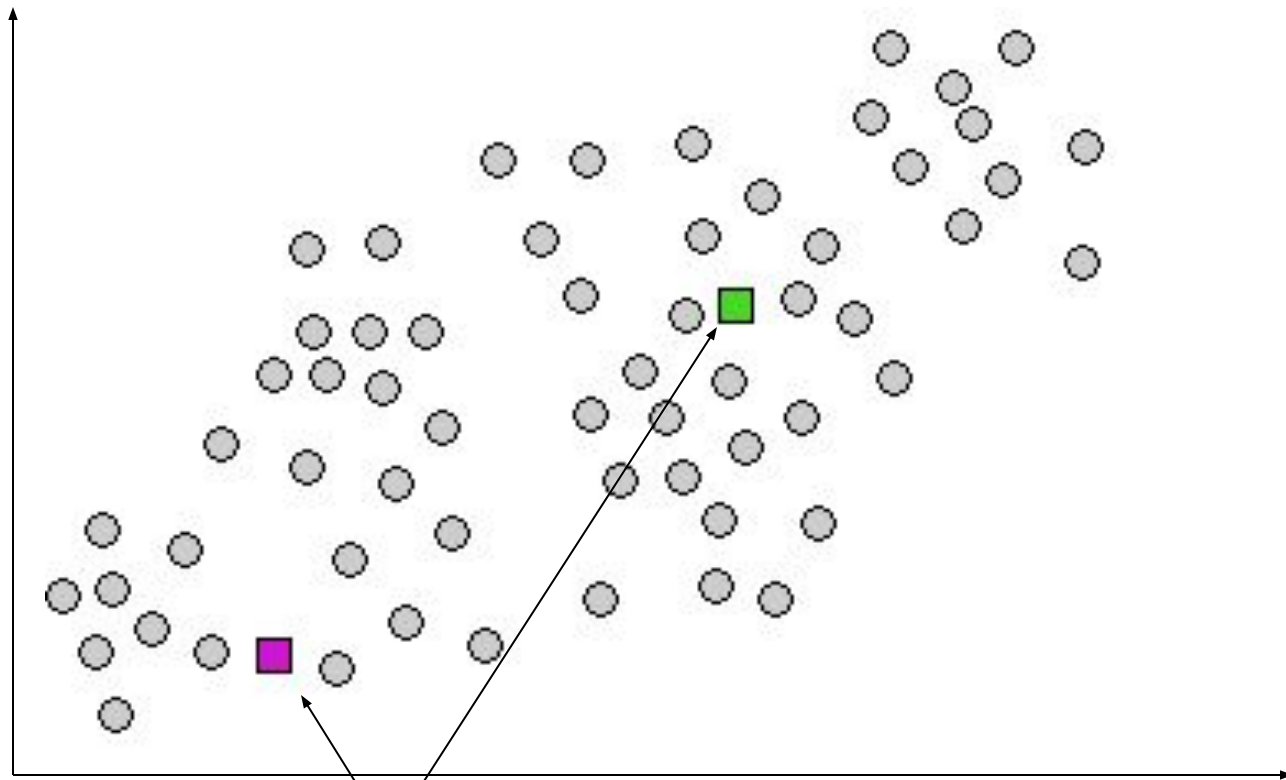
# Lógica do Algoritmo *k-Means*

1. Determinar os centróides
2. Atribuir a cada objeto do grupo o centróide mais próximo.
3. Após atribuir um centróide a cada objeto, recalcular os centróides.
4. Repetir os passos 2 e 3 até que os centróides não sejam modificados.

