



VINI
CARIDÁ

Human-Centered Data & AI



Vinicius Caridá, PhD

- Executivo de Tecnologia, Dados e IA no Itaú
- Professor dos MBAs de Dados e IA- FIAP e ESPM



@vinicius caridá



@vfcarida



@vinicius caridá



@vfcarida



@vinicius caridá



@vfcarida

Aplicações de IA Generativa



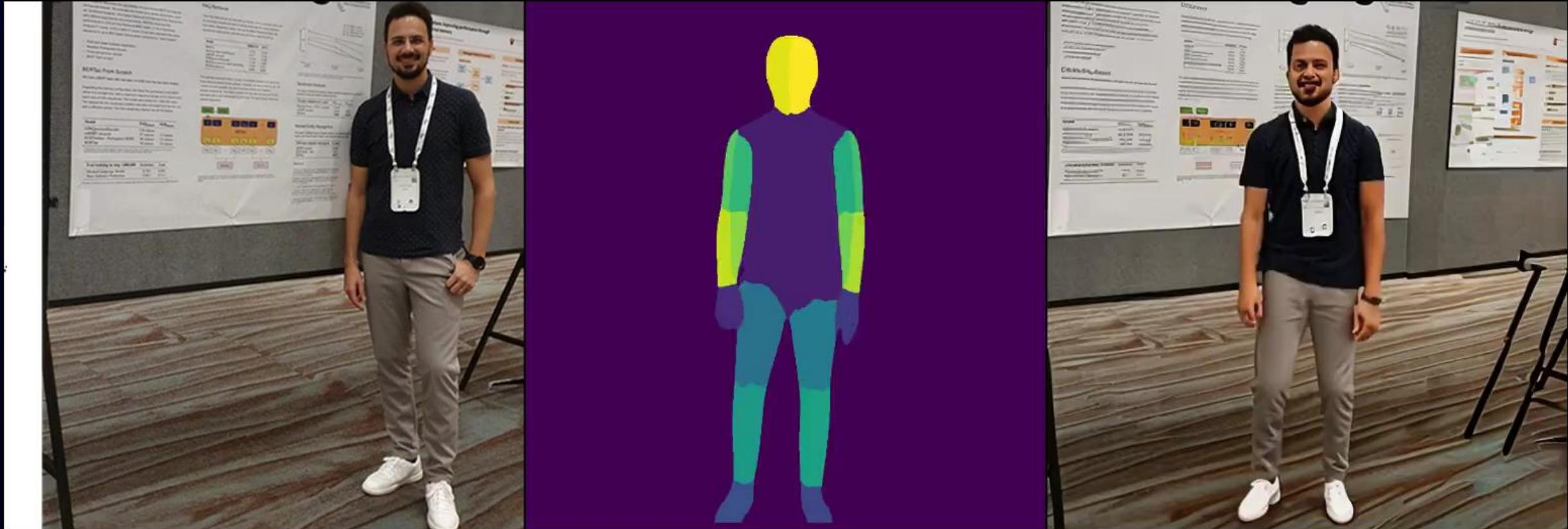
Aplicações de IA Generativa



Aplicações de IA Generativa



Aplicações de IA Generativa



“

Introdução

Modelos de Machine Learning

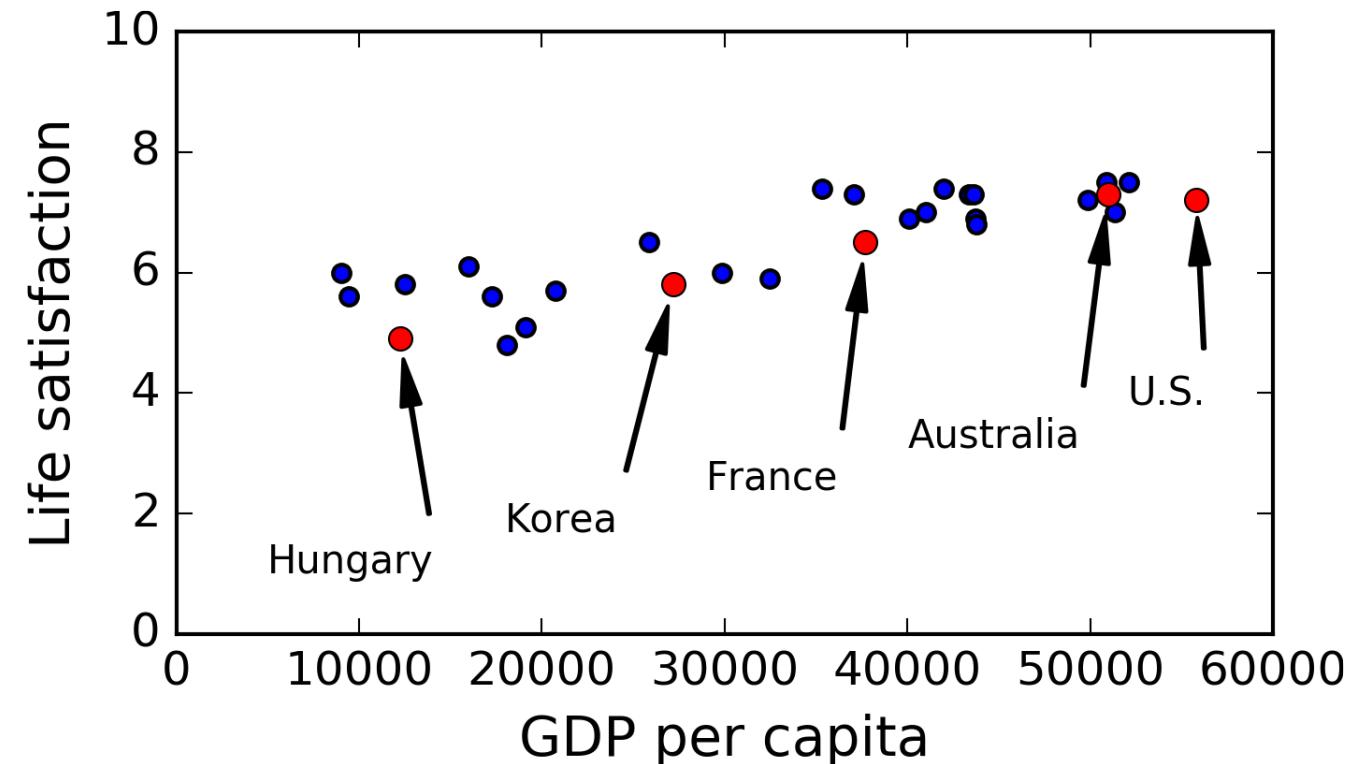
Dinheiro traz felicidade?

Country	GDP per capita (\$US)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United States	55805,0	7,2



O que acham?

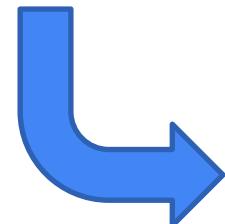
Podemos usar uma função?



Modelos de Machine Learning

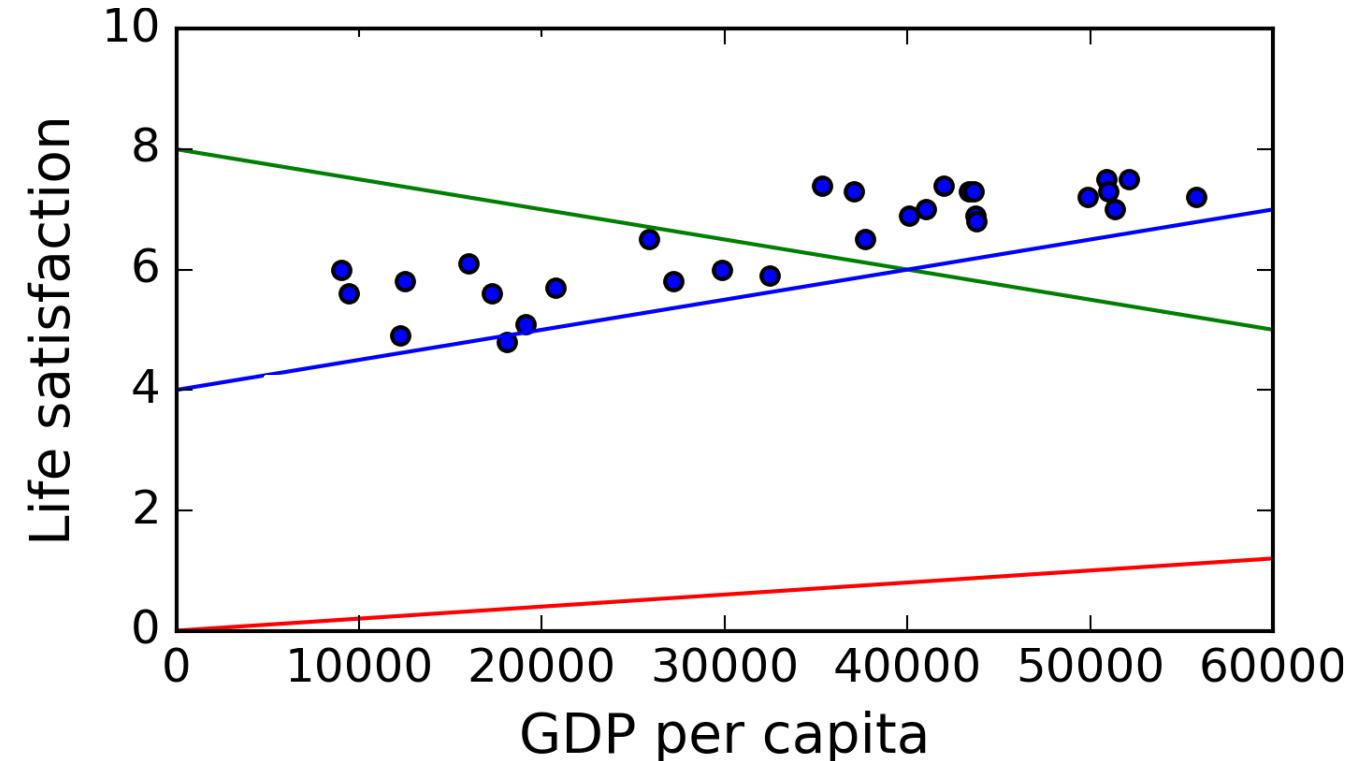
Dinheiro traz felicidade?

Country	GDP per capita (\$US)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United States	55805,0	7,2



Diversas possibilidades

Qual é a melhor e porque?



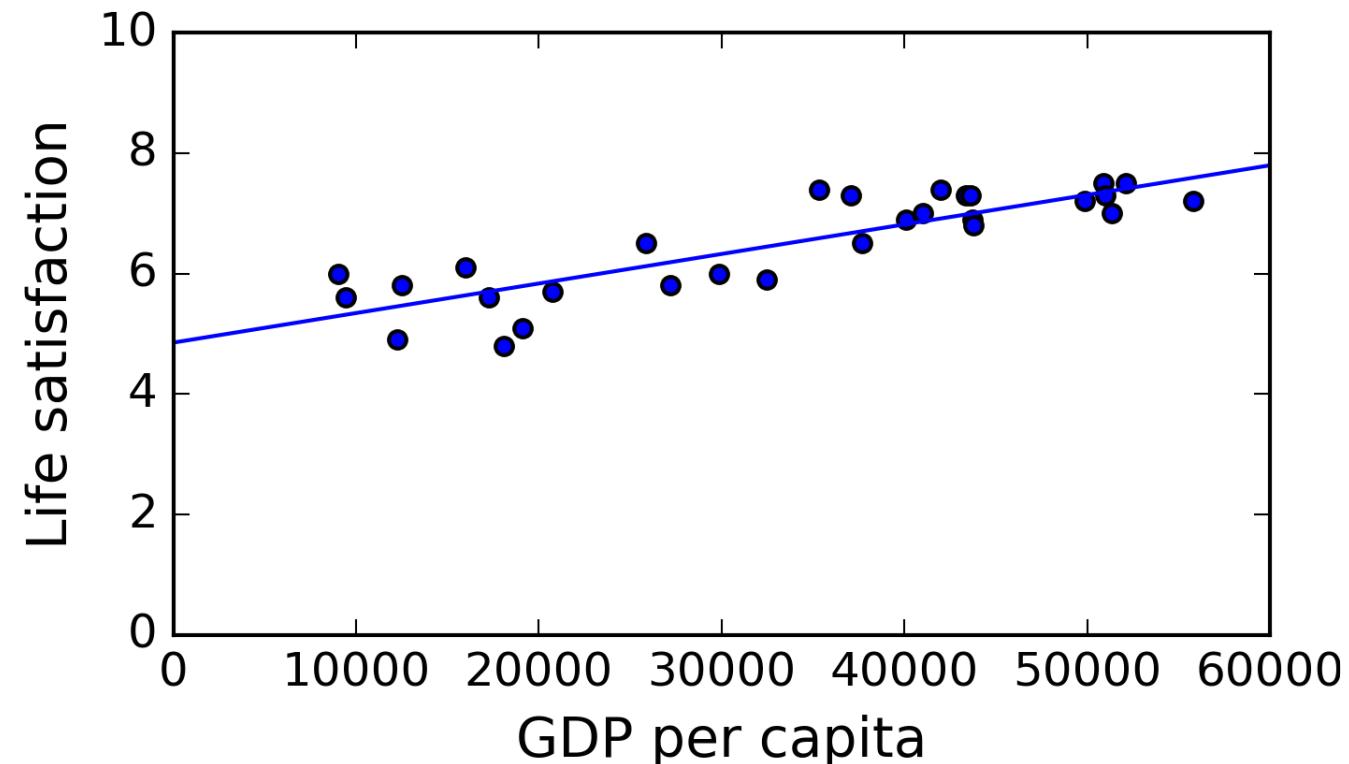
Modelos de Machine Learning

Dinheiro traz felicidade?

Country	GDP per capita (\$US)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United States	55805,0	7,2



Melhor ajuste



Modelos de Machine Learning

Objetivo: “prever” o valor de y (qualidade de vida) usando dados observados de x (renda per capita).

Abordagem: podemos usar uma reta (ou hiperplano, para mais dimensões). Assim:

$$y = f(x)$$

Qual é a cara dessa função?

Modelos de Machine Learning

Abordagem supervisionada simples

Assume uma dependência linear entre a variável resposta Y e os valores X_1, X_2, \dots, X_p

Assume-se o modelo:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

Sendo $\beta_0, \beta_1, \dots, \beta_p$ coeficientes aprendidos pelo modelo

Modelos de Machine Learning

A fim de encontrar o hiperplano de que melhor se ajusta aos dados, devemos minimizar o erro quadrático médio obtido por ele

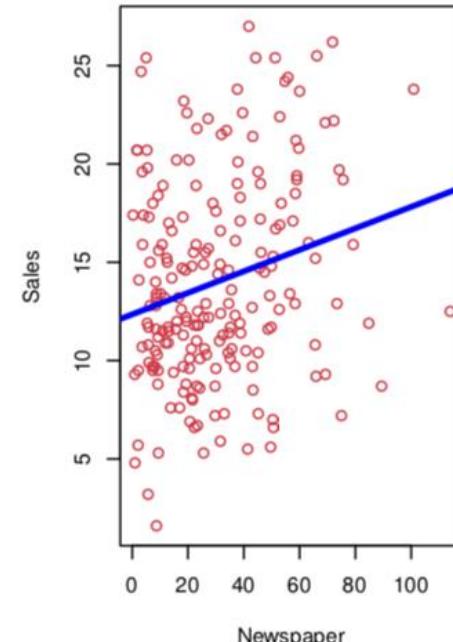
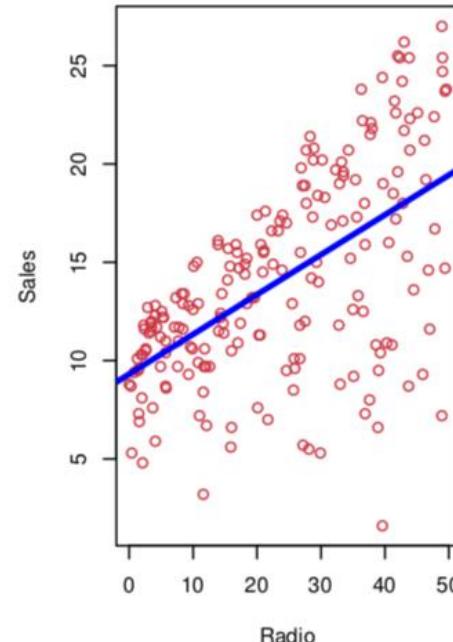
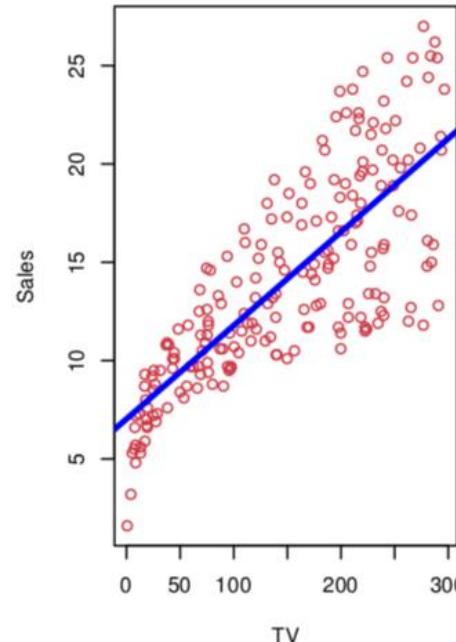
$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{y} - y)^2}{n}}$$

onde \hat{y} é o valor predito, y é o valor real e n o número de exemplos

É muito importante utilizar $(\hat{y} - y)^2$, caso contrário um erro negativo (valor predito menor que o real) cancelaria um positivo (valor predito maior que o real)

Modelos de Machine Learning

- Há alguma relação entre aumento de vendas e propaganda?
- Qual mídia contribui mais para as vendas?



Modelos de Machine Learning

- Para o exemplo do slide 5, foi obtida a seguinte equação do hiperplano:

$$Vendas = 2,939 + 0,046 \times TV + 0,189 \times radio + 0.01 \times Jornal$$

Modelos de Machine Learning

- Para o exemplo do slide 5, foi obtida a seguinte equação do hiperplano:

$$Vendas = 2,939 + 0,046 \times TV + 0,189 \times radio + 0.01 \times Jornal$$

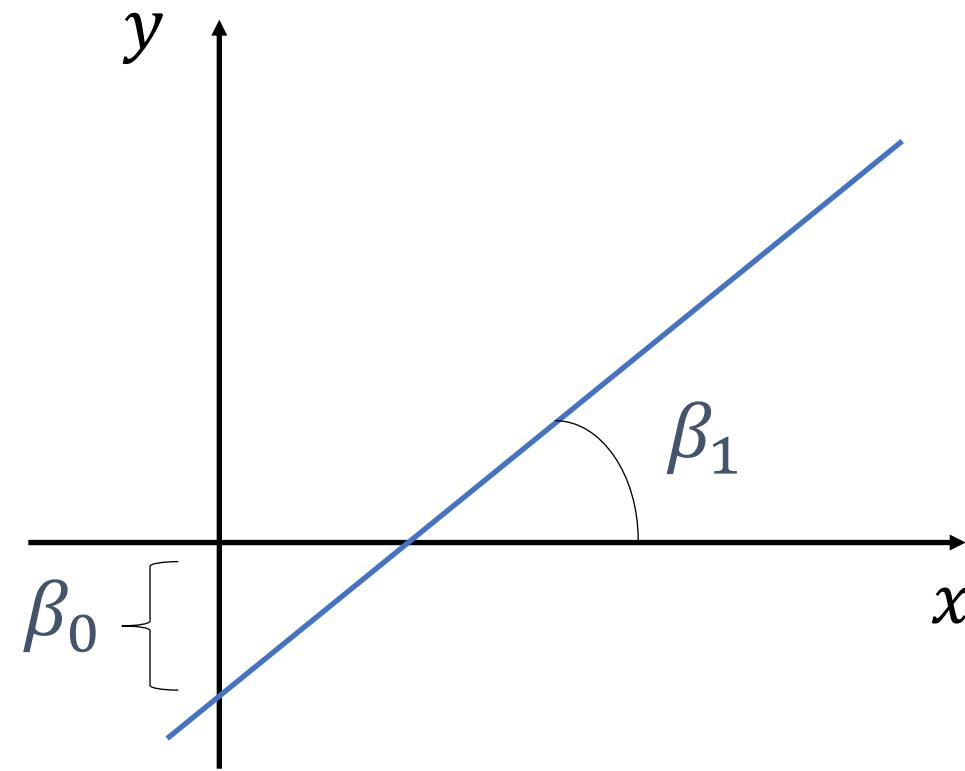
- caso nenhuma ação de propaganda seja feita as vendas serão de 2,939
- Mantendo todos os outros valores constantes, a cada uma unidade aumentada nas ações por TV, as vendas aumentam 0,046
- A influência da utilização de jornal é quase nula

Modelos de Machine Learning

Reta

- β_0 : deslocamento
- β_1 : inclinação

$$y = \beta_0 + \beta_1 x$$

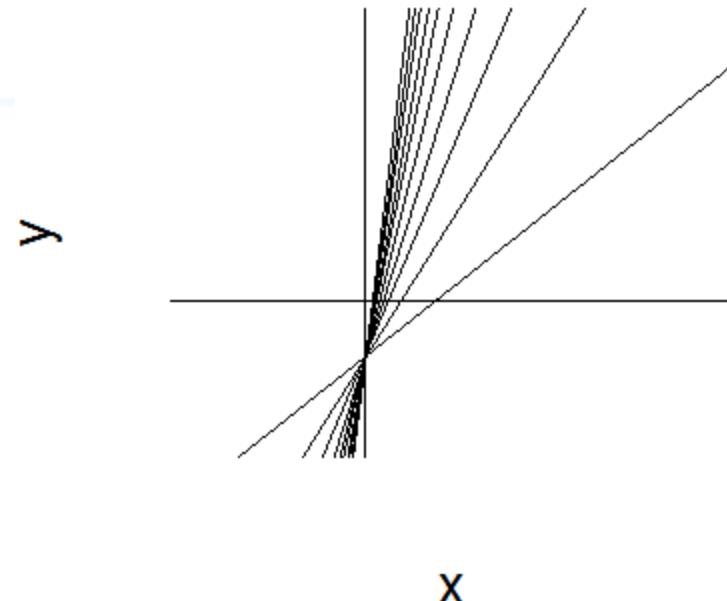


Modelos de Machine Learning

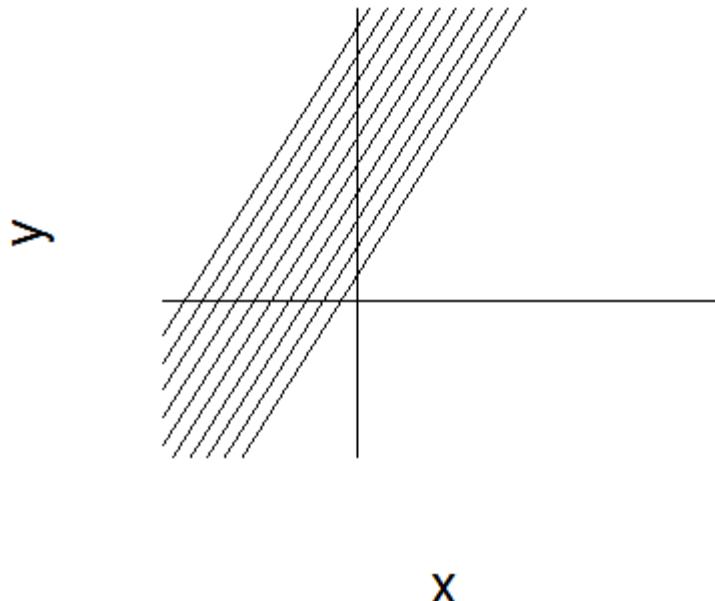
Reta

$$y = \beta_0 + \beta_1 x$$

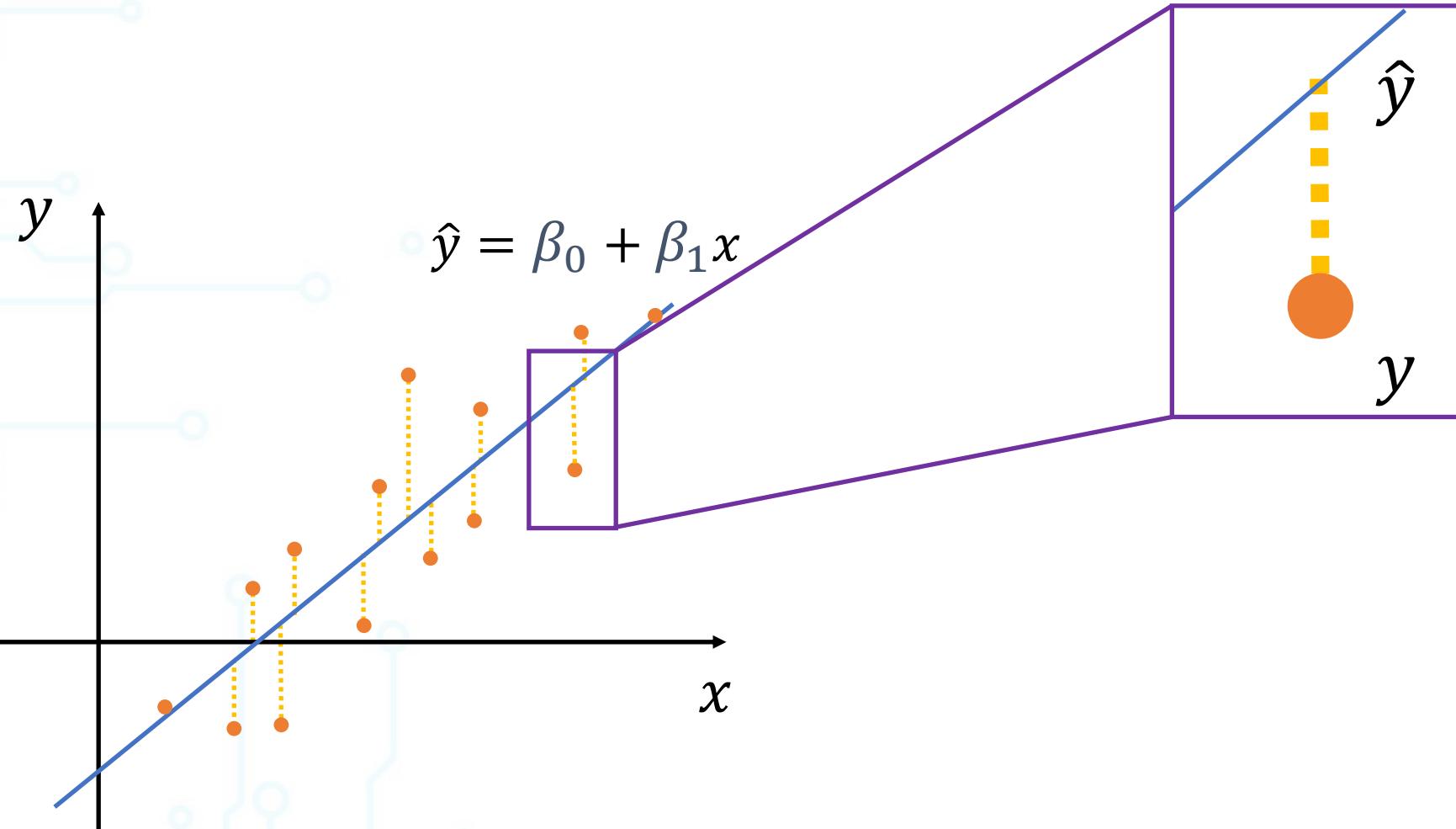
β_0 fixo; β_1 variável



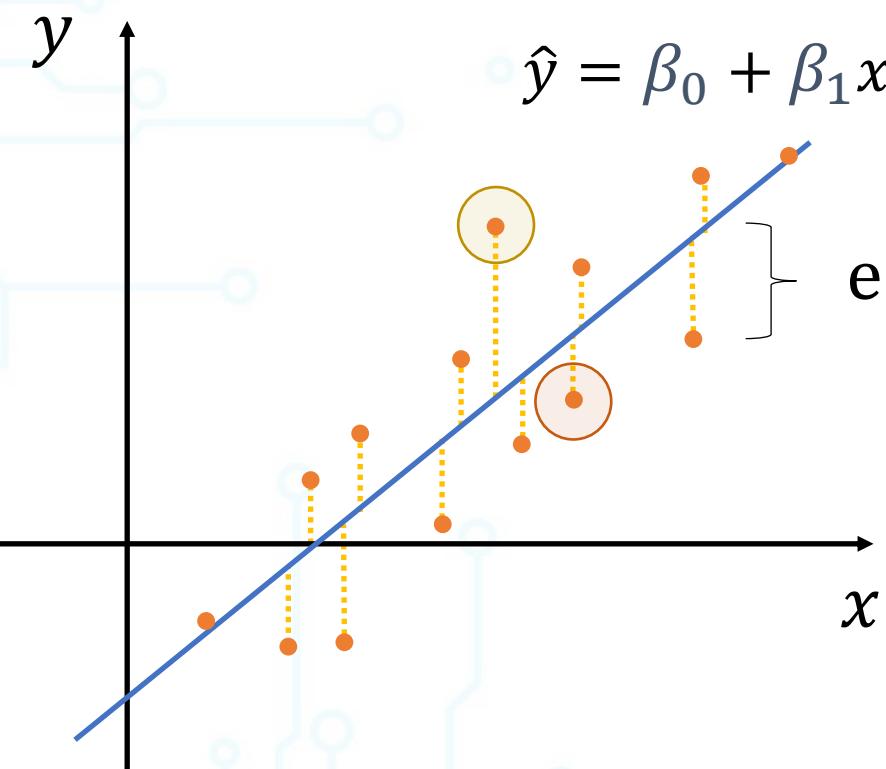
β_0 variável; β_1 fixo



Modelos de Machine Learning



Modelos de Machine Learning



$$e = y - \hat{y}$$
$$e = y - (\beta_0 + \beta_1 x)$$

Positivo

$$y - \hat{y} > 0$$

Negativo

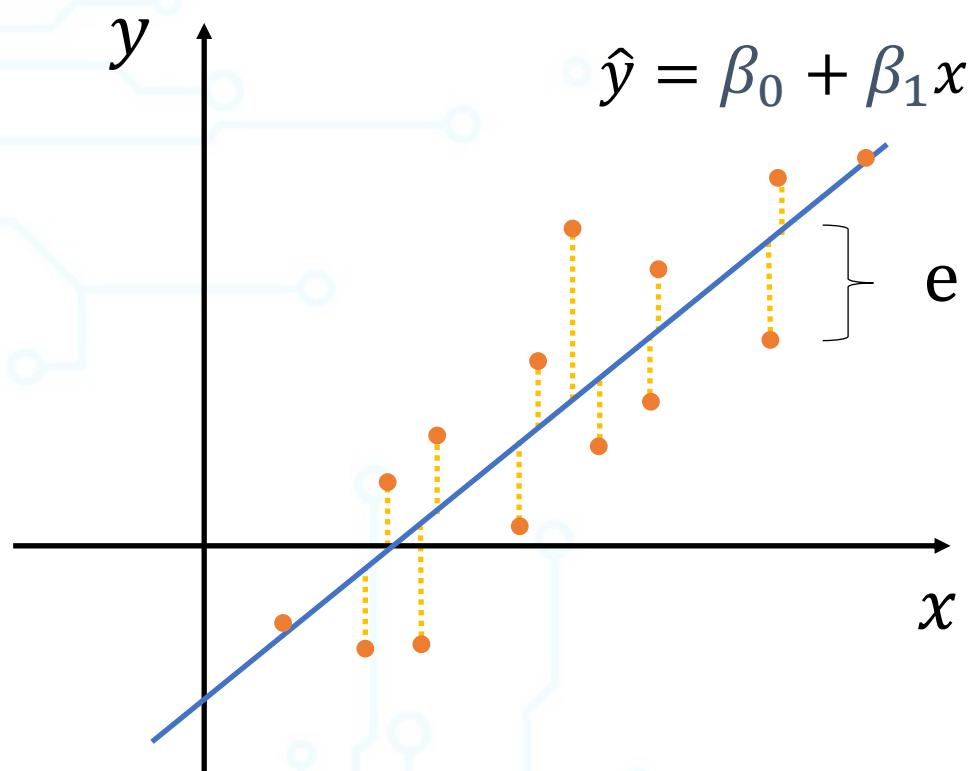
$$y - \hat{y} < 0$$

Solução?

$$(y - \hat{y})^2$$
$$|y - \hat{y}|$$



Modelos de Machine Learning



Como avaliar o erro total?

Erro quadrático
médio
R (MSE)

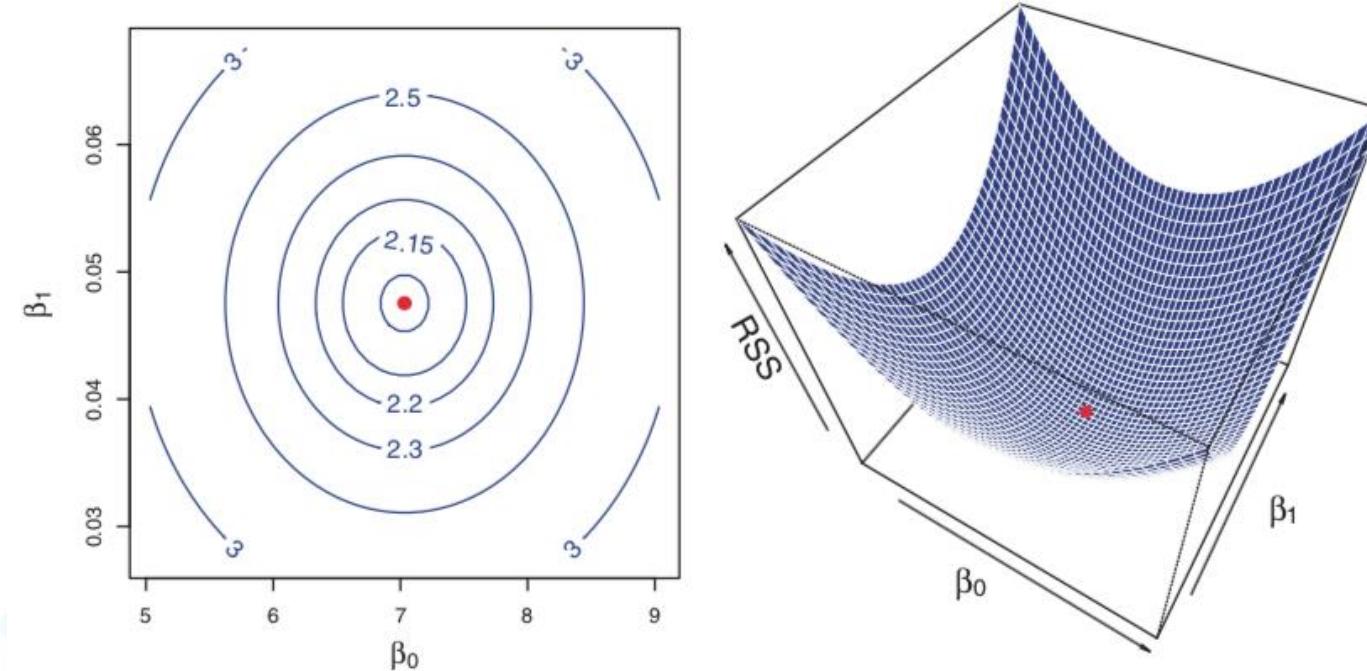
$$\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Erro absoluto
médio
MAE

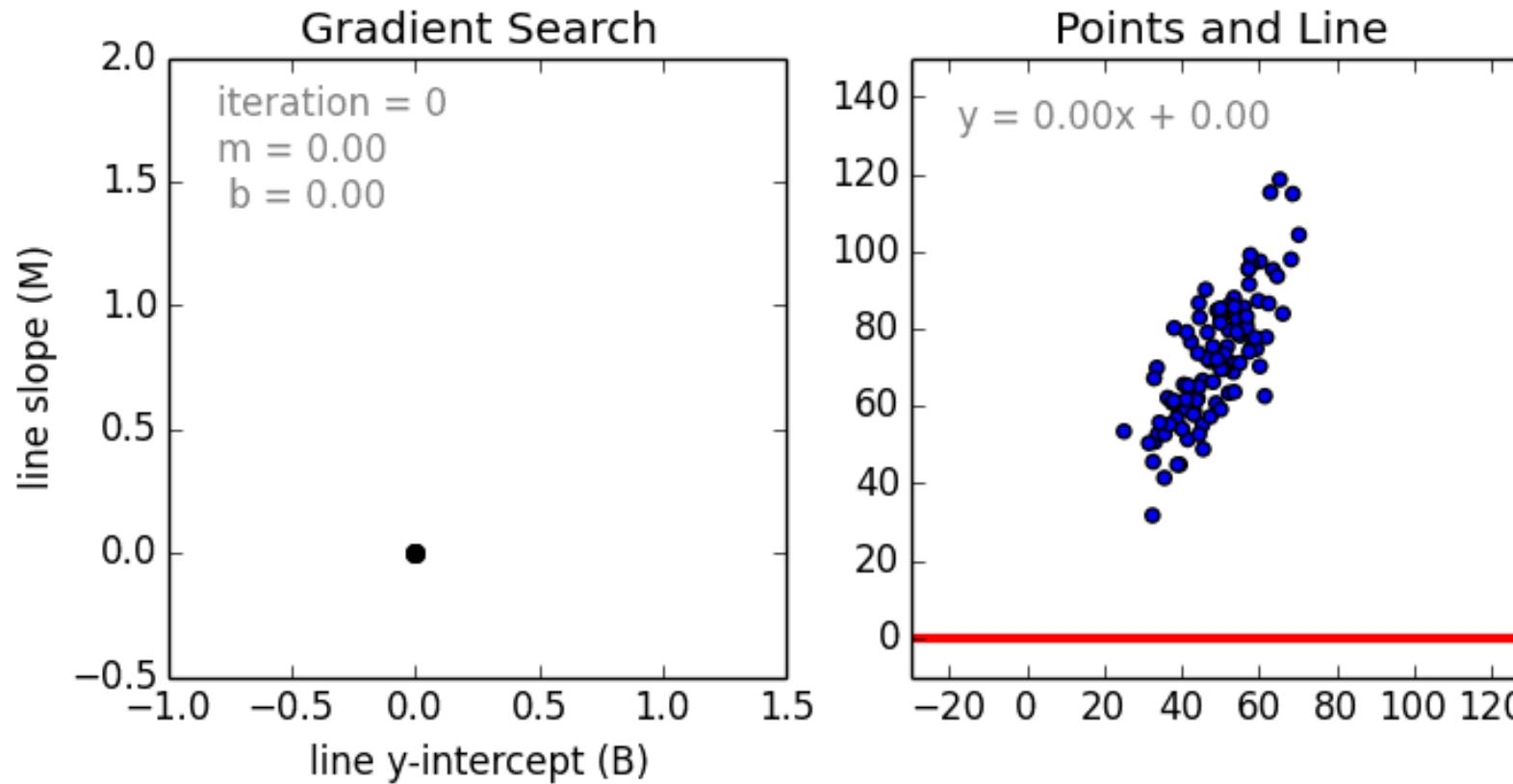
$$\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Modelos de Machine Learning

- Como a função RMSE é convexa, é possível encontrar o valor mínimo por meio de algoritmos de otimização



Modelos de Machine Learning



Modelos de Machine Learning

Peso Altura

	Pessoa 1	80 kg	163
---	----------	-------	-----

	Pessoa 2	85 kg	168
---	----------	-------	-----

	Pessoa 3	90 kg	175
---	----------	-------	-----

	Pessoa 4	95 kg	188
--	----------	-------	-----

Modelos de Machine Learning

Peso Altura

	Pessoa 1	80 kg	163
---	----------	-------	-----

$$\hat{y} = \beta_0 + \beta_1 X_1$$

	Pessoa 2	85 kg	168
---	----------	-------	-----

	Pessoa 3	90 kg	175
---	----------	-------	-----

	Pessoa 4	95 kg	188
--	----------	-------	-----

Modelos de Machine Learning

Peso Altura

	Pessoa 1	80 kg	163
---	----------	-------	-----

$$\hat{y} = \beta_0 + \beta_1 X_1$$

	Pessoa 2	85 kg	168
---	----------	-------	-----

$$\hat{y} = 17 + 1,8 \times 163$$

	Pessoa 3	90 kg	175
---	----------	-------	-----

	Pessoa 4	95 kg	188
--	----------	-------	-----

Modelos de Machine Learning

Peso Altura

	Pessoa 1	80 kg	163
---	----------	-------	-----

$$\hat{y} = \beta_0 + \beta_1 X_1$$

	Pessoa 2	85 kg	168
---	----------	-------	-----

$$\hat{y} = 17 + 1,8 \times 163$$

	Pessoa 3	90 kg	175
---	----------	-------	-----

$$\hat{y} = 310,4$$

	Pessoa 4	95 kg	188
--	----------	-------	-----

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$\hat{y} = 310,4$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

Modelos de Machine Learning

	Peso	Altura
	Pessoa 1	80 kg
	Pessoa 2	85 kg
	Pessoa 3	90 kg
	Pessoa 4	95 kg

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

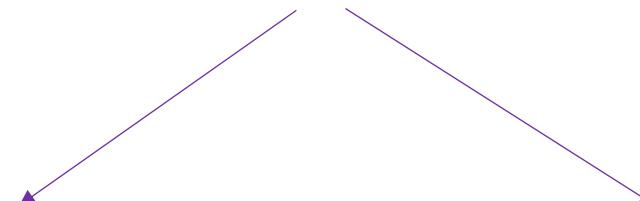
$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

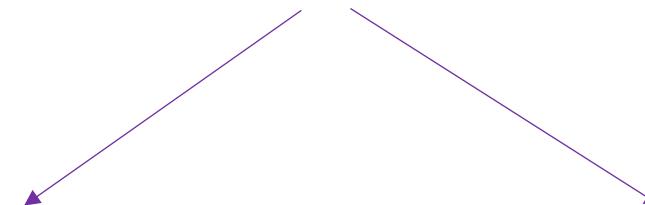
$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$



$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

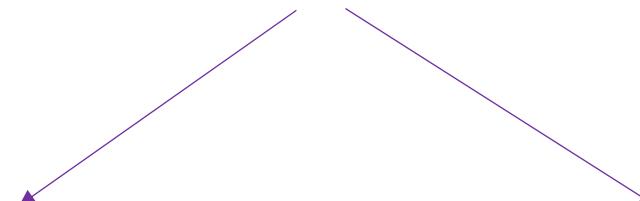
$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

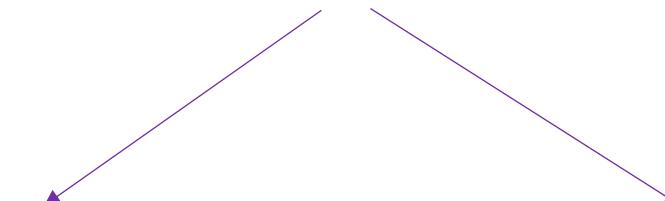
$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$MSE = (80 - 362,3)^2$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

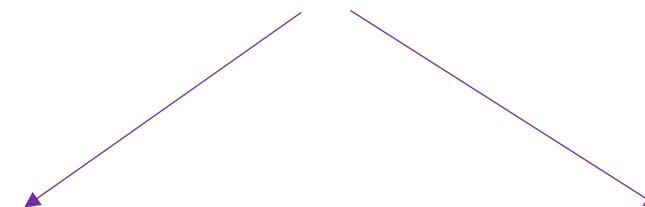
$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$



$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$MSE = (80 - 362,3)^2$$

$$MSE = 79.693,29$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

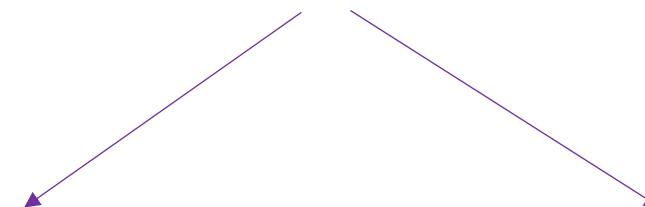
$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$



$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

$$MSE = 79.693,29$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

$$MSE = (80 - 258,5)^2$$

$$MSE = 79.693,29$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

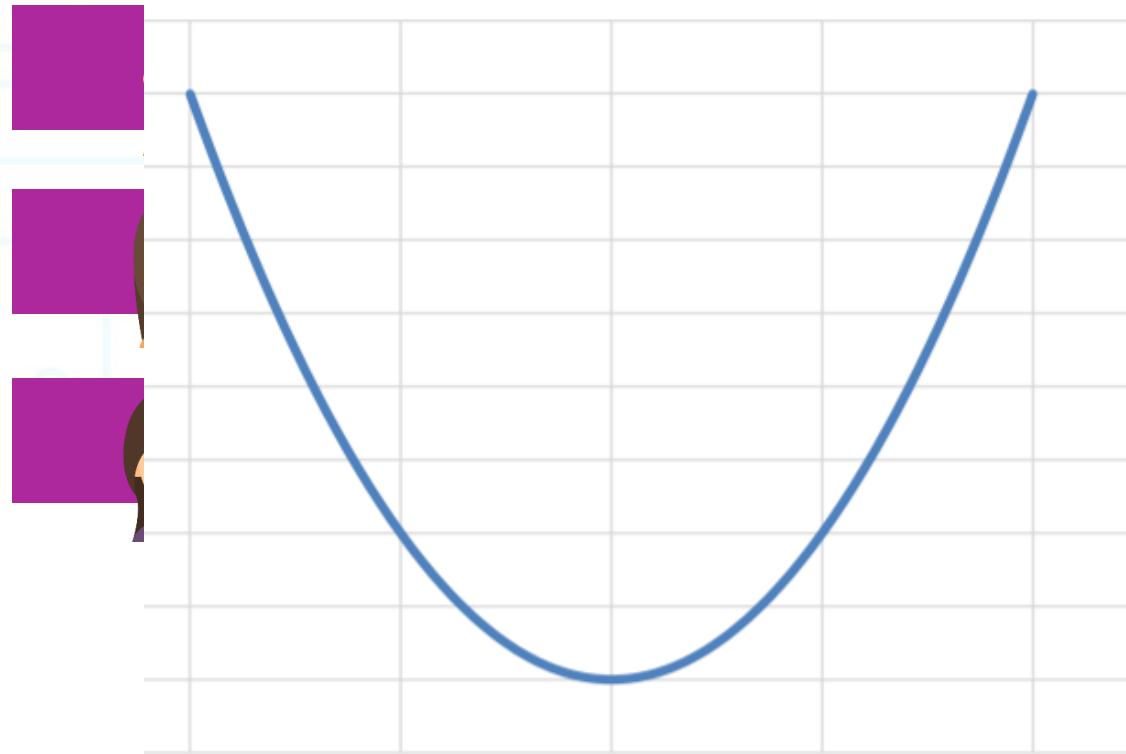
$$MSE = (80 - 258,5)^2$$

$$MSE = 79.693,29$$

$$MSE = 31.862,25$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163



$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

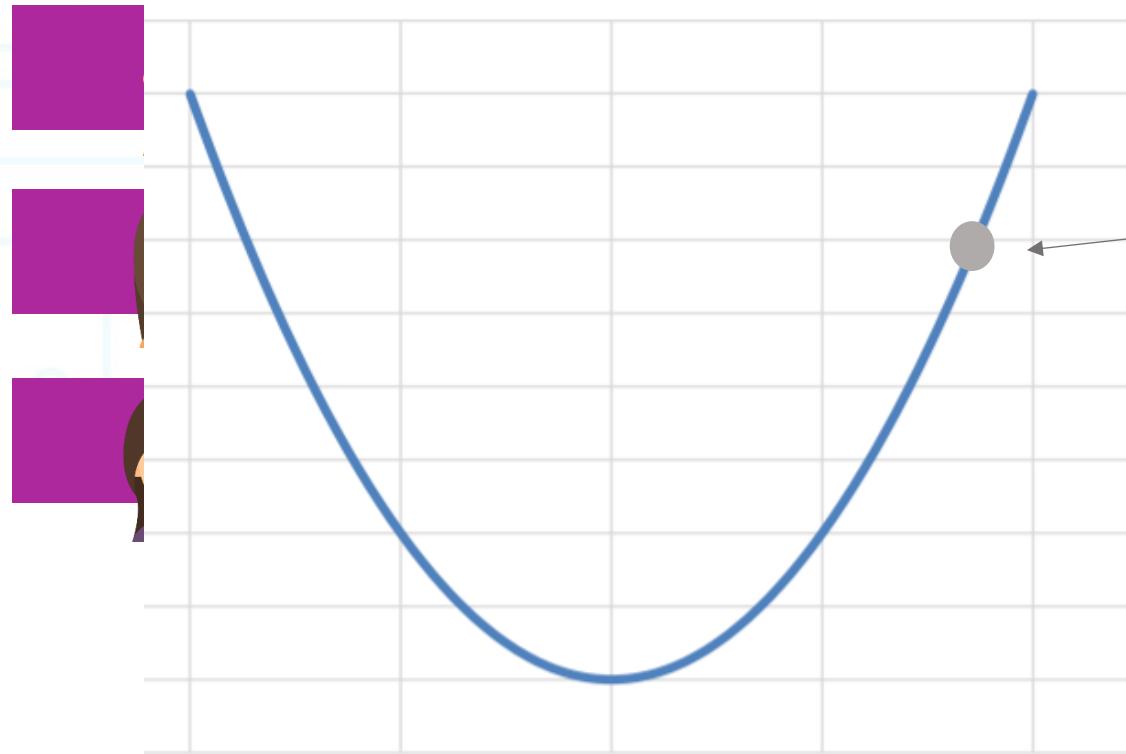
$$MSE = (80 - 258,5)^2$$

$$MSE = 79.693,29$$

$$MSE = 31.862,25$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163



$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

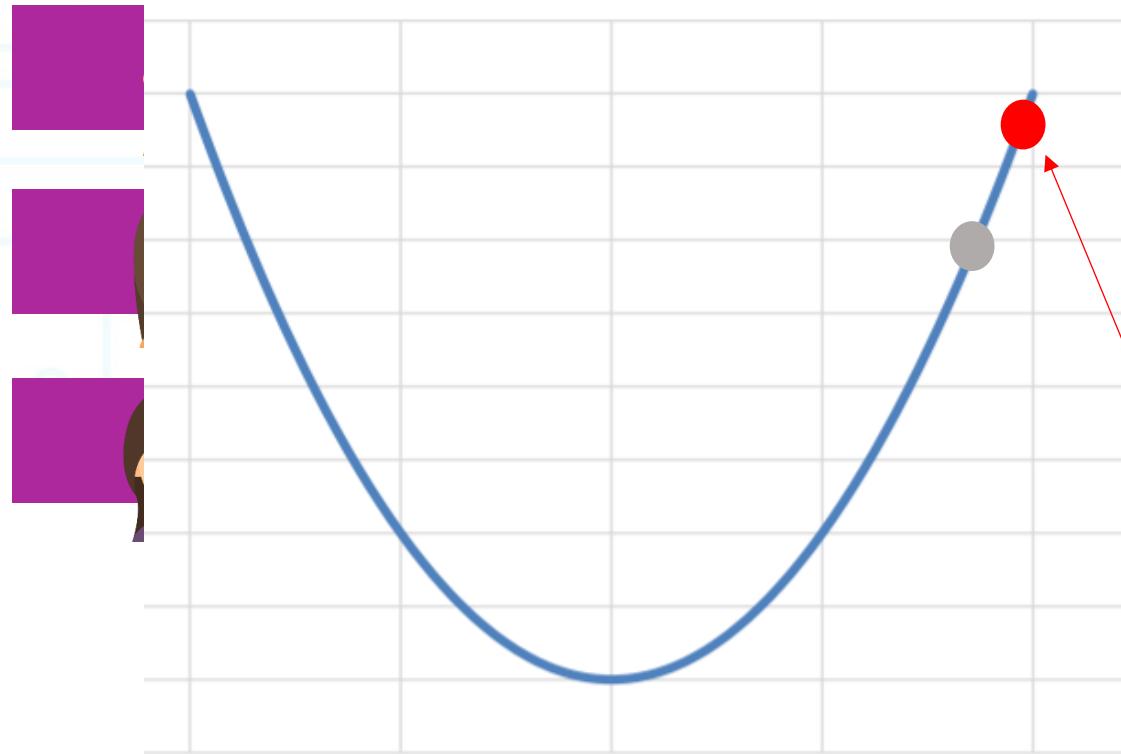
$$MSE = (80 - 258,5)^2$$

$$MSE = 79.693,29$$

$$MSE = 31.862,25$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163



$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

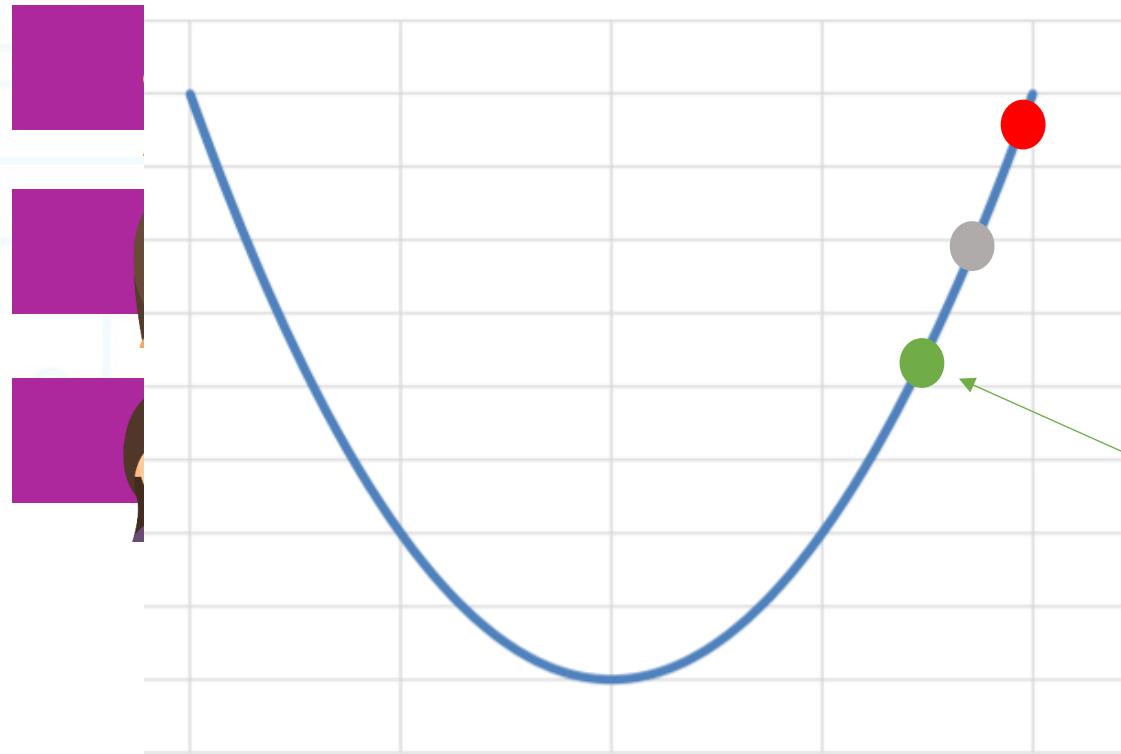
$$MSE = (80 - 258,5)^2$$

$$MSE = 79.693,29$$

$$MSE = 31.862,25$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163



$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 20 + 2,1 \times 163$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$\hat{y} = 362,3$$

$$\hat{y} = 258,5$$

$$MSE = (80 - 362,3)^2$$

$$MSE = (80 - 258,5)^2$$

$$MSE = 79.693,29$$

$$MSE = 31.862,25$$

Modelos de Machine Learning

	Peso	Altura	
Pessoa 1	80 kg	163	←
Pessoa 2	85 kg	168	
Pessoa 3	90 kg	175	
Pessoa 4	95 kg	188	

