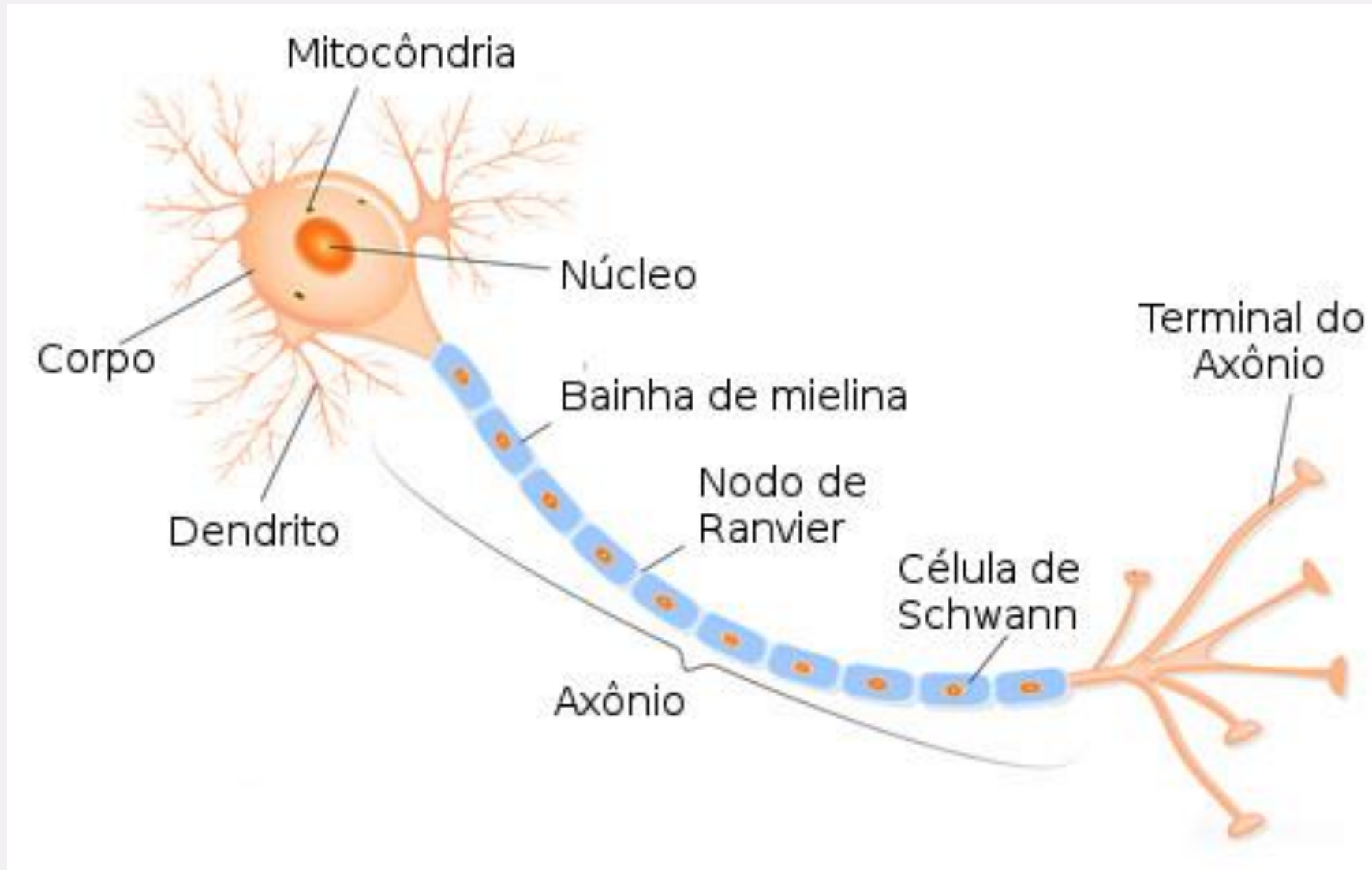


The background features a complex, three-dimensional molecular structure composed of dark, reflective spheres connected by thin, metallic-looking rods. The structure is partially obscured by a semi-transparent white rectangular area containing text. Scattered throughout the scene are several small, solid-colored circles in shades of green, blue, and orange, some of which are larger than others, creating a sense of depth and movement.

