Perceptron

Input: $\mathbf{x} = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$

Output: $y \in \{-1, +1\}$

Hipótese:

$$h(\mathbf{x}) = \text{sign}\Big(\big(\sum_{i=1}^{d} w_i x_i \big) + b \Big), \; b \in \mathbb{R}, w_i \in \mathbb{R}, i = 1, 2, \ldots, d$$

Fazendo

$$\mathbf{x} = (1, x_1, x_2, \ldots, x_d) \in \mathbb{R}^{d+1}$$
$$\mathbf{w} = (w_0, w_1, w_2, \ldots, w_d) \in \mathbb{R}^{d+1}$$

temos

$$h(\mathbf{x}) = \text{sign}\big(\mathbf{w}^T \mathbf{x}\big)$$

Perceptron

Input: $\mathbf{x} = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$

Output: $y \in \{-1, +1\}$

Hipótese:

$$h(\mathbf{x}) = \text{sign}\Big(\big(\sum_{i=1}^{d} w_i x_i \big) + b \Big), \; b \in \mathbb{R}, w_i \in \mathbb{R}, i = 1, 2, \ldots, d$$

Fazendo

$$\mathbf{x} = (1, x_1, x_2, \ldots, x_d) \in \mathbb{R}^{d+1}$$
$$\mathbf{w} = (w_0, w_1, w_2, \ldots, w_d) \in \mathbb{R}^{d+1}$$

temos

$$h(\mathbf{x}) = \text{sign}\big(\mathbf{w}^T \mathbf{x}\big) \qquad (\; h(\mathbf{x}) = 0 \text{ define um hiperplano })$$

Perceptron Algorithm

É um **algoritmo iterativo**

Seja **w** o **peso "atual"**
Seja $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)}), i = 1, \ldots, N\}$ o **conjunto de treinamento**

1. **Escolher** em $\mathcal{D}$ um par $(\mathbf{x}^{(i)}, y^{(i)})$ tal que

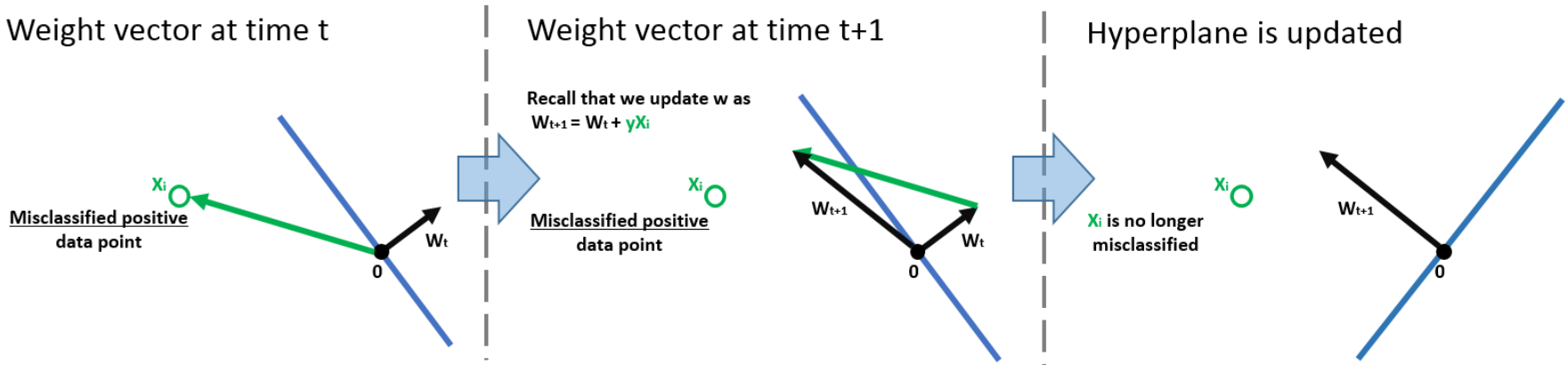
$$\text{sign}(\mathbf{w}^T \mathbf{x}^{(i)}) \neq y^{(i)}$$

   Se não houver tal par, parar.

2. **Atualizar o peso** fazendo

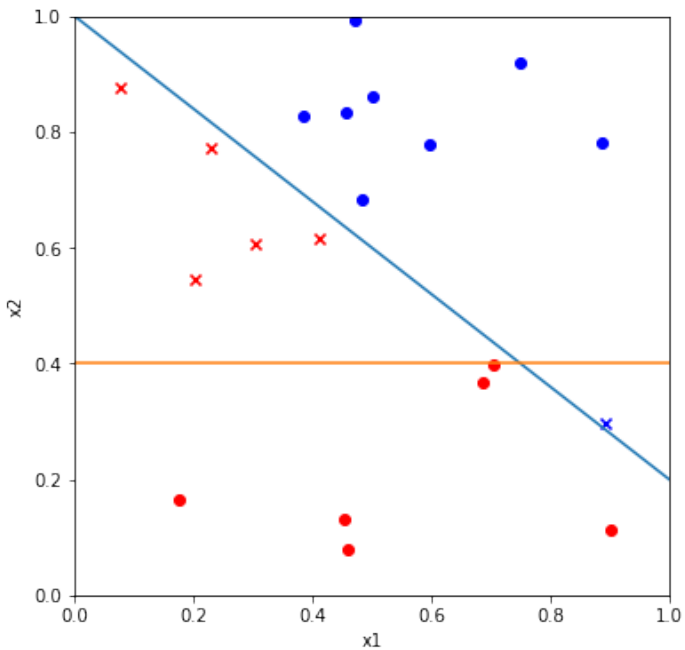
$$\mathbf{w} = \mathbf{w} + y^{(i)}\mathbf{x}^{(i)} \qquad y^{(i)} \in \{-1, +1\}$$

Perceptron Algorithm – intuição



http://www.cs.cornell.edu/courses/cs4780/2015fa/web/lecturenotes/lecturenote03.html