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$MSE = (80 - 258,5)^2$$

$$\hat{y} = 258,5$$

$$MSE = 31.862,25$$

Modelos de Machine Learning

	Peso	Altura
	Pessoa 1	80 kg 163
	Pessoa 2	85 kg 168
	Pessoa 3	90 kg 175
	Pessoa 4	95 kg 188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$MSE = (80 - 258,5)^2$$

$$\hat{y} = 258,5$$

$$MSE = 31.862,25$$

$$\hat{y} = 10 + 1,1 \times 168$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$MSE = (80 - 258,5)^2$$

$$\hat{y} = 258,5$$

$$MSE = 31.862,25$$

$$\hat{y} = 10 + 1,1 \times 168$$

$$\hat{y} = 194,8$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$MSE = (80 - 258,5)^2$$

$$\hat{y} = 258,5$$

$$MSE = 31.862,25$$

$$\hat{y} = 10 + 1,1 \times 168$$

$$MSE = (85 - 194,8)^2$$

$$\hat{y} = 194,8$$

Modelos de Machine Learning

	Peso	Altura
	Pessoa 1	80 kg 163
	Pessoa 2	85 kg 168
	Pessoa 3	90 kg 175
	Pessoa 4	95 kg 188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 163$$

$$MSE = (80 - 310,4)^2$$

$$\hat{y} = 310,4$$

$$MSE = 53.084,16$$

$$\hat{y} = 14 + 1,5 \times 163$$

$$MSE = (80 - 258,5)^2$$

$$\hat{y} = 258,5$$

$$MSE = 31.862,25$$

$$\hat{y} = 10 + 1,1 \times 168$$

$$MSE = (85 - 194,8)^2$$

$$\hat{y} = 194,8$$

$$MSE = 12.056,04$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$\hat{y} = 147$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$



Modelos de Machine Learning

	Peso	Altura
 Pessoa 1	80 kg	163
 Pessoa 2	85 kg	168
 Pessoa 3	90 kg	175
 Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$\hat{y} = 97$$

Modelos de Machine Learning

	Peso	Altura
	Pessoa 1	80 kg 163
	Pessoa 2	85 kg 168
	Pessoa 3	90 kg 175
	Pessoa 4	95 kg 188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$



Modelos de Machine Learning

	Peso	Altura
	Pessoa 1	80 kg 163
	Pessoa 2	85 kg 168
	Pessoa 3	90 kg 175
	Pessoa 4	95 kg 188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

Modelos de Machine Learning

	Peso	Altura
	Pessoa 1	80 kg 163
	Pessoa 2	85 kg 168
	Pessoa 3	90 kg 175
	Pessoa 4	95 kg 188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$\hat{y} = 38,6$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163

Pessoa 2	85 kg	168
----------	-------	-----

Pessoa 3	90 kg	175
----------	-------	-----

Pessoa 4	95 kg	188
----------	-------	-----

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 7 + 0,8 \times 175$$

$$MSE = (90 - 147)^2$$

$$\hat{y} = 147$$

$$MSE = 3.249$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 1 + 0,2 \times 188$$

$$MSE = (95 - 38,6)^2$$

$$\hat{y} = 38,6$$

$$MSE = 3.180,97$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 84,5 \quad MSE = 20,25$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 84,5 \quad MSE = 20,25$$

$$\hat{y} = 87 \quad MSE = 4$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

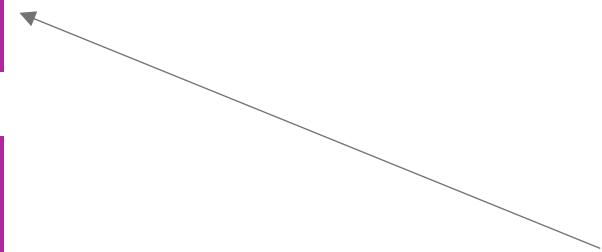
$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 84,5 \quad MSE = 20,25$$

$$\hat{y} = 87 \quad MSE = 4$$

$$\hat{y} = 90,5 \quad MSE = 0,25$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 84,5 \quad MSE = 20,25$$

$$\hat{y} = 87 \quad MSE = 4$$

$$\hat{y} = 90,5 \quad MSE = 0,25$$

$$\hat{y} = 97 \quad MSE = 4$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

$$MSE = 4$$

$$\hat{y} = 84,5 \quad MSE = 20,25$$

$$\hat{y} = 87 \quad MSE = 4$$

$$\hat{y} = 90,5 \quad MSE = 0,25$$

$$\hat{y} = 97 \quad MSE = 4$$

$$MSE = 7,125$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,5 \times 188$$

$$MSE = (95 - 97)^2$$

$$\hat{y} = 97$$

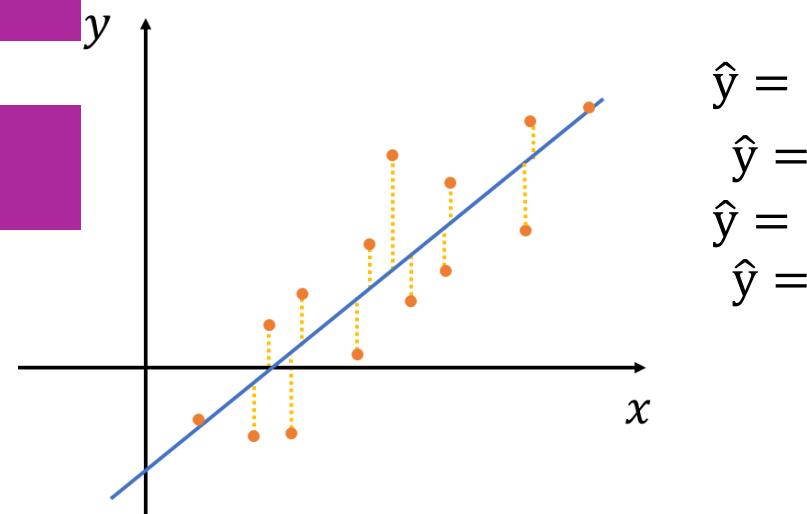
$$MSE = 4$$

$$\hat{y} = 84,5 \quad MSE = 20,25$$

$$\hat{y} = 87 \quad MSE = 4$$

$$\hat{y} = 90,5 \quad MSE = 0,25$$

$$\hat{y} = 97 \quad MSE = 4$$



$$MSE = 7,125$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 3$$

$$\beta_1 = 0,5$$

$$MSE = 7,125$$

$$\hat{y} = 3 + 0,5 X_1$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	?? kg	158

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 3$$

$$\beta_1 = 0,5$$

$$MSE = 7,125$$

$$\hat{y} = 3 + 0,5 X_1$$

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	?? kg	158

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	?? kg	158

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	?? kg	158

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	?? kg	158

$$\hat{y} = \beta_0 + \beta_1 X_1$$

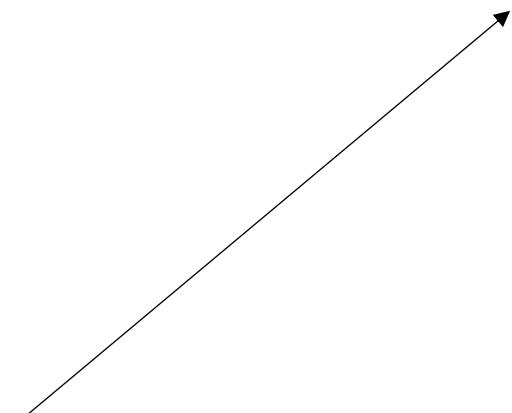
$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 3$$

$$\beta_1 = 0,5$$

$$MSE = 7,125$$

$$\hat{y} = 3 + 0,5 X_1$$



Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	?? kg	158

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 3$$

$$\beta_1 = 0,5$$

$$MSE = 7,125$$

$$\hat{y} = 3 + 0,5 X_1$$

$$\hat{y} = 3 + 0,5 \times 158$$

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	82 kg	158

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 3$$

$$\beta_1 = 0,5$$

$$MSE = 7,125$$

$$\hat{y} = 3 + 0,5 X_1$$

$$\hat{y} = 3 + 0,5 \times 158$$

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	82 kg	158

Modelos de Machine Learning

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	82 kg	158
Pessoa 6	?? kg	163

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 3$$

$$\beta_1 = 0,5$$

$$MSE = 7,125$$

$$\hat{y} = 3 + 0,5 X_1$$

$$\hat{y} = 3 + 0,5 \times 163$$

84,5

Implementação de Exemplo

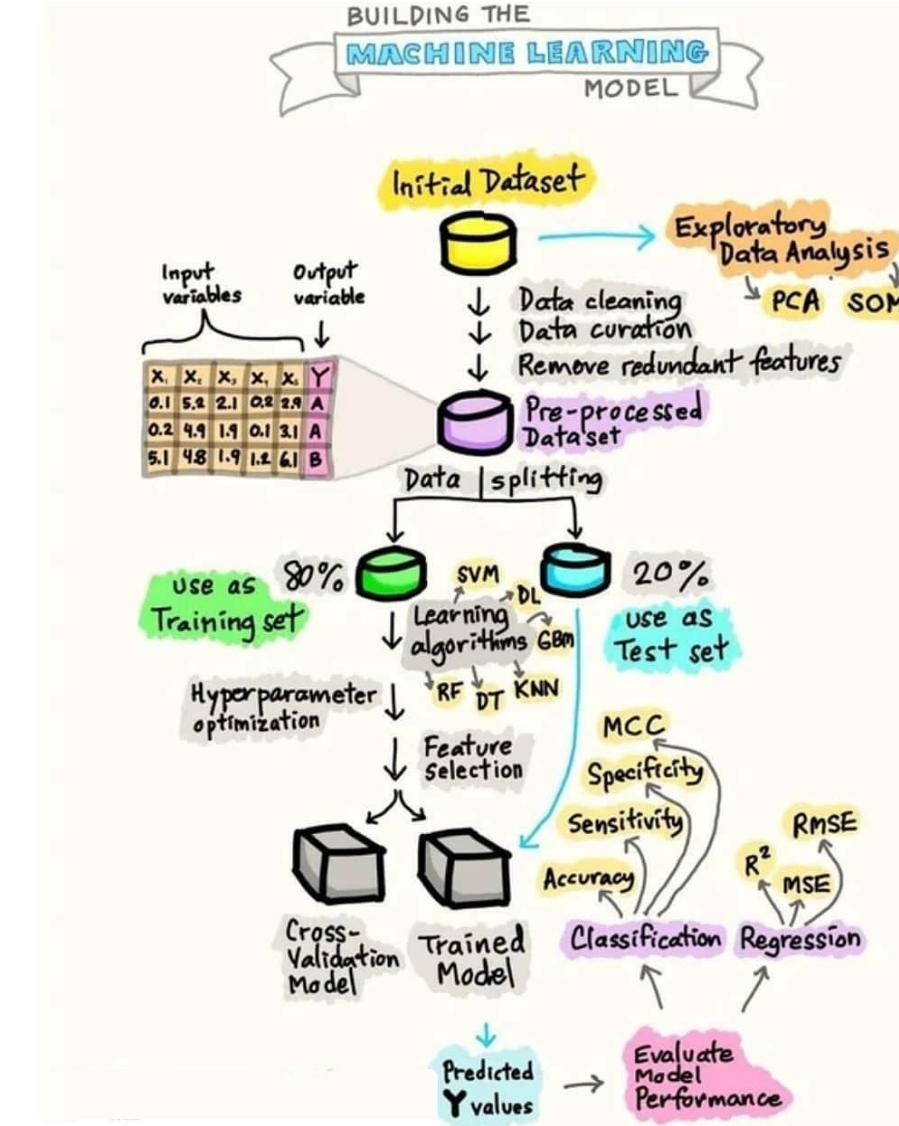
“

Classificação

Classificação



Classificação



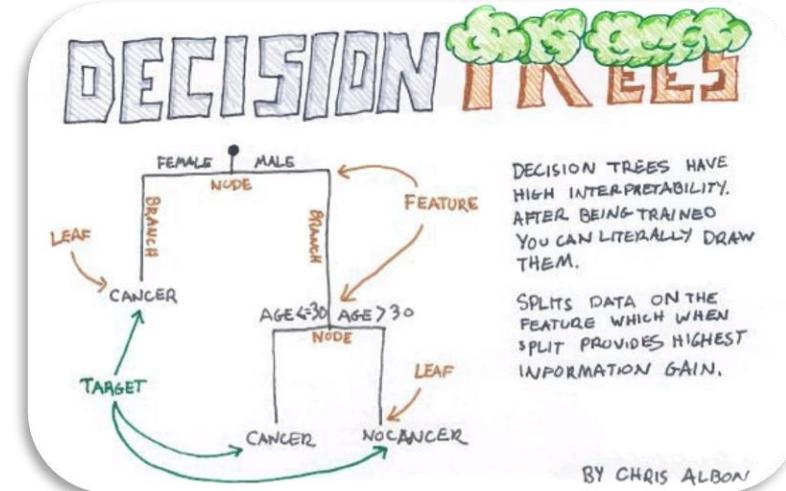
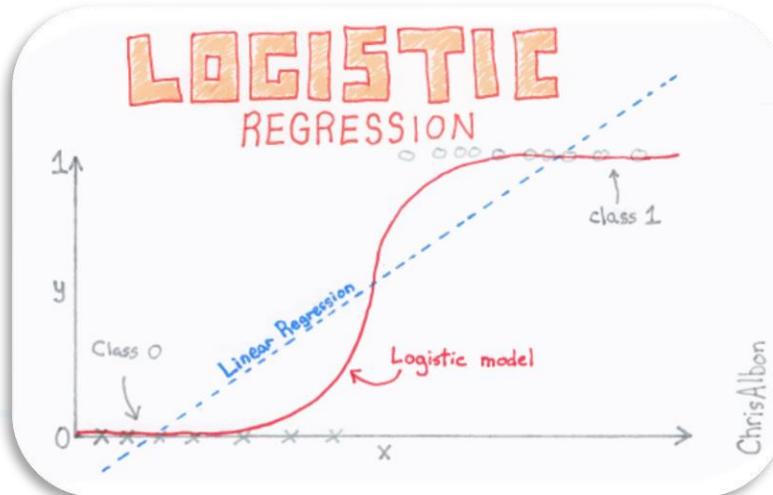
Classificação



97%



Classificação



GAUSSIAN NAIVE BAYES CLASSIFIER

"Gaussian" because this is a normal distribution

$$P(\text{class} | \text{data}) = \frac{P(\text{data} | \text{class}) \times p(\text{class})}{P(\text{data})}$$

We don't calculate this in naive bayes classifiers

This is our prior belief

ChrisAlbon

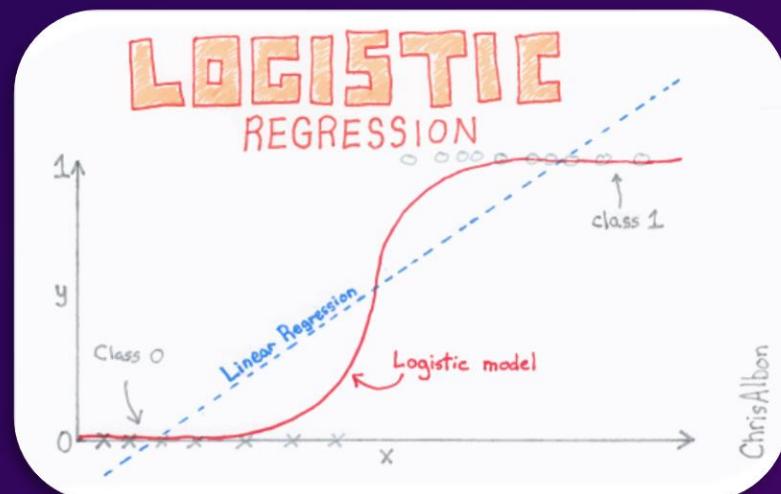
KNN NEIGHBORHOOD SIZE

Small K = Low Bias, High Variance

Large K = High Bias, Low Variance

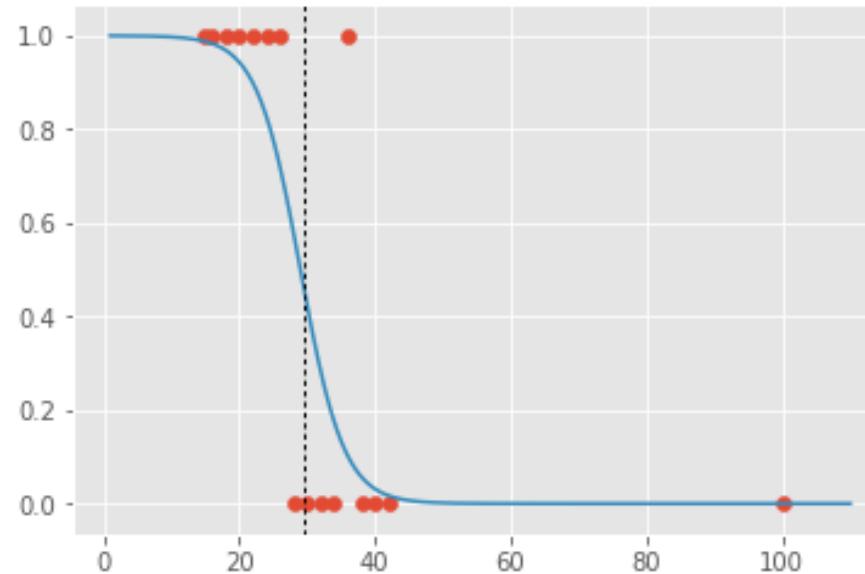
BY CHRIS ALBON

Regressão Logística

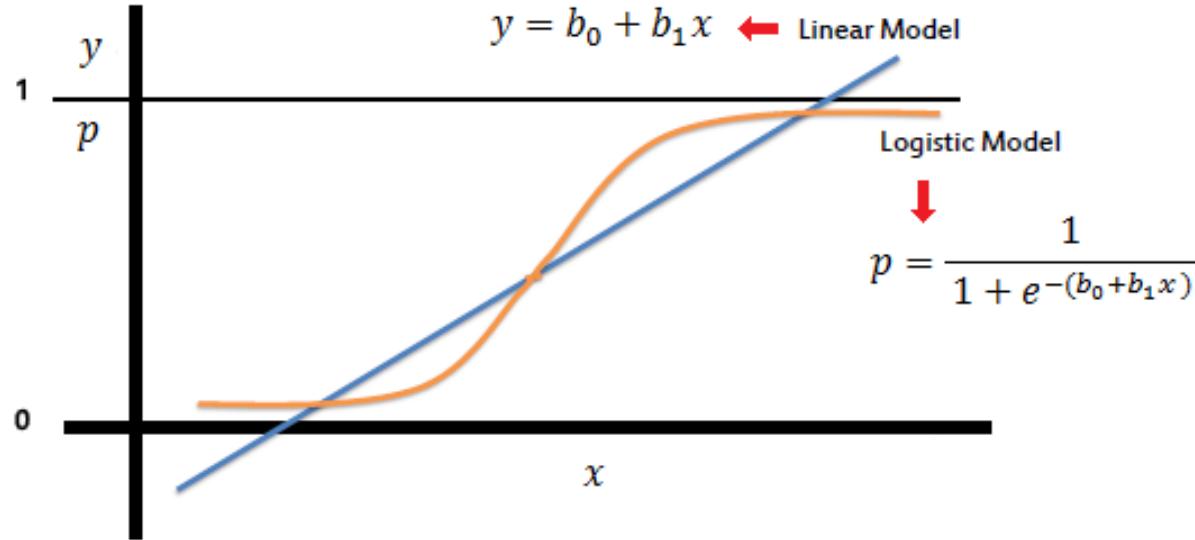


Regressão Logística

- A intuição por trás de regressão logística é bastante simples: em vez de acharmos a reta que melhor se ajusta aos dados, vamos achar uma curva em formato de 'S' que melhor se ajusta aos dados:



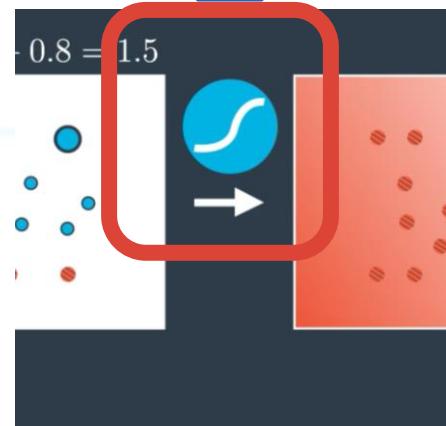
Regressão Logística



A regressão logística é semelhante a uma regressão linear, mas a curva é construída usando o logaritmo natural das “probabilidades” da variável-alvo.

Regressão Logística

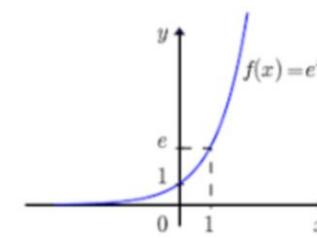
Redes Neurais



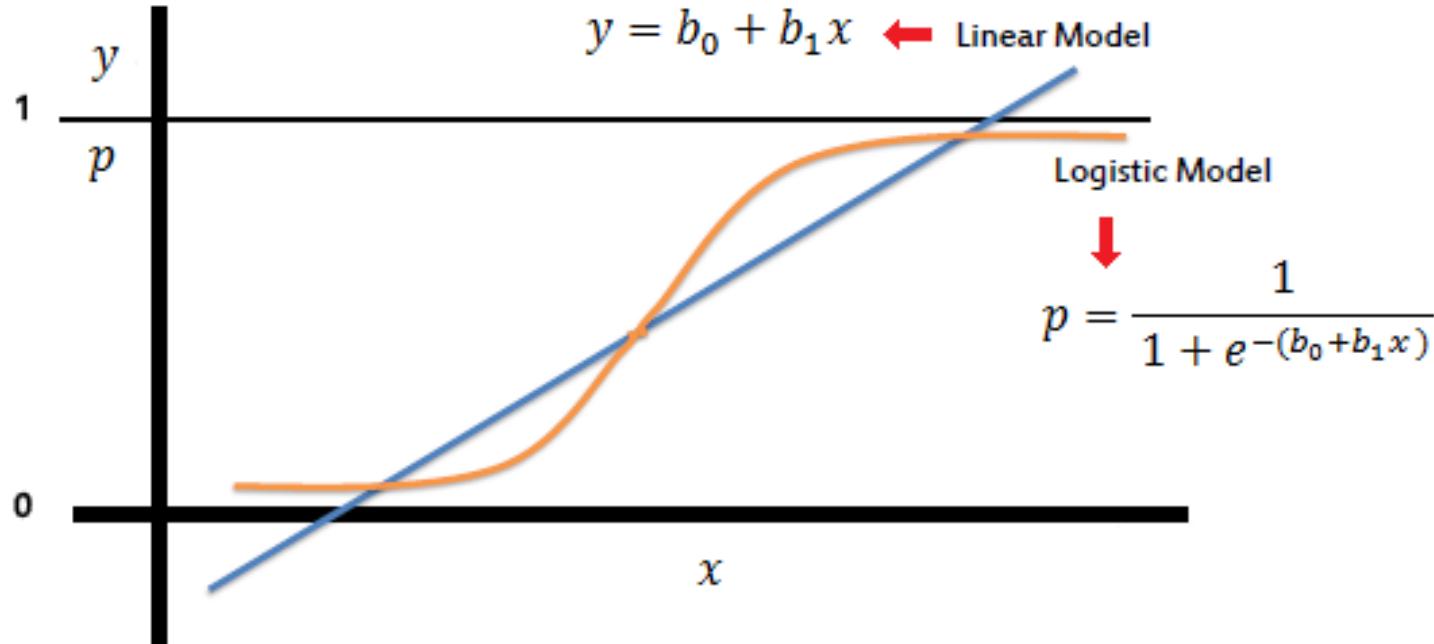
$$f(x) = \frac{1}{1 + e^{-x}} \text{ para todo } x \text{ real.}$$



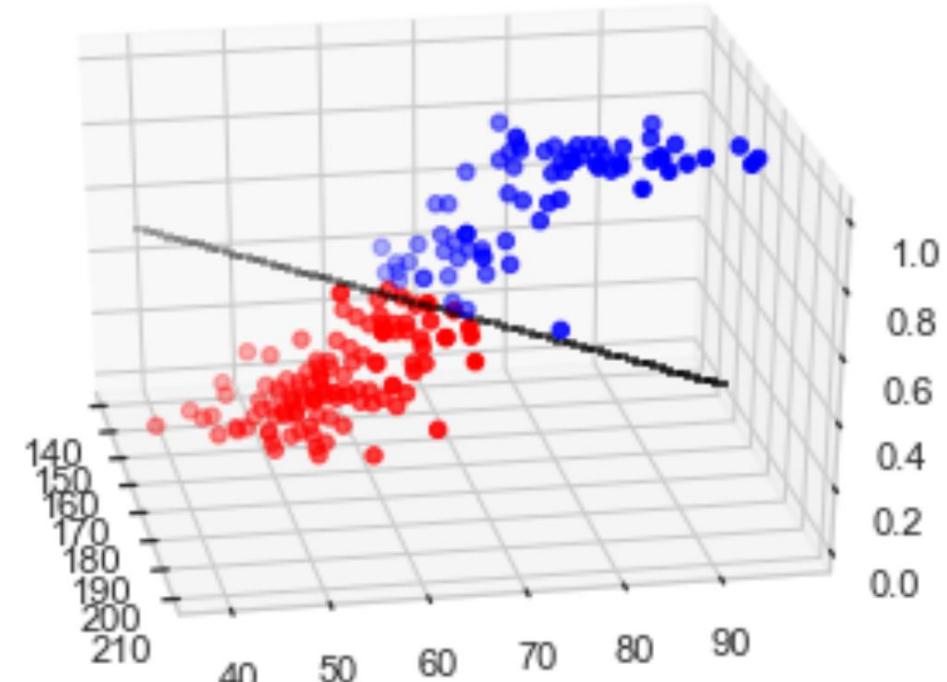
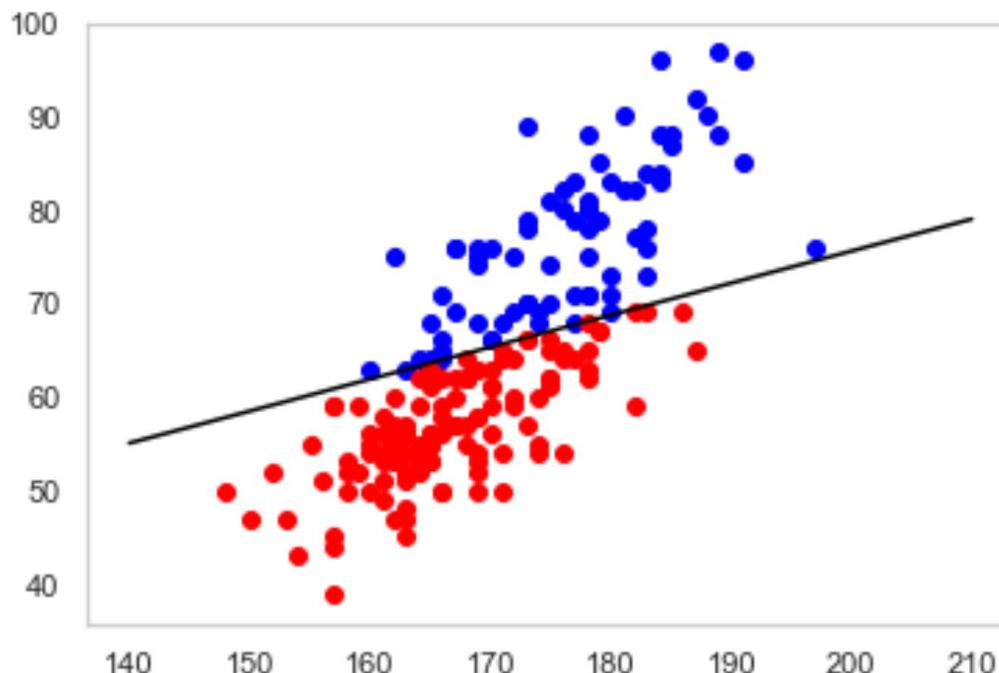
A função **exponencial** natural, denotada e^x ou $\exp(x)$ é a função **exponencial** cuja base é o número de Euler (um número irracional que **vale** aproximadamente 2,718281828).



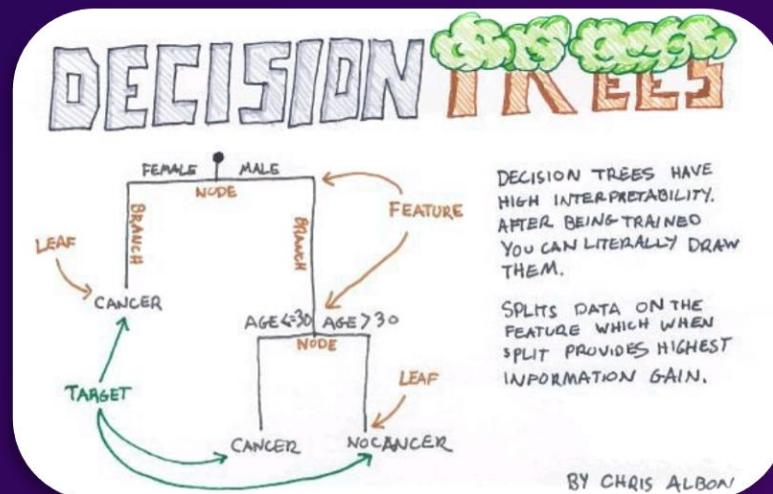
Regressão Logística



Regressão Logística



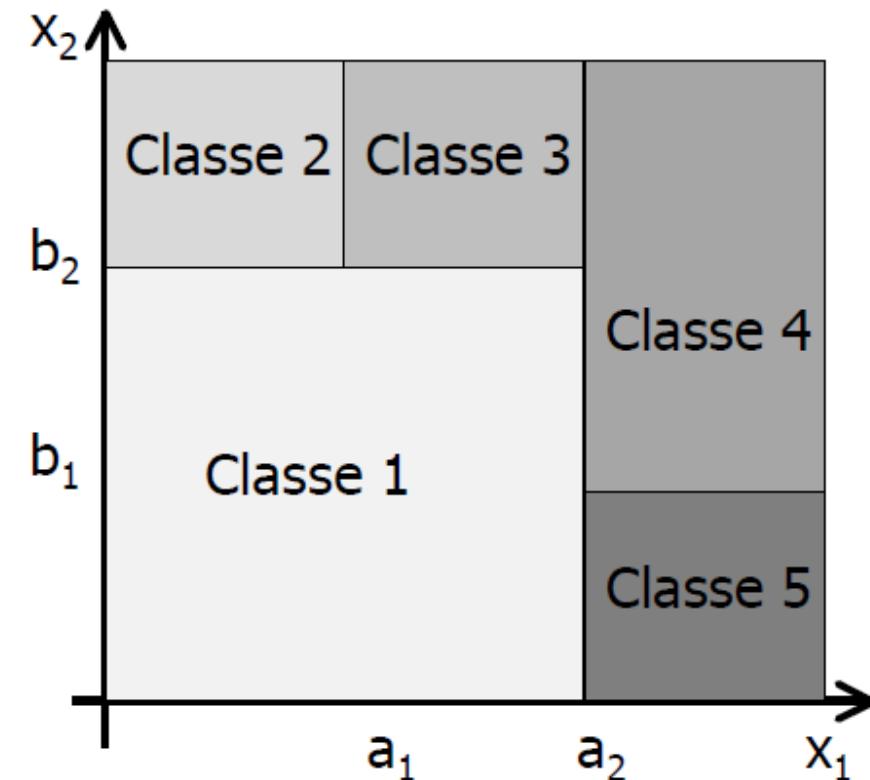
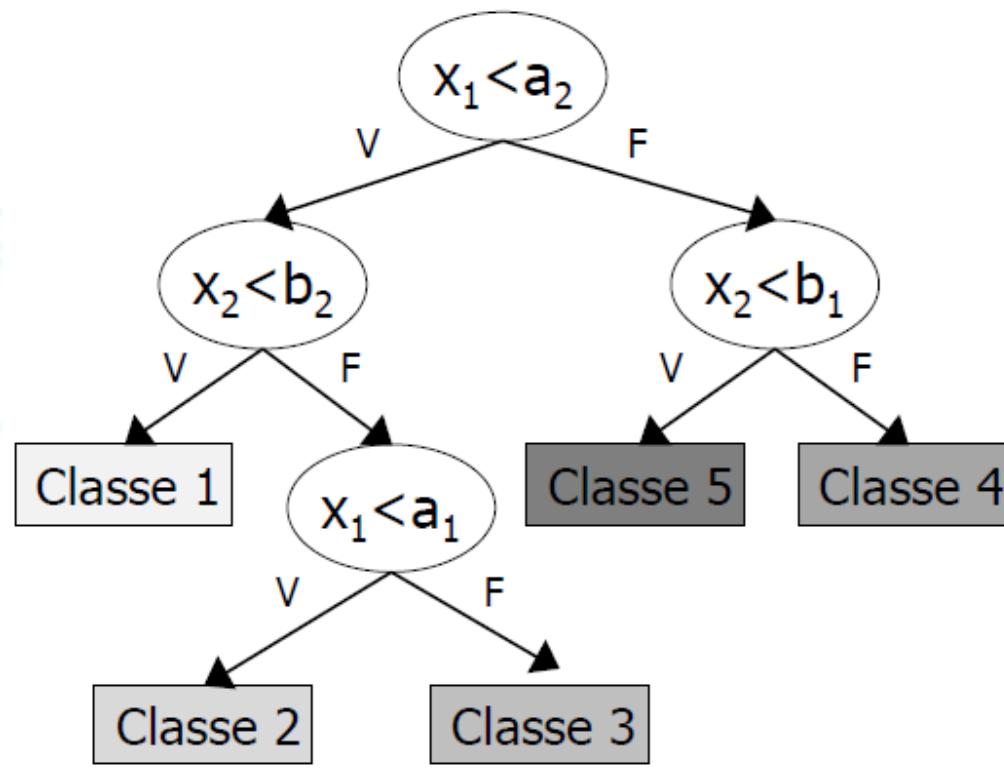
Arvore de Decisão



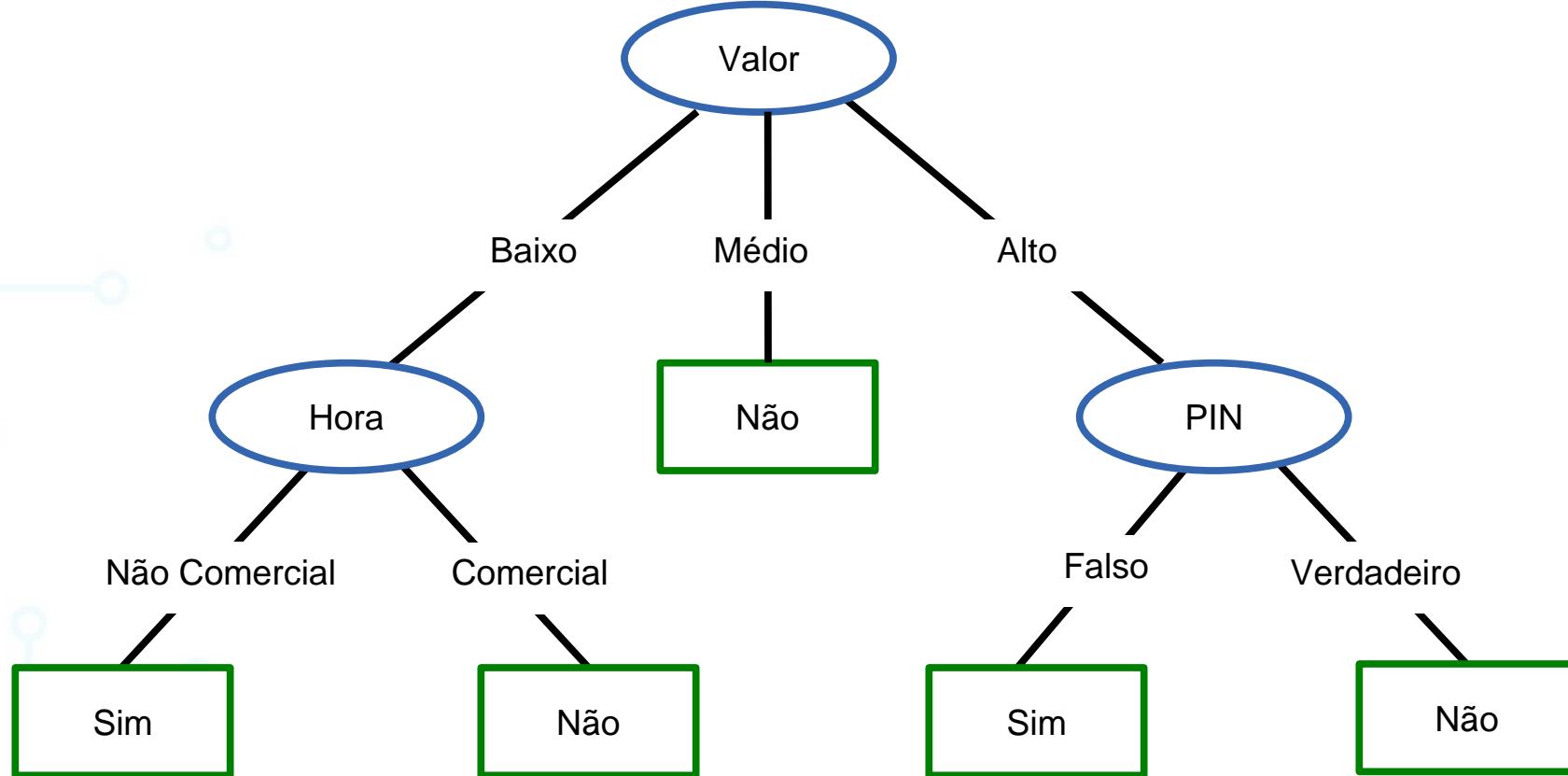
Árvore de Decisão

- Uma das formas de aprendizado indutivo mais utilizadas
 - Aplicações práticas
 - Tarefas de Classificação (Aprendizado supervisionado)
- Aproximação de funções discretas robusta a ruídos
- Aprendizado de expressões disjuntivas

Representação de uma AD



Árvore de Decisão



Árvore de Decisão

- ID3
 - Cria um ranking dos atributos mais adequados
 - Insere (com base no ranking) os atributos na árvore
 - Utiliza o ganho de informação (information gain) para criar o ranking
 - Método com base na entropia

Árvore de Decisão

- Entropia
 - Medida definida na teoria da informação
 - Define a pureza de um conjunto de instâncias
 - Precisa de um conjunto de instâncias positivas e negativas

$$\text{Entropia}(S) = - (p_+ \log_2 p_+) - (p_- \log_2 p_-)$$

p_+ : proporção de instâncias positivas

p_- : proporção de instâncias negativas

Árvore de Decisão

- O ganho de informação de um atributo é a redução esperada na entropia, causada pelo particionamento das instâncias de acordo com este atributo. Ou seja, o quanto se espera que a entropia seja reduzida quando se sabe o valor do atributo A.

$$\text{Ganho}(S, A) = \text{Entropia}(S) - \sum_{v \in \text{valores}(A)} \frac{|S_v|}{|S|} \text{Entropia}(S_v)$$

Valores(A): conjunto de todos os valores possíveis do atributo A

S_v: subconjunto de S, no qual o atributo A possui valor v.

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

values(PIN) = Verdadeiro, Falso

$$S = [9+, 5-]$$

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

$\text{values(PIN)} = \text{Verdadeiro}, \text{ Falso}$

$S = [9+, 5-]$

$S_{\text{verdadeiro}} = [6+, 2-]$

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

$\text{values(PIN)} = \text{Verdadeiro}, \text{ Falso}$

$$S = [9+, 5-]$$

$$S_{\text{verdadeiro}} = [6+, 2-]$$

$$S_{\text{Falso}} = [3+, 3-]$$

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

$\text{values(PIN)} = \text{Verdadeiro}, \text{ Falso}$

$$S = [9+, 5-]$$

$$S_{\text{verdadeiro}} = [6+, 2-]$$

$$S_{\text{Falso}} = [3+, 3-]$$

$$\text{Gain}(S, \text{PIN}) = \text{Entropia}(S) - \sum_F^V \left(\frac{Sv}{S} \right) \text{Entropia}(Sv)$$

$$= \text{Entropia}(S) - (8/14)\text{Entropia}(S_{\text{verdadeiro}}) - (6/14)\text{Entropia}(S_{\text{Falso}})$$

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

$\text{values(PIN)} = \text{Verdadeiro}, \text{ Falso}$

$$S = [9+, 5-]$$

$$S_{\text{verdadeiro}} = [6+, 2-]$$

$$S_{\text{Falso}} = [3+, 3-]$$

$$\text{Gain}(S, \text{PIN}) = \text{Entropia}(S) - \sum_F^V \left(\frac{Sv}{S} \right) \text{Entropia}(Sv)$$

$$= \text{Entropia}(S) - (8/14)\text{Entropia}(S_{\text{verdadeiro}}) - (6/14)\text{Entropia}(S_{\text{Falso}})$$

$$\text{Entropia}(S) = - (9 \log_2 9) - (5 \log_2 5)$$

$$\text{Entropia}(S_{\text{verdadeiro}}) = - (6 \log_2 6) - (2 \log_2 2)$$

$$\text{Entropia}(S_{\text{Falso}}) = - (3 \log_2 3) - (3 \log_2 3)$$

Árvore de Decisão

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

$\text{values(PIN)} = \text{Verdadeiro}, \text{ Falso}$

$$S = [9+, 5-]$$

$$S_{\text{verdadeiro}} = [6+, 2-]$$

$$S_{\text{Falso}} = [3+, 3-]$$

$$\text{Gain}(S, \text{PIN}) = \text{Entropia}(S) - \sum_F^V \left(\frac{Sv}{S} \right) \text{Entropia}(Sv)$$

$$= \text{Entropia}(S) - (8/14)\text{Entropia}(S_{\text{verdadeiro}}) - (6/14)\text{Entropia}(S_{\text{Falso}})$$

$$\text{Entropia}(S) = - (9 \log_2 9) - (5 \log_2 5)$$

$$\text{Entropia}(S_{\text{verdadeiro}}) = - (6 \log_2 6) - (2 \log_2 2)$$

$$\text{Entropia}(S_{\text{Falso}}) = - (3 \log_2 3) - (3 \log_2 3)$$

$$= 0.048$$

Árvore de Decisão

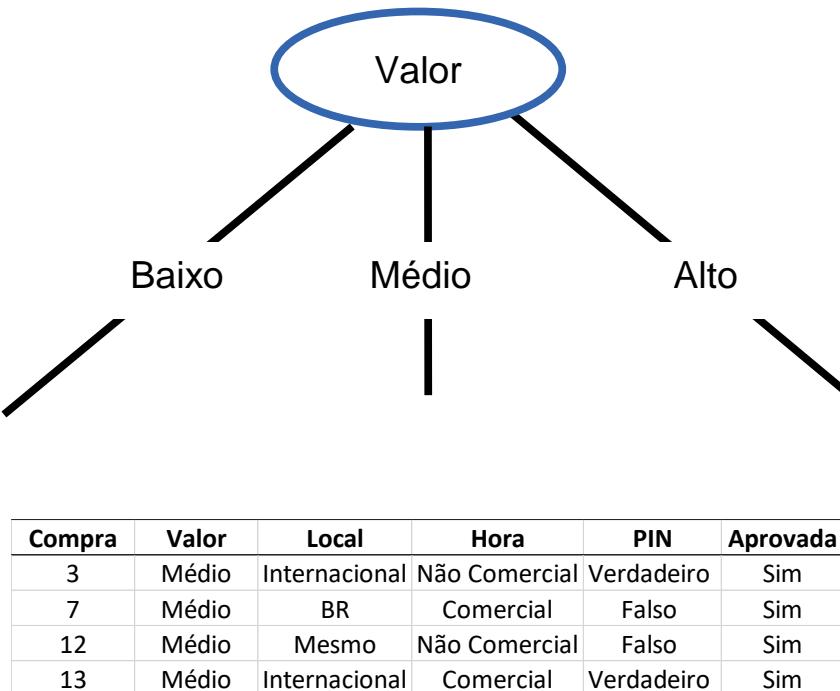
$\text{Gain}(S, \text{ valor}) = 0.246$

$\text{Gain}(S, \text{ Hora}) = 0.151$

$\text{Gain}(S, \text{ PIN}) = 0.048$

$\text{Gain}(S, \text{ Local}) = 0.029$

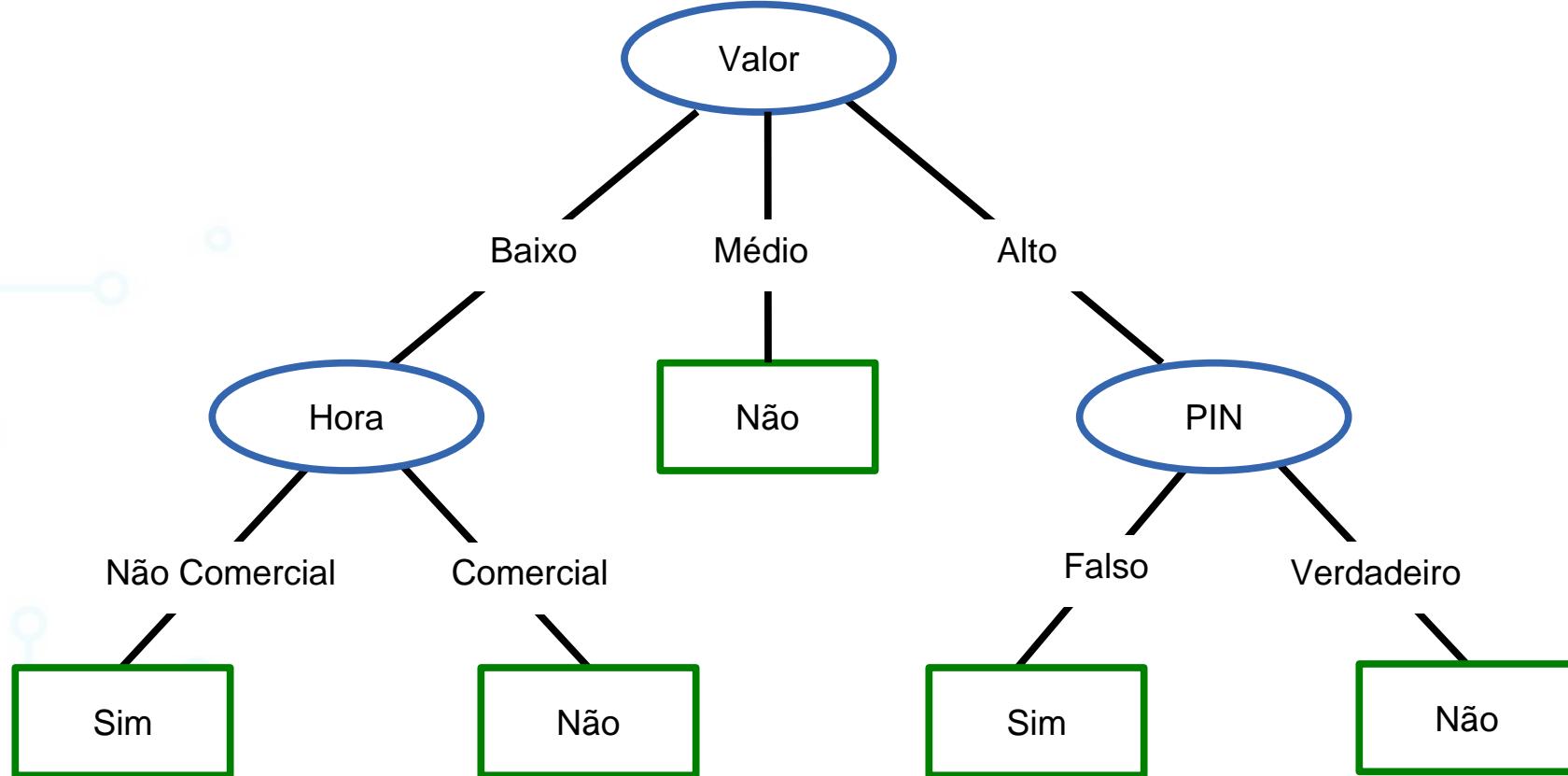
Árvore de Decisão



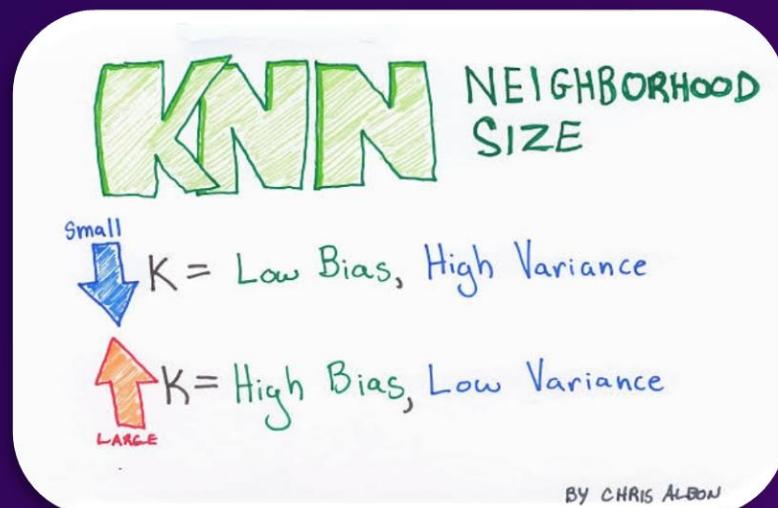
Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim

Compra	Valor	Local	Hora	PIN	Aprovada
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

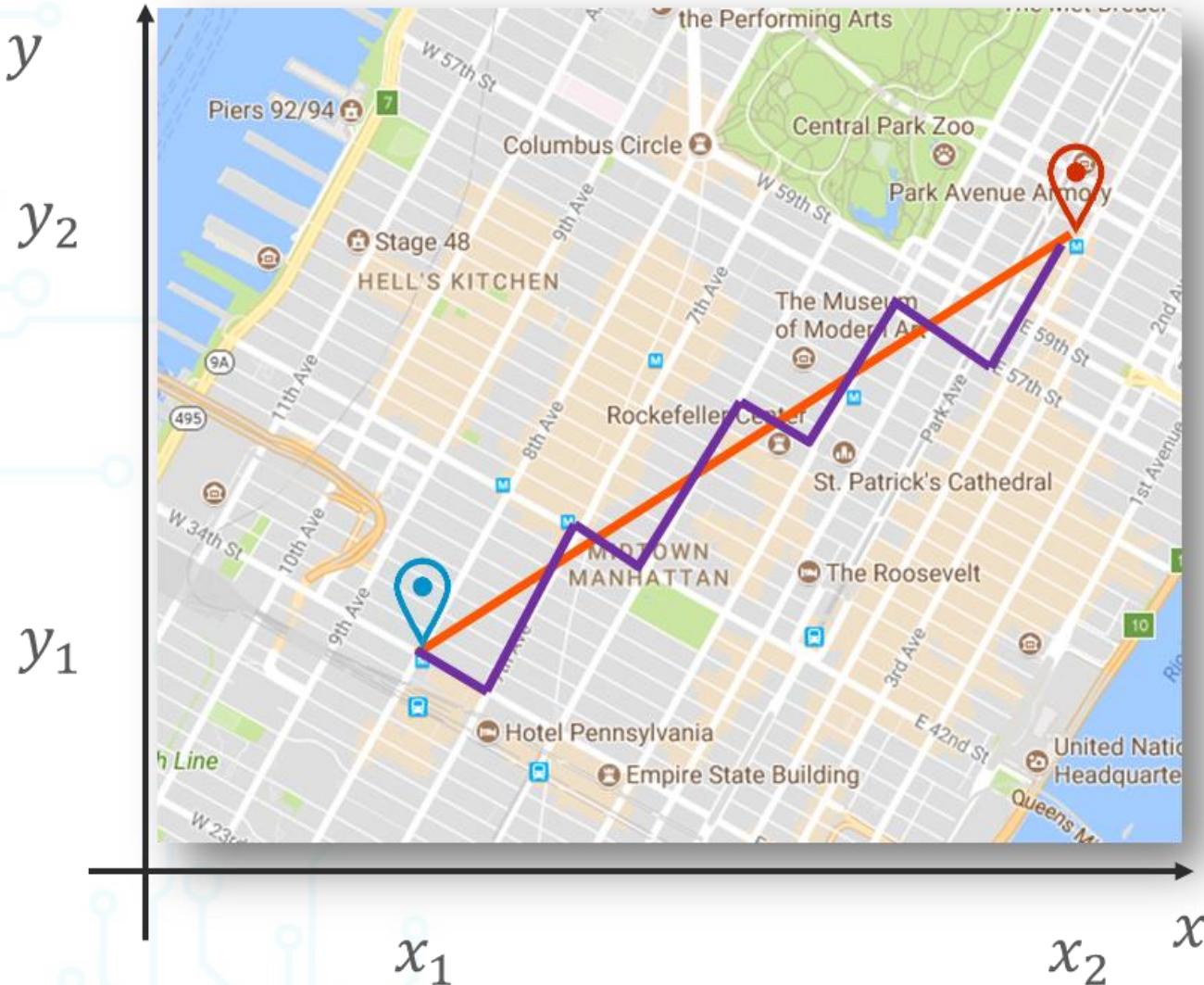
Árvore de Decisão



“ K-Nearest Neighbors (KNN)



Medidas de Distância



Manhattan

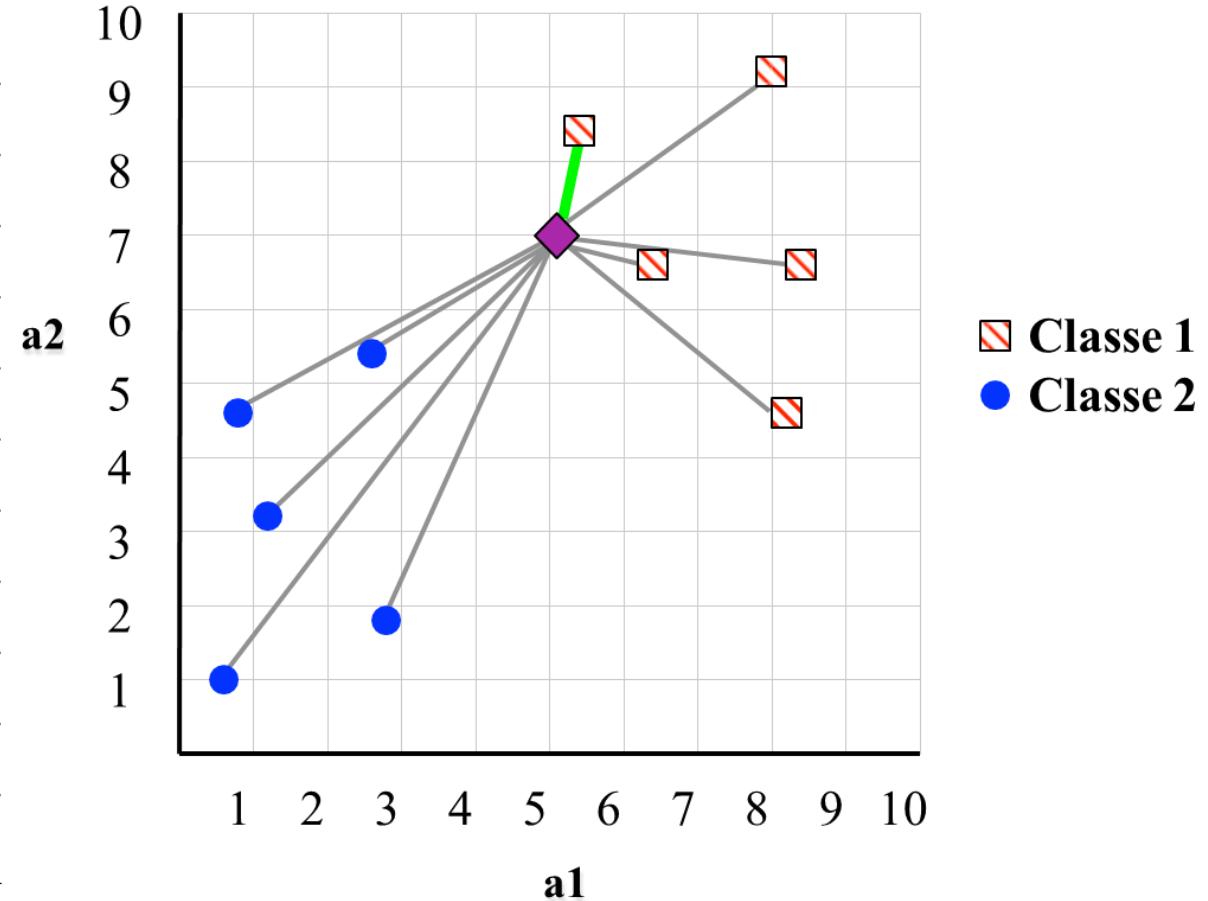
$$d_1 = |x_1 - x_2| + |y_1 - y_2|$$

Euclidiana

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

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	?