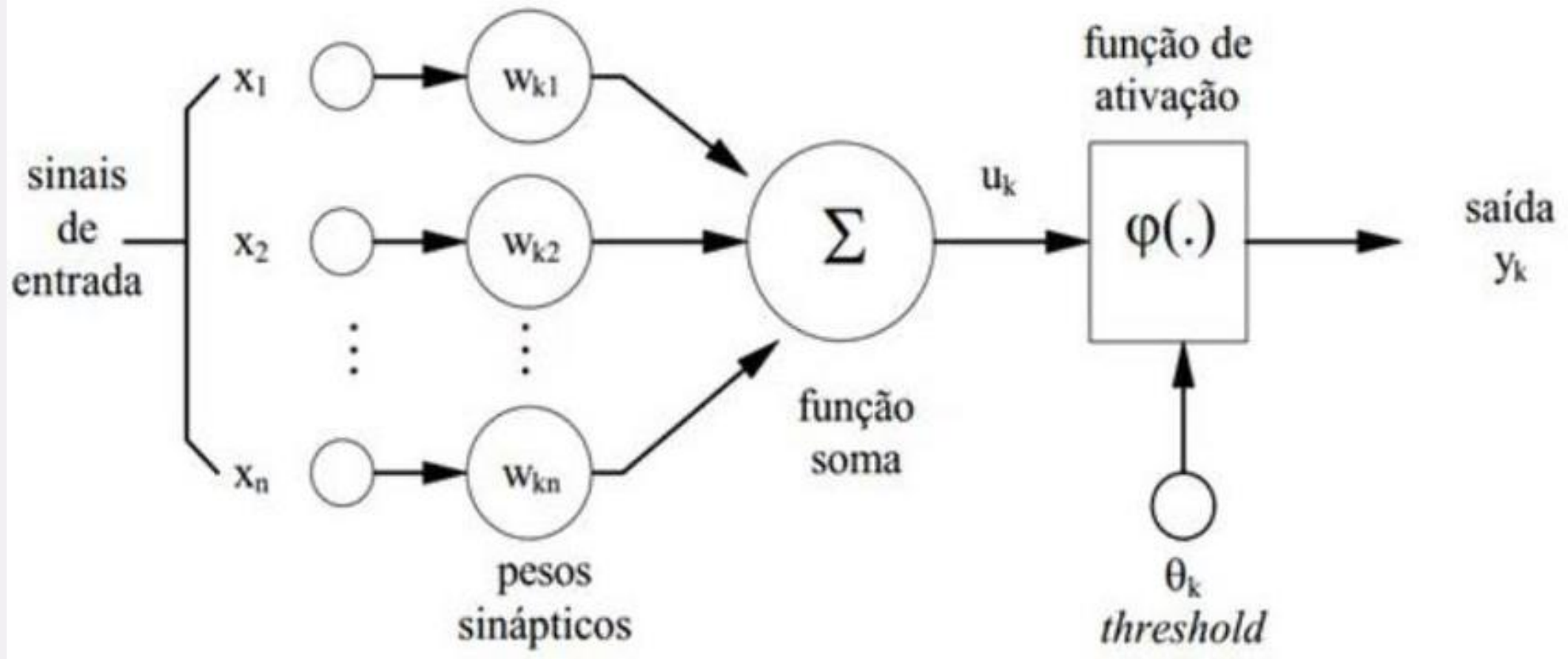
Redes Neurais Artificiais

Prof. Anderson Vanin

Neurônio Físico



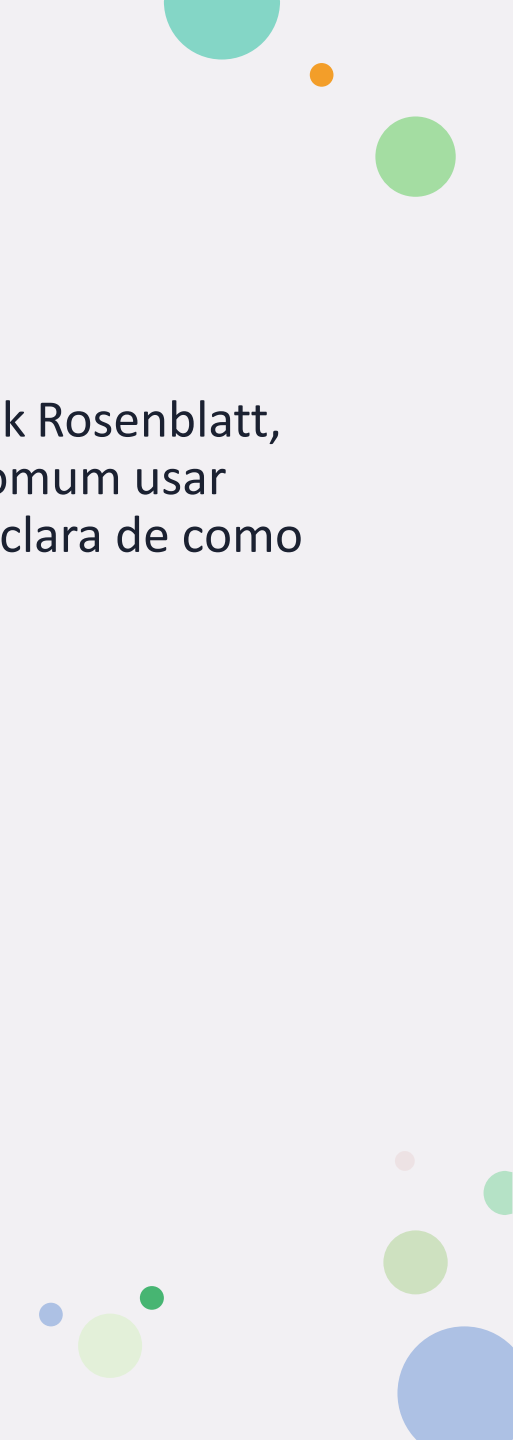
Neurônio Artificial



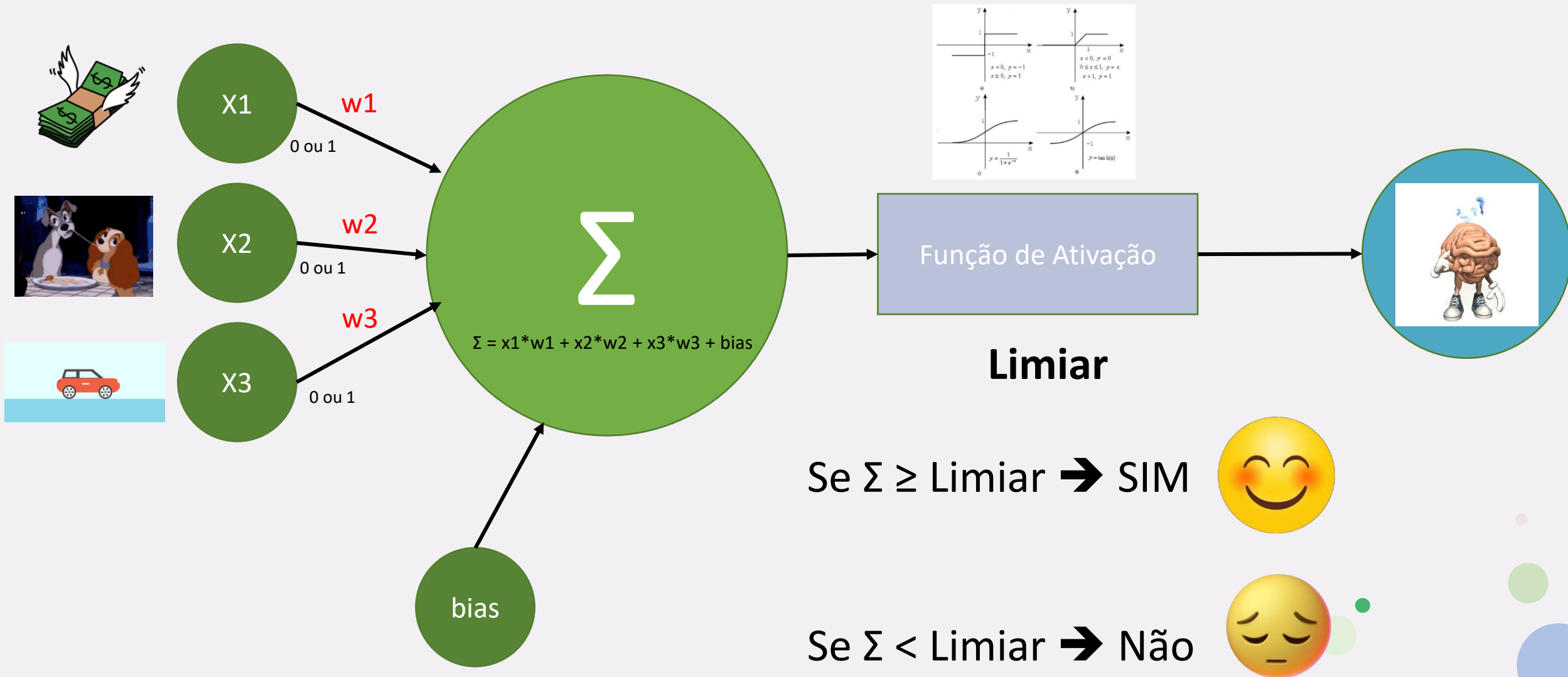


Perceptron

O Modelo Perceptron foi desenvolvido nas décadas de 1950 e 1960 pelo cientista Frank Rosenblatt, inspirado em trabalhos anteriores de Warren McCulloch e Walter Pitts. Hoje, é mais comum usar outros modelos de neurônios artificiais, mas o Perceptron permite uma compreensão clara de como funciona uma rede neural em termos matemáticos, sendo uma excelente introdução.



