Aprendendo com os Vizinhos

SCC0244 - Mineração a partir de Grandes Bases de Dados

Rafael Geraldeli Rossi
Solange Oliveira Rezende

Slides baseados em [Han et al., 2011]

Conteúdo

- Paradigma de aprendizado baseado em instâncias
- Abordagem "*lazy*" (preguiçosa)
  - O algoritmo de aprendizado aguarda até o último instante para construir um modelo e classificar um exemplo
  - Dado os exemplos de treinamento, o aprendizado *lazy* apenas armazena os exemplos e espera até que seja dado um exemplo de teste para realizar algum tipo de processamento
  - Classifica um exemplo baseado na similaridade dos exemplos de treinamento
- Diferente dos outros paradigmas, há um menor esforço na etapa de aprendizado e um maior esforço na etapa de classificação
- Requer técnicas eficientes de armazenamento e recuperação
- Naturalmente suportam aprendizado incremental
- São capazes de modelar espaços de decisões complexos
- Veja: `http://www.cs.cmu.edu/~alad/knn.html`

- *k*NN - *k Nearest Neighbors*
- Algoritmo amplamente utilizado na área de reconhecimento de padrões
- A classificação utilizando os vizinhos mais próximos, como o próprio nome diz, faz uso dos rótulos dos vizinho para descobrir a classe de um objeto não rotulado
- No caso do *k*NN são utilizados os rótulos dos *k* vizinhos mais próximos
- É atribuído a classe mais frequente dos *k* vizinhos ao exemplo de teste
- Normalmente os valores dos atributos são normalizados

