FIAP

FIAP

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

DISRUPTIVE ARCHITECTURES: IOT, IOB & IA

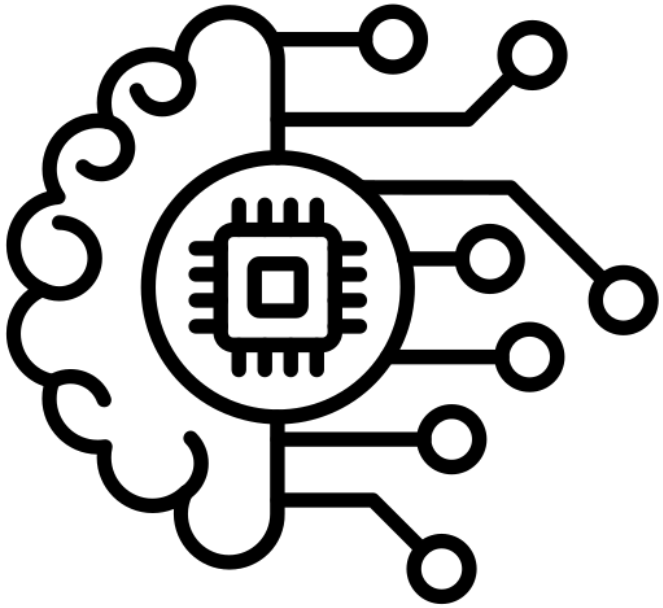
06 – Introdução a Redes Neurais Artificiais


Prof. Airton Y. C. Toyofuku


profairton.toyofuku@fiap.com.br

O que é uma Rede Neural Artificial?

- É um modelo computacional inspirado no funcionamento do cérebro humano;
- É uma técnica de Inteligência Artificial utlizada para tarefas de classificação, reconhecimento de padrões, previsão de valores, entre outras;
- É um conjunto de neurônios artificiais (nós), organizados em camadas, que processam informações por meio de operações matemáticas (funções de ativação);



Quando surgiu a Rede Neural Artificial?

FIAP

1943

- McCullock e Pitts
  - Primeira Rede Neural
  - Neurônio como processamento binário

1948

- N. Wiener
- Conceito de Cibernética

1949

- Donald O. Hebb
- Hipótese da alteração da força sináptica no comportamento
- Contribui na teoria das RNA

1957

- Neurocomputador Mark I Perceptron por Frank Rosenblatt e outros
- Primeiro algoritmo para ajustes de pesos

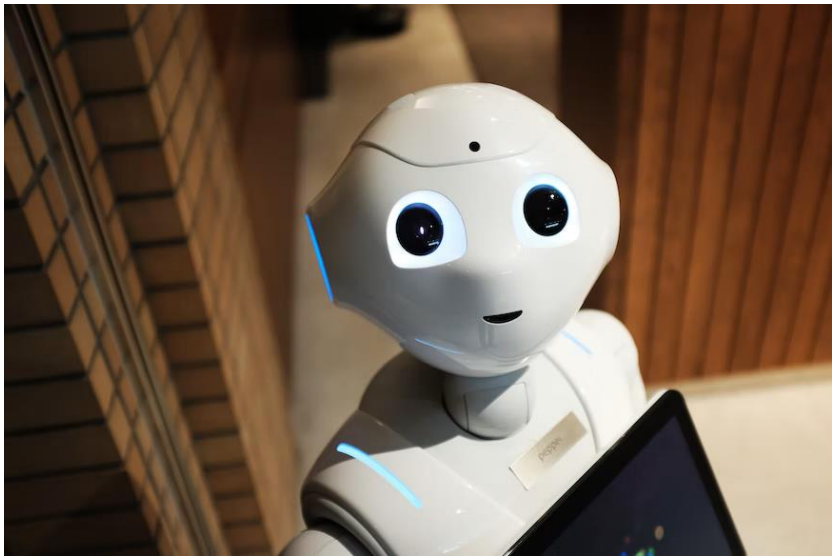
1969

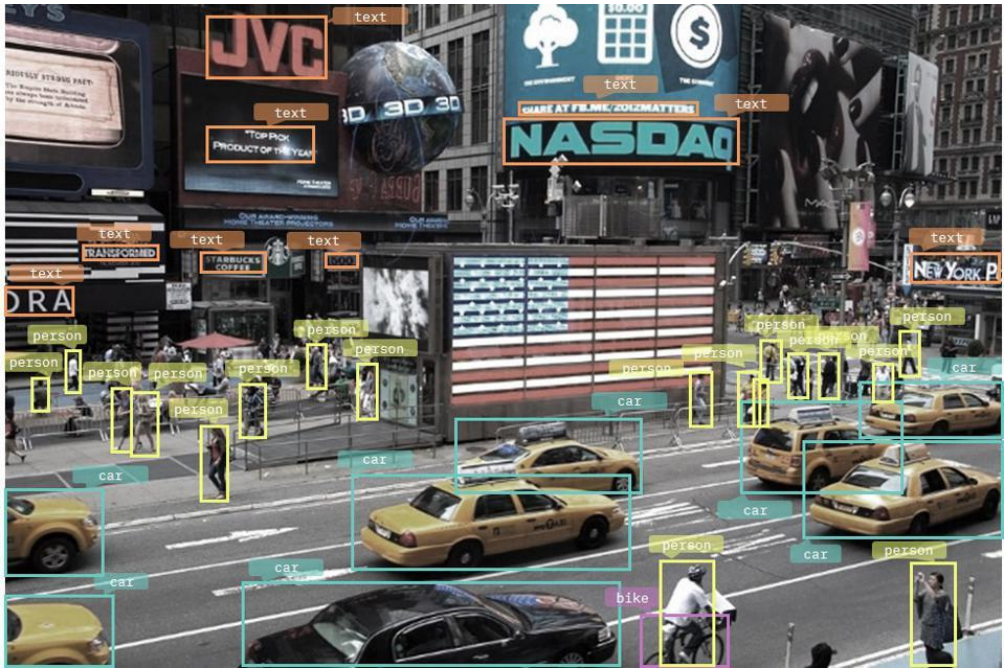
- L.M. Minsky e S.A. Papert
- Livro “Perceptrons”
- Impossibilidade de se implementar regras lógicas simples

1982

- Melhor conhecimento da estrutura do cérebro
- Maior poder computacional
- Novos algoritmos de aprendizado

