Ministério da Ciência, Tecnologia e Inovações

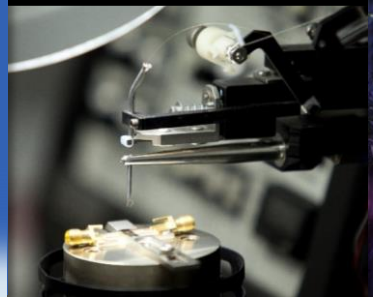
Centro Brasileiro de Pesquisas Físicas

Métodos para Análise de grande volume de dados e Astroinformática

Clécio Roque De Bom – debom@cbpf.br

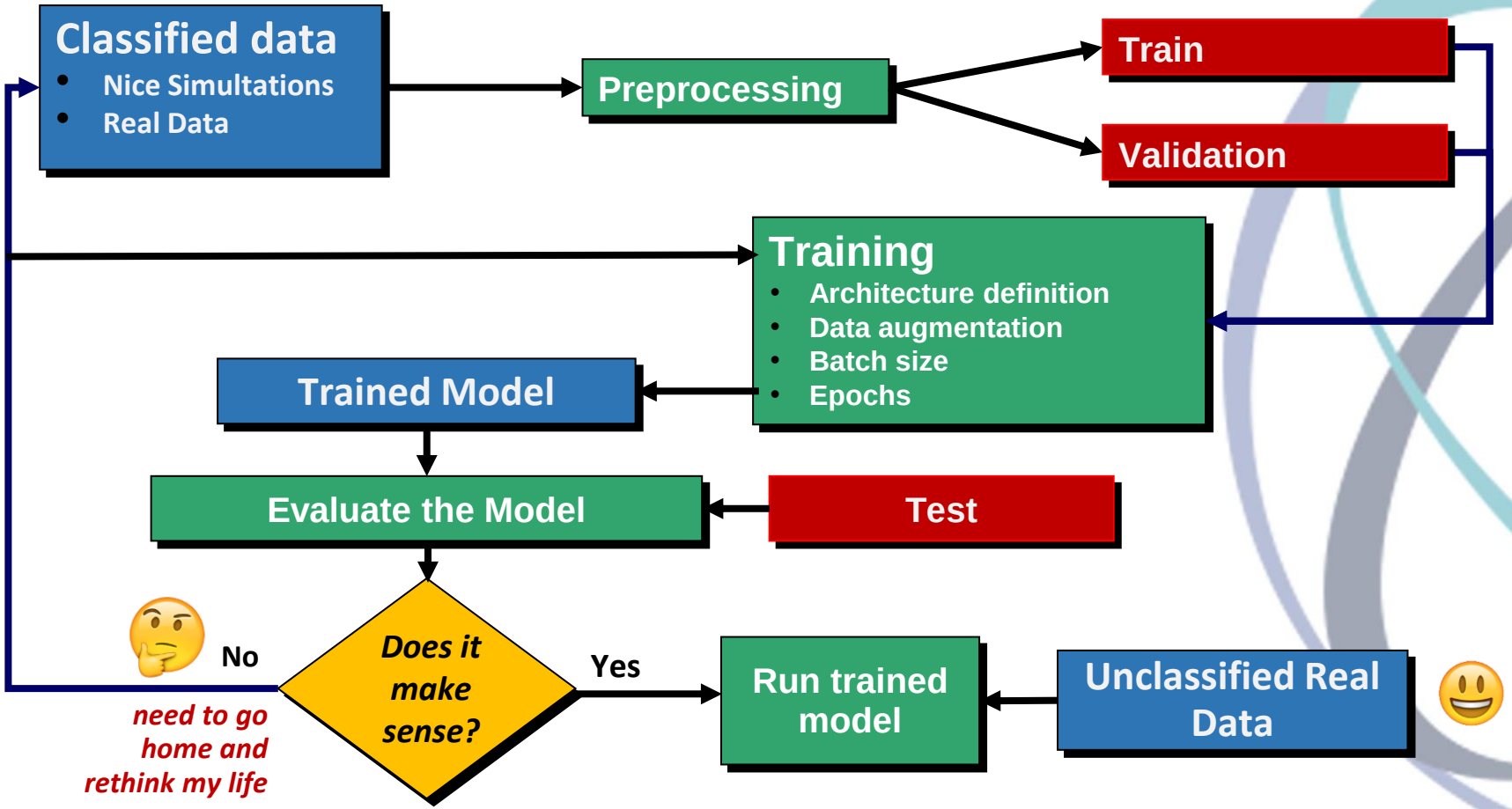
CBPF

clearnightsrthebest.com



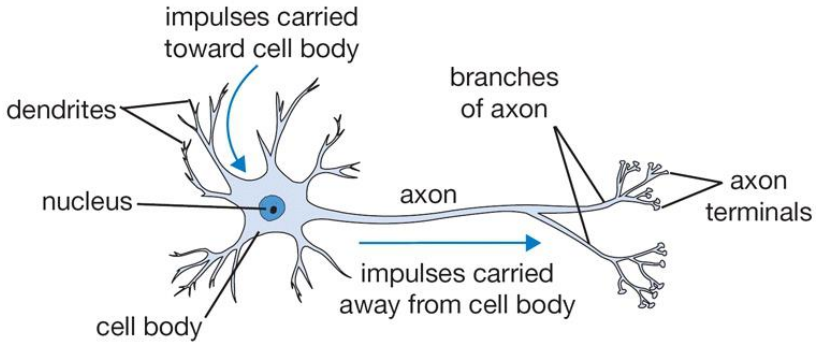


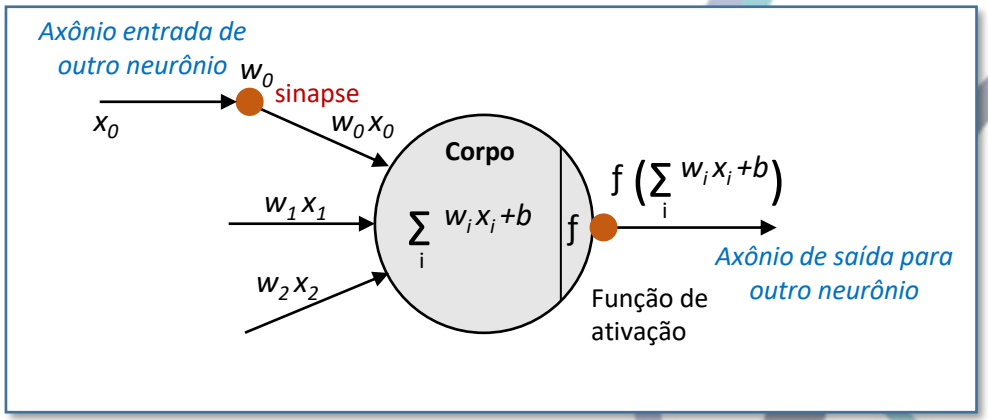

ML: Workflow Supervised – Again!



Neurônio

Computação (Neurônio Artificial) → Inspiração na Biologia





Neurônio Biológico: bloco computacional de processamento do cérebro.