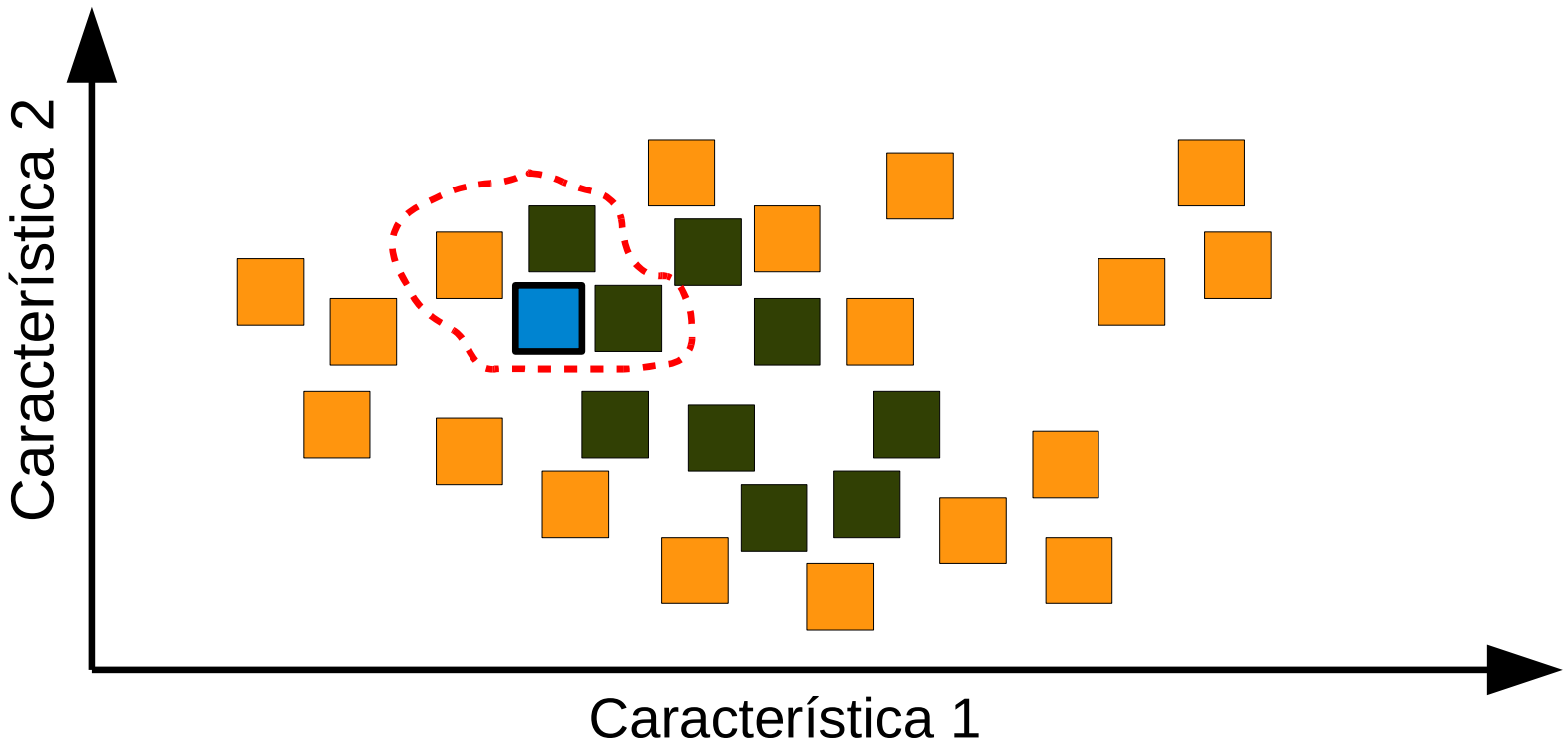

Figura: Exemplo de classificação utilizando os 3 vizinhos mais próximos

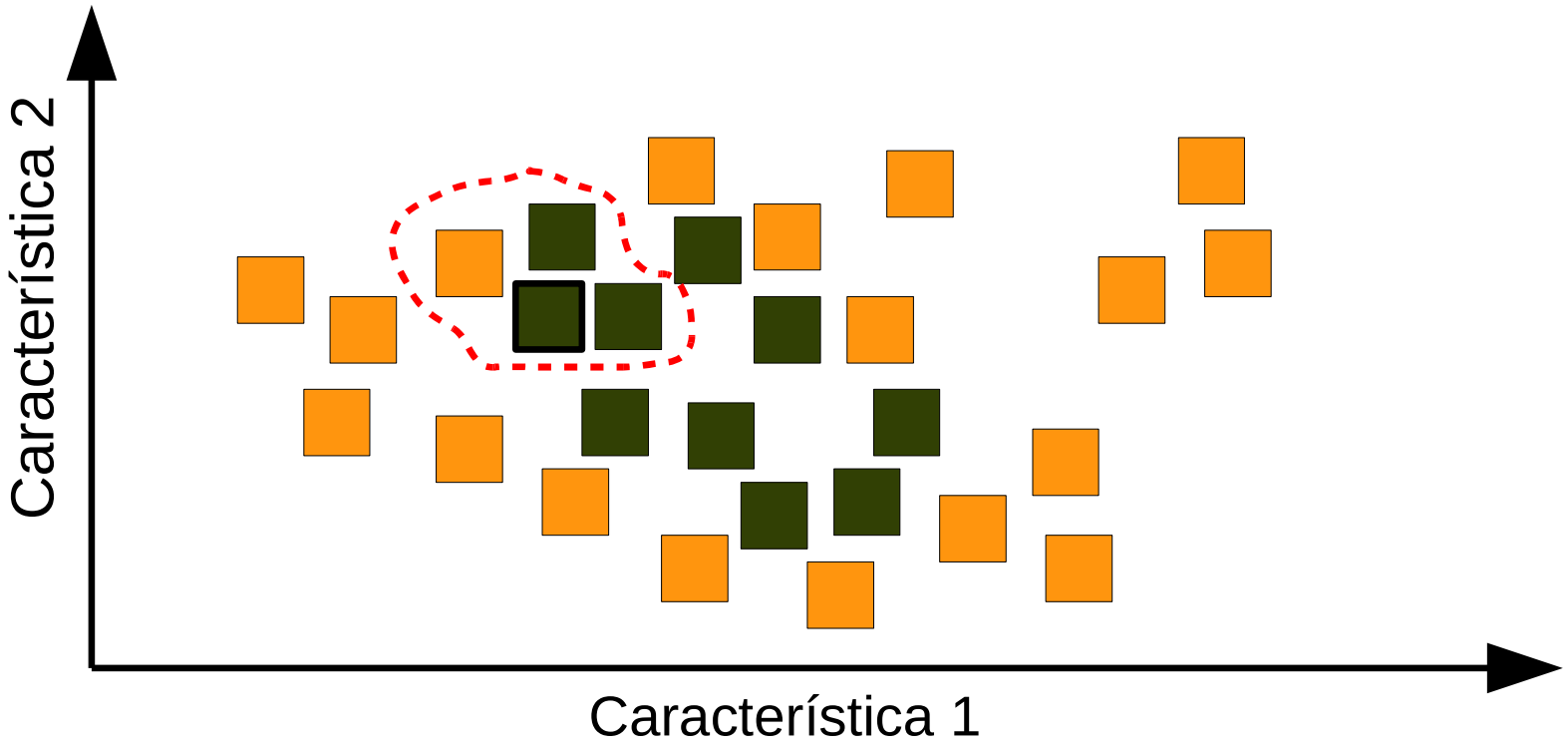

Figura: Exemplo de classificação utilizando os 3 vizinhos mais próximos

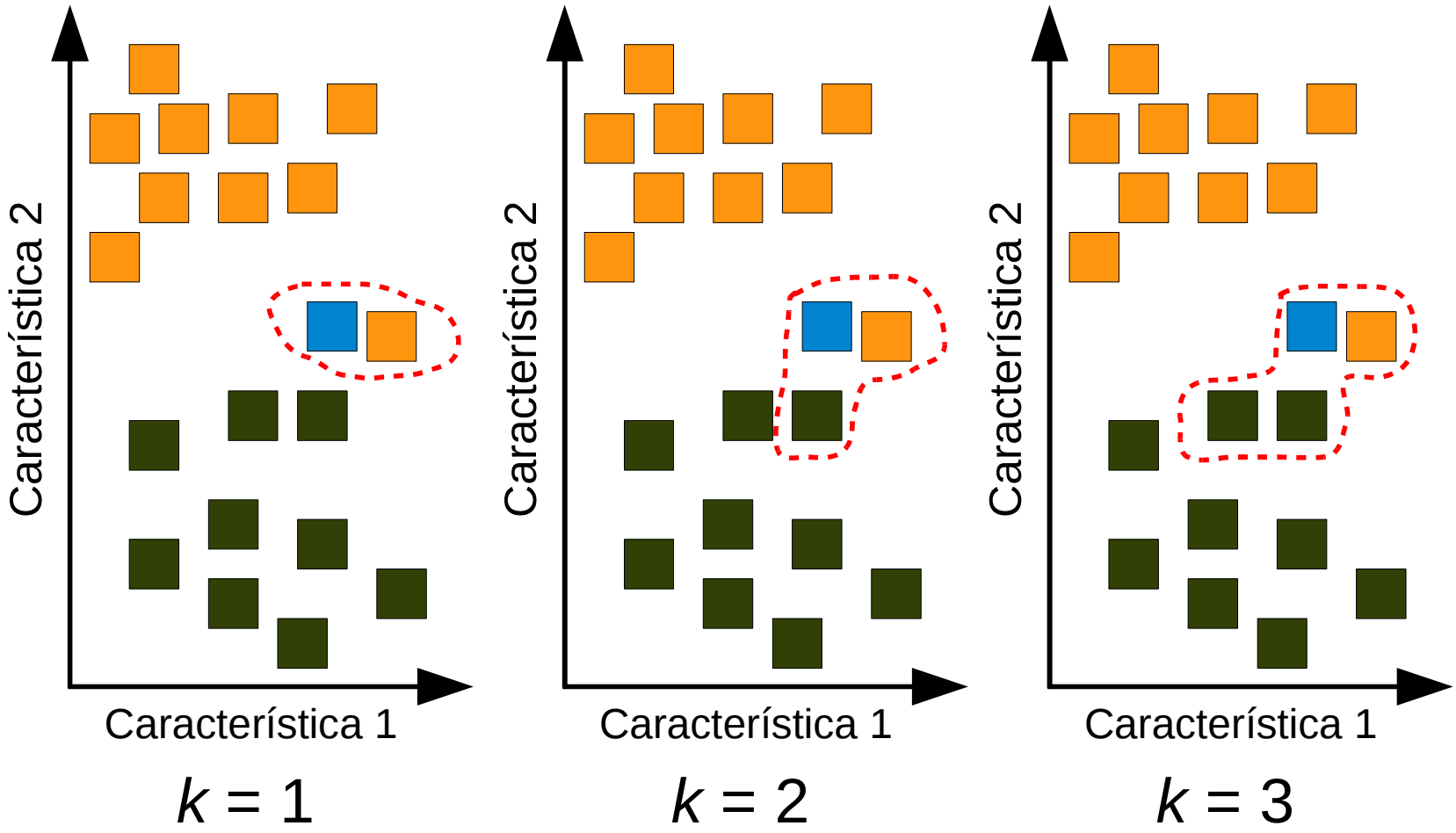

Figura: Efeito do valor de *k*

Exemplo

Tabela: Parte do conjunto de dados *Iris*

ID	Sepal Length	Sepal Width	Petal Length	Petal Width	Class
1	5,1	3,5	1,4	0,2	Iris-setosa
2	4,9	3,0	1,4	0,2	Iris-setosa
3	7,0	3,2	4,7	1,4	Iris-versicolor
4	6,4	3,2	4,5	1,5	Iris-versicolor
5	6,3	3,3	6,0	2,5	Iris-virginica
6	5,8	2,7	5,1	1,9	Iris-virginica

Sepal Length	Sepal Width	Petal Length	Petal Width	Class
5,4	3,1	2,5	1,0	**???**

