

## *Machine Learning*



## Redes neurais artificiais e introdução ao *Deep Learning*

Bloco 1

Lucas Claudino



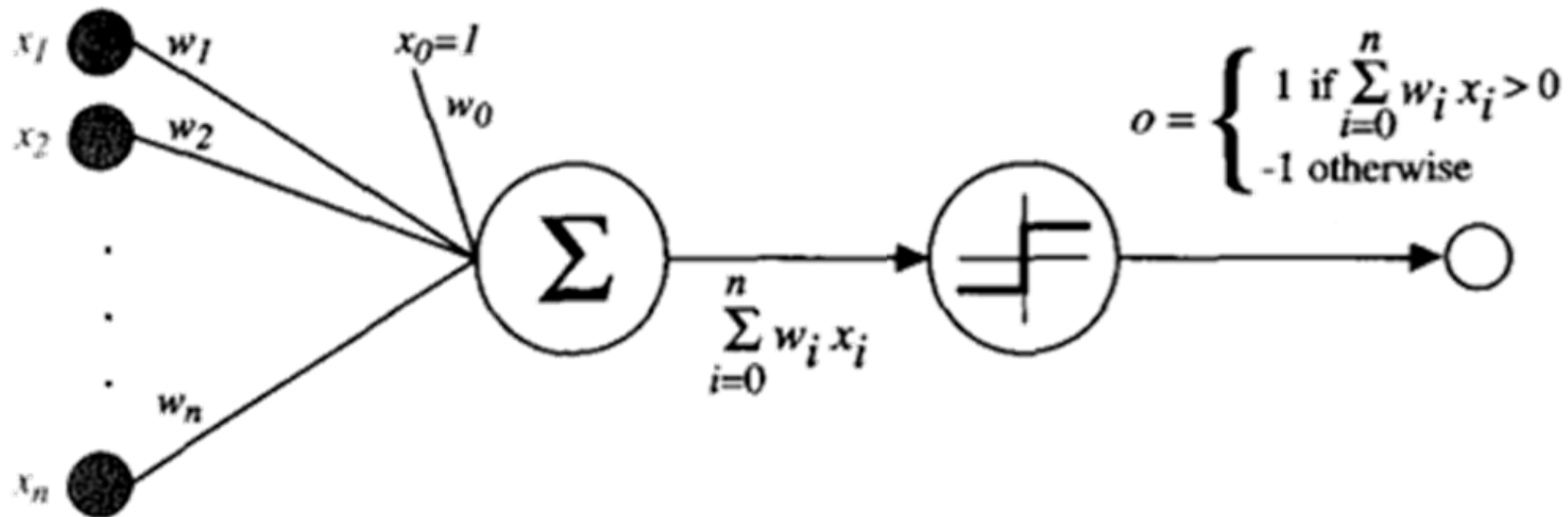


## RNA

- Baseada na forma de trabalho do sistema nervoso.
- Neurônios (nós) carregam informações.
- Pesos para cada nó.
- Camadas de uma RNA:
  - Entrada.
  - Oculta.
  - Saída.

