



Human-Centered Data & AI

# Vinicius Caridá, Ph.D.



Data Science Manager, Itaú Unibanco  
MBA Professor, FIAP  
GDE – Machine Learning



@vinicius caridá



@vfcarida



@vinicius caridá



@vfcarida



Google  
Scholar  
@vinicius caridá



@vfcarida

“

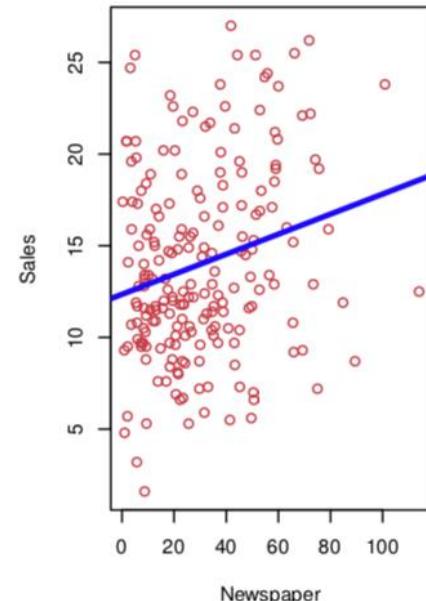
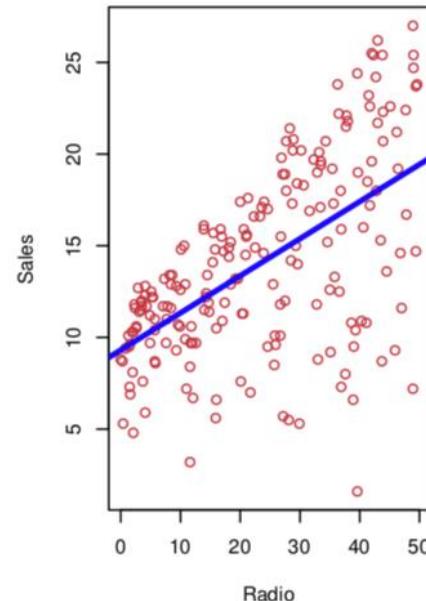
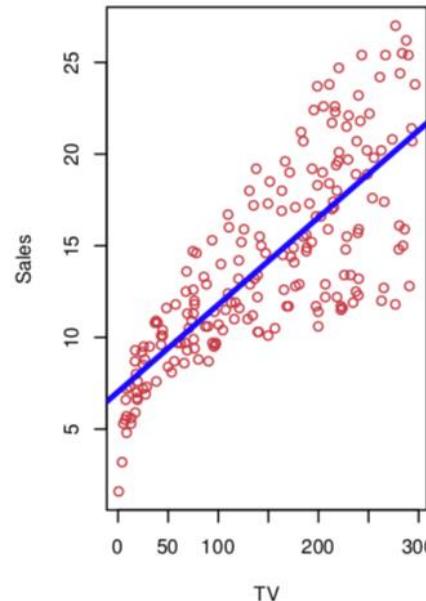
# Redes Neurais

“

# Regressão Linear

# Regressão Linear

- Há alguma relação entre aumento de vendas e propaganda?
- Qual mídia contribui mais para as vendas?



# Régressão Linear

- Para o exemplo do slide 4, foi obtida a seguinte equação do hiperplano:

$$Vendas = 2,939 + 0,046 \times TV + 0,189 \times radio + 0.01 \times Jornal$$

# Régressão Linear

- Para o exemplo do slide 4, foi obtida a seguinte equação do hiperplano:

$$Vendas = 2,939 + 0,046 \times TV + 0,189 \times radio + 0.01 \times Jornal$$

- caso nenhuma ação de propaganda seja feita as vendas serão de 2,939
- Mantendo todos os outros valores constantes, a cada uma unidade aumentada nas ações por TV, as vendas aumentam 0,046
- A influência da utilização de jornal é quase nula

# Régressão Linear

- Abordagem supervisionada simples
- Assume uma dependência linear entre a variável resposta  $Y$  e os valores  $X_1, X_2, \dots, X_p$
- Assume-se o modelo:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

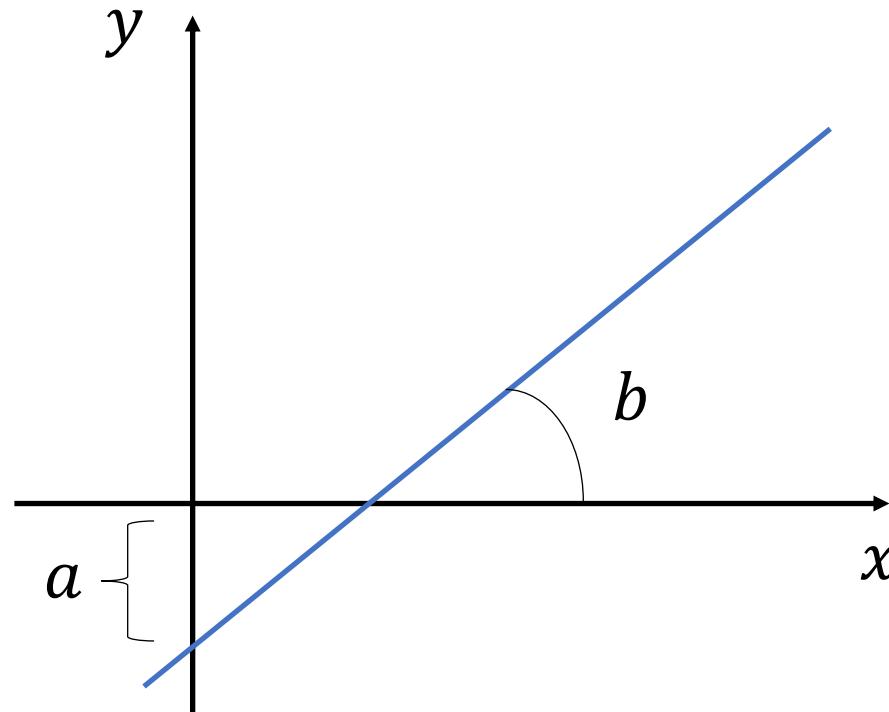
Sendo  $\beta_0, \beta_1, \dots, \beta_p$  coeficientes aprendidos pelo modelo

# Rregressão Linear

Reta

- $a$ : deslocamento
- $b$ : inclinação

$$y = a + bx$$

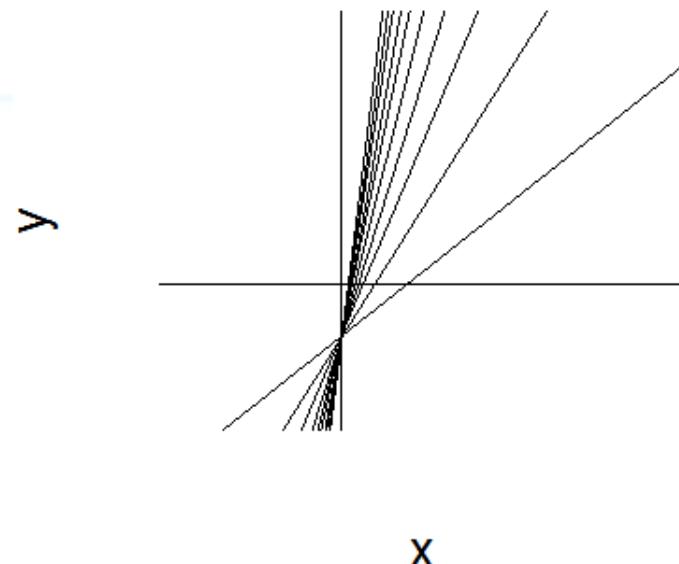


# Regressão Linear

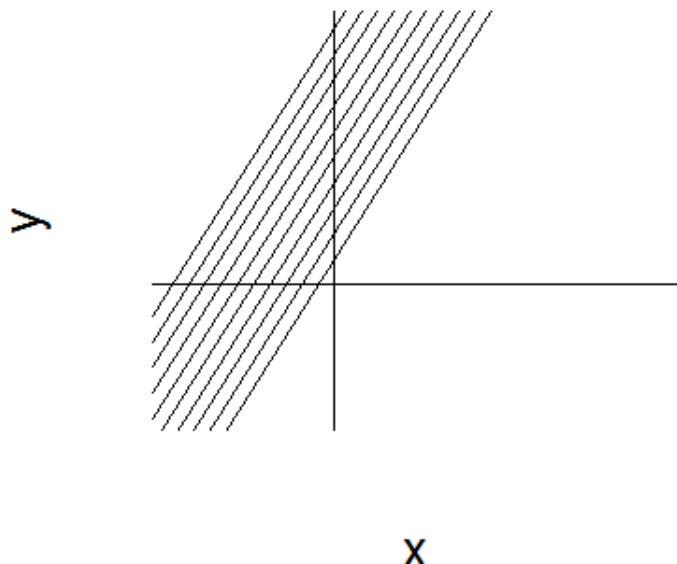
Reta

$$y = a + bx$$

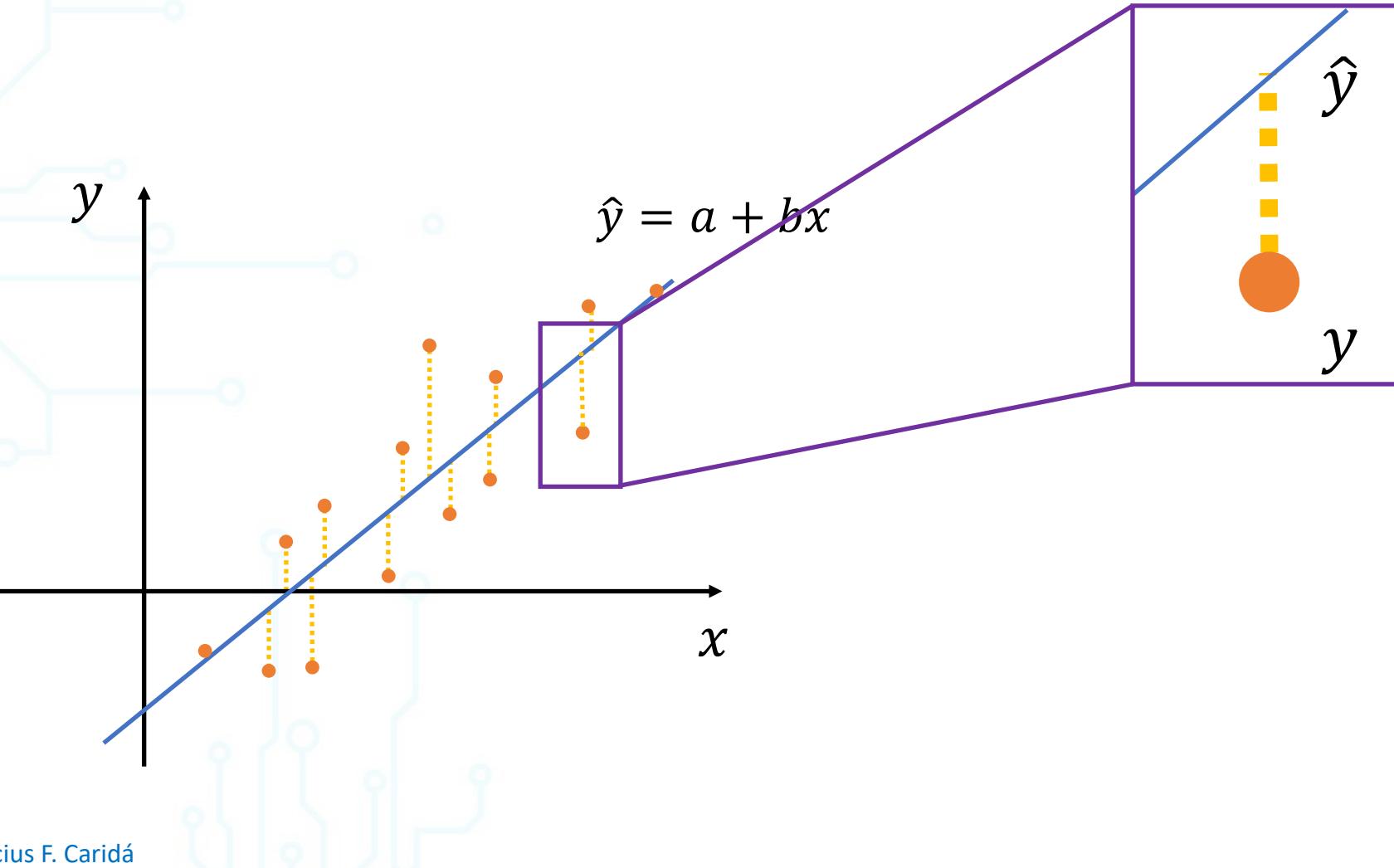
$a$  fixo;  $b$  variável



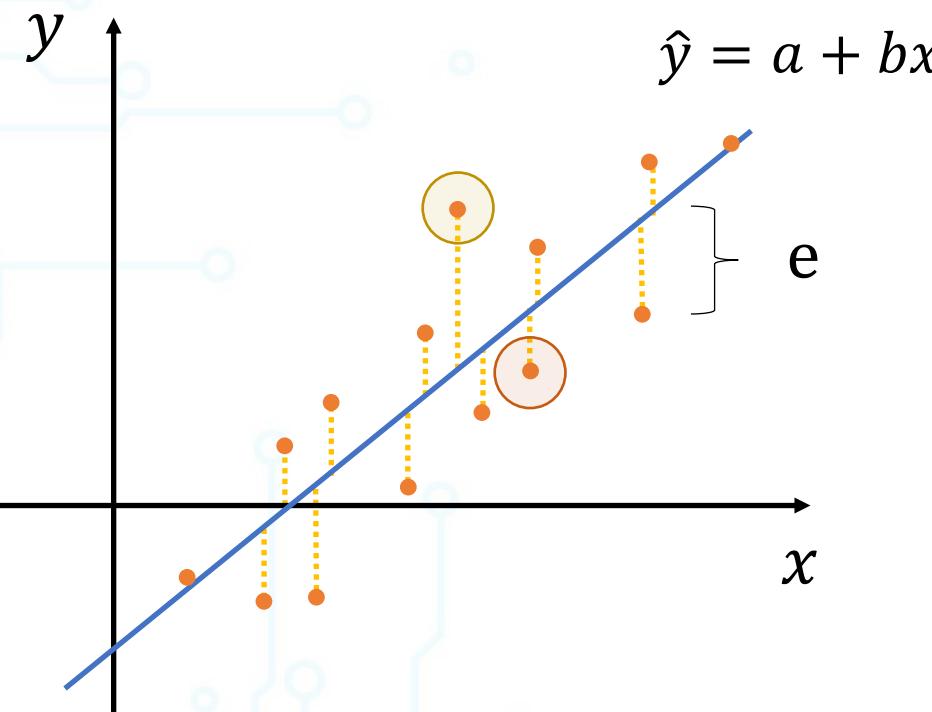
$a$  variável;  $b$  fixo



# Regressão Linear



# Regressão Linear



$$e = y - \hat{y}$$

$$e = y - (a + bx)$$

**Positivo**

$$y - \hat{y} > 0$$

**Negativo**

$$y - \hat{y} < 0$$

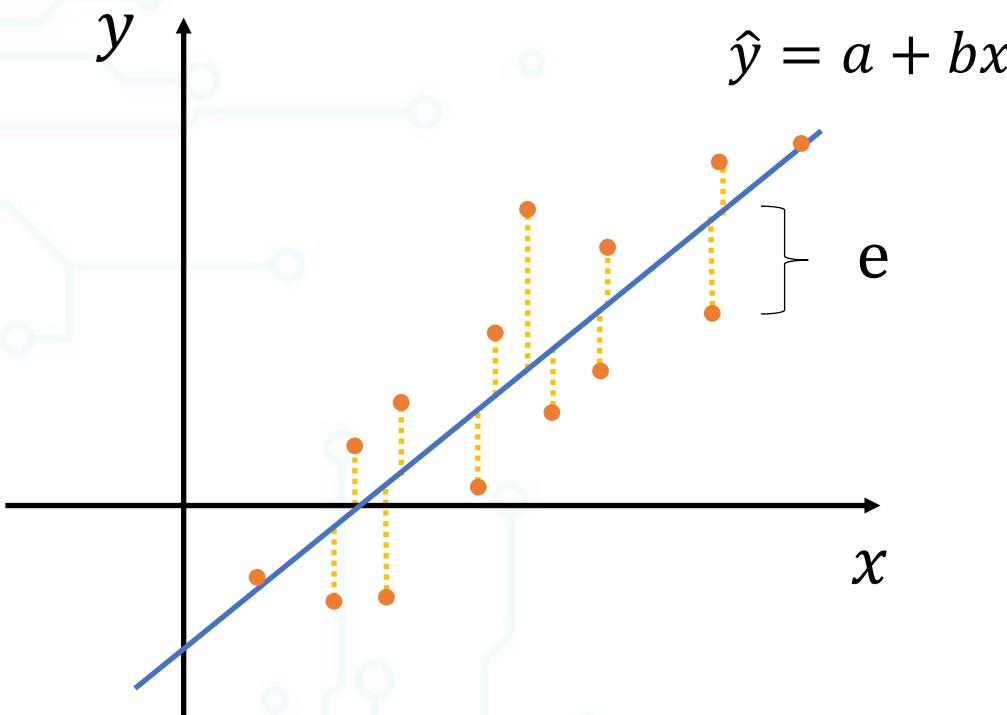
**Solução?**

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

$$|y - \hat{y}|$$



# Regressão Linear



**Como avaliar o erro total?**

**Erro quadrático  
médio  
R (MSE)**

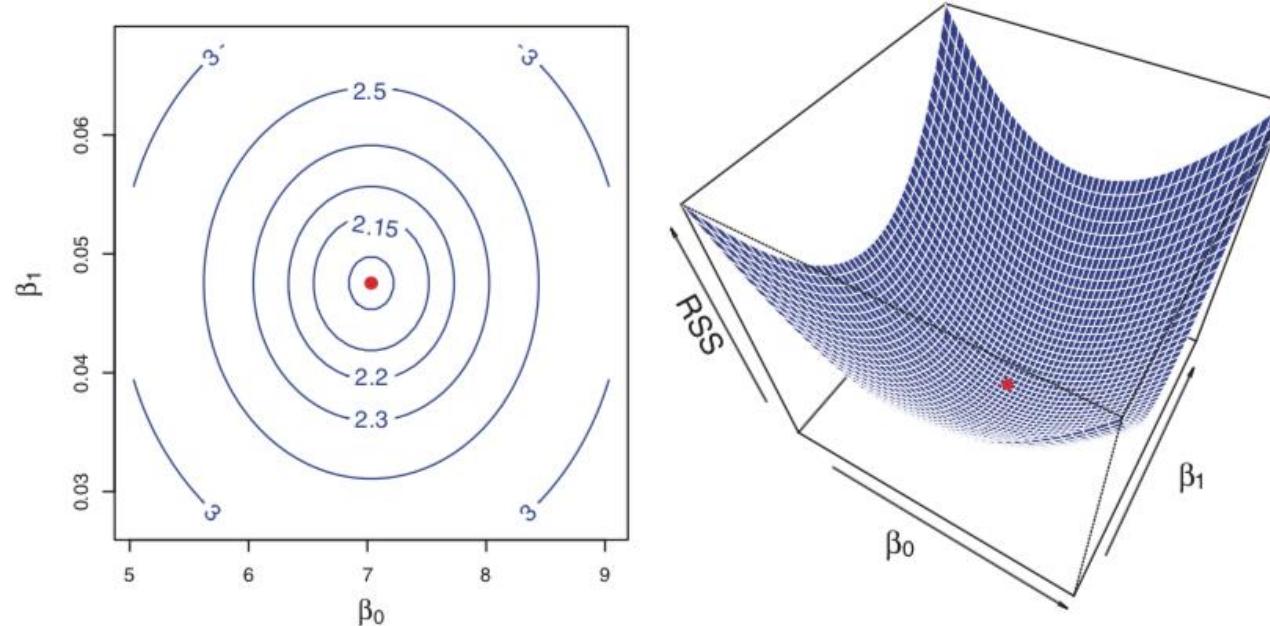
$$\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