$$d(t,1) = \sqrt{(5,4-5,1)^2 + (3,1-3,5)^2 + (2,5-1,4)^2 + (1,0-0,2)^2}$$
$$d(t,1) = \sqrt{(0,09 + 0,16 + 1,21 + 0,64)} = \sqrt{2,1} = 1,44$$

$$d(t,2) = \sqrt{(5,4-4,9)^2 + (3,1-3,0)^2 + (2,5-1,4)^2 + (1,0-0,2)^2}$$
$$d(t,2) = \sqrt{(0,25 + 0,01 + 1,21 + 0,64)} = \sqrt{2,11} = 1,45$$

$$d(t,3) = \sqrt{(5,4-7,0)^2 + (3,1-3,2)^2 + (2,5-4,7)^2 + (1,0-1,4)^2}$$
$$d(t,3) = \sqrt{2,56 + 0,01 + 4,84 + 0,16} = \sqrt{7,21} = 2,68$$

$$d(t,4) = \sqrt{(5,4-6,4)^2 + (3,1-3,2)^2 + (2,5-4,5)^2 + (1-1,5)^2}$$
$$d(t,4) = \sqrt{1,0 + 0,01 + 4,0 + 0,25} = \sqrt{5,26} = 2,29$$

$$d(t,5) = \sqrt{(5,4-6,3)^2 + (3,1-3,3)^2 + (2,5-6,0)^2 + (1,0-2,5)^2}$$
$$d(t,5) = \sqrt{0,81 + 0,04 + 12,25 + 2,25} = \sqrt{15,35} = 3,91$$

$$d(t,6) = \sqrt{((5,4-5,8))^2 + (3,1-2,7)^2 + (2,5-5,1)^2 + (1,0-1,9)^2}$$
$$d(t,6) = \sqrt{0,16 + 0,16 + 6,76 + 0,81} = \sqrt{7,89} = 2,80$$

Tabela: *Ranking* dos vizinhos mais próximos

Ranking	**ID**	**Distância**	**Classe**
1º	1	1,44	Iris-setosa
2º	2	1,45	Iris-setosa
3º	4	2,29	Iris-versicolor
4º	3	2,68	Iris-versicolor
5º	6	2,80	Iris-virginica
6º	5	3,91	Iris-virginica

- Resultados de classificação
  - 1-NN: Iris-setosa
  - 2-NN: Iris-setosa
  - 3-NN: Iris-setosa
  - 4-NN: Empate entre Iris-setosa e Iris-versicolor
  - 5-NN: Empate entre Iris-setosa e Iris-versicolor
  - 6-NN: Empate entre Iris-setosa, Iris-versicolor e Iris-virginica