Cérebro Humano: ~100 – 1.000 trilhões de sinapses

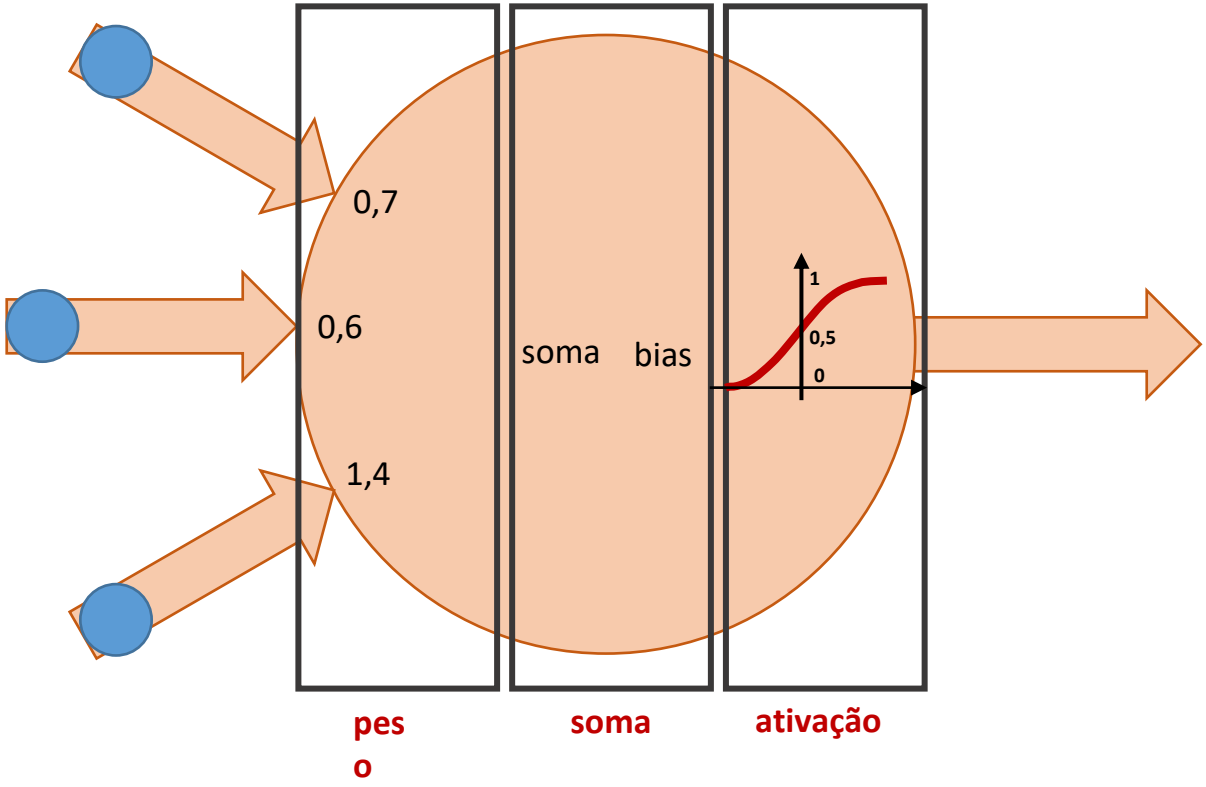
10.000 x

Neurônio Artificial: bloco computacional de processamento das Redes Neurais Artificiais.

Rede Neural Artificial : ~1 – 10 bilhões de sinapses.

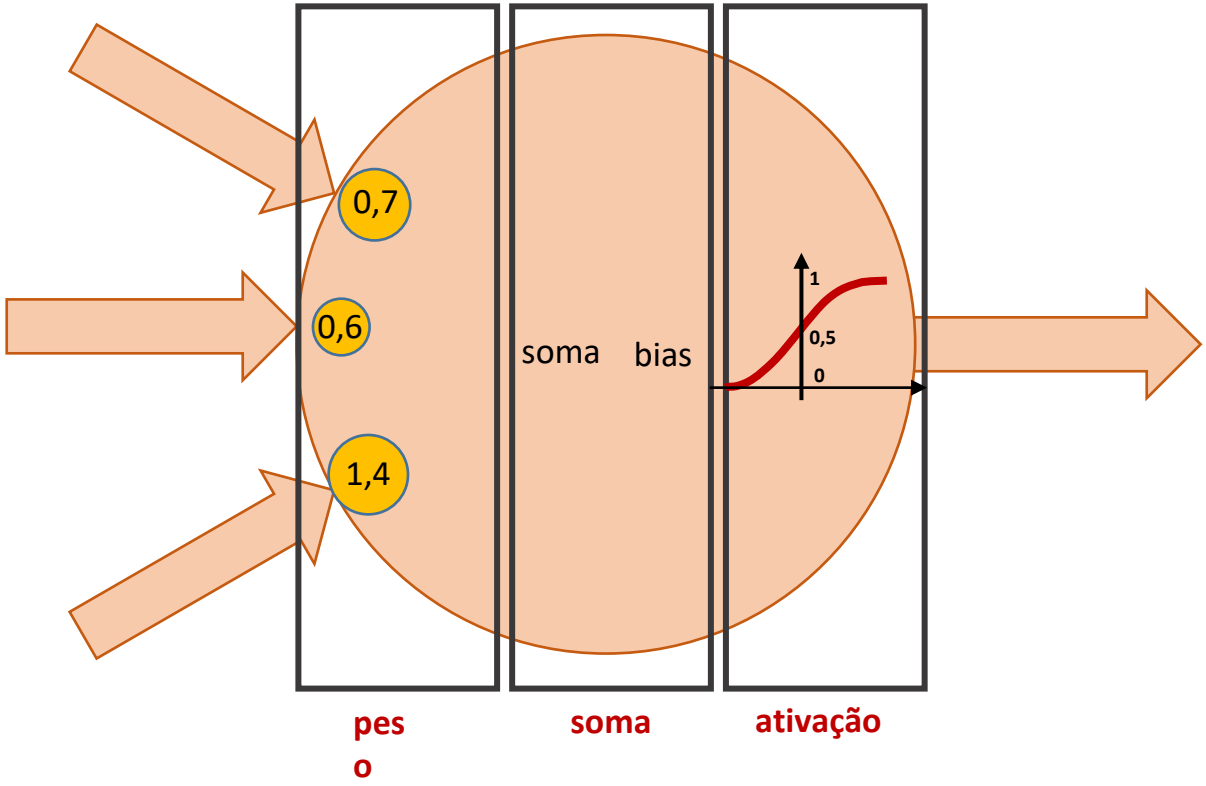
Perceptron

Frank Rosenblatt (1957)



