

0 Perceptron



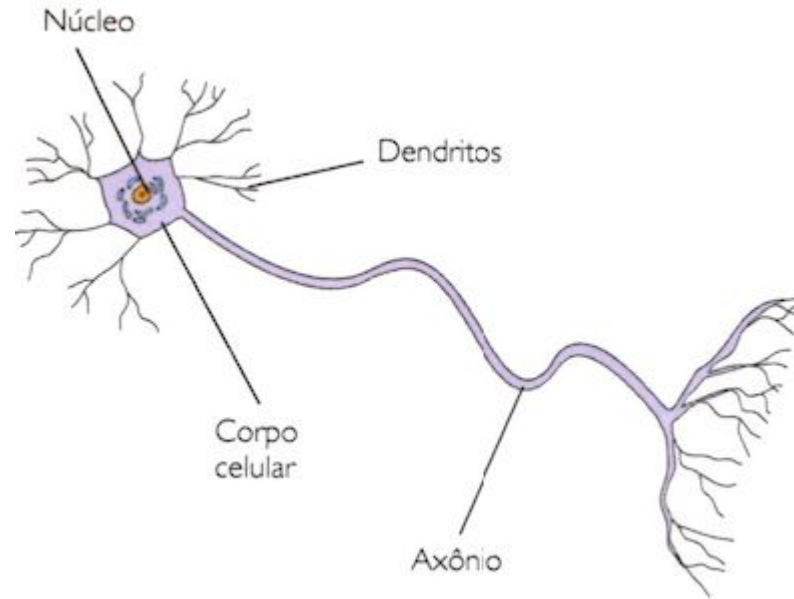
CIC260 - INTELIGÊNCIA ARTIFICIAL
Isabela Neves Drummond
Flávio Belizário da Silva Mota

Neurônios Biológicos

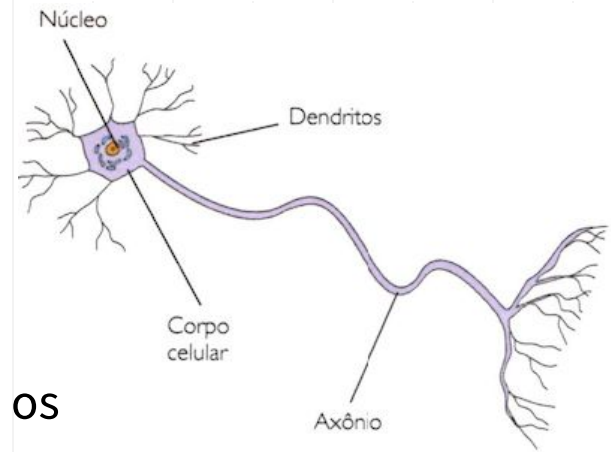
- O que é um neurônio?
- O neurônio é a unidade fundamental do sistema nervoso. Ele se distingue das outras células por apresentar excitabilidade, o que faz com que ele responda a estímulos internos e externos.



Neurônios Biológicos

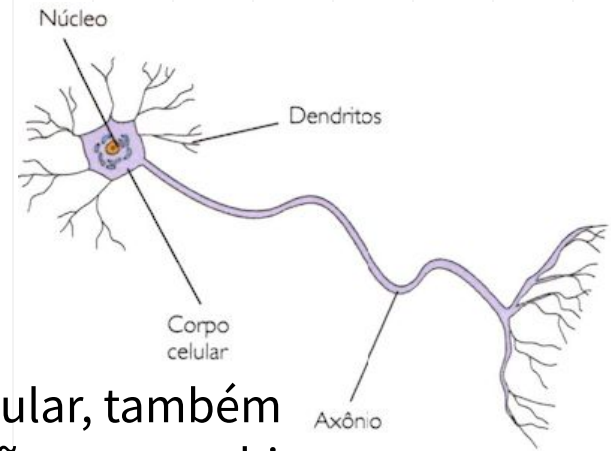


Neurônios Biológicos



- Os componentes principais de um neurônio são os dendritos, corpo celular e axônio.
- Os dendritos são a parte do neurônio responsáveis por receber os estímulos nervosos de outros neurônios.