- Pode-se dar um peso diferente ao voto de cada vizinho
  - O peso do voto é dado por

$$voto = \frac{1}{dist(x, novo)}$$

    na qual $dist(x, novo)$ é a distância de um objeto x da base de treinamento ao objeto a ser classificado
  - É realizado um somatório com o peso do voto dos objetos de cada classe
  - O objeto é classificado com a classe que obteve o maior somatório de votos (considerando o peso)
  - Reduz a sensibilidade da escolha do valor de k

Tabela: *Ranking* dos vizinhos mais próximos

Ranking	ID	Distância	Classe
1º	1	1,44	Iris-setosa
2º	2	1,45	Iris-setosa
3º	4	2,29	Iris-versicolor
4º	3	2,68	Iris-versicolor
5º	6	2,80	Iris-virginica
6º	5	3,91	Iris-virginica

- Resultados da classificação
  - 1-NN: **Iris-setosa** $= 1/1,44$; Iris-versicolor $= 0$; Iris-virginica $= 0$
  - 2-NN: **Iris-setosa** $= 1/1,44 + 1/1,45$; Iris-versicolor $= 0$; Iris-virginica $= 0$
  - 3-NN: **Iris-setosa** $= 1/1,44 + 1/1,45$; Iris-versicolor $= 1/2,29$; Iris-virginica $= 0$
  - 4-NN: **Iris-setosa** $= 1/1,44 + 1/1,45$; Iris-versicolor $= 1/2,29 + 1/2,68$; Iris-virginica $= 0$
  - 5-NN: **Iris-setosa** $= 1/1,44 + 1/1,45$; Iris-versicolor $= 1/2,29 + 1/2,68$; Iris-virginica $= 1/2,80$
  - 6-NN: **Iris-setosa** $= 1/1,44 + 1/1,45$; Iris-versicolor $= 1/2,29 + 1/2,68$; Iris-virginica $= 1/2,80 + 1/3,91$

- O uso de peso nos votos pode gerar erros devido a presença de *outliers*
- Um objeto pode estar tão próximo a um *outlier* de modo que o peso os votos de outros objetos não sejam o suficiente para definir a classe de um objeto

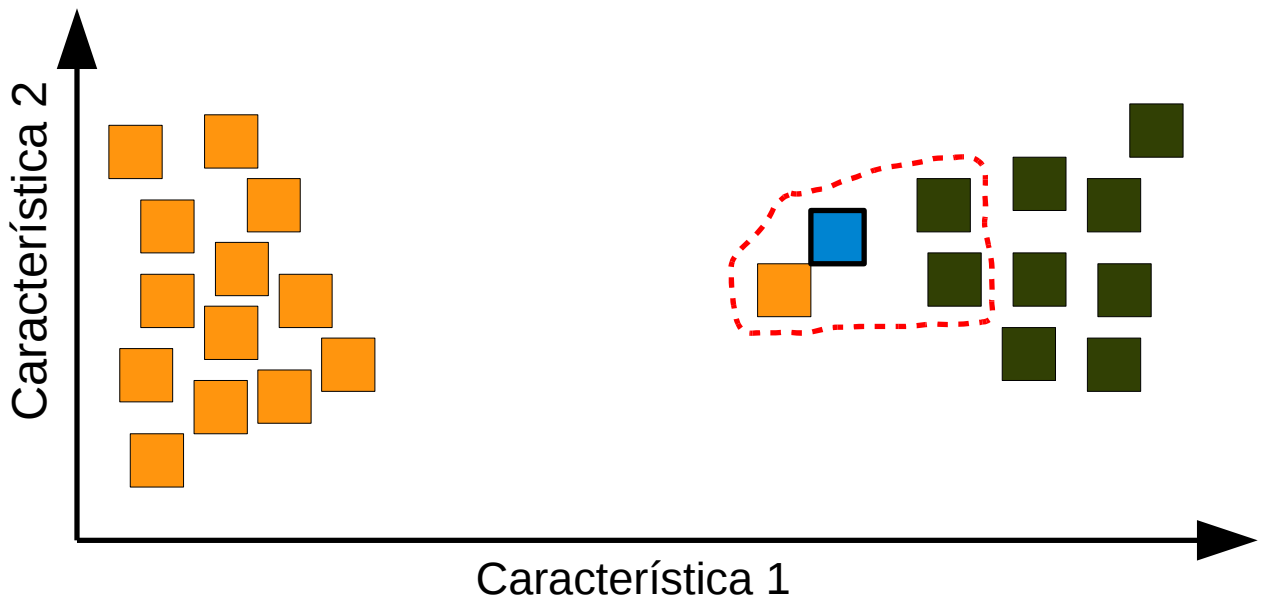

Figura: Exemplo de problemas na classificação utilizando 3-NN devido a presença de *outliers*