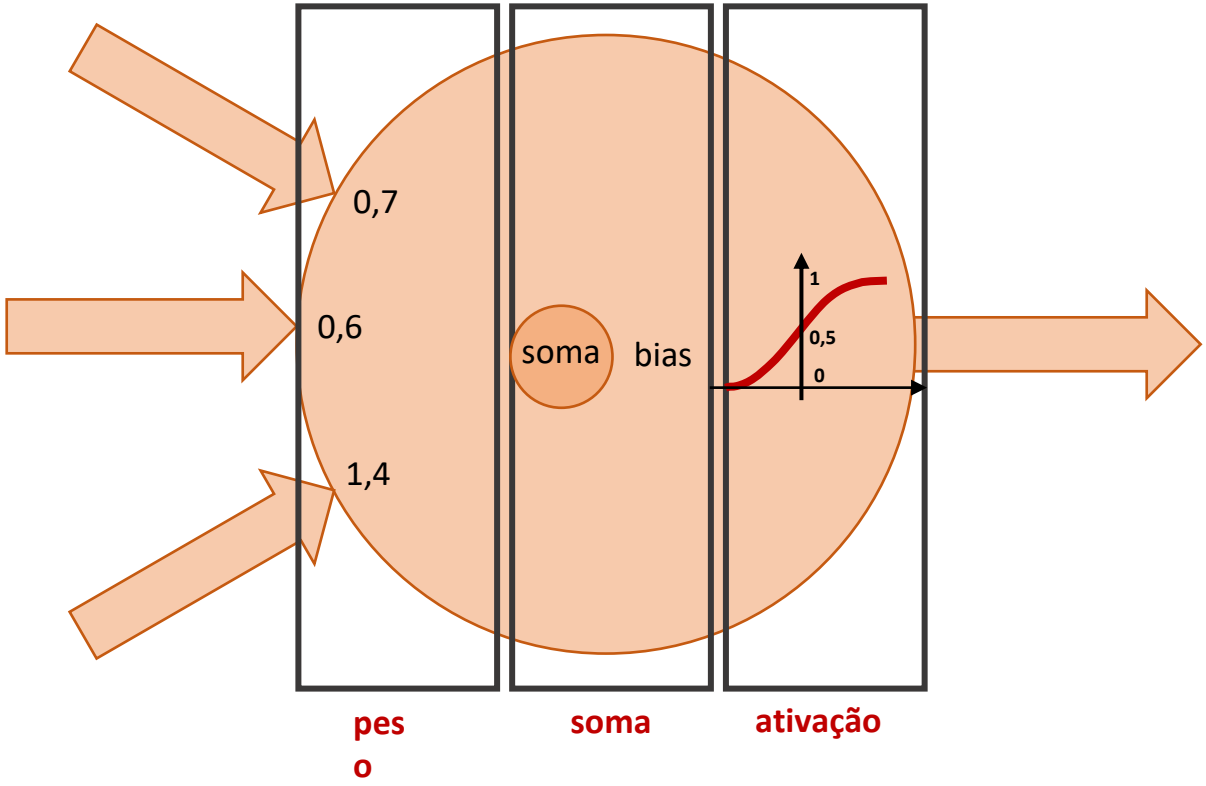
Perceptron

Frank Rosenblatt (1957)



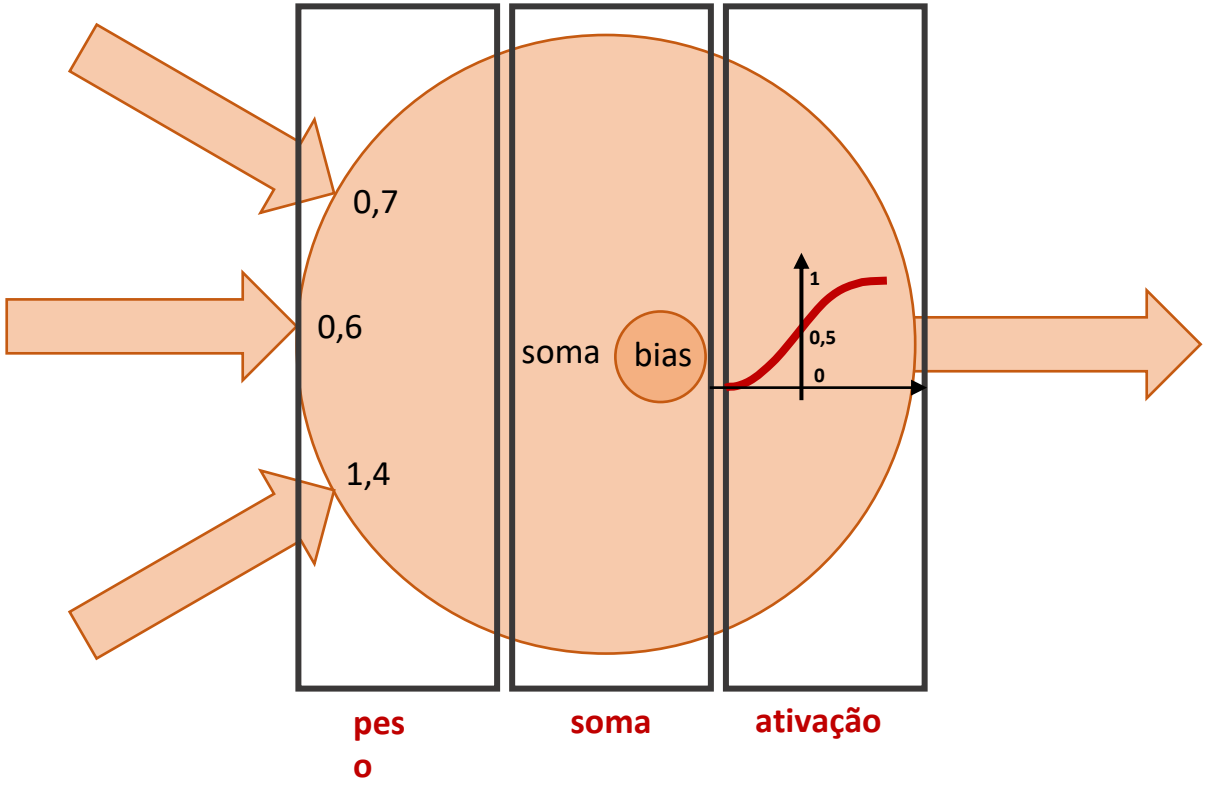
Perceptron

Frank Rosenblatt (1957)