Perceptron - exemplo de animação



Target

$f(x_1, x_2) = -1 + 0.8x_1 + x2$

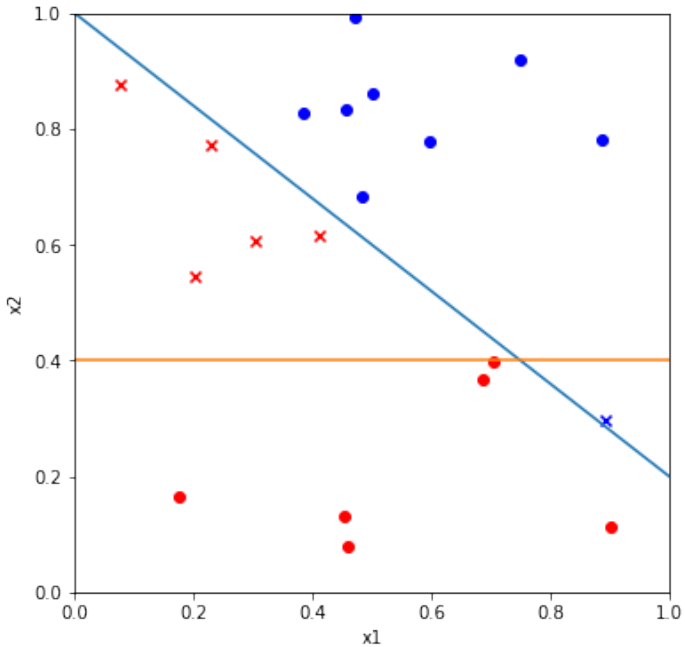
Hipótese inicial

$h_0(x_1, x_2) = -0.4 + x_2$

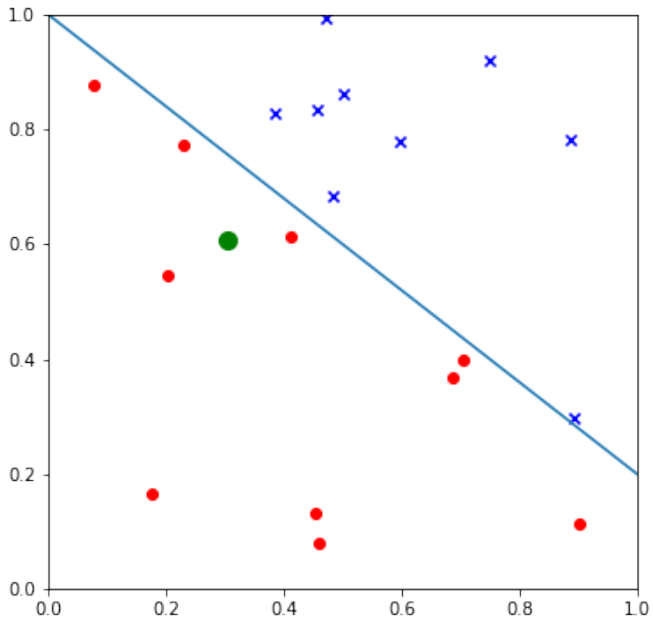
Acertos e erros de h_0:

- true positive $\surd$
- true negative $\surd$
- $\times$ false negative X
- $\times$ false positive X