**Erro absoluto  
médio  
(MAE)**

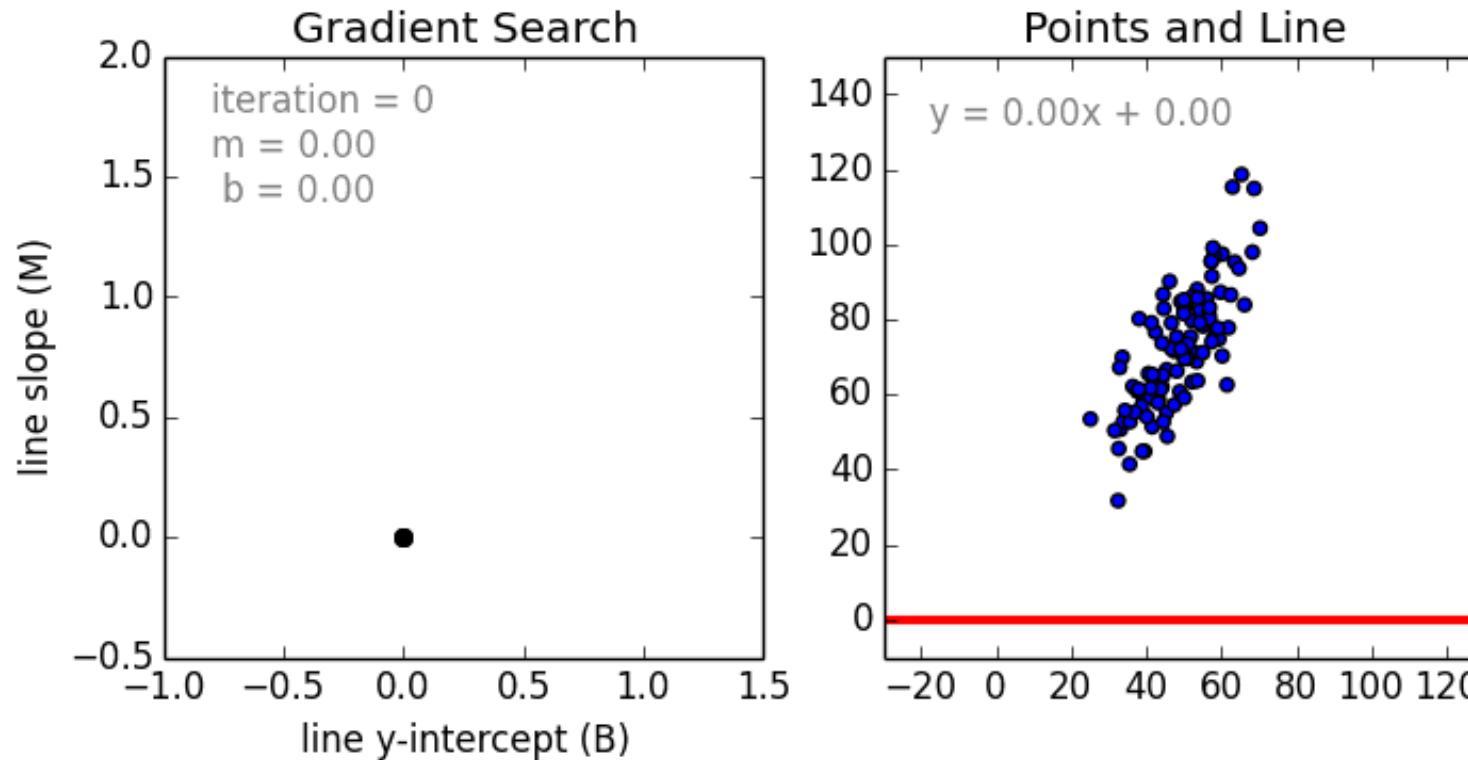
$$\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

# Regressão Linear

- Como a função RMSE é convexa, é possível encontrar o valor mínimo por meio de algoritmos de otimização



# Regressão Linear



# Regressão Linear

Peso      Altura

	Pessoa 1	80 kg	163
-----------------------------------------------------------------------------------	----------	-------	-----

	Pessoa 2	85 kg	168
-----------------------------------------------------------------------------------	----------	-------	-----

	Pessoa 3	90 kg	175
-----------------------------------------------------------------------------------	----------	-------	-----

	Pessoa 4	95 kg	188
-----------------------------------------------------------------------------------	----------	-------	-----

# Regressão Linear

Peso      Altura

	Pessoa 1	80 kg	163
-----------------------------------------------------------------------------------	----------	-------	-----

$$\hat{y} = \beta_0 + \beta_1 X_1$$

	Pessoa 2	85 kg	168
-----------------------------------------------------------------------------------	----------	-------	-----

	Pessoa 3	90 kg	175
-----------------------------------------------------------------------------------	----------	-------	-----

	Pessoa 4	95 kg	188
-----------------------------------------------------------------------------------	----------	-------	-----

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg
	Pessoa 2	85 kg
	Pessoa 3	90 kg
	Pessoa 4	95 kg

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$\hat{y} = 3 + 0,7 \times 80$$

$$\hat{y} = 60$$

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg
	Pessoa 2	85 kg
	Pessoa 3	90 kg
	Pessoa 4	95 kg

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,7 \times 80$$

$$\hat{y} = 60$$

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg      163
	Pessoa 2	85 kg      168
	Pessoa 3	90 kg      175
	Pessoa 4	95 kg      188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,7 \times 80$$

$$MSE = \sum (163 - 60)^2$$

$$\hat{y} = 60$$

$$MSE = 10.609$$

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg      163
	Pessoa 2	85 kg      168
	Pessoa 3	90 kg      175
	Pessoa 4	95 kg      188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,7 \times 80$$

$$MSE = \sum (163 - 60)^2$$

$$\hat{y} = 60$$

$$MSE = 10.609$$

$$\hat{y} = 5 + 1,3 \times 85$$

$$MSE = \sum (168 - 115,5)^2$$

$$\hat{y} = 115,5$$

$$MSE = 2.756,25$$

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg      163
	Pessoa 2	85 kg      168
	Pessoa 3	90 kg      175
	Pessoa 4	95 kg      188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 3 + 0,7 \times 80$$

$$MSE = \sum (163 - 60)^2$$

$$\hat{y} = 60$$

$$MSE = 10.609$$

$$\hat{y} = 5 + 1.3 \times 85$$

$$MSE = \sum (168 - 115,5)^2$$

$$\hat{y} = 115,5$$

$$MSE = 2.756,25$$

$$\hat{y} = 10 + 1.6 \times 90$$

$$MSE = \sum (175 - 154)^2$$

$$\hat{y} = 154$$

$$MSE = 441$$

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg
	Pessoa 2	85 kg
	Pessoa 3	90 kg
	Pessoa 4	95 kg

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 95$$

$$MSE = \sum (188 - 188)^2$$

$$\hat{y} = 188$$

$$MSE = 0$$

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg      163
	Pessoa 2	85 kg      168
	Pessoa 3	90 kg      175
	Pessoa 4	95 kg      188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{y} = 17 + 1,8 \times 95$$

$$MSE = \sum (188 - 188)^2$$

$$\hat{y} = 188$$

$$MSE = 0$$

$$\hat{y} = 161 \quad MSE = 4$$

$$\hat{y} = 170 \quad MSE = 4$$

$$\hat{y} = 179 \quad MSE = 16$$

$$\hat{y} = 188 \quad MSE = 0$$

$$MSE = 6$$

# Regressão Linear

	Peso	Altura
 Pessoa 1	80 kg	163
 Pessoa 2	85 kg	168
 Pessoa 3	90 kg	175
 Pessoa 4	95 kg	188

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 17$$

$$\beta_1 = 1,8$$

$$MSE = 6$$

$$\hat{y} = 17 + 1,8 X_1$$

# Regressão Linear

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	88 kg	???

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 17$$

$$\beta_1 = 1,8$$

$$MSE = 6$$

$$\hat{y} = 17 + 1,8 X_1$$

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	88 kg	???

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	88 kg	???

	Peso	Altura
Pessoa 1	80 kg	163
Pessoa 2	85 kg	168
Pessoa 3	90 kg	175
Pessoa 4	95 kg	188
Pessoa 5	88 kg	???

# Regressão Linear

	Peso	Altura
	Pessoa 1	80 kg      163
	Pessoa 2	85 kg      168
	Pessoa 3	90 kg      175
	Pessoa 4	95 kg      188
	Pessoa 5	88 kg      175,4

$$\hat{y} = \beta_0 + \beta_1 X_1$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\beta_0 = 17$$

$$\beta_1 = 1,8$$

$$MSE = 6$$

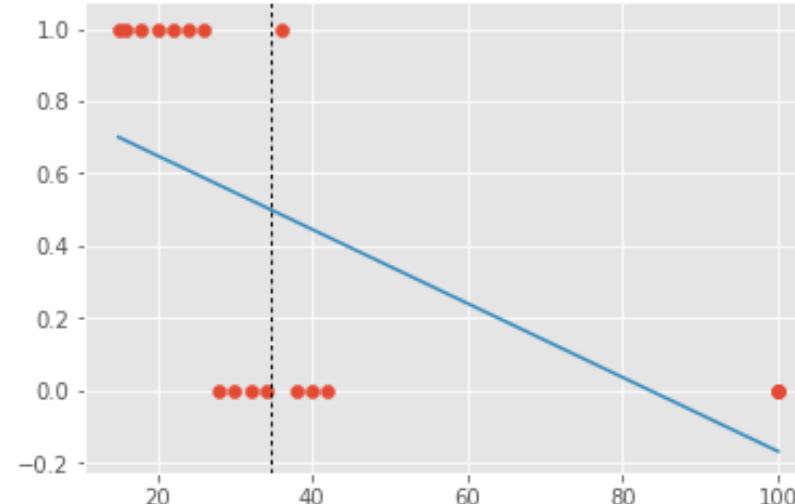
$$\hat{y} = 17 + 1,8 X_1$$

“

# Regressão Logística

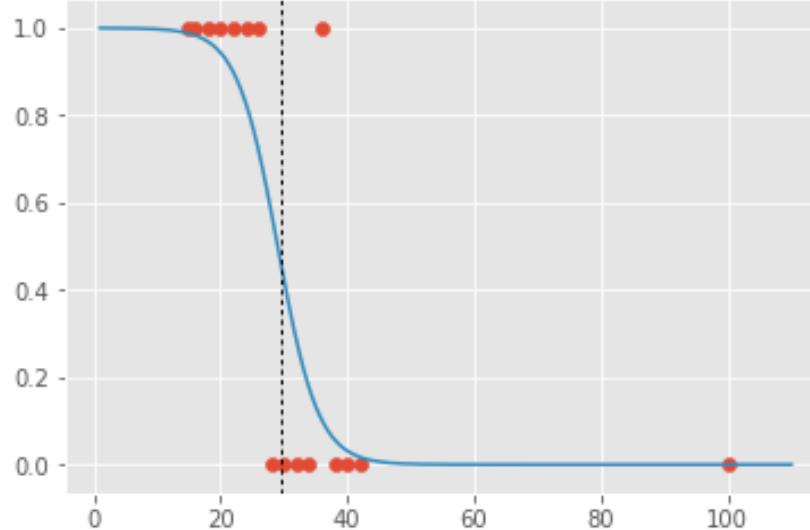
# Regressão Logística

- Queremos algum modelo capaz de classificar se a pessoa sofre um acidente com base no tempo que durou a auto-escola dela

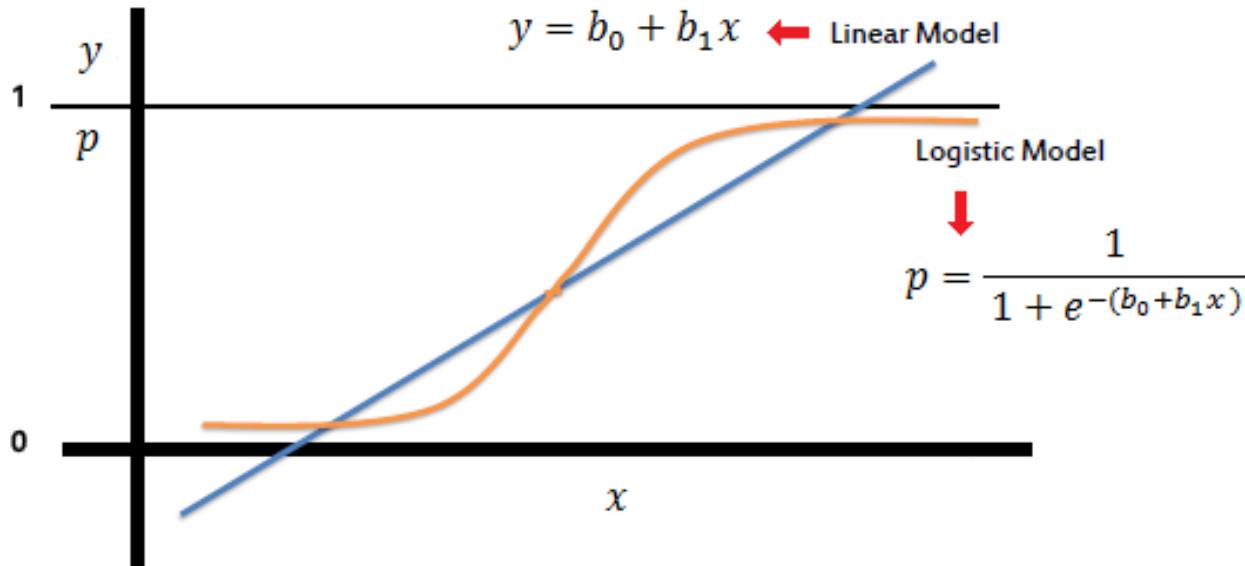


# Regressão Logística

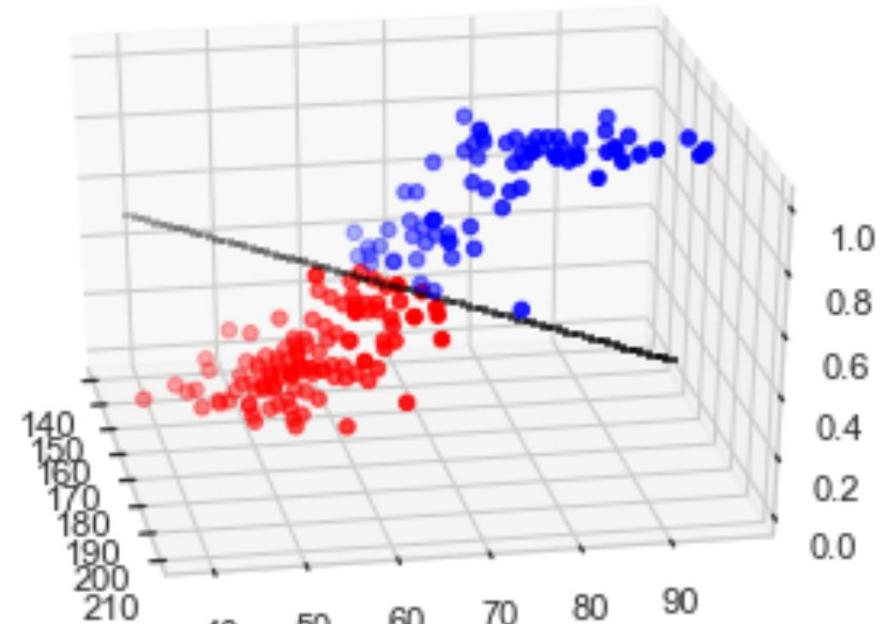
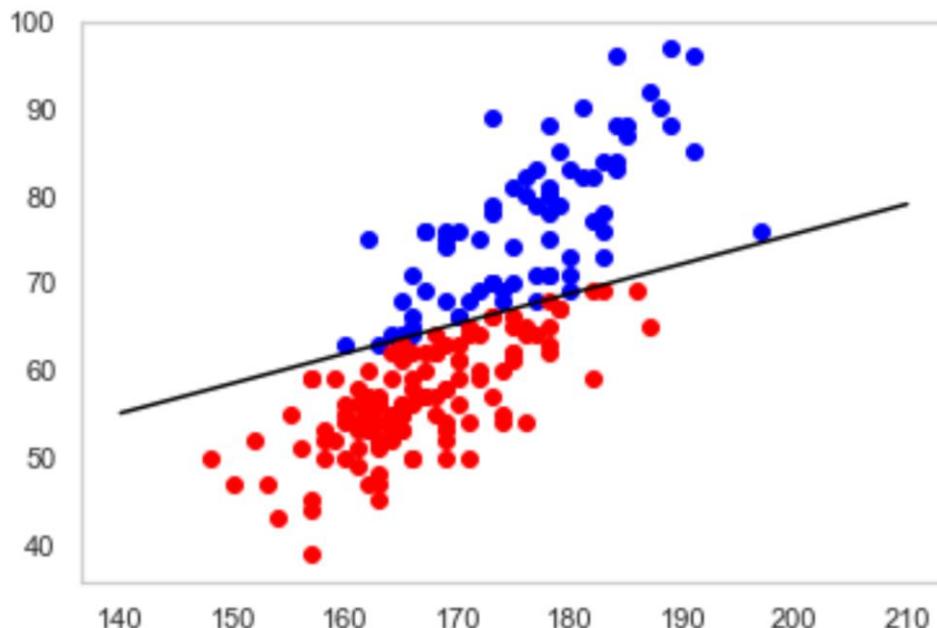
- A intuição por trás de regressão logística é bastante simples: em vez de acharmos a reta que melhor se ajusta aos dados, vamos achar uma curva em formato de 'S' que melhor se ajusta aos dados:



# Regressão Logística



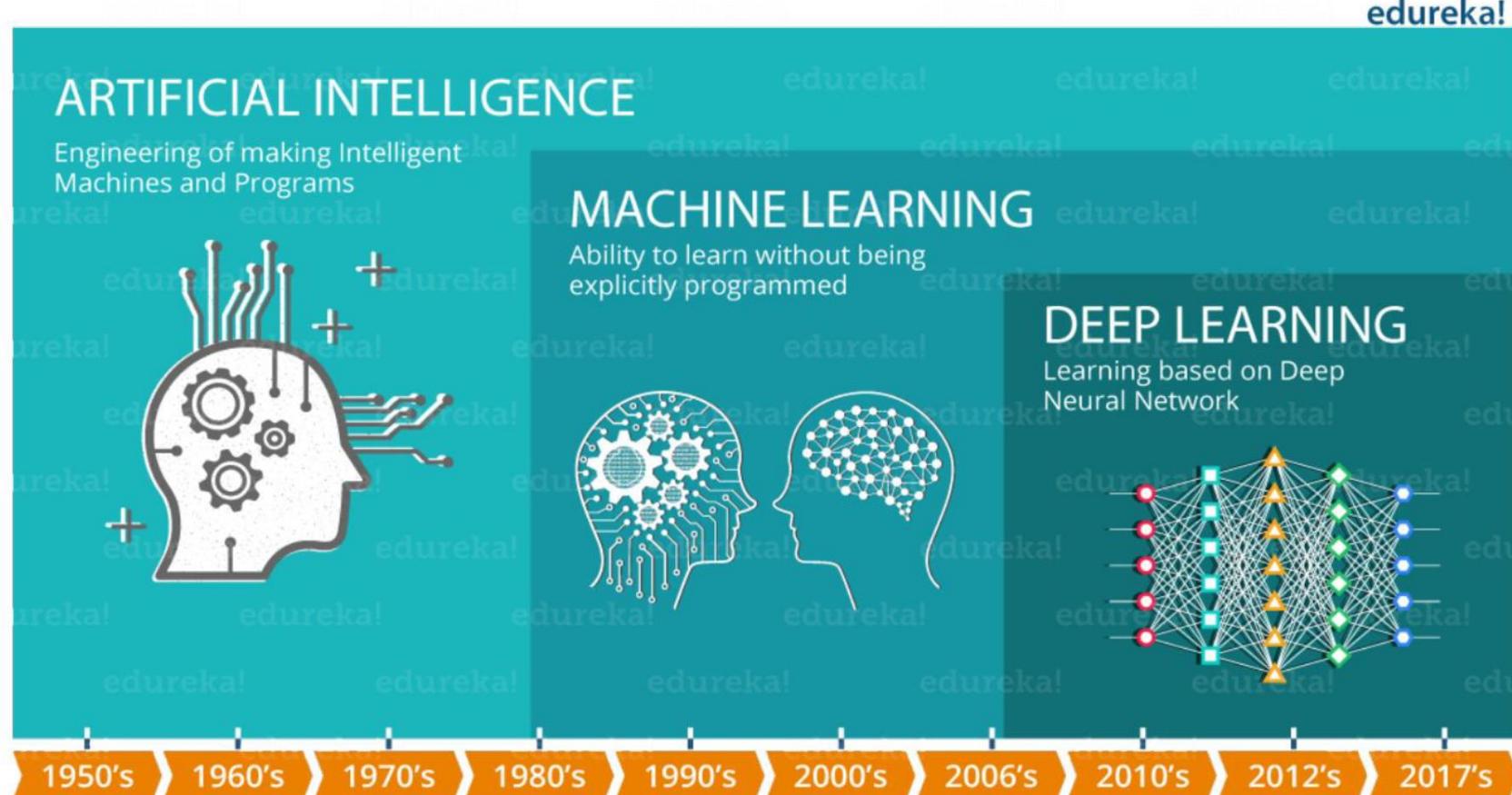
# Regressão Logística



“

# Redes Neurais

# Redes Neurais



# Redes Neurais

Albert Einstein: Insanity Is Doing  
the Same Thing Over and Over Again  
and Expecting Different Results

Machine learning:



# Redes Neurais

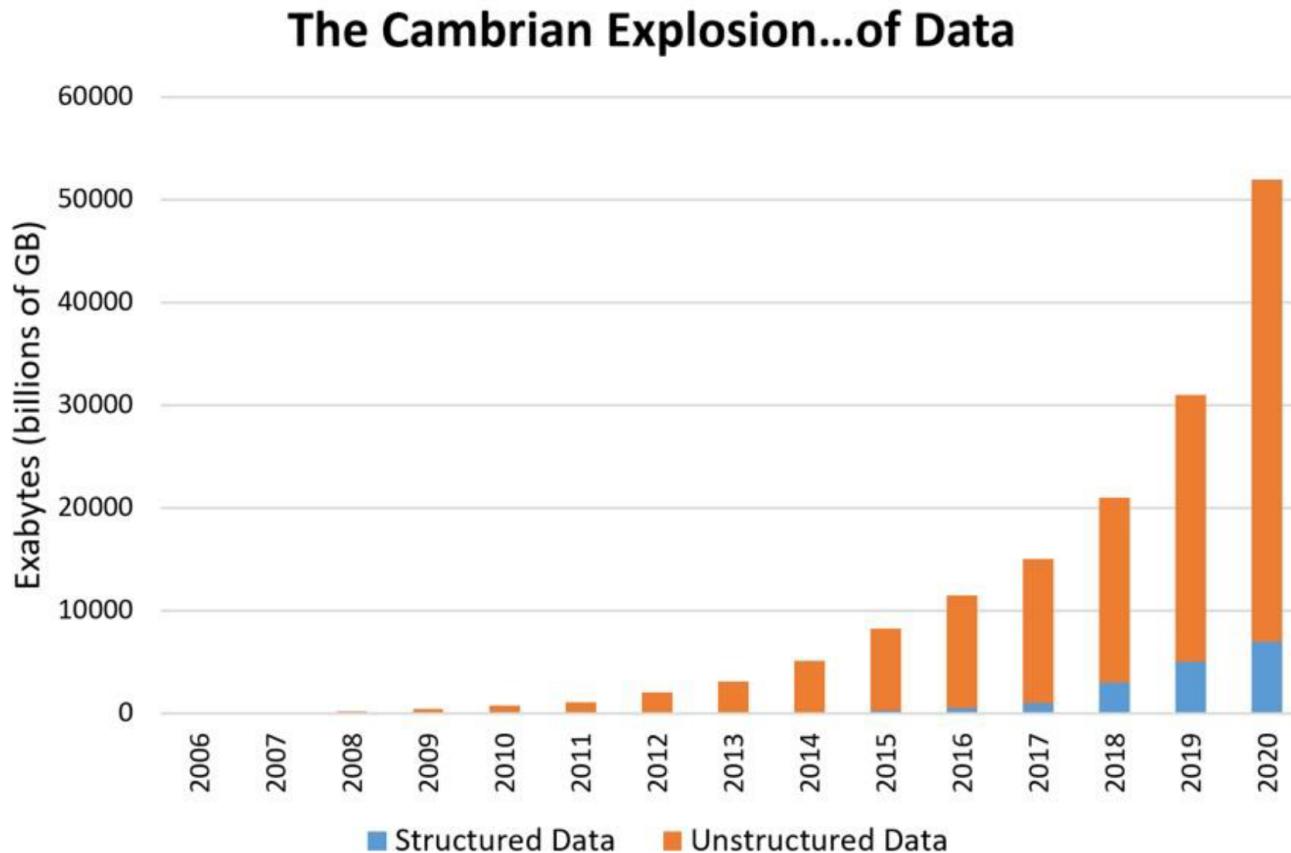


Figura 16 – Taxa de expansão na produção de dados estruturados e não-estruturados

Fonte: [https://www.eetimes.com/author.asp?section\\_id=36&doc\\_id=1330462](https://www.eetimes.com/author.asp?section_id=36&doc_id=1330462)

# Redes Neurais

## Motivação biológica

- O cérebro faz com que tarefas de classificação pareçam fáceis;
- O processo cerebral é realizado por redes de neurônios;
- Cada neurônio é conectado a vários outros neurônios;

# Redes Neurais

- O neurônio recebe impulsos (sinais) de outros neurônios por meio dos seus dendritos;
- O neurônio envia impulsos para outros neurônios por meio do seu axônio;
- O axônio termina num tipo de contato chamado sinapse, que conecta-o com o dendrito de outro neurônio.

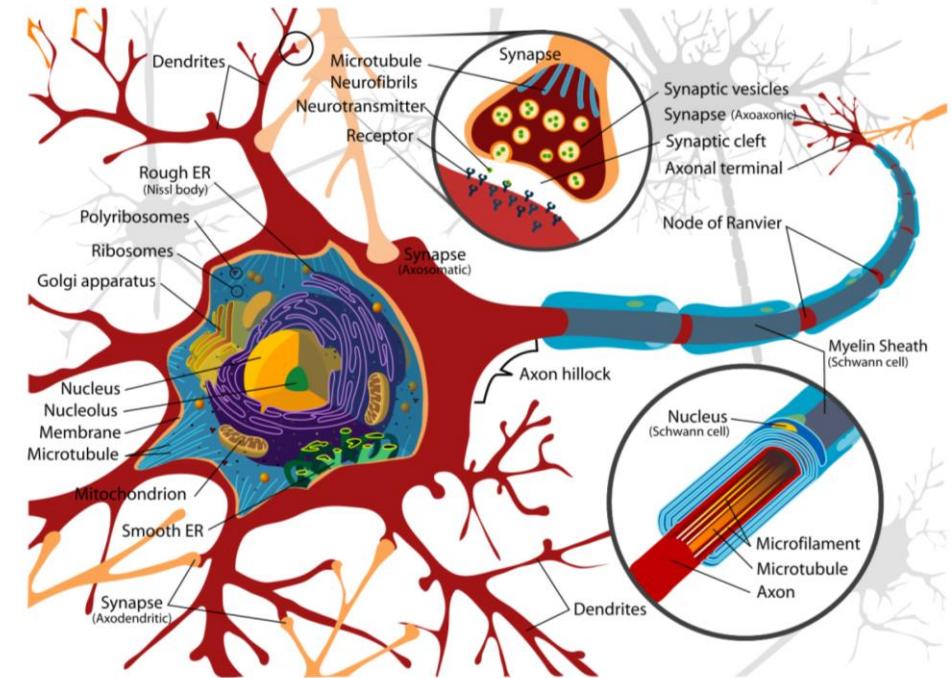
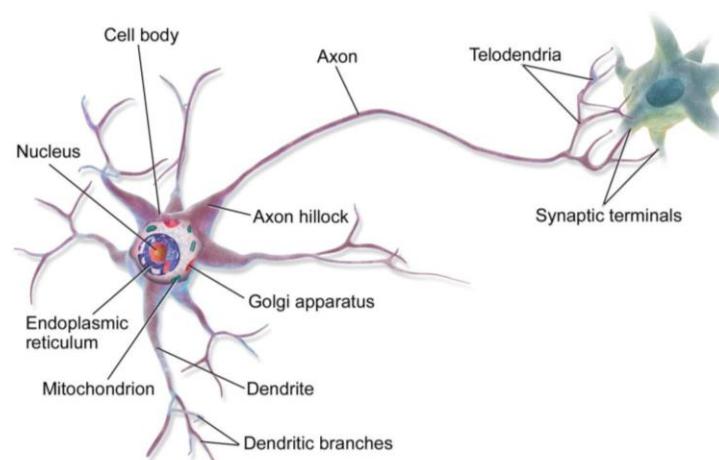


Figura 2 – Elementos constituintes de um neurônio

Extraído de: [[http://upload.wikimedia.org/wikipedia/commons/thumb/a/a9/Complete\\_neuron\\_cell\\_diagram\\_en.svg/1280px-Complete\\_neuron\\_cell\\_diagram\\_en.svg.png](http://upload.wikimedia.org/wikipedia/commons/thumb/a/a9/Complete_neuron_cell_diagram_en.svg/1280px-Complete_neuron_cell_diagram_en.svg.png)]

# Redes Neurais

- A sinapse libera substâncias químicas chamadas de neurotransmissores, em função do pulso elétrico disparado pelo axônio;
- O fluxo de neurotransmissores nas sinapses pode ter um efeito excitatório ou inbitório sobre o neurônio receptor



Destaque para os telodendros (ramificações do axônio)

Fonte: [https://en.wikipedia.org/wiki/Neural\\_circuit](https://en.wikipedia.org/wiki/Neural_circuit)

# Redes Neurais

- O aprendizado ocorre por sucessivas modificações nas sinapses que interconectam os neurônios, em função da maior ou menor liberação de neurotransmissores;
- À medida que novos eventos ocorrem, determinadas ligações entre neurônios são reforçadas, enquanto outras enfraquecidas;
- Este ajuste nas ligações entre os neurônios é uma das características das redes neurais artificiais.

# Redes Neurais

- Modelo matemático: Simplificações da realidade com o propósito de representar aspectos relevantes de um sistema em estudo, sendo que detalhes de menor significância são descartados para viabilizar a modelagem.

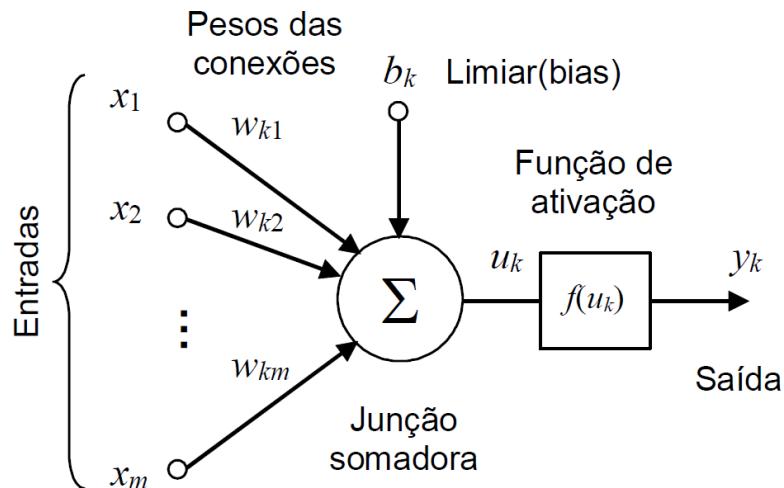


Figura 7 – Modelo matemático de um neurônio artificial

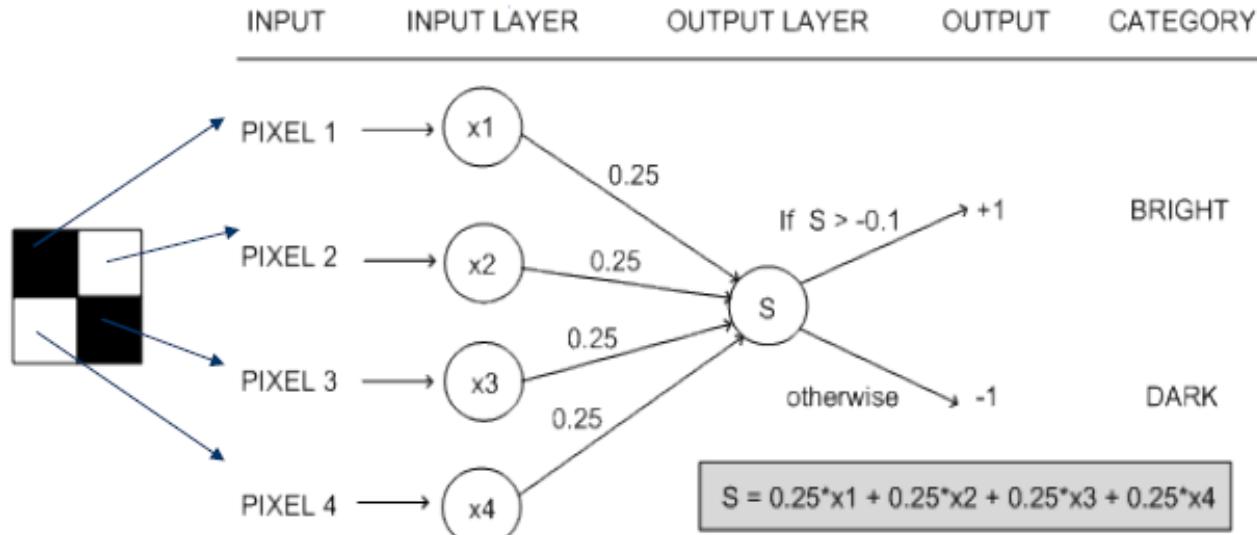
- A saída do neurônio  $k$  pode ser descrita por:  $y_k = f(u_k) = f\left(\sum_{j=1}^m w_{kj}x_j + b_k\right)$

# Perceptrons: exemplo

## Classificação de imagens preto e branco por uma matriz de pixel 2x2

- Em “clara” ou “escura”;
- Pixels brancos = 1 e pixel pretos = -1.
- Pode se representar o problema por essa regra:
  - Se resultado da Rede  $> -0.1$  é clara; caso contrário escura;
- Arquitetura do Perceptron:
  - 4 unidades de entrada, uma para cada pixel da matriz;
  - Uma unidade de saída: +1 para “clara”; -1 “escura”.

# Perceptrons: exemplo

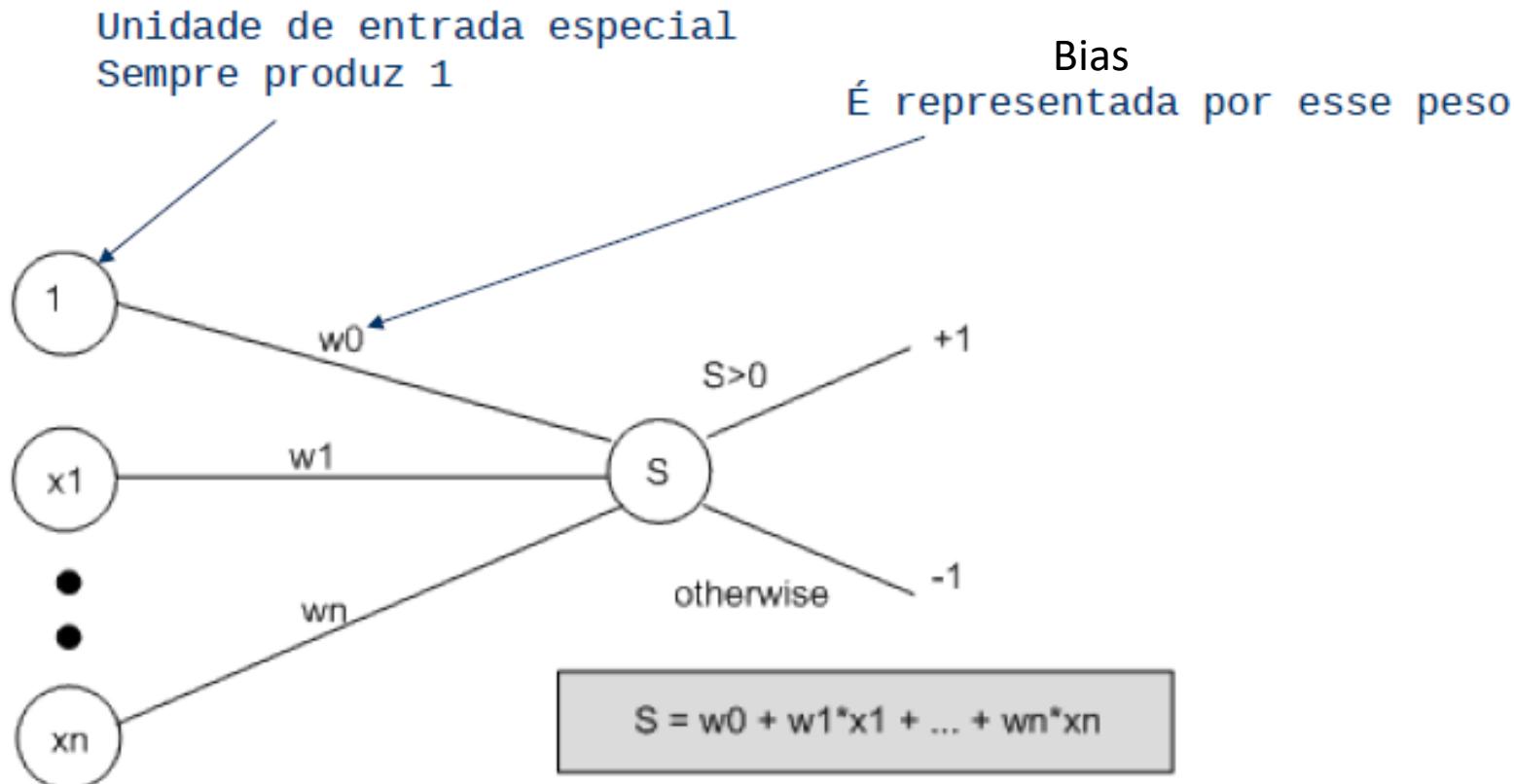


- Exemplo de entrada:  $x_1=-1, x_2=1, x_3=1, x_4=-1$ 
  - $S = 0.25 \cdot (-1) + 0.25 \cdot (1) + 0.25 \cdot (-1) + 0.25 \cdot (1) = 0$
- $0 > -0.1$ , portanto a saída para a rede é +1
  - A imagem é classificada como “clara”

# Aprendizagem em Perceptrons

- É necessário aprender:
  - Os pesos entre as unidades de entrada e saída;
  - O valor do bias.
- Para tornar os cálculos mais fáceis:
  - Considera-se o bias como um peso referente a uma unidade de entrada especial, cujo sinal é sempre 1 (ou -1);
    - Agora, o único objetivo resume-se a aprender os pesos da rede.

