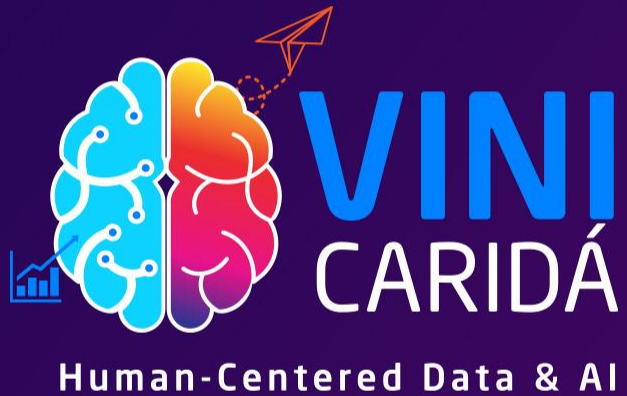
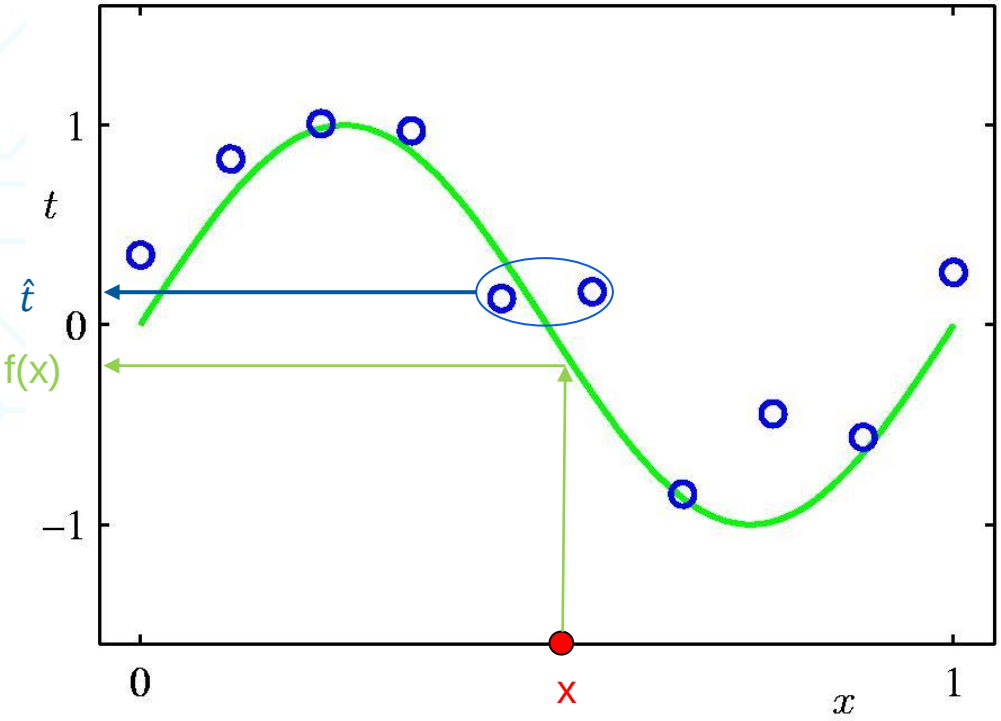
FIAP