Perceptron - exemplo de animação



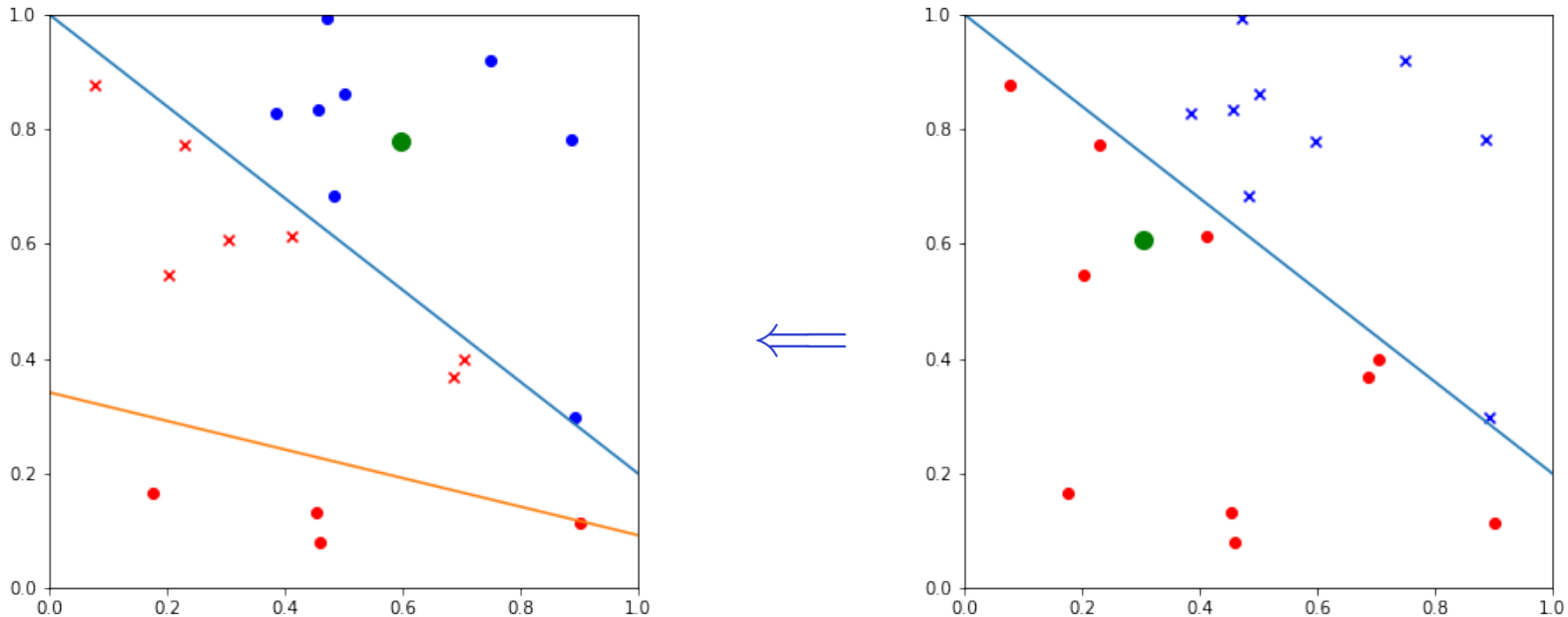
$\Longrightarrow$



$$\mathbf{w}(0) = (-0.4,\ 0,\ 1) \overset{-(1,\,0.3,\,0.6)}{\Longrightarrow} \mathbf{w}(1) = (-1.4,\ -0.3,\ 0.4)$$