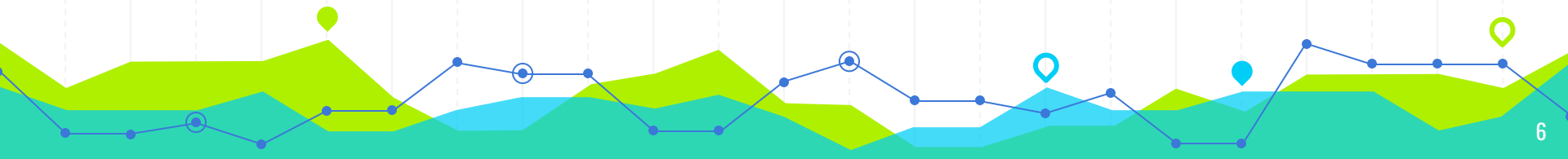
Neurônios Biológicos



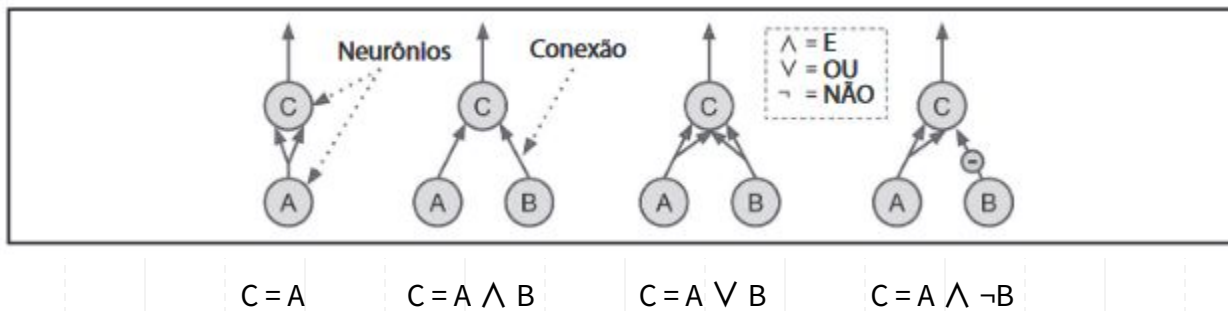
- Esses estímulos são transmitidos para o corpo celular, também chamado de soma. O soma coleta essas informações, as combina e processa. De acordo com a intensidade desse estímulo, um novo impulso é gerado e enviado para o axônio.
- O axônio é um prolongamento do neurônio que conduz os estímulos elétricos produzidos no soma até suas terminações que, usualmente, são outros neurônios.

Neurônio Artificial

- Proposto em 1943 por McCulloch e Walter Pitts
- Possui uma ou mais entradas binárias (ligado/desligado) e uma saída também binária
- Esse modelo ativa sua saída quando um certo número de suas entradas está ativa
- O modelo foi capaz de demonstrar na época redes de neurônios que calculavam operações lógicas