# *k*-Means – Um Exemplo

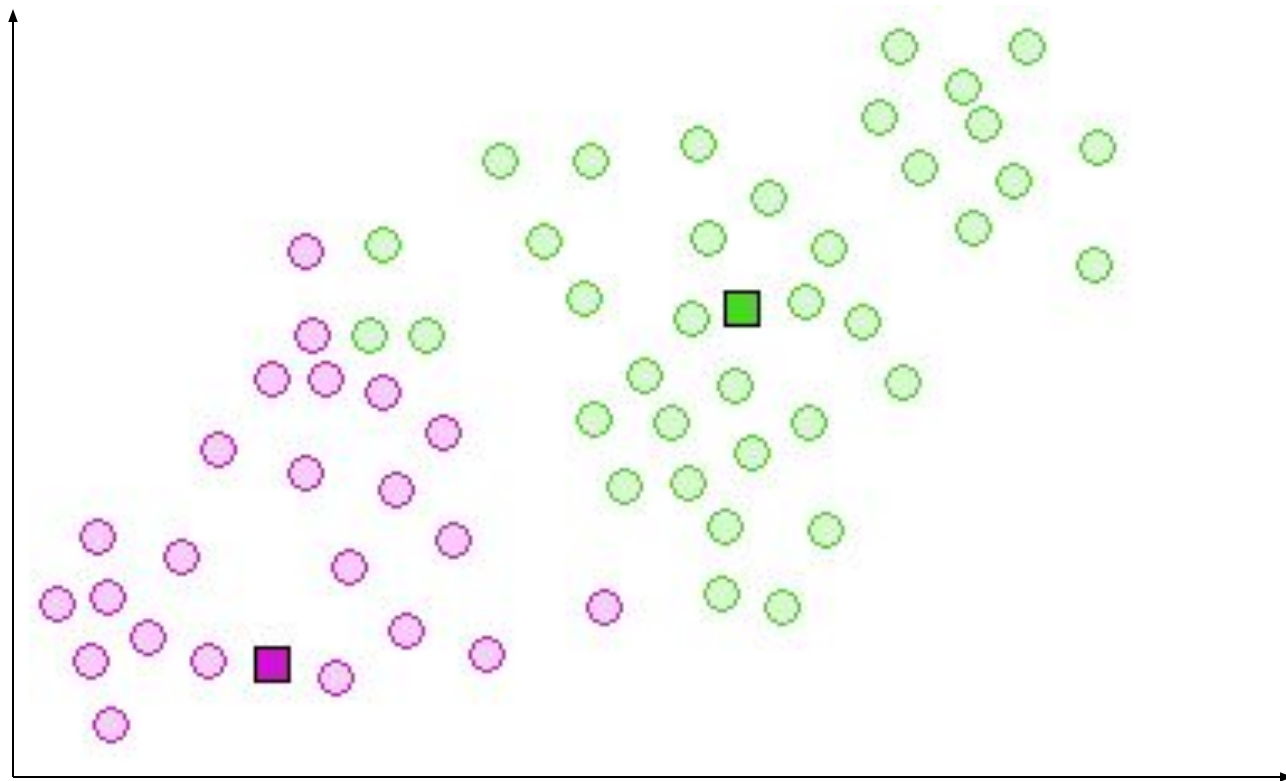


# *k*-Means – Um Exemplo



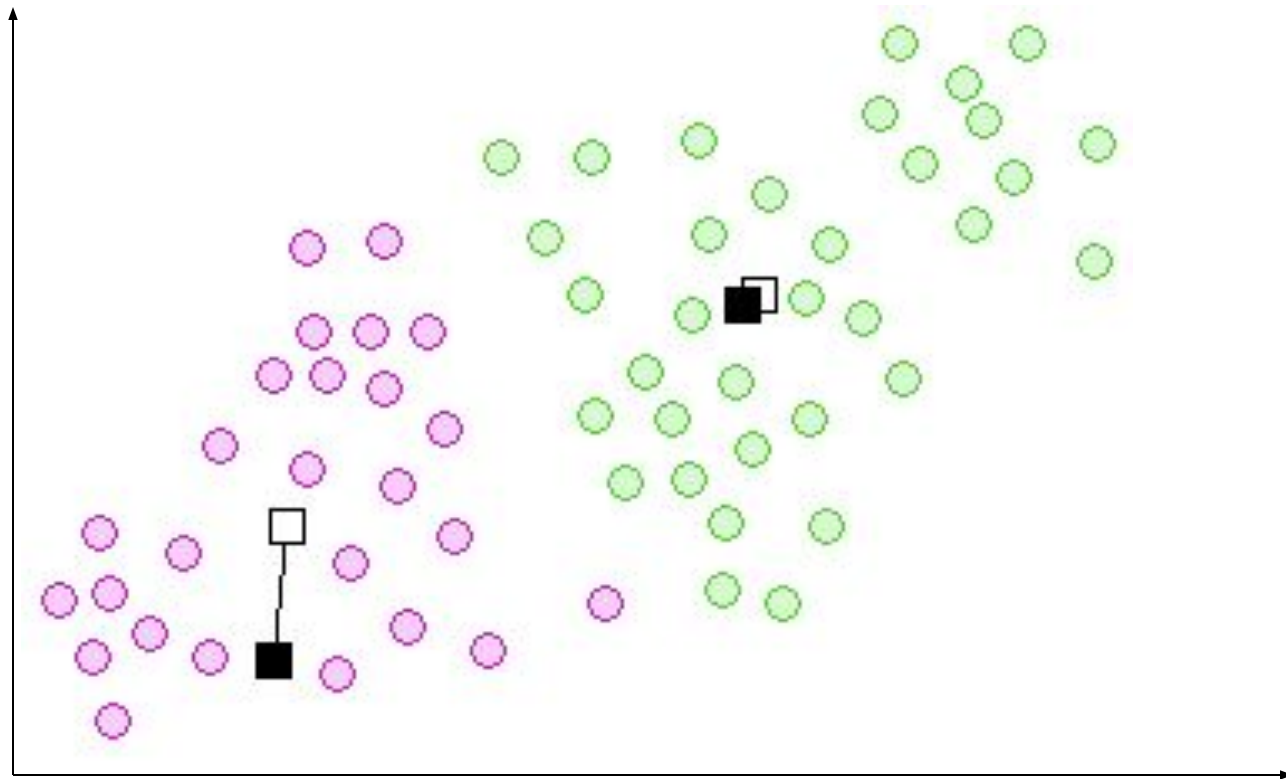
Passo 1: Centróides inseridos aleatoriamente