Fonte: Keogh, E. A Gentle 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 $\mathbf{x}=[x_1, x_2, \dots, x_n]$ e $\mathbf{y}=[y_1, y_2, \dots, y_n]$, 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

Compra	Valor	Local	Hora	PIN	Aprovada
1	Baixo	Internacional	Não Comercial	Verdadeiro	Não
2	Baixo	Internacional	Não Comercial	Falso	Não
3	Médio	Internacional	Não Comercial	Verdadeiro	Sim
4	Alto	Mesmo	Não Comercial	Verdadeiro	Sim
5	Alto	BR	Comercial	Verdadeiro	Sim
6	Alto	BR	Comercial	Falso	Não
7	Médio	BR	Comercial	Falso	Sim
8	Baixo	Mesmo	Não Comercial	Verdadeiro	Não
9	Baixo	BR	Comercial	Verdadeiro	Sim
10	Alto	Mesmo	Comercial	Verdadeiro	Sim
11	Baixo	Mesmo	Comercial	Falso	Sim
12	Médio	Mesmo	Não Comercial	Falso	Sim
13	Médio	Internacional	Comercial	Verdadeiro	Sim
14	Alto	Mesmo	Não Comercial	Falso	Não

“ Naive Bayes

GAUSSIAN
NAIVE BAYES
CLASSIFIER

“Gaussian” because this is a normal distribution

This is our prior belief

$$P(\text{class} | \text{data}) = \frac{P(\text{data} | \text{class}) \times P(\text{class})}{P(\text{data})}$$

We don't calculate this in naive bayes classifiers

Chris Albon

Naive Bayes

- O teorema de Bayes permite o cálculo da posteriori de h
- MAP (Maximum a Posteriori) Hipótese Máxima a Posteriori
 - $P(h|D) = (P(D|h) P(h)) / P(D)$

Naive Bayes

- O NB, simplifica o cálculo de $P(a_1, a_2, \dots, a_n | v_j)$ assumindo que todos os atributos são condicionalmente independentes entre si dado o valor da classe.
- Independência dos atributos condicionada à classe:

$$P(a_1, a_2 \dots a_n | v_j) = \prod_i P(a_i | v_j)$$

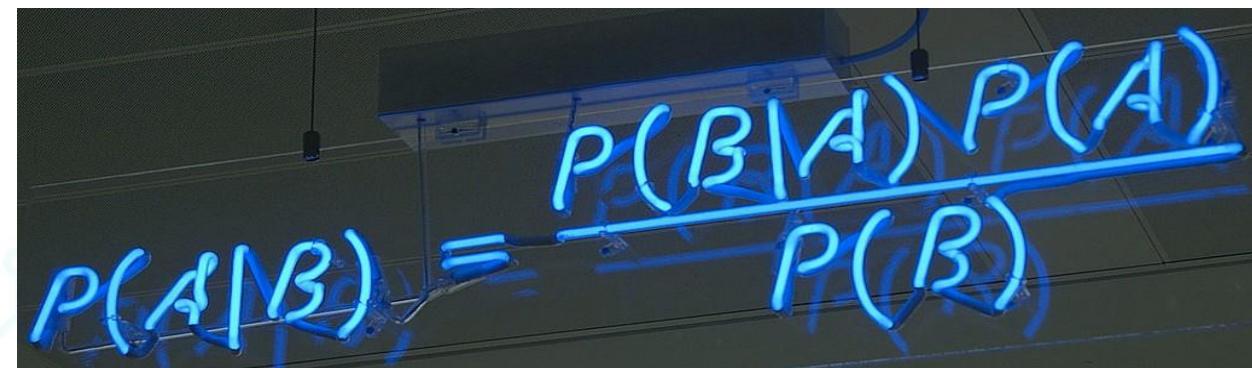
Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11)

$$P(\text{Classe}=0|A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$P(A_1=0, A_2=1, A_3=0, A_4=1|\text{Classe}=0) * P(\text{Classe}=0)$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1



Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11)

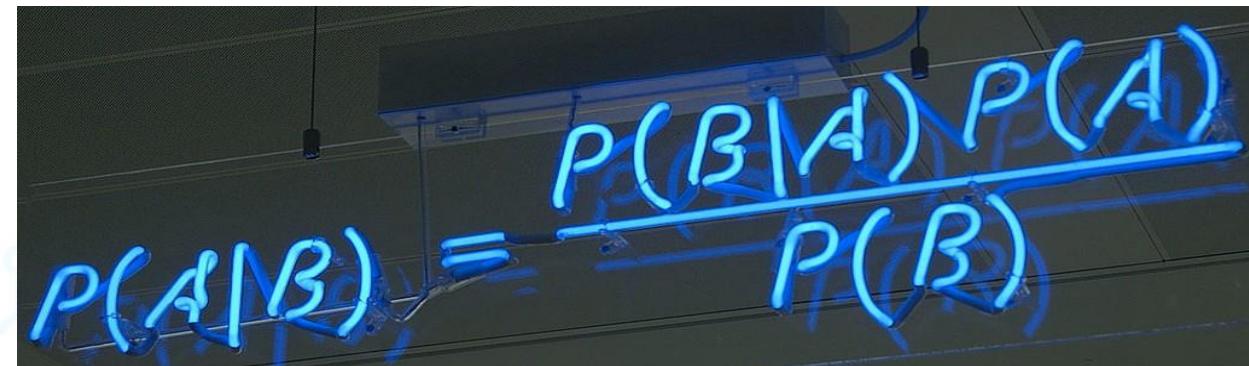
$$P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$P(A_1=0, A_2=1, A_3=0, A_4=1 | \text{Classe}=0) * P(\text{Classe}=0)$$

Como assumimos a independência condicional, temos que:

$$P(A_1=0, A_2=1, A_3=0, A_4=1 | \text{Classe}=0) = \prod P(A_i=x_i | \text{Classe}=0)$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1



Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11)

$$P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$P(A_1=0, A_2=1, A_3=0, A_4=1 | \text{Classe}=0) * P(\text{Classe}=0)$$

Como assumimos a independência condicional, temos que:

$$P(A_1=0, A_2=1, A_3=0, A_4=1 | \text{Classe}=0) = \prod P(A_i=x_i | \text{Classe}=0)$$

Assim, temos:

$$P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$(\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0)$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$(\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) =$$

$$(P(A_1=0 | C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe} = 0)$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$(\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) =$$

$$(P(A_1=0 | C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe} = 0)$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 & P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 & (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 & (P(A_1=0 | C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe} = 0) \\
 & (1/5) *
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) =$$

$$(\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) =$$

$$(P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0)$$

$$(1/5) *$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) &= \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) & \\
 (1/5) *
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 (1/5) * (2/5)
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0)= \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 (1/5) * (2/5)
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 & P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 & (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 & (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 & (1/5) * (2/5)
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * \textcolor{blue}{P(A_4=1|C=0)}) * P(\text{Classe}=0) \\
 (1/5) * (2/5) * (5/5) *
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 & P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 & (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 & (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * \textcolor{blue}{P(A_4=1|C=0)}) * P(\text{Classe}=0) \\
 & (1/5) * (2/5) * (5/5) *
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 (1/5) * (2/5) * (5/5) * (1/5)
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado , deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 & P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 & (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 & (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 & (1/5) * (2/5) * (5/5) * (1/5) * (5/10)
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) &= \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) & \\
 (1/5) * (2/5) * (5/5) * (1/5) * (5/10) &=
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) &= \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) & \\
 (1/5) * (2/5) * (5/5) * (1/5) * (5/10) &= \\
 0.2 * 0.4 * 1 * 0.2 * 0.5 & = 0.008
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A1=0, A2=1, A3=0, A4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 (P(A1=0|C=0) * P(A2=1|C=0) * P(A3=0|C=0) * P(A4=1|C=0)) * P(\text{Classe}=0) \\
 (1/5) * (2/5) * (5/5) * (1/5) * (5/10) = \\
 0.2 * 0.4 * 1 * 0.2 * 0.5 = 0.008
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Agora fazemos o mesmo processo para a classe=1:

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) &= \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) & \\
 (1/5) * (2/5) * (5/5) * (1/5) * (5/10) &= \\
 0.2 * 0.4 * 1 * 0.2 * 0.5 & = 0.008
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Agora fazemos o mesmo processo para a classe=1:

$$\begin{aligned}
 P(\text{Classe}=1 | A_1=0, A_2=1, A_3=0, A_4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=1)) * P(\text{Classe}=1) &=
 \end{aligned}$$

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 & P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 & (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 & (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 & (1/5) * (2/5) * (5/5) * (1/5) * (5/10) = \\
 & 0.2 * 0.4 * 1 * 0.2 * 0.5 = 0.008
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Agora fazemos o mesmo processo para a classe=1:

$$\begin{aligned}
 & P(\text{Classe}=1 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 & (\prod P(A_i=x_i | \text{Classe}=1)) * P(\text{Classe}=1) = \\
 & (P(A_1=0|C=1) * P(A_2=1|C=1) * P(A_3=0|C=1) * P(A_4=1|C=1)) * P(\text{Classe}=1)
 \end{aligned}$$

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A1=0, A2=1, A3=0, A4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) &= \\
 (P(A1=0|C=0) * P(A2=1|C=0) * P(A3=0|C=0) * P(A4=1|C=0)) * P(\text{Classe}=0) & \\
 (1/5) * (2/5) * (5/5) * (1/5) * (5/10) &= \\
 0.2 * 0.4 * 1 * 0.2 * 0.5 &= 0.008
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Agora fazemos o mesmo processo para a classe=1:

$$\begin{aligned}
 P(\text{Classe}=1 | A1=0, A2=1, A3=0, A4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=1)) * P(\text{Classe}=1) &= \\
 (P(A1=0|C=1) * P(A2=1|C=1) * P(A3=0|C=1) * P(A4=1|C=1)) * P(\text{Classe}=1) & \\
 (5/5) * (5/5) * (0/5) * (5/5) * (5/10) &=
 \end{aligned}$$

Naive Bayes

Considere por exemplo, que a partir do conjunto de instâncias de treinamento dado, deseja-se classificar uma nova instância (11):

$$\begin{aligned}
 P(\text{Classe}=0 | A1=0, A2=1, A3=0, A4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) &= \\
 (P(A1=0|C=0) * P(A2=1|C=0) * P(A3=0|C=0) * P(A4=1|C=0)) * P(\text{Classe}=0) & \\
 (1/5) * (2/5) * (5/5) * (1/5) * (5/10) &= \\
 0.2 * 0.4 * 1 * 0.2 * 0.5 & = 0.008
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Agora fazemos o mesmo processo para a classe=1:

$$\begin{aligned}
 P(\text{Classe}=1 | A1=0, A2=1, A3=0, A4=1) &= \\
 (\prod P(A_i=x_i | \text{Classe}=1)) * P(\text{Classe}=1) &= \\
 (P(A1=0|C=1) * P(A2=1|C=1) * P(A3=0|C=1) * P(A4=1|C=1)) * P(\text{Classe}=1) & \\
 (5/5) * (5/5) * (0/5) * (5/5) * (5/10) &= \\
 1 * 1 * 0 * 1 * 0.5 & = 0
 \end{aligned}$$

Naive Bayes

Solução

$$\begin{aligned}
 P(\text{Classe}=0 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=0)) * P(\text{Classe}=0) = \\
 (P(A_1=0|C=0) * P(A_2=1|C=0) * P(A_3=0|C=0) * P(A_4=1|C=0)) * P(\text{Classe}=0) \\
 (1+0.5)/(5+1) * (2+0.5)/(5+1) * (5+0.5)/(5+1) * (1+0.5)/(5+1) * \\
 (5/10) = 0.25 * 0.41 * 0.91 * 0.25 * 0.5 = 0.011
 \end{aligned}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
		1	0	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Agora fazemos o mesmo processo para a classe=1:

$$\begin{aligned}
 P(\text{Classe}=1 | A_1=0, A_2=1, A_3=0, A_4=1) = \\
 (\prod P(A_i=x_i | \text{Classe}=1)) * P(\text{Classe}=1) = \\
 (P(A_1=0|C=1) * P(A_2=1|C=1) * P(A_3=0|C=1) * P(A_4=1|C=1)) * P(\text{Classe}=1) \\
 (5+0.5)/(5+1) * (5+0.5)/(5+1) * (0+0.5)/(5+1) * (5+0.5)/(5+1) * \\
 (5/10) = 0.91 * 0.91 * 0.09 * 0.91 * 0.5 = \mathbf{0.033}
 \end{aligned}$$

Naive Bayes

Solução

$$P(\text{Classe}=0|A_1=0, A_2=1, A_3=0, A_4=1) = 0.011$$

$$P(\text{Classe}=1|A_1=0, A_2=1, A_3=0, A_4=1) = \mathbf{0.033}$$

Instâncias	Classe	A1	A2	A3	A4
1	1	0	1	1	1
2	1	0	1	1	1
3	0	1	0	0	0
4	1	0	1	1	1
5	1	0	1	1	1
6	0	0	0	0	1
7	0	1	1	0	0
8	1	0	1	1	1
9	0	1	0	0	0
10	0	1	1	0	0
11	?	0	1	0	1

Normalizando, temos:

$$P(\text{Classe}=0|A_1=0, A_2=1, A_3=0, A_4=1) = 0.011/(0.011+\mathbf{0.033}) = 0.25$$

$$P(\text{Classe}=1|A_1=0, A_2=1, A_3=0, A_4=1) = \mathbf{0.033}/(0.011+\mathbf{0.033}) = 0.75$$

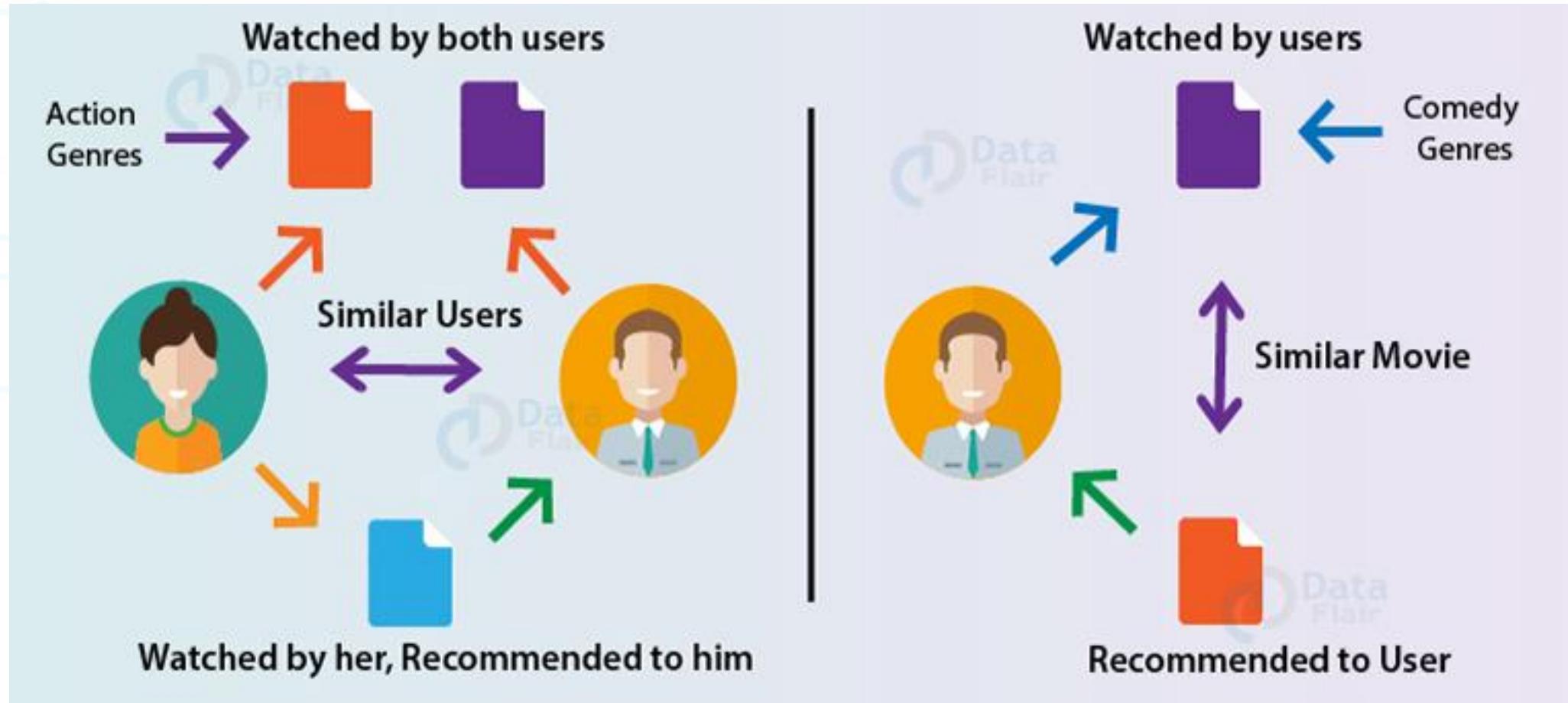
A classe mais provável é a classe 1.

Implementação de Exemplo

“

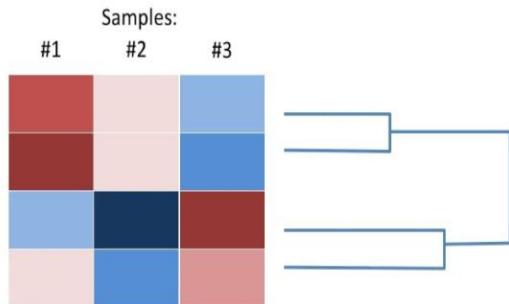
Agrupamento

Agrupamento



Agrupamento

Hierarchical Clustering, Clearly Explained!!!!



ACCOLERATIVE CLUSTERING

All observations start as their own cluster. Clusters meeting some criteria are merged. This process is repeated, growing clusters until some end point is reached.

ChrisAlbon

K-MEANS CLUSTERING

1. k centerpoints are randomly initialized.
2. Observations are assigned to the closest centerpoint.
3. Centerpoints are moved to the center of their members.
4. Repeat steps 2 and 3 until no observation changes membership in step 2.

ChrisAlbon

DBSCAN

DBSCAN looks for densely packed observations and makes no assumptions about the number or shape of clusters.

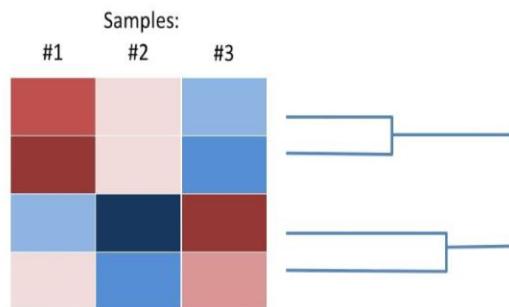
1. A random observation, x_i , is selected
2. If x_i has a minimum of close neighbors, we consider it part of a cluster.
3. Step 2 is repeated recursively for all of x_i 's neighbors, then neighbors' neighbors etc... These are the cluster's core members.
4. Once Step 3 runs out of observations, a new random point is chosen.

Afterwards, observations not part of a core are assigned to a nearby cluster or marked as outliers.

ChrisAlbon

Hierárquico

Hierarchical Clustering,
Clearly Explained!!!!



AGGLOMERATIVE CLUSTERING

All observations start as their own cluster. Clusters meeting some criteria are merged. This process is repeated, growing clusters until some end point is reached.

Chris Albon

Hierarquia: Conceitos Básicos

Hierarquias são comumente usadas para organizar informação

[Web Site Directory](#) - Sites organized by subject

[Suggest your site](#)

[Business & Economy](#)

[B2B](#), [Finance](#), [Shopping](#), [Jobs](#)...

[Regional](#)

[Countries](#), [Regions](#), [US States](#)...

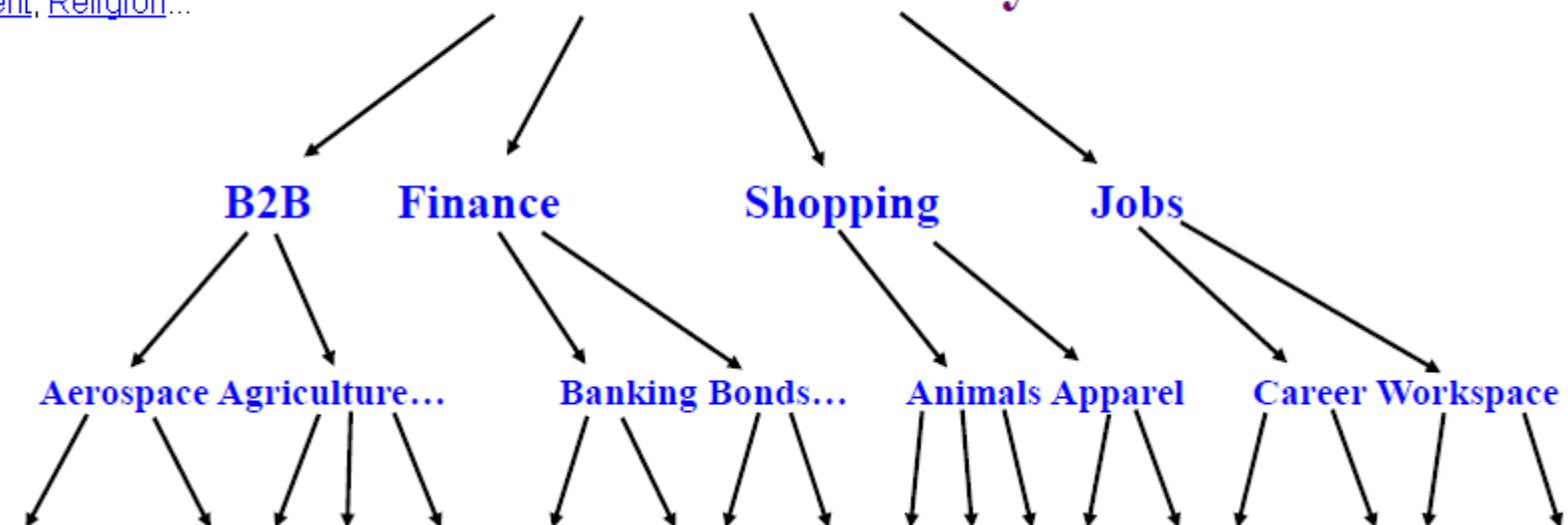
[Computers & Internet](#)

[Internet](#), [WWW](#), [Software](#), [Games](#)...

[Society & Culture](#)

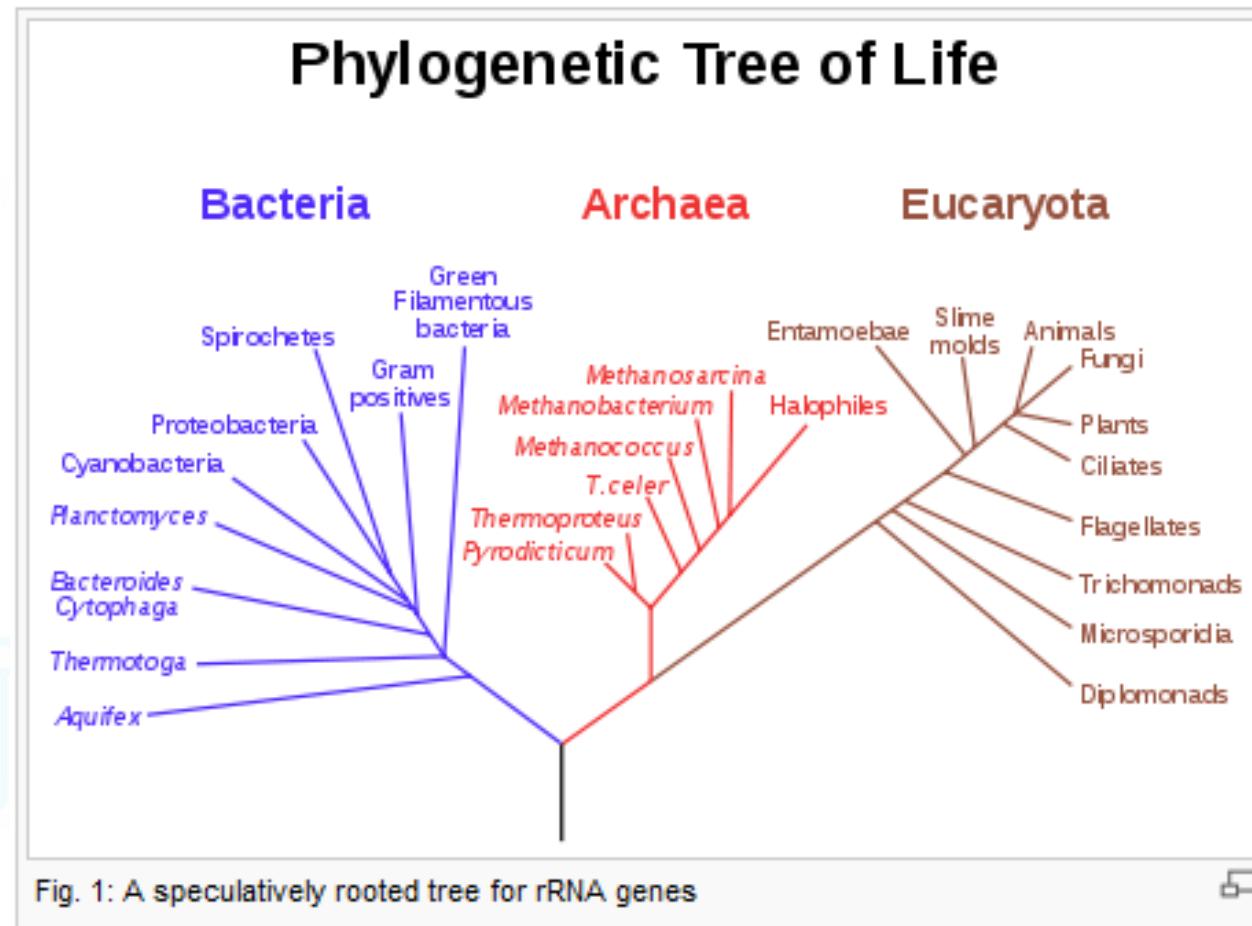
[People](#), [Environment](#), [Religion](#)...

Business & Economy

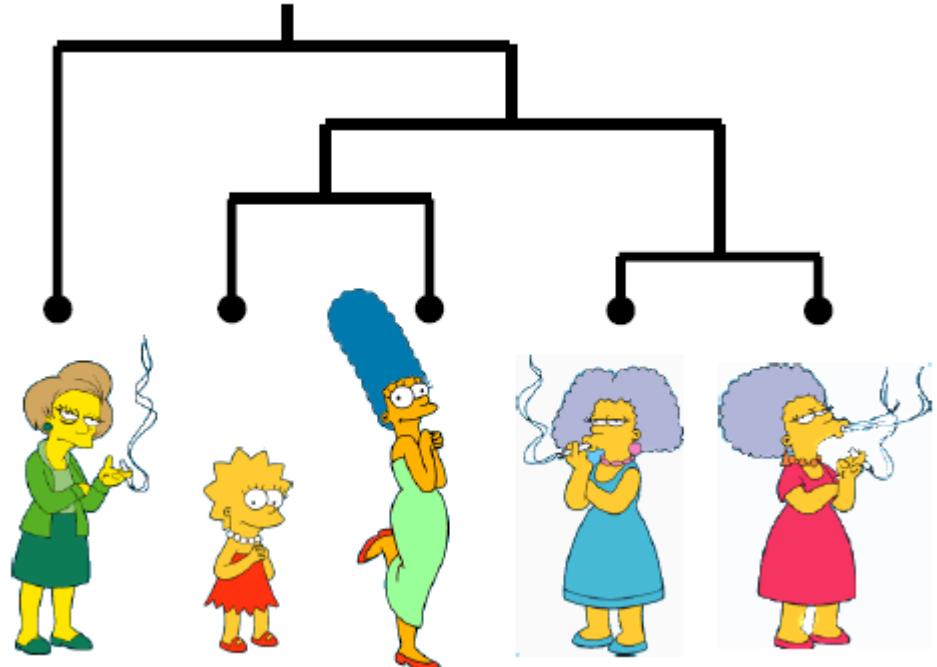


Hierarquia: Conceitos Básicos

Exemplo: árvores filogenéticas em biologia



Métodos Clássicos para Agrupamento Hierárquico



Bottom-Up (aglomerativos):

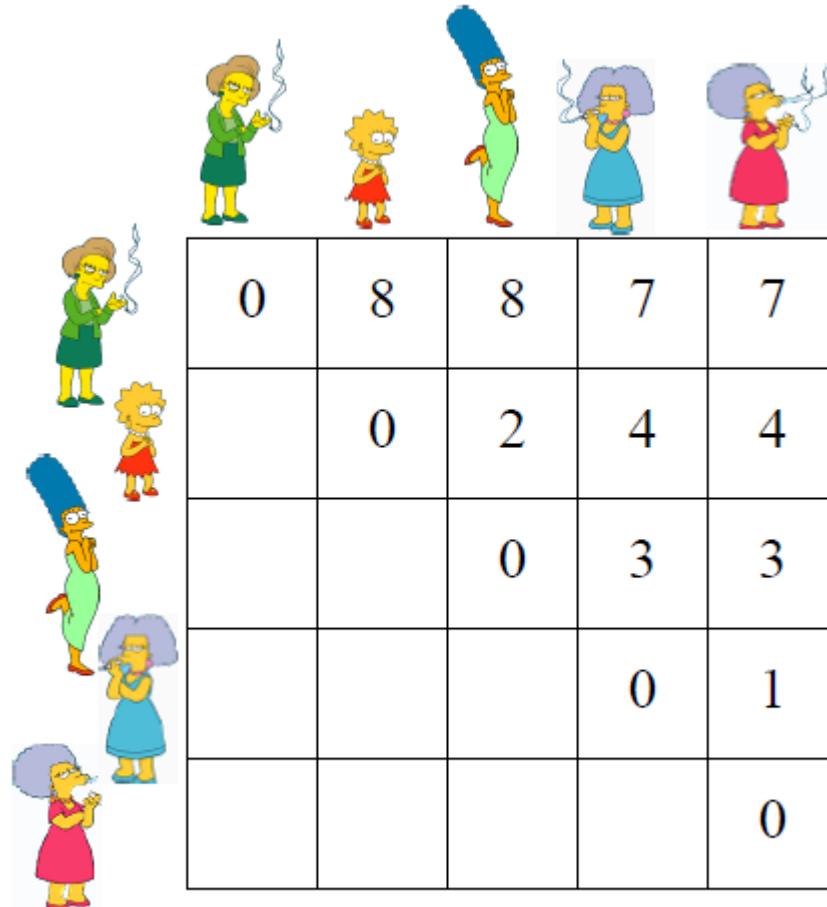
- Iniciar colocando cada objeto em um *cluster*
- Encontrar o melhor par de *clusters* para unir
- Unir o par de *clusters* escolhido
- Repetir até que todos os objetos estejam reunidos em um só *cluster*

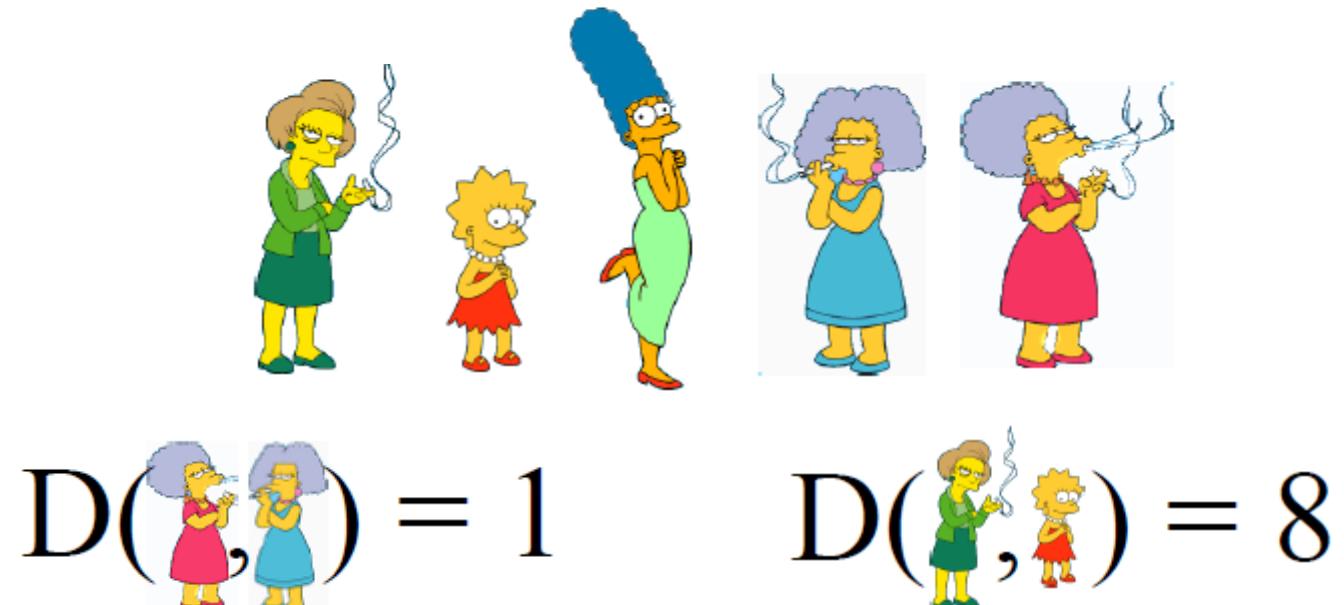
Top-Down (divisivos):

- Iniciar com todos objetos em um único *cluster*
- Sub-dividir o *cluster* em dois novos *clusters*
- Aplicar o algoritmo recursivamente em ambos, até que cada objeto forme um *cluster* por si só

Métodos Clássicos para Agrupamento Hierárquico

Algoritmos hierárquicos podem operar somente sobre uma matriz de distâncias.

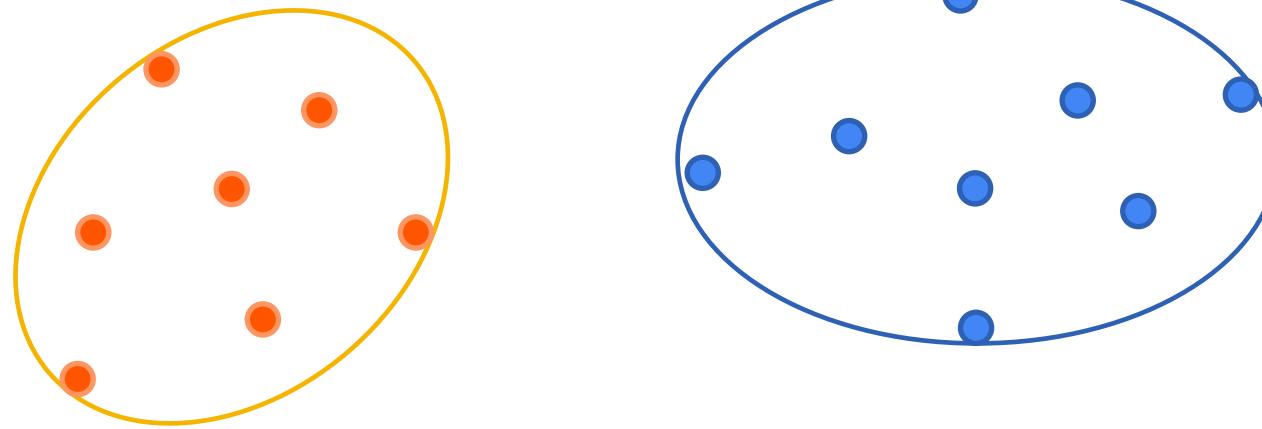



$$D(\text{, } \text{ }) = 1$$
$$D(\text{, } \text{ }) = 8$$

Como definir Inter-Cluster (Dis)similaridade

Matriz de Distância

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

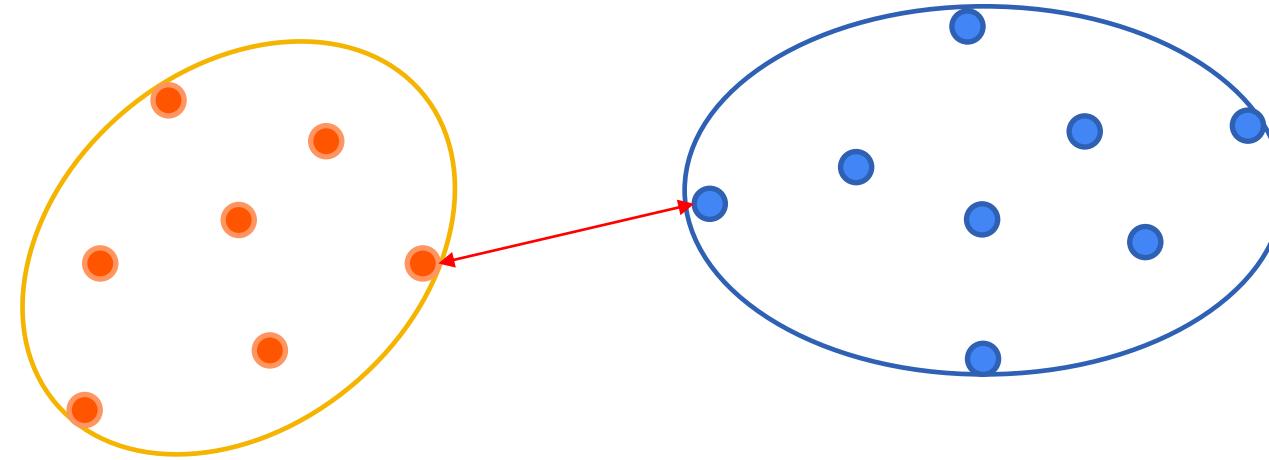


- MIN
- MAX
- Group Average
- ...

Como definir Inter-Cluster (Dis)similaridade

Matriz de Distância

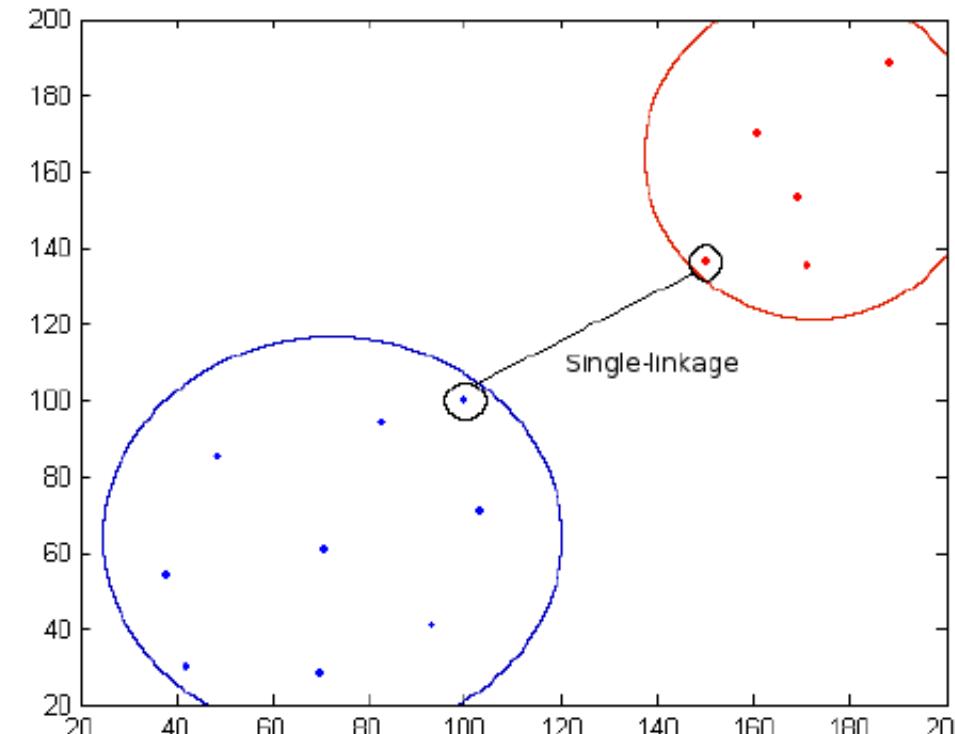
	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						



- MIN
- MAX
- Group Average
- ...

Single Linkage (Florek, 1951)

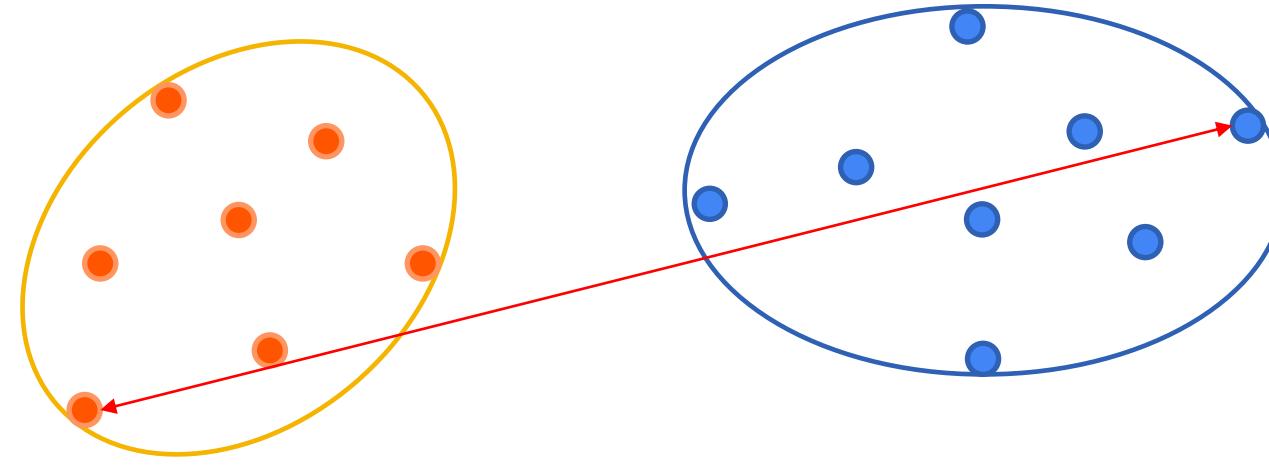
- Dissimilaridade entre clusters é dada pela menor dissimilaridade entre 2 objetos (um de cada cluster)
 - Originalmente baseado em Grafos: menor aresta entre dois vértices de subconjuntos distintos



Como definir Inter-Cluster (Dis)similaridade

Matriz de Distância

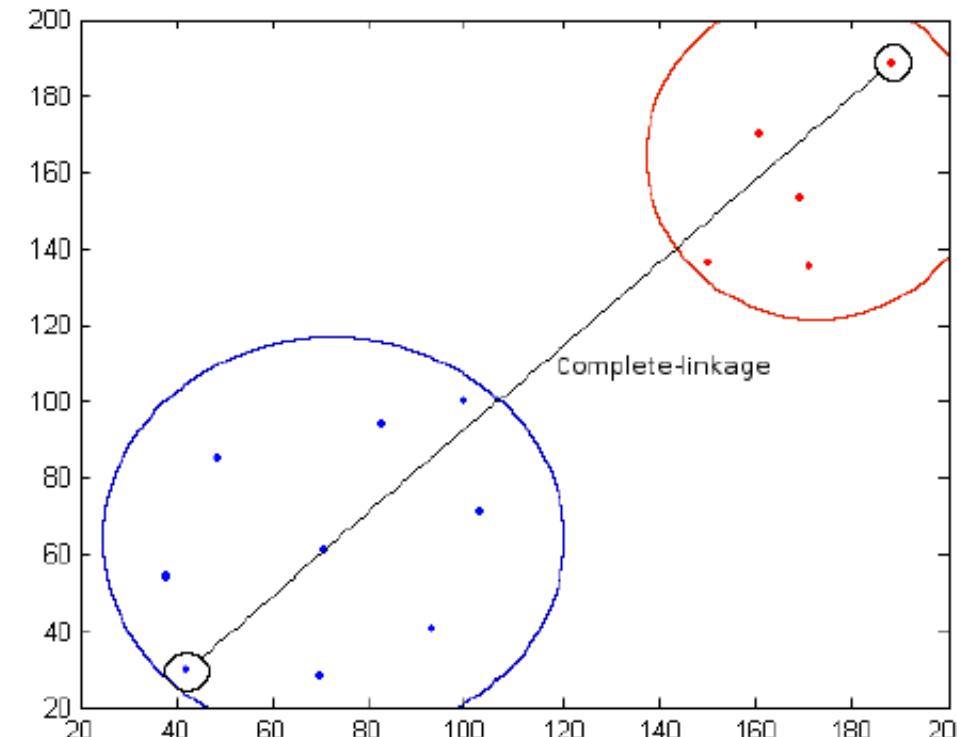
	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						



- MIN
- MAX
- Group Average
- ...