# Aprendizagem em Perceptrons



# Redes Neurais

## O problema do OU-exclusivo em MLP

- Considere os pontos  $(0,0), (0,1), (1,0)$  e  $(1,1)$  no plano  $\mathbb{R}^2$ , conforme apresentado na Figura 26. O objetivo é determinar uma rede com duas entradas  $x_i \in \{0,1\}$  ( $i=1,2$ ), e uma saída  $y \in \{0,1\}$  de maneira que:  $\begin{cases} (x_1, x_2) = (0,0) \text{ ou } (1,1) \Rightarrow y = 0 \\ (x_1, x_2) = (1,0) \text{ ou } (0,1) \Rightarrow y = 1 \end{cases}$

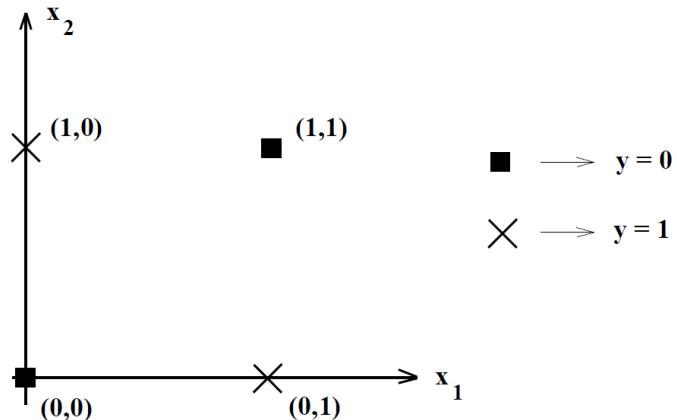
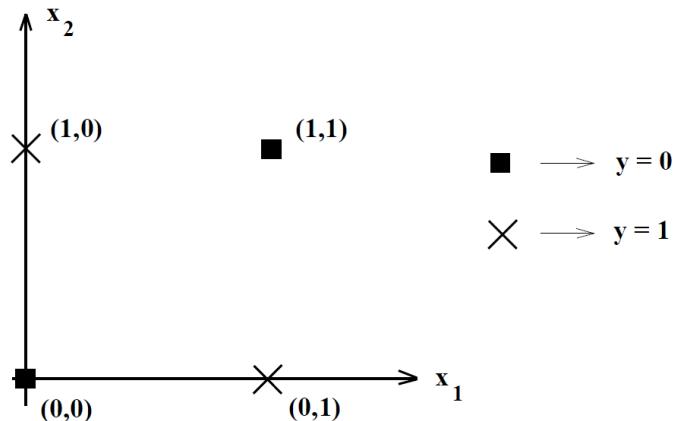


Figura 26 – O problema do OU-exclusivo

# Redes Neurais

## O problema do OU-exclusivo em MLP

- Considere os pontos  $(0,0), (0,1), (1,0)$  e  $(1,1)$  no plano  $\mathbb{R}^2$ , conforme apresentado na Figura 26. O objetivo é determinar uma rede com duas entradas  $x_i \in \{0,1\}$  ( $i=1,2$ ), e uma saída  $y \in \{0,1\}$  de maneira que:  $\begin{cases} (x_1, x_2) = (0,0) \text{ ou } (1,1) \Rightarrow y = 0 \\ (x_1, x_2) = (1,0) \text{ ou } (0,1) \Rightarrow y = 1 \end{cases}$



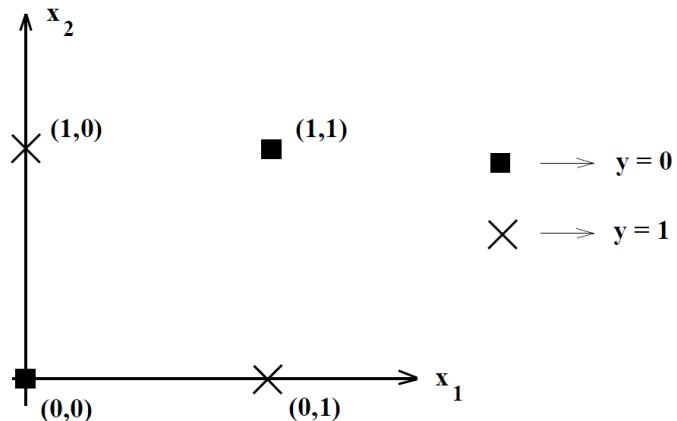
		AND	OR	XOR
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

Figura 26 – O problema do OU-exclusivo

# Redes Neurais

## O problema do OU-exclusivo em MLP

- Considere os pontos  $(0,0), (0,1), (1,0)$  e  $(1,1)$  no plano  $\mathbb{R}^2$ , conforme apresentado na Figura 26. O objetivo é determinar uma rede com duas entradas  $x_i \in \{0,1\}$  ( $i=1,2$ ), e uma saída  $y \in \{0,1\}$  de maneira que:  $\begin{cases} (x_1, x_2) = (0,0) \text{ ou } (1,1) \Rightarrow y = 0 \\ (x_1, x_2) = (1,0) \text{ ou } (0,1) \Rightarrow y = 1 \end{cases}$



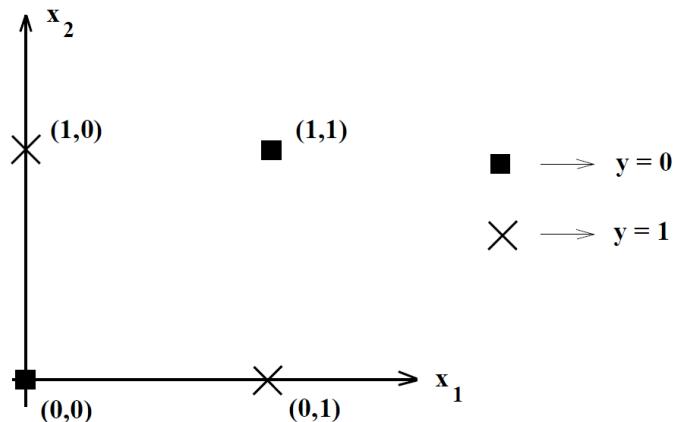
		AND	OR	XOR
0	0	0		
0	1	0		
1	0	0		
1	1	1		

Figura 26 – O problema do OU-exclusivo

# Redes Neurais

## O problema do OU-exclusivo em MLP

- Considere os pontos  $(0,0), (0,1), (1,0)$  e  $(1,1)$  no plano  $\mathbb{R}^2$ , conforme apresentado na Figura 26. O objetivo é determinar uma rede com duas entradas  $x_i \in \{0,1\}$  ( $i=1,2$ ), e uma saída  $y \in \{0,1\}$  de maneira que:  $\begin{cases} (x_1, x_2) = (0,0) \text{ ou } (1,1) \Rightarrow y = 0 \\ (x_1, x_2) = (1,0) \text{ ou } (0,1) \Rightarrow y = 1 \end{cases}$



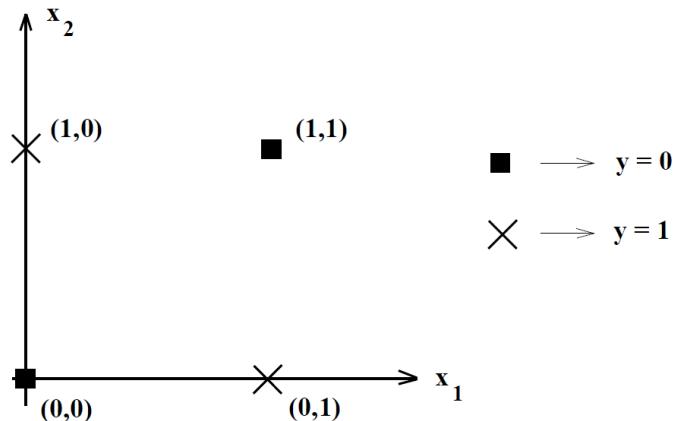
		AND	OR	XOR
0	0	0	0	
0	1	0	1	
1	0	0	1	
1	1	1	1	

Figura 26 – O problema do OU-exclusivo

# Redes Neurais

## O problema do OU-exclusivo em MLP

- Considere os pontos  $(0,0), (0,1), (1,0)$  e  $(1,1)$  no plano  $\mathbb{R}^2$ , conforme apresentado na Figura 26. O objetivo é determinar uma rede com duas entradas  $x_i \in \{0,1\}$  ( $i=1,2$ ), e uma saída  $y \in \{0,1\}$  de maneira que:
 
$$\begin{cases} (x_1, x_2) = (0,0) \text{ ou } (1,1) \Rightarrow y = 0 \\ (x_1, x_2) = (1,0) \text{ ou } (0,1) \Rightarrow y = 1 \end{cases}$$

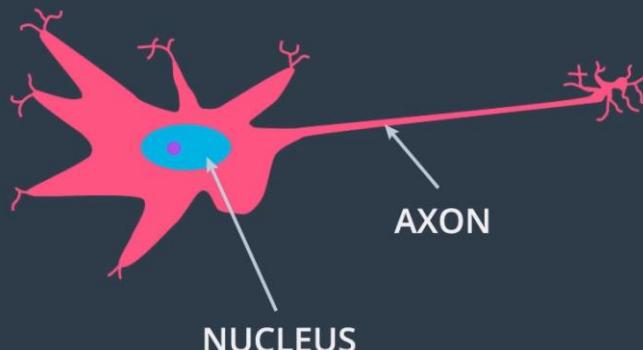
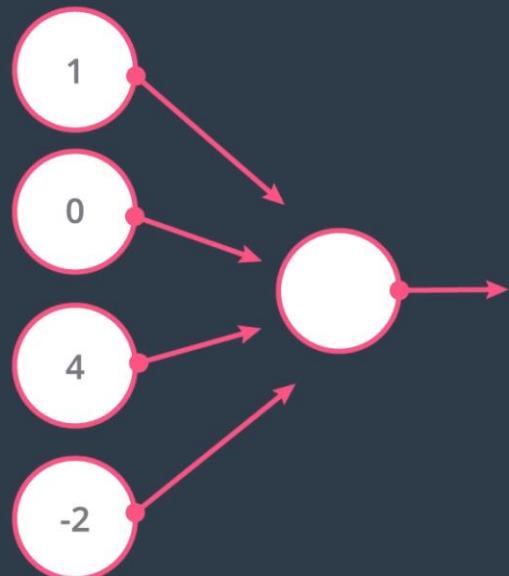


		AND	OR	XOR
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

Figura 26 – O problema do OU-exclusivo

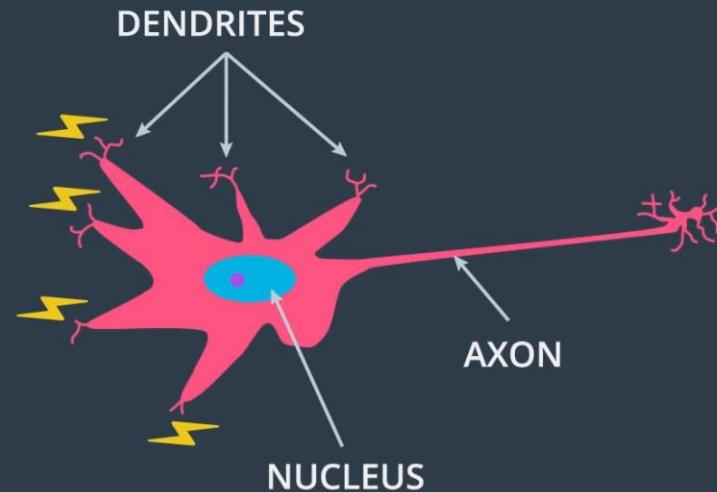
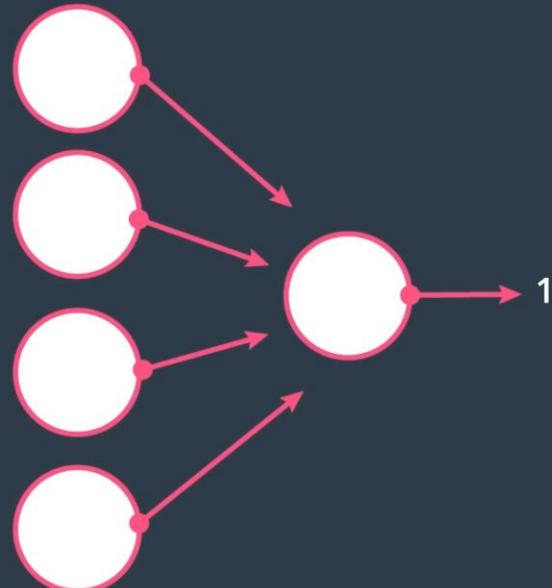
# Redes Neurais

## Perceptron



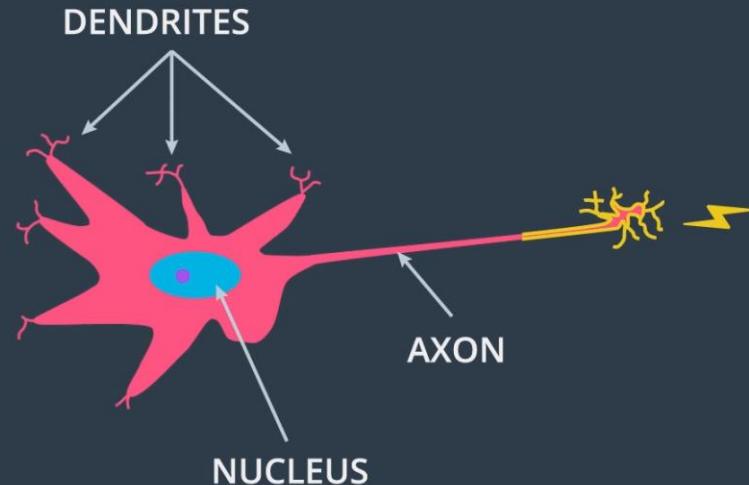
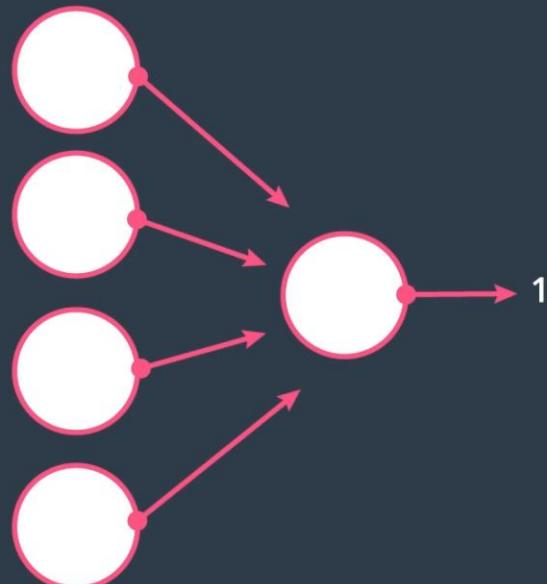
# Redes Neurais

Perceptron



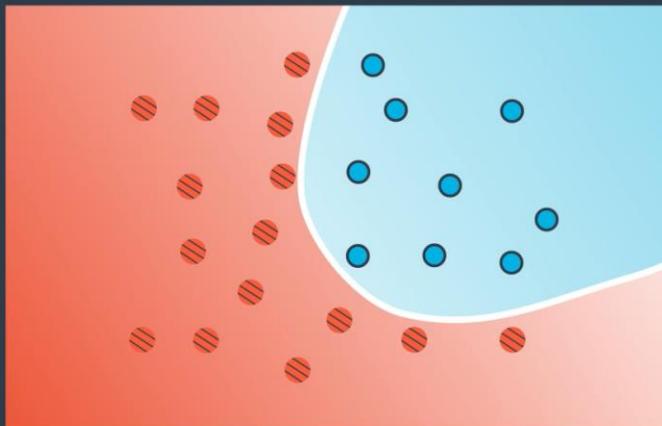
# Redes Neurais

Perceptron



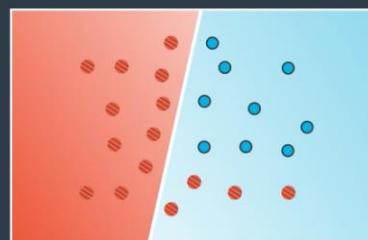
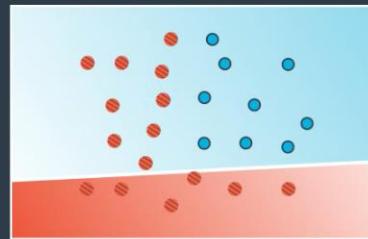
# Redes Neurais - Arquitetura

Non-Linear Regions



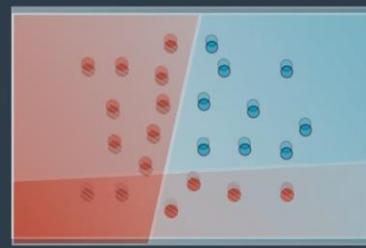
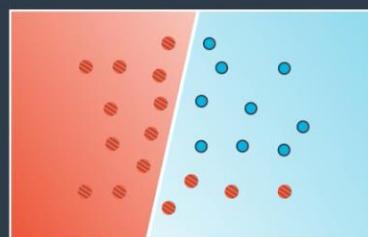
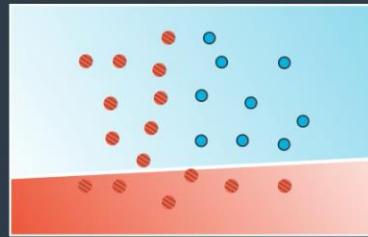
# Redes Neurais - Arquitetura

## Combining Regions



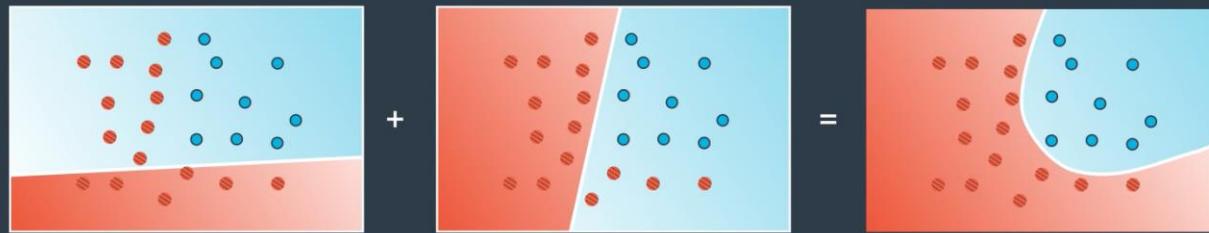
# Redes Neurais - Arquitetura

Combining Regions



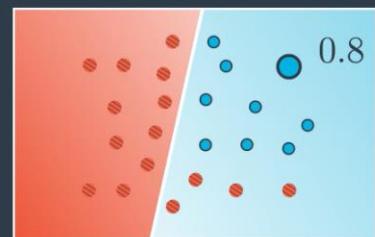
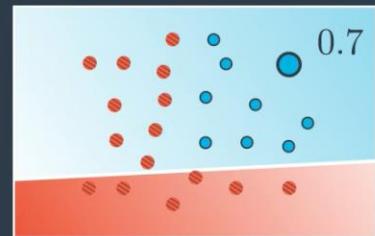
# Redes Neurais - Arquitetura