MBA+ | Artificial Intelligence & Machine Learning


VINI
CARIDÁ
Human-Centered Data & AI


VINI
CARIDÁ
Human-Centered Data & AI
“
KNN
Vinicius F. Caridá

k-Nearest Neighbors

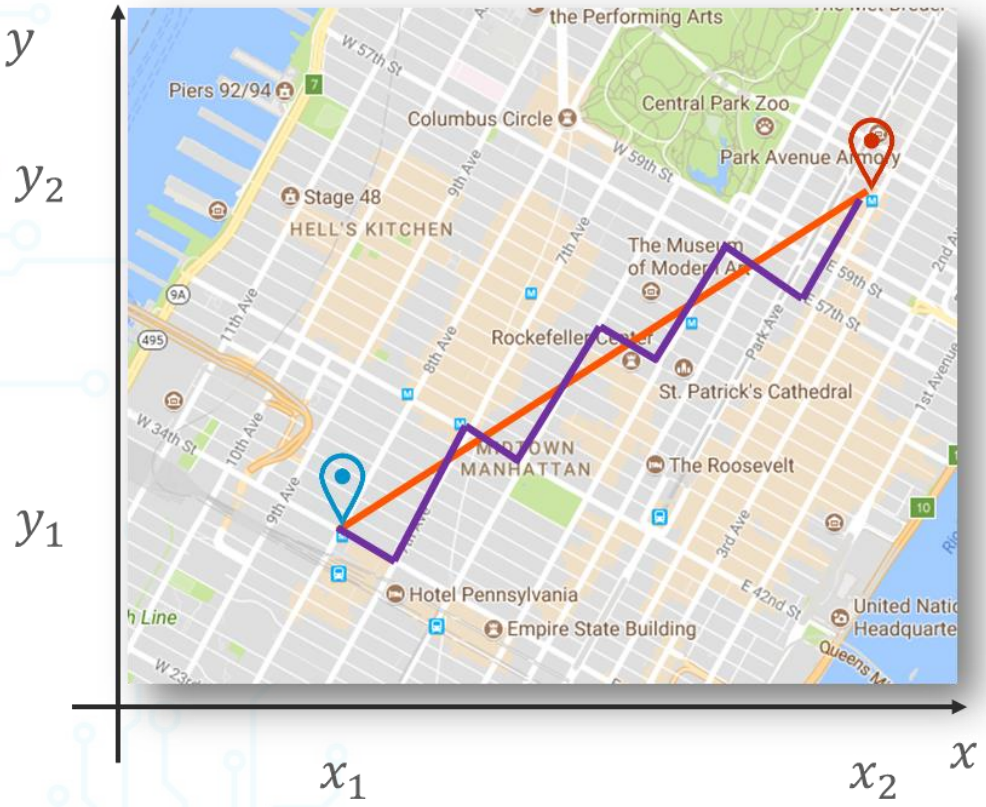


Dado x, $t = f(x)$?

Problema: $f(\cdot)$ é desconhecida.

Solução k-NN: Estimar *t* por meio da média dos vizinhos mais próximos (em relação aos valores de x apenas).