Complete Linkage (Sorenson, 1948)

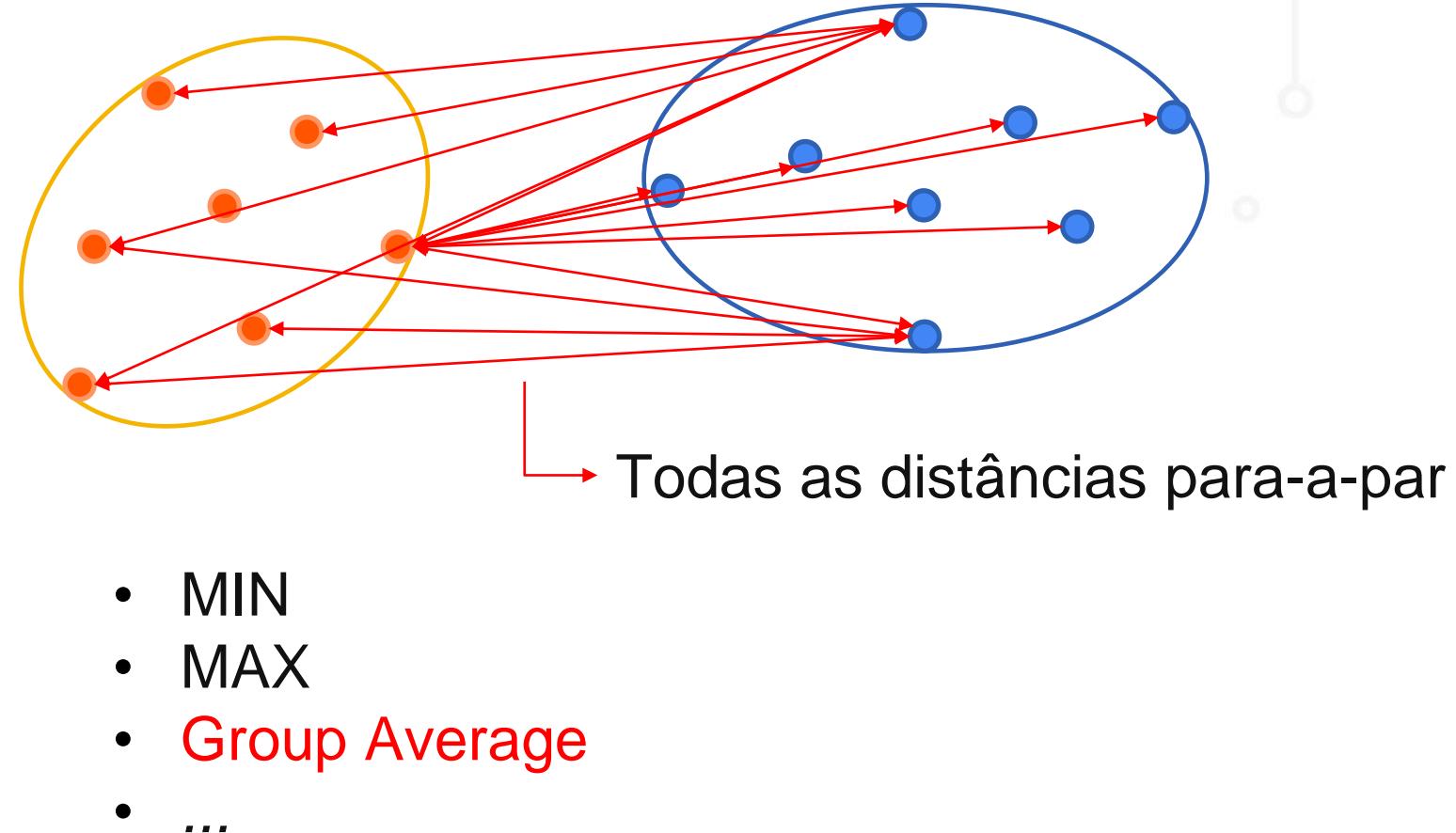
- Dissimilaridade entre clusters é dada pela maior dissimilaridade entre 2 objetos (um de cada cluster)
 - Originalmente baseado em Grafos: maior aresta entre dois vértices de subconjuntos distintos



Como definir Inter-Cluster (Dis)similaridade

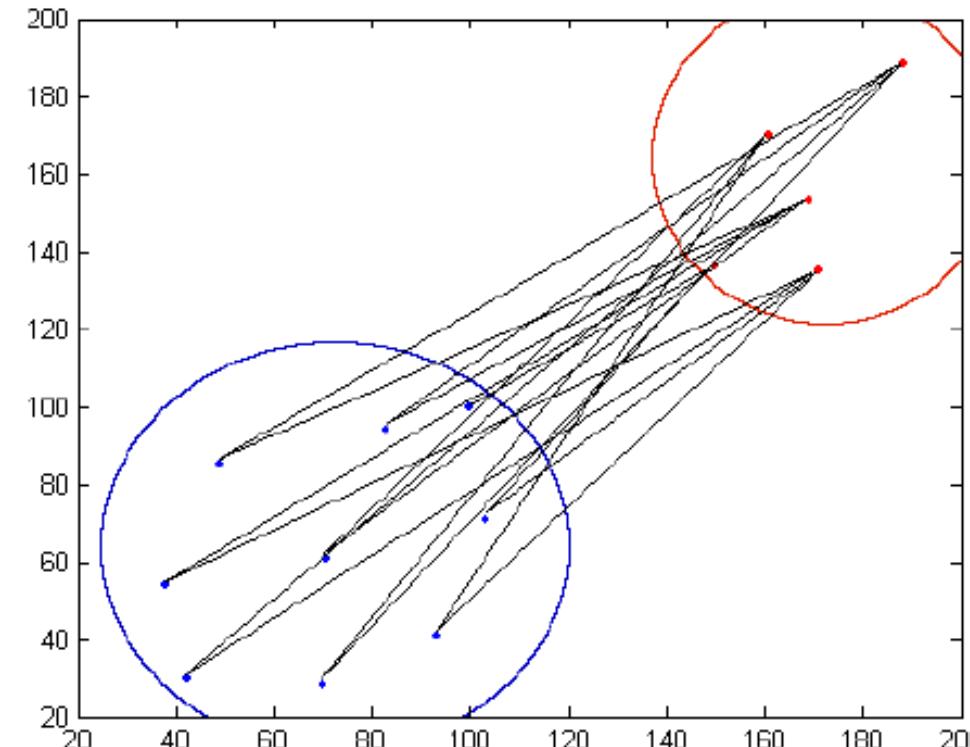
Matriz de Distância

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						



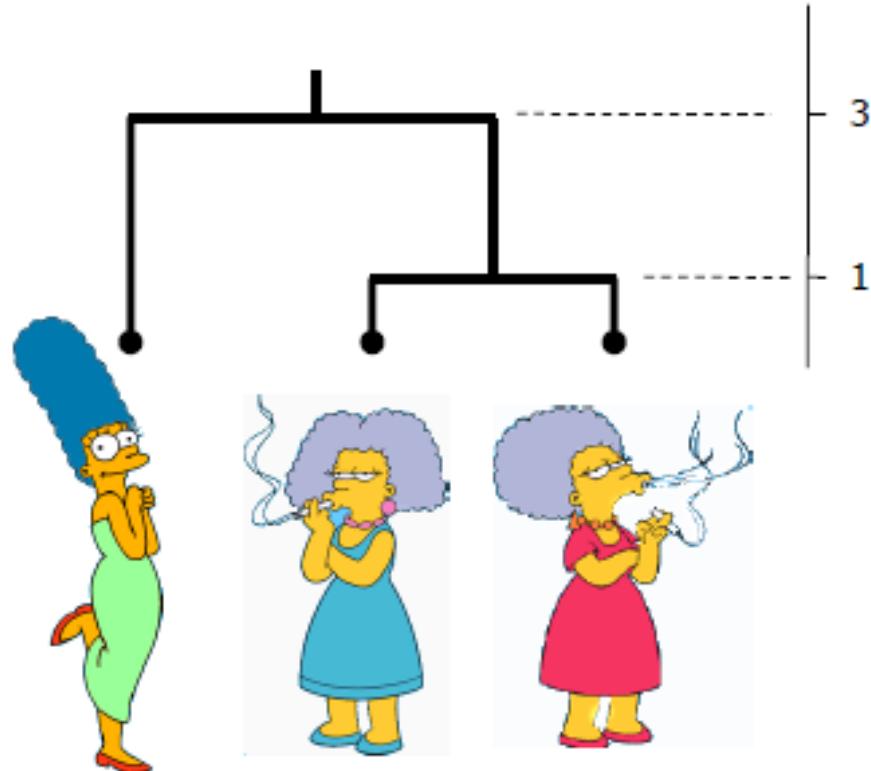
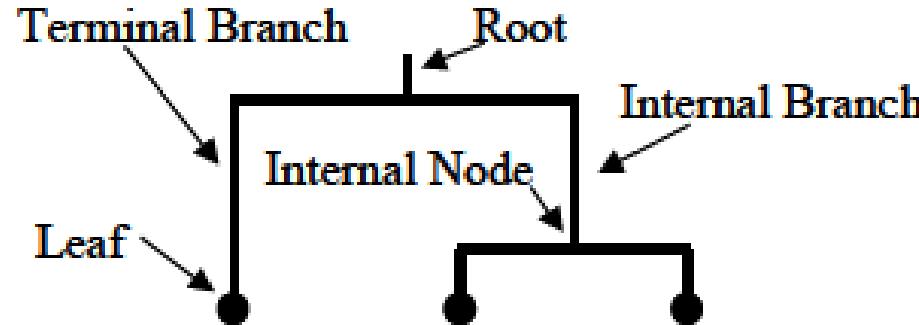
Complete Linkage (Sokal R and Michener C, 1958)

- Dissimilaridade entre clusters é dada pela **distância média** entre cada par de objetos (um de cada cluster)
- Também conhecido como UPGMA – Unweighted Pair Group Method using Arithmetic averages

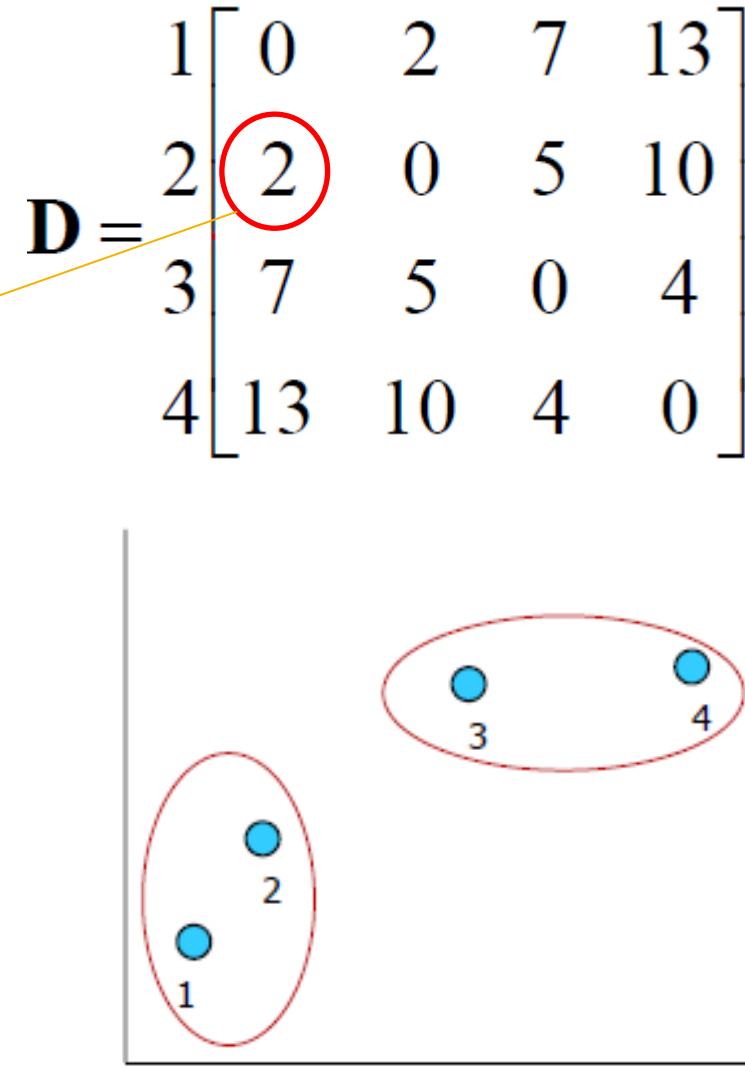
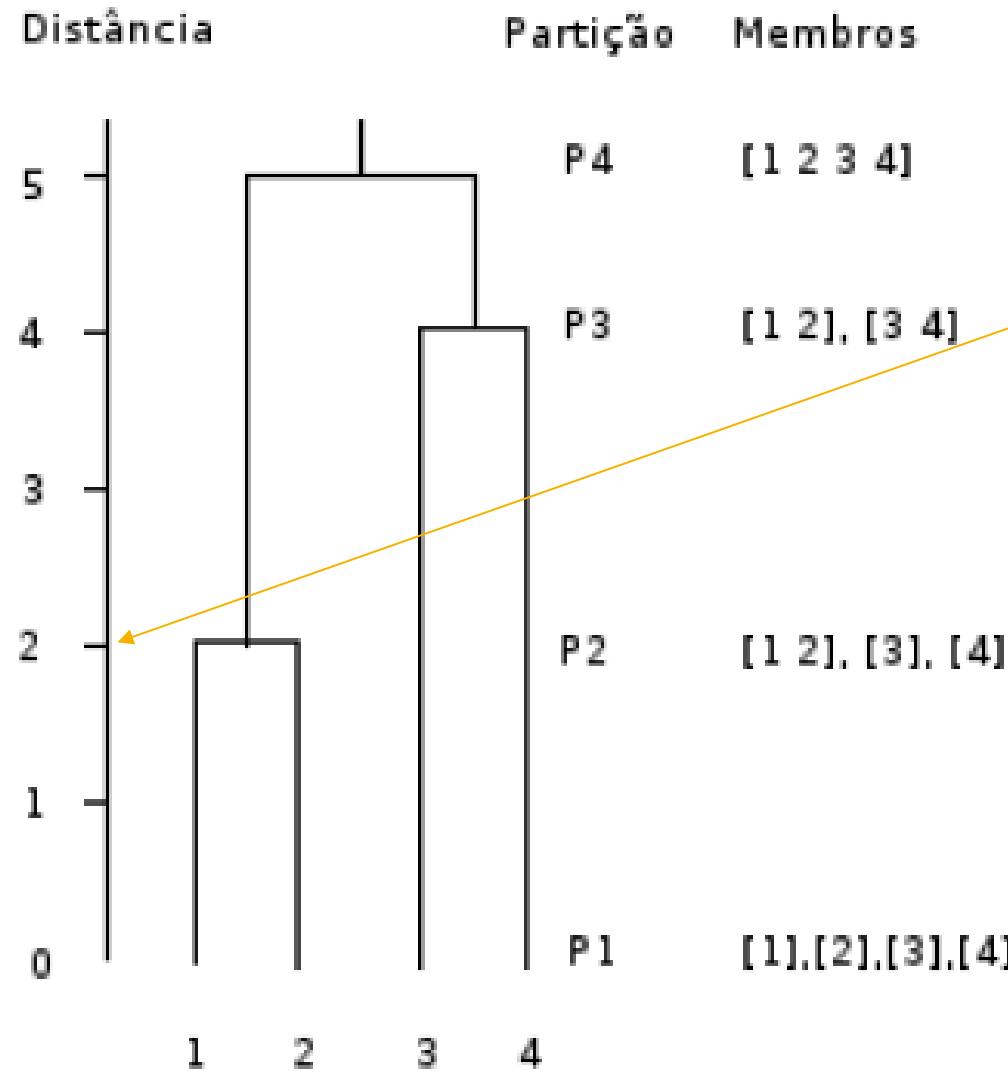


Dendrograma = Hierarquia + Dissimilaridade entre Clusters

A dissimilaridade entre dois clusters (possivelmente **singletons**) é representada como a altura do nó interno mais baixo compartilhado



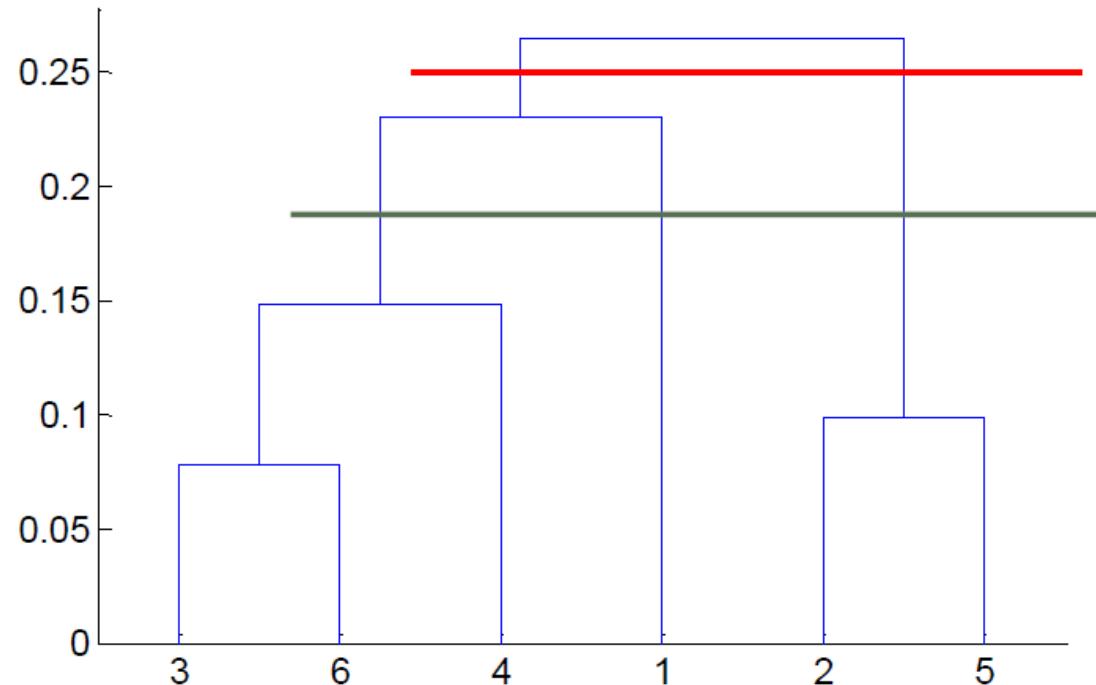
Dendrogramma



Dendrograma -> Grupos

Partições são obtidas via **cortes** no dendrograma

- cortes horizontais
- no. de grupos da partição = no. de interseções

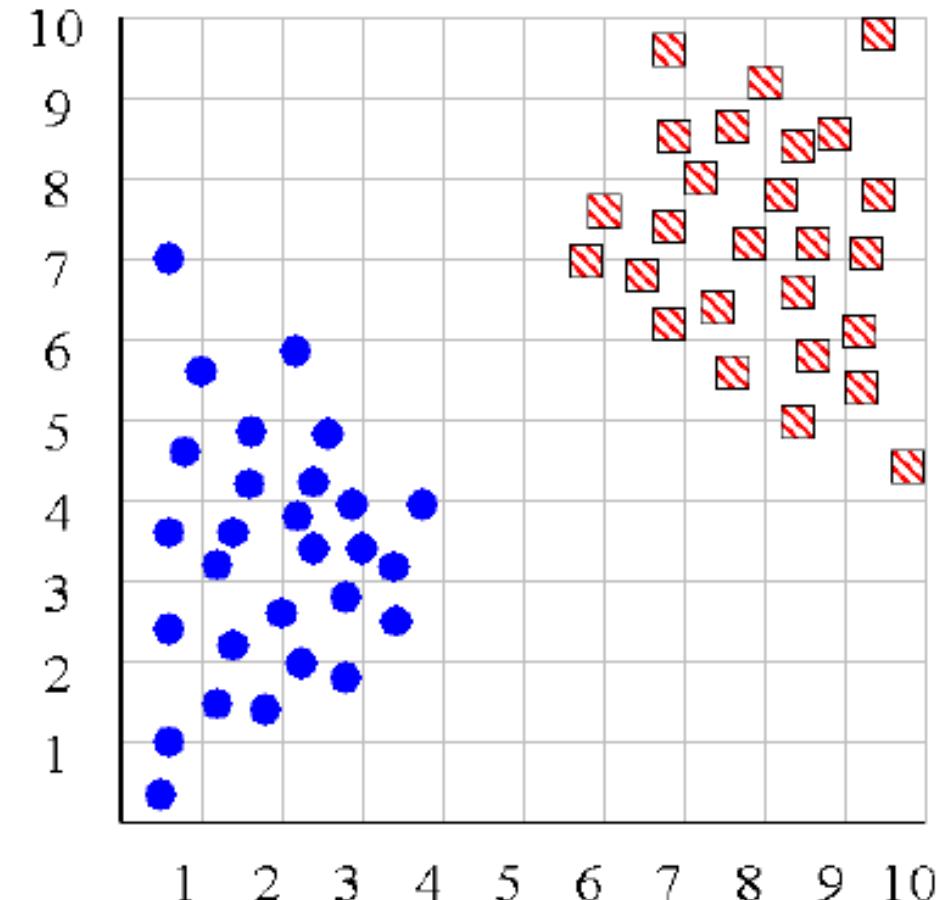
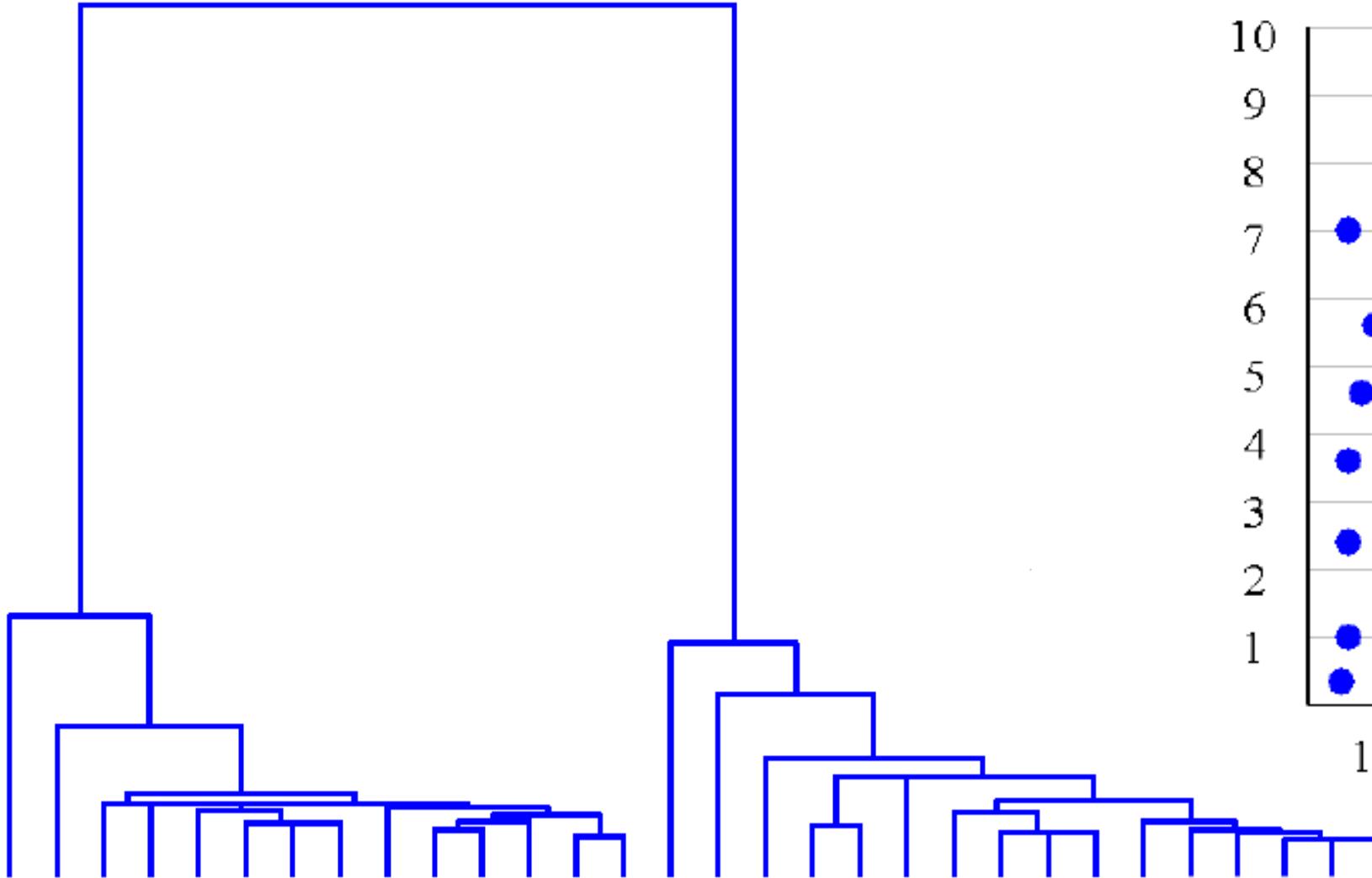


$$G_1 = \{ (x1, x3, x4, x6), (x2, x5) \}$$

$$G_2 = \{ (x1), (x3, x4, x6), (x2, x5) \}$$

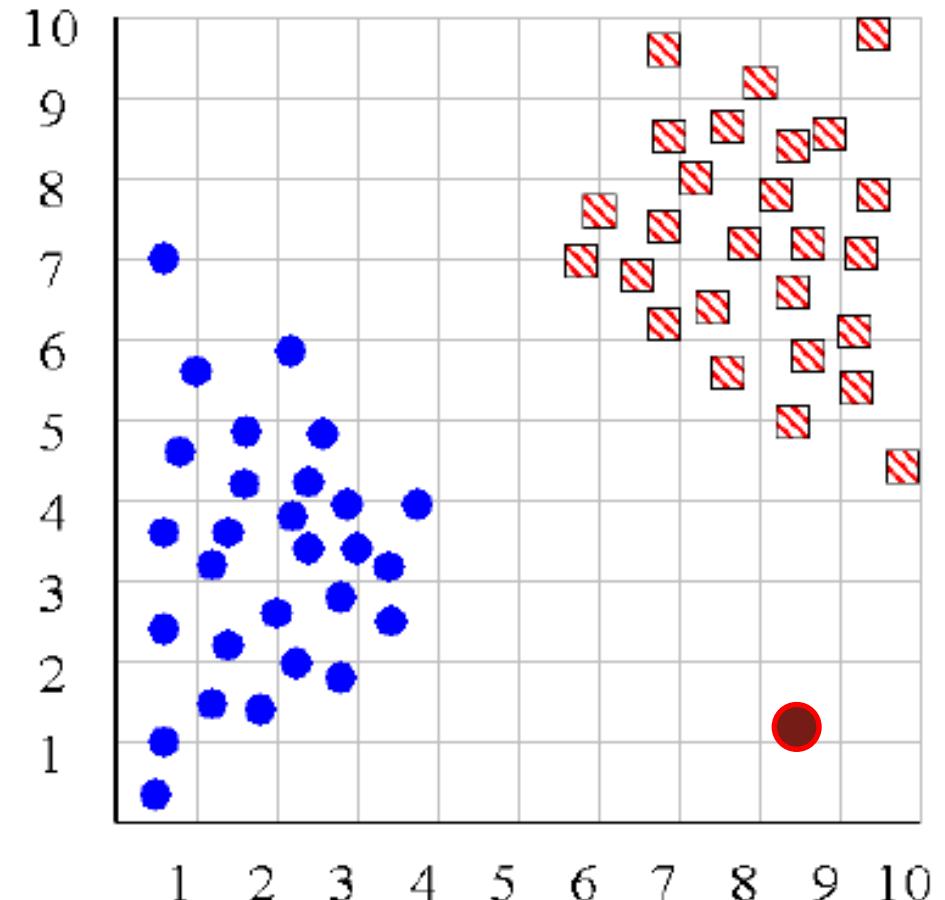
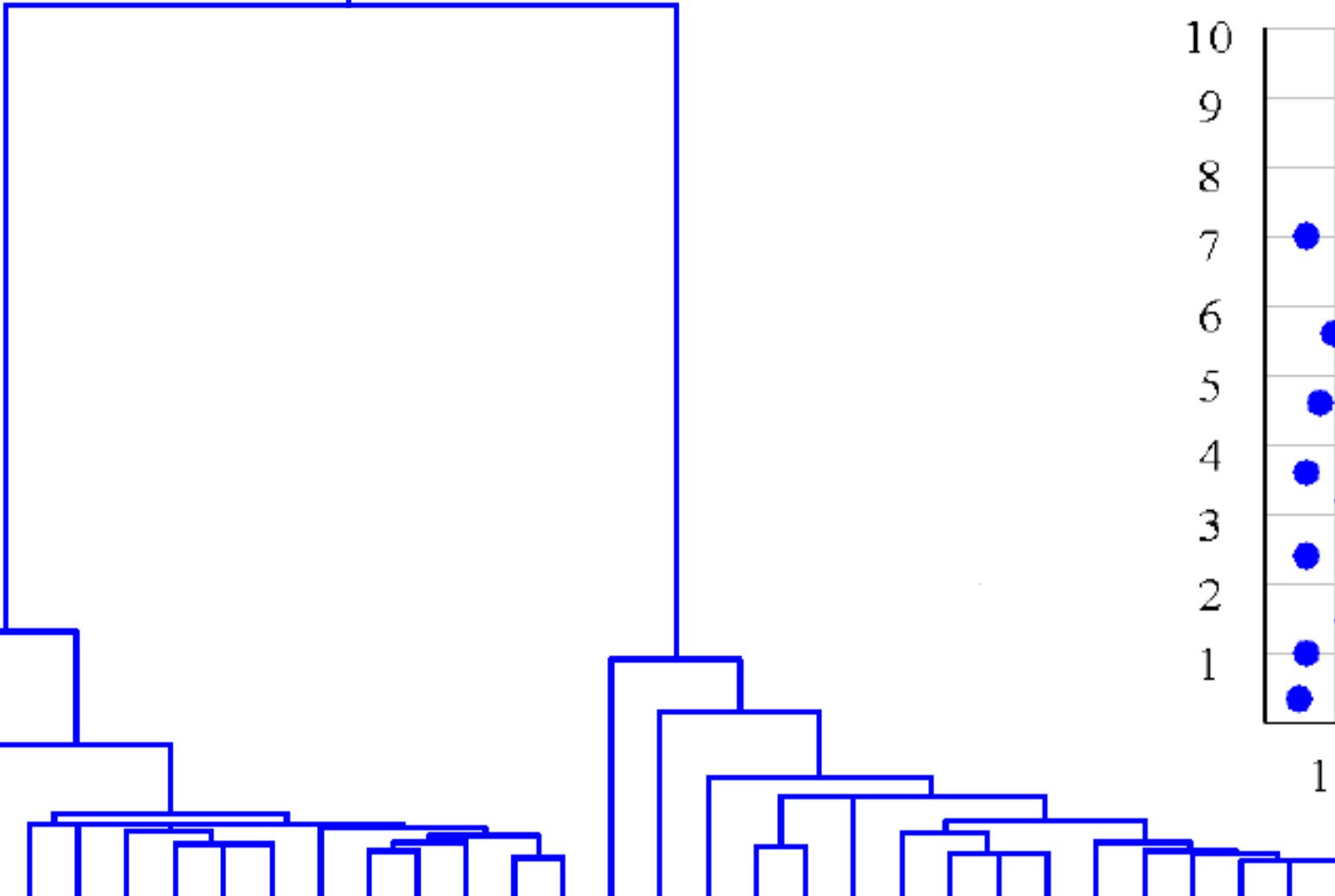
Dendrograma -> Grupos

Pode-se examinar o dendrograma para tentar estimar o número mais natural de clusters.



Dendrograma -> Outlier

Pode-se examinar o dendrograma para tentar detectar a presença de outliers.



K-means

K-MEANS CLUSTERING

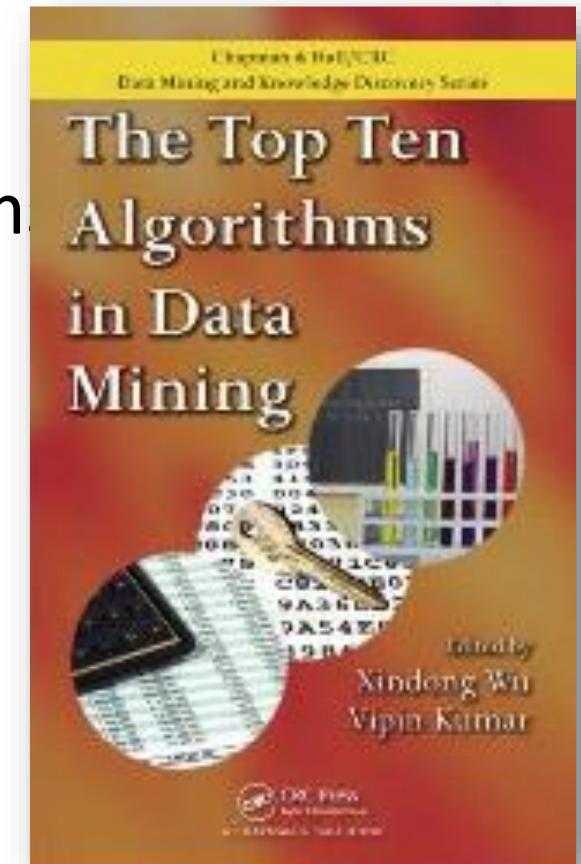
1. k centerpoints are randomly initialized.
2. Observations are assigned to the closest centerpoint.
3. Centerpoints are moved to the center of their members.
4. Repeat steps 2 and 3 until no observation changes membership in step 2.

Chris Albon

K-Means

Aqui veremos 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
- 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

Referência Mais Aceita como Original:

J. B. MacQueen, Some methods of classification and analysis of multivariate observations, In Proceedings 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, California, USA, 1967, 281–297

Porém...

“K-means has a rich and diverse history as it was independently discovered in different scientific fields by Steinhaus (1956), Lloyd (proposed in 1957, published in 1982), Ball & Hall (1965) and MacQueen (1967)” [Jain, Data Clustering: 50 Years Beyond K-Means, Patt. Rec. Lett., 2010]

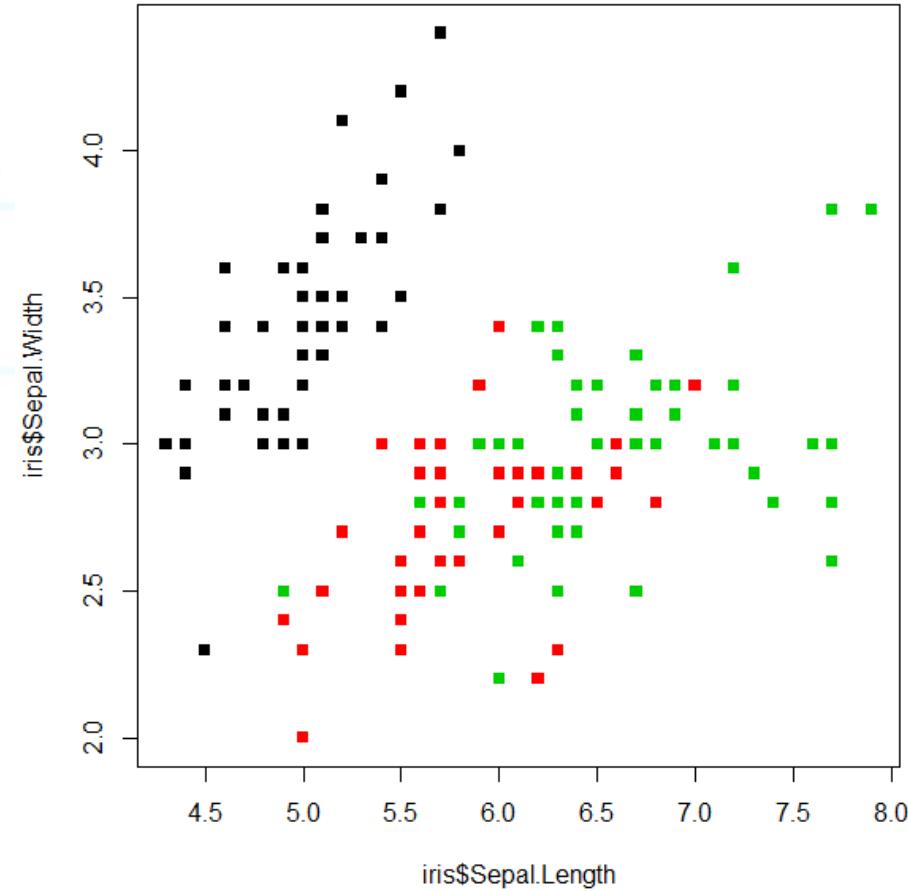
... e tem sido assunto por mais de meio século !

Douglas Steinley, K-Means Clustering: A Half-Century Synthesis, British Journal of Mathematical and Statistical Psychology, Vol. 59, 2006

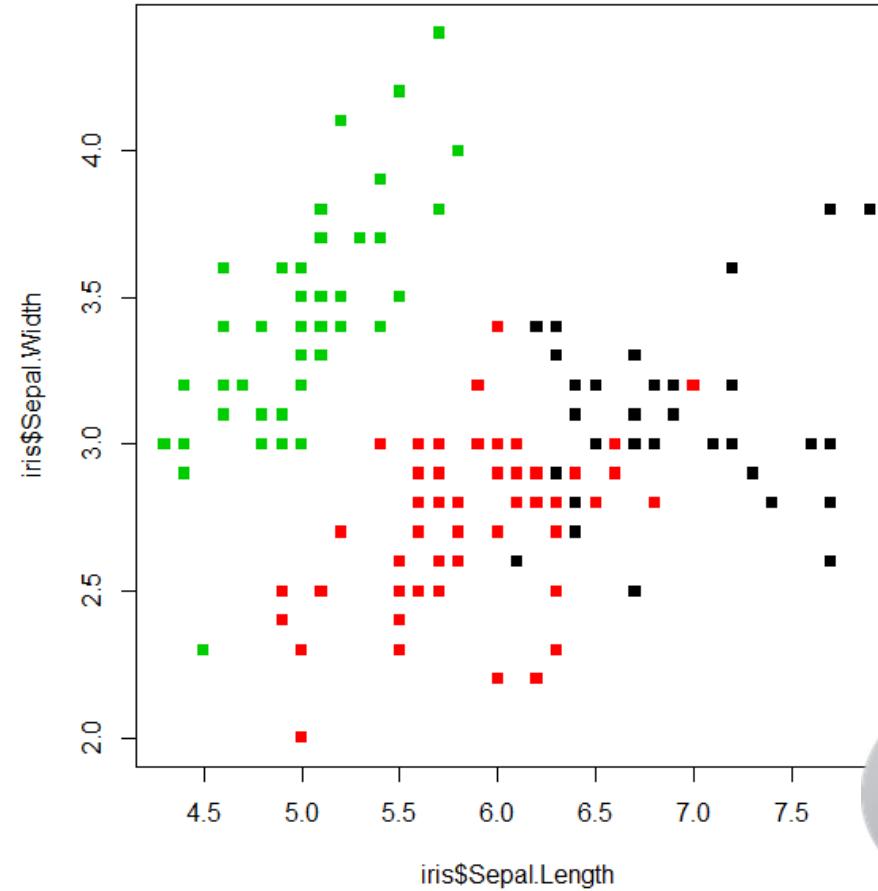
K-Means

```
1 data(iris) #Carrega os dados  
2 groups = kmeans(iris[1:4], center=3, iter.max=10)
```

Original



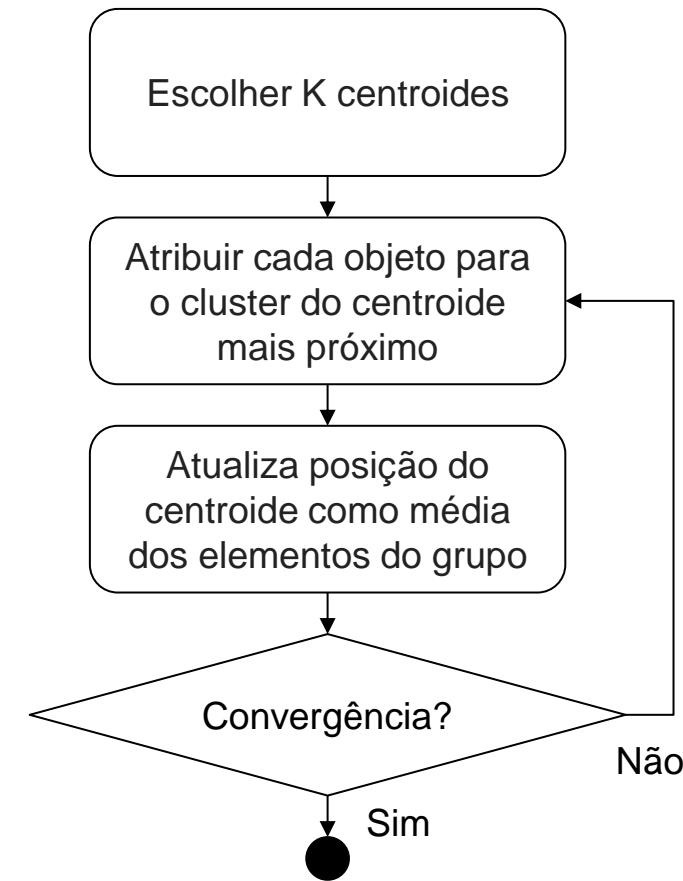
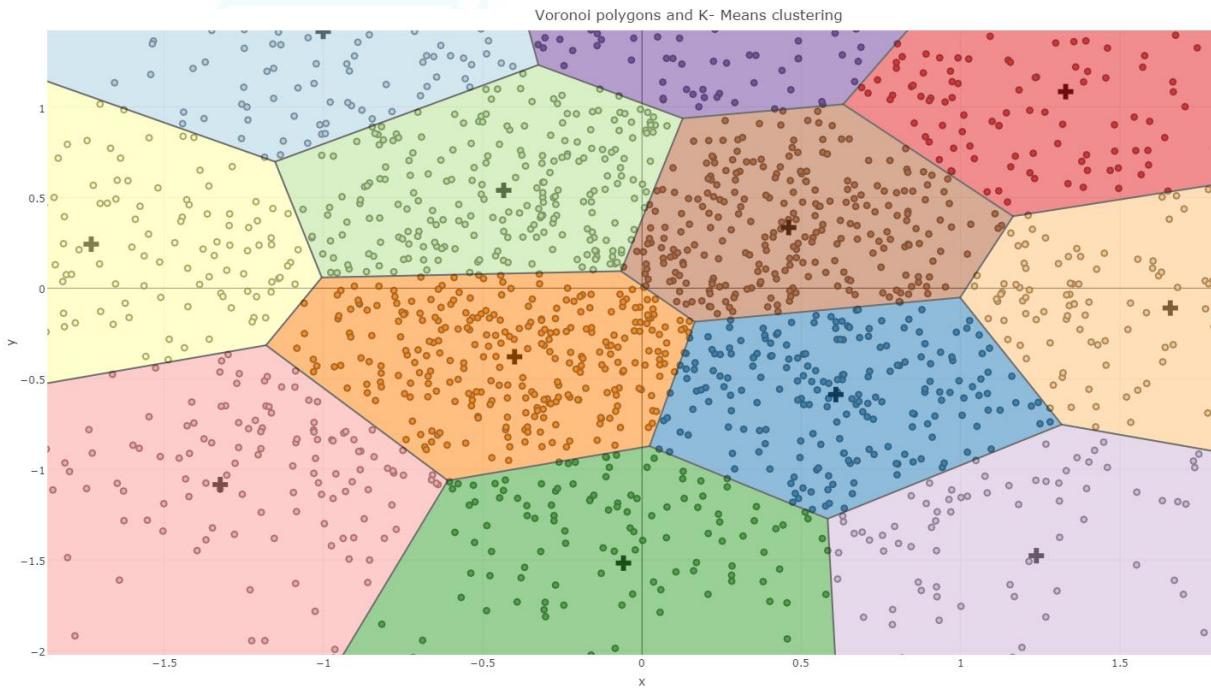
Kmeans



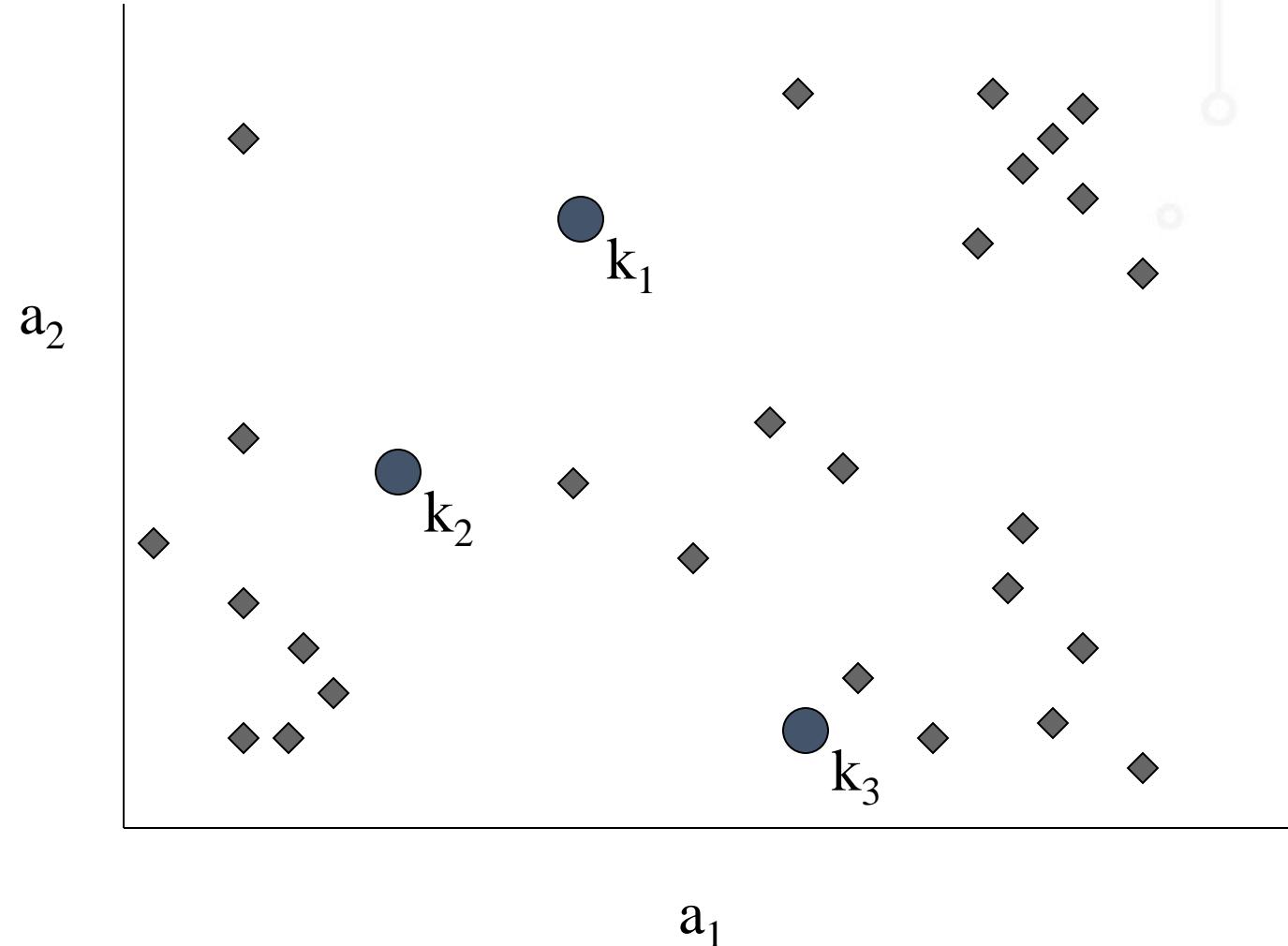
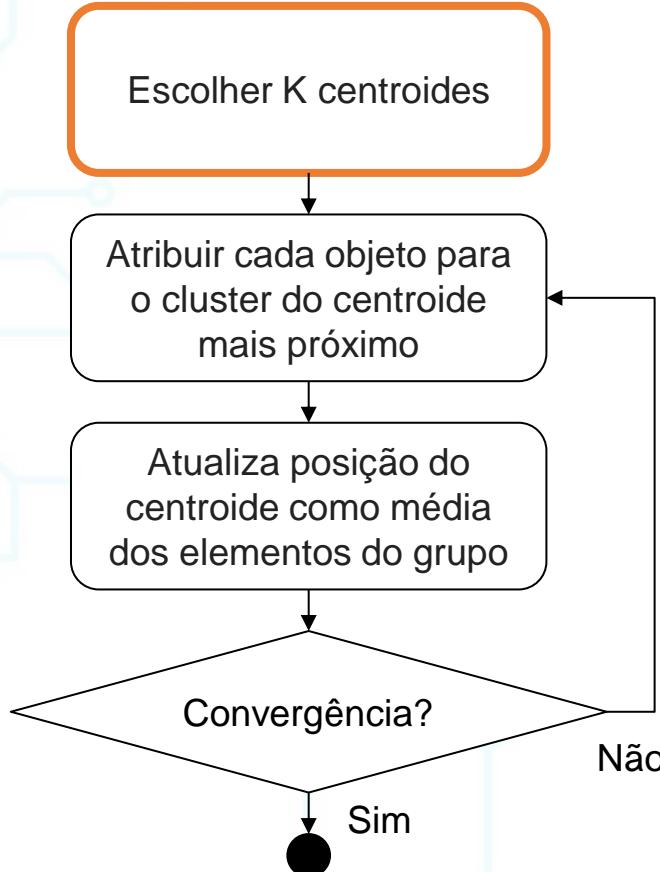
K-Means

Objetiva partitionar N observações dentre k grupos em que cada observação pertence ao grupo mais próximo da média. Isso resulta em uma divisão do espaço de dados em um Diagrama de Voronoi.