# *k-Means* – Um Exemplo



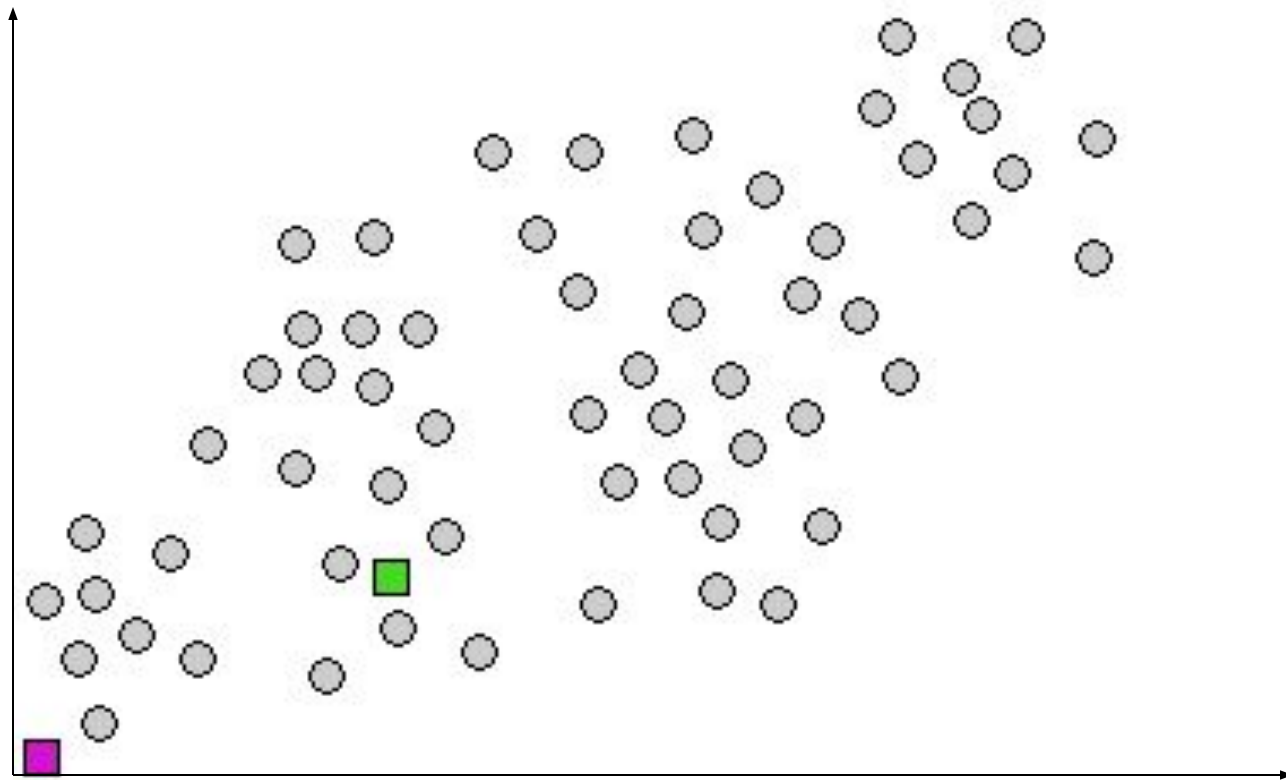
Passo 2: Atribuir a cada objeto o centróide mais próximo