Neurônio Artificial

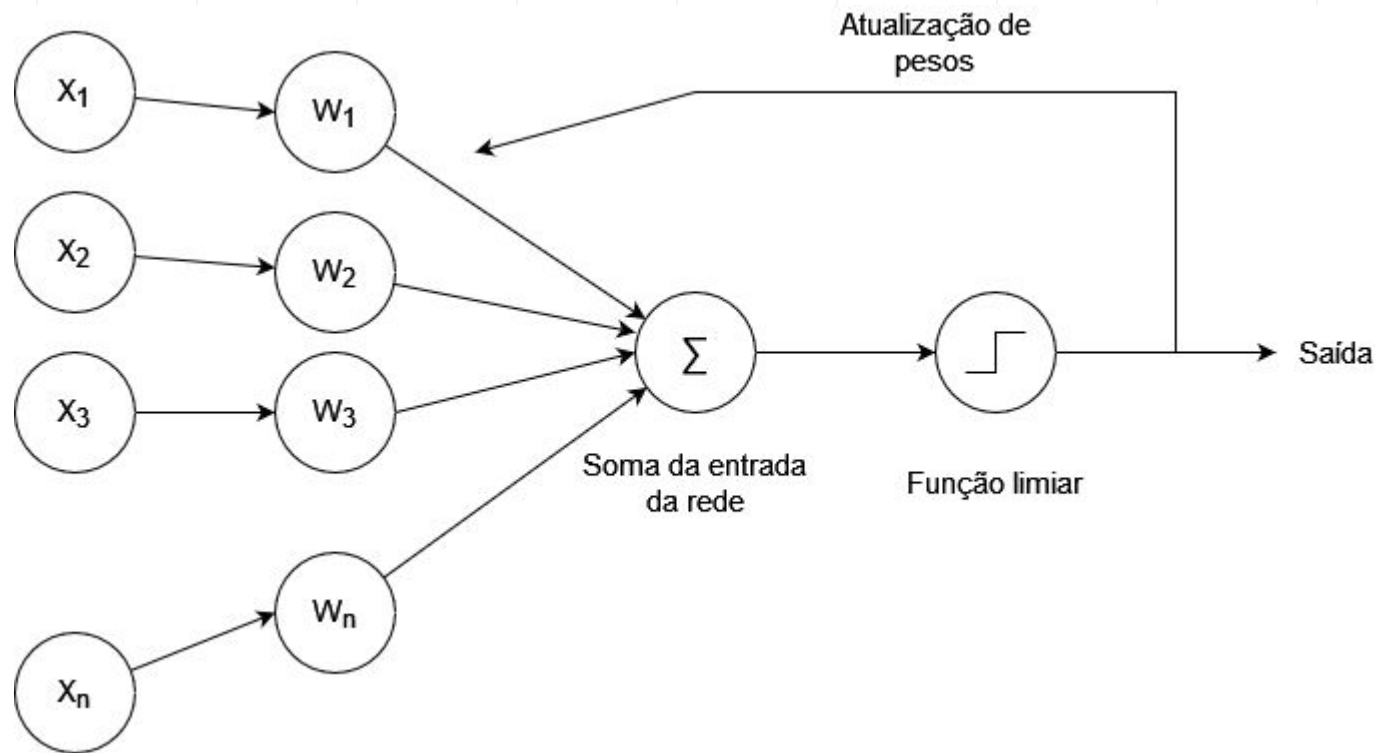


0 Perceptron

- Criado por Frank Rosenblatt (1958)
- Considerado a arquitetura mais simples de uma Rede Neural Artificial
- As entradas (X) e saídas são números e cada conexão está associada a um peso (W)
- Existe uma função de soma ponderada das entradas e a aplicação de uma função limiar



Neurônio Artificial



0 Perceptron

- A entrada da rede é dada pelo somatório dos valores das entradas multiplicados pelos pesos de cada conexão:

$$\Sigma = X_1.W_1 + X_2.W_2 + X_3.W_3 + \dots + X_n.W_n$$



0 Perceptron

- A função limiar leva em consideração um limite (t) e produz uma saída em função do valor resultante da soma da entrada da rede, como por exemplo:
 - 1 se $\sum \geq t$
 - -1 se $\sum < t$



Como o Perceptron aprende?

- O Perceptron é um algoritmo de aprendizado supervisionado, ou seja, para que ele aprenda é necessário saber os valores de saída esperados
- Durante o aprendizado, ou treinamento, para uma determinada entrada de dados, os pesos do Perceptron são ajustados conforme o erro produzido na saída do modelo



Como o Perceptron aprende?

- Um peso é ajustado seguindo a equação:

$$W_n = W_n + \eta.(y_n - \hat{y}_n).X_n$$

- Onde:

- η é a taxa de aprendizado
- y_n é a saída esperada para entrada n
- \hat{y}_n é a saída prevista pelo modelo para a entrada n
- X_n é a própria entrada



Um exemplo numérico

X_1	X_2	y
1	0	0
1	1	1

Considerando:

$W_1 = 0,5$ e $W_2 = 0,2$

$\eta = 0,1; t = 0$

$X_1 = 1; X_1 = 0; y = 0$

$\Sigma = 1.0,5 + 0.0,2 = 0,5 \rightarrow \hat{y} = 1$

$y - \hat{y} = 0 - 1 = -1$

$W_1 = 0,5 + 0,1 \cdot (-1) \cdot 1 = 0,4$

$W_2 = 0,2 + 0,1 \cdot (-1) \cdot 0 = 0,2$

$X_1 = 1; X_1 = 1; y = 1$

$\Sigma = 1.0,4 + 1.0,2 = 0,6 \rightarrow \hat{y} = 1$

$y - \hat{y} = 1 - 1 = 0$

$W_1 = 0,4$

$W_2 = 0,2$

$X_1 = 1; X_1 = 0; y = 0$

$\Sigma = 1.0,4 + 0.0,2 = 0,4 \rightarrow \hat{y} = 1$

$y - \hat{y} = 0 - 1 = -1$

$W_1 = 0,4 + 0,1 \cdot (-1) \cdot 1 = 0,3$

$W_2 = 0,2 + 0,1 \cdot (-1) \cdot 0 = 0,2$

0 Perceptron

- A atualização dos pesos pode continuar até que o erro seja igual a 0 para todos os exemplos ou até que um número de iterações (épocas) seja atingido
- A convergência do modelo só acontece caso as duas classes que queremos encontrar sejam linearmente separáveis e a taxa de aprendizado seja suficientemente pequena.



Prática - Google Colab

colab



Implementando um Perceptron



Exercício

Utilizando o módulo *make_blobs*, gere dois novos conjuntos alterando o parâmetro *random_state* para o valor 6 e 30.

Varie o número de épocas do Perceptron em 10, 50 e 100 e a taxa de aprendizado em 0.02 e 0.003.

Para cada base gerada e cada configuração de Perceptron, mostre o gráfico das atualizações por época e a fronteira de decisão gerada, explicando o comportamento do modelo em função da convergência.



Referências

1. LUGER, G. **Inteligência artificial**. 6. ed. Pearson, 2013.
2. RASCHKA, S.; MIRJALILI, V. **Python Machine Learning**. 2. ed. Packt, 2017.
3. FACELI, K. et al. **Inteligência Artificial: uma abordagem de aprendizado de máquina**. 2. ed. Rio de Janeiro: Editora LTC, 2021.
4. GÉRON, A. **Mãos à Obra - Aprendizado de Máquina com Scikit-Learn e TensorFlow**. S.l.: Alta Books, 2019.