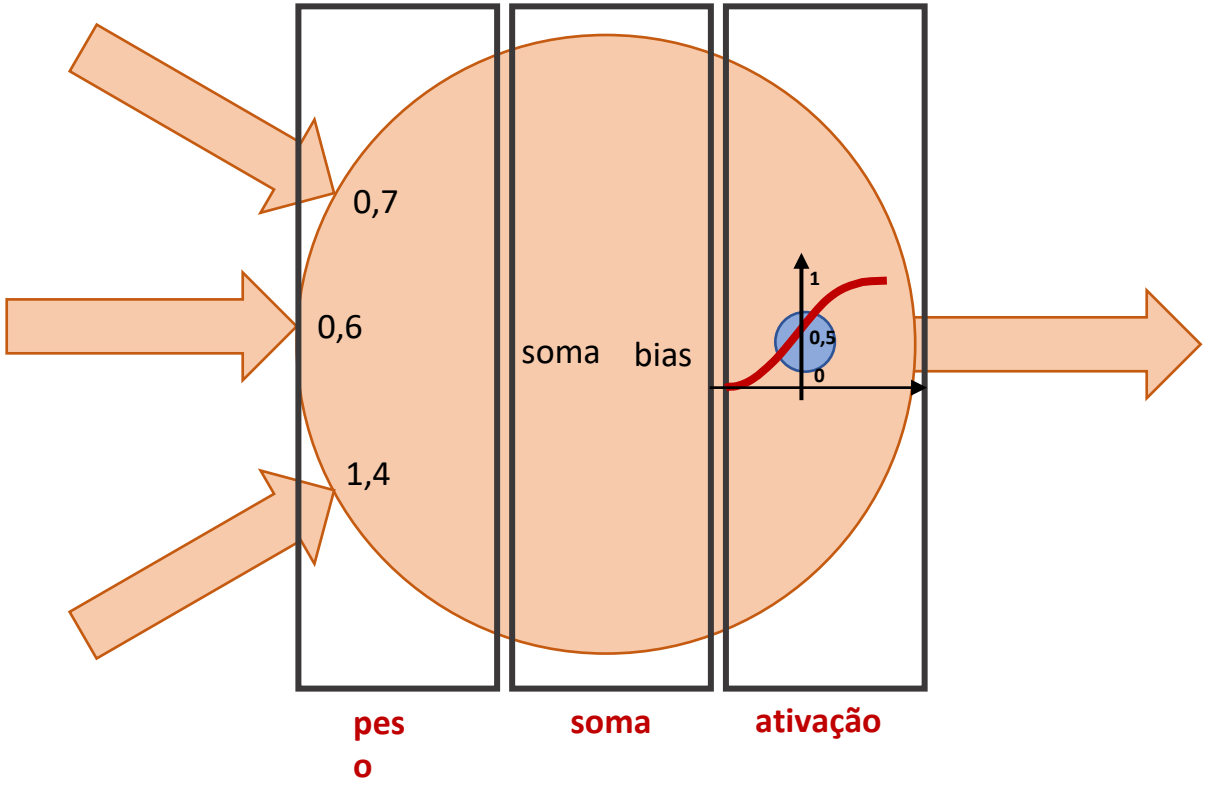
Perceptron

Frank Rosenblatt (1957)



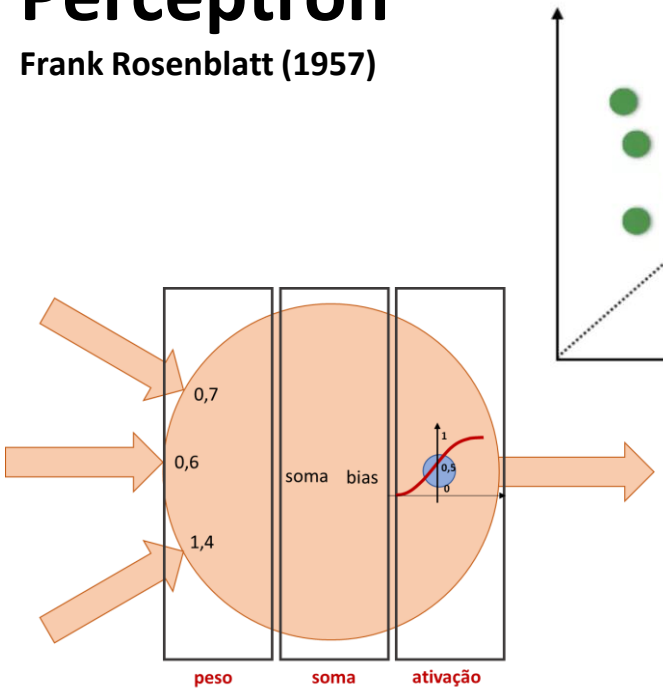
Perceptron

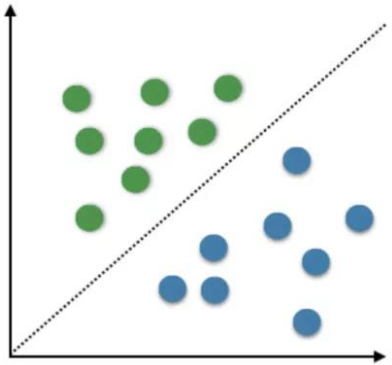
Frank Rosenblatt (1957)



Perceptron

Frank Rosenblatt (1957)



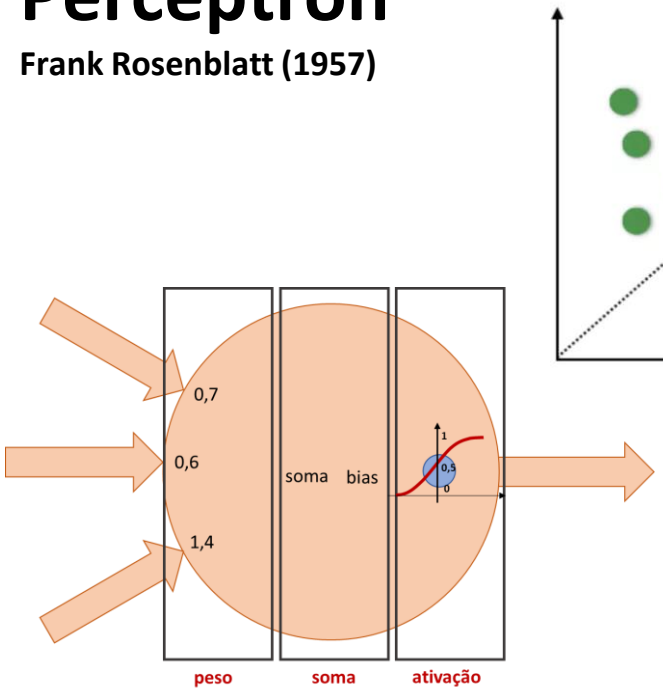


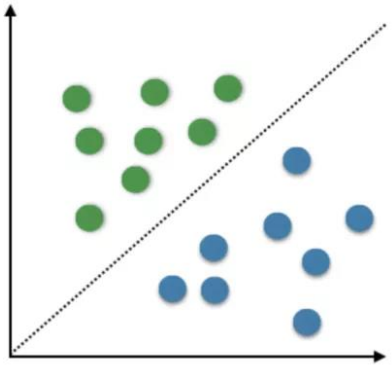
Algoritmo

- Inicialize a rede Perceptron com pesos (w) aleatórios;
- Para uma data entrada, processe a saída da rede;
- Se a saída da rede não for igual a saída desejada, então a rede deve ser alterada, trocando os valores dos pesos (w) das sinapses;
- Repita esse procedimento com todos os dados de treinamento até a rede Perceptron não apresentar mais erros.