# *k-Means* – Um Exemplo



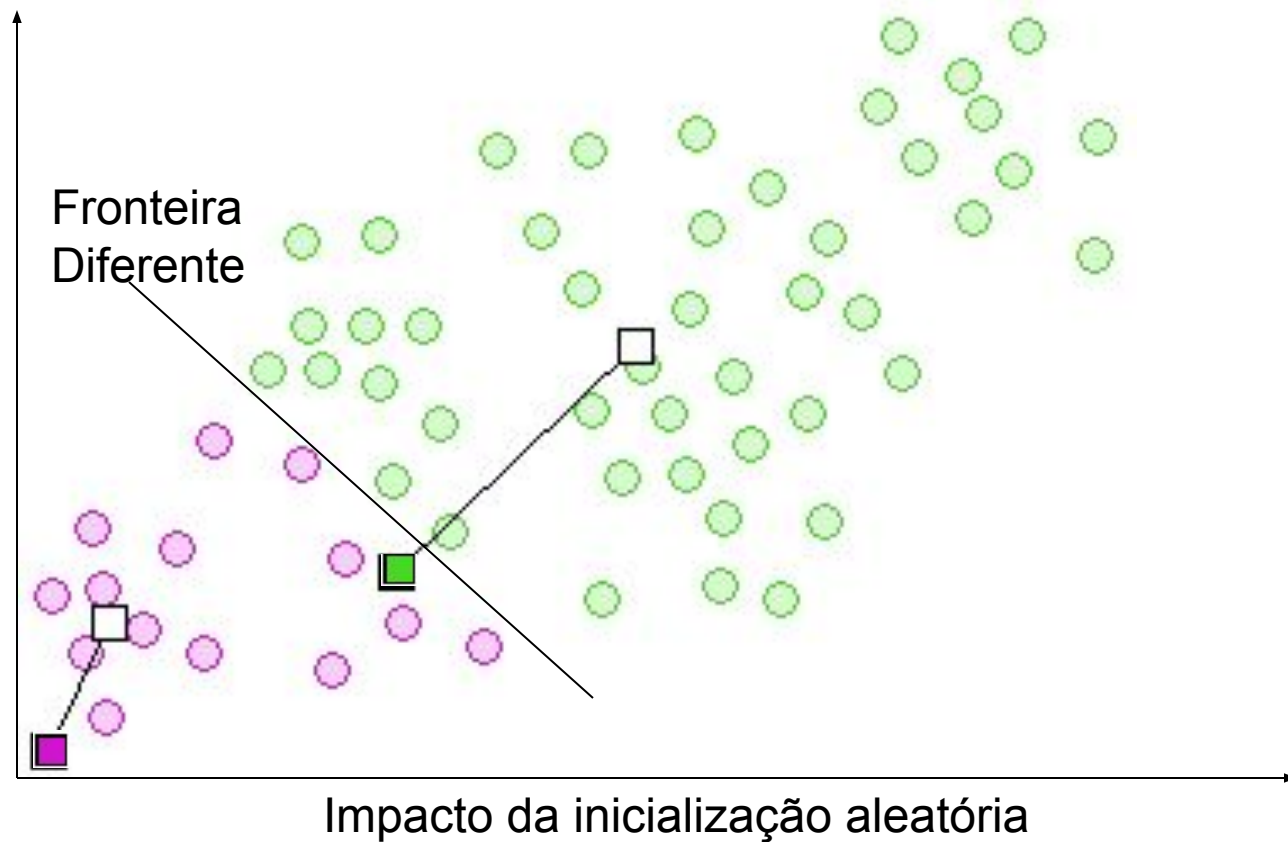
Passo 3: Recalcular os centróides

# *k*-Means – Um Exemplo



Impacto da inicialização aleatória.


# *k*-Means – Um Exemplo



# *k-Means* – Inicialização

- Impacto da inicialização
  - Quando se tem noção dos centróides, pode-se melhorar a convergência do algoritmo.
  - Execução do algoritmo várias vezes permite reduzir o impacto da inicialização aleatória.