Perceptron - Funcionamento



Exemplo - 01 Prático

Imagine que você tenha um conjunto de dados e gostaria de saber se uma determinada pessoa com base em suas características irá receber ou não um seguro.

Exemplo prático no colab:

https://github.com/profandersonvanin01/machine_learning/blob/main/01_Perceptron.ipynb

Problema

Suponha que o fim de semana esteja chegando e você ouviu falar que haverá um festival em sua cidade. Você adora ir a festivais e está tentando decidir se deve ou não ir ao festival. Você pode tomar sua decisão pesando três fatores:

- O tempo está bom?
- Seu namorado ou namorada quer acompanhá-lo(a)?
- O festival está perto de transporte público? (Você não possui um carro)

Podemos representar estes três fatores pelas variáveis binárias correspondentes x_1 , x_2 e x_3 . Por exemplo, teríamos $x_1 = 1$ se o tempo estiver bom e $x_1 = 0$ se o tempo estiver ruim. Da mesma forma, $x_2 = 1$ se seu namorado ou namorada quiser ir ao festival com você, e $x_2 = 0$, se não. E similarmente para x_3 e transporte público.

Problema

Agora, suponha que você esteja muito disposto a ir ao festival, mesmo que seu namorado ou namorada não esteja interessado e o festival fica em um lugar de difícil acesso e sem transporte público amplamente disponível. Além disso, você realmente detesta mau tempo, e não há como ir ao festival se o tempo estiver ruim. Você pode usar Perceptrons para modelar esse tipo de tomada de decisão.

Uma maneira de fazer isso é escolher um peso $w_1 = 6$ para o tempo e $w_2 = 2$ e $w_3 = 2$ para as outras condições. O valor maior de w_1 indica que o tempo é muito importante para você, muito mais do que se seu namorado ou namorada vai acompanhá-lo(a) ou se o festival é próximo do transporte público. Finalmente, suponha que você escolha um (limiar) threshold de 5 para o Perceptron. Com essas escolhas, o Perceptron implementa o modelo de tomada de decisão desejado, produzindo 1 sempre que o tempo estiver bom e 0 sempre que o tempo estiver ruim. Não faz diferença para o resultado se seu namorado ou namorada quer ir, ou se o transporte público está acessível.

Exemplo - 02 Prático

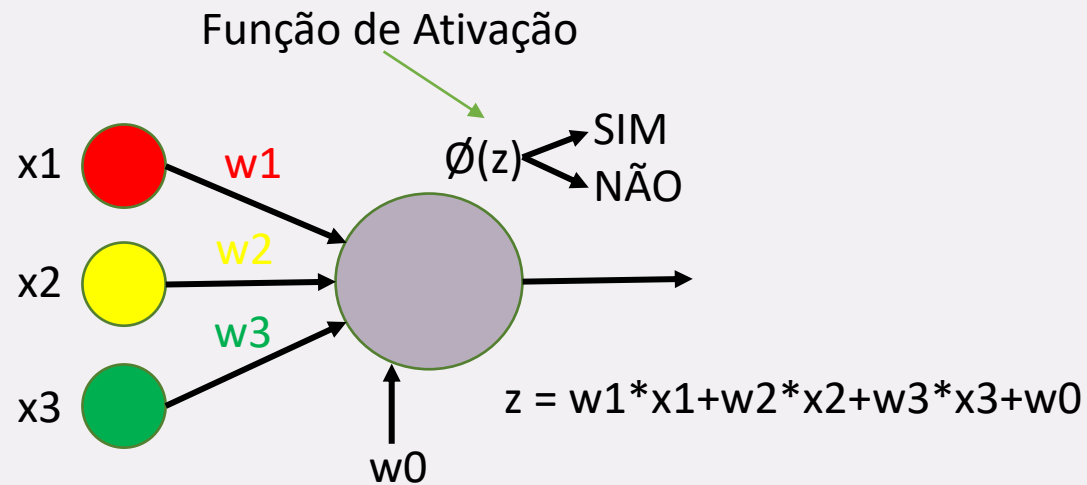
x1 = longe?

x2 = caro?

x3 = amigos?

x1	x2	x3	y
0	0	1	1
1	0	1	1
1	1	1	0
1	1	0	0

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

Para começar inicializamos os dados com valores aleatórios para os pesos.