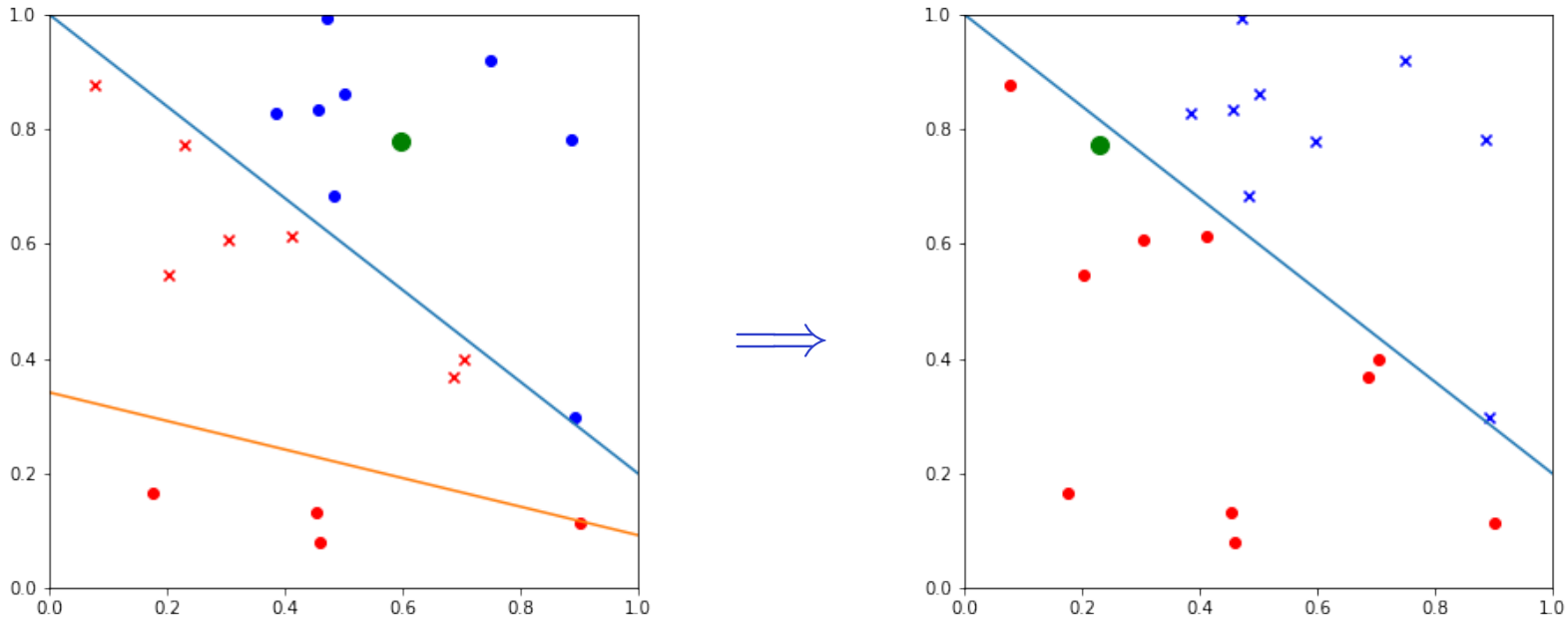
iteração 1

Perceptron - exemplo de animação



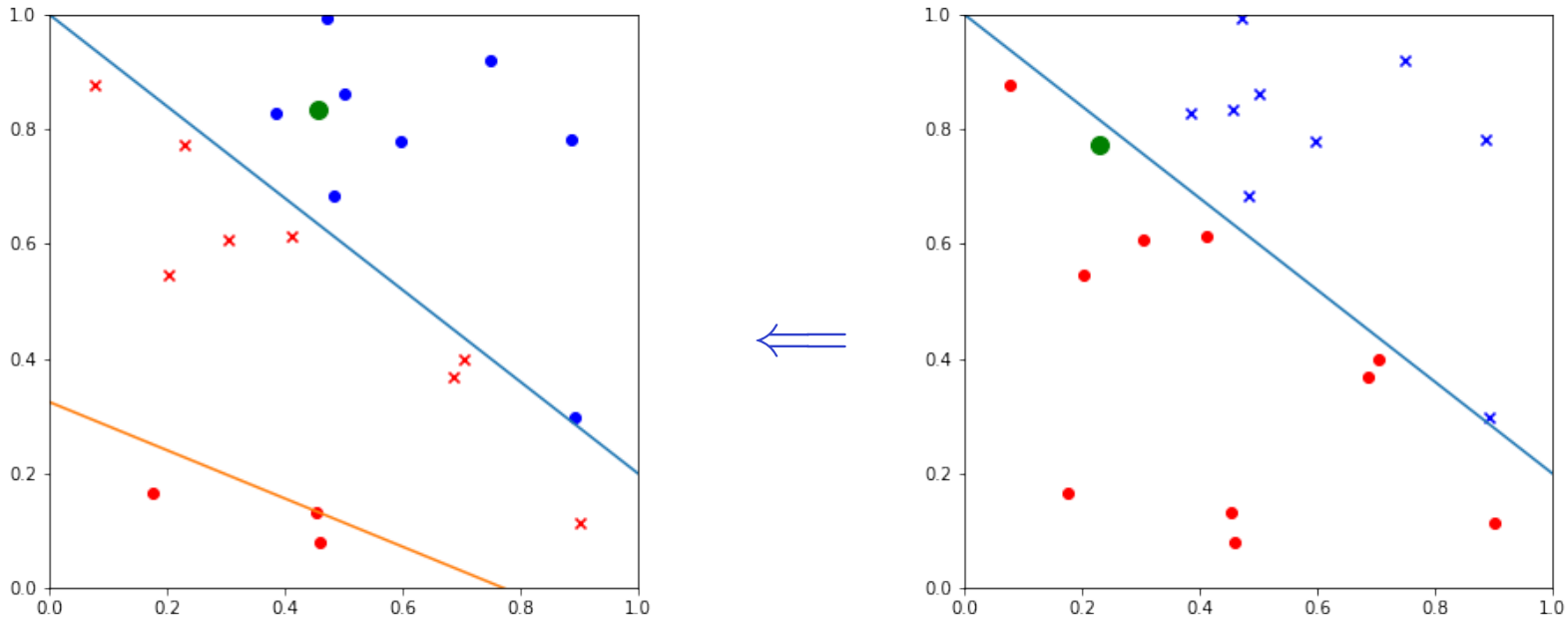
iteração 2

Perceptron - exemplo de animação



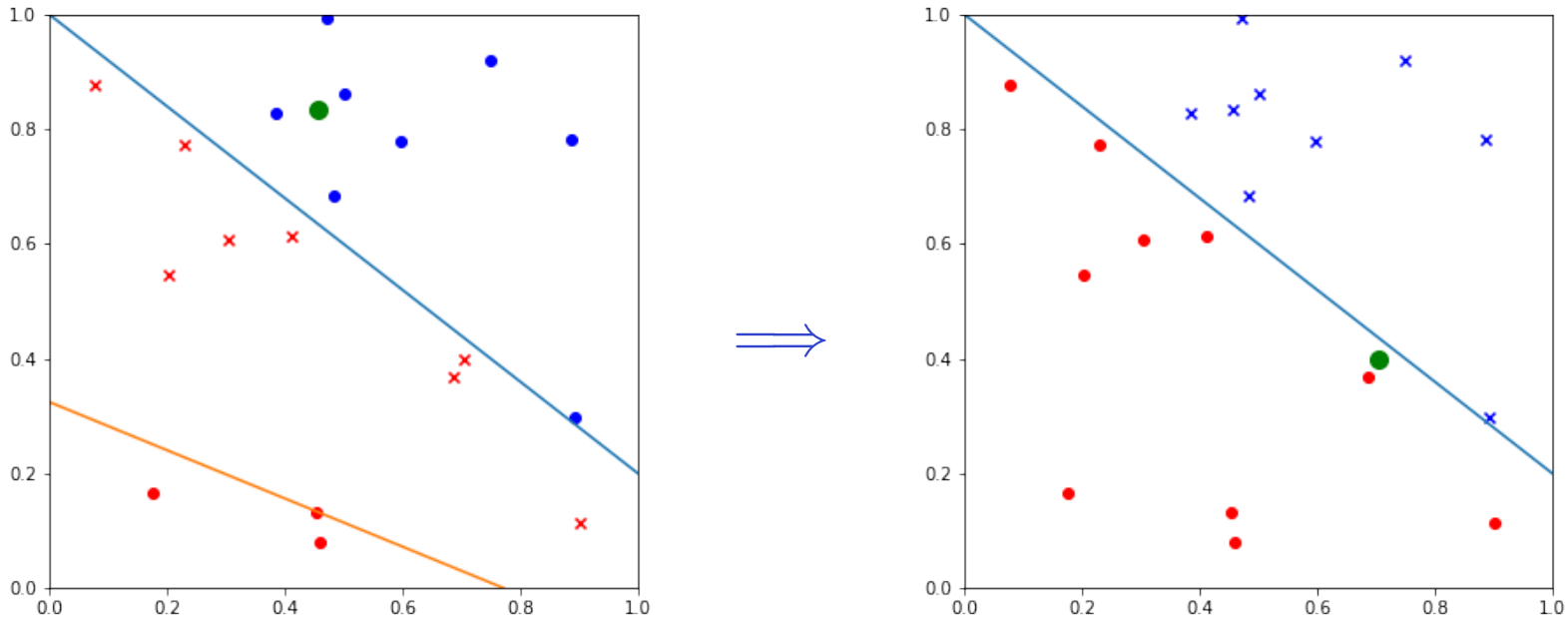
iteração 3

Perceptron - exemplo de animação



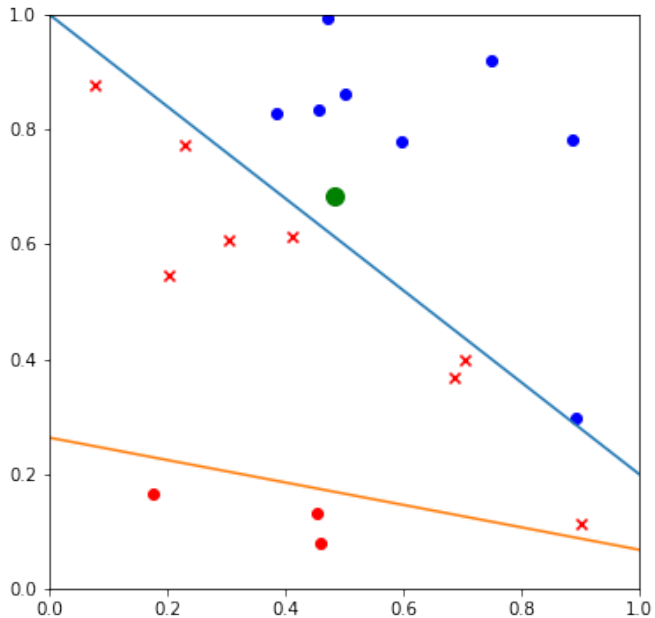
iteração 4

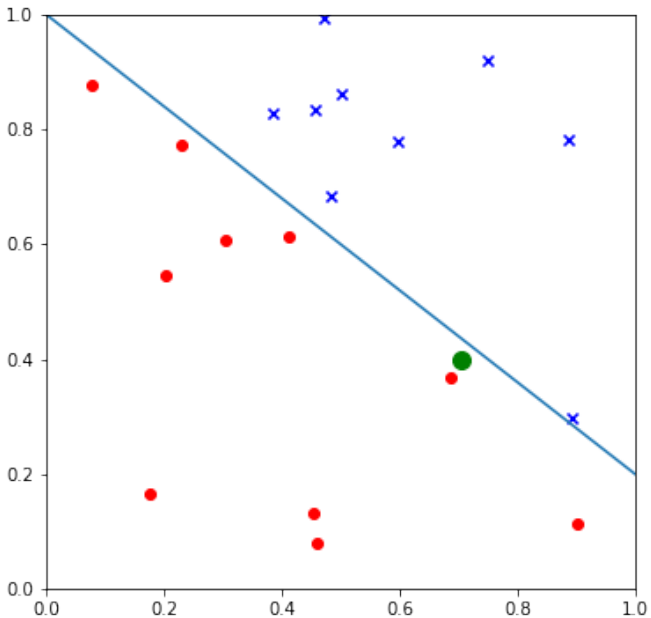
Perceptron - exemplo de animação



iteração 5

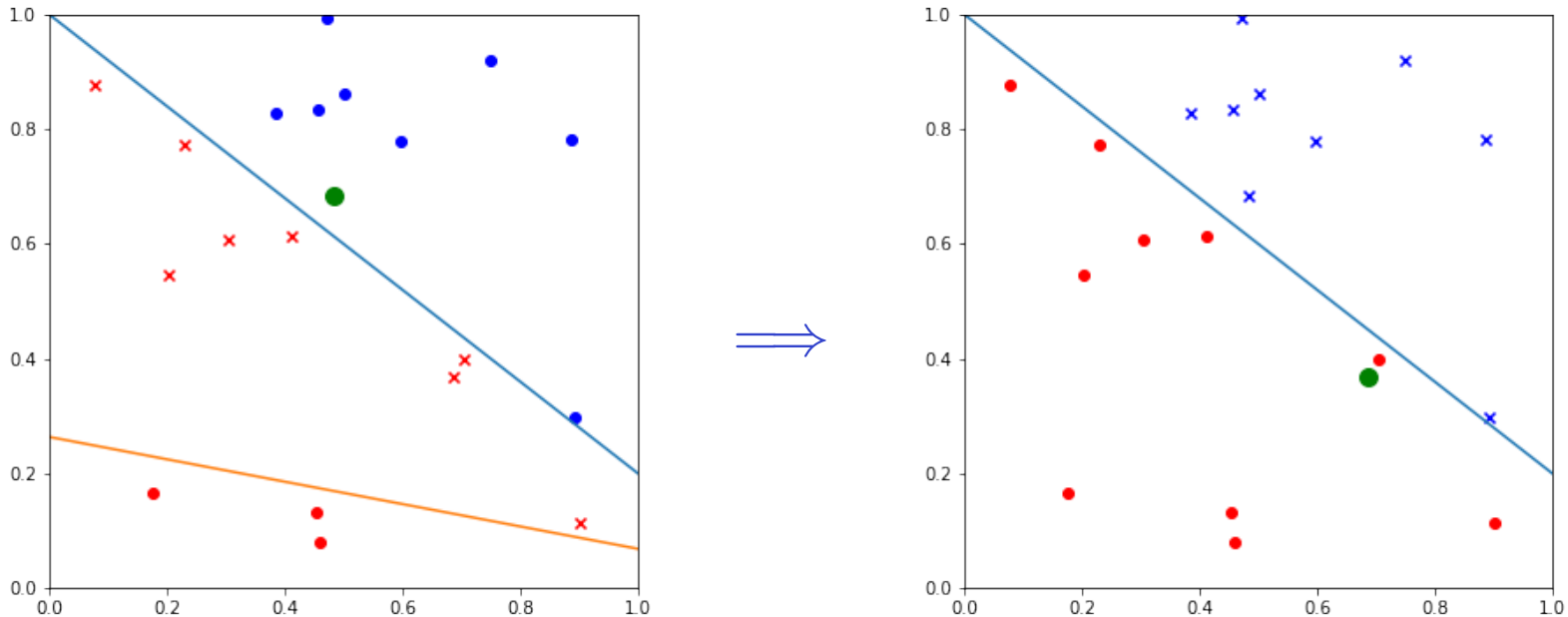