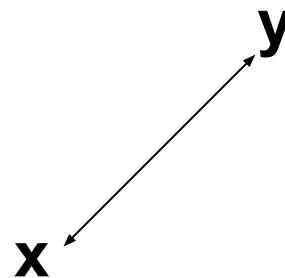
# *k-Means* – Simulador

[Utah University](#)

# Medidas de distância

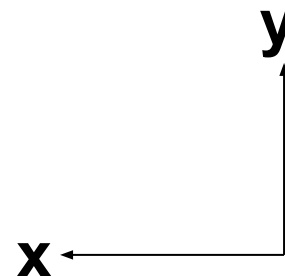
- Distância Euclidiana

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$



- Manhattan (City Block)

$$d = \sum_{i=1}^n |x_i - y_i|$$



# Medidas de distância

- Minkowski
  - Parâmetro  $r$ 
    - $r = 2$ , distância Euclidiana
    - $r = 1$ , City Block

$$d = \left( \sum_{i=1}^n (x_i - y_i)^r \right)^{1/r}$$

# Calculando Distâncias

## ■ Mahalanobis

- Leva em consideração as variações estatísticas dos pontos. Por exemplo, se  $x$  e  $y$  são dois pontos da mesma distribuição, com matriz de covariância  $C$ , a distância é dada pela equação

$$d = (x - y)' C^{-1} (x - y)^{\frac{1}{2}}$$

- Se a matriz  $C$  for uma matriz identidade, essa distância é igual a distância Euclidiana.



# CrITÉrios de OtimizaÇão

- O problema consiste em encontrar os *clusters* que minimizam/maximizam um dado critério.
- Alguns critérios de otimização:
  - Soma dos Erros Quadrados.
  - Critérios de Dispersão.
  - Índice Silhueta

# Soma dos Erros Quadrados

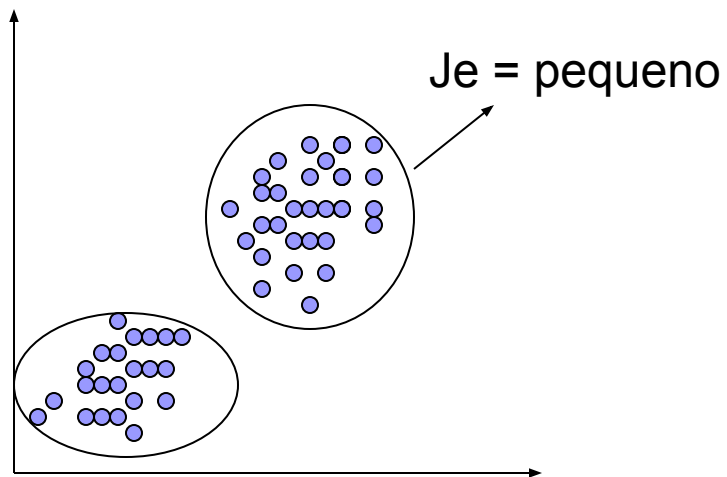
- É o mais simples e usado critério de otimização em *clustering*.
- Seja  $n_i$  o número de exemplos no cluster  $D_i$  e  $m_i$  a média desses exemplos