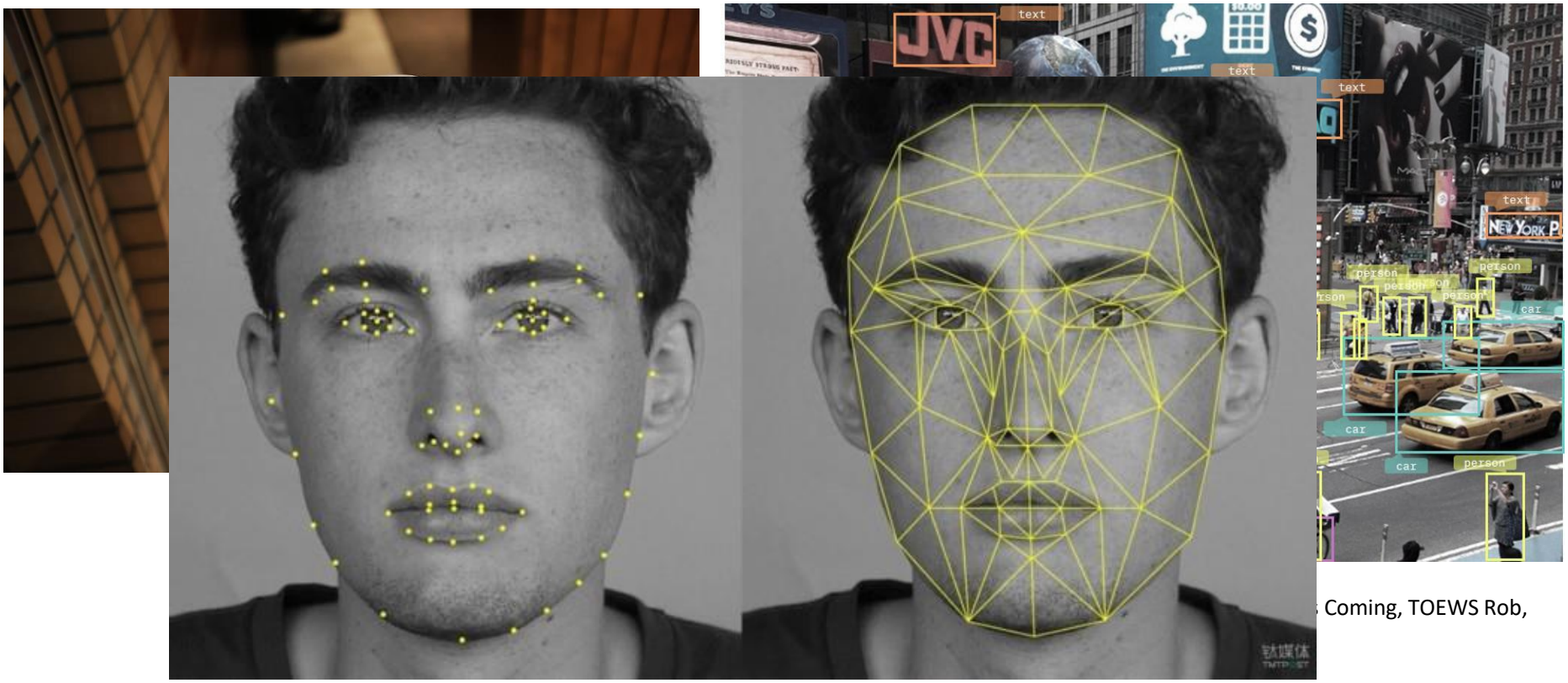
Onde podemos aplicar uma Rede Neural Artificial?





Fonte: “A Wave of Billion-Dollar Computer Vision Startups Is Coming, TOEWS Rob, Forbes, 2021;

Onde podemos aplicar uma Rede Neural Artificial?



Coming, TOEWS Rob,

Fonte: “Top 10 Facial Recognition APIs & Software of 2021”, WALLING Alex, rapidapi.com, 2021;

Onde podemos aplicar uma Rede Neural Artificial?

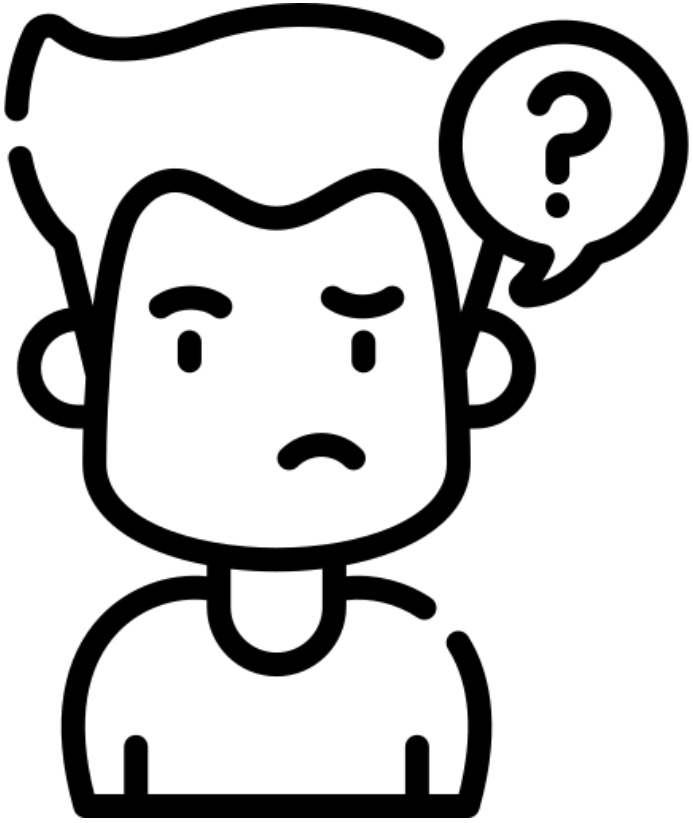
Coming, TOEWS Rob,

Fonte: “Top 10 Facial Recognition APIs & Software of 2021”, WALLING Alex, rapidapi.com, 2021;

Onde podemos aplicar uma Rede Neural Artificial?

Quando usar uma rede Neural?

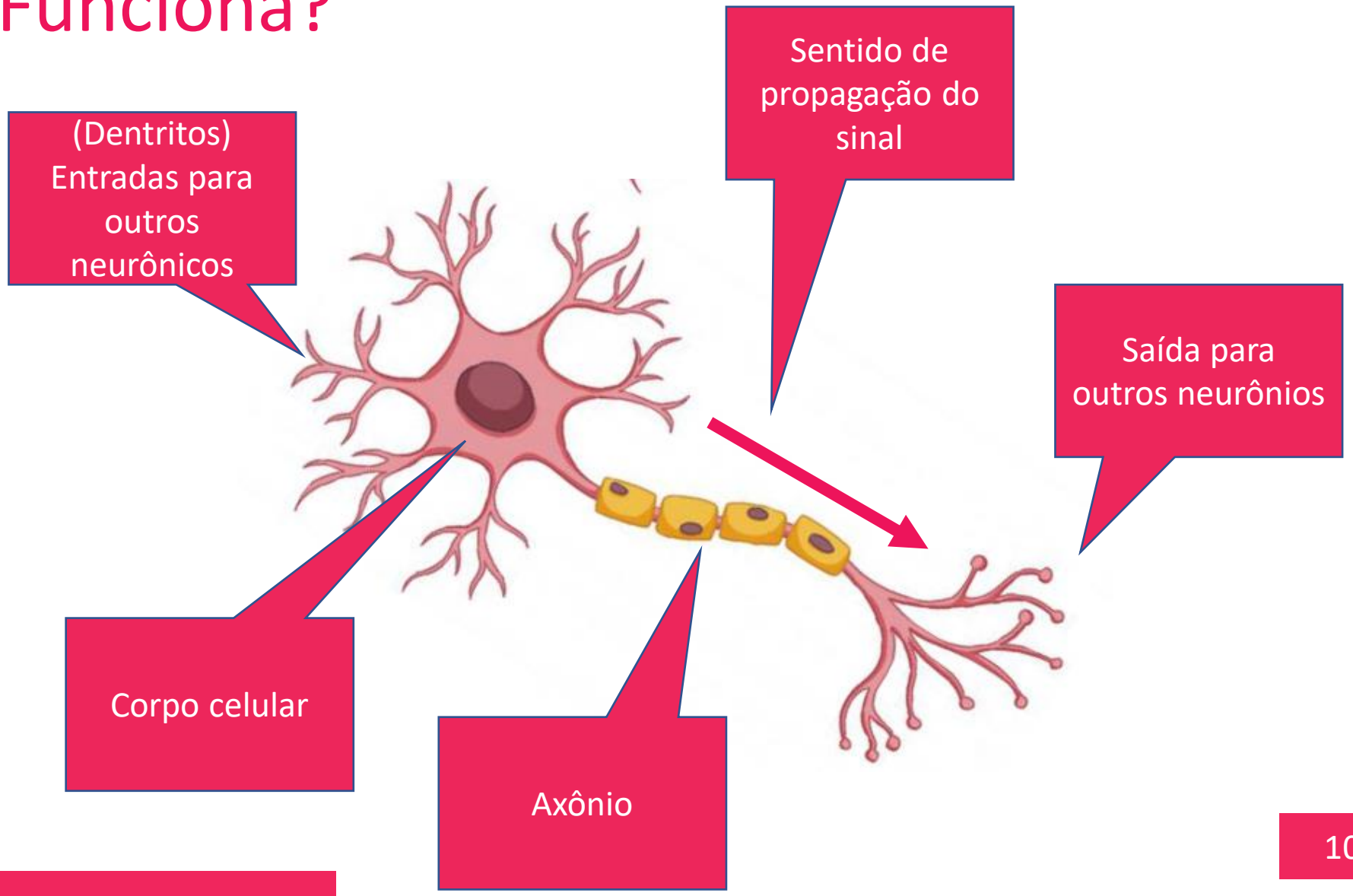
FIAP



- ✓ Quando NÃO existe um algortimo que modele o Sistema / problema;
- ✓ Quando existirem MUITOS dados para analisar;
- ✓ Quando o seu problema é COMPLEXO