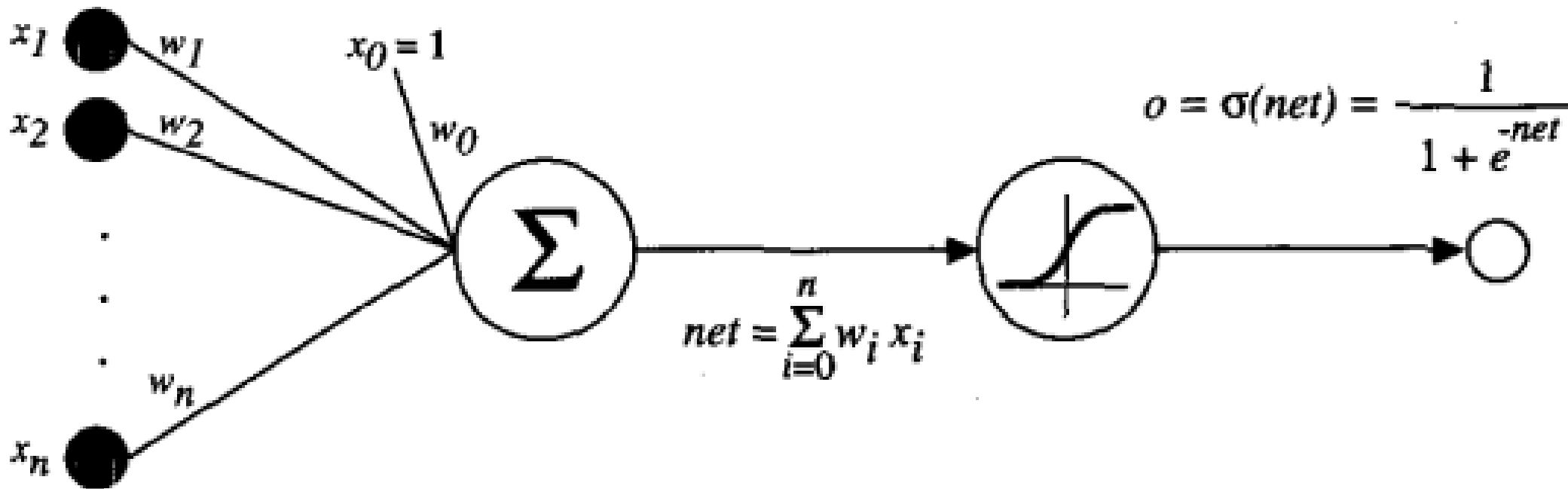
## ► RNA *perceptron*

Figura 1 – Rede neural artificial do tipo *perceptron*



## ► RNA com função sigmoide como ativação

Figura 2 – Rede Neural Artificial com uso de cálculo de limiar através da função sigmoide