Combining Regions



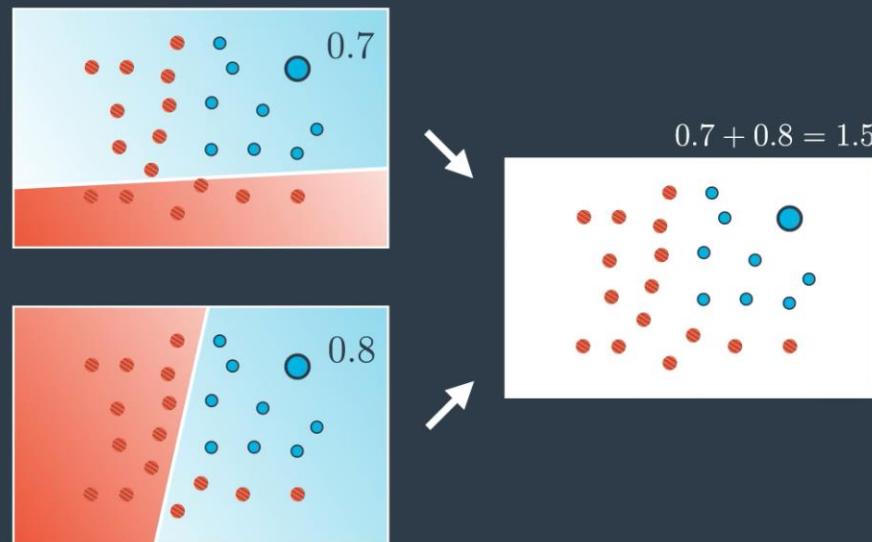
# Redes Neurais - Arquitetura

Neural Network



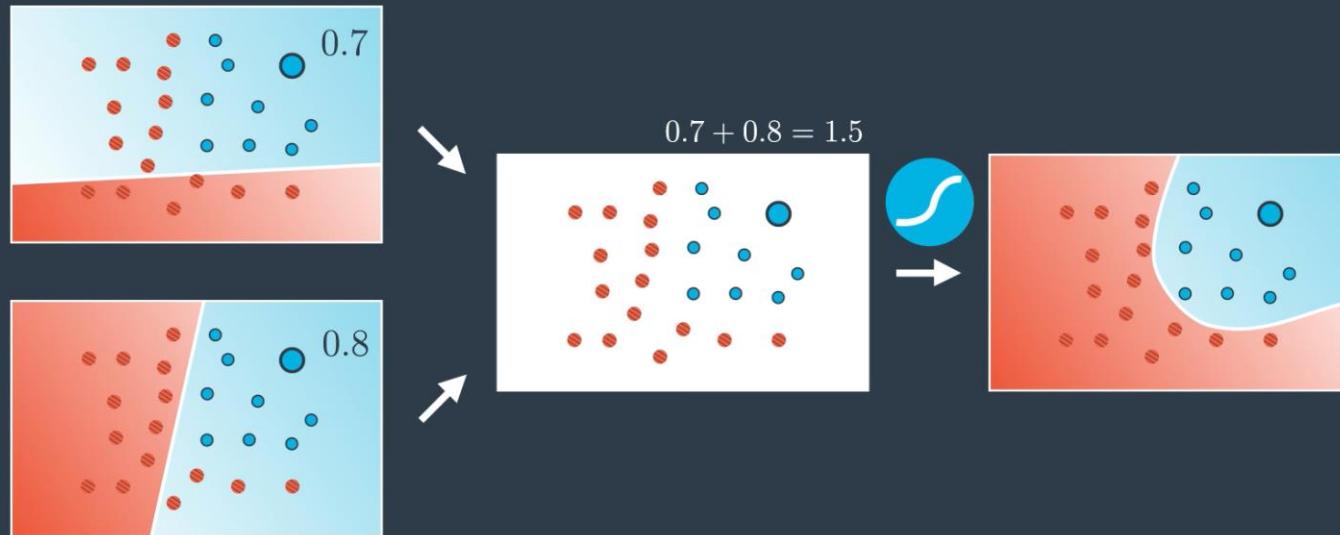
# Redes Neurais - Arquitetura

Neural Network

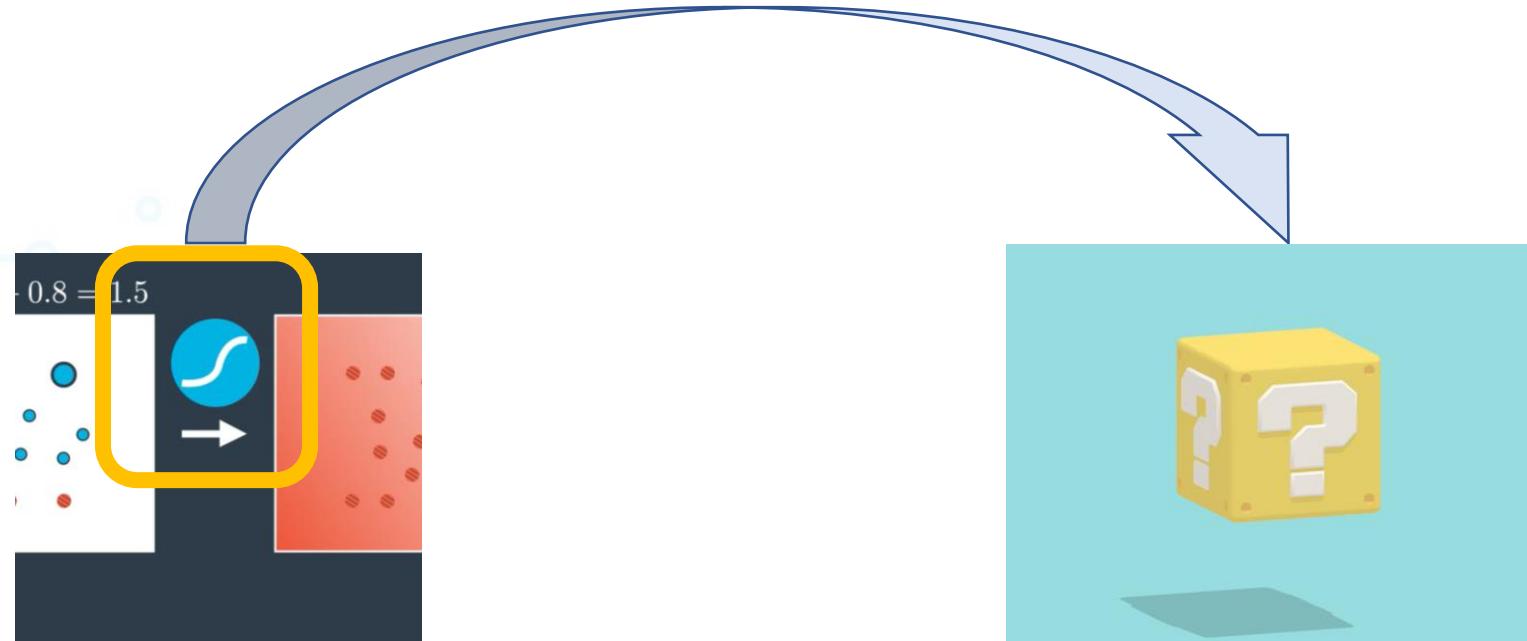


# Redes Neurais - Arquitetura

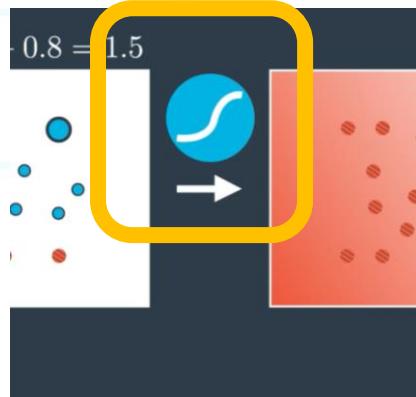
Neural Network



# Redes Neurais - Arquitetura

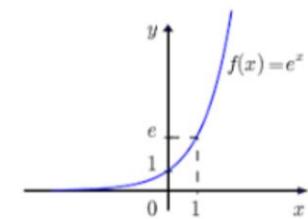


# Redes Neurais - Arquitetura



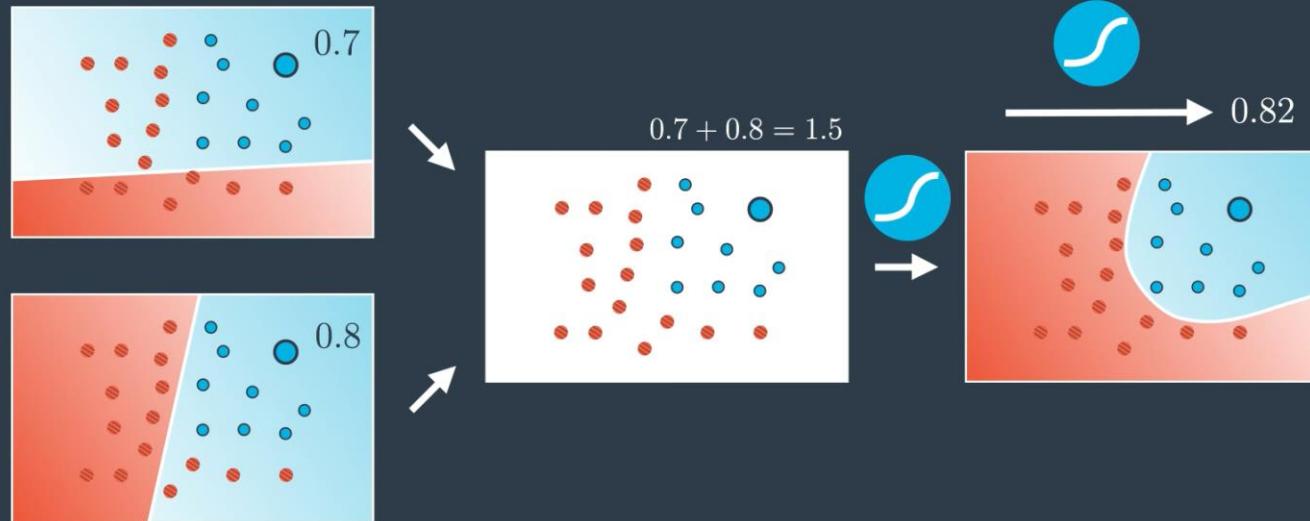
$$f(x) = \frac{1}{1 + e^{-x}} \text{ para todo } x \text{ real.}$$

A função **exponencial** natural, denotada  $e^x$  ou  $\exp(x)$  é a função **exponencial** cuja base é o número de Euler (um número irracional que **vale** aproximadamente 2,718281828).



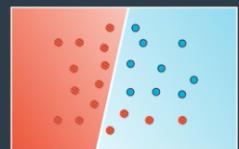
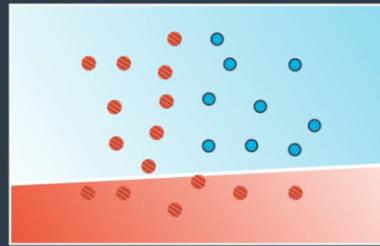
# Redes Neurais - Arquitetura

Neural Network



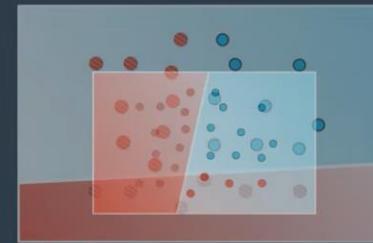
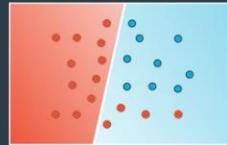
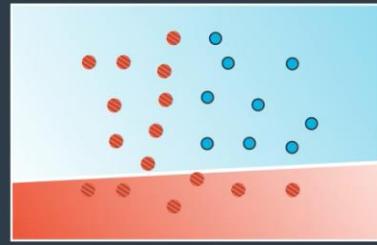
# Redes Neurais - Arquitetura

## Combining Regions



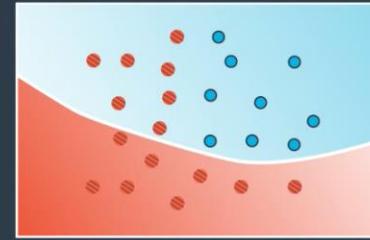
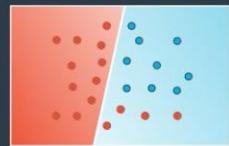
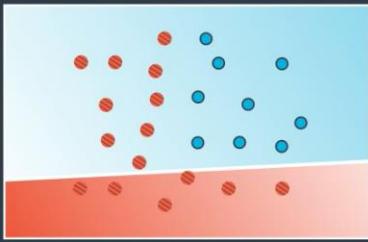
# Redes Neurais - Arquitetura