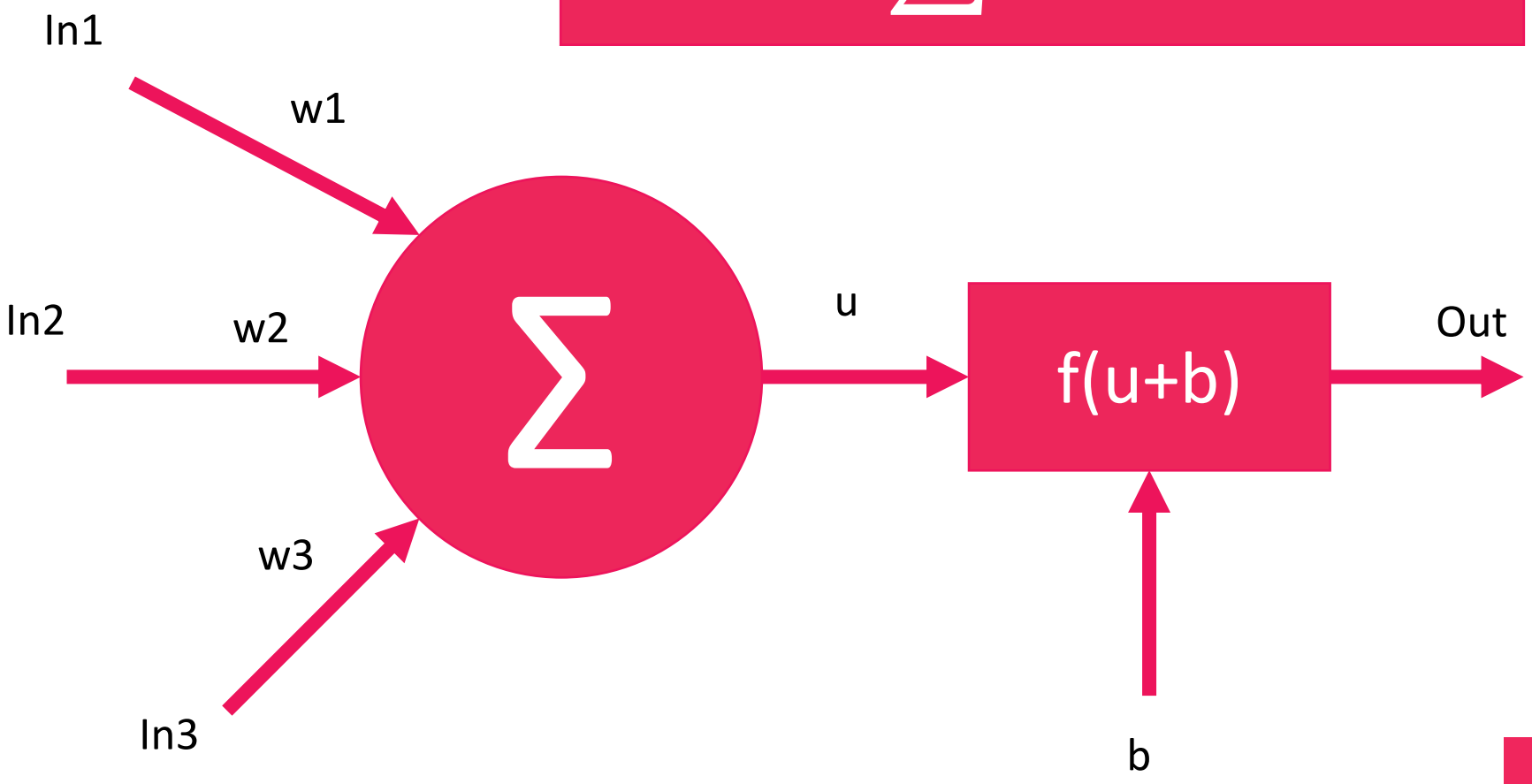
Como uma Rede Neural Artificial Funciona?

FIAP



Como uma Rede Neural Artificial Funciona?

$$Out = f(\sum(In_i \times w_i) + b)$$

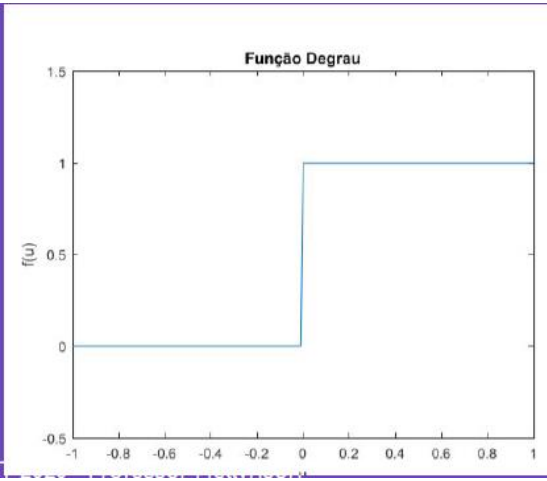


Função de Ativação do Neurônio

FIAP

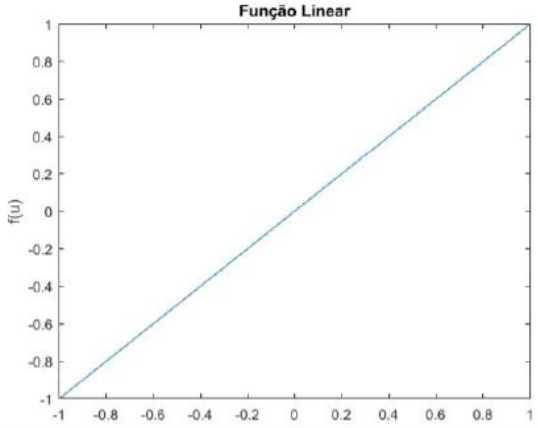
Degrau

$$f(u) = \begin{cases} 0, & u < 1 \\ 1, & u \geq 1 \end{cases}$$



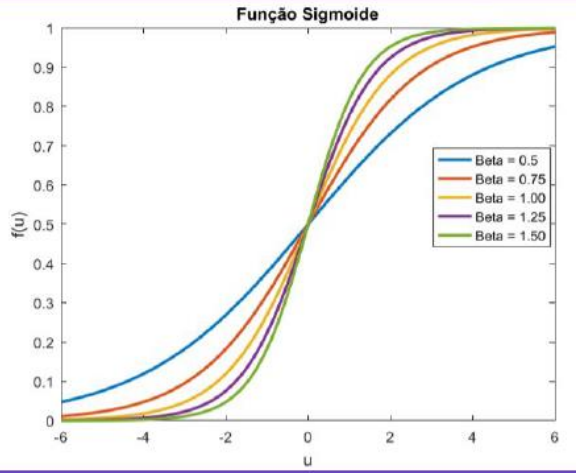
Linear

$$f(u) = u$$



Sigmóide

$$f(u) = \frac{1}{(1 + e^{-\beta u})}$$



Exercício 01

FIAP

- Dado uma rede neural de uma camada cujas entradas são 1, 2 e 3, e os pesos são 0.1, 0.2 e 0.3, calcule a saída deste perceptron considerando uma função de ativação degrau e bias = 0.