Perceptron

Frank Rosenblatt (1957)

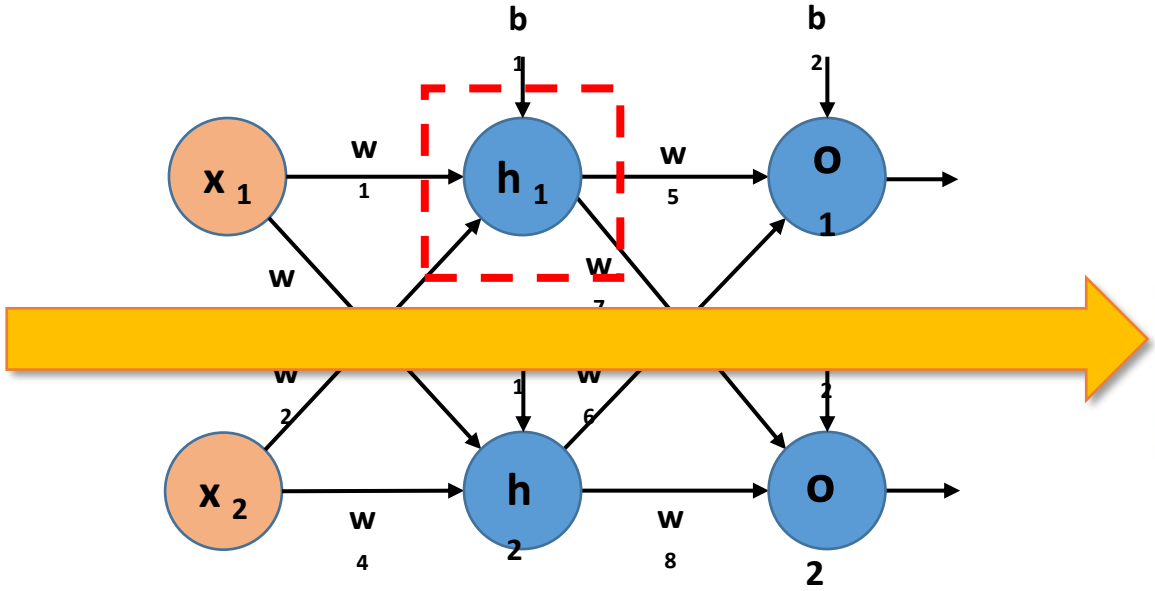




- **Inicialize a rede Perceptron com pesos (w) aleatórios;**
- **Para uma data entrada, processe a saída da rede;**
- **Se a saída da rede não for igual a saída desejada, então a rede deve ser alterada, trocando os valores dos pesos (w) das sinapses;**
- **Repita esse procedimento com todos os dados de treinamento até a rede Perceptron não apresentar mais erros.**