Perceptron - exemplo de animação





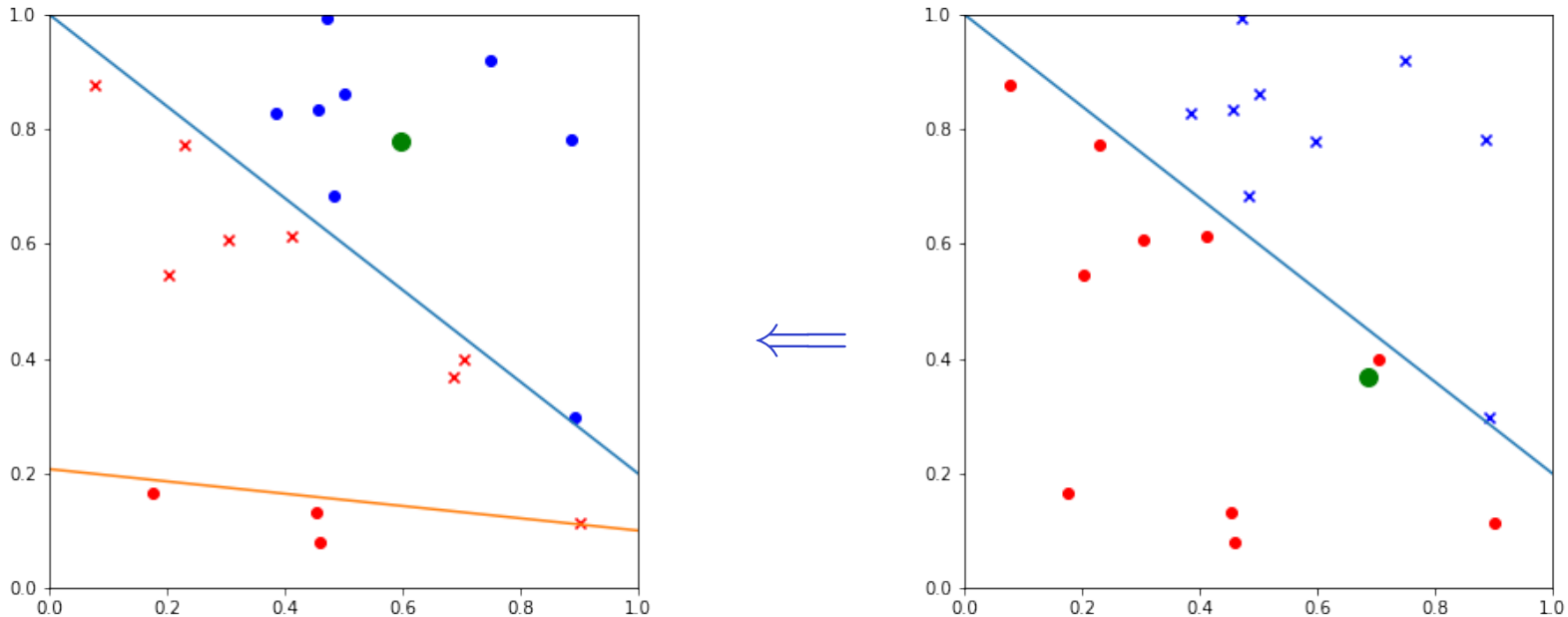
iteração 6

Perceptron - exemplo de animação



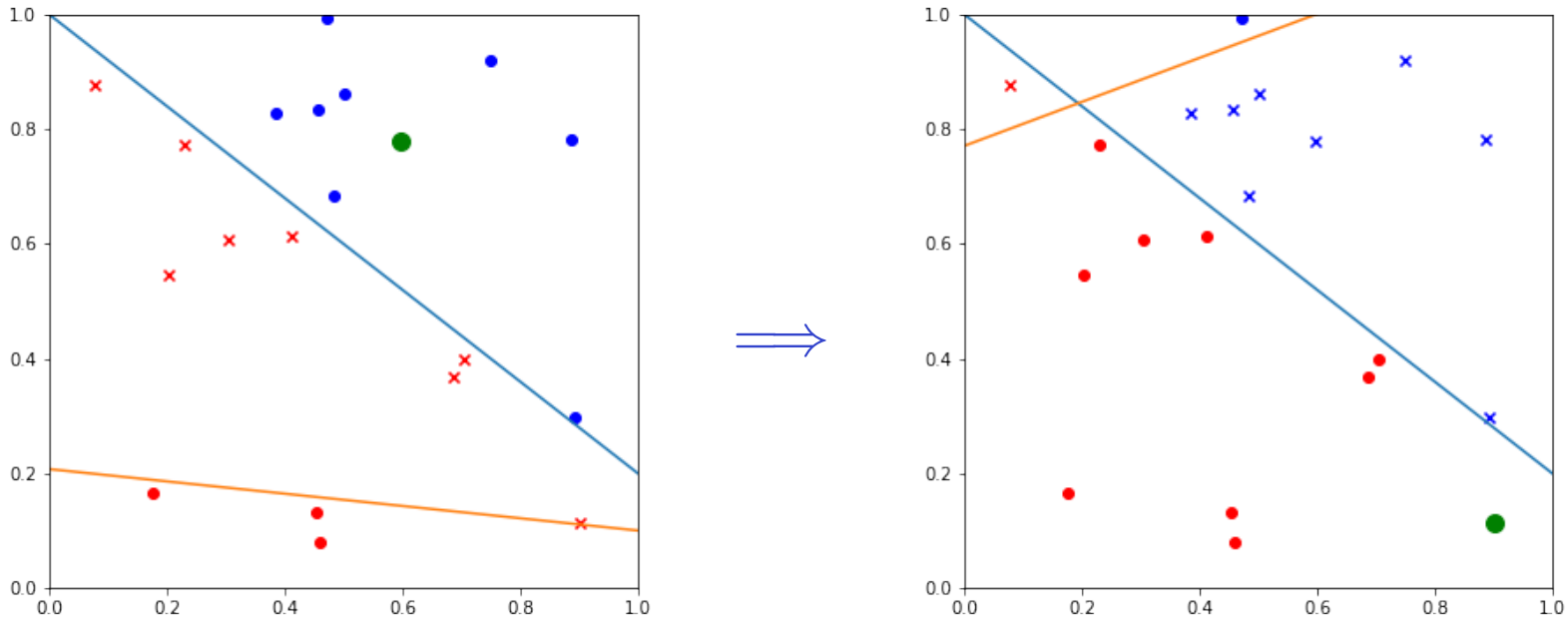
iteração 7

Perceptron - exemplo de animação



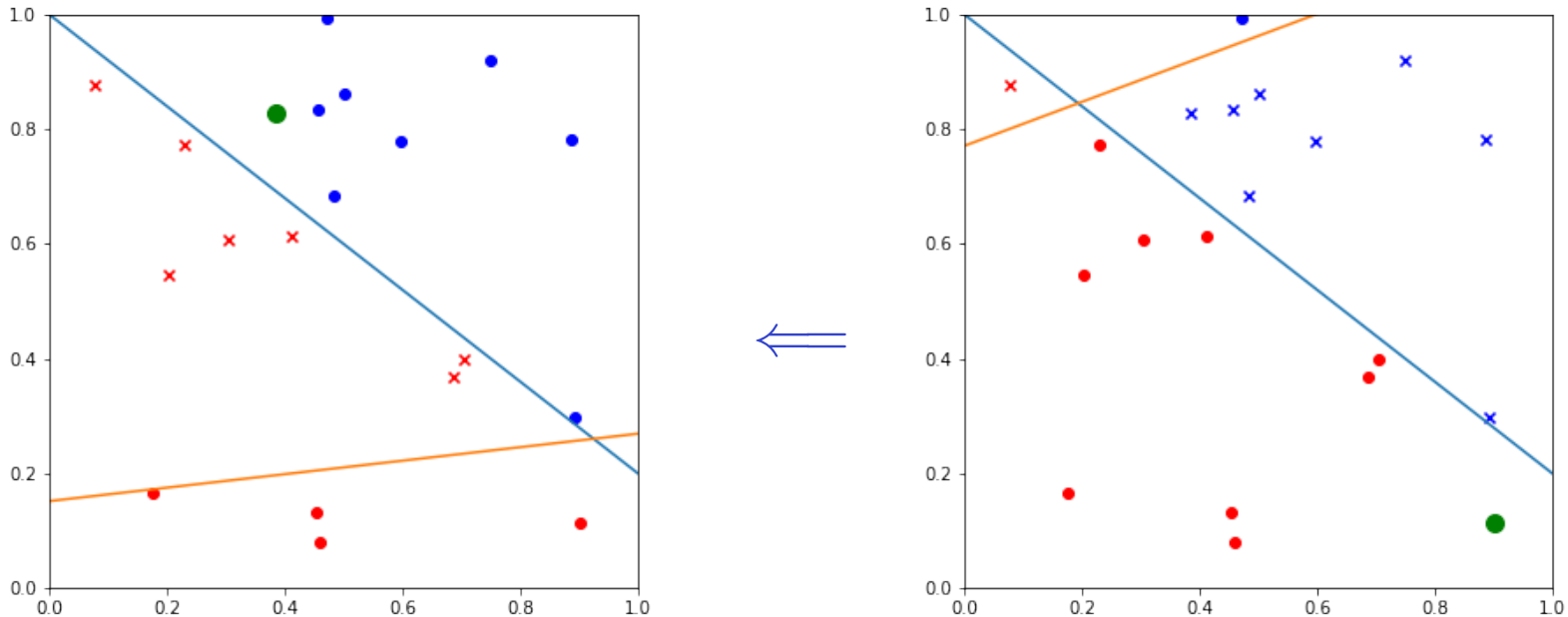
iteração 8

Perceptron - exemplo de animação



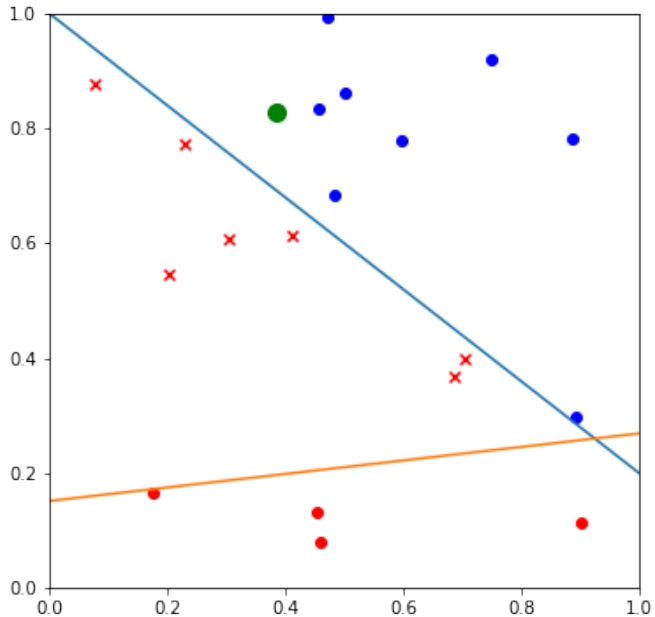
iteração 9

Perceptron - exemplo de animação



iteração 10