## Combining Regions



# Redes Neurais - Arquitetura

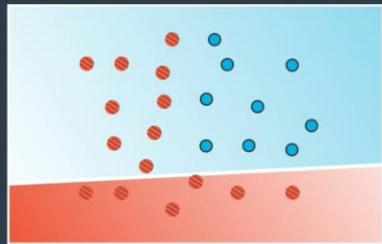
## Combining Regions



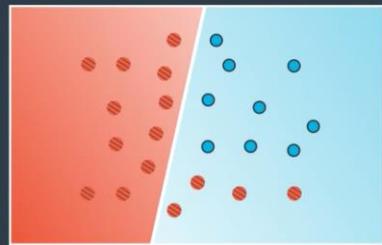
# Redes Neurais - Arquitetura

Neural Network

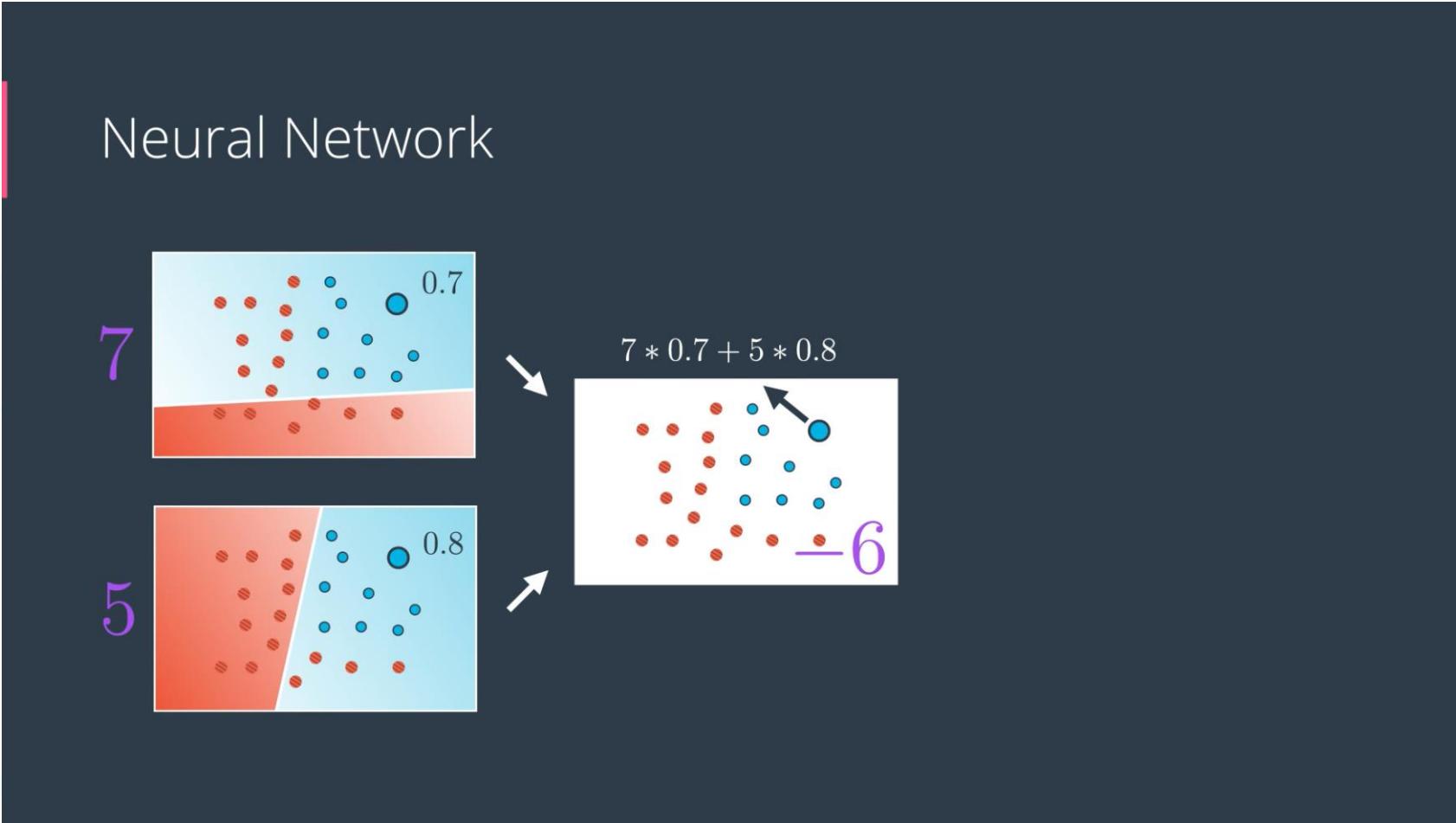
7



5

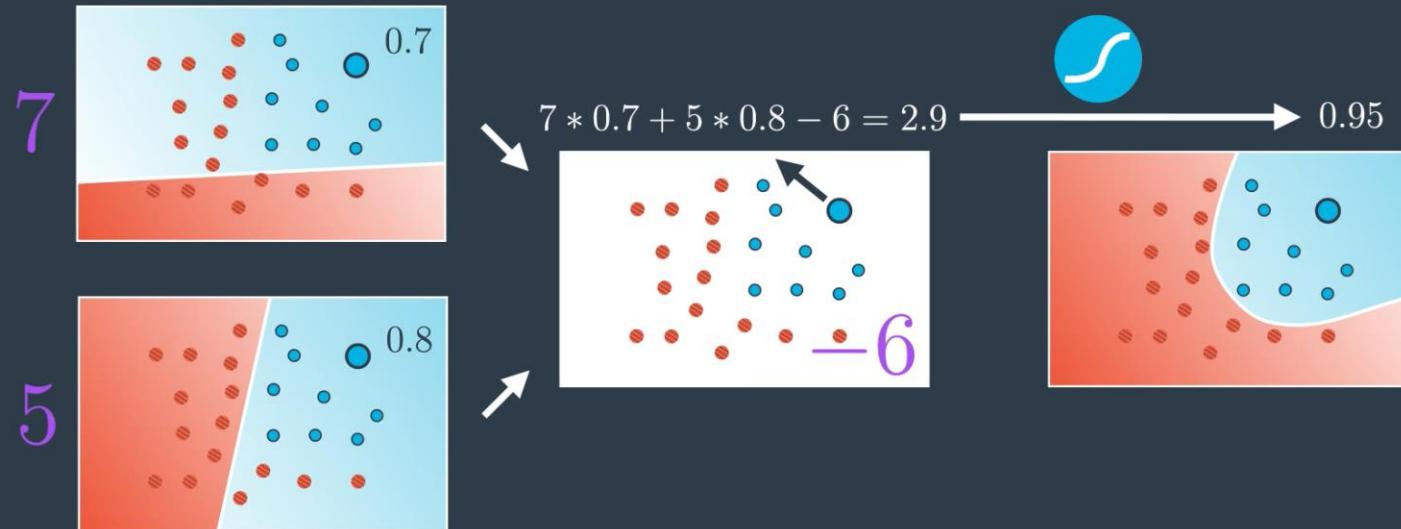


# Redes Neurais - Arquitetura



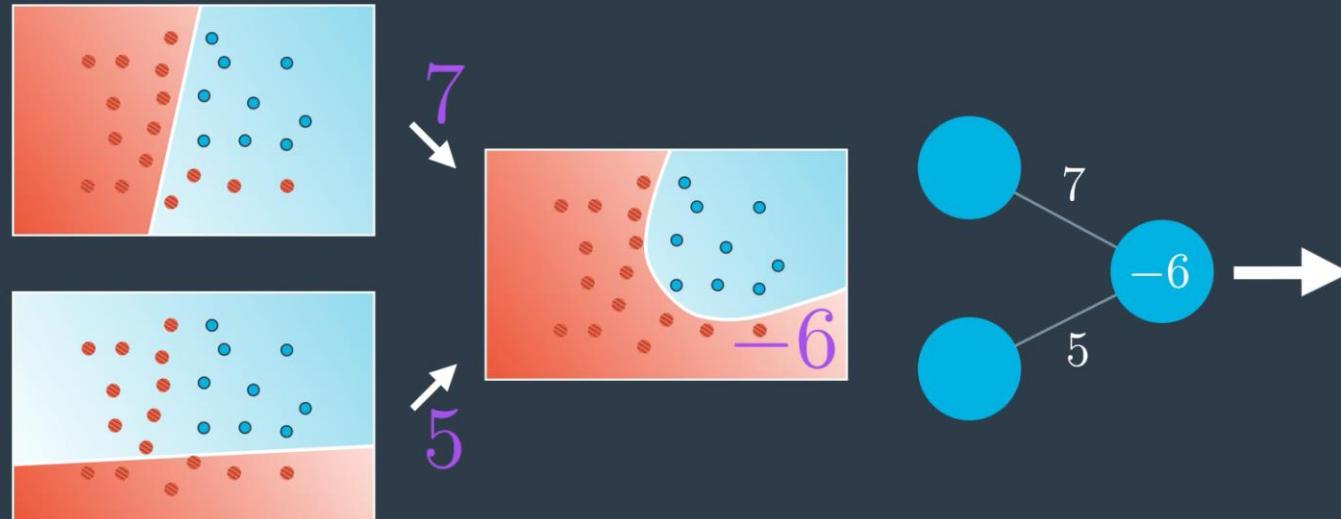
# Redes Neurais - Arquitetura

Neural Network



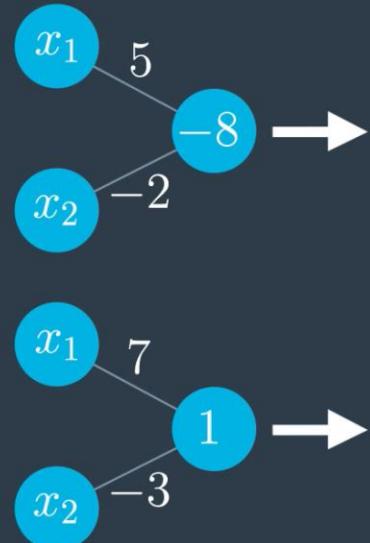
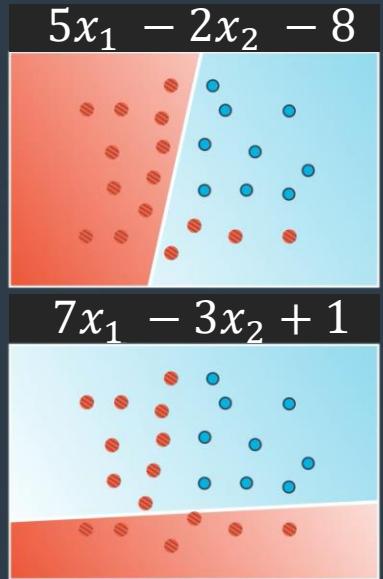
# Redes Neurais - Arquitetura

Neural Network

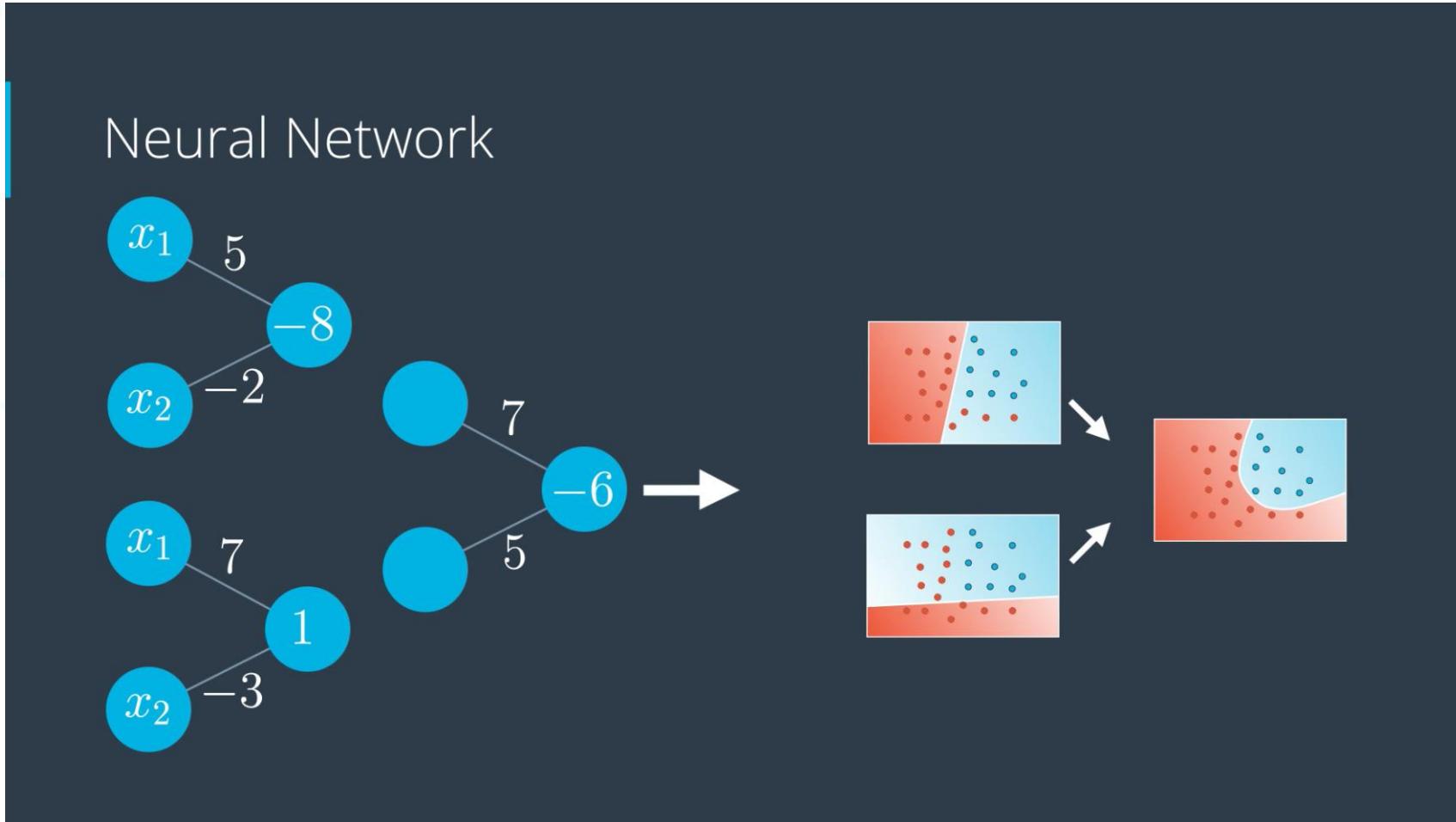


# Redes Neurais - Arquitetura

## Neural Network

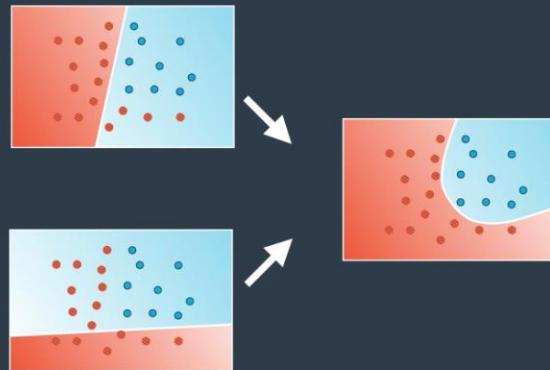
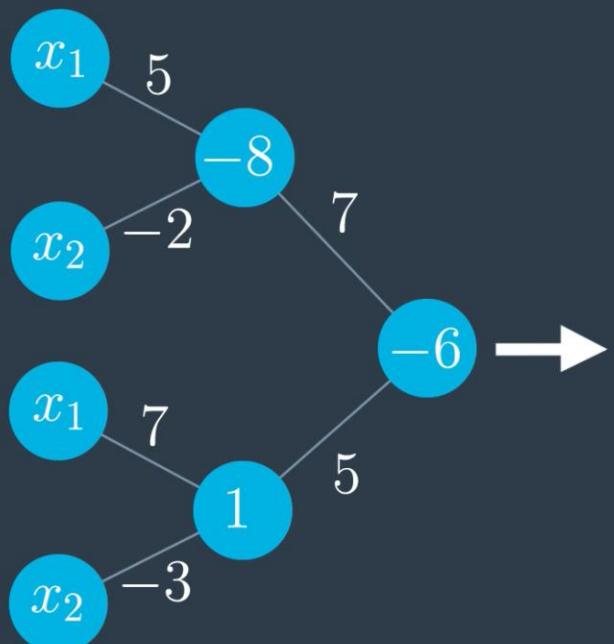


# Redes Neurais - Arquitetura



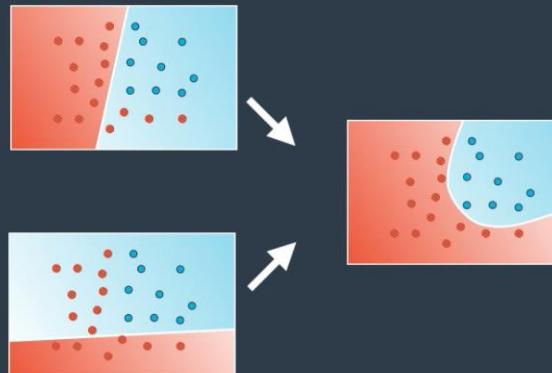
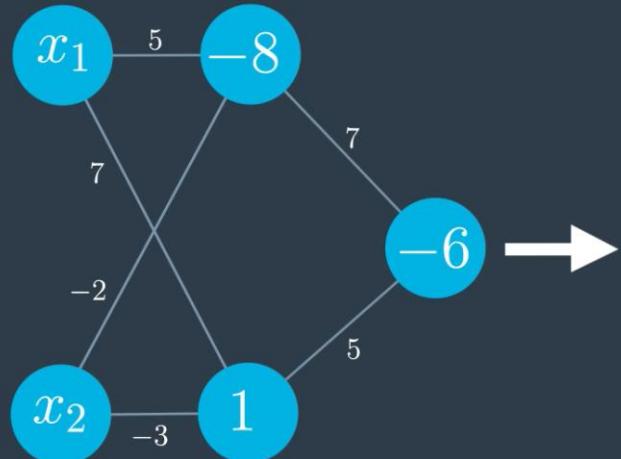
# Redes Neurais - Arquitetura

Neural Network



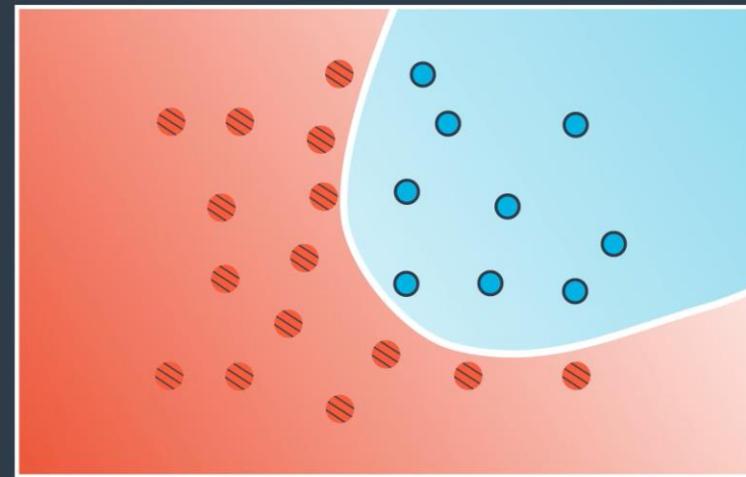
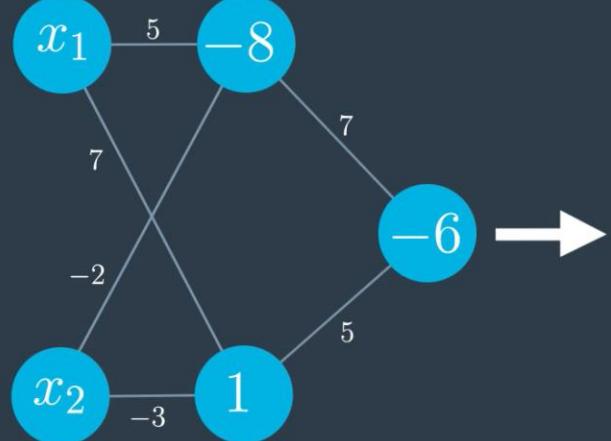
# Redes Neurais - Arquitetura

Neural Network

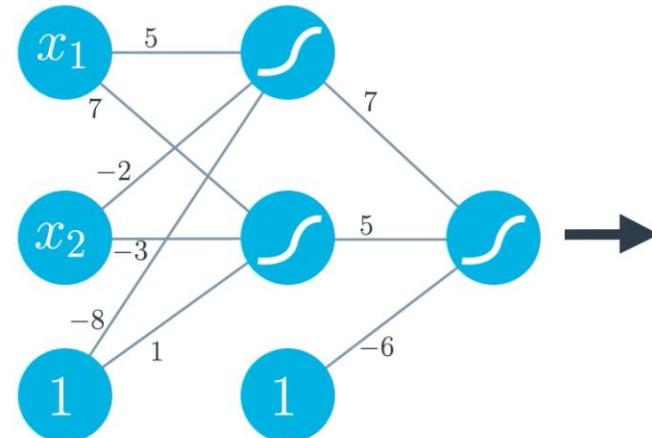
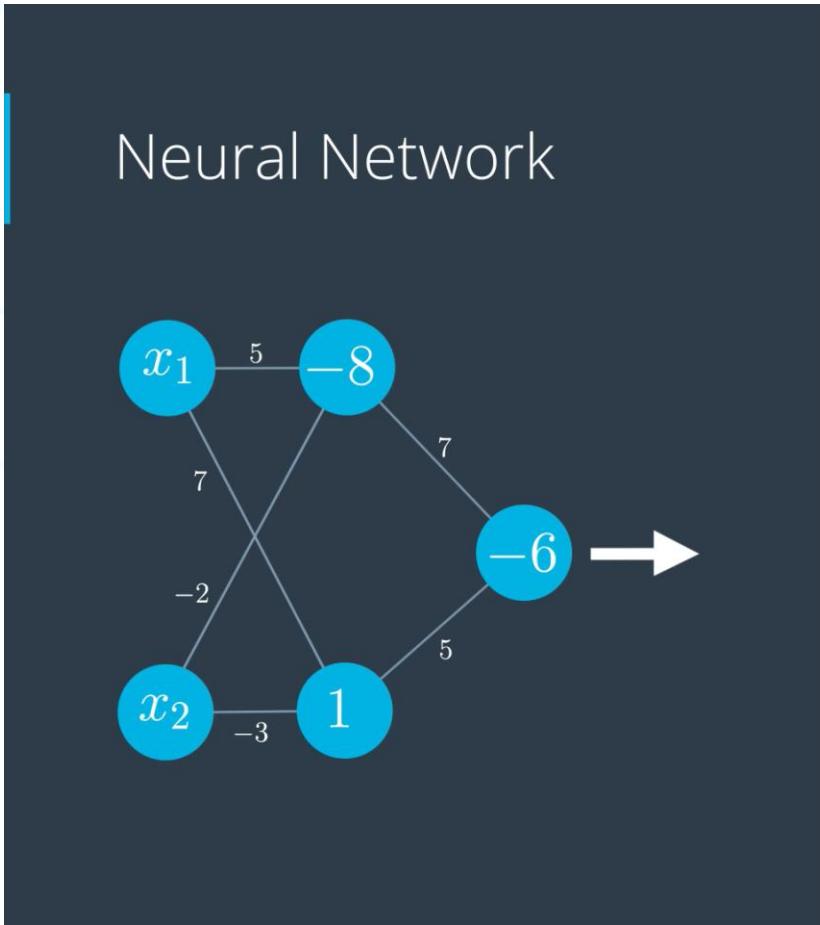