$$m_i = \frac{1}{n_i} \sum_{x \in D_i} x$$

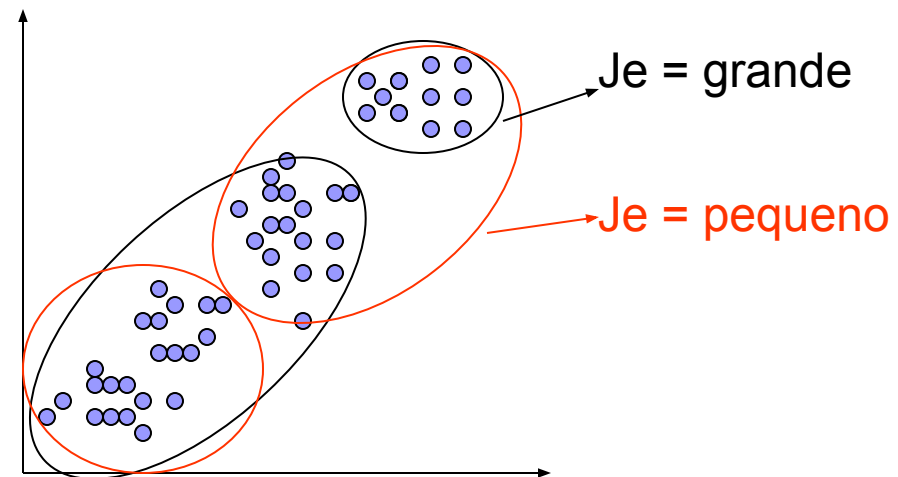
- A soma dos erros quadrados é definida

$$J_e = \sum_{i=1}^c \sum_{x \in D_i} (x - m_i)^2$$

# Soma dos Erros Quadrados



Adequado nesses casos  
- Separação natural



Não é muito adequado para dados  
mais dispersos.  
*Outliers* podem afetar bastante os  
vetores médios  $\mathbf{m}$

# Critérios de Dispersão

- Vetor médio do cluster  $i$

$$m_i = \frac{1}{n_i} \sum_{x \in D_i} x$$

- Vetor médio total

$$m = \frac{1}{n} \sum_D x$$

- Dispersão do cluster  $i$

$$S_i = \sum_{x \in D_i} (x - m_i)(x - m_i)^t$$

- Within-cluster

$$S_w = \sum_{i=1}^c S_i$$

- Between-cluster

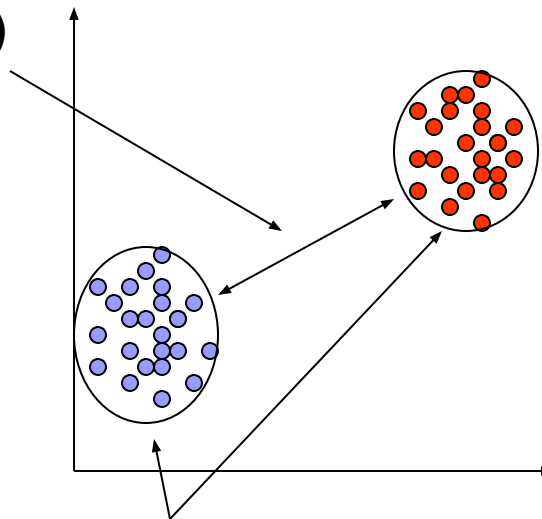
$$S_B = \sum_{i=1}^c n_i (m_i - m)(m_i - m)^t$$



# Critérios de Dispersão

## ■ Relação Within-Between

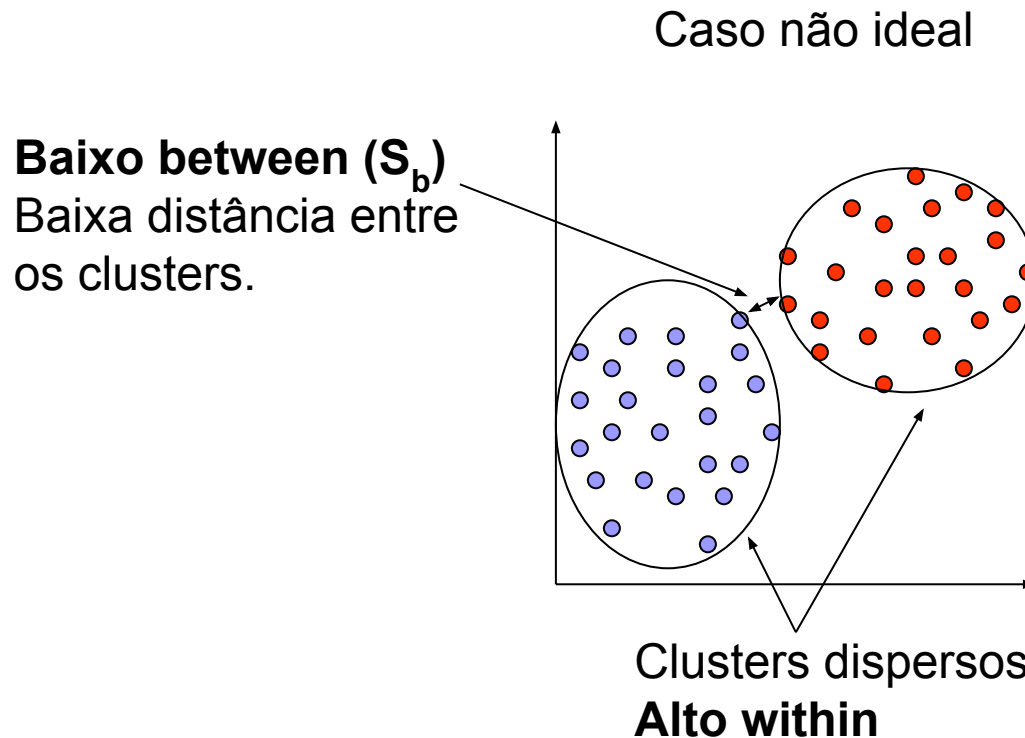
**Alto between ( $S_b$ )**  
Clusters distantes  
um do outro.