Entradas:
In1 = 1
In2 = 2
In3 = 3

Pesos:
w1 = 0.1
w2 = 0.2
w3 = 0.3

bias:
b = 0

$$Out = f(u + b) \rightarrow Out = f(\sum (In_i \times w_i) + b)$$

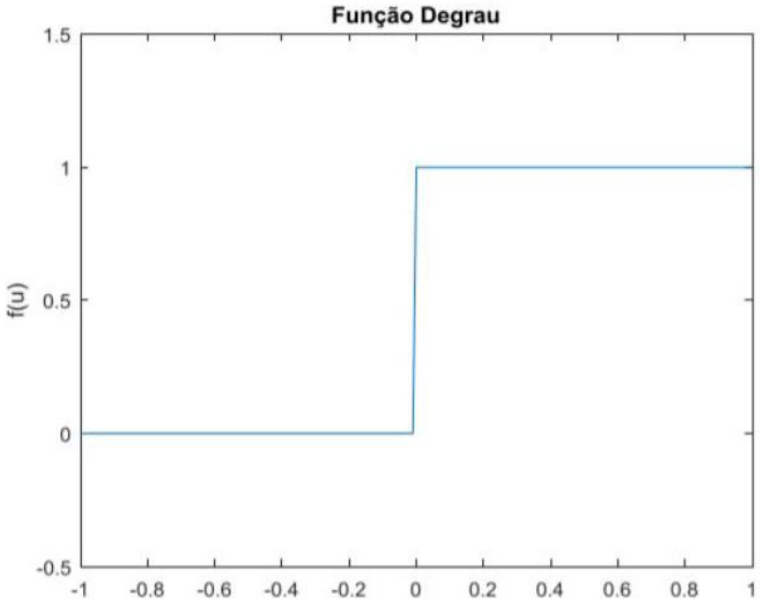
Exercício 01

$$u = 1 * 0.1 + 2 * 0.2 + 3 * 0.3 \ \therefore u = 1.4$$

$$out = 1.4 + 0 \ \therefore out = 1.4$$

Degrau

$$f(u) = \begin{cases} 0, & u < 1 \\ 1, & u \geq 1 \end{cases}$$



$$f(1.4) = 1$$

Exercício 02

FIAP

- Dado uma rede neural de uma camada cujas entradas são -1.0, 5.0 e 0.2, e os pesos são 0.11, -0.2 e 4.0, calcule a saída deste perceptron considerando uma função de ativação sigmóide e β= 1.25.

Entradas:
In1 = -1.0
In2 = 5.0
In3 = 0.2

Pesos:
w1 = 0.11
w2 = -0.2
w3 = 4.0

$$Out = f(u + b) \rightarrow Out = f(\sum (In_i \times w_i) + b)$$

Exercício 02

$$u = -1 * 0.11 + 5 * -0.2 + 0.2 * 4 \ \therefore u = -0.31$$

Sigmóide

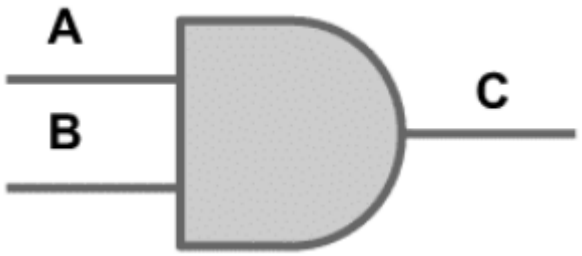
$$f(u) = \frac{1}{(1 + e^{-\beta u})}$$

$$Out = f(u + b) = \frac{1}{1 + e^{-1,25*(-0.31)}} \cong 0,40432$$

Exercício 03 – Para Casa

- Implemente o exercício 1 e 2 usando Python e a biblioteca numpy

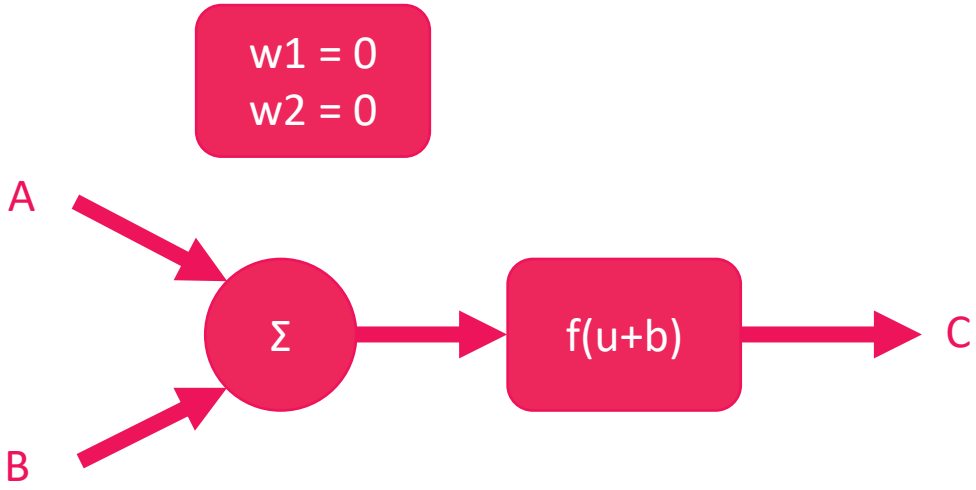
Ajuste de pesos – Treino da Rede Neural!



A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

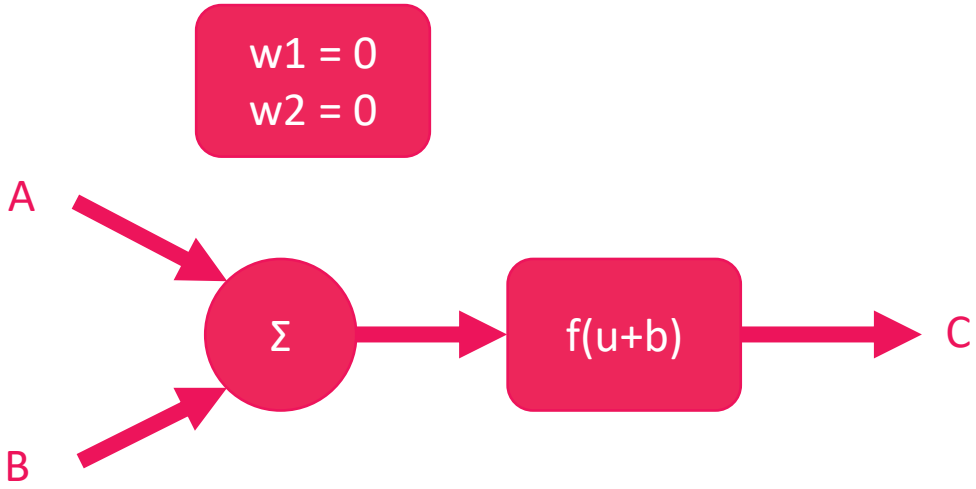
Ajuste de pesos – Treino da Rede Neural!

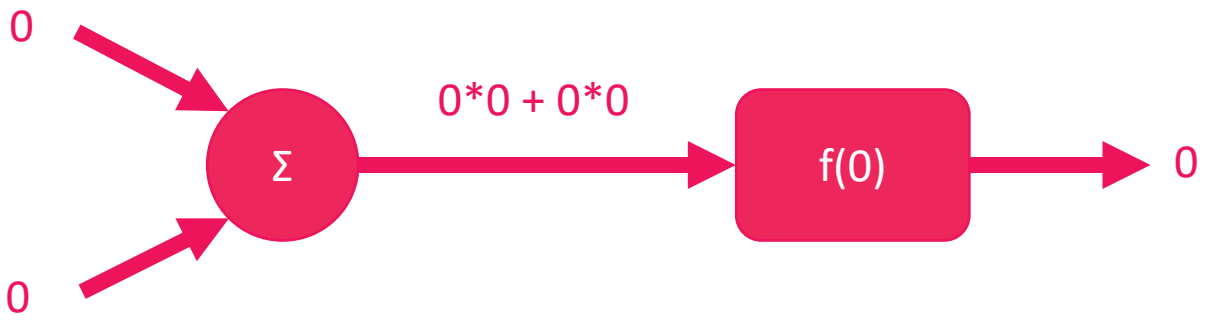
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1



Ajuste de pesos – Treino da Rede Neural!