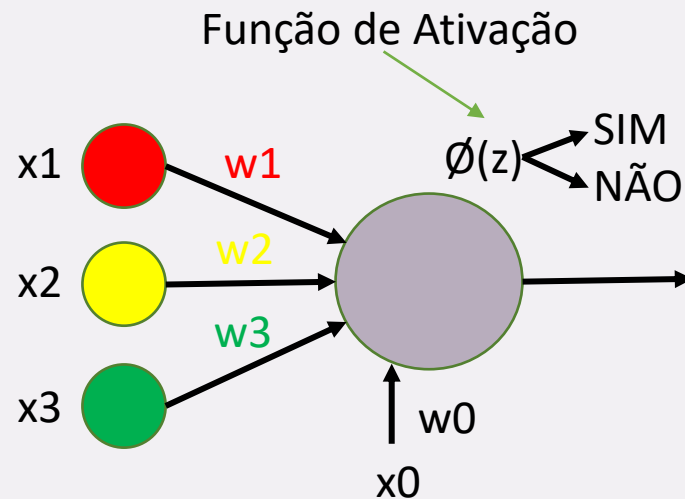
Ou seja:

$$w1 = 0, w2 = 0, w3 = 0 \text{ e } w0 = 0$$

Na sequência, verificamos se a rede neural acerta os valores de **y** que já conhecemos com base nas entradas x1, x2 e x3.

Para isso, começamos pegando a Amostra 1 e verificando o resultado previsto com o conhecido.

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



Para o bias, X0, será sempre iniciado com valor 1

$$z = w1*x1 + w2*x2 + w3*x3 + w0$$

$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

Para a Amostra 1:

$$z = 0 * 0 + 0 * 0 + 0 * 1 + 0 \rightarrow z = 0$$

$\phi(0) = 0$, resposta **NÃO**

Repare na tabela que a resposta correta seria 1 (SIM) mas a rede informou 0 (NÃO), então nesse caso é considerado um erro e a rede deve ajustar os pesos para corrigir este erro.

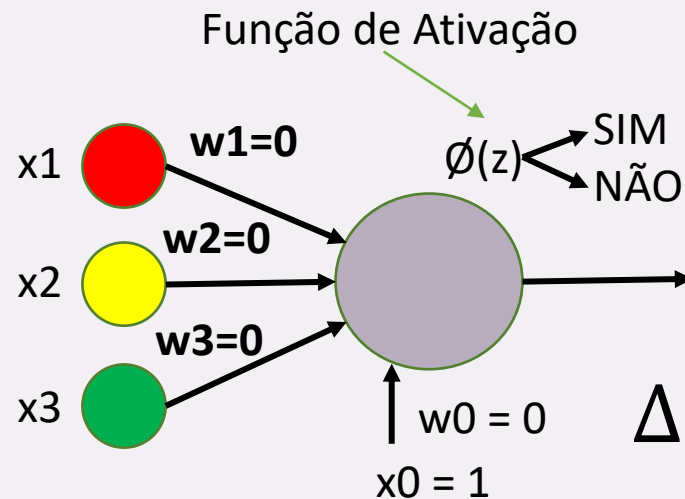
$$w_j = w_j + \Delta w_j \quad \Delta w_j = \eta * (y^i - y'^i) * x_j$$

$\eta \rightarrow$ taxa de aprendizado
(valores pequenos diferentes de zero geralmente entre 0,1 e 0,5)

$\eta \rightarrow$ taxa de aprendizado. Valores grandes fazem sua rede ficar "BIPOLAR"



amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w1 * x1 + w2 * x2 + w3 * x3 + w0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Para a Amostra 1:

$$z = 0 * 0 + 0 * 0 + 0 * 1 + 0 \rightarrow z = 0$$

$\phi(0) = 0$, resposta **NÃO**

$$\Delta w1 = \eta * (\text{real} - \text{previsto}) * x1$$

$$\Delta w1 = 0, 1 * (1 - 0) * 0 = 0$$

$$w1 = w1 + \Delta w1 = 0 + 0 = 0$$

$$\Delta w2 = 0, 1 * (1 - 0) * 0 = 0$$

$$w2 = w2 + \Delta w2 = 0 + 0 = 0$$

$$\Delta w3 = 0, 1 * (1 - 0) * 1 = 0, 1$$

$$w3 = w3 + \Delta w3 = 0 + 0, 1 = 0, 1$$

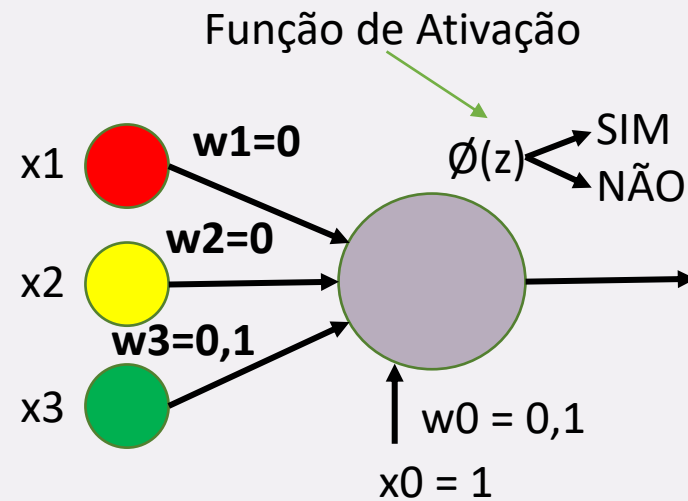
$$\Delta w0 = 0, 1 * (1 - 0) * 1 = 0, 1$$

$$w0 = w0 + \Delta w0 = 0 + 0, 1 = 0, 1$$

Agora a rede ajustará os novos pesos $w1$, $w2$, $w3$ e $w0$.

Agora esses novos valores dos pesos, são aplicados e a nova amostra (2) é passada pela rede com estes novos pesos.

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w1*x1 + w2*x2 + w3*x3 + w0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Para a Amostra 2:

$$z = 0 * 1 + 0 * 0 + 0,1 * 1 + 0,1 \rightarrow z = 0,2$$

$\phi(0,2) = 1$, resposta **SIM** (Aqui a rede acertou! – Não ajusta os pesos! Permanecem os já calculados)

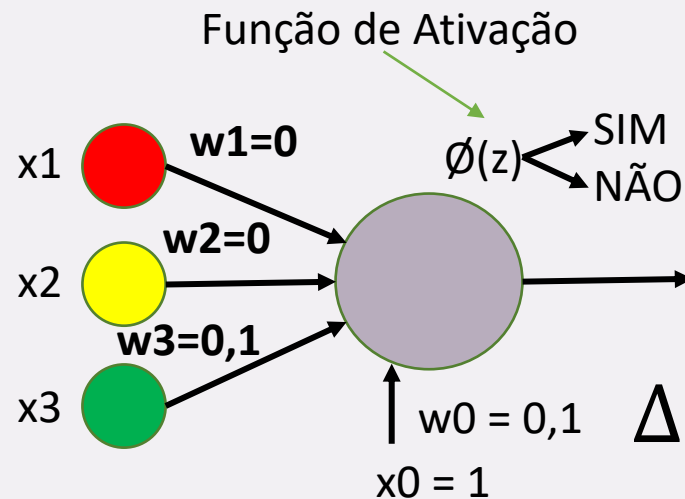
$$w1 = 0$$

$$w2 = 0$$

$$w3 = 0,1$$

$$w0 = 0,1$$

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 + w_0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Para a Amostra 3:

$$z = 0 * 1 + 0 * 1 + 0,1 * 1 + 0,1 \rightarrow z = 0,2$$

$\phi(0,2) = 1$, resposta **SIM** (A rede errou! Ajustar os pesos)

$$\Delta w_1 = \eta * (\text{real} - \text{previsto}) * x_1$$

$$\Delta w_1 = 0,1 * (0 - 1) * 1 = -0,1$$

$$w_1 = w_1 + \Delta w_1 = 0 + (-0,1) = -0,1$$

$$\Delta w_2 = 0,1 * (0 - 1) * 1 = -0,1$$

$$w_2 = w_2 + \Delta w_2 = 0 + (-0,1) = -0,1$$

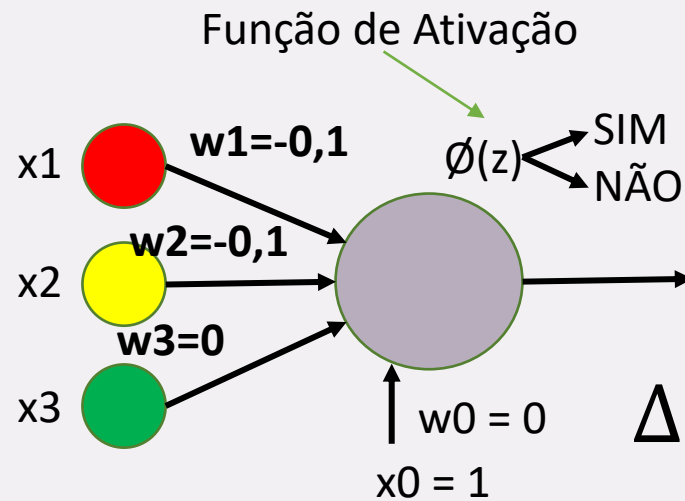
$$\Delta w_3 = 0,1 * (0 - 1) * 1 = -0,1$$

$$w_3 = w_3 + \Delta w_3 = 0,1 + (-0,1) = 0$$

$$\Delta w_0 = 0,1 * (0 - 1) * 1 = -0,1$$

$$w_0 = w_0 + \Delta w_0 = 0,1 + (-0,1) = 0$$

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w1 * x1 + w2 * x2 + w3 * x3 + w0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Para a Amostra 4:

$$z = (-0,1) * 1 + (-0,1) * 1 + 0 * 0 + 0 \rightarrow z = -0,2$$

$\phi(-0,2) = 0$, resposta **NÃO** (A rede **ACERTOU! NÃO Ajustar os pesos**)