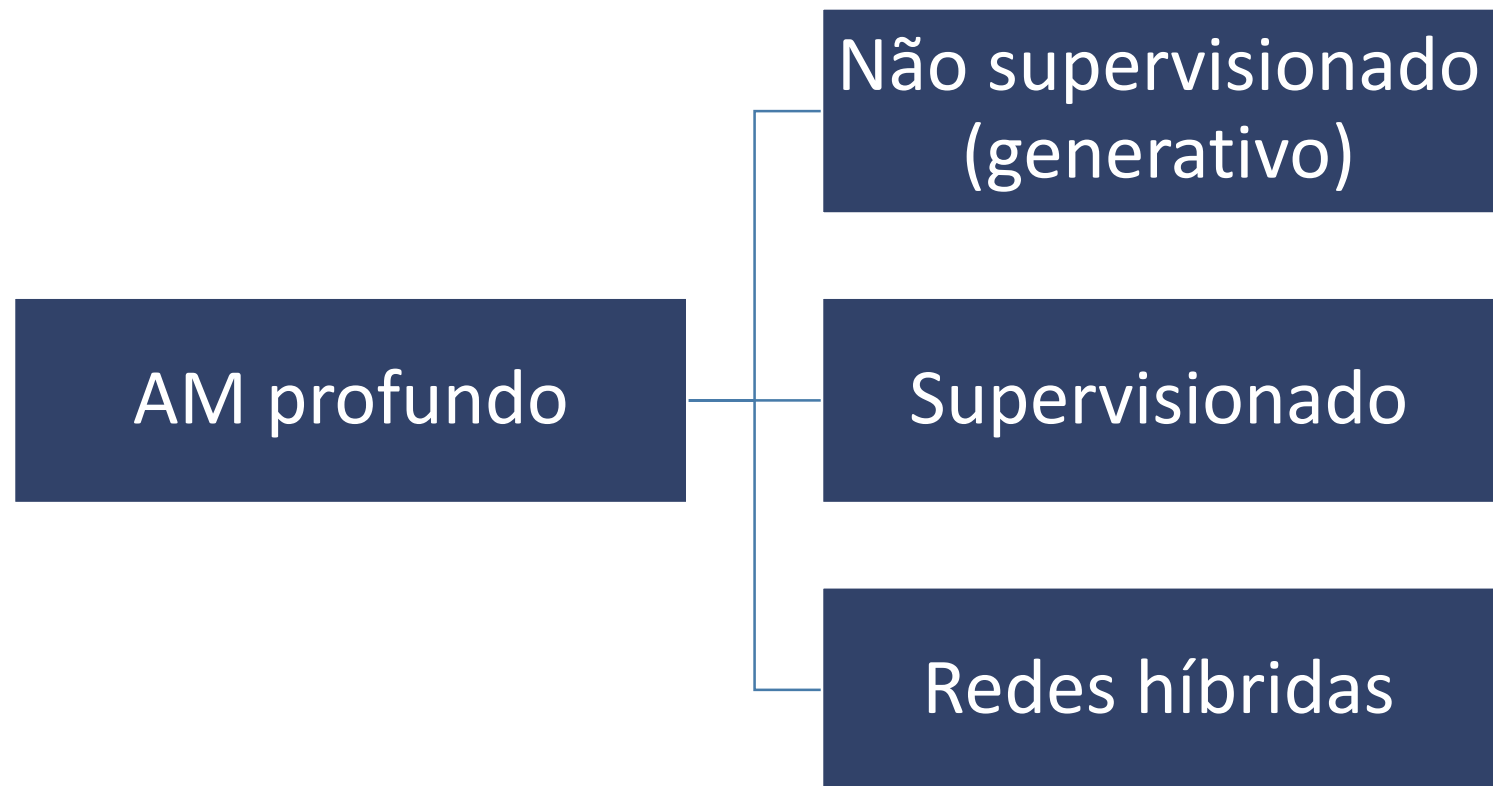




## ► *Deep Learning*

- Atualmente: muita informação (sensores, câmeras etc.).
- *Deep Learning* analisa e extrai padrões.
- Rede neural densa (profunda): muitas camadas ocultas.
  - Redes Neurais Convolucionais.
  - Redes Neurais Recorrentes.
  - Redes Neurais Recursivas.

## ► *Deep Learning*





## ► Aplicações de *deep learning*

- Coloração de imagens preto e brancas.
- Tradução automática.
- Classificação e detecção de objetos em fotografias.
- Jogadores automáticos (*bots*).
- Chat *bots*.



## RNA: treinamento e exemplo

Bloco 2

Lucas Claudino



## ► Treinamento – determinação de pesos

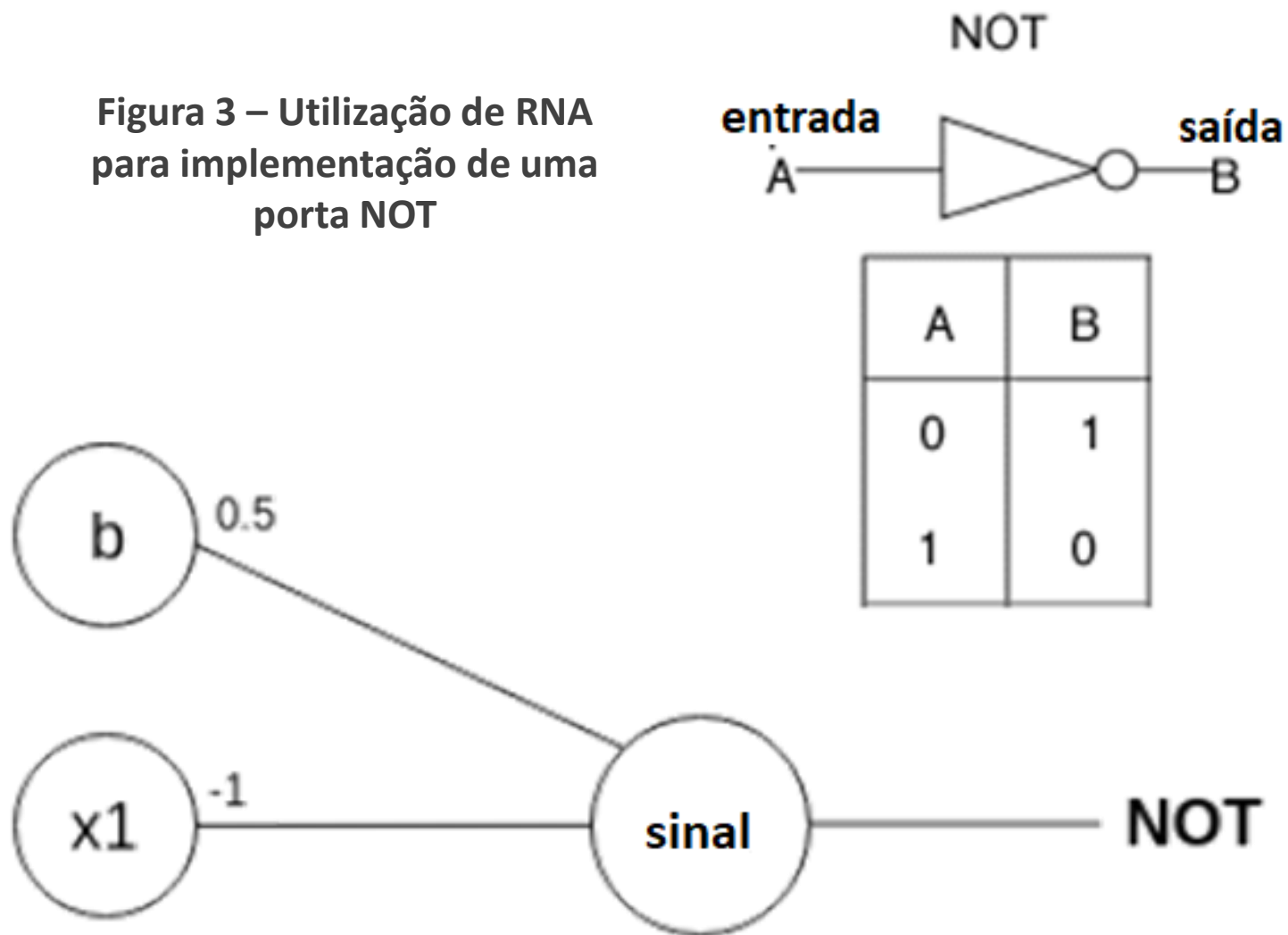
$$w_i \leftarrow w_i + \Delta w_i$$
$$\Delta w_i = \eta(t - o)x_i$$