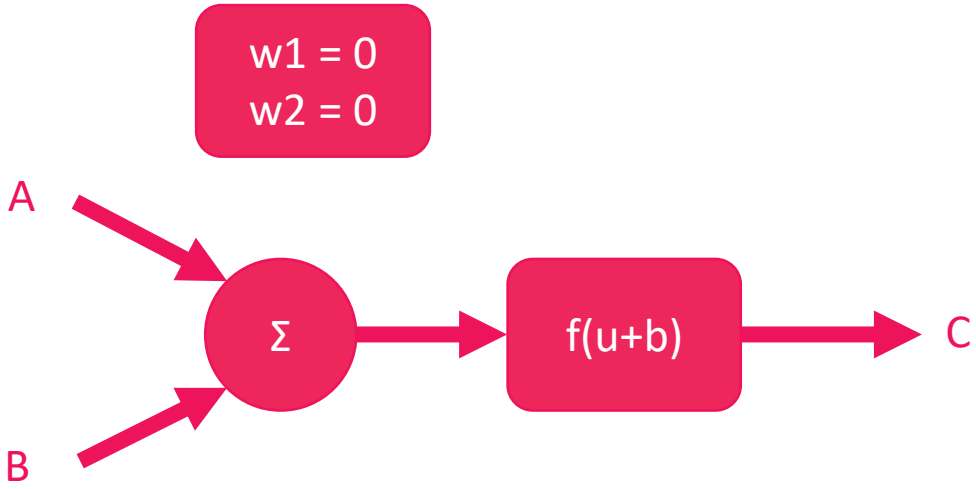
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

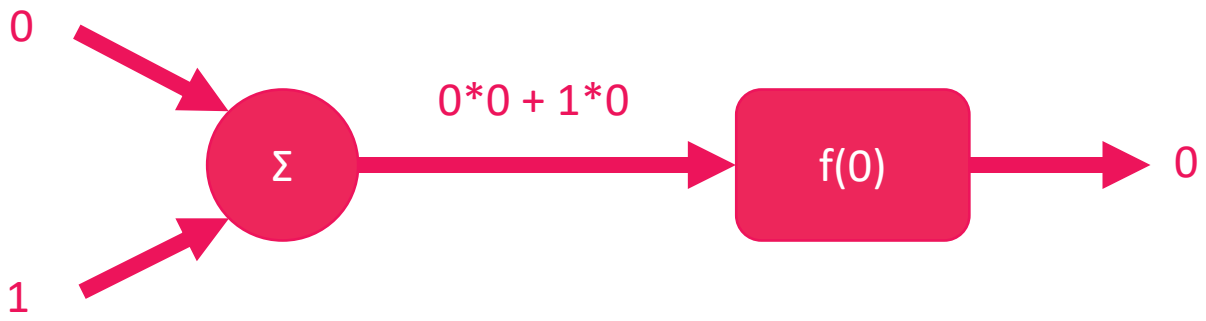




Ajuste de pesos – Treino da Rede Neural!

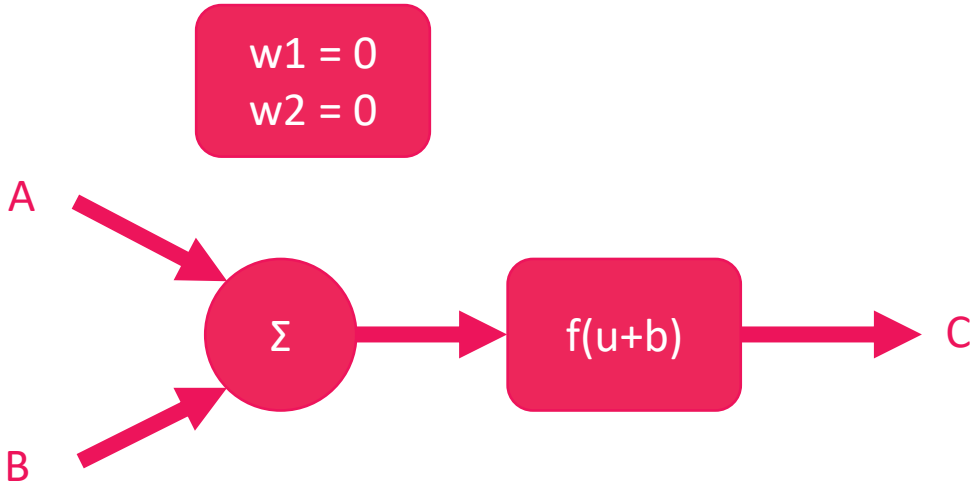
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

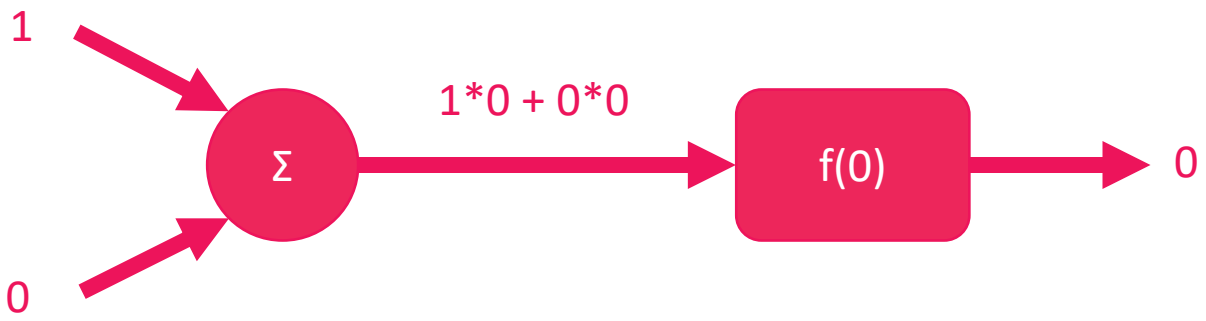




Ajuste de pesos – Treino da Rede Neural!

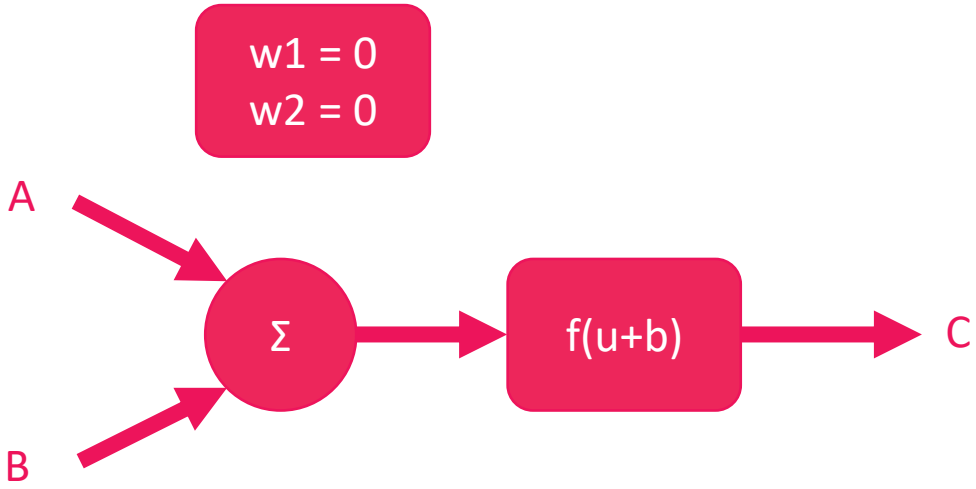
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