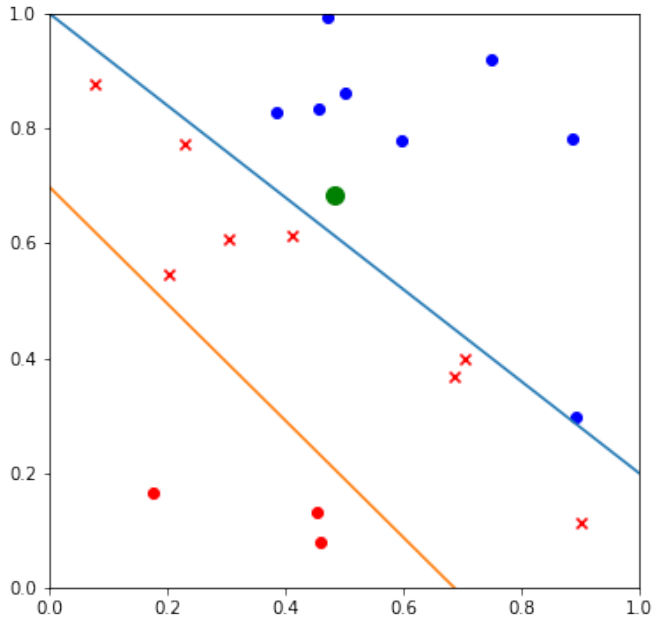
Perceptron - exemplo de animação

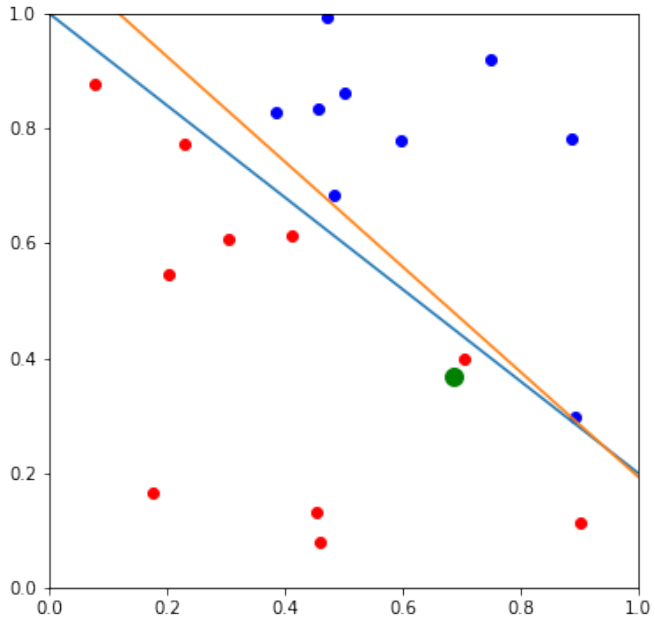


$\Longrightarrow \quad \cdots$

e assim por diante ...

Perceptron - exemplo de animação





Após 77 iterações

Convergência do Perceptron algorithm

Ideia: Suponha que os elementos das duas classes podem ser separados por um hiperplano com margem γ — existe um vetor de pesos $\mathbf{w}$ com norma 1 (i.e., $||\mathbf{w}|| = 