**Baixo within ( $S_w$ )**  
(boa compactação)

Caso ideal

# Critérios de Dispersão



# Índice Silhueta

## Largura da Silhueta

- Cada cluster é representado por uma silhueta. Nos mostra que instâncias se posicionam bem dentro do cluster e o tamanho de cada cluster.

A silhueta é calculada para cada instância (i) conforme abaixo:

$$s(i) = (b(i) - a(i)) / \max(a(i), b(i))$$

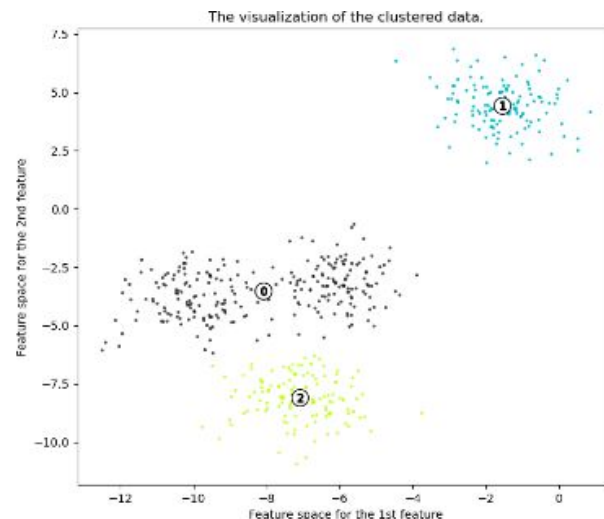
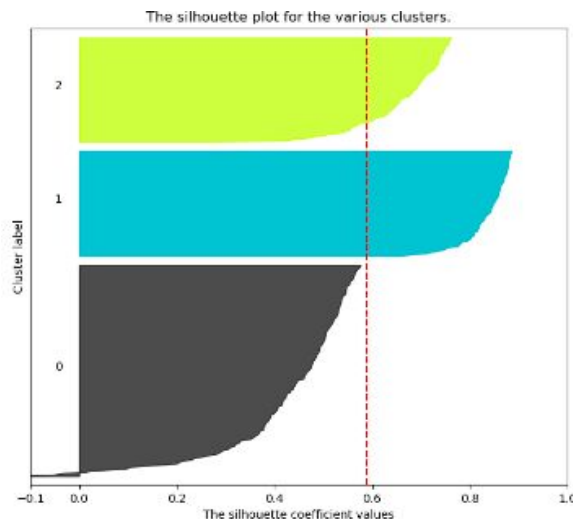
onde:

- $a(i)$  é a dissimilaridade média da instância i em relação a todas as outras instâncias do seu cluster
- $b(i)$  é a dissimilaridade média da instância i em relação a todas as outras instâncias do cluster vizinho mais próximo.

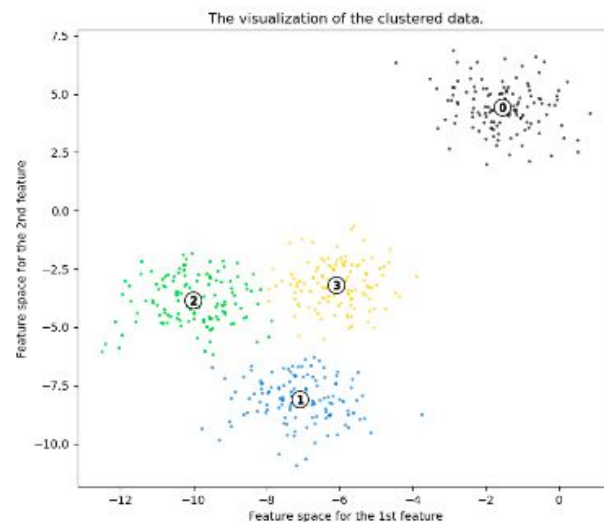
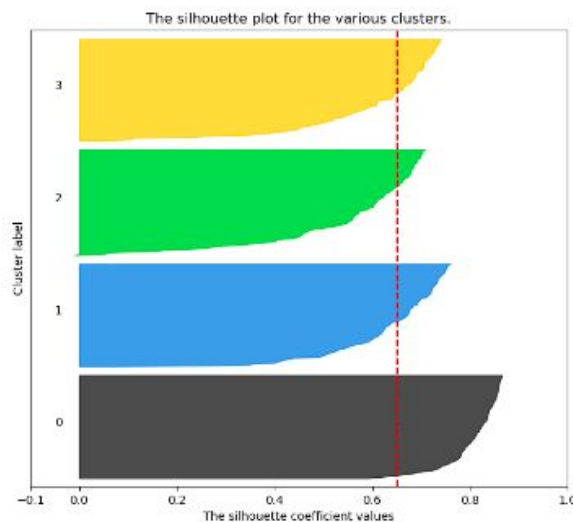
# Índice Silhueta (Exemplo)