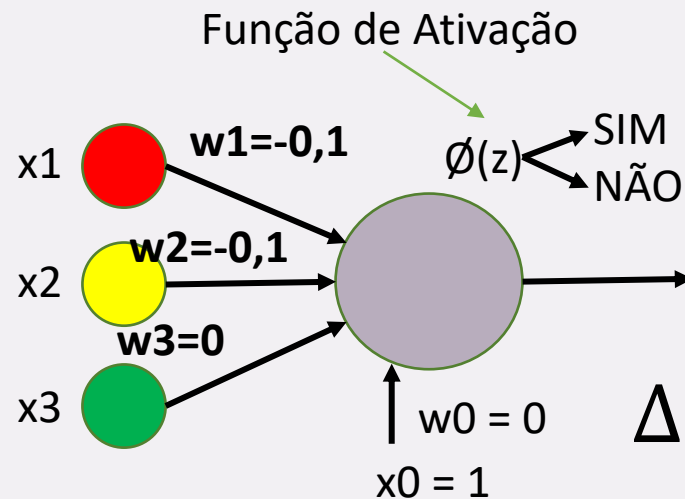
$$w1 = -0,1$$

$$w2 = -0,1$$

$$w3 = 0$$

$$w0 = 0$$

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w1 * x1 + w2 * x2 + w3 * x3 + w0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Retornando Para a Amostra 1:

$$z = (-0,1) * 0 + (-0,1) * 0 + 0 * 1 + 0 \rightarrow z = 0$$

$\phi(0) = 0$, resposta **NÃO** (A rede errou! Ajustar os pesos)

$$\Delta w1 = \eta * (\text{real} - \text{previsto}) * x1$$

$$\Delta w1 = 0,1 * (1 - 0) * 0 = 0$$

$$w1 = w1 + \Delta w1 = (-0,1) + 0 = -0,1$$

$$\Delta w2 = 0,1 * (1 - 0) * 0 = 0$$

$$w2 = w2 + \Delta w2 = (-0,1) + 0 = -0,1$$

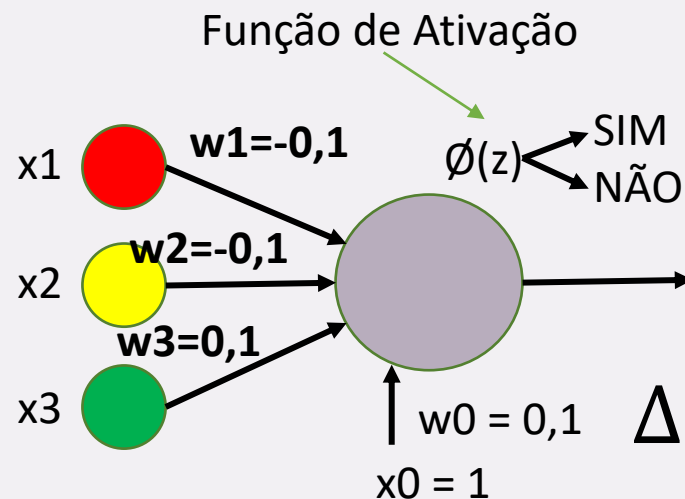
$$\Delta w3 = 0,1 * (1 - 0) * 1 = 0,1$$

$$w3 = w3 + \Delta w3 = 0 + 0,1 = 0,1$$

$$\Delta w0 = 0,1 * (1 - 0) * 1 = 0,1$$

$$w0 = w0 + \Delta w0 = 0 + 0,1 = 0,1$$

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 + w_0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Retornando Para a Amostra 2:

$$z = (-0,1) * 1 + (-0,1) * 0 + 0,1 * 1 + 0,1 \rightarrow z = 0,1$$

$\phi(0,1) = 1$, resposta **SIM** (A rede **ACERTOU!** NÃO Ajustar os pesos)