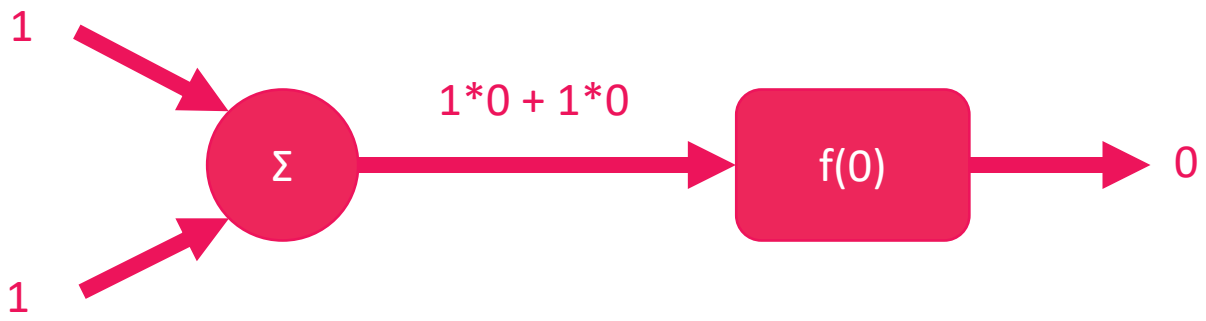




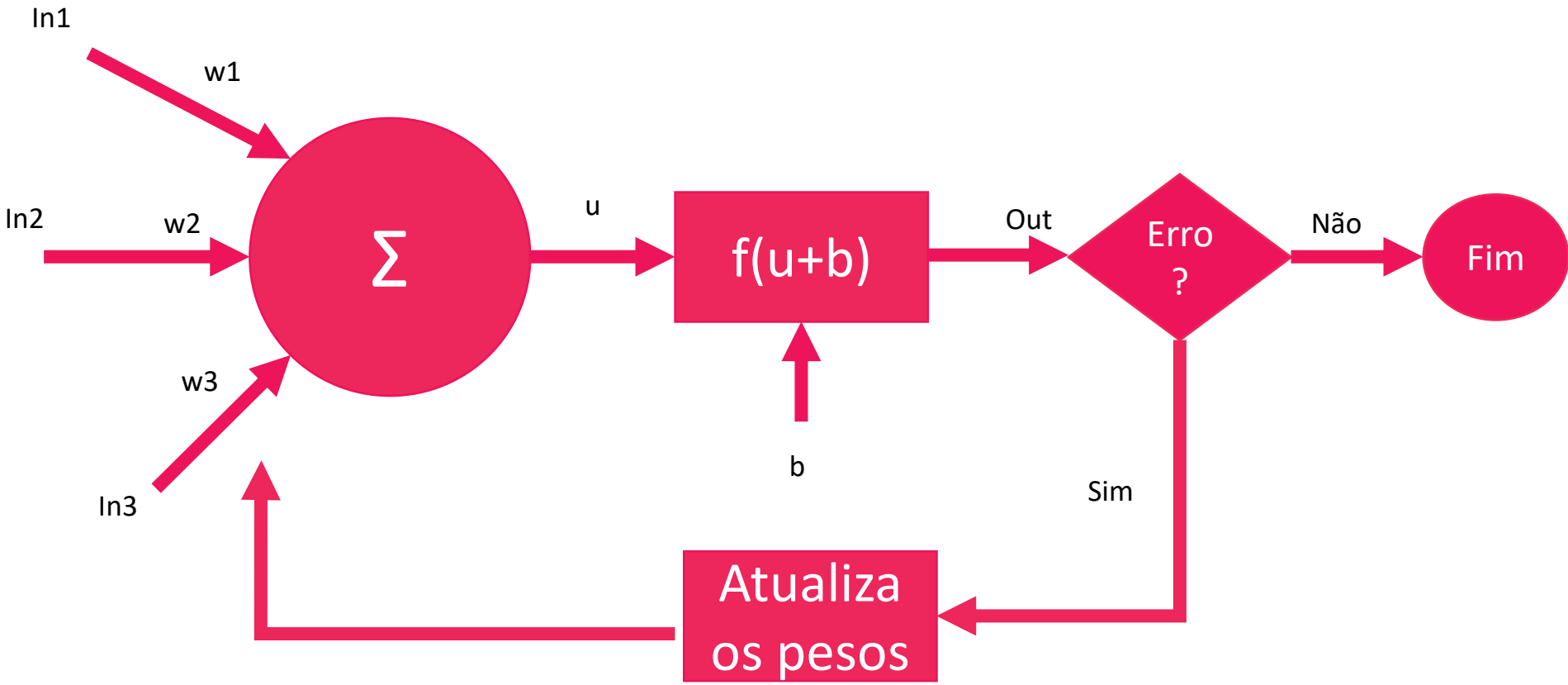
Ajuste de pesos – Treino da Rede Neural!

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1





Ajuste de pesos – Treino da Rede Neural!



Ajuste de pesos – Calculado o Erro e atualizando os pesos

FIAP

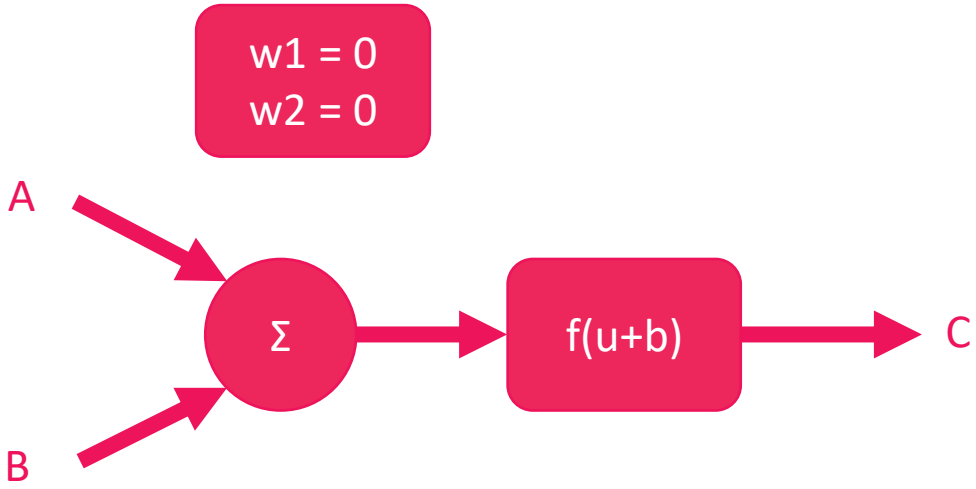
- Erro = respostaCorreta – respostaCalculada

- Precisamos atualizar os pesos até os erros serem pequenos ou nulos;

$$pesos(n+1) = peso(n) + (taxaAprendizagem * entrada * erro)$$

Ajuste de pesos – Treino da Rede Neural!

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1



Fixando a taxa de aprendizagem em 0.1

- Erro = respostaCorreta – respostaCalculada = 1 – 0 = 1

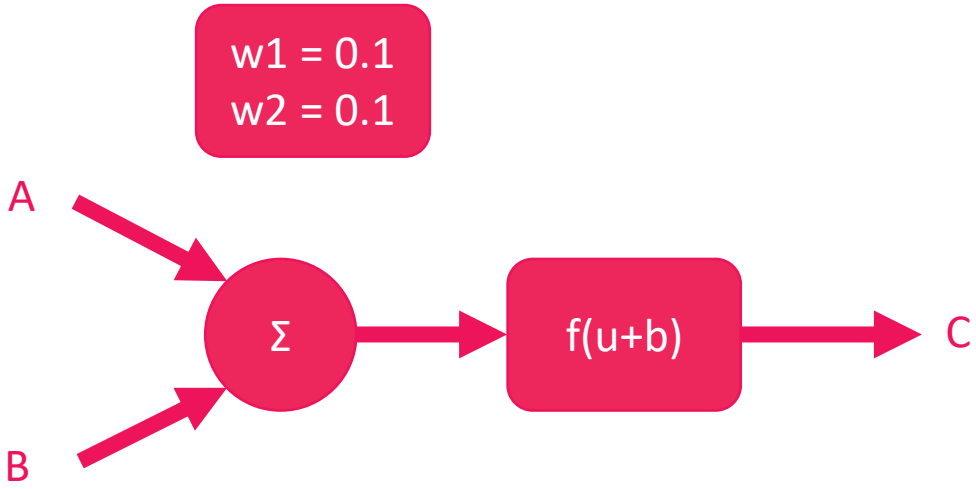
$$pesos(n+1) = peso(n) + (taxaAprendizagem * entrada * erro)$$

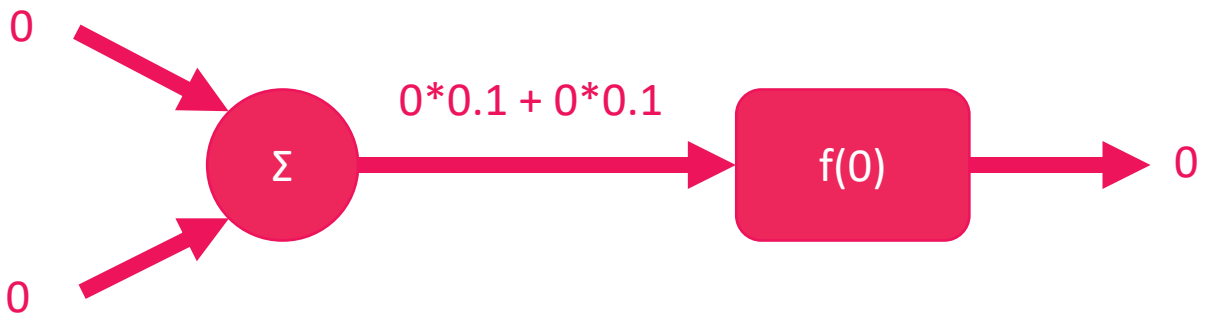
$$A: 0 + (0.1 * 1 * 1) = 0.1$$

$$B: 0 + (0.1 * 1 * 1) = 0.1$$

Ajuste de pesos – Treino da Rede Neural!