Variação de peso indica em qual direção o algoritmo deve andar.

É relacionado ao gradiente da função erro (minimizar o erro).

## ► Exemplo: porta NOT

Figura 3 – Utilização de RNA para implementação de uma porta NOT



Fonte: adaptada de Dukor (2018).

# PÓS-GRADUAÇÃO

## Teoria em prática

### Bloco 3

Lucas Claudino



## ► Teoria em prática: aplicação de RNA

Quadro 1 -amostras para o  
treinamento da rede neural  
*perceptron*

X1	X2	t
0	0	-1
0	1	-1
1	0	-1
1	1	1

Fonte: elaborada pelo autor.

Quadro 2 -amostras para o  
treinamento da rede neural  
*perceptron (2)*

X1	X2	X3	t
0,1	0,4	0,7	1
0,3	0,7	0,2	-1
0,6	0,9	0,8	-1
0,5	0,7	0,1	1

Fonte: elaborada pelo autor.

## ► Passos a seguir

- Definir a matriz de dados a ser utilizada.
- Iniciar os pesos com zero ou um valor aleatório.
- Escolher uma taxa de aprendizagem.
- Calcular a saída a partir dos pesos atuais.
- Enquanto ainda houver diferença entre a saída obtida e a saída desejada, atualizar os pesos com a regra já estudada:

$$w_{i+1} = w_i + \Delta w_i = w_i + \eta(t - o)x_i$$

Figura 4 – Exemplo de implementação de RNA

```
for i = 1:n1,  
    erro=1;  
    while erro == 1, % enquanto erro for 1 faça  
        u = amostra(i,:) * w';  
        if (u >= 0)  
            y = 1;  
        else  
            y = -1; end  
        if (y ~= target(i,1))  
            r = n * (target(i,1) - y) * amostra(i,:);  
            w = w + r;  
            epoca = epoca + 1;  
            erro = 1;  
        else  
            erro = 0;  
            epoca = epoca + 1; end end  
    i = i + 1; end
```

Figura 5 – Exemplo de implementação de RNA (2)

```
% Operação
file = 'dados1.xlsx';
dados = xlsread(file) % lê os dados da tabela excel
y = size(dados); % pega o tamanho da matriz Amostras
y1= y(1,1);
y2 =y(1,2);
i = 1;
dado = ones(y1,y2+1);
dado(:,1)= -1;
dado(:,2:end)= dados(:,1:end);
for i=1:y1,
    u = dado(i,:)*w';
    if (u >= 0)
        y = 1
    else
        y = -1end
    i = i+1;end
```



## Dica do professor

Bloco 4

Lucas Claudino





## ► Dica do professor

- Decida entre treinar sua RNA ou usar uma RNA feita.
  - Dados do tipo ImageNet -> VGG net, ResNet, DenseNet.
- Sempre utilize um layer de normalização.
- Restrinja os valores possíveis dos pesos.
- Embaralhe os dados **antes** e **durante** o treinamento.
- Treinamento requer muito processamento: use o GoogleColab (processamento em nuvem).



## ► Referências

FACELI, K. *et al.* **Inteligência artificial**: uma abordagem de aprendizado de máquina. São Paulo: LTC Editora, 2011.

MITCHEL, M. B. **Machine Learning**. New York: McGraw-Hill Science/Engineering/Math, 1997.

DUKOR, Stanley O. Neural Representation of AND, OR, NOT, XOR and XNOR Logic Gates (Perceptron Algorithm). Medium, [s.l.], 13 de novembro de 2018. Disponível em: <https://medium.com/@stanleydukor/neural-representation-of-and-or-not-xor-and-xnor-logic-gates-perceptron-algorithm-b0275375fea1>. Acesso em: 13 set. 2019.