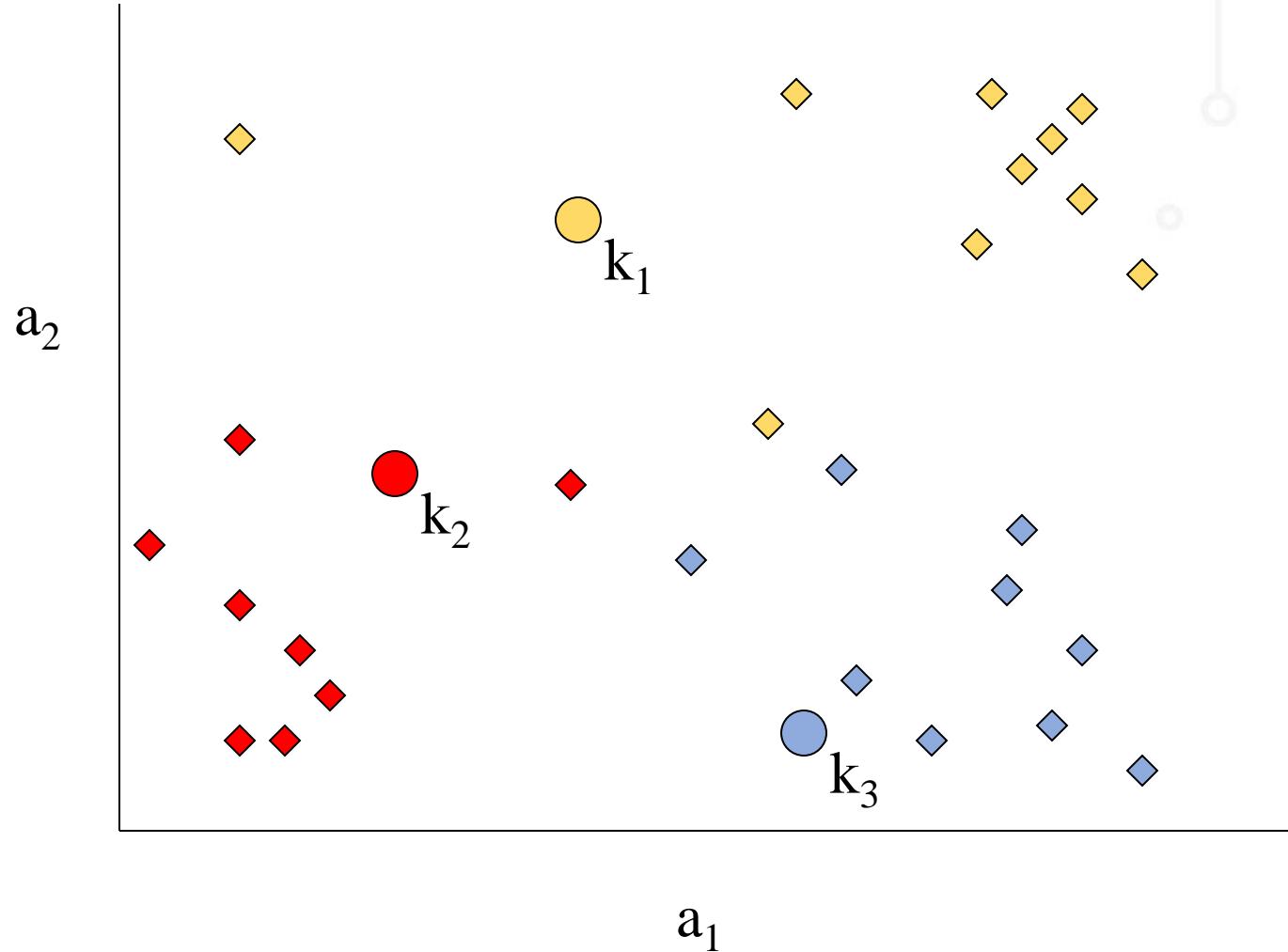
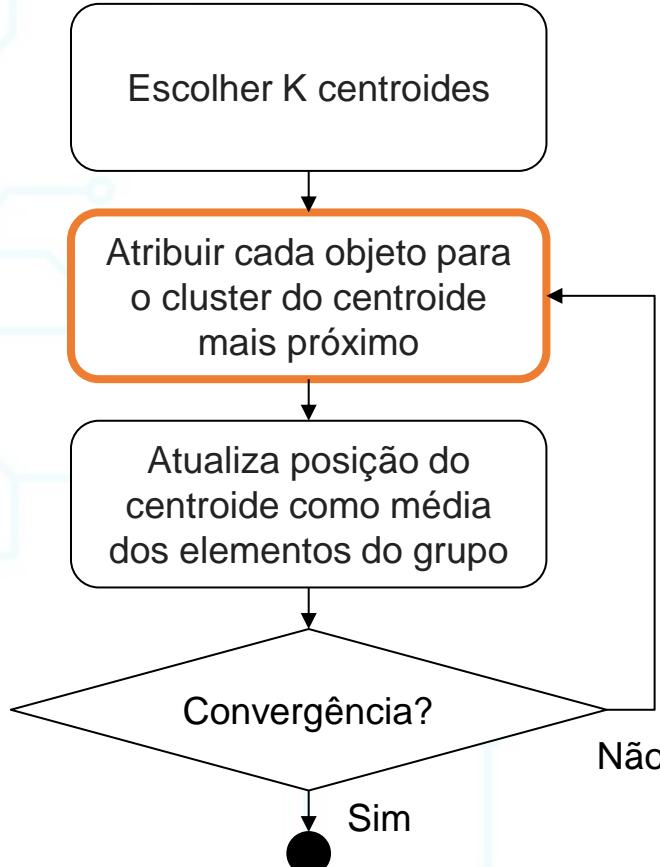
Calculado por meio da triangulação de Delaunay



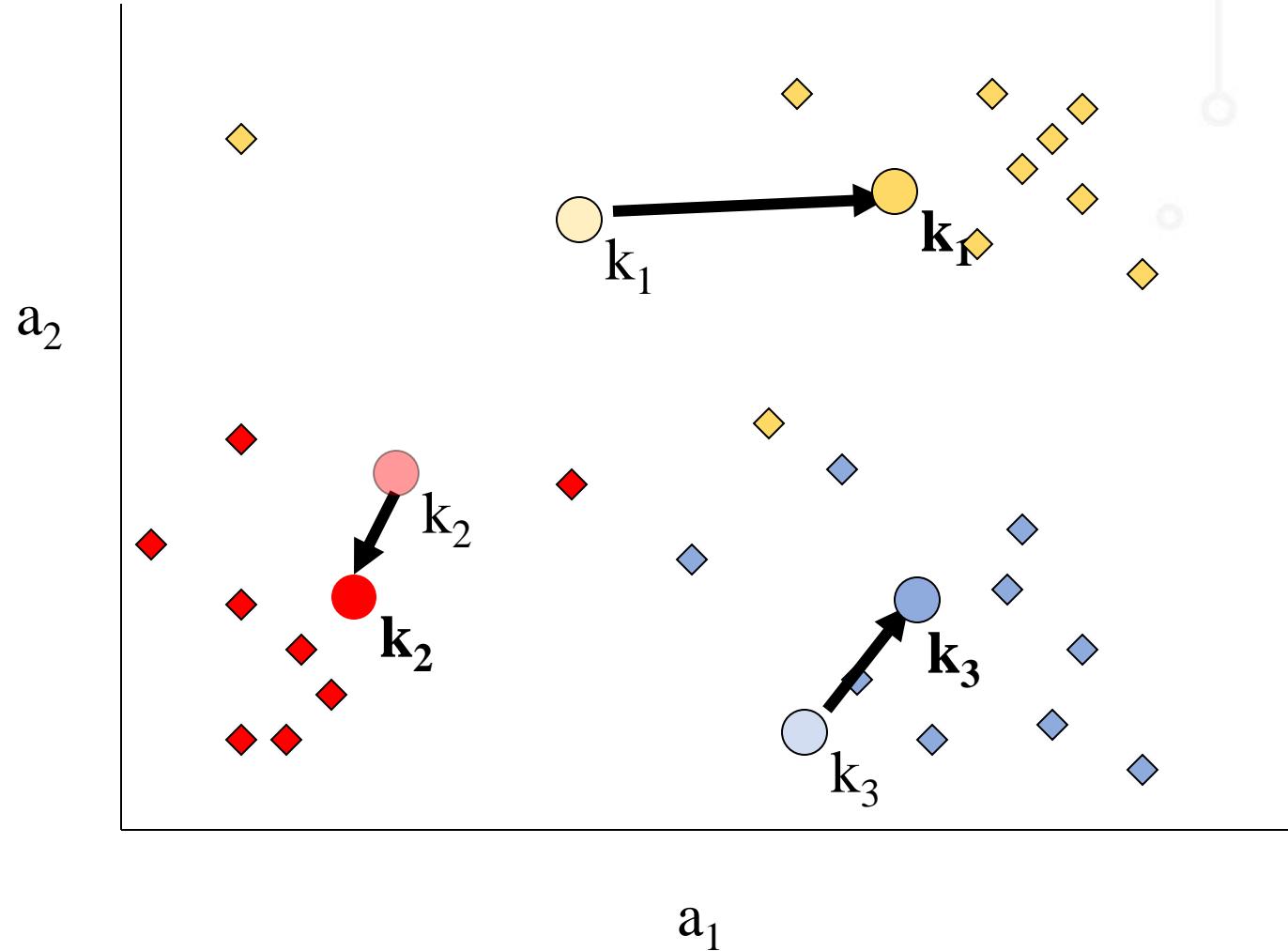
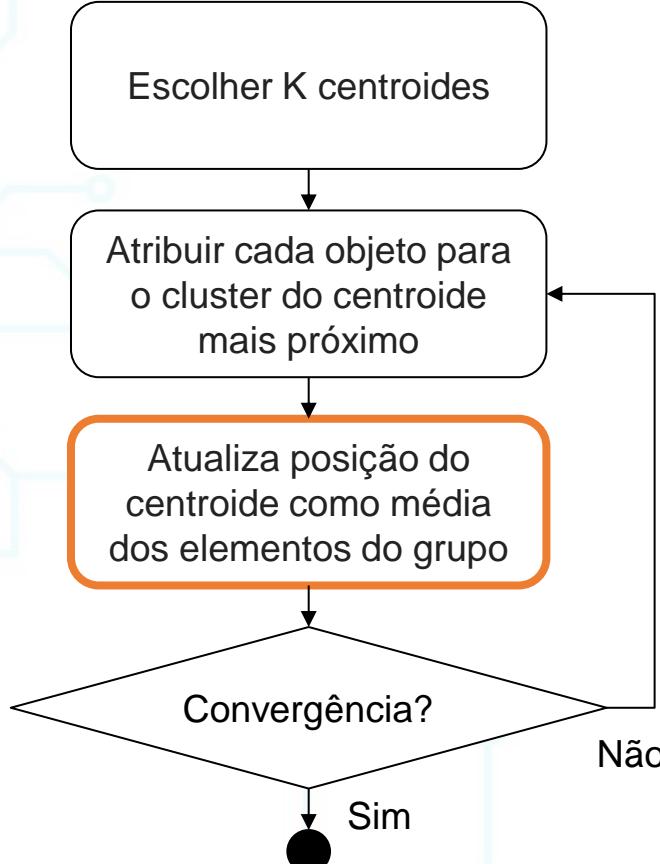
K-Means - Simulação



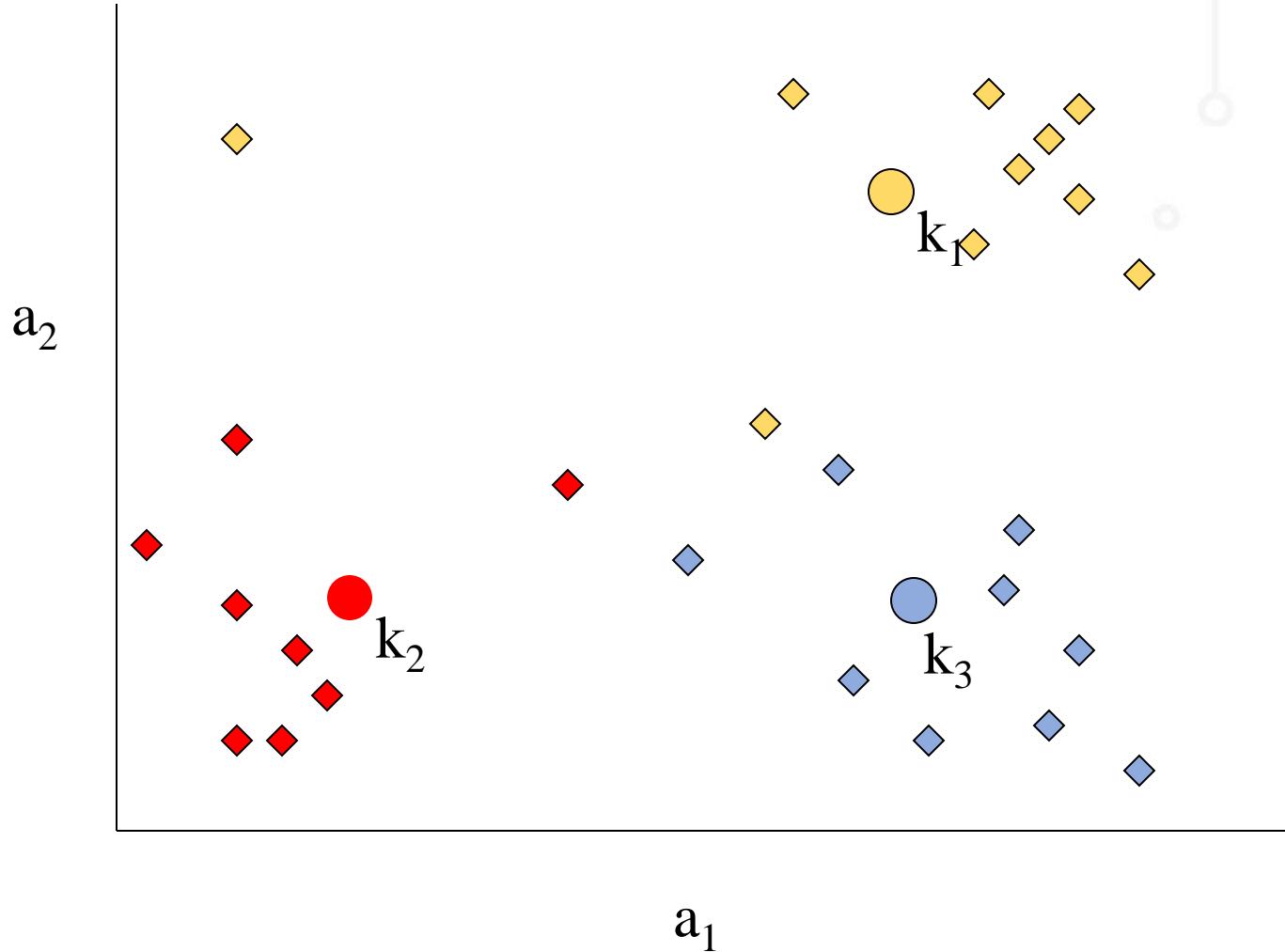
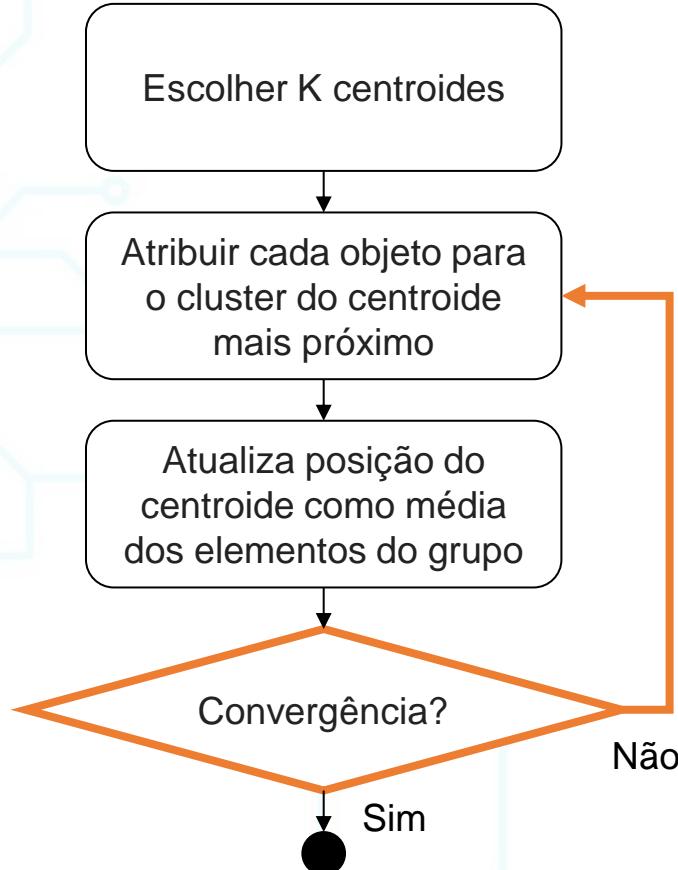
K-Means - Simulação



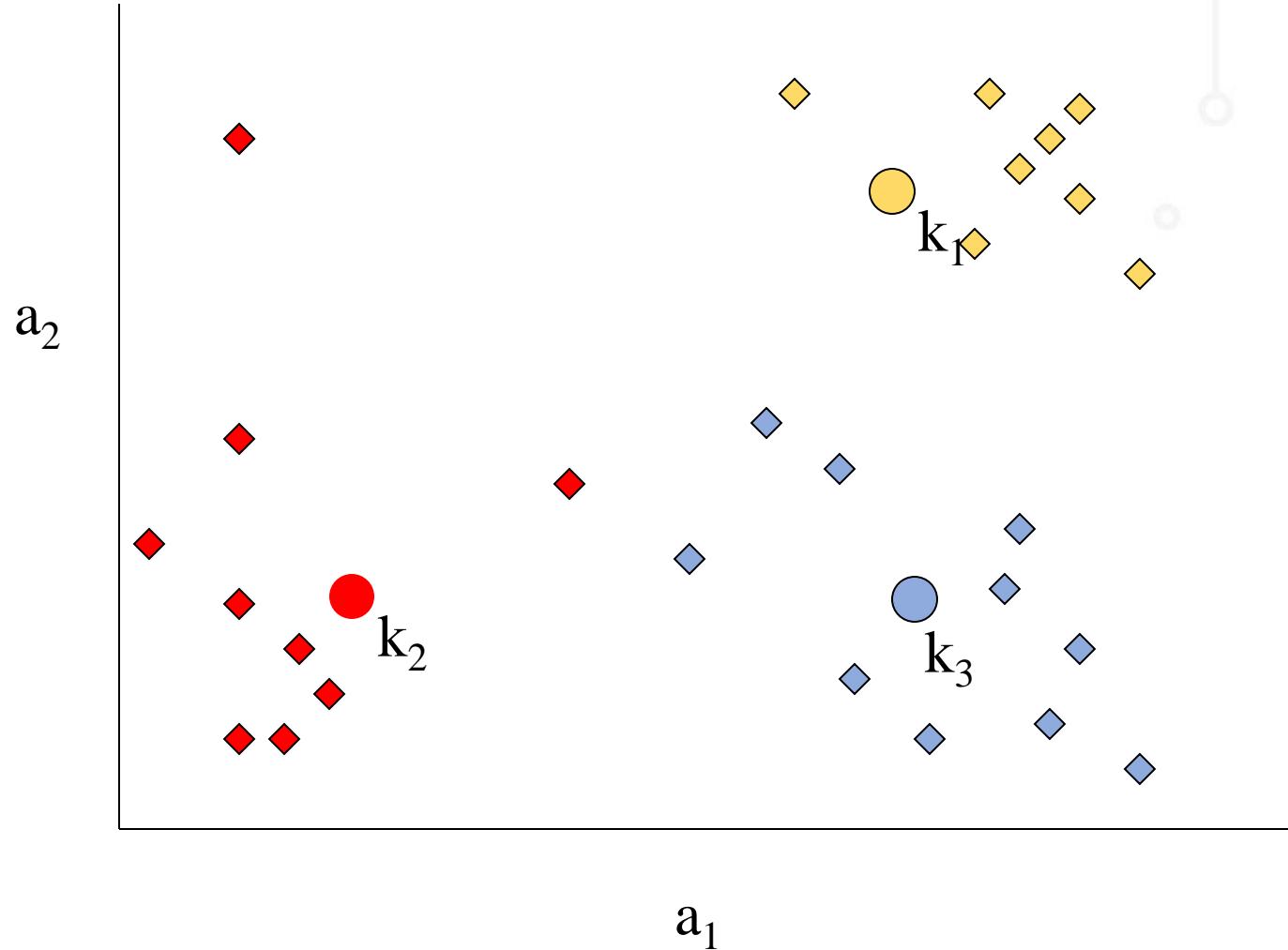
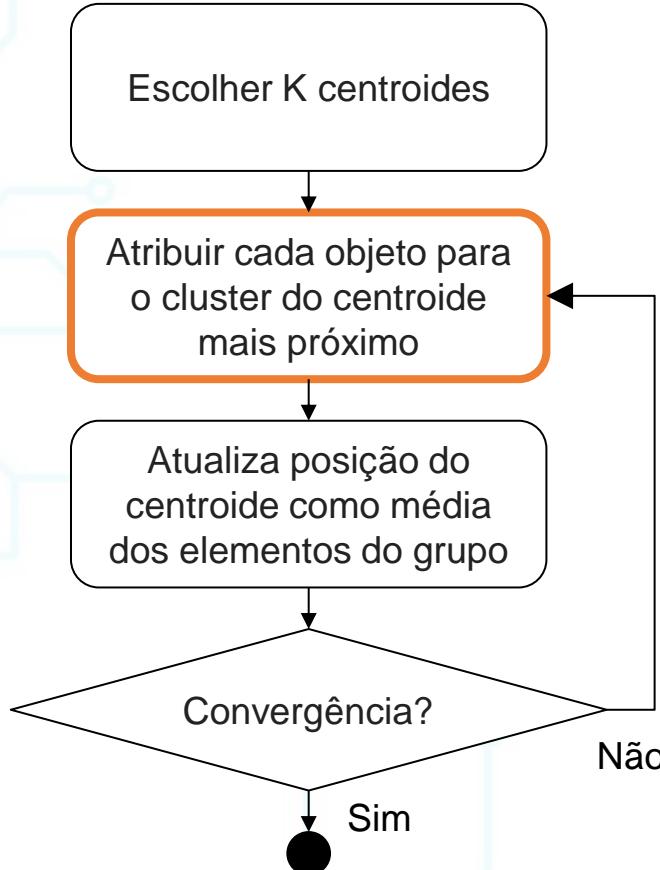
K-Means - Simulação



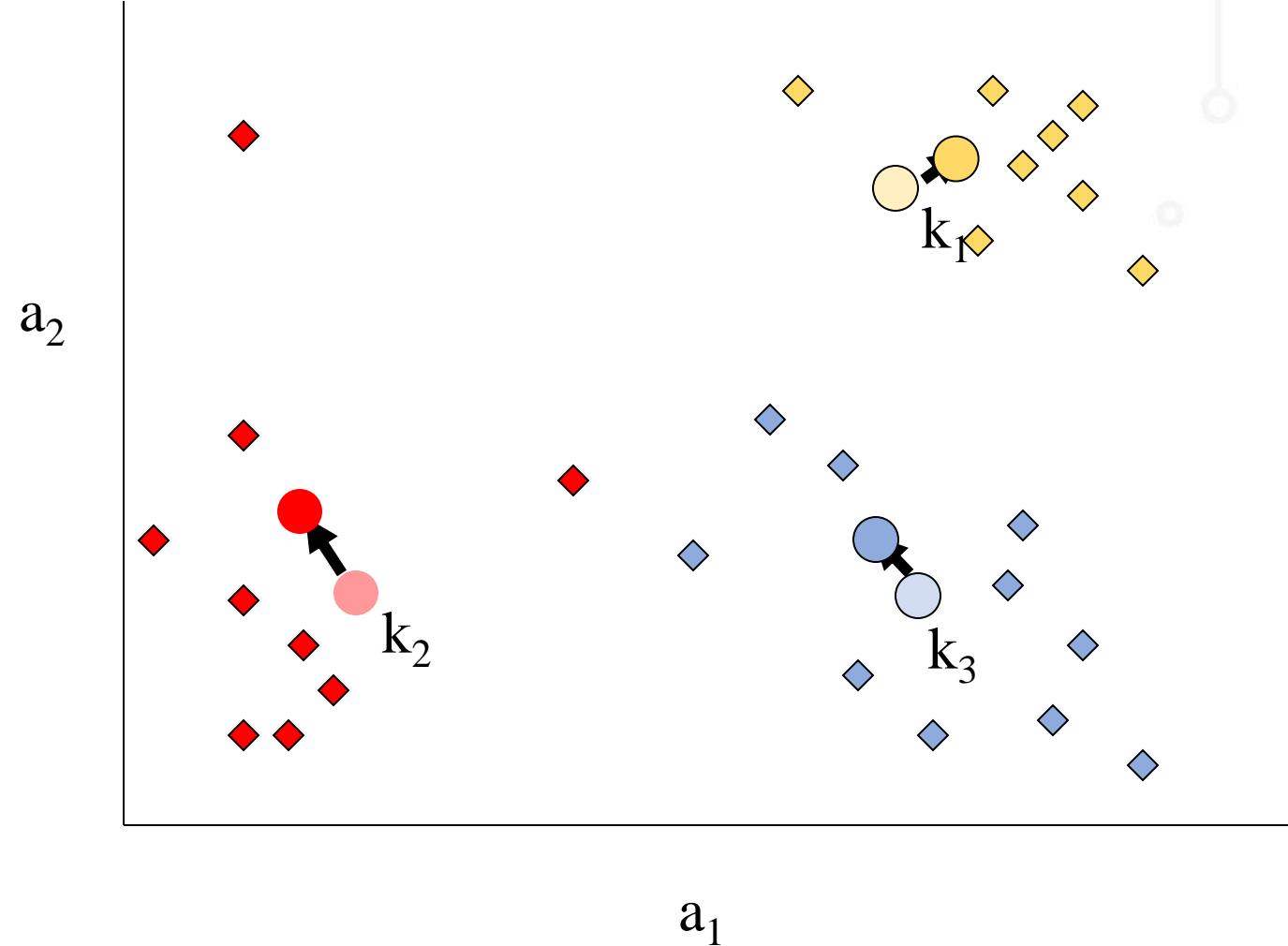
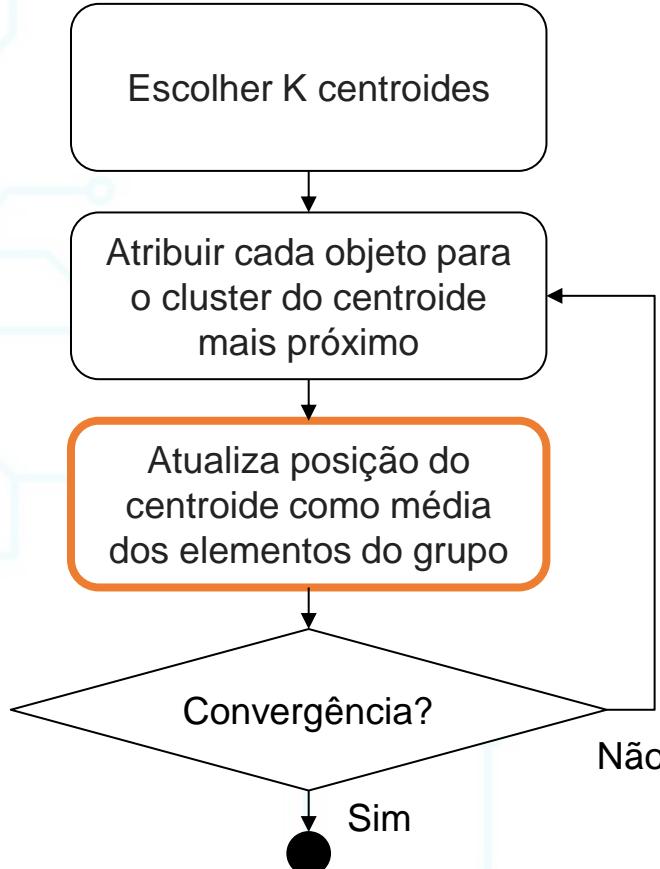
K-Means - Simulação



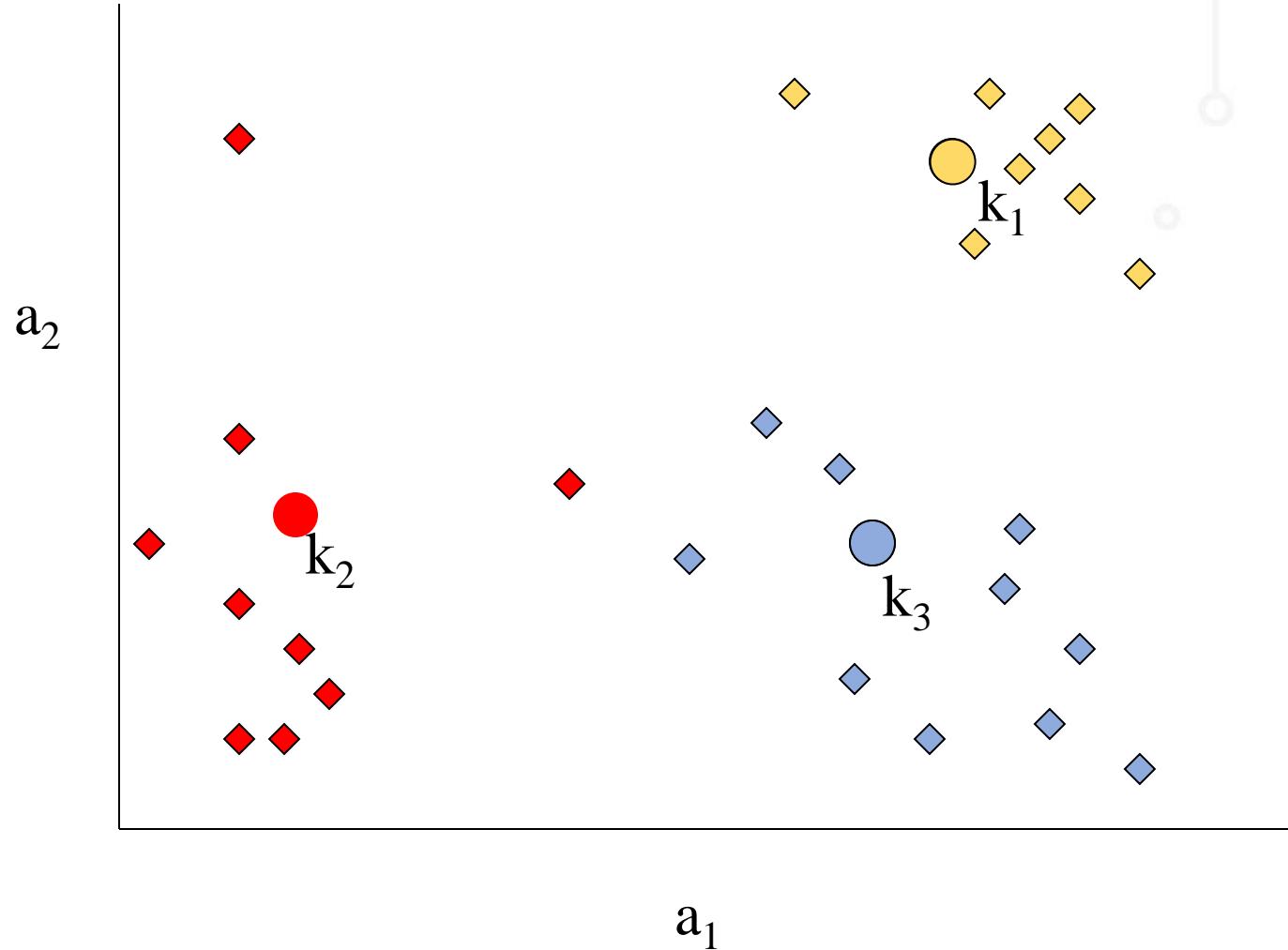
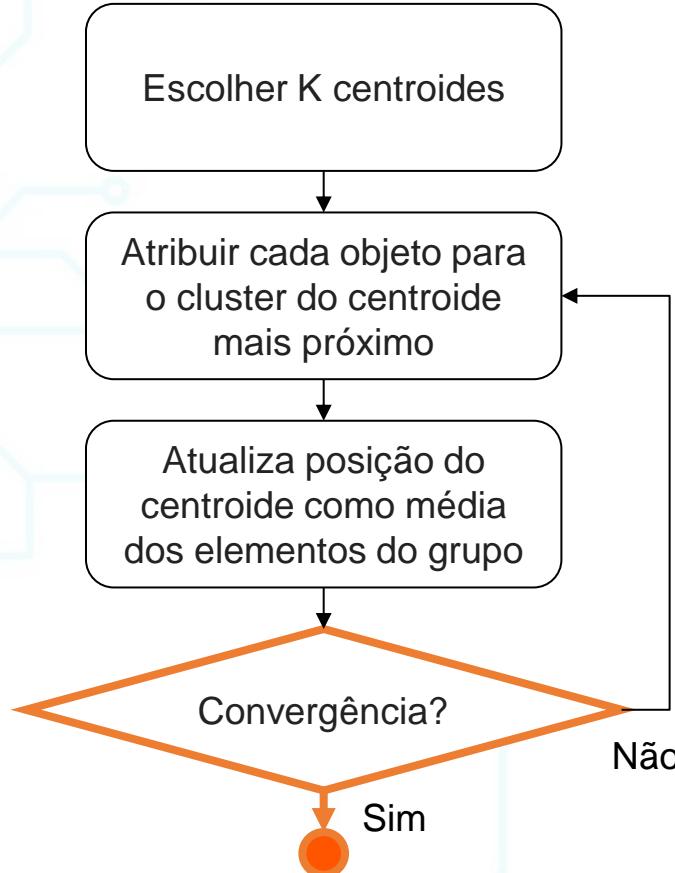
K-Means - Simulação



K-Means - Simulação



K-Means - Simulação



K-Means - Animação

<https://www.naftaliharris.com/blog/visualizing-k-means-clustering/>



K-Means

CALMA

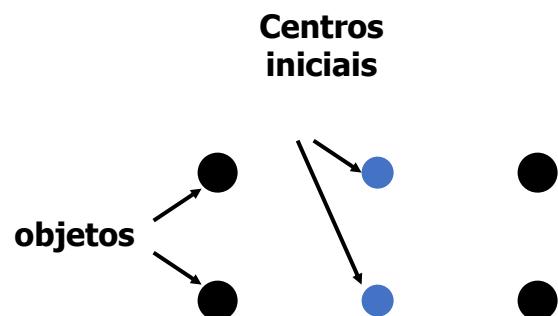


NEM TUDO SÃO FLORES

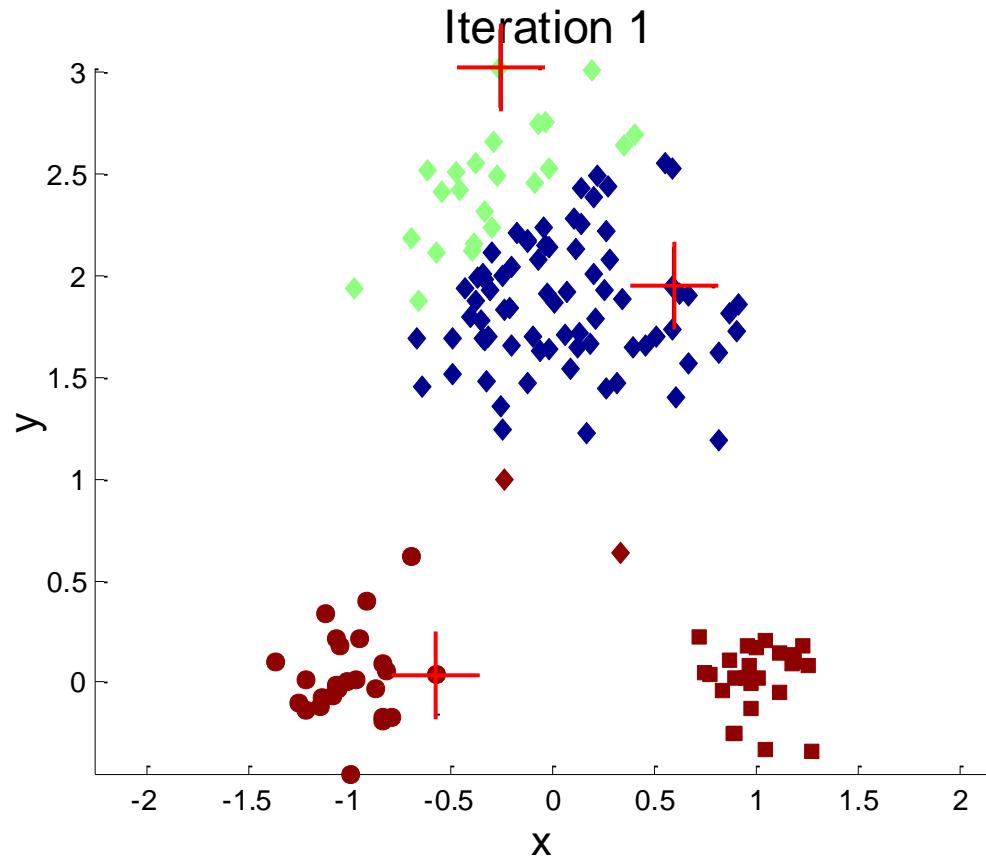
K-Means: Sensibilidade em relação à inicialização

Resultado pode variar significativamente dependendo da escolha das sementes (protótipos) iniciais

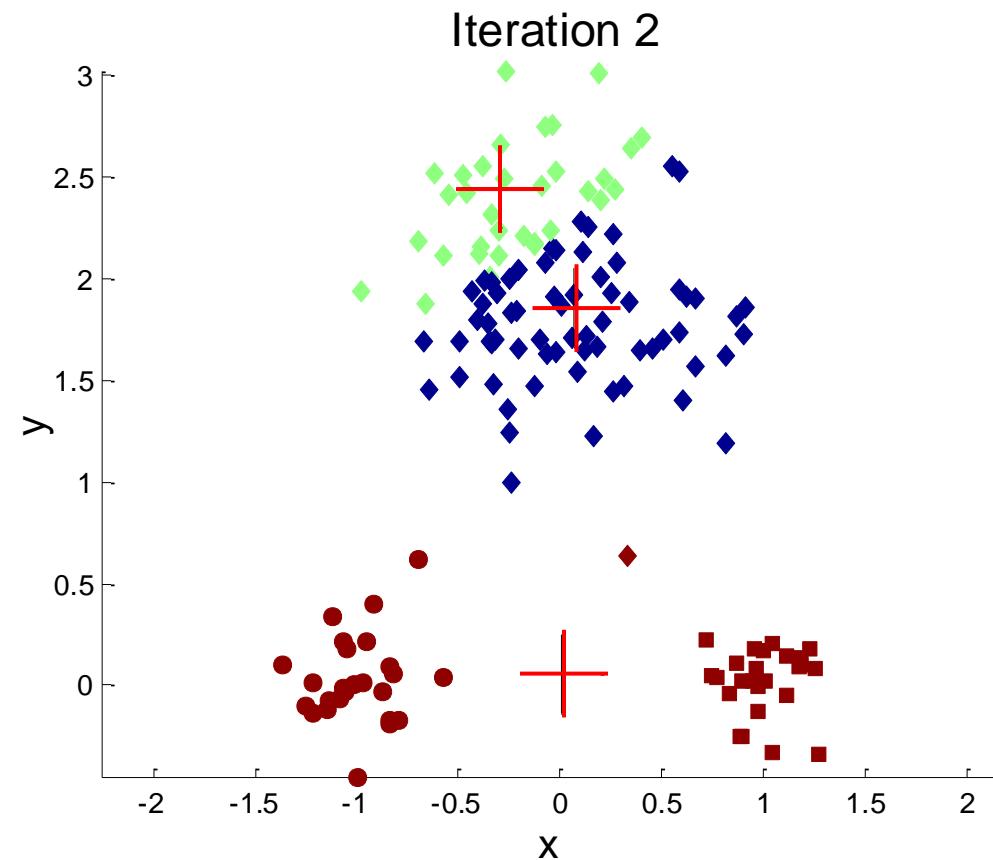
k-means pode “ficar preso” em ótimos locais:



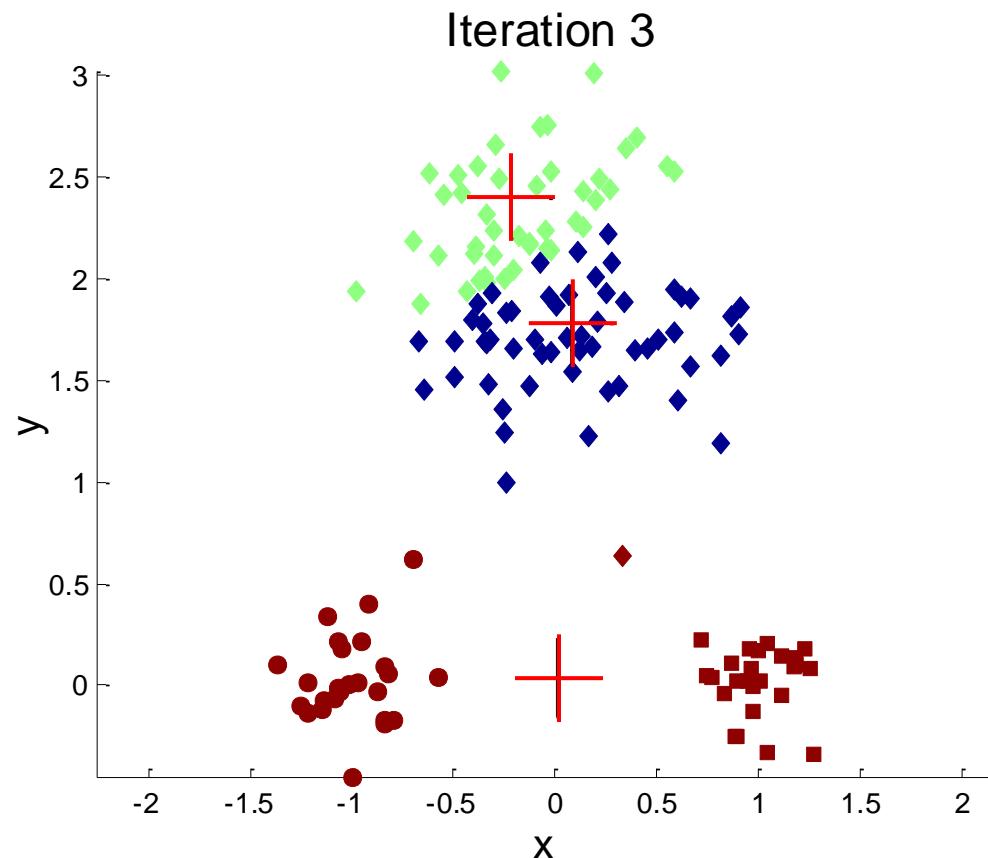
K-Means: Sensibilidade em relação à inicialização



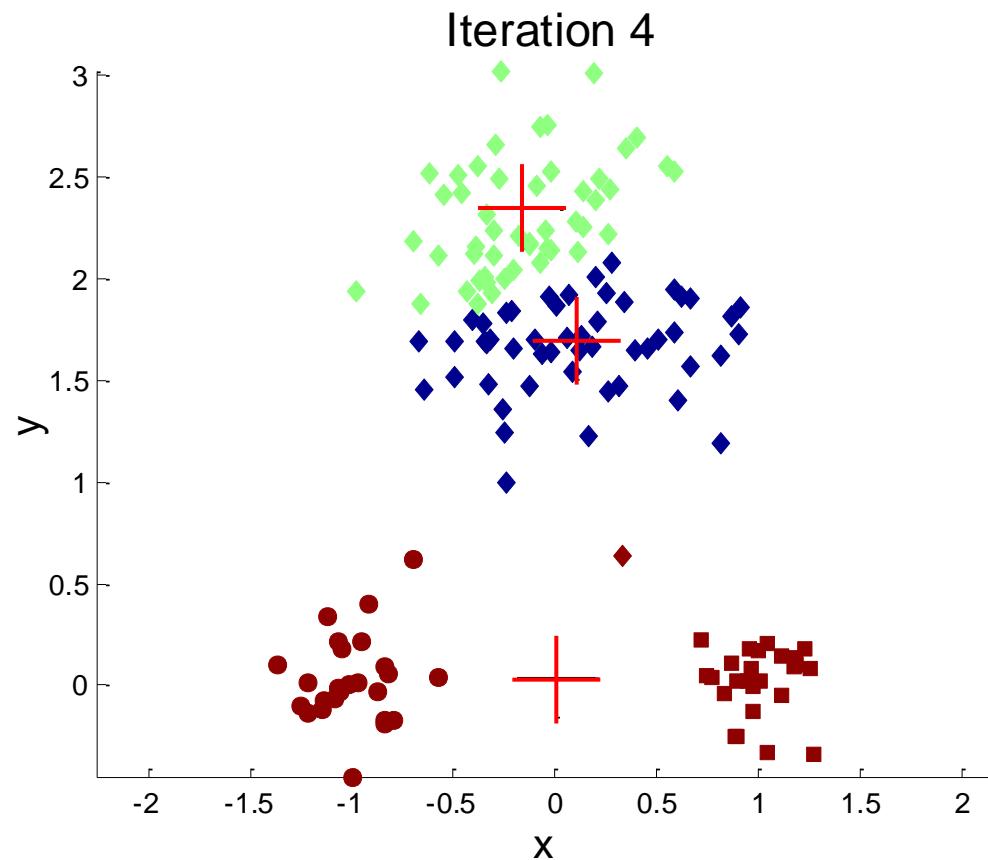
K-Means: Sensibilidade em relação à inicialização



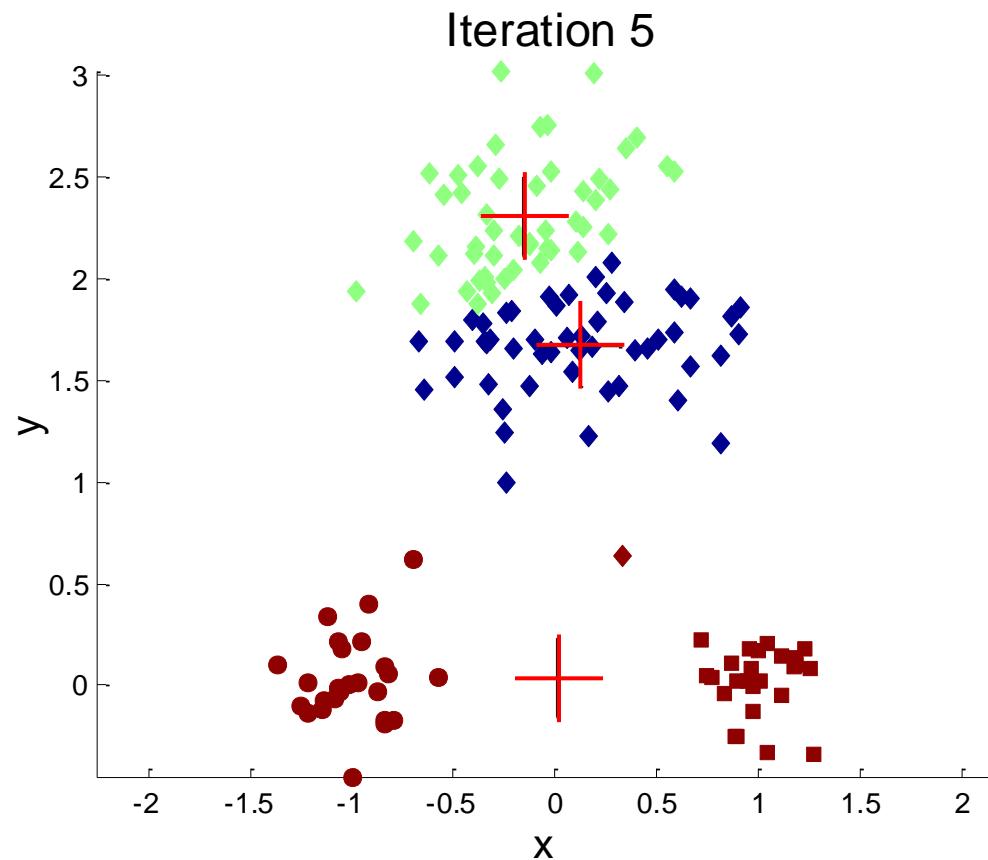
K-Means: Sensibilidade em relação à inicialização



K-Means: Sensibilidade em relação à inicialização



K-Means: Sensibilidade em relação à inicialização



K-Means: Sensibilidade em relação à inicialização

Múltiplas Execuções (inicializações aleatórias):

- Funciona bem em muitos problemas;
- Pode demandar muitas execuções (especialmente com k alto).

Agrupamento Hierárquico:

- agrupa-se uma amostra dos dados para tomar os centros da partição com k grupos.

Seleção “informada” em uma amostra dos dados:

- Tomar 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.

Busca Guiada:

- X-means, k-means evolutivo, ...

K-Means

CALMA AÍ!



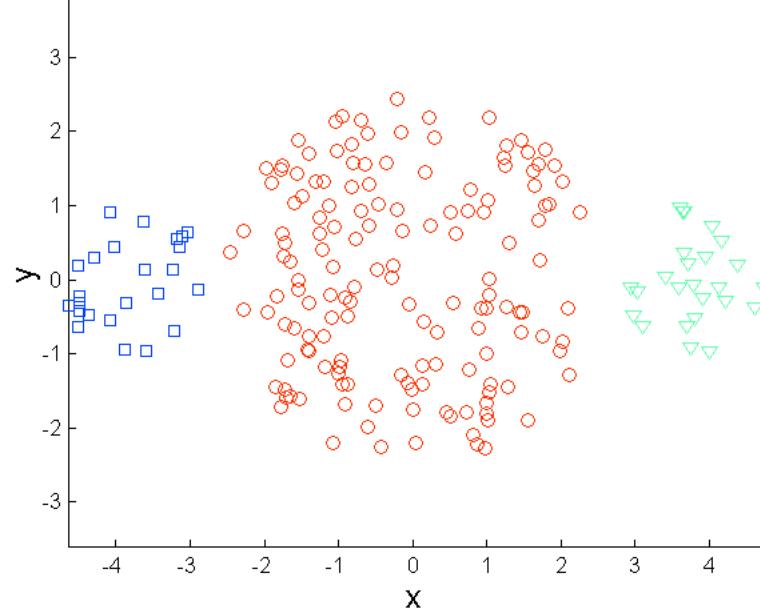
NÃO É SÓ ISSO..

K-Means: Problemas estruturais

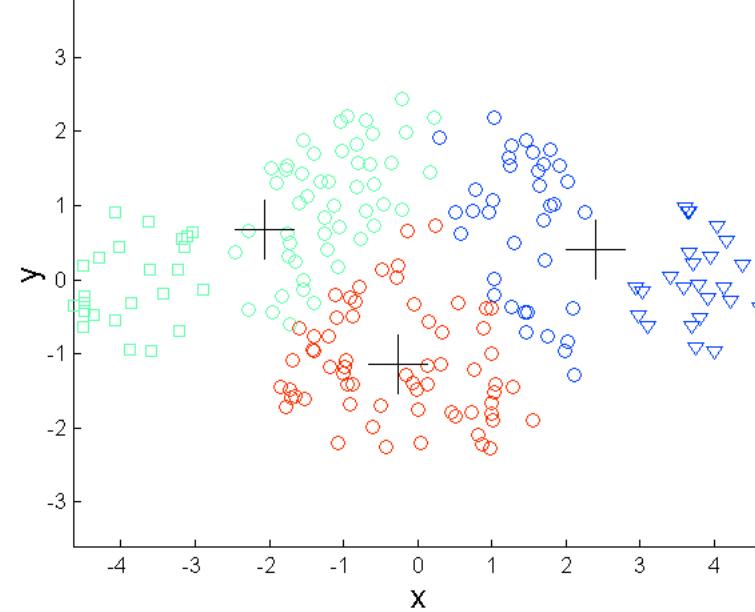
Algoritmo *k*-means funciona bem se:

- Clusters são (hiper)esféricos e bem separados
- Clusters de volumes aproximadamente iguais
- Cluster com quantidades de pontos semelhantes
- Formas Globulares

K-Means: Problemas estruturais

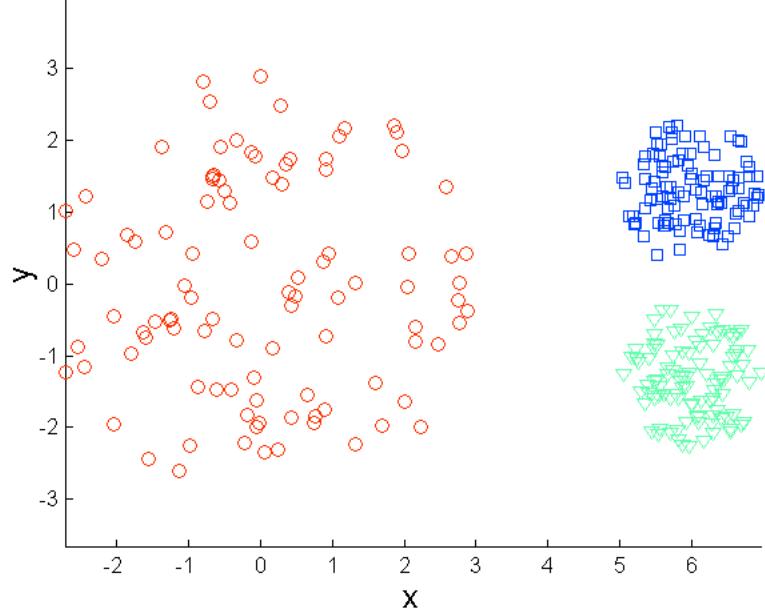


Estrutura correta

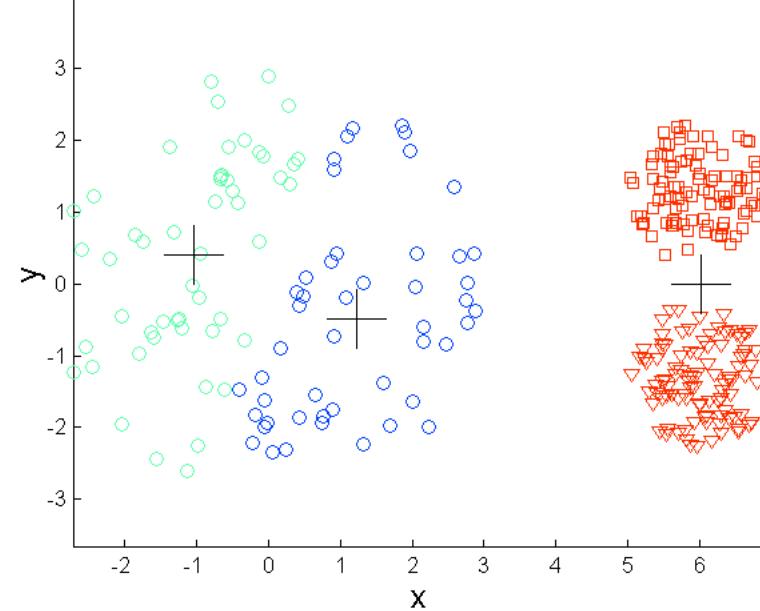


k-means (3 Clusters)

K-Means: Problemas estruturais

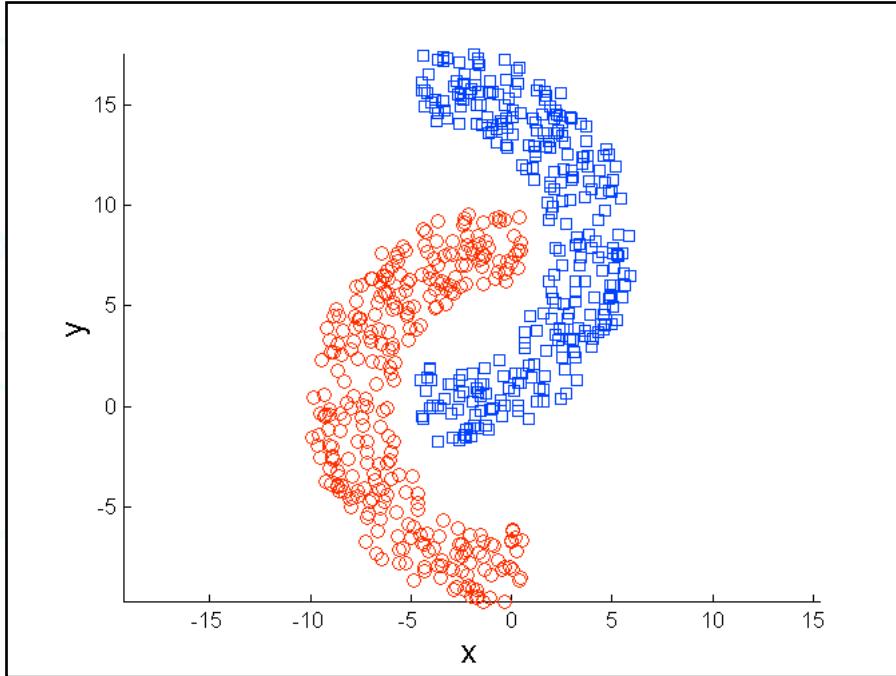


Estrutura correta

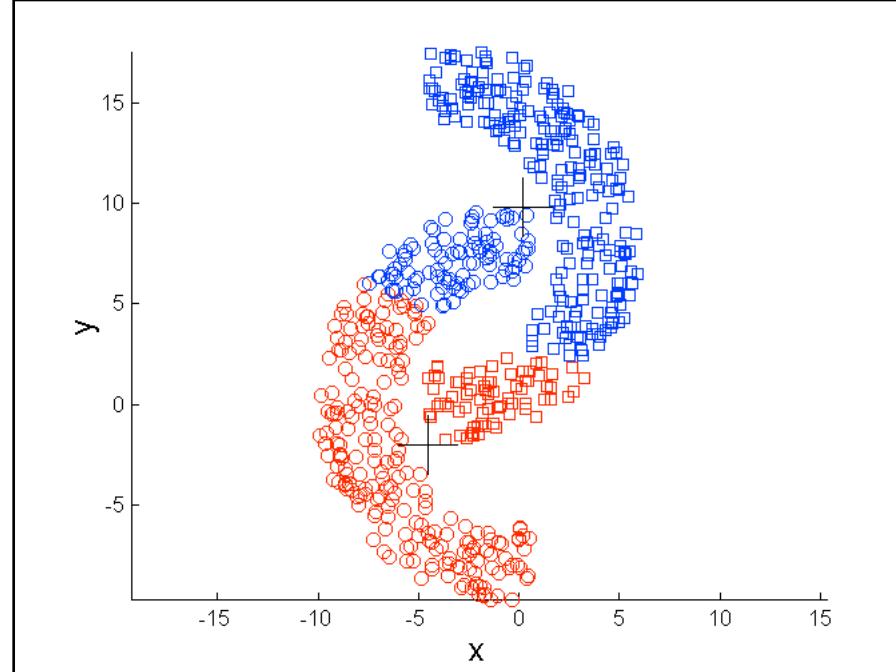


K-means (3 Clusters)

K-Means: Problemas estruturais



Estrutura correta



K-means (3 Clusters)

(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.