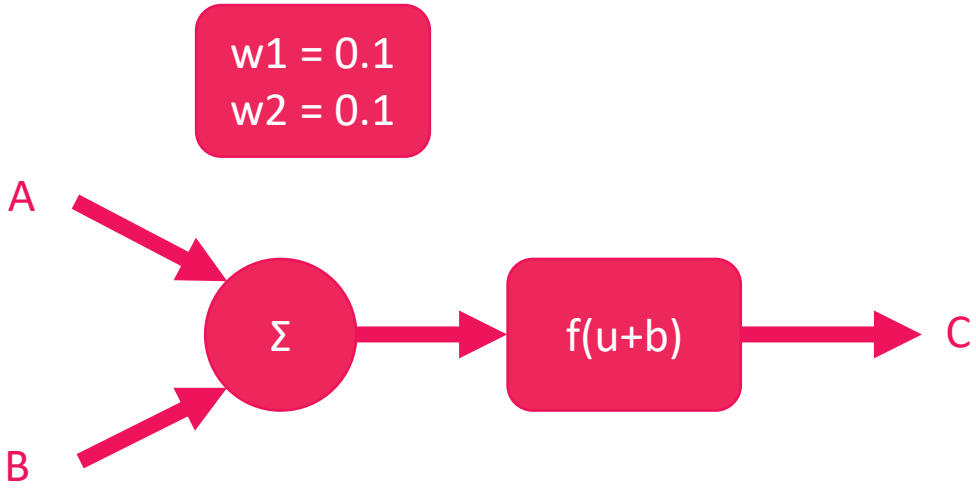
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

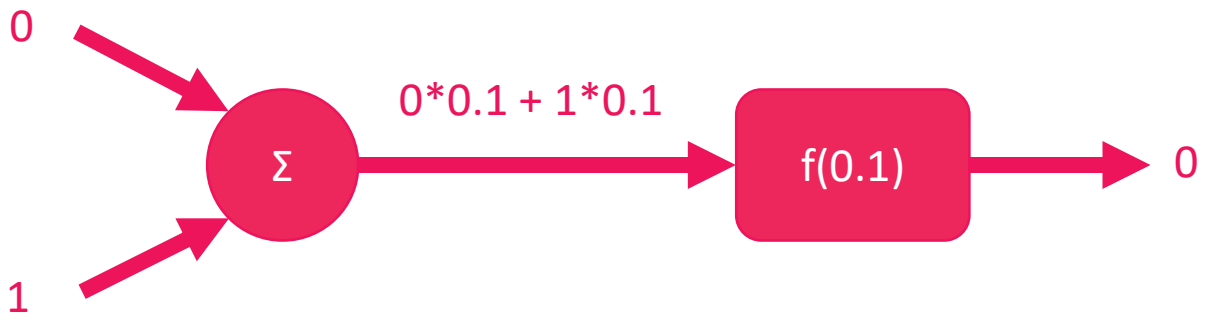




Ajuste de pesos – Treino da Rede Neural!

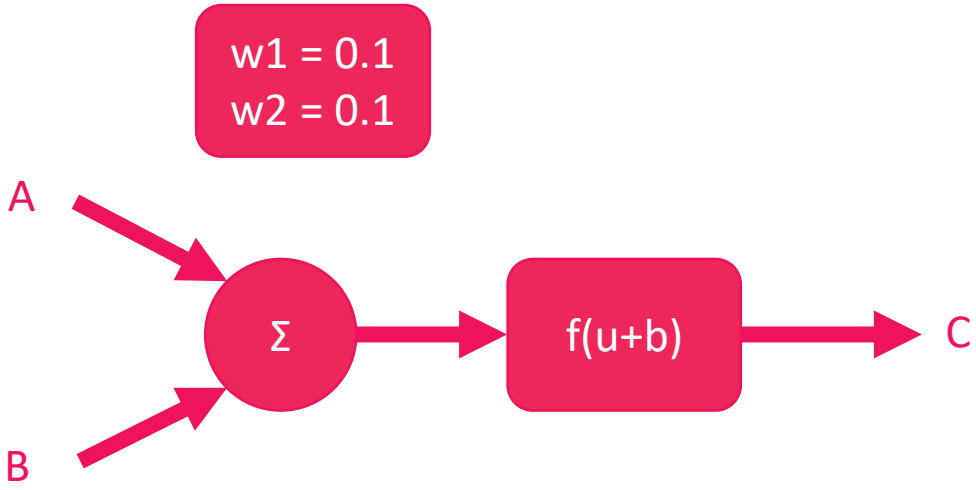
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

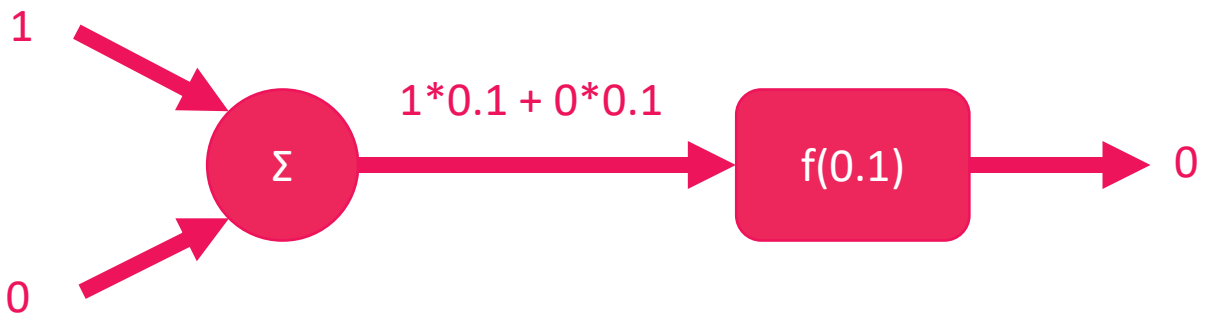




Ajuste de pesos – Treino da Rede Neural!