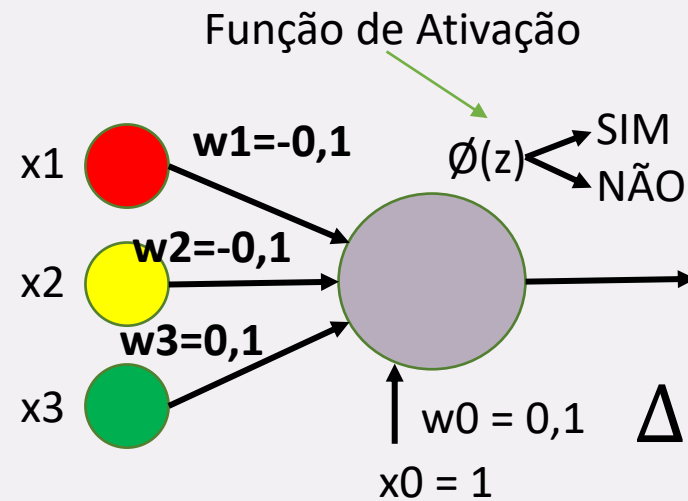
$$w_1 = -0,1$$

$$w_2 = -0,1$$

$$w_3 = 0,1$$

$$w_0 = 0,1$$

amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 + w_0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Retornando Para a Amostra 3:

$$z = (-0,1) * 1 + (-0,1) * 1 + 0,1 * 1 + 0,1 \rightarrow z = 0$$


$\phi(0) = 0$, resposta **NÃO** (A rede **ACERTOU!** NÃO Ajustar os pesos)

$$w_1 = -0,1$$

$$w_2 = -0,1$$

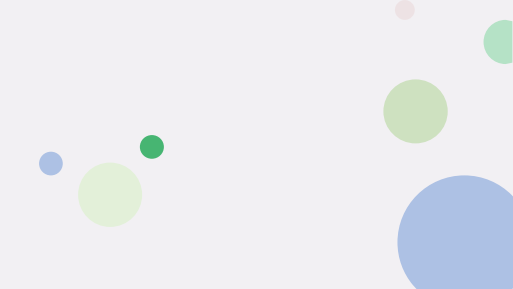
$$w_3 = 0,1$$

$$w_0 = 0,1$$

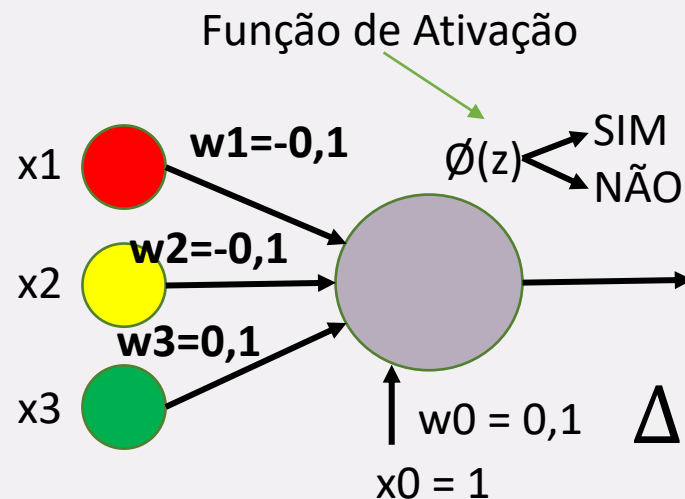


Neste ponto dizemos que a rede convergiu, ou seja, agora ela
APRENDEU com estes dados e irá acertar todos os valores
conhecidos!

Podemos fazer um outro teste, utilizando outros valores que a
rede nunca viu.



amostra	x1	x2	x3	y
1	0	0	1	1
2	1	0	1	1
3	1	1	1	0
4	1	1	0	0
5	0	0	0	?



$$\phi(z) = \begin{cases} 1, & \text{se } z > 0 \\ 0, & \text{se } z \leq 0 \end{cases}$$

$$z = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 + w_0$$

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta * (y^i - y'^i) * x_j$$

Testando um valor novo Amostra 5 para saber a resposta:

$$z = (-0,1) * 0 + (-0,1) * 0 + 0,1 * 0 + 0,1 \rightarrow z = 0,1$$

$\phi(0,1) = 1$, resposta **SIM**

(A rede PREVIU QUE COM ESTAS CONDIÇÕES A PESSOA VAI AO SHOW)

Concluimos que a rede foi coerente na resposta, pois mesmo que a pessoa não tenha amigos ($x_3 = 0$), o show era perto ($x_1 = 0$) e o show não era caro ($x_2 = 0$).

Exemplo - 02 Prático

Ir ao show ou não

Exemplo prático no colab:

[https://github.com/profandersonvanin01/machine_learning/blob/main/02 Perceptron Exemplo Show.ipynb](https://github.com/profandersonvanin01/machine_learning/blob/main/02%20Perceptron%20Exemplo%20Show.ipynb)