Varia de -1 a 1, sendo -1 indesejável.

Silhueta negativa significa que a distância média dos objetos para seu cluster é maior que distância média para outros clusters.

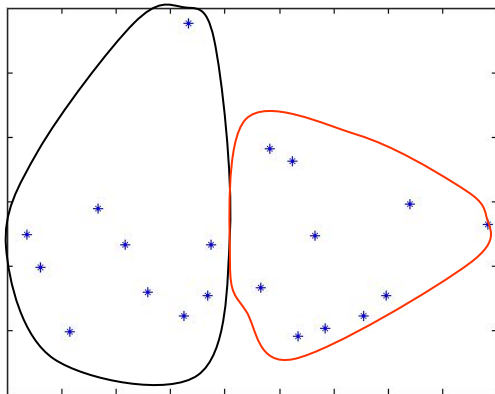


**Silhouette analysis for KMeans clustering on sample data with  $n\_clusters = 4$**

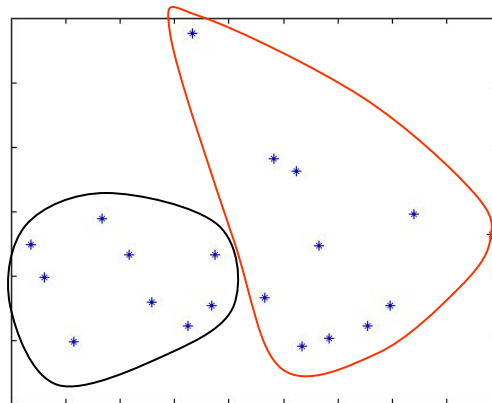


# Diferentes clusters para $c=2$ usando diferentes critérios de otimização

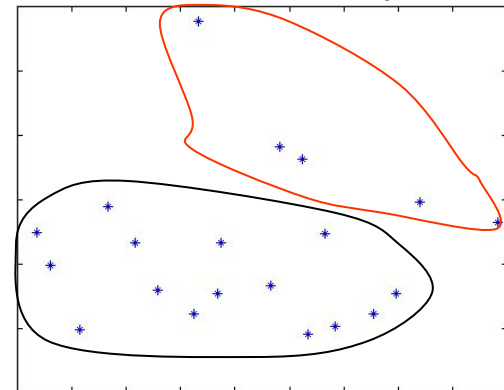
Erro Quadrado



$S_w$



Relação  $S_w/S_b$





# Algumas Aplicações de *Clustering*

- Marketing: Encontrar grupos de consumidores com comportamento similares
- Biologia: Classificar grupos de plantas e animais.
- Bibliotecas: Organização de livros.
- Administração: Organização de cidades, classificando casas de acordo com suas características.
- WWW: Classificação de conteúdos.



# Problemas

- Vetores de atributos muito grandes -> tempo de processamento elevado.
- Definição da melhor medida de distância -> dependente do problema.
- O resultado do *clustering* pode ser interpretado de diferentes maneiras -> dependente de especialista da área.

# Principais Técnicas

- K-means
- X-Means: K-means, onde K é definido automaticamente. Usa BIC (Bayesian Information Criterion).
- Fuzzy C-means: usa noção de pertinência. Uma instância pode pertencer a mais de um cluster.
- Hirárquico: organiza os grupos em uma estrutura hierárquica.
- Mixture of Gaussians: baseado em modelo. EM (Expectation Maximization)





# Questões?

- Obrigado.