- O *k*-NN também pode ser utilizado para predição numérica, na qual o valor retornado é a média dos valores dos *k* vizinhos
  - Calcular a média do atributo alvo utilizando o valor do mesmo atributo dos *k*-vizinhos mais próximos
- Exemplo: calcular o salário do exemplo $\{Idade = 31; Tempo\ de\ Serviço = 13\}$ utilizando 3 vizinhos mais próximos e o seguinte conjunto de treinamento

Tabela: Conjunto de dados original

ID	Idade	Tempo de Serviço	Salário
1	20	2	2000
2	25	3	2500
3	50	25	8000
4	30	10	5000
5	27	5	3000
6	33	10	2700

Tabela: Conjunto de dados padronizado para o cálculo das distâncias

ID	Idade	Tempo de Serviço	Salário
1	0,00	0,00	2000
2	0,17	0,04	2500
3	1,00	1,00	8000
4	0,33	0,35	5000
5	0,23	0,13	3000
6	0,43	0,35	2700

- Exemplo de teste padronizado:
  $\{Idade = 0,37; Tempo\ de\ Serviço = 0,48\}$

Tabela: *Ranking* dos vizinhos mais próximos

Ranking	ID	Distância	Salário
1º	4	0,1	5000
2º	6	0,1	2700
3º	2	0,47	2500
4º	1	0,6	2000
5º	3	0,66	8000
6º	5	0,7	3000

- Salário do exemplo de teste

$$Salário = \frac{5000 + 2700 + 2500}{3} = 3400,00$$

- O mesmo procedimento pode ser utilizado para a imputação de valores ausentes
  - Deve-se desconsiderar o atributo que possui valor ausente no cálculo das distâncias

Tabela: Conjunto de dados original

ID	Idade	Tempo de Serviço	Salário
1	20	2	2000
2	25	–	2500
3	50	25	8000
4	30	10	5000
5	27	5	3000
6	33	10	2700

Tabela: Conjunto de dados padronizados

ID	Idade	Tempo de Serviço	Salário
1	0,00	2	0,00
2	0,17	–	0,08
3	1,00	25	1,00
4	0,33	10	0,50
5	0,23	5	0,17
6	0,43	10	0,12

Tabela: *Ranking*

Ranking	**ID**	**Distância**	**Tempo de Serviço**
1º	5	0,1	5
2º	1	0,14	2
3º	6	0,24	10
4º	4	0,43	10
5º	3	1,23	25

- Utilizando 2 vizinho mais próximos temos

$$\textit{Tempo de Serviço} = \frac{5+2}{2} = 3,5$$

- O valor de k é determinado experimentalmente
- A cada valor de k, é realizada uma avaliação em um conjunto de teste
- É escolhido o valor de k com melhor desempenho no conjunto de teste
- Em geral
  - Valor de k pequeno
    - Função de discriminação entre classes é muito flexível
    - Sensível a ruído
  - Valor de k grande
    - Função de discriminação entre classes é menos flexível
    - Tende a incluir objetos de outras classes
    - Menos sensível a ruído

- A escolha da métrica de distância é fundamental
- Seja $|D|$ o número de exemplos de treinamento e $|A|$ o número de atributos, a complexidade do k-NN é $O(|D| \times |A|)$
- Técnica para acelerar a classificação
  - Implementações paralelas
  - Calculo da distância baseada em um subconjunto de atributos
  - Remover exemplos de treinamento que são inconsistentes com seus próprios vizinhos
  - kD-tree
  - ...

Referências Bibliográficas I

Han, J., Kamber, M., and Pei, J. (2011).
Data Mining: Concepts and Techniques.
The Morgan Kaufmann Series in Data Management Systems.
Elsevier.