# Redes Neurais - Arquitetura

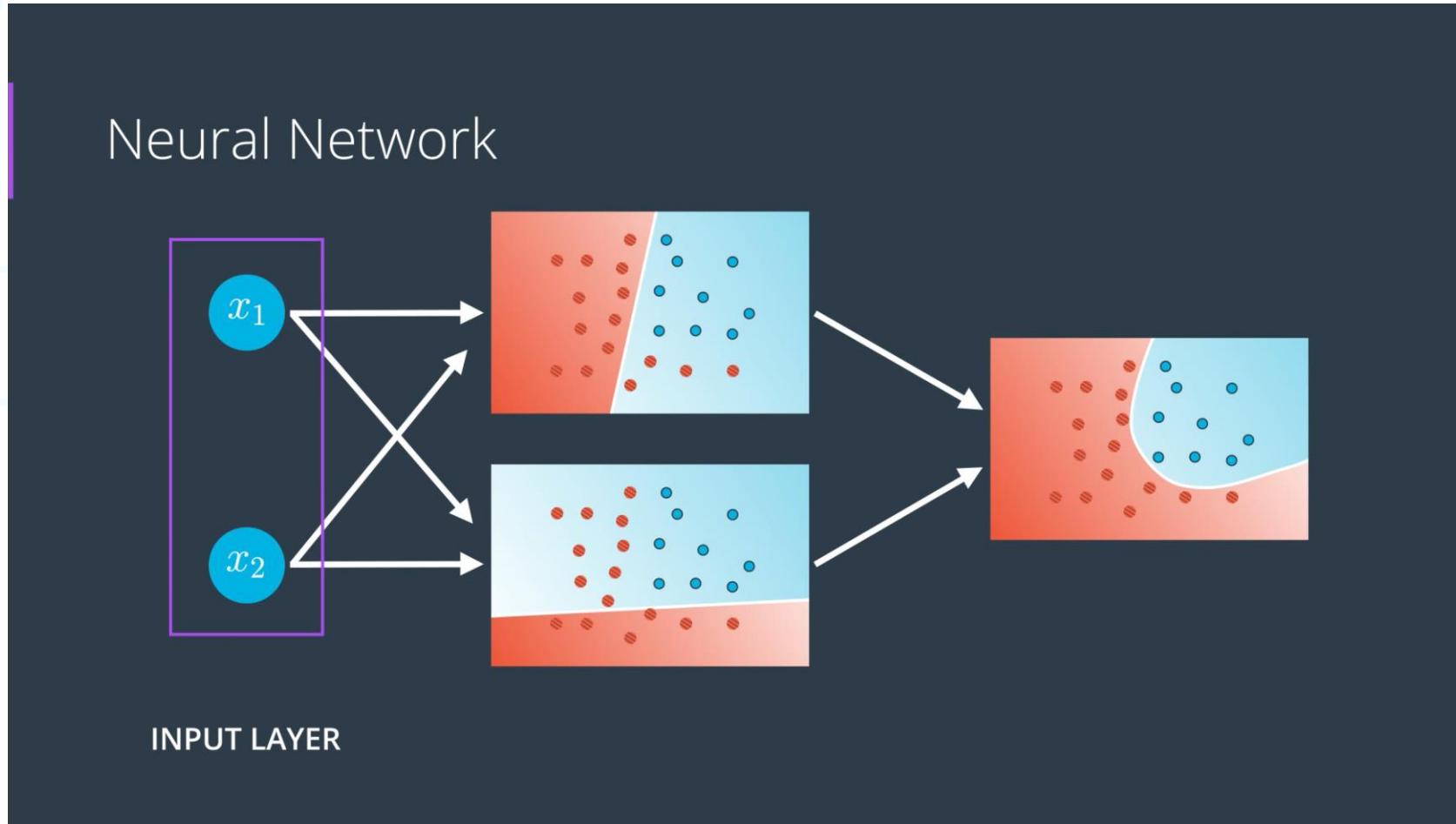
Neural Network



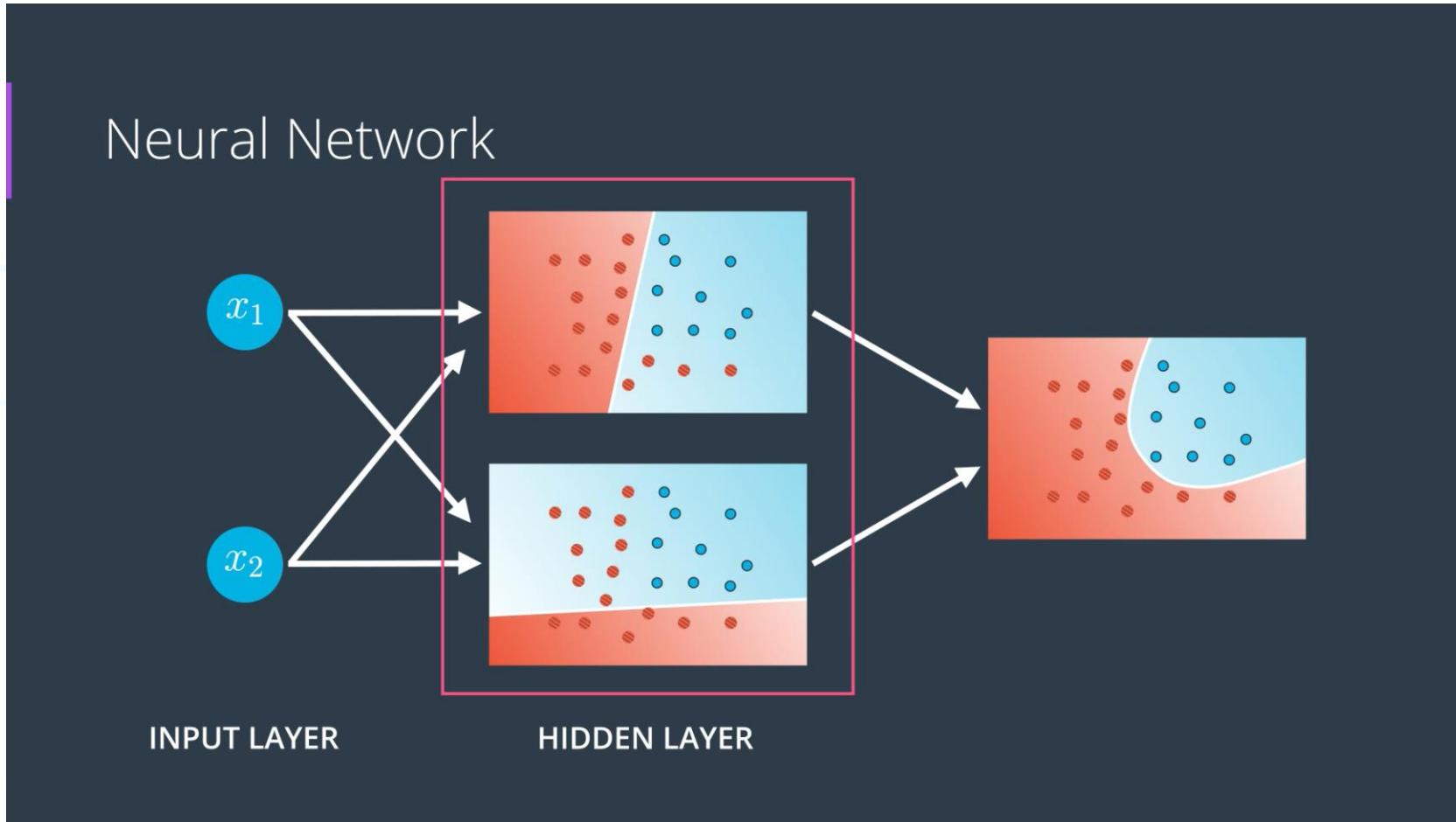
# Redes Neurais - Arquitetura



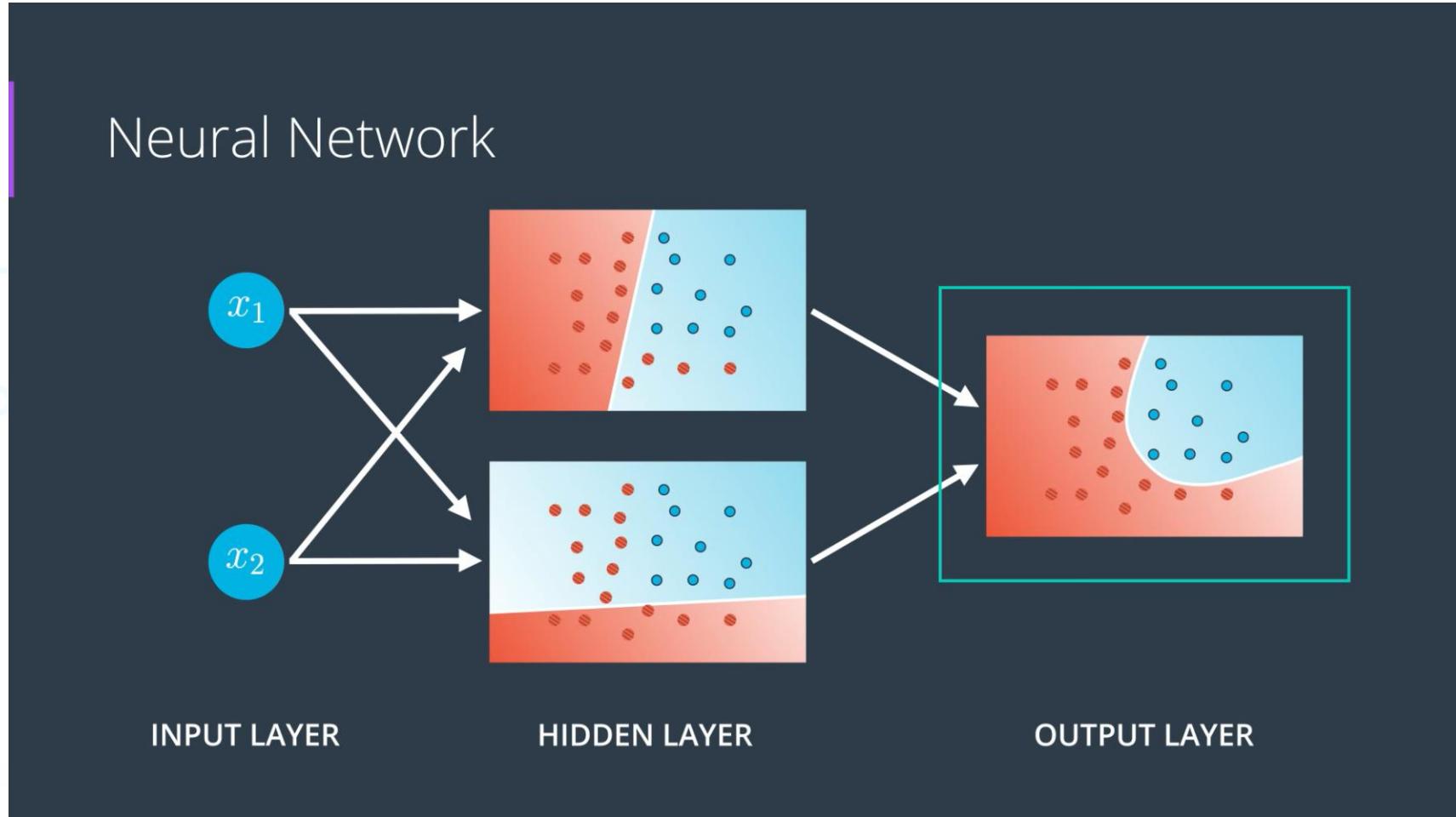
# Redes Neurais – Multiple Layers



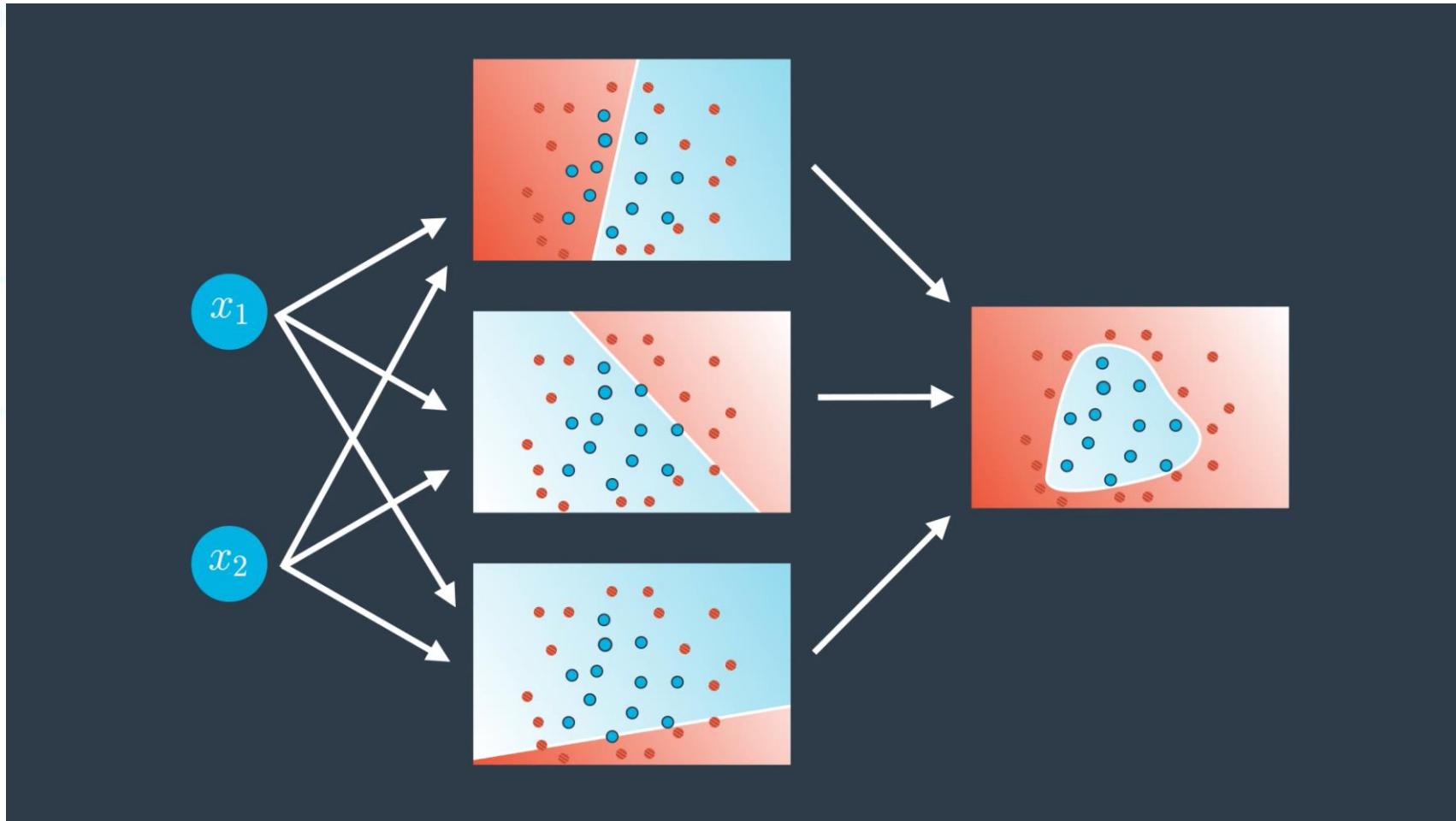
# Redes Neurais – Multiple Layers



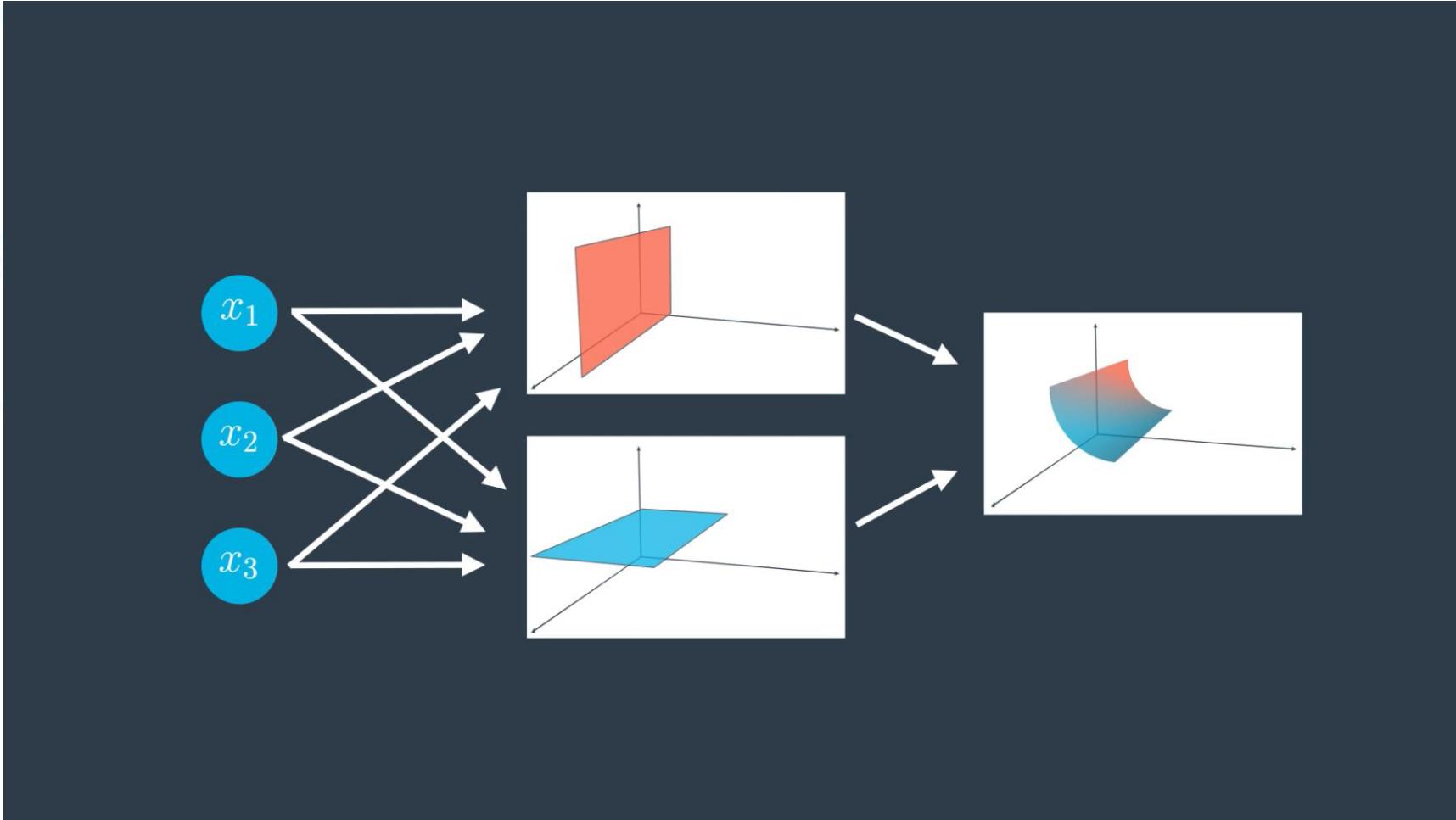
# Redes Neurais – Multiple Layers



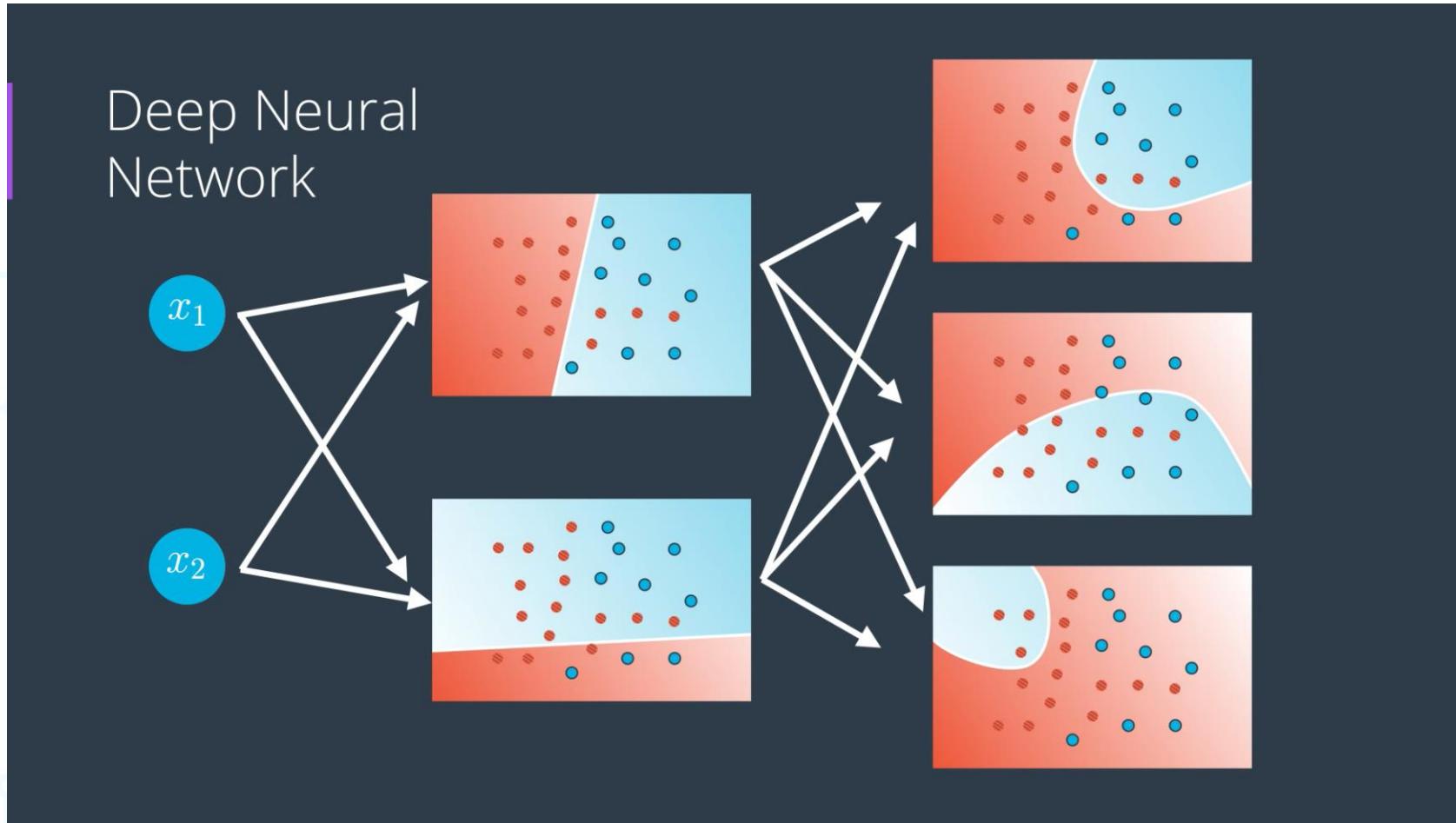
# Redes Neurais – Multiple Layers



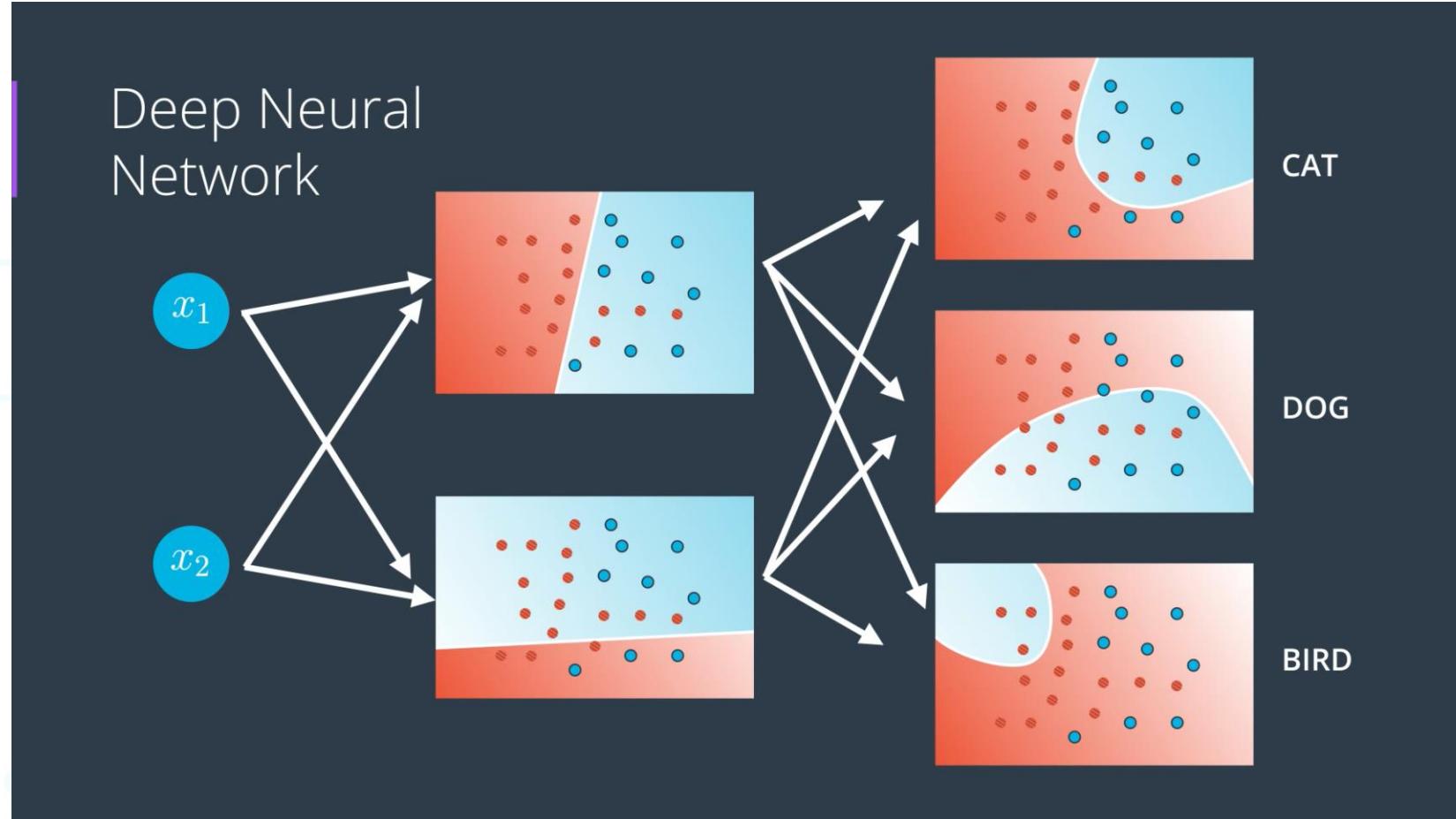
# Redes Neurais – Multiple Layers



# Redes Neurais – Multiple Layers

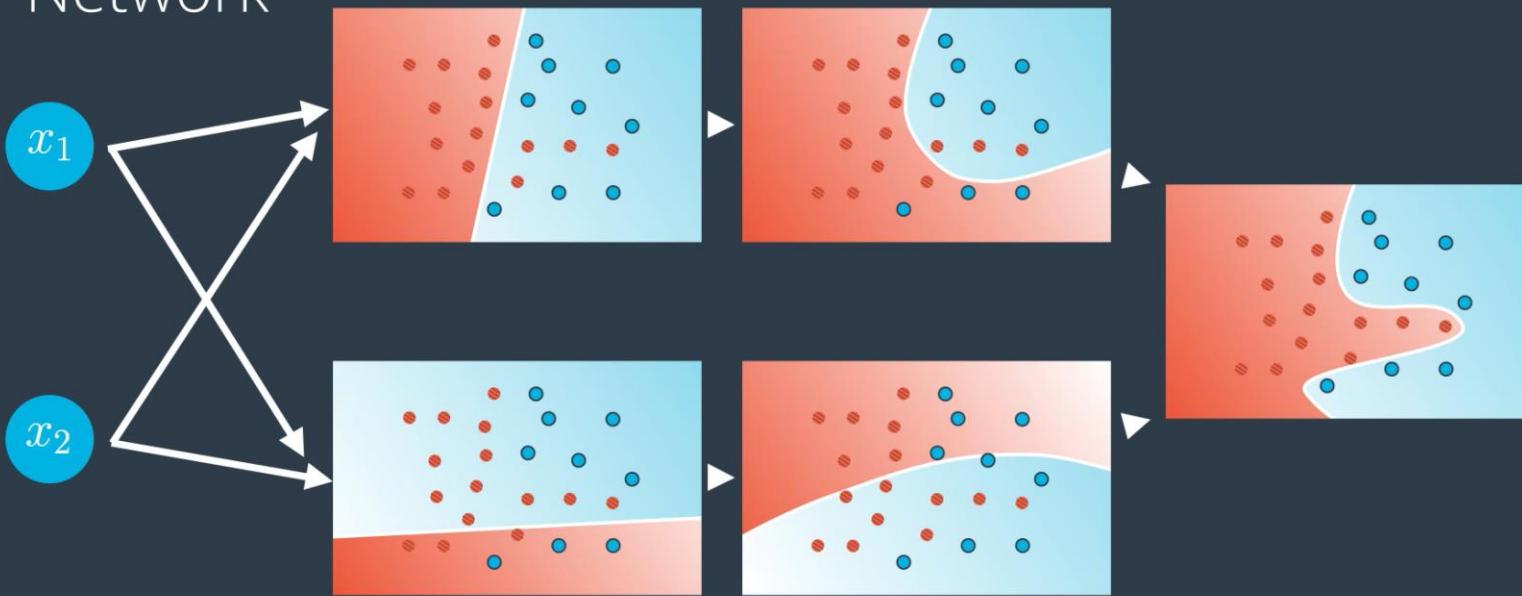


# Redes Neurais – Multiple Layers



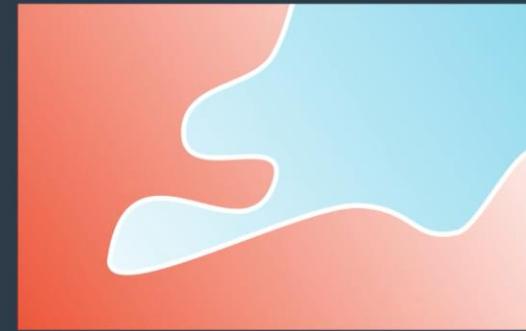