Feedforward and Backpropagation

Fase 1
Propagação

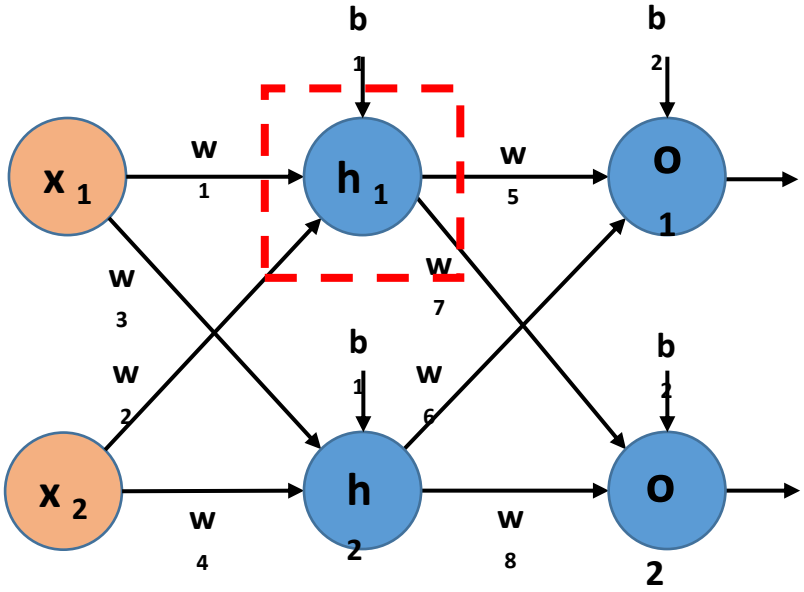


$$uh_1 = x_1 * w_1 + x_2 * w_2 + b_1 * 1$$

$$g(h_1) = g(uh_1) = \frac{1}{1+e^{uh_1}}$$

Feedforward and Backpropagation

Fase 1
Propagação

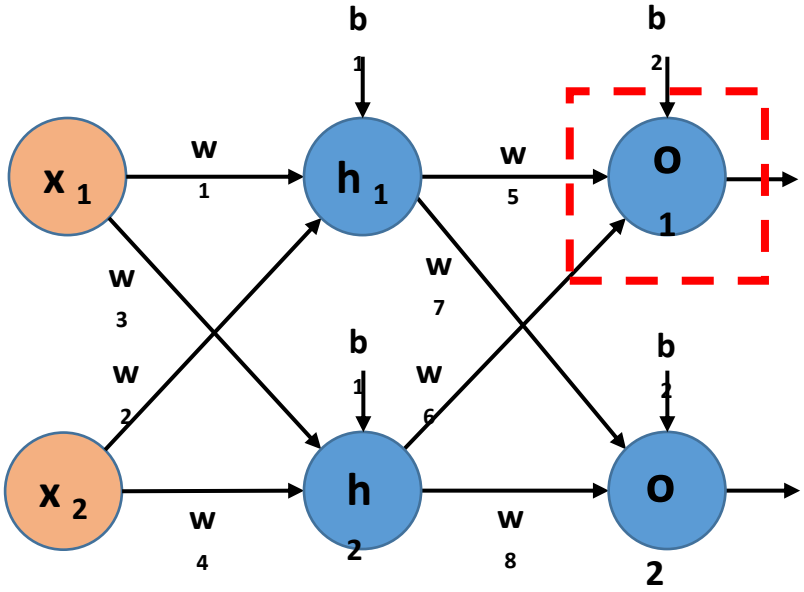


$$uh_2 = x_1 * w_3 + x_2 * w_4 + b_1 * 1$$

$$g(h_2) = g(uh_2) = \frac{1}{1+e^{uh_2}}$$

Feedforward and Backpropagation

Fase 1
Propagação

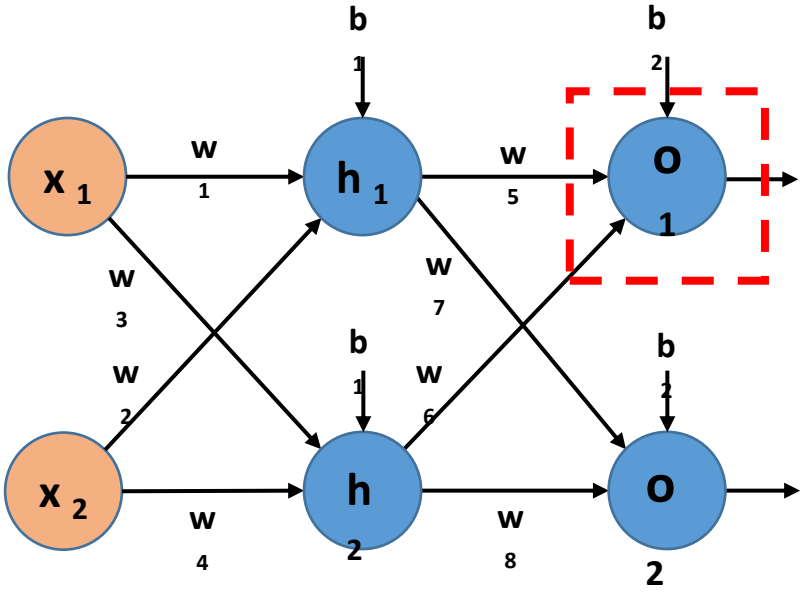


$$uo_1 = \mathrm{g}(h_1) * w_5 + \mathrm{g}(h_2) * w_6 + b_2 * 1$$

$$\hat{y}_1 = \mathrm{g}(o_1) = \mathrm{g}(uo_1) = \frac{1}{1+e^{uo_1}}$$

Feedforward and Backpropagation

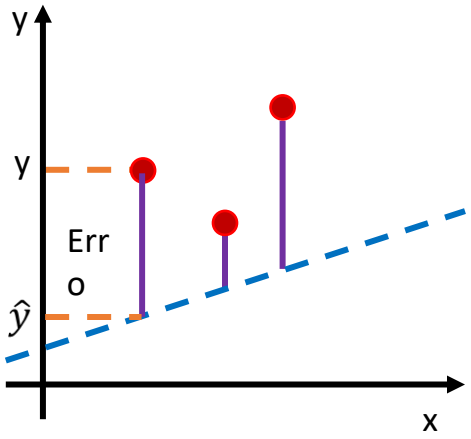
Fase 1
Propagação



$$uo_2 = \mathrm{g}(h_1) * w_7 + \mathrm{g}(h_2) * w_8 + b_2 * 1$$

$$\hat{y}_2 = \mathrm{g}(o_2) = \mathrm{g}(uo_2) = \frac{1}{1+e^{uo_2}}$$

Rede Neural Artificial: Erro/Custo



y = valor original

$\hat{y}$ = valor predito

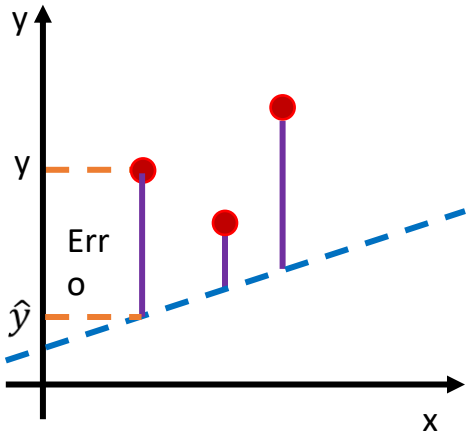
$$\hat{y} = w_0 + w_1 x$$

$$J(w_0, w_1) = \frac{\sum_{i=1}^{m} (\hat{y}_i - y_i)^2}{m}$$ (média)

CUST
COST

MSE (*Mean Square Error* – Erro quadrático médio)

Rede Neural Artificial: Erro/Custo



y = valor original

$\hat{y}$ = valor predito

$$\hat{y} = w_0 + w_1 x$$

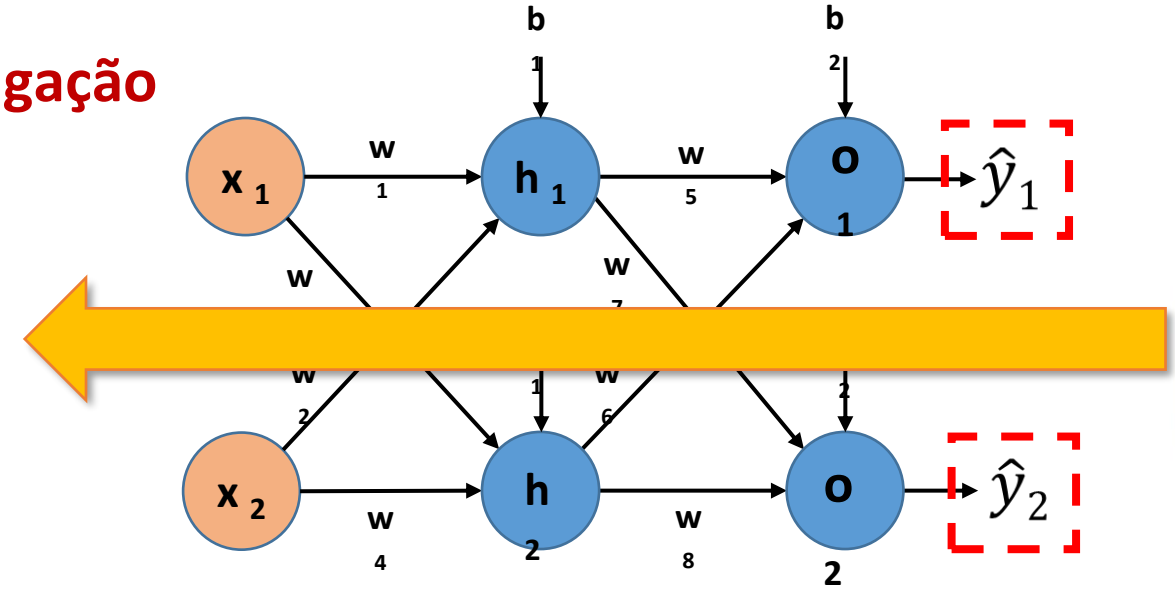
$$J(w_0, w_1) = \frac{\mathbf{1}}{\mathbf{2m}} \sum_{i=1}^{m} (\hat{y}_i - y_i)^2$$

MSE (*Mean Square Error* – Erro quadrático médio)