K-Means: Custo Computacional

Complexidade (assintótica) de tempo:

$$O(i \cdot K \cdot N \cdot n)$$

- O que isso significa?

O que dizer sobre a constante de tempo?

→ Computar Distância Euclidiana via aproximações sucessivas (Newton-Raphson) custa caro.

Se também tenho problema de espaço em memória...

→ Solução aproximada (*sampling*).
→ Paralelizar (mesmo computador) ou distribuir (e.g., map-reduce) o processamento.



Resumo das (des)vantagens do k-means

Vantagens

- Simples e intuitivo
- Complexidade **linear** em todas as variáveis críticas
- Eficaz em muitos cenários de aplicação
- Resultados de interpretação simples

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**)
- Limitado a atributos numéricos
- Sensível a *outliers*

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...

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)

“

DB Scan

DBSCAN

DBSCAN looks for densely packed observations and makes no assumptions about the number or shape of clusters.

1. A random observation, x_i , is selected
2. If x_i has a minimum of close neighbors, we consider it part of a cluster.
3. Step 2 is repeated recursively for all of x_i 's neighbors, then neighbors' neighbors etc... These are the cluster's core members.
4. Once Step 3 runs out of observations, a new random point is chosen

Afterwards, observations not part of a core are assigned to a nearby cluster or marked as outliers.

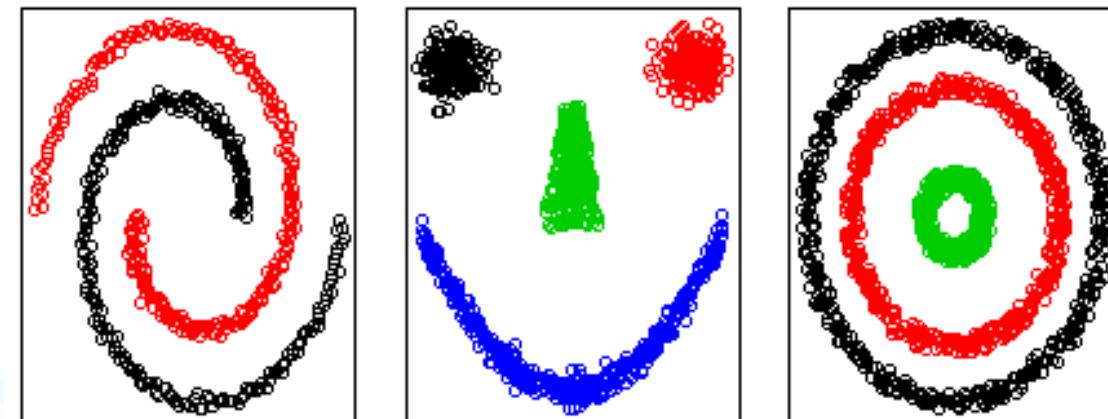
ChrisAlbon

Algoritmos Baseados em Densidade

Paradigma de Agrupamento por Densidade

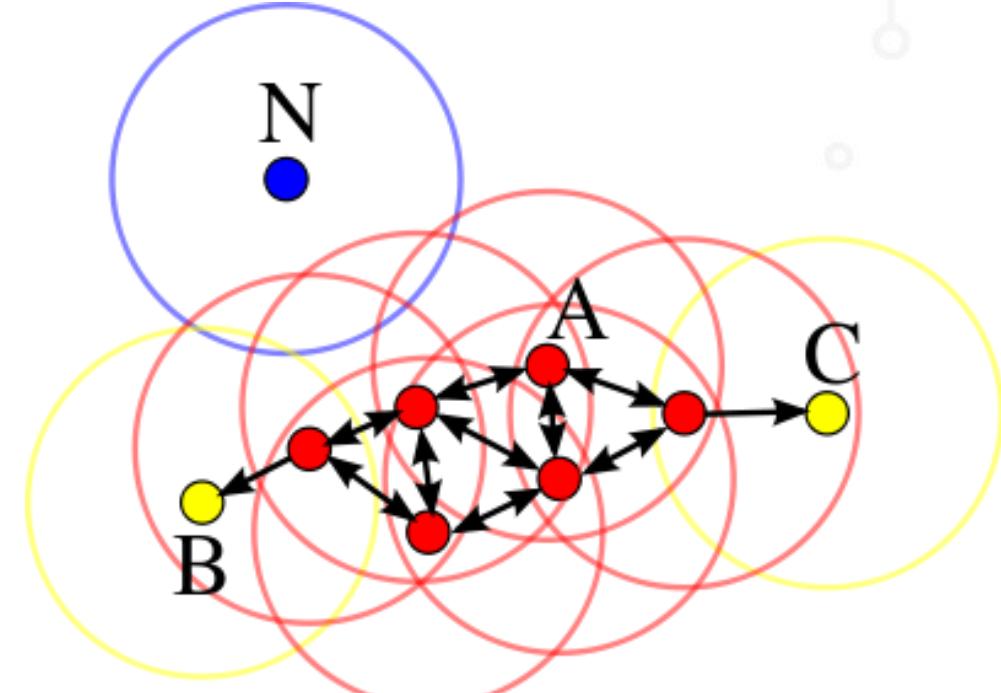
- Clusters como regiões de alta concentração de objetos separadas por regiões de baixa concentração de objetos
- Paradigma alternativo àquele baseado em protótipos: K-means e variantes, EM, etc

Existem vários algoritmos, veremos a seguir um dos mais conhecidos: **DBSCAN**



DBScan: definições

- A point is a **core** point if it has at least a specified number of points (MinPts) within the radius Eps (including the point itself)
 - These are points that are in the interior of a cluster
- A **border** point has fewer than MinPts within Eps, but is in the neighborhood (within the radius) of at least 1 core point
- A **noise** point is neither a core point nor a border point

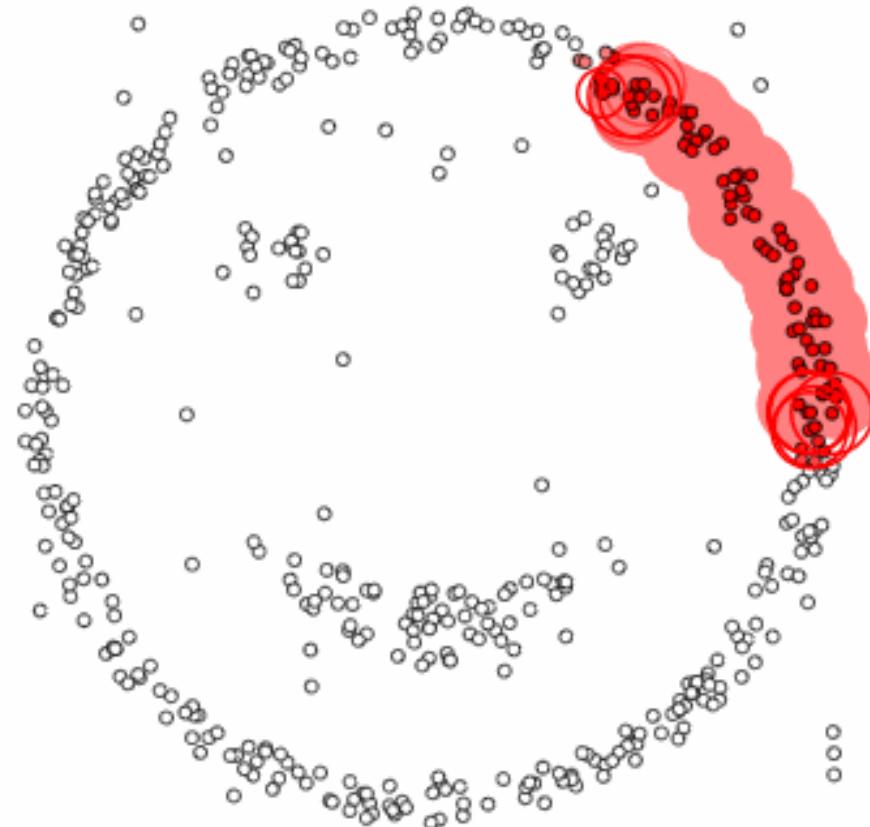


Algoritmo Conceitual:

1. Percorra a BD e rotule os objetos como core, border ou noise
2. Elimine aqueles objetos rotulados como **noise**
3. Insira uma aresta entre cada par de objetos **core** vizinhos
 - 2 objetos são vizinhos se um estiver dentro do raio Eps do outro
4. Faça cada componente conexo resultante ser um cluster
5. Atribua cada **border** ao cluster de um de seus core associados
 - Resolva empates se houver objetos core associados de diferentes clusters

DBScan: algoritmo

epsilon = 1.00
minPoints = 4

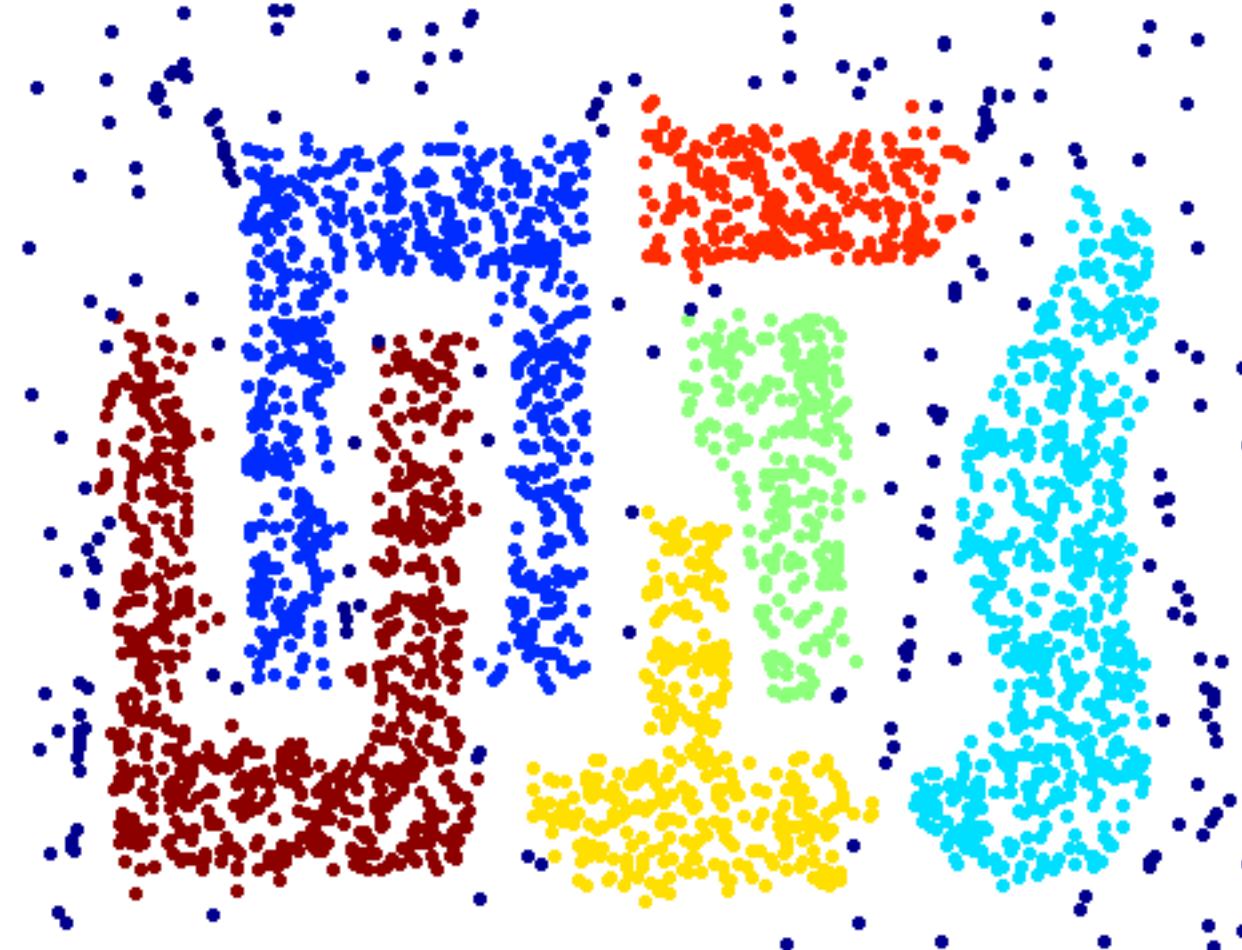


Restart



Pause

DBScan: Exemplo



Point types: **core**, **border** and **noise**

Resumo das (des)vantagens do DBScan

Vantagens

- Não necessita do número de clusters a priori
- Consegue encontrar clusters com formatos arbitrários
- Tem uma definição de ruído e é robusto a outliers
- Necessita de apenas dois parametros:
 - Raio
 - Número de vizinhos para virar core (minpts)

Desvantagens

- Extremamente sensível aos parametros Raio e minPts
- Depende da distância utilizada para determiner se um ponto está ou nao presente dentro do raio. (tipicamente se utiliza euclidiana)
- Não consegue clusterizar dados com grupos com grandes diferenças de densidades
- Se a escala dos dados não for conhecida, desteminar o raio pode ser difícil

Implementação de Exemplo

“

AI Generativa

[https://www.proquest.com/openview/a17fb9188f537c2ab3df4091453547ba/1?pq-](https://www.proquest.com/openview/a17fb9188f537c2ab3df4091453547ba/1?pq-origsite=gscholar&cbl=18750&diss=y)

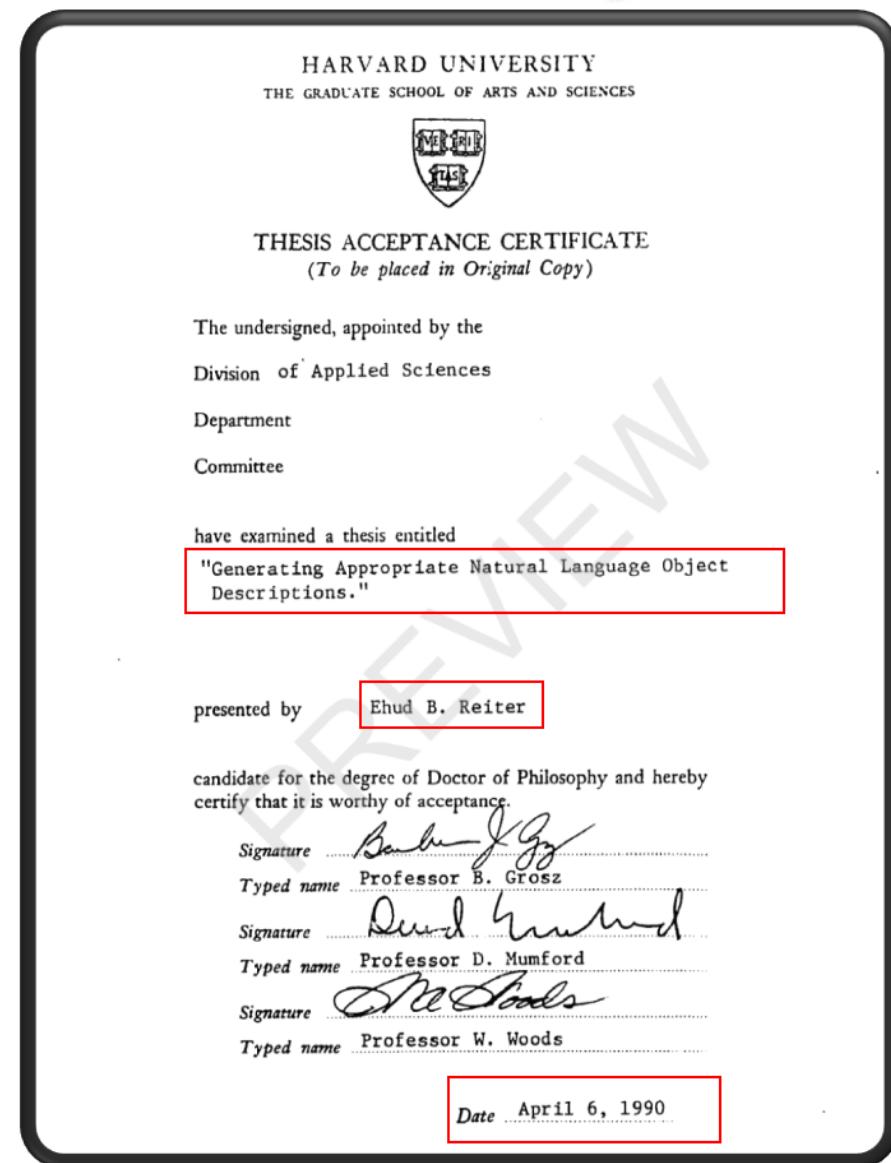
AI Generativa

Artificial Intelligence

Machine Learning

Deep Learning

Generative AI



Como representar contexto/significado das palavras

Você sabe qual o significado da palavra **tezgüino**?



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:

- Uma garrafa de **tezgüino** está sobre a mesa.



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:

- Uma garrafa de **tezgüino** está sobre a mesa.
- Todo mundo gosta de beber **tezgüino**.



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:

- Uma garrafa de **tezgüino** está sobre a mesa.
- Todo mundo gosta de beber **tezgüino**.
- Você pode ficar bêbado com **tezgüino**.



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:

- Uma garrafa de **tezgüino** está sobre a mesa.
- Todo mundo gosta de beber **tezgüino**.
- Você pode ficar bêbado com **tezgüino**.
- **Tezgüino** é feito de milho.



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:

- Uma garrafa de **tezgüino** está sobre a mesa.
- Todo mundo gosta de beber **tezgüino**.
- Você pode ficar bêbado com **tezgüino**.
- **Tezgüino** é feito de milho.

Consegue entender o que é **tezgüino**?



Como representar contexto/significado das palavras

Observe a palavra **tezgüino** em diferentes contextos:

- Uma garrafa de **tezgüino** está sobre a mesa.
- Todo mundo gosta de beber **tezgüino**.
- Você pode ficar bêbado com **tezgüino**.
- **Tezgüino** é feito de milho.

Com o contexto, conseguimos identificar do que se refere a palavra **tezgüino**.

Tezgüino:= é uma bebida alcoólica feita a base de milho.



Como representar contexto/significado das palavras

Como o cérebro faz isso?



Como representar contexto/significado das palavras

Quais outras palavras se “*encaixam*” nos slots das perguntas 1 até 4?

1. Uma garrafa de _____ está sobre a mesa.



Como representar contexto/significado das palavras

Quais outras palavras se “*encaixam*” nos slots das perguntas 1 até 4?

1. Uma garrafa de _____ está sobre a mesa.
2. Todo mundo gosta de beber _____.



Como representar contexto/significado das palavras

Quais outras palavras se “*encaixam*” nos slots das perguntas 1 até 4?

1. Uma garrafa de _____ está sobre a mesa.
2. Todo mundo gosta de beber _____.
3. Você pode ficar bêbado com _____.



Como representar contexto/significado das palavras

Quais outras palavras se “*encaixam*” nos slots das perguntas 1 até 4?

1. Uma garrafa de _____ está sobre a mesa.
2. Todo mundo gosta de beber _____.
3. Você pode ficar bêbado com _____.
4. _____ é feito de milho.



Como representar contexto/significado das palavras

Inserindo contexto de forma manual...

1. Uma garrafa de _____ está sobre a mesa.
2. Todo mundo gosta de beber _____.
3. Você pode ficar bêbado com _____.
4. _____ é feito de milho.

	(1)	(2)	(3)	(4) ← contextos
tezgüino	1	1	1	1
som	0	0	0	0
suco de laranja	1	1	0	0
vinho	1	1	1	0



Como representar contexto/significado das palavras

Inserindo contexto de forma manual...

1. Uma garrafa de _____ está sobre a mesa.
2. Todo mundo gosta de beber _____.
3. Você pode ficar bêbado com _____.
4. _____ é feito de milho.

	(1)	(2)	(3)	(4)	← contextos
tezgüino	1	1	1	1	
som	0	0	0	0	
suco de laranja	1	1	0	0	
vinho	1	1	1	0	

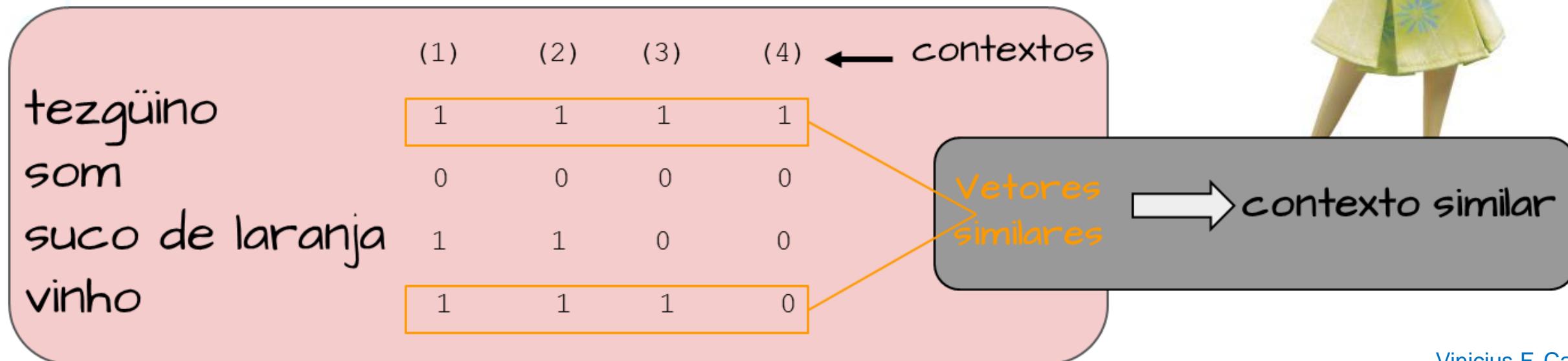
Vetores similares



Como representar contexto/significado das palavras

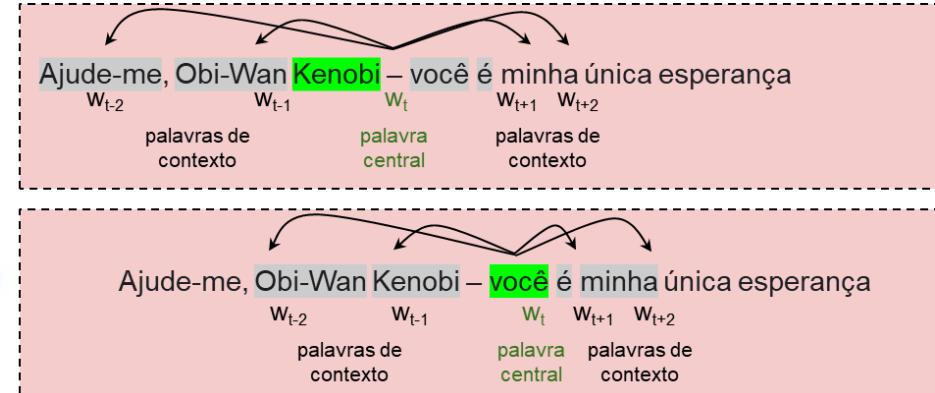
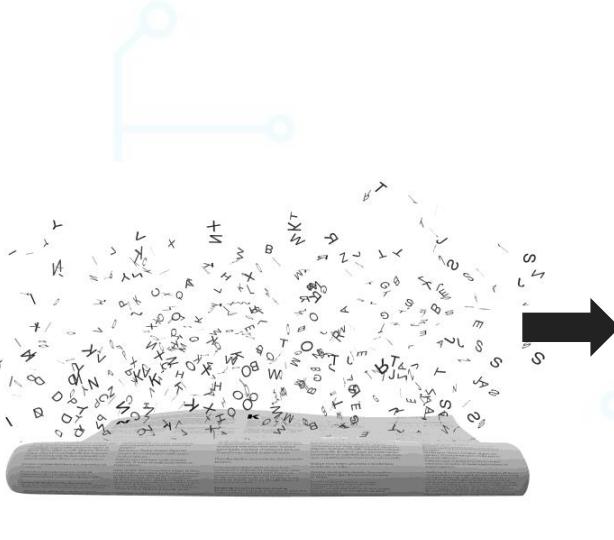
Inserindo contexto de forma manual...

1. Uma garrafa de _____ está sobre a mesa.
2. Todo mundo gosta de beber _____.
3. Você pode ficar bêbado com _____.
4. _____ é feito de milho.



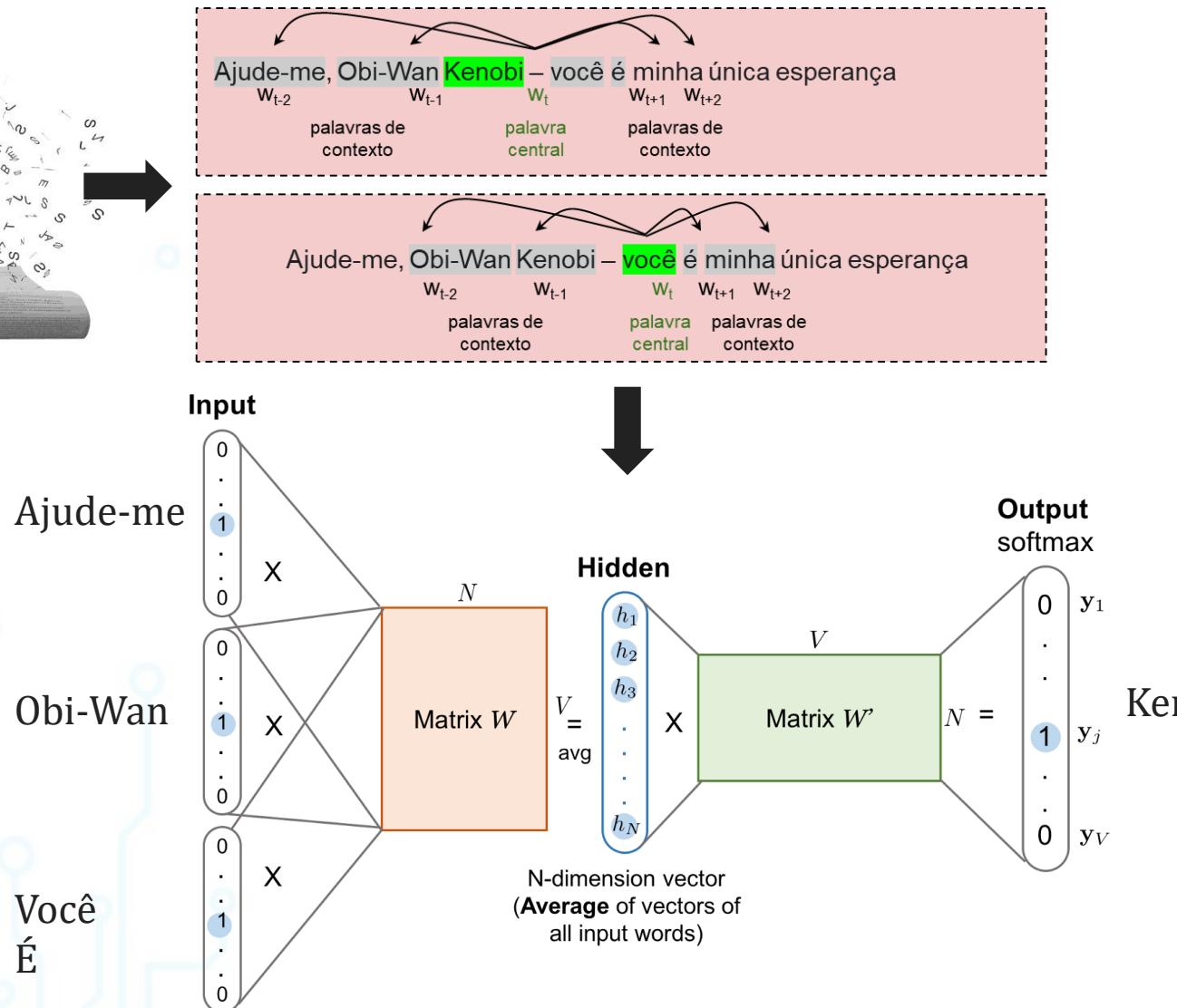
Processamento de Linguagem Natural

Natural Language Processing (NLP)



Processamento de Linguagem Natural

Natural Language Processing (NLP)

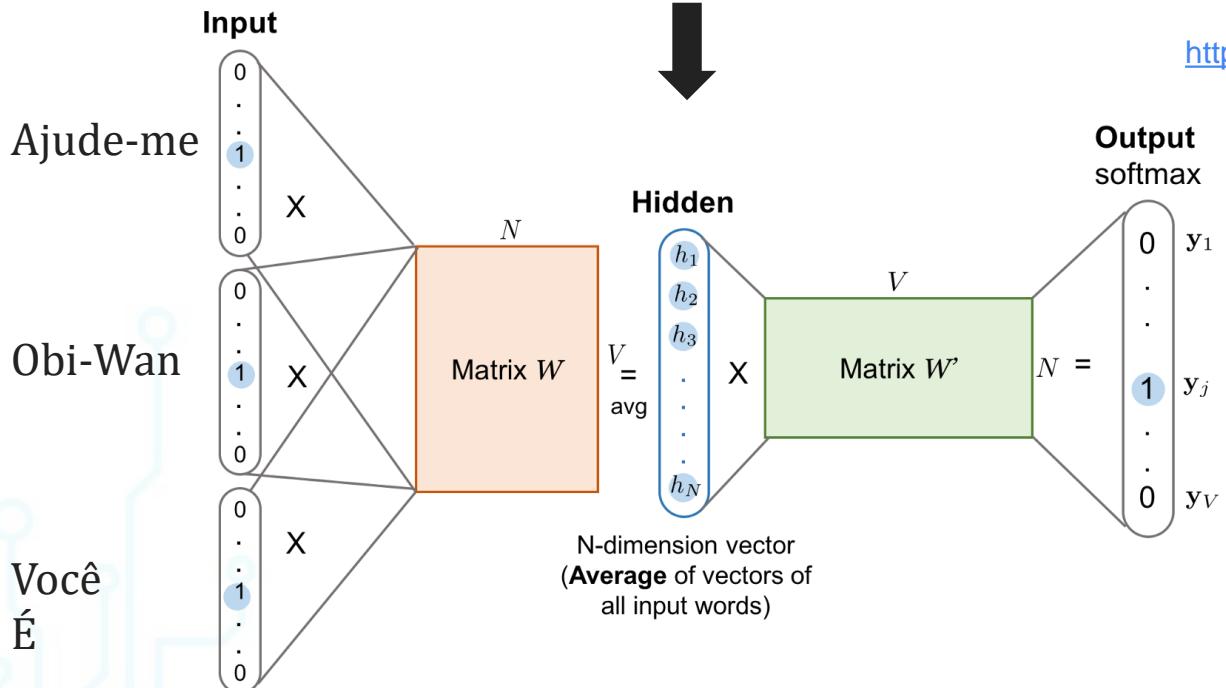
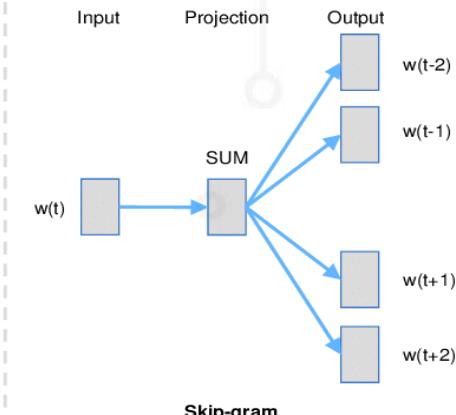
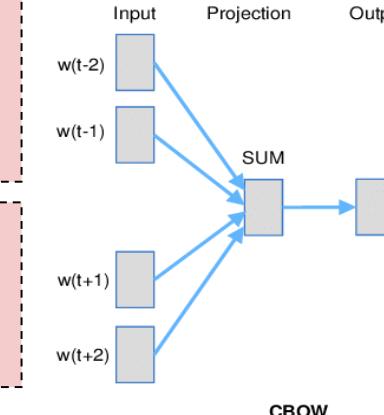
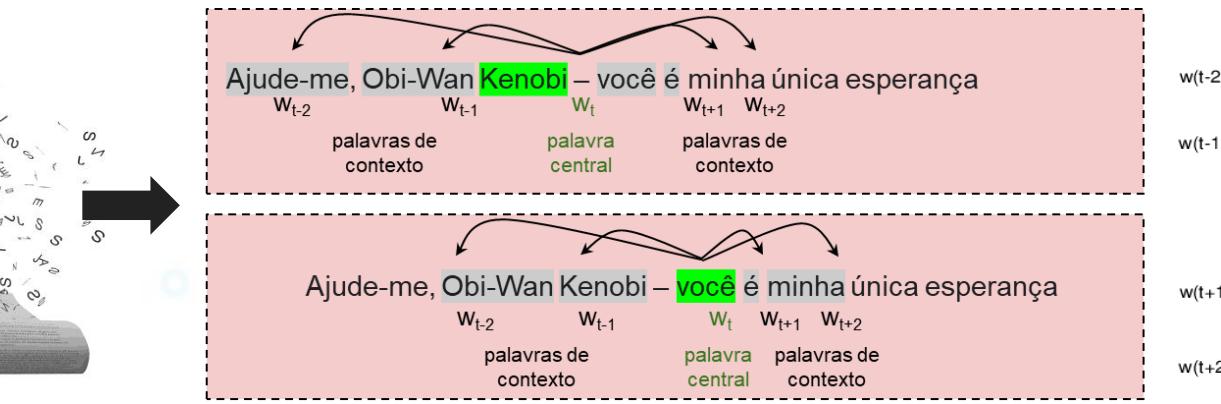


MultiLayer Perceptron MLP
<https://dl.acm.org/doi/10.5555/1639537.1639542>

Self Supervised Learning (SSL)
<https://arxiv.org/abs/2110.09327>

Processamento de Linguagem Natural

Natural Language Processing (NLP)



<https://arxiv.org/abs/1301.3781>

<https://arxiv.org/pdf/1310.4546.pdf>

MultiLayer Perceptron MLP

<https://dl.acm.org/doi/10.5555/1639537.1639542>

Self Supervised Learning (SSL)

<https://arxiv.org/abs/2110.09327>

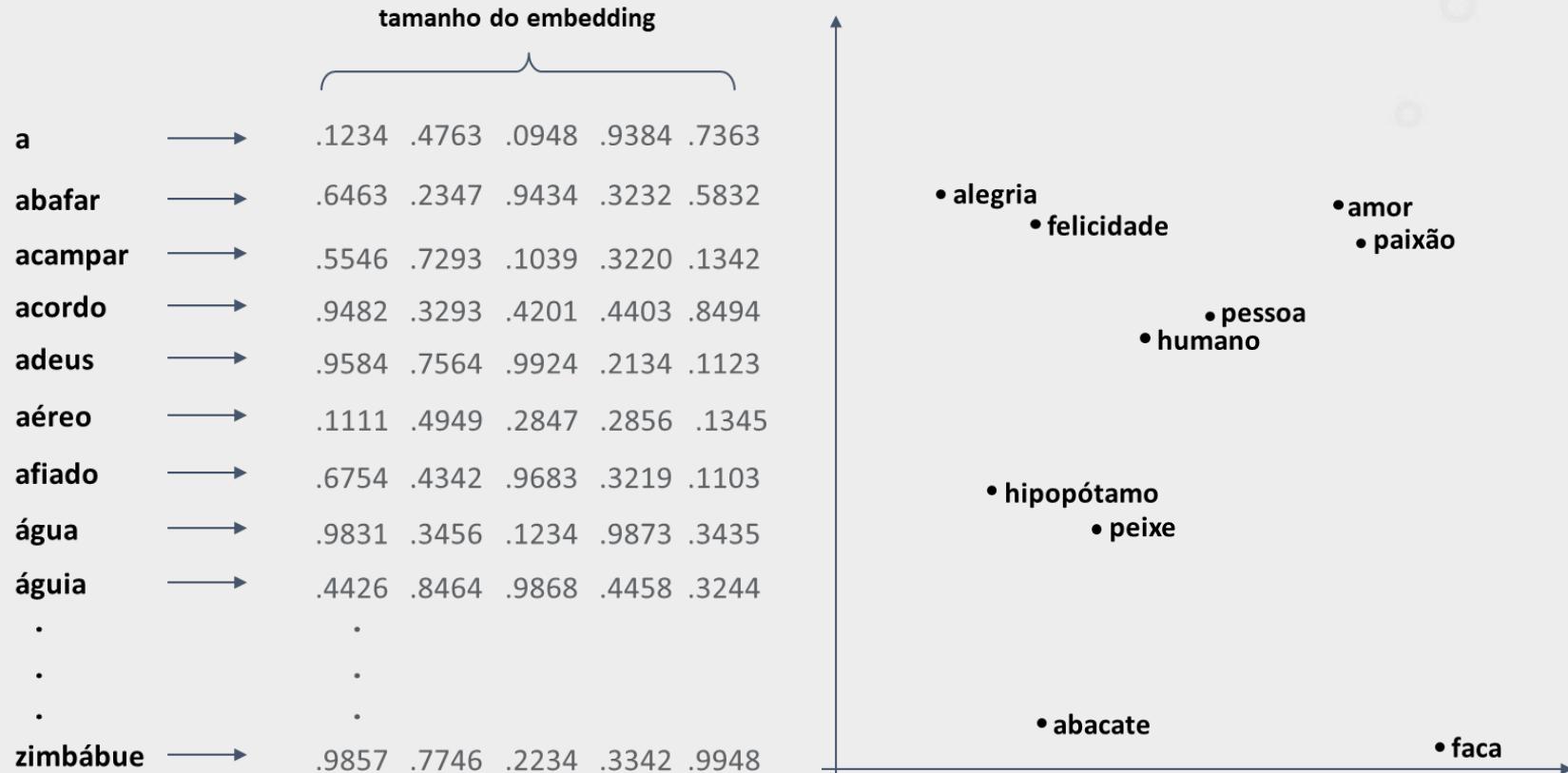
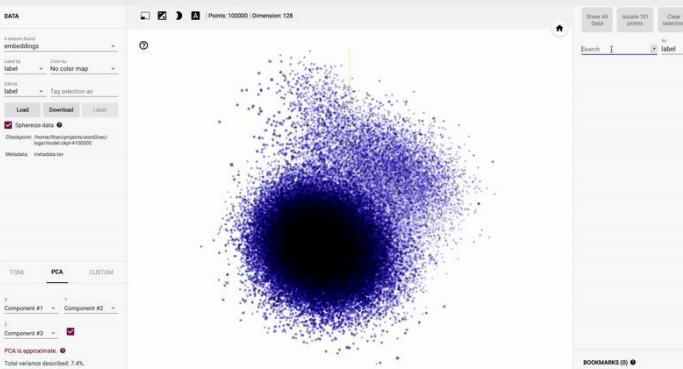
Processamento de Linguagem Natural

Natural Language Processing (NLP)

Word Embedding

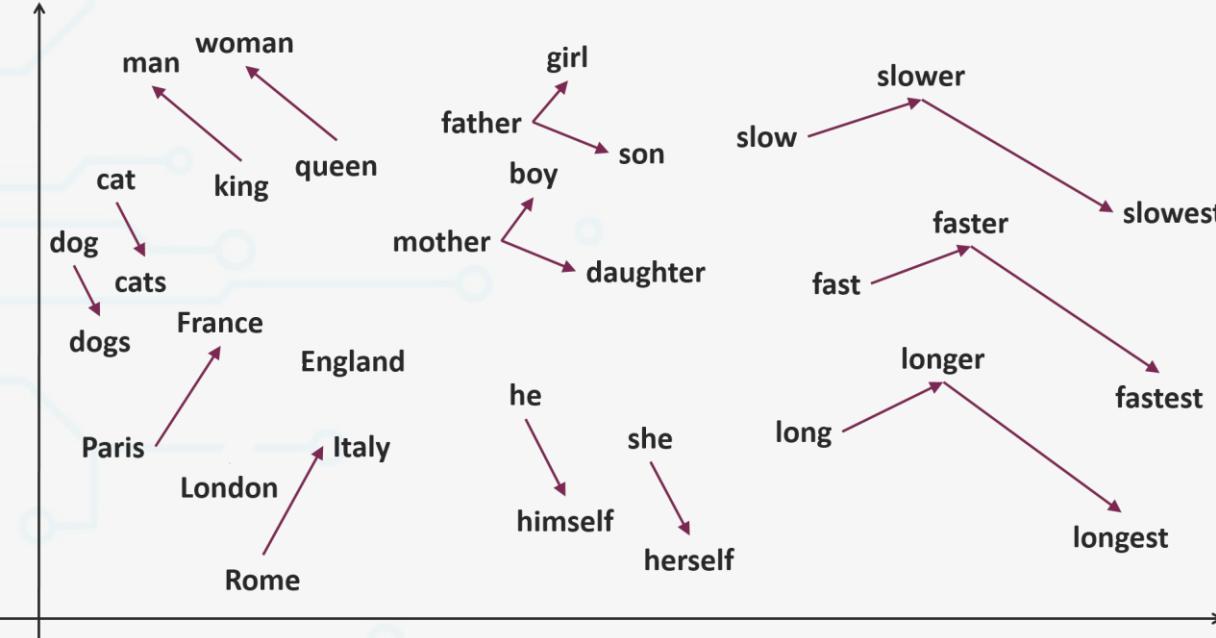
Representação densa

Palavras com significado similar
próximas (semântica)



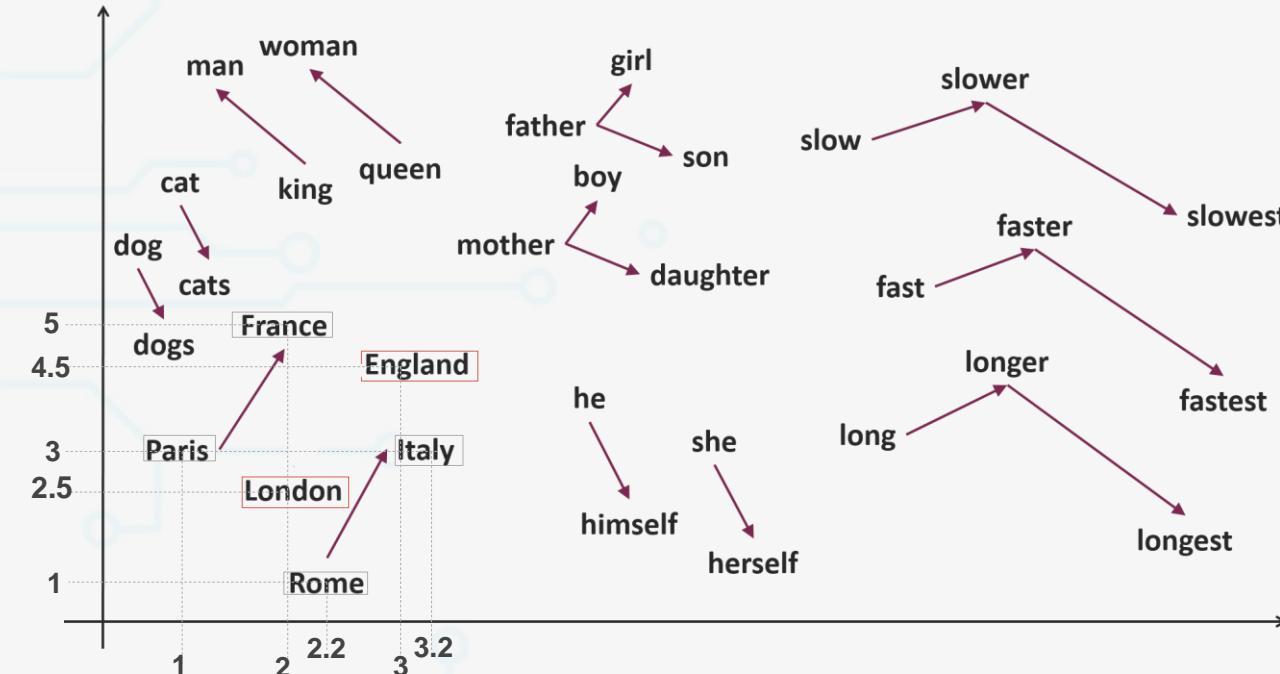
Processamento de Linguagem Natural

Natural Language Processing (NLP)



Processamento de Linguagem Natural

Natural Language Processing (NLP)

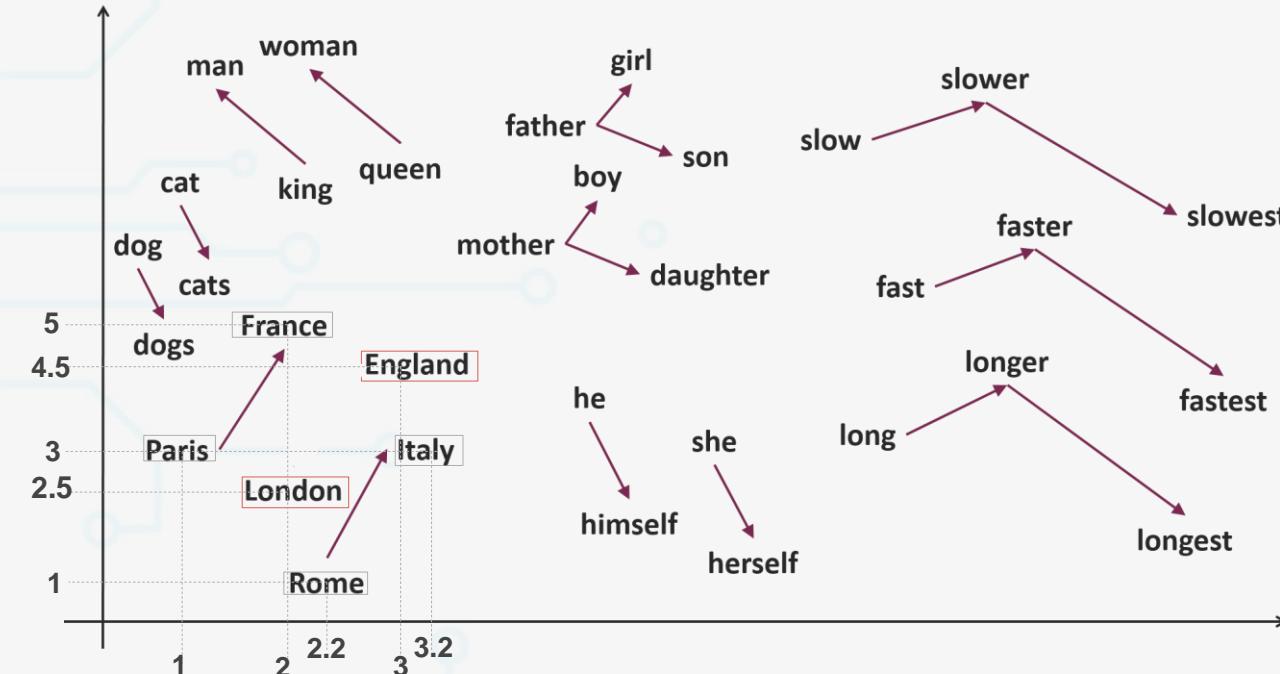


- Paris [1, 3]
- France [2, 5]
- London [2, 2.5]
- England [3, 4.5]
- Rome [2.2, 1]
- Italy [3.2, 3]

Processamento de Linguagem Natural

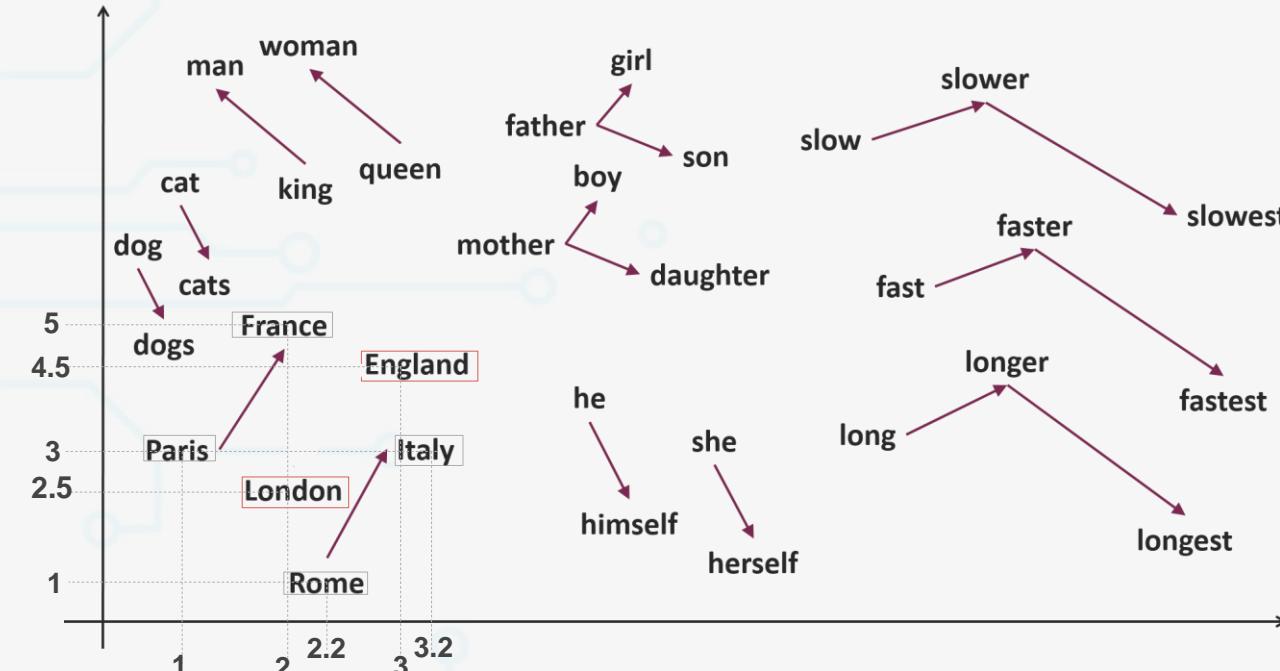
Natural Language Processing (NLP)

Qual a capital da Inglaterra?