Medidas de Distância



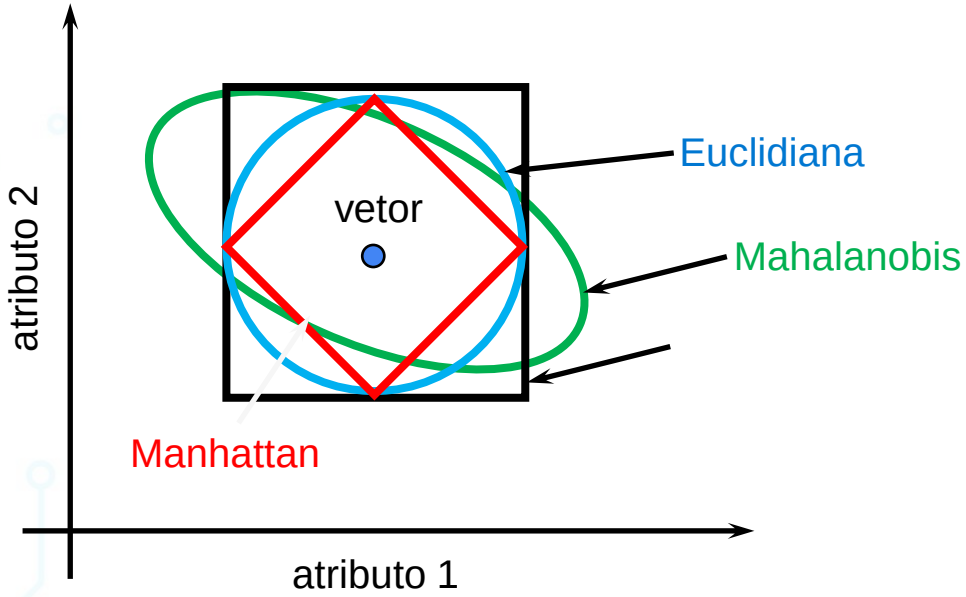
Manhattan

$$d_1 = |x_1 - x_2| + |y_1 - y_2|$$

Euclidiana

$$d_2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Medidas de Distância

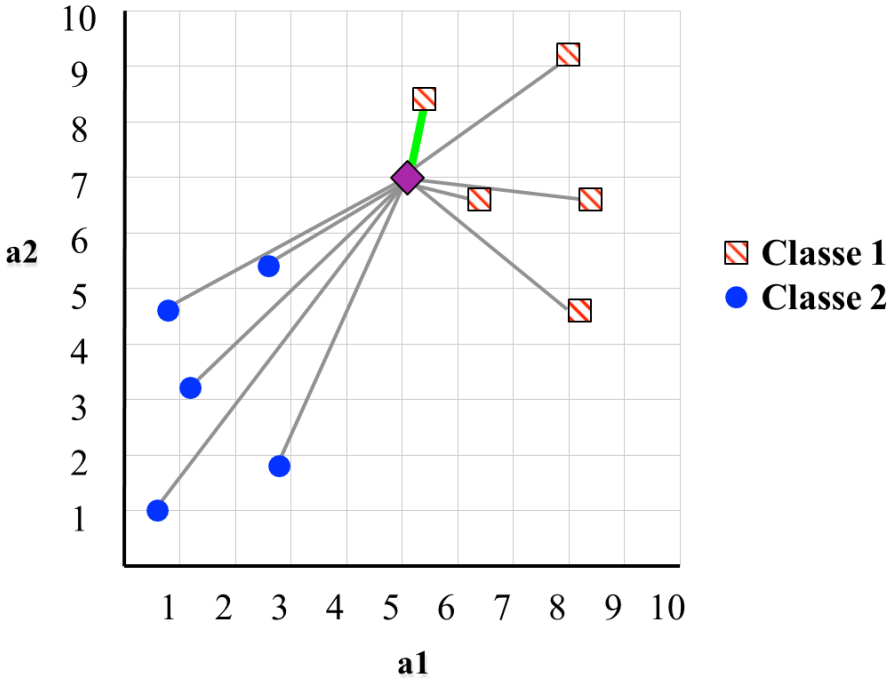


Aprendizado baseado em exemplos

- Aprendizado baseado em exemplos (*Instance- based Learning*):
  - Não constroem descrições gerais e explícitas (função alvo) a partir dos exemplos de treinamento;
  - Generalização é *adiada* até o momento da classificação;
  - Armazena-se uma base de exemplos (*instances*) que é usada para realizar a classificação de uma nova *query* (exemplo *não visto*);
  - Em geral apresenta alto custo computacional.

Aprendizado baseado em exemplos

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**



Fonte: Keogh, E. AGentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

KNN

- *K-Nearest Neighbors (KNN);*
- Exemplos correspondem a pontos no espaço n-dimensional ($\Re^n$);
- Em geral os vizinhos são definidos em função de uma medida de distância. Por exemplo, considerando-se dois vetores $\mathbf{x}=[x_1,x_2,\ldots,x_n]$ e $\mathbf{y}=[y_1,y_2,\ldots,y_n]$, a distância Euclidiana entre estes dois vetores é dada por:

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#1,\#11) = \sqrt{(0.5-5)^2+(1-7)^2}$$

$$d_E(\#1,\#11) = \sqrt{(-4.5)^2+(-6)^2}$$

$$d_E(\#1,\#11) = \sqrt{20.25+36}$$

$$d_E(\#1,\#11) = \sqrt{56.25}$$

$$d_E(\#1,\#11) = 7.5$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#2,\#11) = \sqrt{(2.9-5)^2+(1.9-7)^2}$$

$$d_E(\#2,\#11) = \sqrt{(-2.1)^2+(-5.1)^2}$$

$$d_E(\#2,\#11) = \sqrt{4.41+26.1}$$

$$d_E(\#2,\#11) = \sqrt{30.51}$$

$$d_E(\#2,\#11) = 5.5$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#3,\#11) = 5.4$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#4,\#11) = 4.7$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#5,\#11) = 2.8$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#6,\#11) = 3.8$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#7,\#11) = 3.3$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#8,\#11) = 1.3$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#9,\#11) = 3.6$$

KNN

#	a1	a2	Classe
1	0.5	1	2
2	2.9	1.9	2
3	1.2	3.1	2
4	0.8	4.7	2
5	2.7	5.4	2
6	8.1	4.7	1
7	8.3	6.6	1
8	6.3	6.7	1
9	8	9.1	1
10	5.4	8.4	1
11	**5**	**7**	**?**

$$d_E(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_E(\#10, \#11) = 1.4$$

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**?**	

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**?**	

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**1**	

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	5	7	?	