Como reduzir o custo? $\min_{(w_0, w_1)} J(w_0, w_1)$

Feedforward and Backpropagation

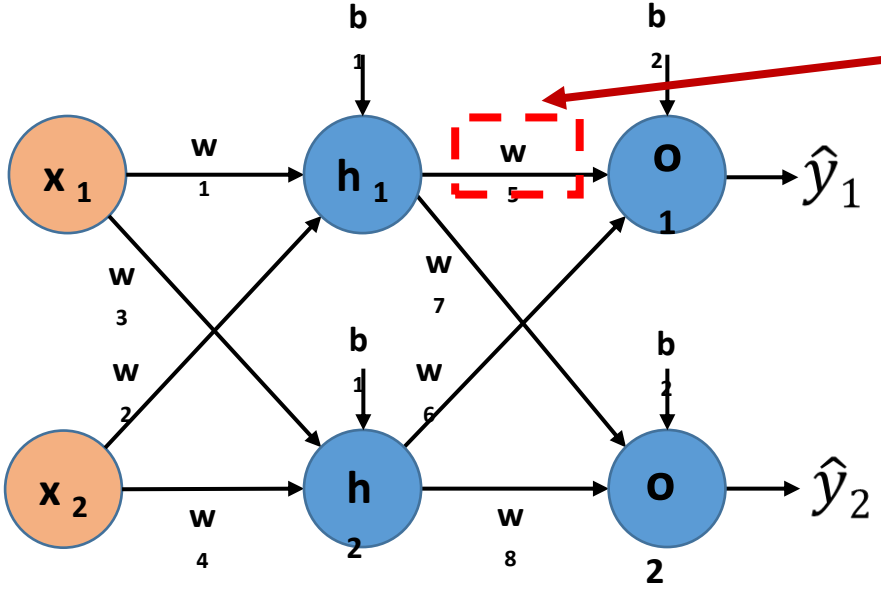
Fase 1
Retropropagação



$$E_{total} = \frac{1}{2}\sum_{k=1}^{N}(\hat{y}_k - y_k)^2 = E_{o1} + E_{o2}$$

Feedforward and Backpropagation

Fase 1
Retropropagação

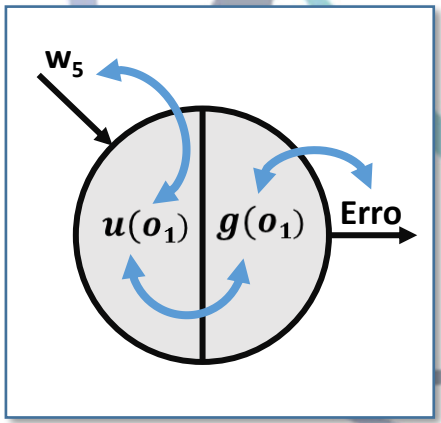


Correção de w5:
Queremos estimar quanto w5 afeta o Erro total

$$E_{total}$$

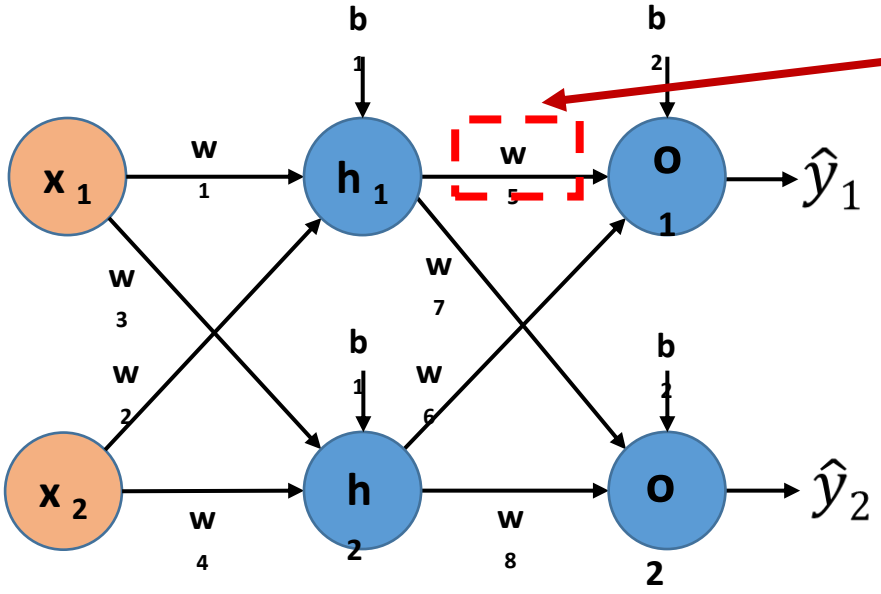
$$\frac{\partial E_{total}}{\partial w_5} = gradiente\ em\ relação\ a\ w_5$$

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial go_1} * \frac{\partial go_1}{\partial uo_1} * \frac{\partial uo_1}{\partial w_5}$$



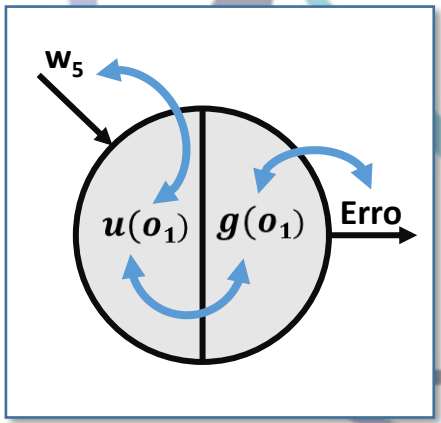
Feedforward and Backpropagation

Fase 1
Retropropagação



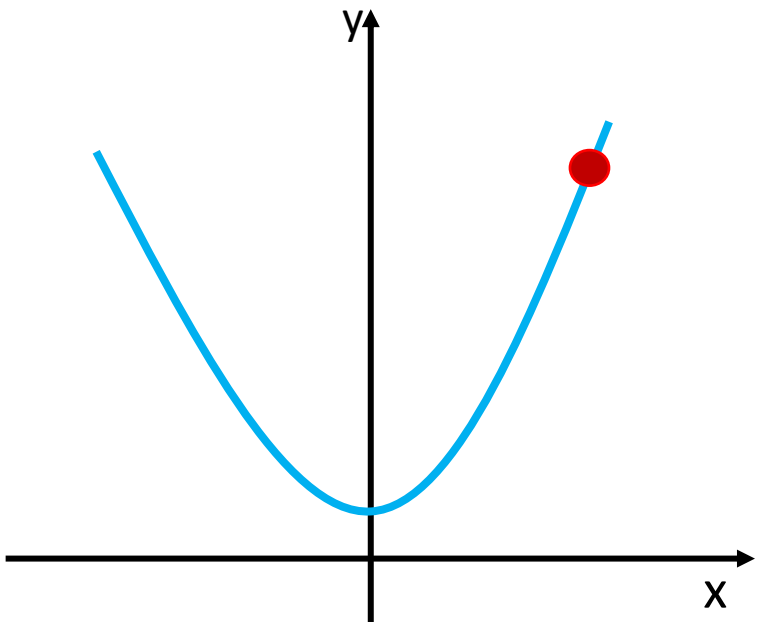
Correção de w5:
Queremos estimar quanto w5 afeta o Erro total