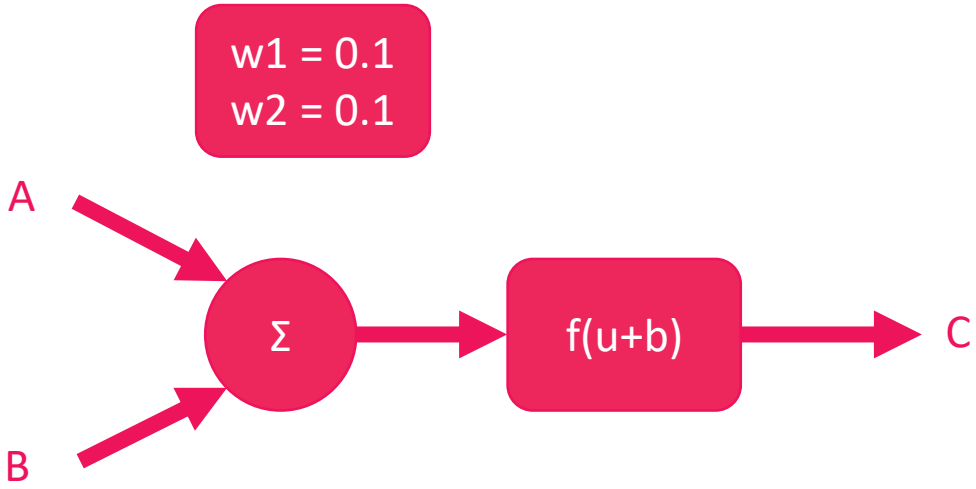
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

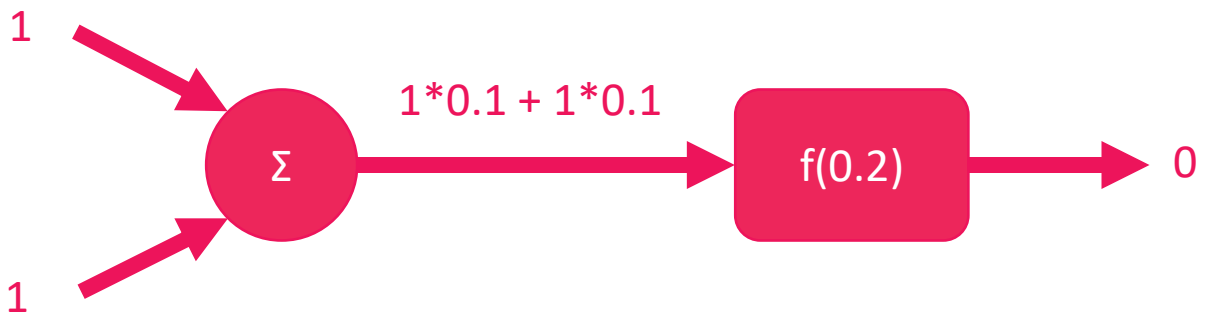




Ajuste de pesos – Treino da Rede Neural!

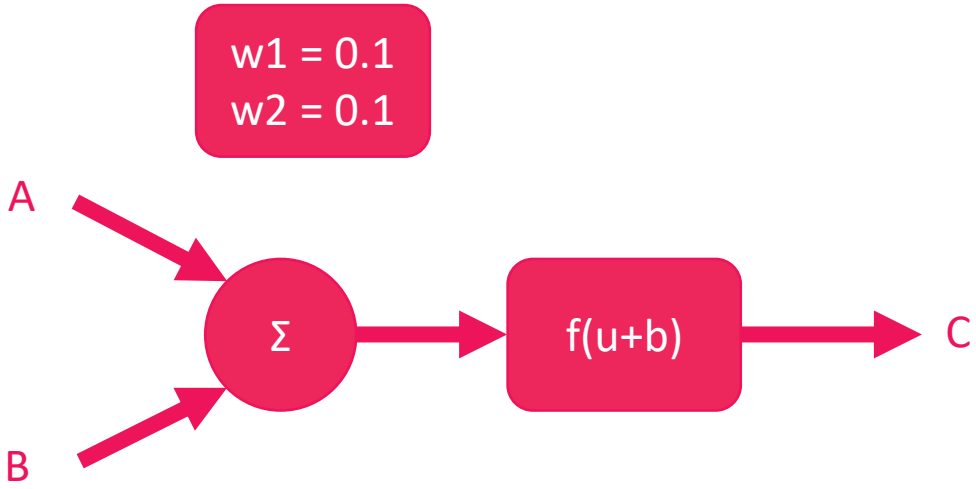
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1





Ajuste de pesos – Treino da Rede Neural!

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1



Fixando a taxa de aprendizagem em 0.1

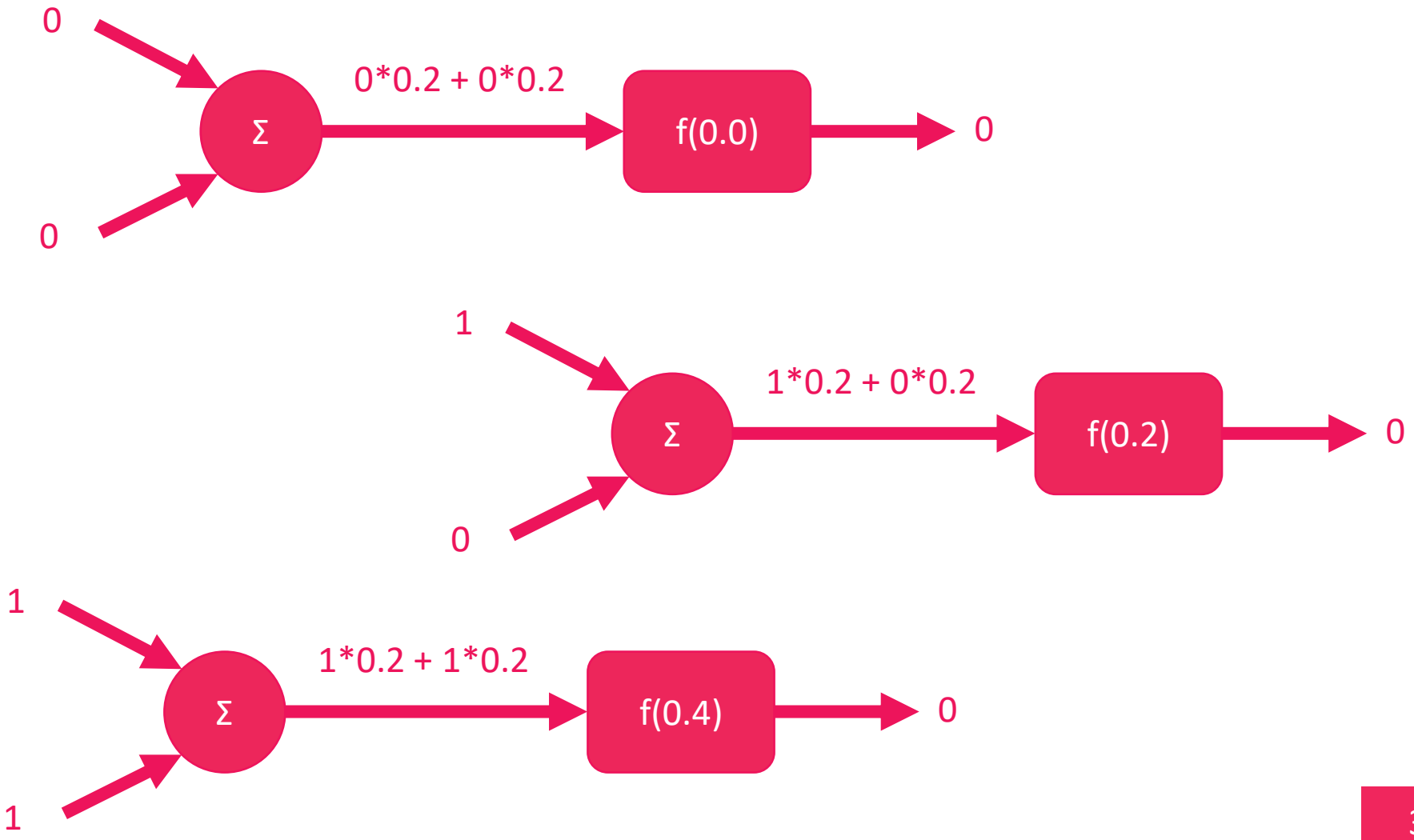
- Erro = respostaCorreta – respostaCalculada = 1 – 0 = 1

$$pesos(n+1) = peso(n) + (taxaAprendizagem * entrada * erro)$$

$$A: 0.1 + (0.1 * 1 * 1) = 0.2$$

$$B: 0.1 + (0.1 * 1 * 1) = 0.2$$

Ajuste de pesos – Treino da Rede Neural!



Exercício 04

- Encontre os pesos ideias para a Rede Neural Artificial apresentada, mantendo a taxa de aprendizagem de 0.1 e partindo da ultima iteração apresentada.

FIAP

This presentation has been designed using images from Flaticon.com