Para k=1, temos o 1-NN e a classe da instância #11 seria 1.

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**?**	

Para k=2, temos o 2-NN e a classe da instância #11 seria ?

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	5	7	?	

Para k=2, temos o 2-NN e a classe da instância #11 seria 1

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	5	7	?	

Para k=3, temos o 3-NN e a classe da instância #11 seria ?

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	5	7	?	

Para k=3, temos o 3-NN e a classe da instância #11 seria 1, pois temos 2 vizinho da classe 1 e apenas 1 vizinho da classe 2.

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**?**	

Para k=3, temos o 3-NN e a classe da instância #11 seria 1, pois temos 2 vizinho da classe 1 e apenas 1 vizinho da classe 2.
E se houvesse

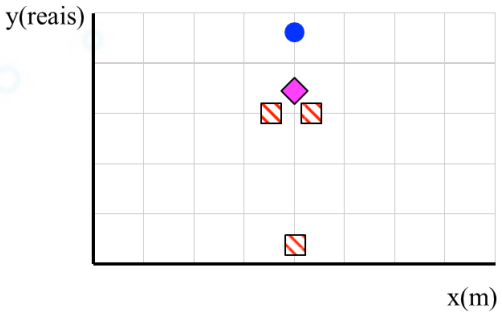
KNN

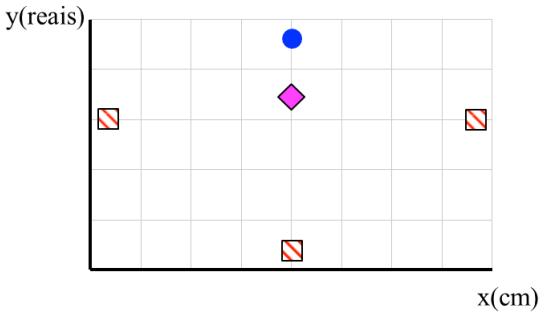
#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**?**	

Para k=4, temos o 4-NN e a classe da instância #11 seria ?

KNN

Sensibilidade em relação a escala





Como diminuir este problema?

- Normalização linear;
- *Score z;*

KNN

Como lidar com atributos nominais?

- Mudar a função de distância. Por exemplo, pode-se usar o procedimento chamado de *simple matching*:

$$d_{SM}(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{i=n} s_i \quad \begin{cases} (x_i = y_i) \Rightarrow s_i = 0; \\ (x_i \neq y_i) \Rightarrow s_i = 1; \end{cases}$$

- Existem várias outras medidas de distância propostas na literatura (e.g. Kaufman & Rousseeuw, 1990);
- Como lidar com bases dados formadas por atributos ordinais, contínuos, nominais, binários?

KNN

KNN ponderado pela distância

10.1 - Função alvo discreta:

$$f(\mathbf{x}_q) \leftarrow \underset{v \in V}{arg\,max} \sum_{i=1}^{k} w_i \delta(v, f(\mathbf{x}_i)) \begin{cases} (a = b) \Rightarrow \delta(a,b) = 1 \\ (a \neq b) \Rightarrow \delta(a,b) = 0 \end{cases}$$

10.2 - Função alvo contínua:

$$y = f(\mathbf{x}_q) = \frac{\sum_{i=1}^{k} w_i f(\mathbf{x}_i)}{\sum_{i=1}^{k} w_i}$$

Ponderação:

$$w_i = \frac{1}{d(\mathbf{x}_q, \mathbf{x}_i)^2}$$

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	5	7	?	

Para k=1, se não usamos a ponderação pela distância, temos o 1-NN e a classe da instância #11 seria 1. Isso ocorre, pois temos 1 vizinho mais próximo sendo da classe 1, e 0 (zero vizinho) mais próximo sendo da classe 2.

Agora, se usamos a ponderação pela distância, dividimos a "força da ocorrência" de cada vizinho pelo quadrado da sua distância. Assim teríamos, 1/distância vizinho da classe 1 e 0/distância vizinho da classe 2. Ou seja:
$1/(1.3)^2 = 1/1.69 = 0.59$ vizinho da classe 1;
$0/(\text{distância})^2 = 0$ vizinho da classe 2.

A classe é 1!