1$) e bias term b tal que $y^{(i)}\mathbf{w}^T\mathbf{x}^{(i)} > \gamma$, $\forall i$.
R a norma máxima de amostras $\mathbf{x}^{(i)} \in \mathcal{D}$.
A prova consiste em se mostrar que o algoritmo converge em $\mathcal{O}(R^2/\gamma^2)$ iterações; a cada iteração a direção do vetor de pesos sendo atualizado aproxima-se consistentemente do vetor de pesos do hiperplano separador.

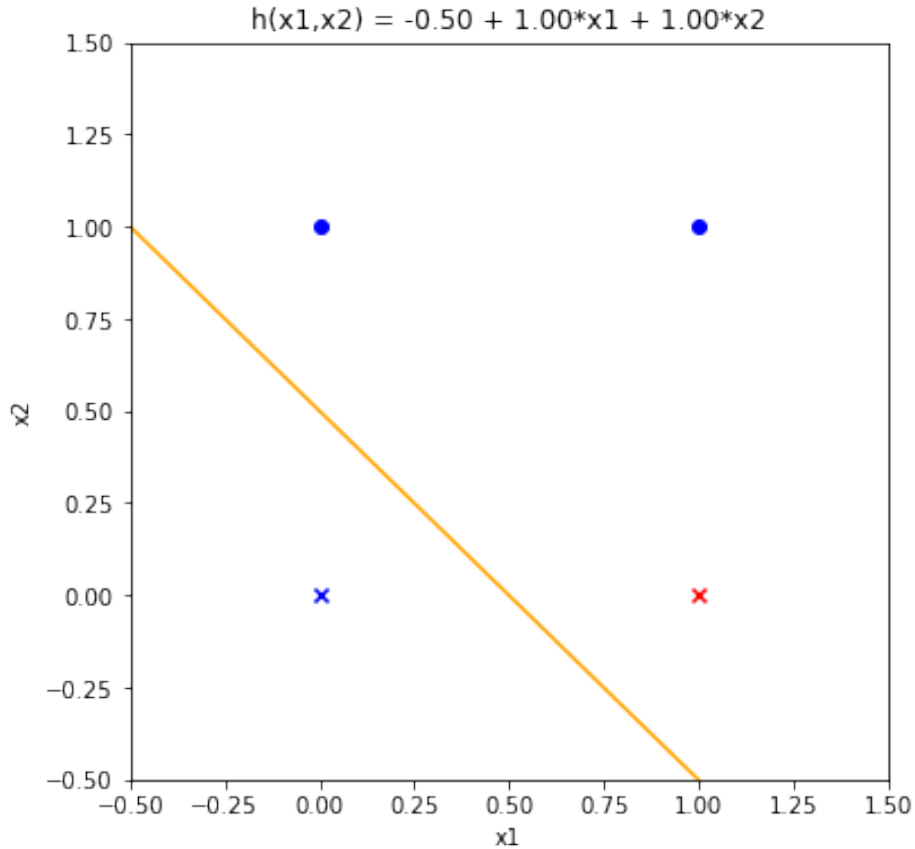
Mais detalhes:

- Livro: Marvin Minsky and Seymour Papert, Perceptrons
- https://www.cse.iitb.ac.in/~shivaram/teaching/old/cs344+386-s2017/resources/classnote-1.pdf
- etc ...

Exemplo para ser feito à mão

$$\mathbf{w}(0) = (-0.5,\ 1,\ 1)$$

x_1 x_2	y
0 0	1
0 1	1
1 0	−1
1 1	1



Processe os exemplos ciclicamente, começando pelo 0 0

Resposta: $(0.5,\ -1,\ 2)$ (pode variar dependendo da ordem na qual os exemplos foram visitados)