Processamento de Linguagem Natural

Natural Language Processing (NLP)

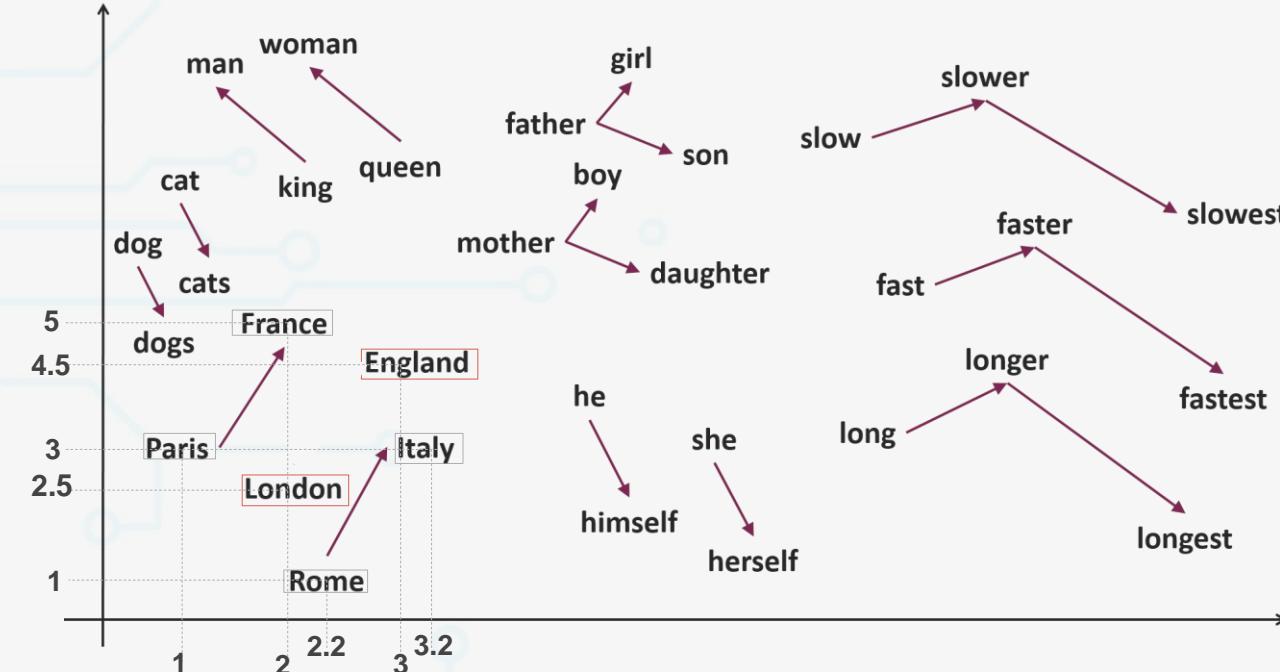


Qual a capital da Inglaterra?

Paris – France + England = ?

Processamento de Linguagem Natural

Natural Language Processing (NLP)



Qual a capital da Inglaterra?

Paris – France + England = ?

Paris [1, 3]

France [2, 5]

=

Result. [-1, -2]

+

England [3, 4.5]

=

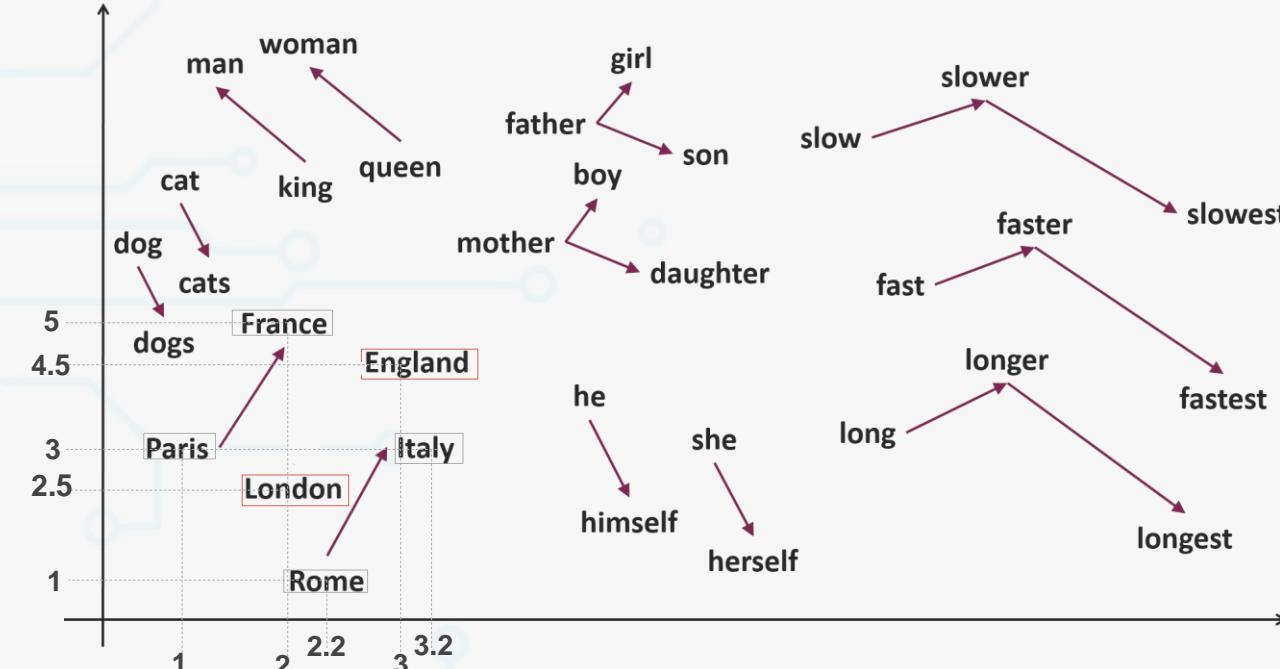
Result. [2, 2.5]

==

London [2, 2.5]

Processamento de Linguagem Natural

Natural Language Processing (NLP)



Qual a capital da Inglaterra?

Paris – France + England = ?

Paris [1, 3]

France [2, 5]

=

Result. [-1, -2]

+

England [3, 4.5]

=

Result. [2, 2.5]

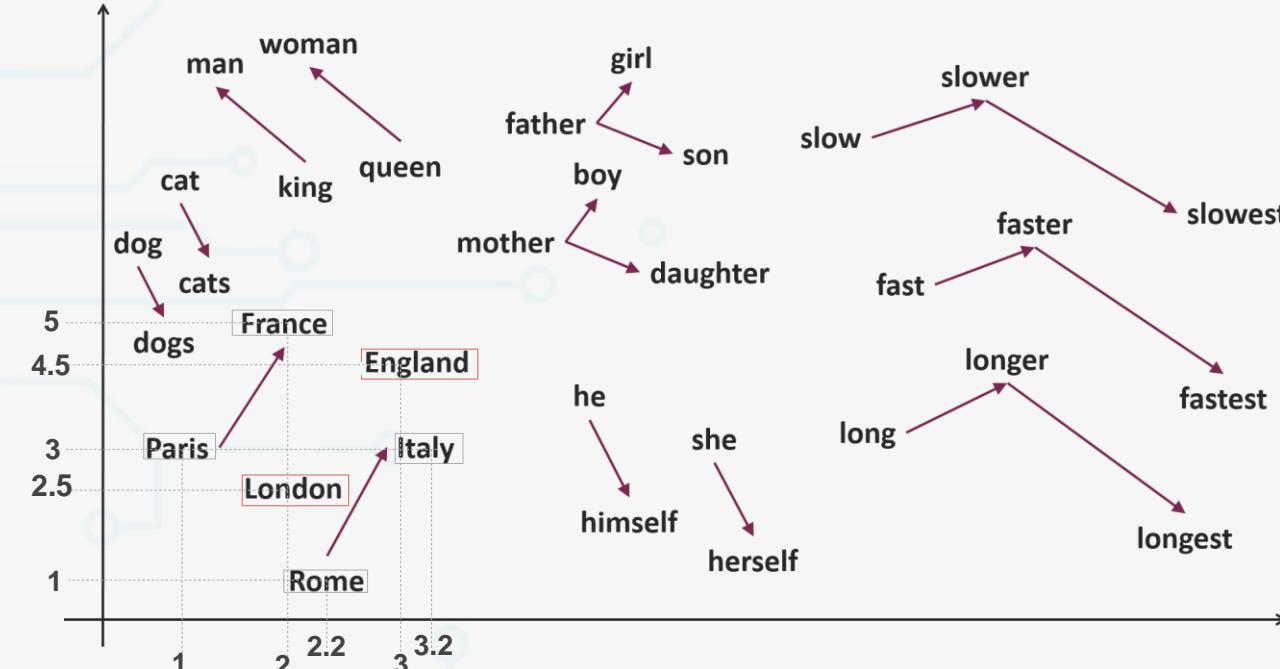
==

London [2, 2.5]

Paris – France + England = London

Processamento de Linguagem Natural

Natural Language Processing (NLP)



Qual a capital da Inglaterra?

Paris – France + England = ?
Paris [1, 3]
France [2, 5]
=
Result. [-1, -2]
+
England [3, 4.5]
=
Result. [2, 2.5]
==
London [2, 2.5]

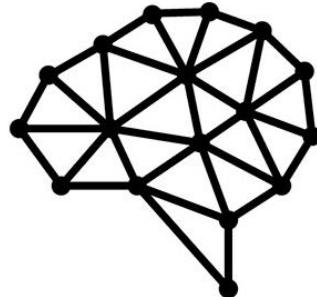
Rome – Italy + England = ?
Rome [2.2, 1]
Italy [3.2, 3]
=
Result. [-1, -2]
+
England [3, 4.5]
=
Result. [2, 2.5]
==
London [2, 2.5]

Paris – France + England = London

Rome – Italy + England = London

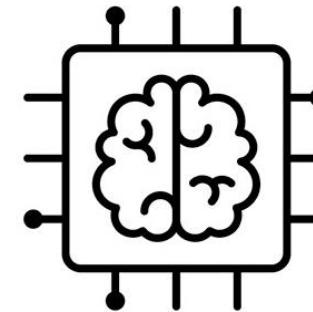
Processamento de Linguagem Natural

Natural Language Processing (NLP)

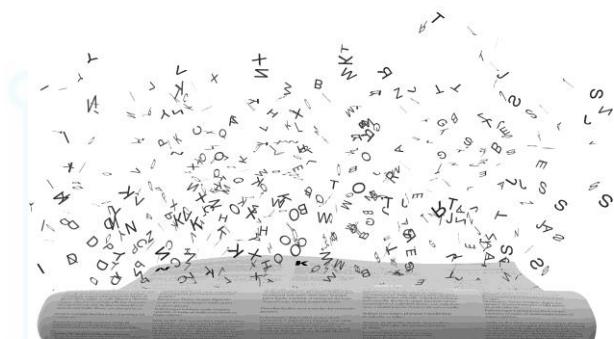


Representação
do Texto

Semi Supervisionado



Modelo de Machine Learning
Treinado para uma tarefa de NLP



Auto supervisionado (SSL)

Examples

Good price! Quality not bad! I'm happy I bought it.

Bad quality! I'm sad! I bought it I will return it.

Price not good. Quality bad! I'm not happy I bought it.

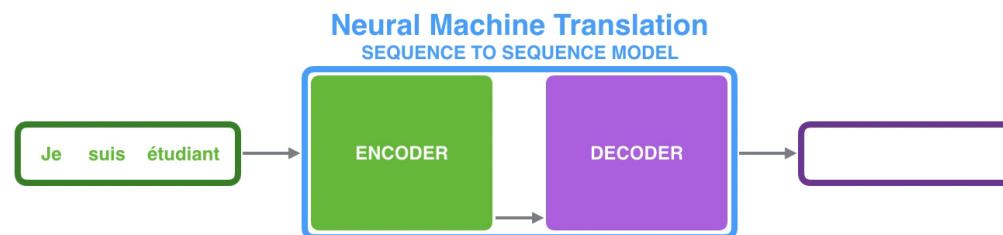
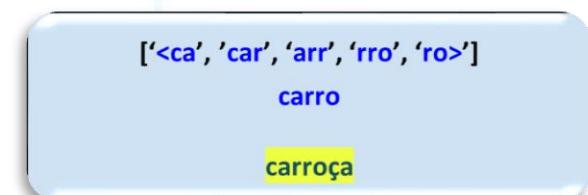
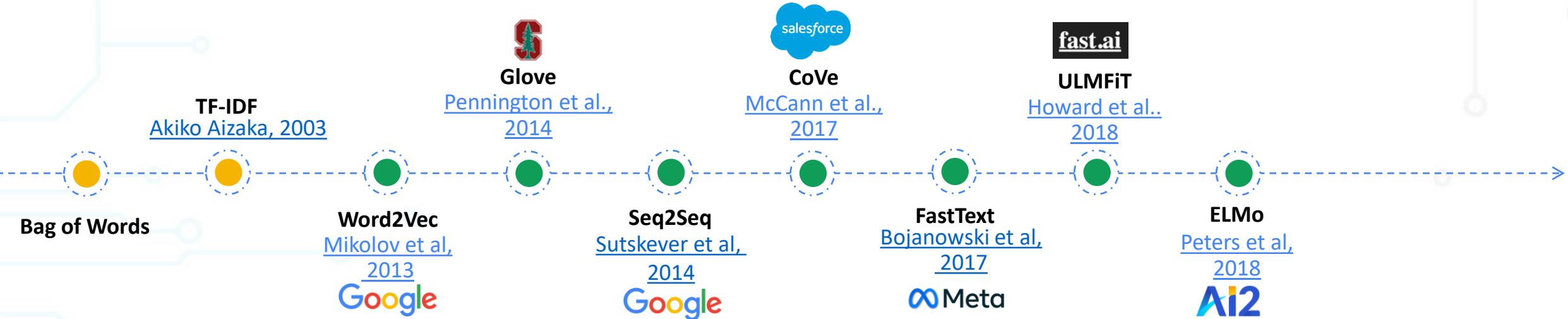
Labels



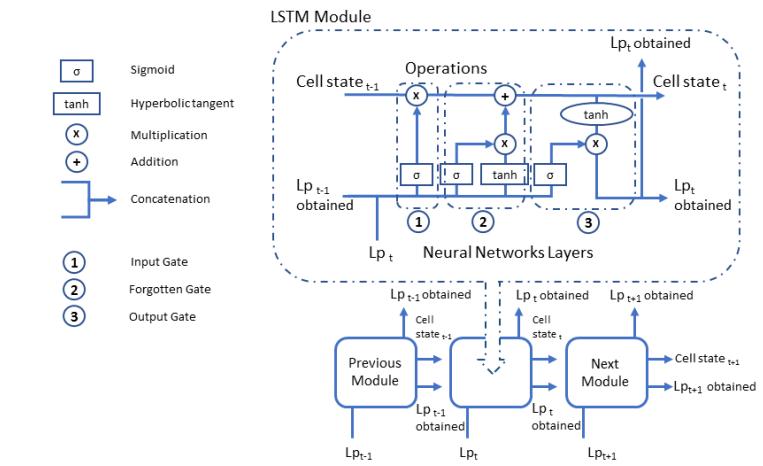
Supervisionado

Processamento de Linguagem Natural

Natural Language Processing (NLP)



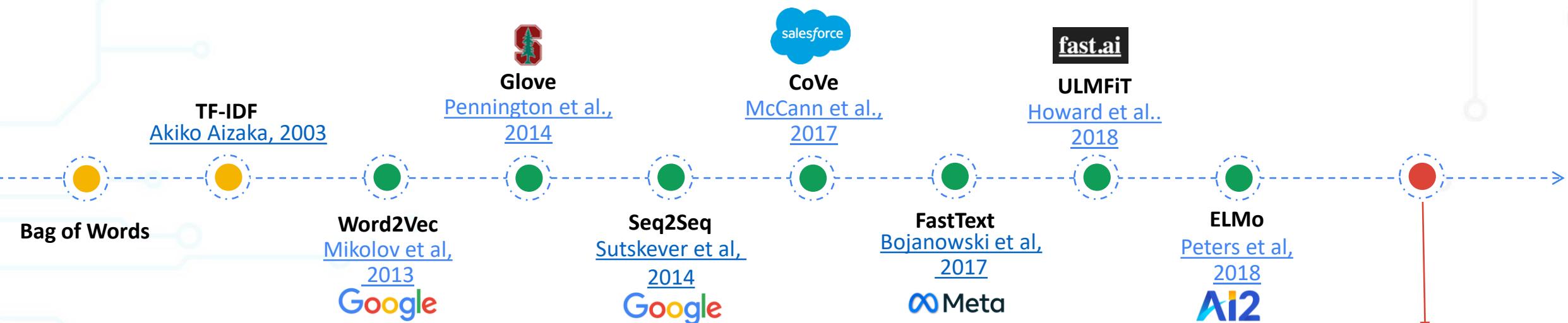
sent₁: **banco da praça**
sent₂: **app do banco**



Long Short-Term Memory Networks (LSTM)
<https://arxiv.org/pdf/1909.09586.pdf>

Processamento de Linguagem Natural

Natural Language Processing (NLP)



Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Lukasz Kaiser*
Google Brain
lukaszkaiser@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Google

<https://arxiv.org/abs/1706.03762>

Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

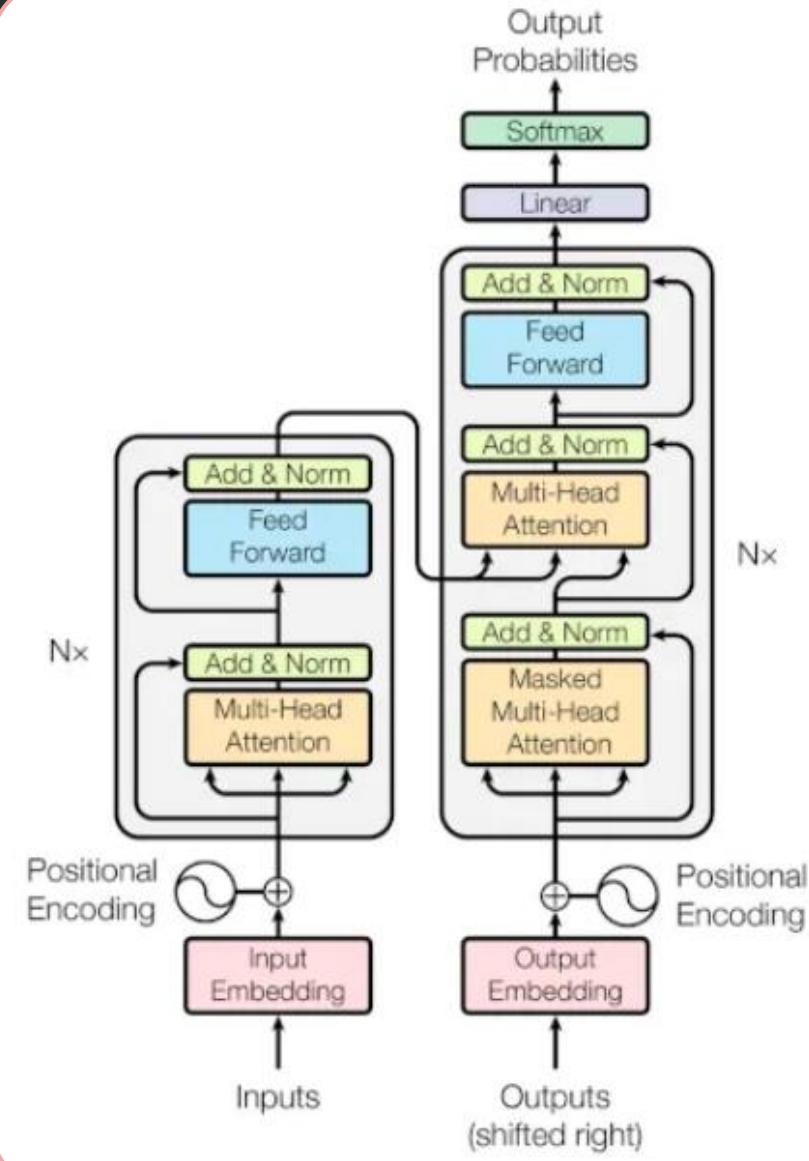
Lukasz Kaiser*
Google Brain
lukasz.kaiser@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

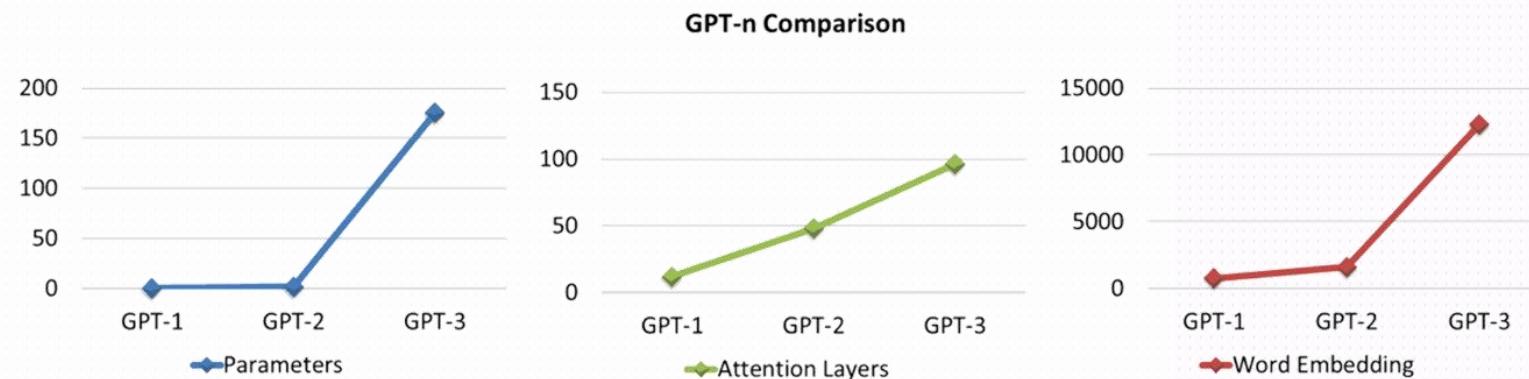
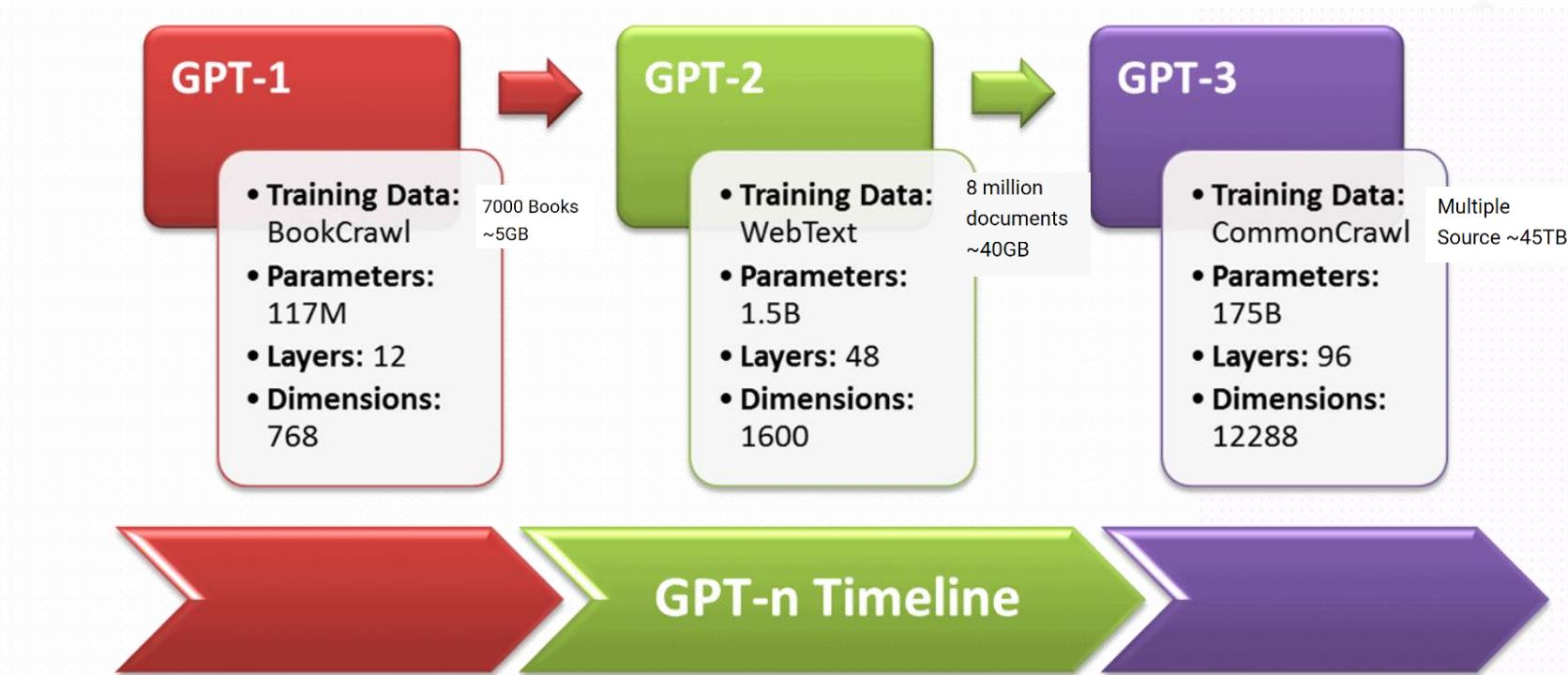
Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single model state-of-the-art BLEU score of 41.9 after

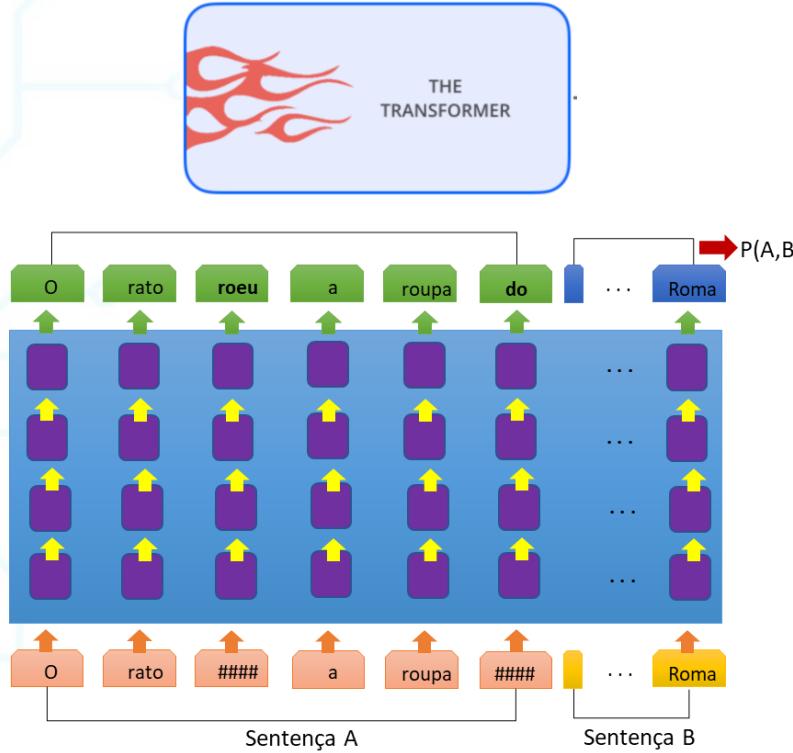
<https://arxiv.org/abs/1706.03762>



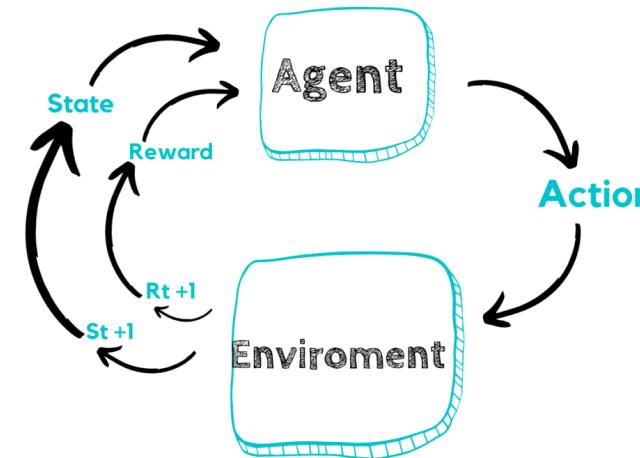
Família GPT



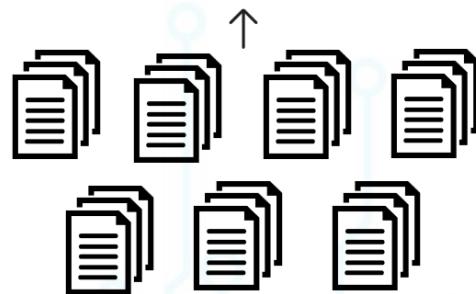
ChatGPT



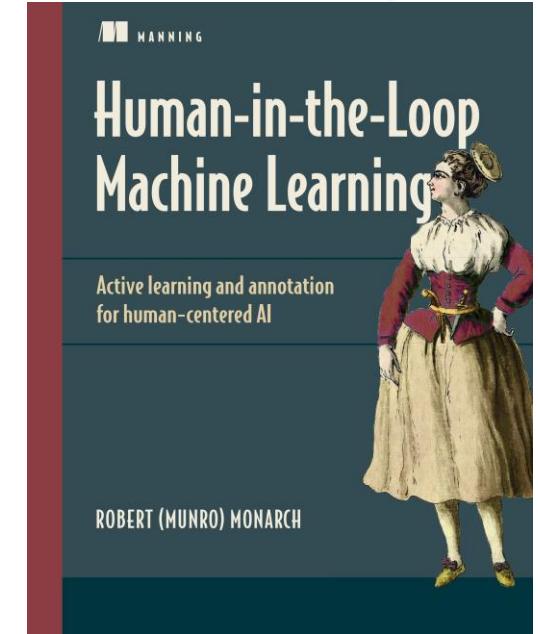
+



Reinforcement Learning



Reinforcement Learning from Human Feedback (RLHF)



Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

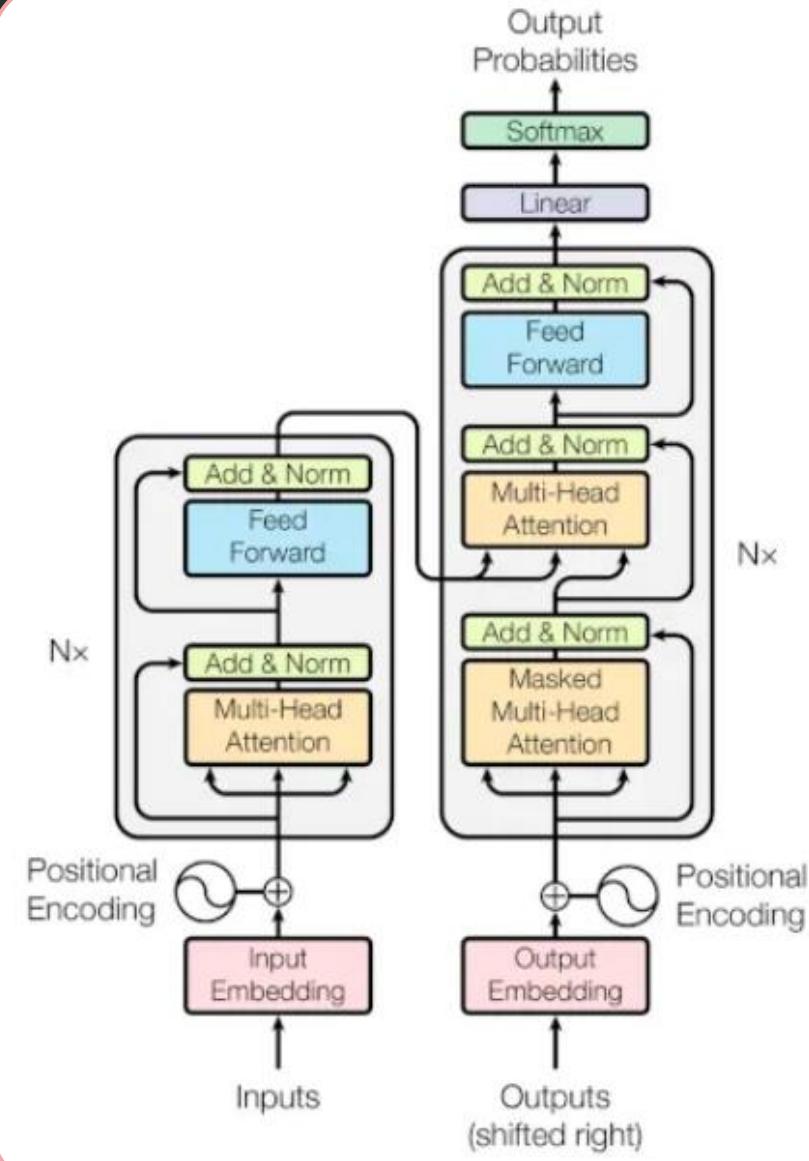
Lukasz Kaiser*
Google Brain
lukasz.kaiser@google.com

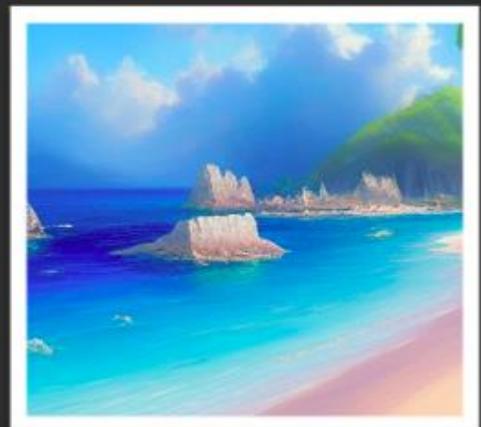
Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Abstract

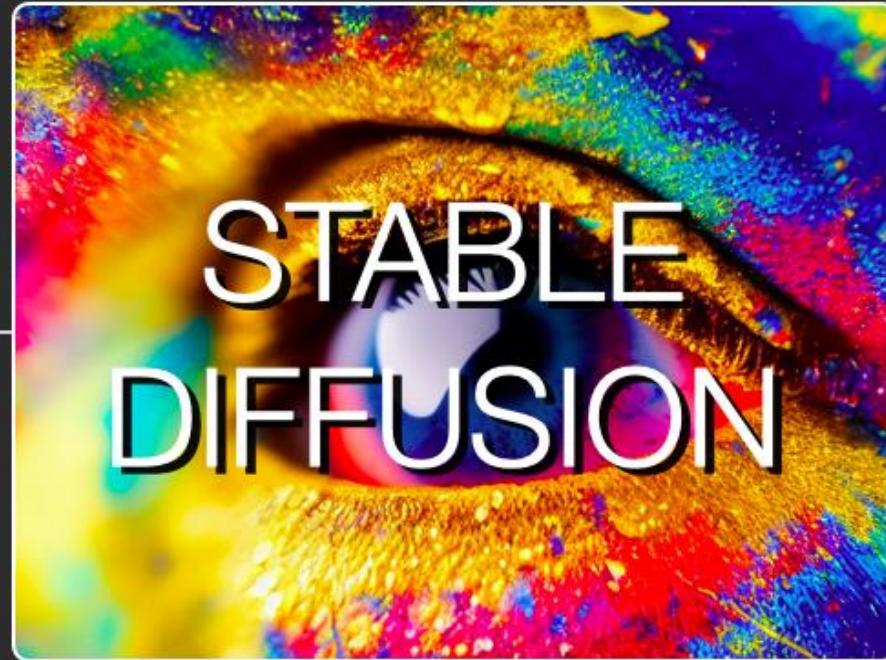
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single model state-of-the-art BLEU score of 41.9 after

<https://arxiv.org/abs/1706.03762>





Attention Is All You Need



<https://arxiv.org/abs/1706.03762>

Inputs



Outputs
(shifted right)



Output
Probabilities

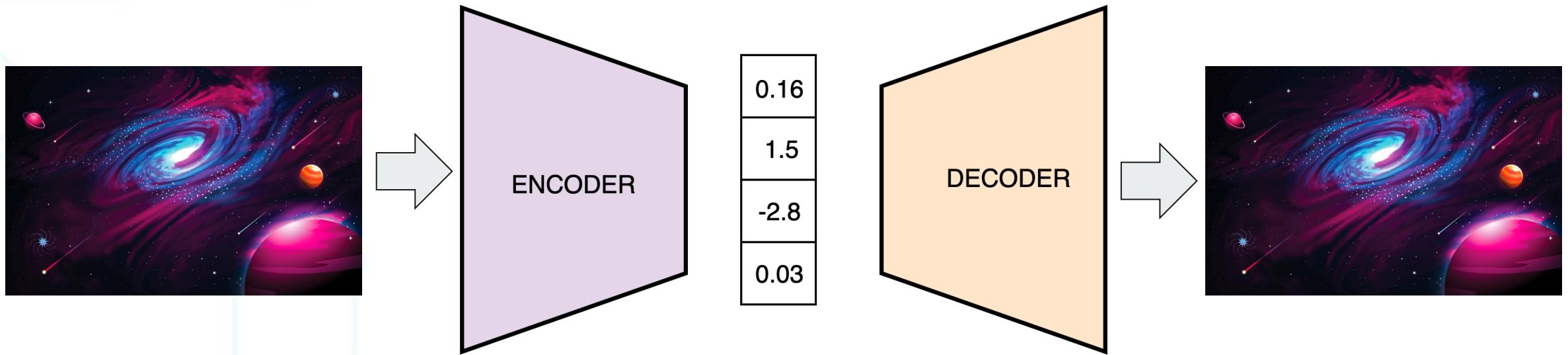


Stable Diffusion

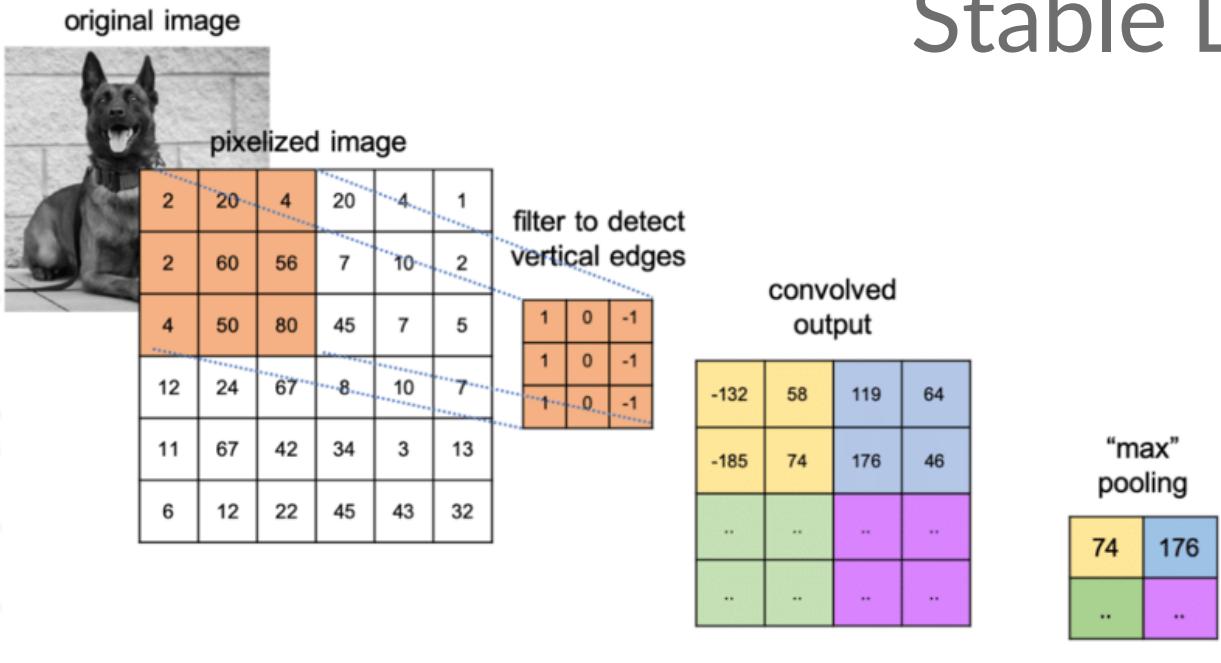
- NÃO É UM MODELO! É um sistema composto por várias redes neurais.
- Permite a geração de imagens condicionada a um texto.
- Foi o primeiro a ser disponibilizado de forma totalmente open-source e com requisitos computacionais relativamente baixos, pois realiza a maioria das operações no espaço latente.

Stable Diffusion

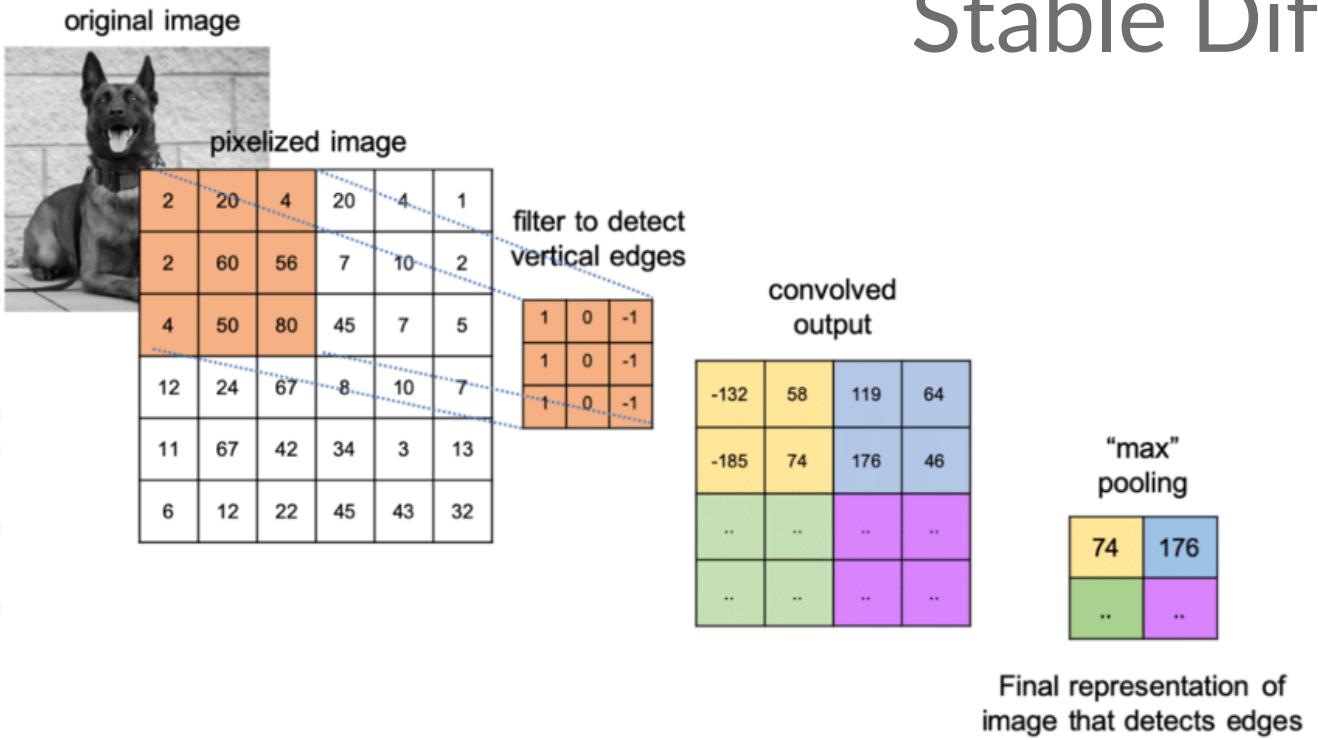
Espaço latente



Stable Diffusion

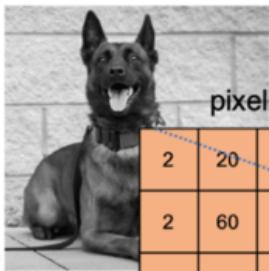


Stable Diffusion



Stable Diffusion

original image



pixelized image

2	20	4	20	4	1
2	60	56	7	10	2
4	50	80	45	7	5
12	24	67	8	10	7
11	67	42	34	3	13
6	12	22	45	43	32

filter to detect vertical edges

1	0	-1
1	0	-1
1	0	-1

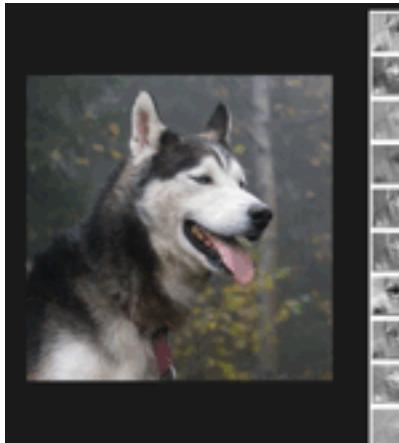
convolved output

-132	58	119	64
-185	74	176	46
..
..

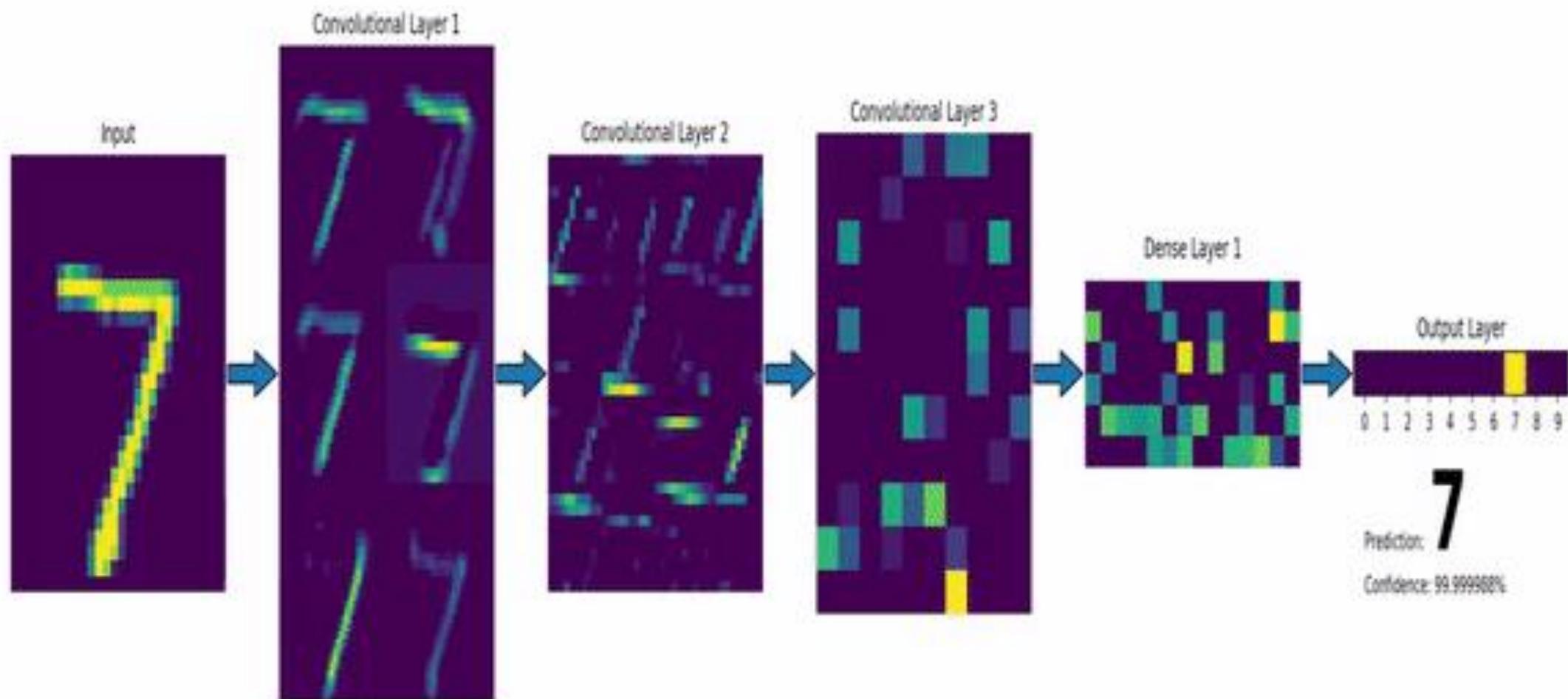
"max" pooling

74	176
..	..

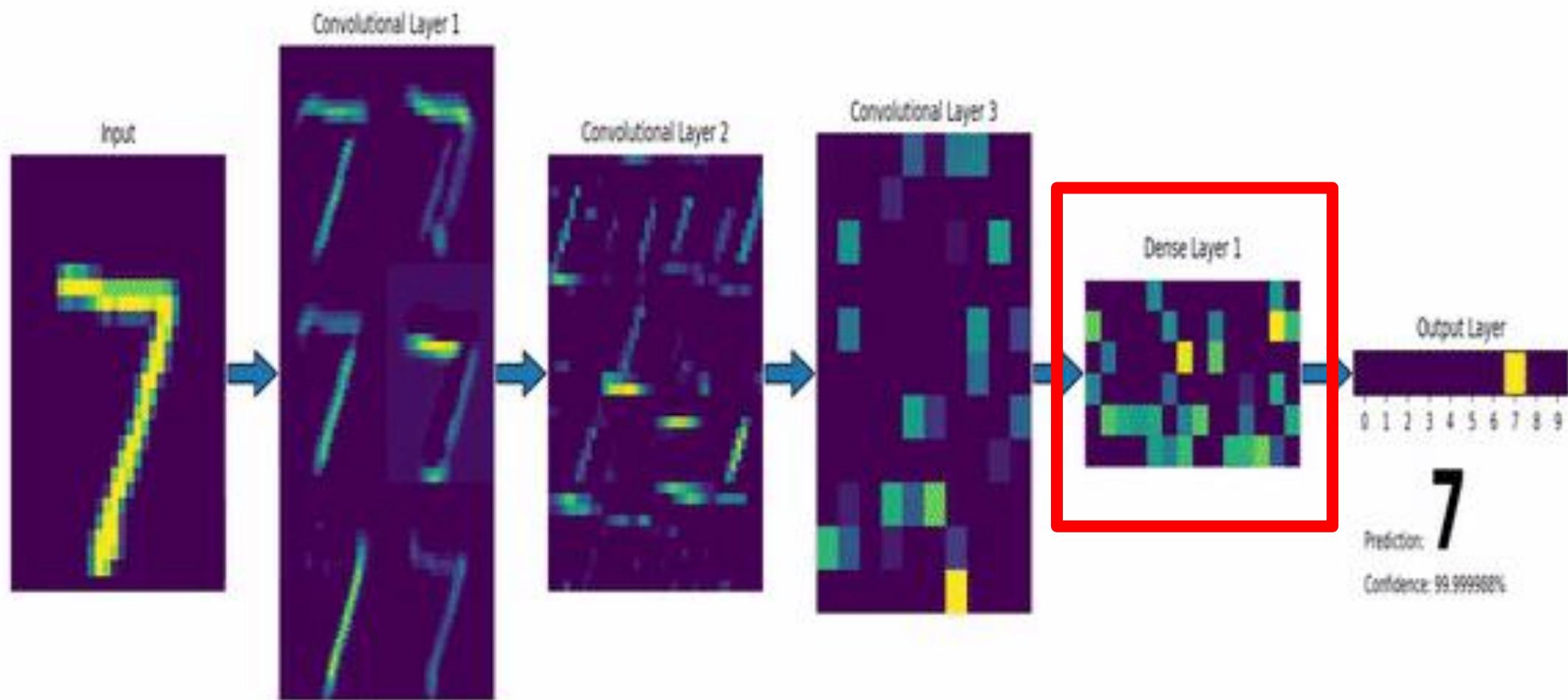
Final representation of image that detects edges



Stable Diffusion

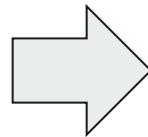


Stable Diffusion



Stable Diffusion

Carregando uma imagem na GPU

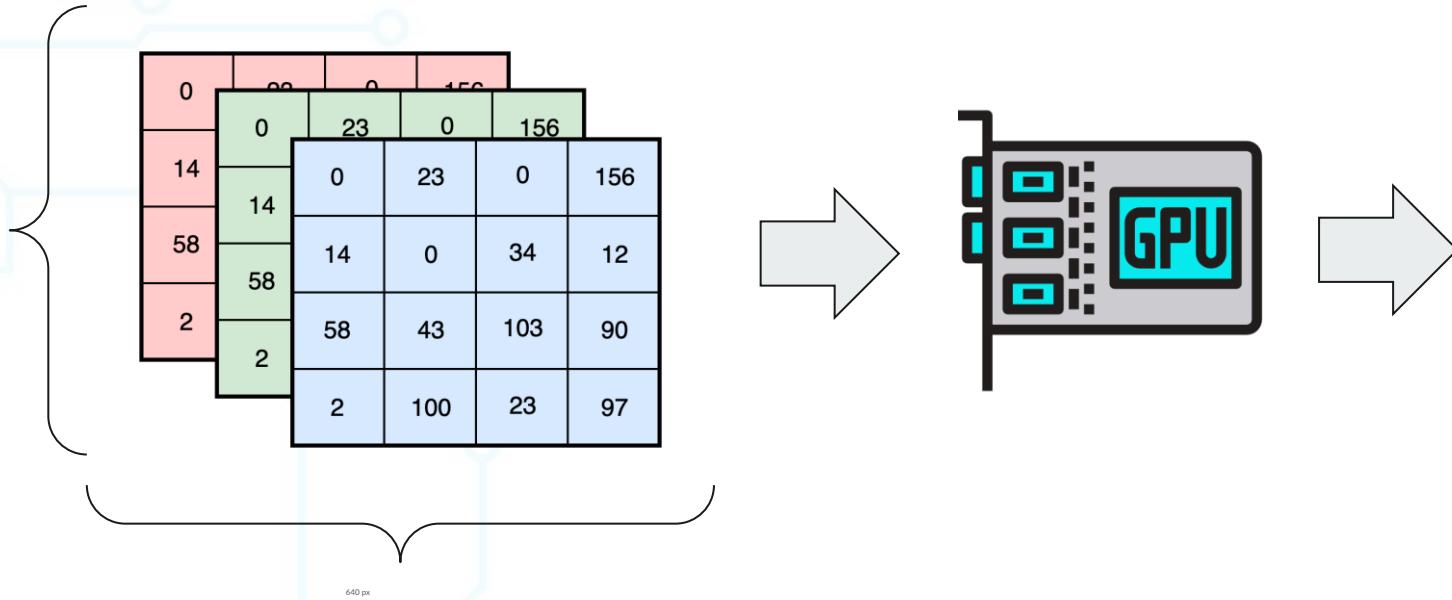


0	0	0	156
14	0	23	156
58	14	0	156
2	58	34	12
		58	90
		2	97
		100	23

0-255 = 8 bits
(int8)

Stable Diffusion

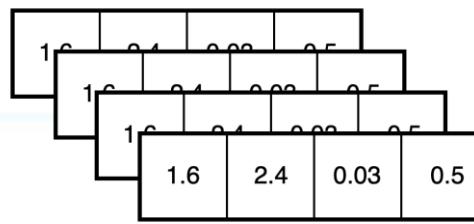
Carregando uma imagem na GPU



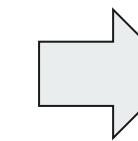
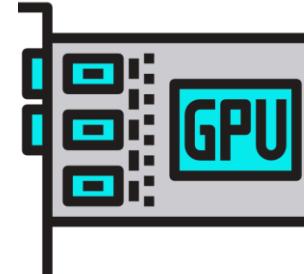
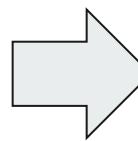
$$640 * 640 * 3 (\text{RGB}) * 1 \text{ byte} = 1,22 \text{ MB}$$

Stable Diffusion

Carregando uma imagem na GPU



$4 * 1 * 4$ (canais)



$4 * 1 * 4$ (canais) * 4 bytes (float32)= 64 B

Stable Diffusion



Modelos de difusão

São modelos probabilísticos criados para aprender uma distribuição $p(x)$ através da remoção gradual de ruído de uma variável normalmente distribuída, o que corresponde a aprender o processo inverso de uma cadeia de Markov fixa de tamanho T.