$$E_{total}$$



$$w_5^+ = w_5 - \eta * \frac{\partial E_{total}}{\partial w_5} :$$

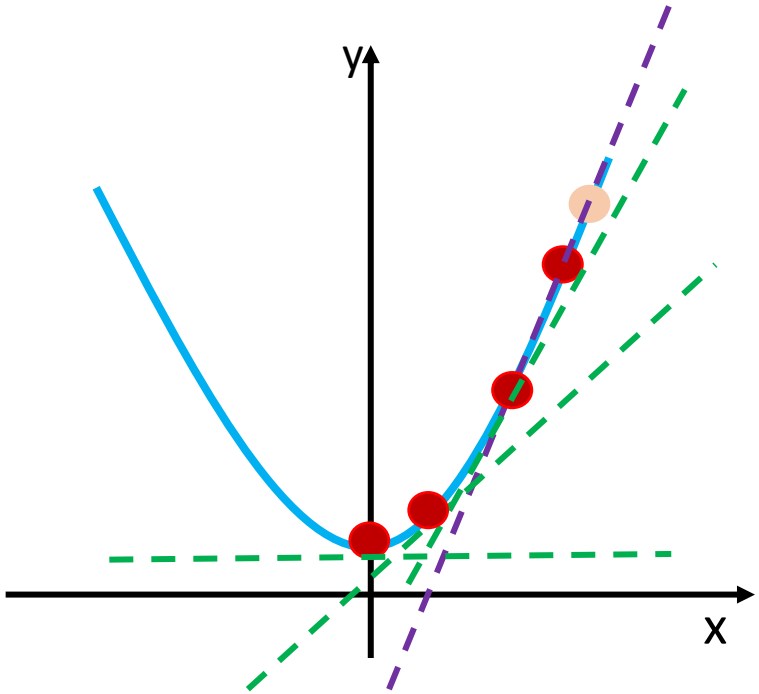
Rede Neural Artificial: Otimização



Função quadrática

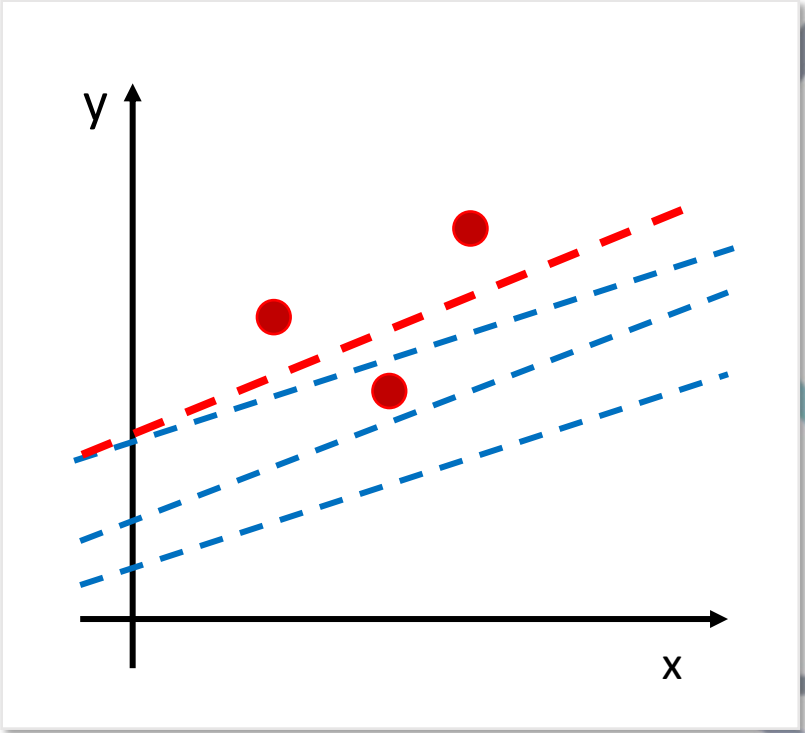
$$(\hat{y}_i - y_i)^2$$

Rede Neural Artificial: Otimização

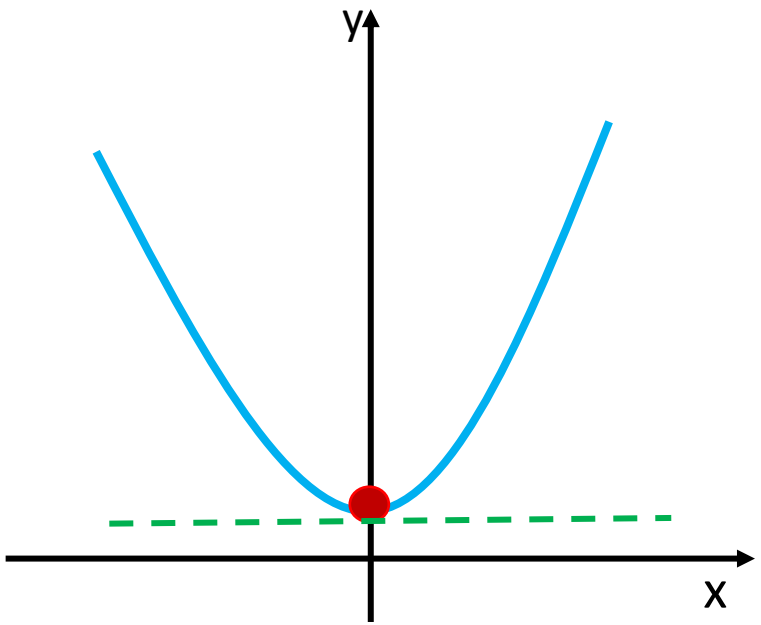


Função quadrática

$$(\hat{y}_i - y_i)^2$$

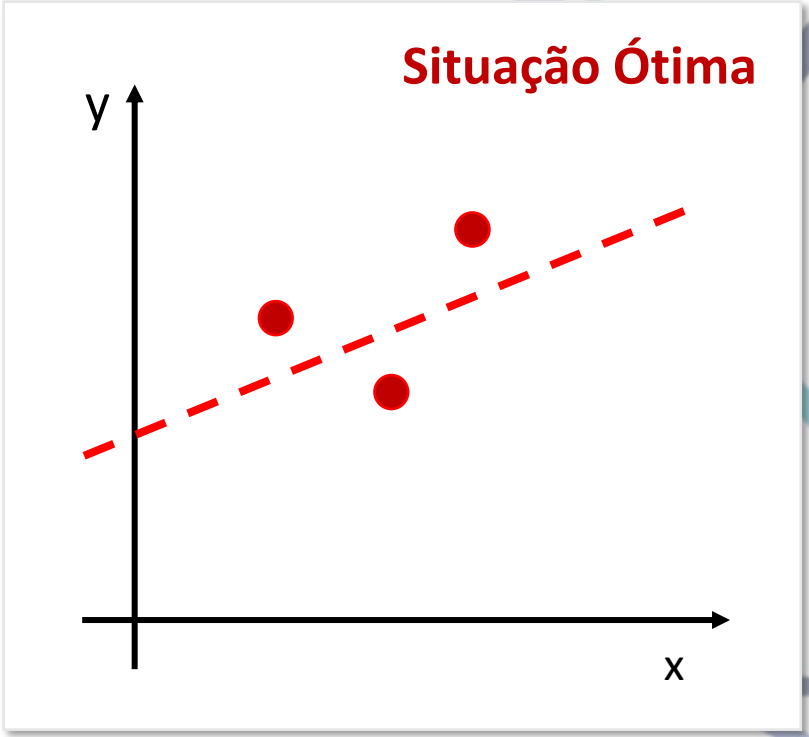


Rede Neural Artificial: Otimização



Função quadrática

$$(\hat{y}_i - y_i)^2$$



Ministério da Ciência, Tecnologia e Inovações

Centro Brasileiro de Pesquisas Físicas

Métodos para Análise de grande volume de dados e Astroinformática

Clécio Roque De Bom – debom@cbpf.br

CBPF

clearnightsrthebest.com