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	**5**	**7**	**?**	

Para k=2, temos o 2-NN e, sem a ponderação pela distância temos 2 vizinhos mais próximos sendo da classe 1 e 0 vizinhos sendo da classe 2. Assim, a classe seria 1.

Já com a ponderação temos:
$1/(1.3)^2 + 1/(1.4)^2 = 0.59 + 0.51 = 1.1$ vizinhos sendo da classe 1.
$0/(\text{distâcia})^2 = 0$ vizinho sendo da classe 2.

A classe é 1!

KNN

#	a1	a2	Classe	Distância para a instância #11
1	0.5	1	2	7.5
2	2.9	1.9	2	5.5
3	1.2	3.1	2	5.4
4	0.8	4.7	2	4.7
5	2.7	5.4	2	2.8
6	8.1	4.7	1	3.8
7	8.3	6.6	1	3.3
8	6.3	6.7	1	1.3
9	8	9.1	1	3.6
10	5.4	8.4	1	1.4
11	5	7	?	

Para k=3, temos o 3-NN e, sem a ponderação, a classe da instância #11 seria 1.

Já com a ponderação temos:
$1/(1.3)^2 + 1/(1.4)^2 + 0/(2.8)^2 =$
$0.59 + 0.51 + 0 = 1.1$ vizinhos sendo da classe 1.
$1/(2.8)^2 = 1/7.84 = 0.12$ vizinho sendo da classe 2.

A classe é 1!

Note que há menos probabilidade de empate!

KNN

#	Classe	Distância para a instância #11
1	2	7.5
2	1	5.5
3	2	5.4
4	2	4.9
5	2	7.8
6	1	4.8
7	1	7.3
8	1	5.3
9	1	4.6
10	2	1.4
11	?	

Exemplo onde a ponderação gera impacto no resultado

Para k=2, temos o 2-NN e, sem ponderação, temos empate entre as duas classes. Assim, a classe da instância #11 seria qualquer uma das duas classes, pois temos 1 vizinho da classe 1 e 1 vizinho da classe 2.

Já com a ponderação temos:
$1/(1.4)^2 + 0/(4.6)^2 =$ 0.51 vizinho sendo da classe 2.
$0/(1.4)^2 + 1/(4.6)^2 = 0/1.96 + 1/21.1 =$
0 + 0.04 = 0.04 vizinho da classe 1

A classe é 2!

KNN

#	Classe	Distância para a instância #11
1	2	7.5
2	1	5.5
3	2	5.4
4	2	4.9
5	2	7.8
6	1	4.8
7	1	7.3
8	1	5.3
9	1	4.6
10	2	1.4
11	?	

Exemplo onde a ponderação gera impacto no resultado

Para k=3, temos o 3-NN e, sem ponderação, a classe da instância #11 seria 1, pois temos 2 vizinhos da classe 1 e apenas 1 vizinho da classe 2.

Já com a ponderação temos:
$1/(1.4)^2 + 0/(4.6)^2 + 0/(4.8)^2 =$ 0.51 vizinho sendo da classe 2.
$0/(1.4)^2 + 1/(4.6)^2 + 1/(4.8)^2 =$
$0/1.96 + 1/21.1 + 1/23.0 =$
$0 + 0.04 + 0.04 = 0.08$ vizinho da classe 1

A classe é 2!

Vantagens e Desvantagens

Performance

Não constrói um modelo de classificação.

Processo de classificação de uma tupla é lento.

Outros classificadores gastam tempo para construir o modelo. O processo de classificação de uma tupla é rápido.

Sensível a ruídos

KNN faz predição baseando-se em informações locais à tupla sendo classificada.

Arvores de decisão, redes neurais e bayesianas encontram modelo global que se leva em conta todo o banco de dados de treinamento.

KNN

- **Itens importantes durante o processo**

1) Diferentes tamanhos de conjuntos de dados
2) Valor do K
3) Escala e tipo dos atributos
4) Validação cruzada
5) Classes desbalanceadas
6) Matriz de confusão

Thanks !

VINICARIDÁ

Human-Centered Data & AI

Vinicius Fernandes Caridá

vfcarida@gmail.com


@vinicius caridá


@vfcarida


@vinicius caridá


@vfcarida


@vinicius caridá


@vfcarida

O que você achou da aula de hoje?

Questions and Feedback