Modelos de difusão

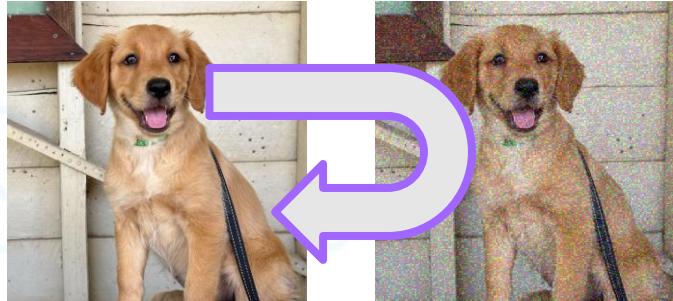
Diffusion model



Adds noise and learns how to work backwards to the original image.

Modelos de difusão

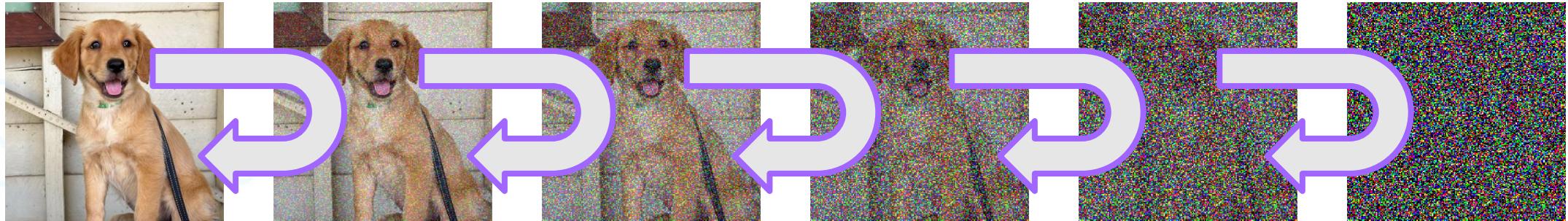
Diffusion model



Adds noise and learns how to work backwards to the original image.

Modelos de difusão

Diffusion model



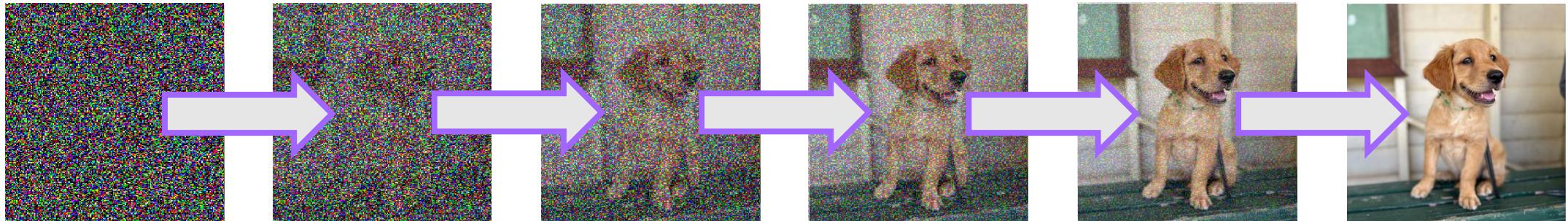
Adds noise and learns how to work backwards to the original image.

Modelos de difusão

Diffusion model

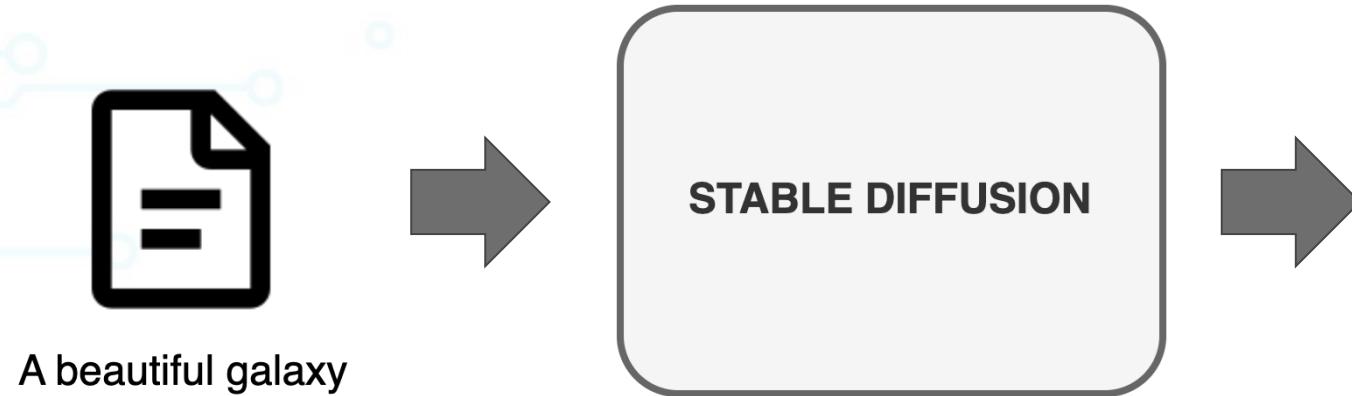


Adds noise and learns how to work backwards to the original image.

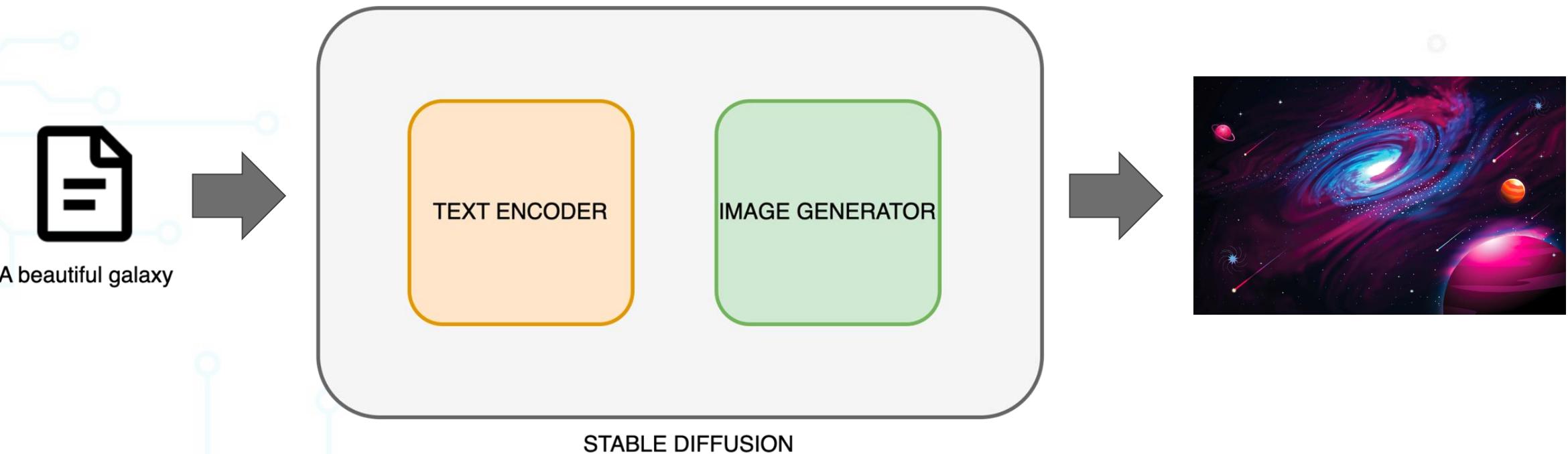


Trained model works from random noise to generate an image.

Stable Diffusion



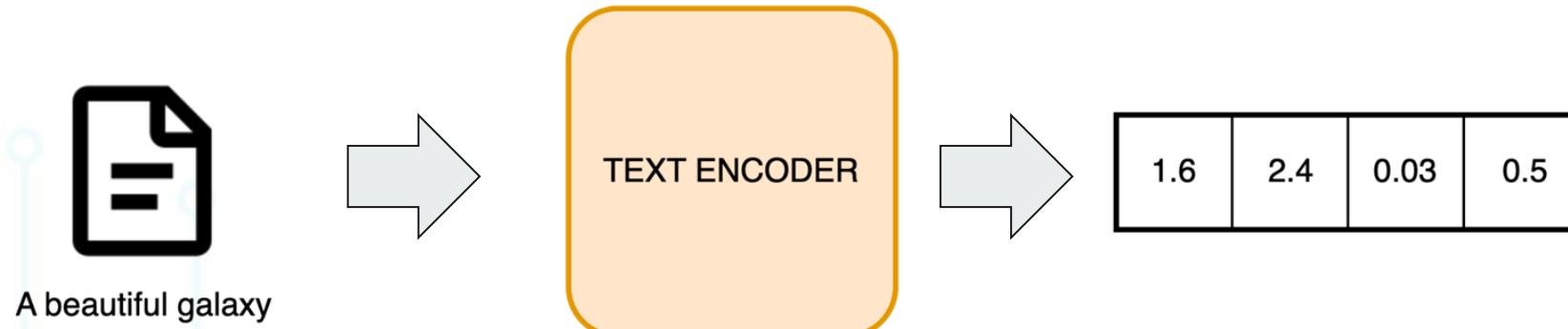
Stable Diffusion



Stable Diffusion

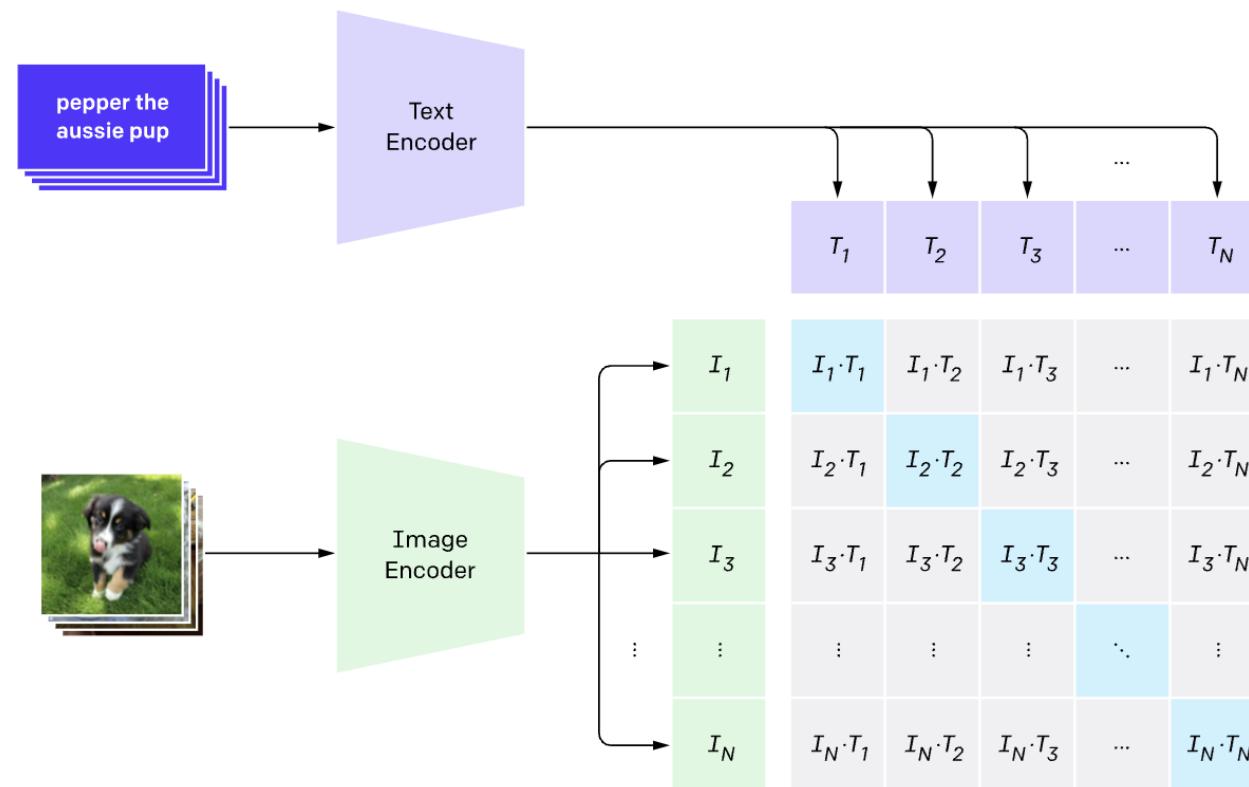
Text Encoder - CLIP

- Modelo treinado com o objetivo de gerar o mesmo embedding para uma imagem e sua descrição.
- Modelo possui um encoder de imagem e um encoder de texto que podem ser usados.
- No Stable Diffusion é usado o encoder de texto.



Stable Diffusion

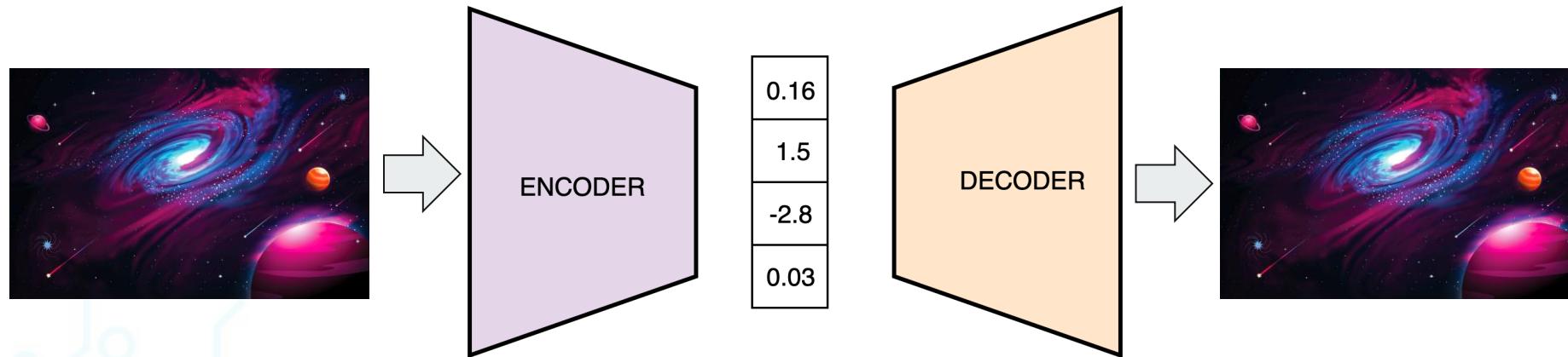
Text Encoder - CLIP



Stable Diffusion

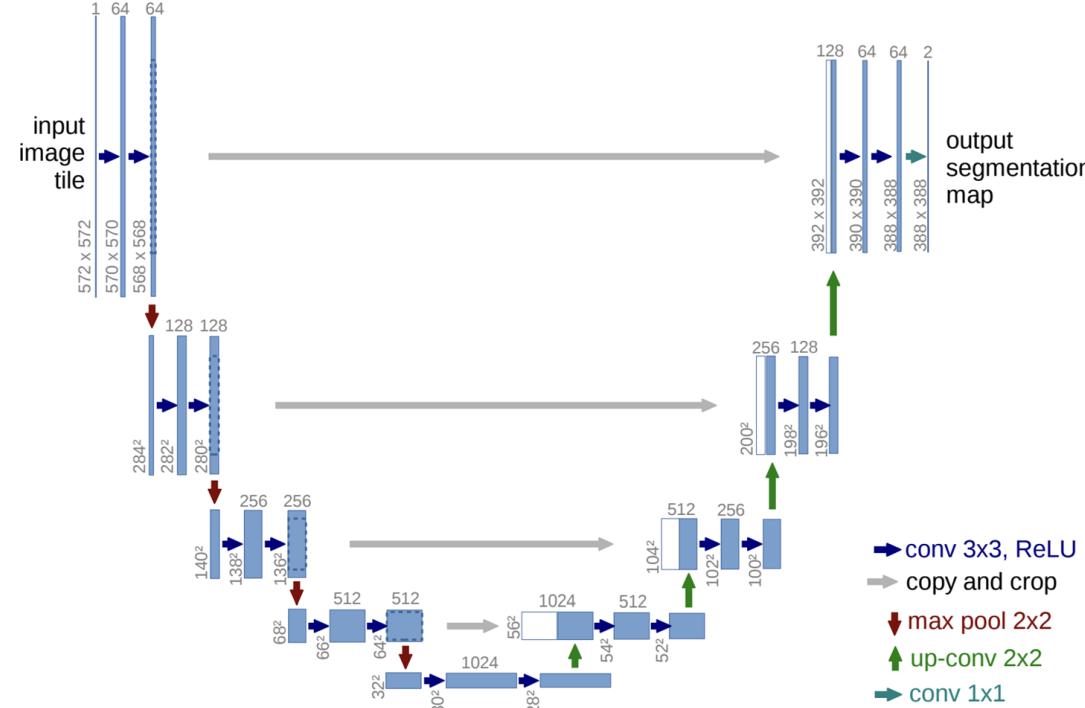
Autoencoder - VAE

- Transforma a imagem em um tensor de (4, 64, 64) que é usado no processo de *denoising*.
- No treinamento usamos o encoder e o decoder.
- Na predição usamos somente o decoder.



Stable Diffusion

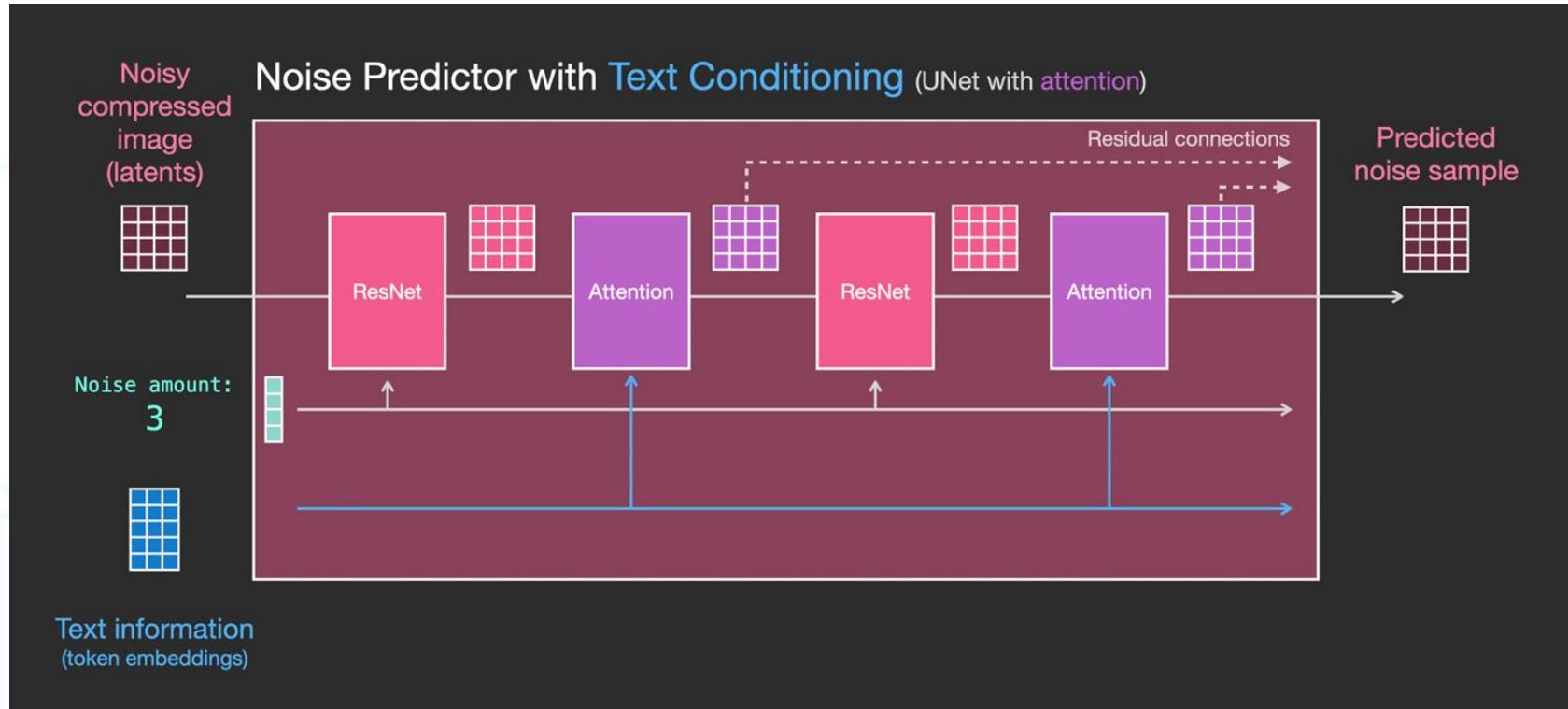
Predictor de ruído - UNet Condicionada



Olaf Ronneberger, Philipp Fischer, and Thomas Brox. [U-Net: Convolutional Networks for Biomedical Image Segmentation](#)

Stable Diffusion

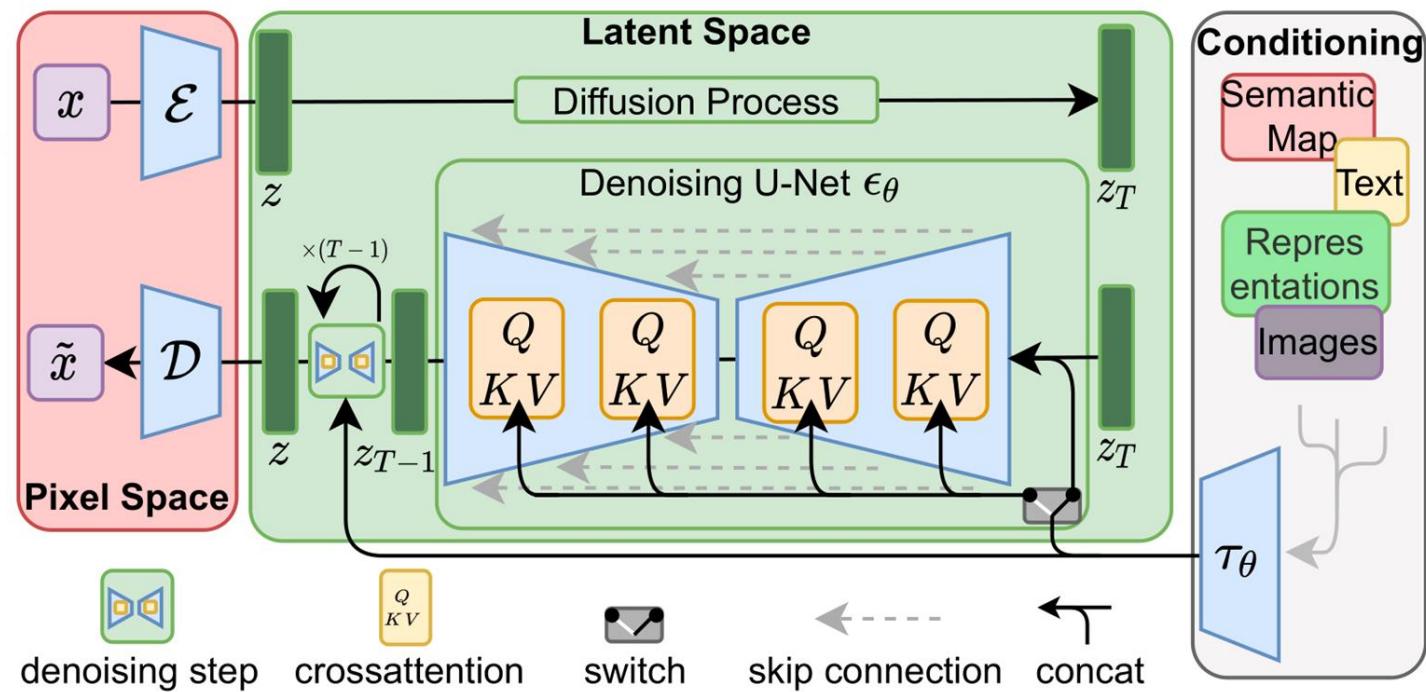
Predictor de ruído - UNet Condicionada



[The Illustrated Stable Diffusion](#)

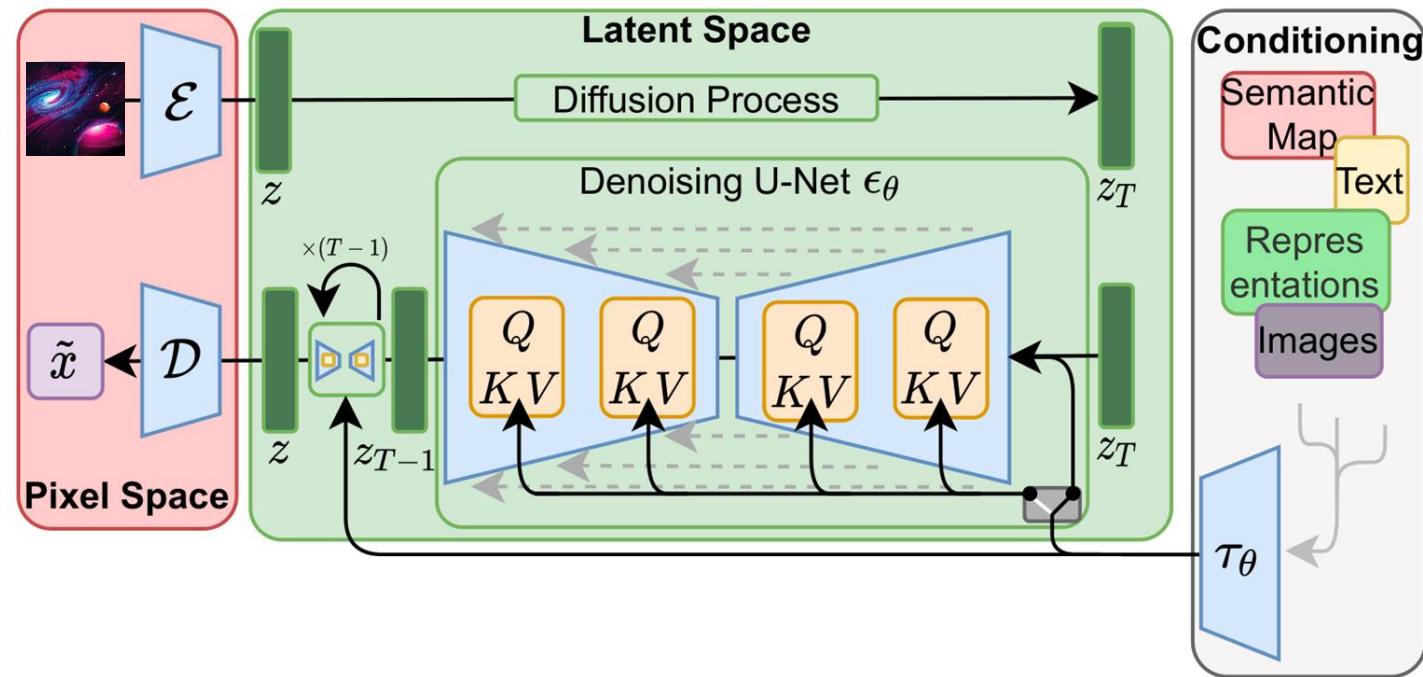
Stable Diffusion

Treinamento



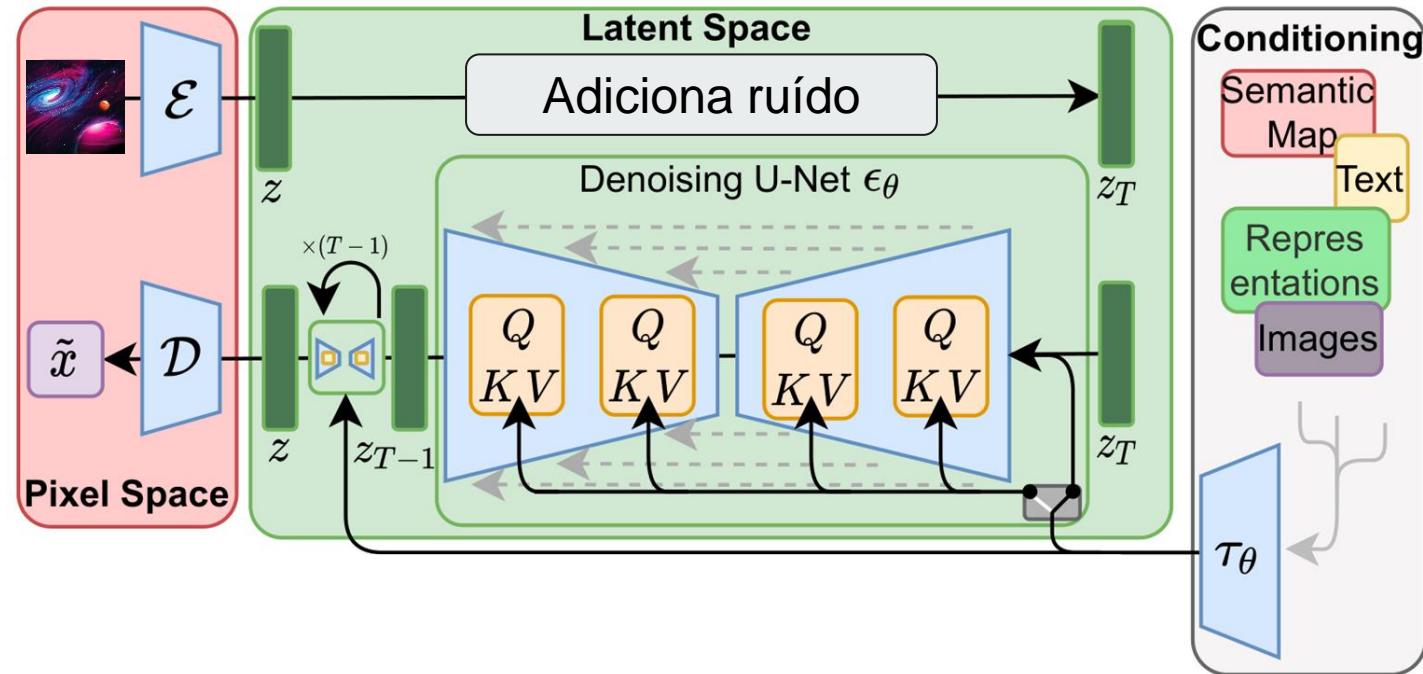
Stable Diffusion

Treinamento



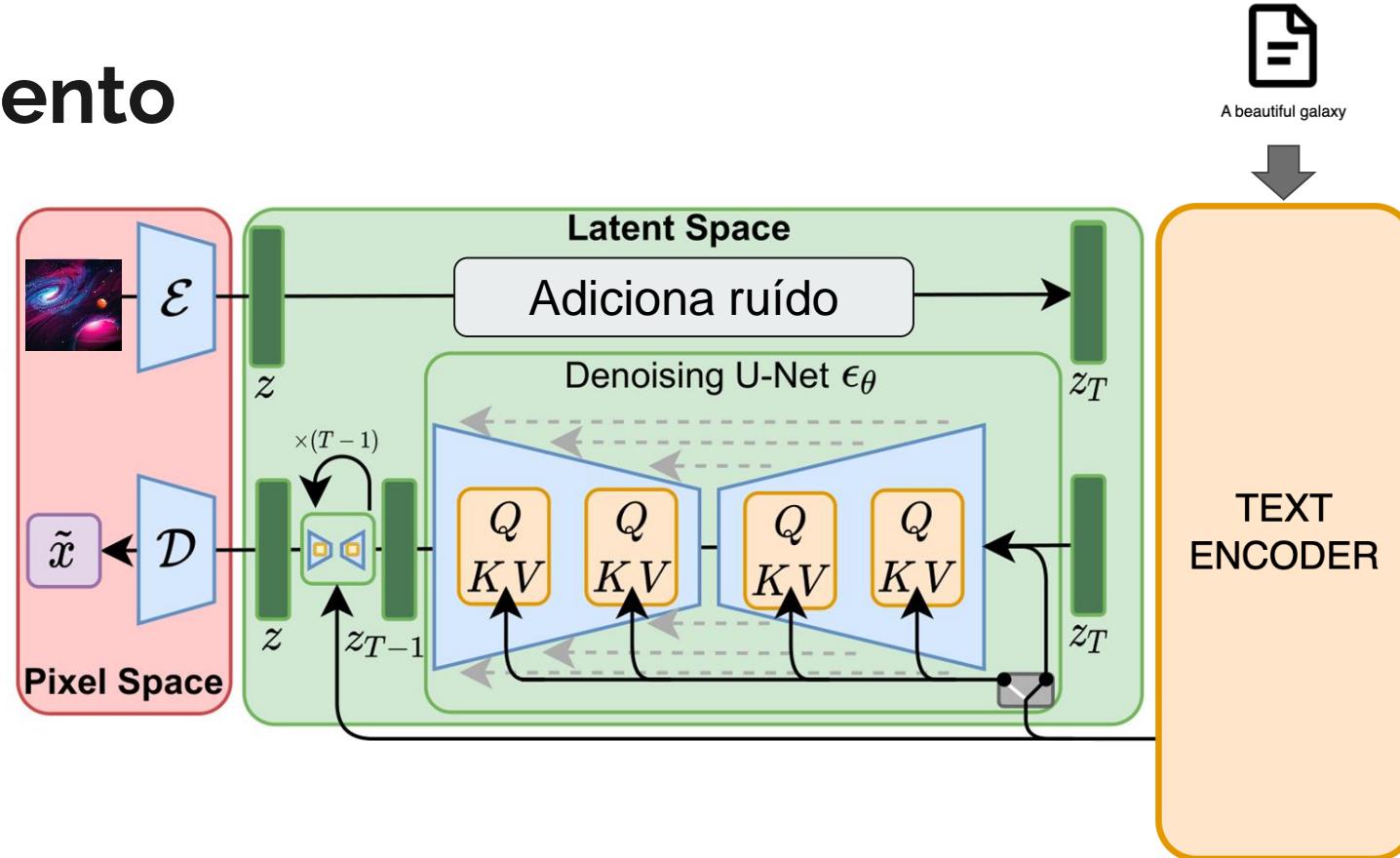
Stable Diffusion

Treinamento



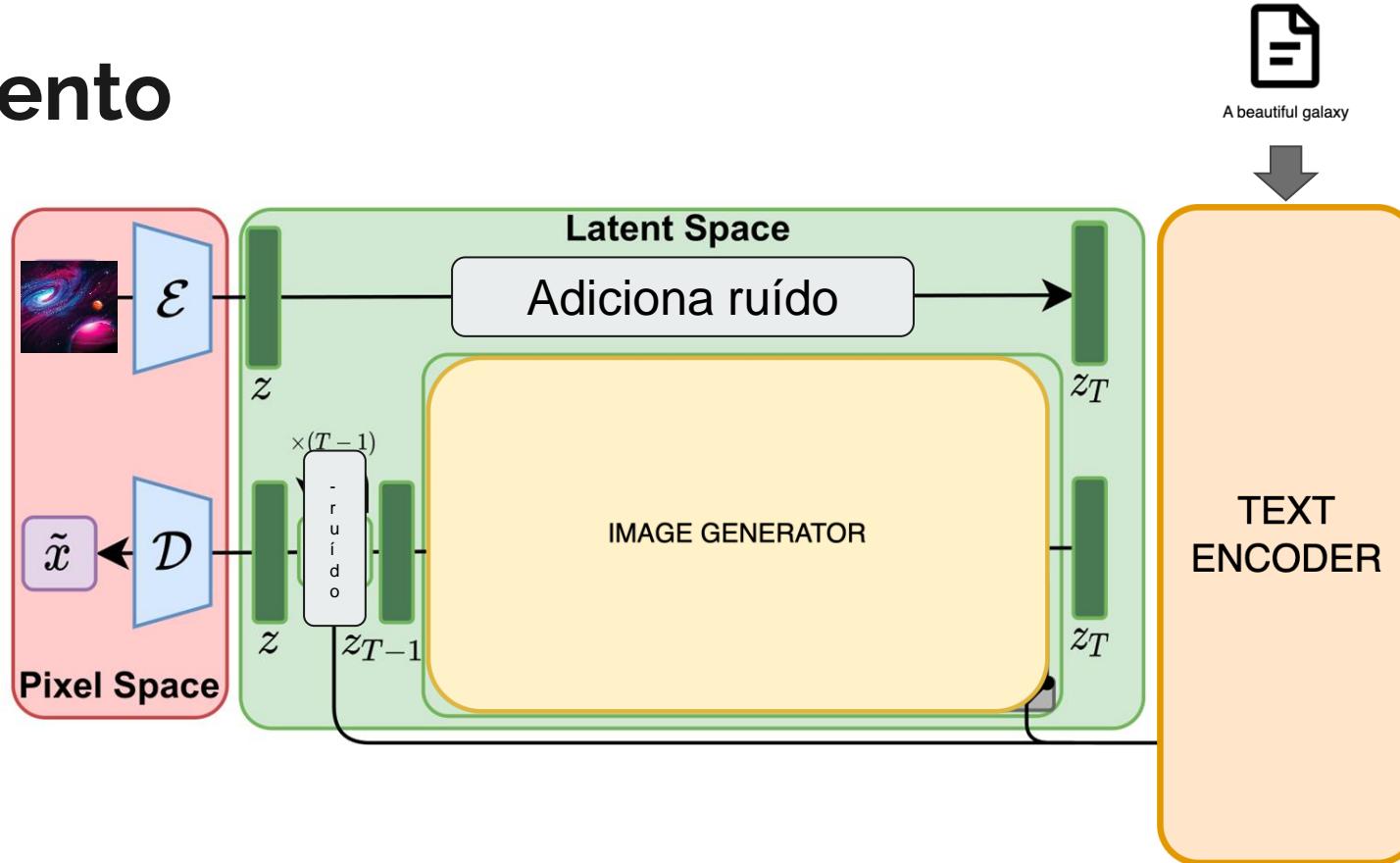
Stable Diffusion

Treinamento



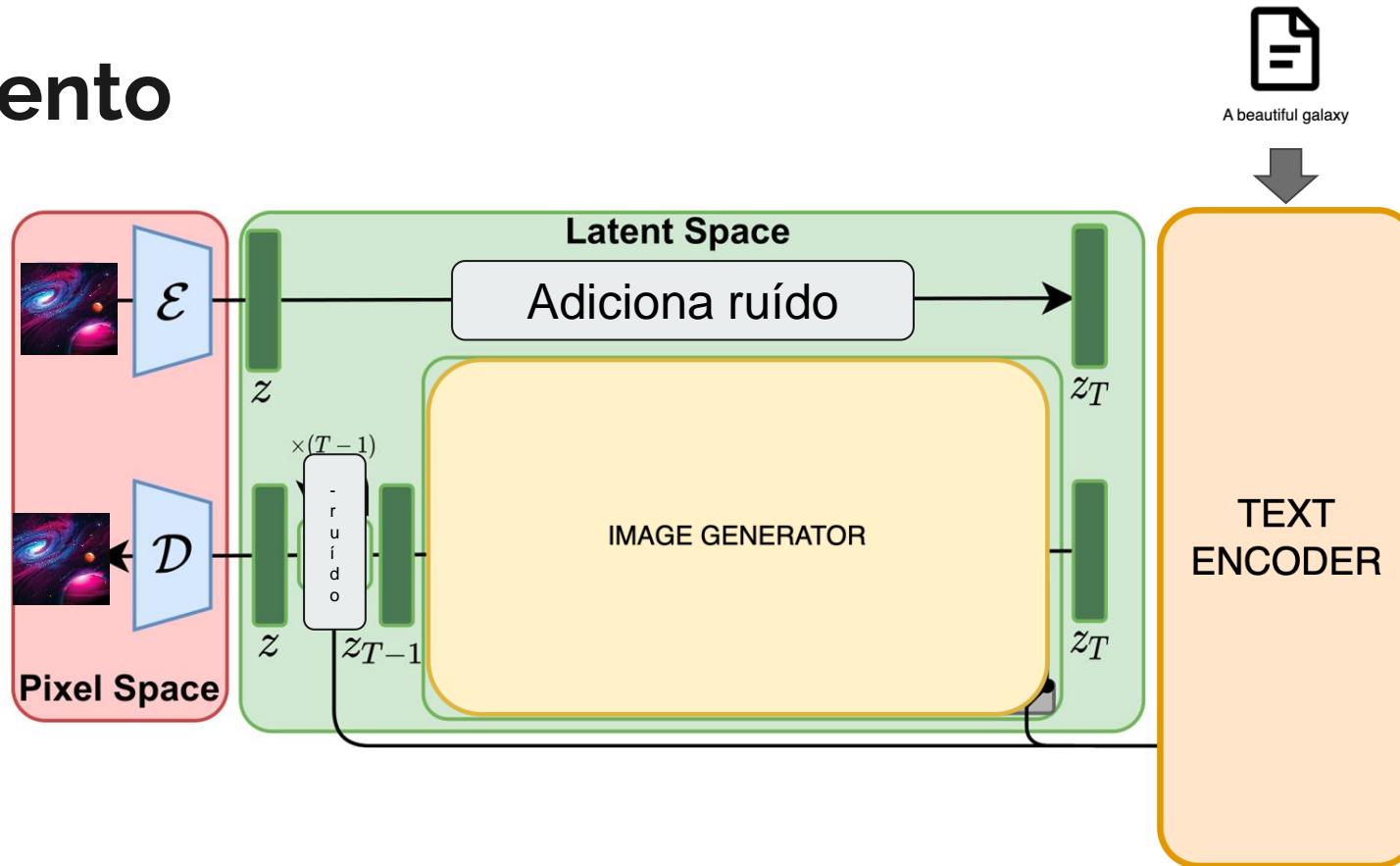
Stable Diffusion

Treinamento



Stable Diffusion

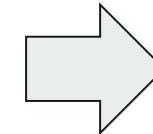
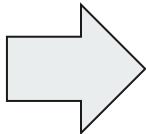
Treinamento



Textual Inversion

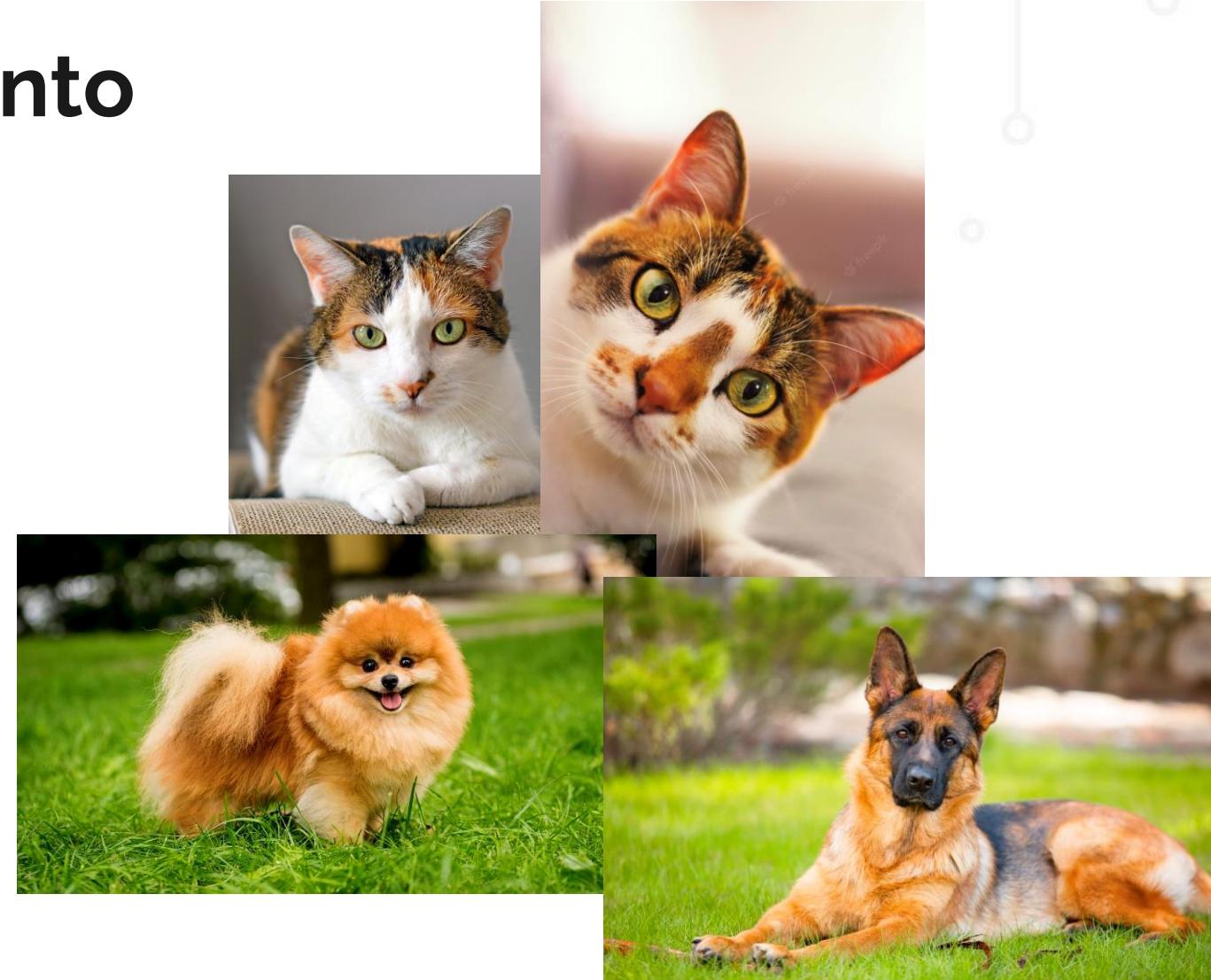


A beautiful galaxy



Textual Inversion

Dados fora do treinamento



Textual Inversion

Dados fora do treinamento

- Um gato em uma praça ✓



Textual Inversion

Dados fora do treinamento

- Um gato em uma praça ✓
- Um cachorro na grama ✓



Textual Inversion

Dados fora do treinamento

- Um gato em uma praça ✓
- Um cachorro na grama ✓
- Um coelho no mato ✗



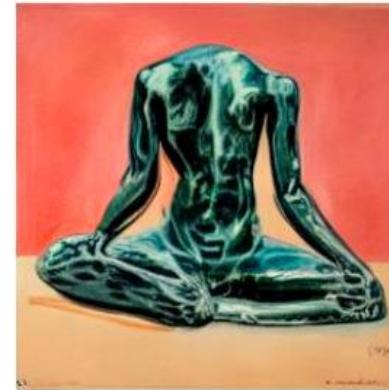
Textual Inversion

A inversão textual é uma técnica para capturar novos conceitos a partir de imagens de exemplo, que no geral é limitado. Os conceitos aprendidos podem ser usados para controlar melhor as imagens geradas a partir de pipelines de texto para imagem. De maneira simplista, o objetivo é aprender um novo token no espaço de embeddings do text encoder, os quais são usados para geração de imagem personalizada.

Textual Inversion



Input samples $\xrightarrow{\text{invert}}$ “ S_* ”



“An oil painting of S_* ”



“App icon of S_* ”



“Elmo sitting in the same pose as S_* ”



“Crochet S_* ”



Input samples $\xrightarrow{\text{invert}}$ “ S_* ”



“Painting of two S_* fishing on a boat”



“A S_* backpack”

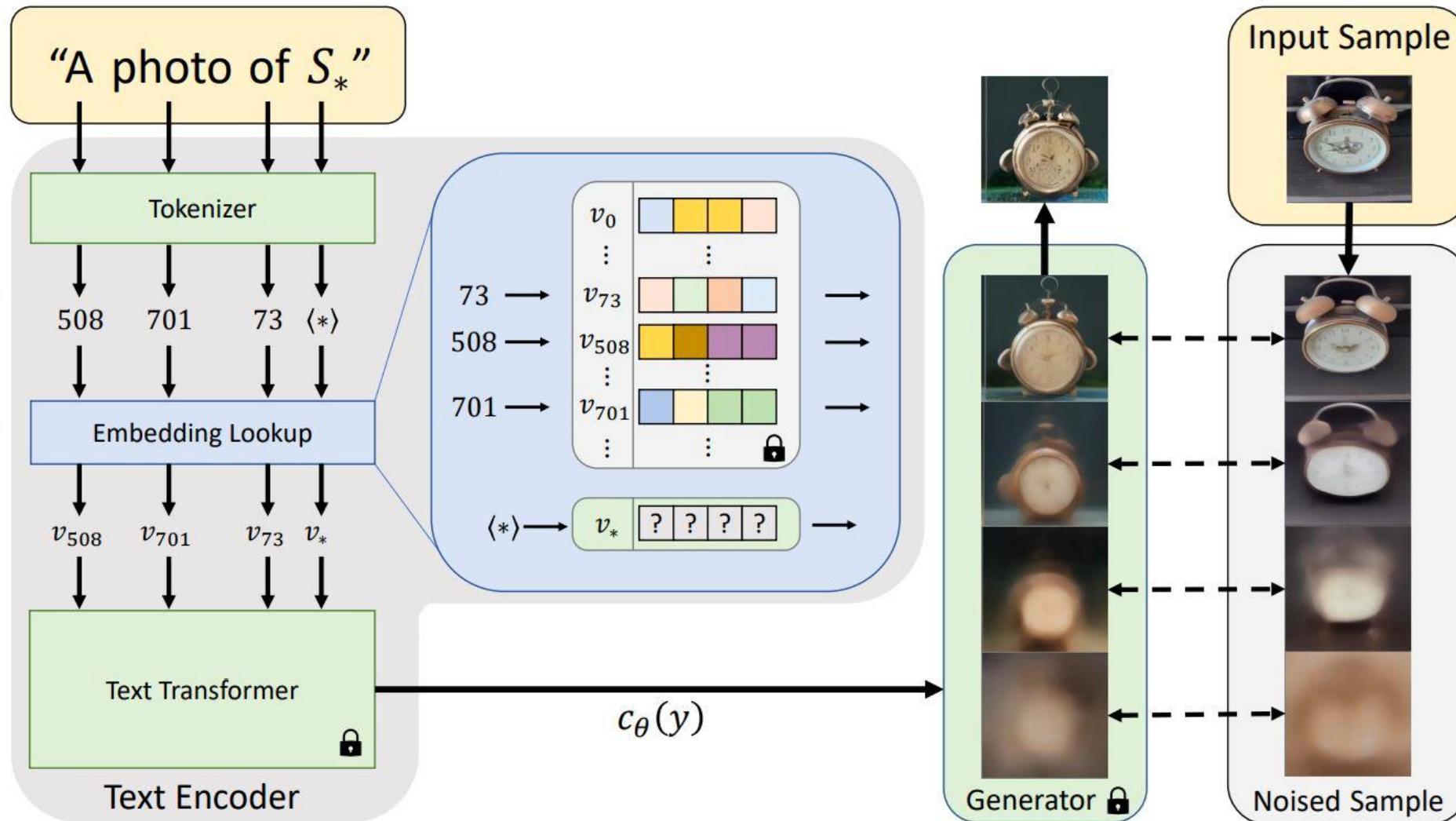


“Banksy art of S_* ”



“A S_* themed lunchbox”

Textual Inversion



Implementação de Exemplo

Muito Obrigado!



Preencha a pesquisa de satisfação no aplicativo móvel

Vinicius Caridá

@vfcarida

Vinicius Caridá



@vinicius_caridá



@vfcarida



@vinicius_caridá



@vfcarida



@vinicius_caridá



@vfcarida