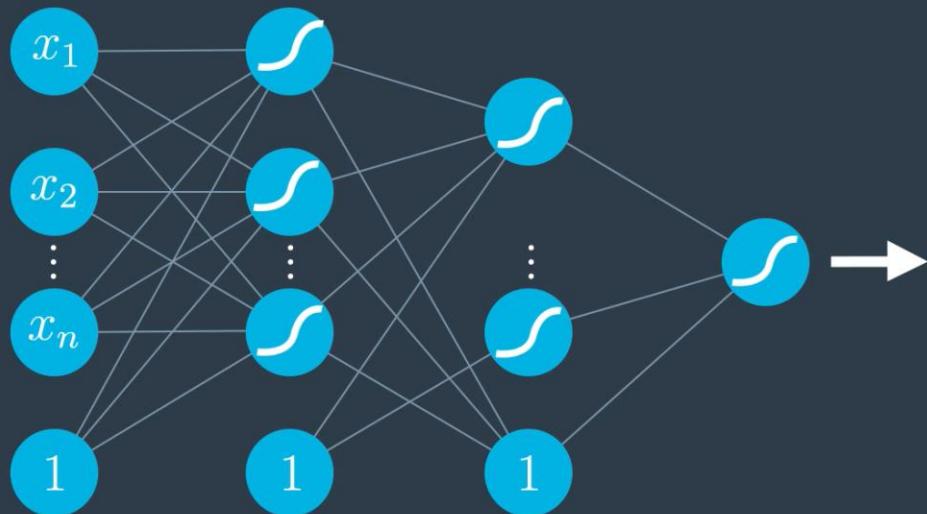
# Redes Neurais – Multiple Layers

Deep Neural Network



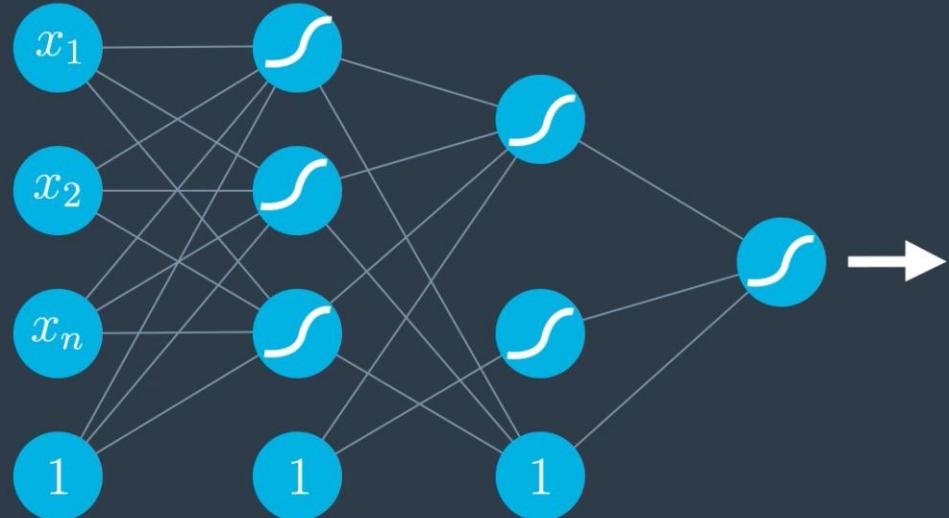
# Redes Neurais – Multiple Layers

Neural Network



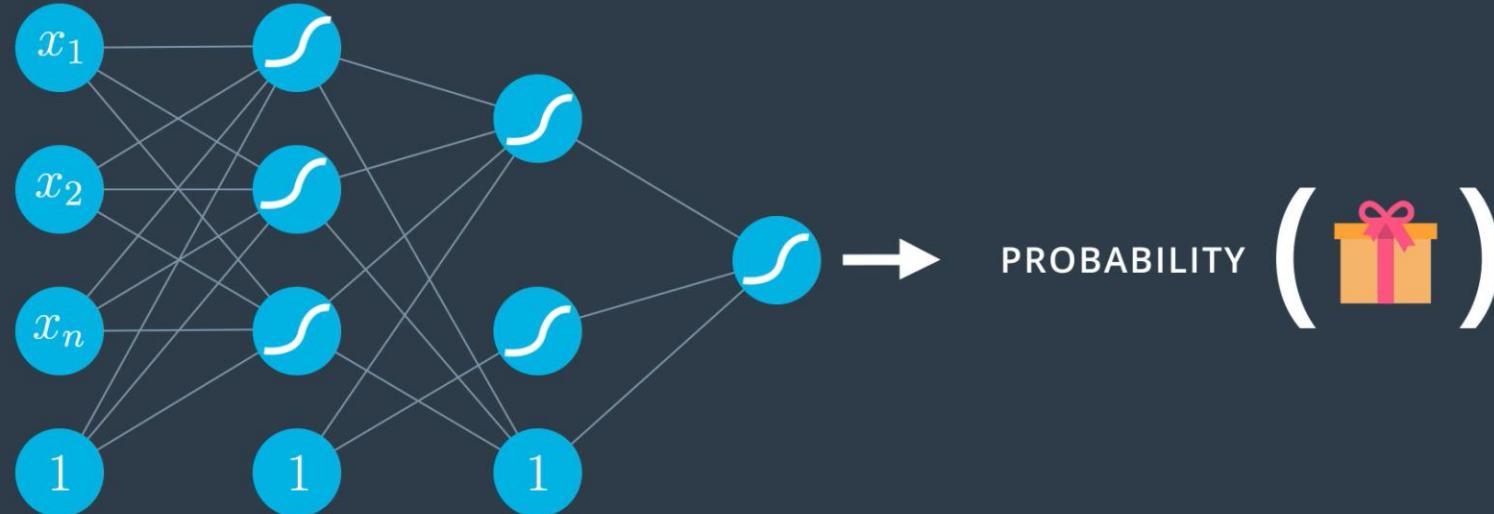
# Redes Neurais – Multi-Class classification

Binary Classification



# Redes Neurais – Multi-Class classification

Binary Classification



# Redes Neurais – Multi-Class classification

Multi-Class  
Classification

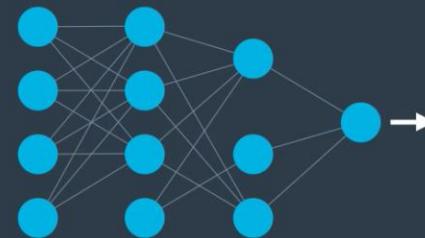
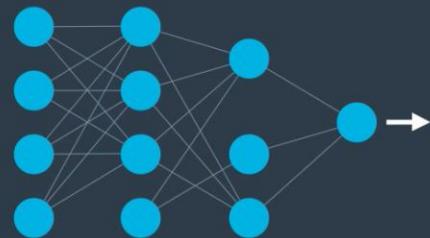
PROBABILITY 

PROBABILITY 

PROBABILITY 

# Redes Neurais – Multi-Class classification

## Multi-Class Classification

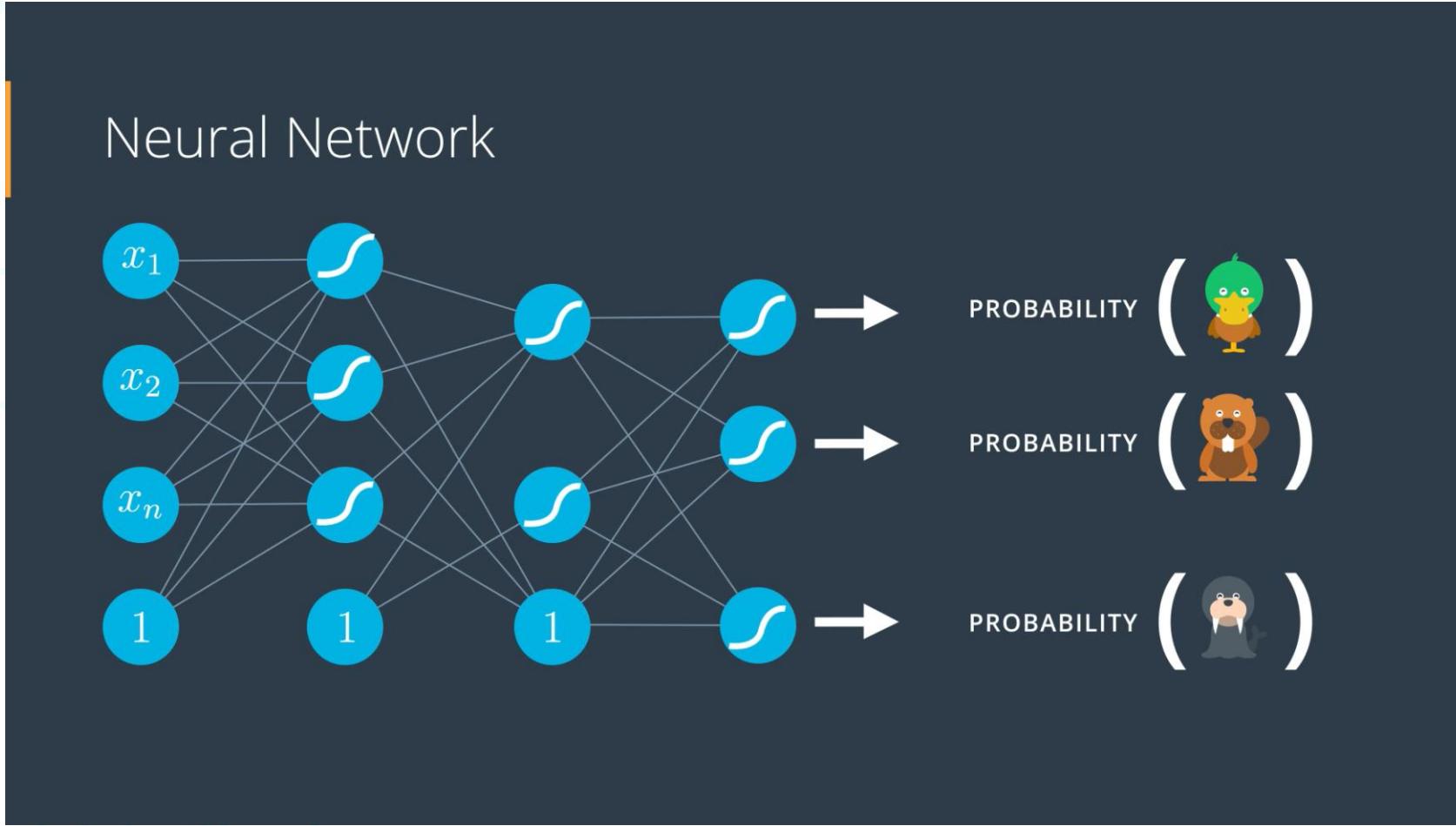


PROBABILITY 

PROBABILITY 

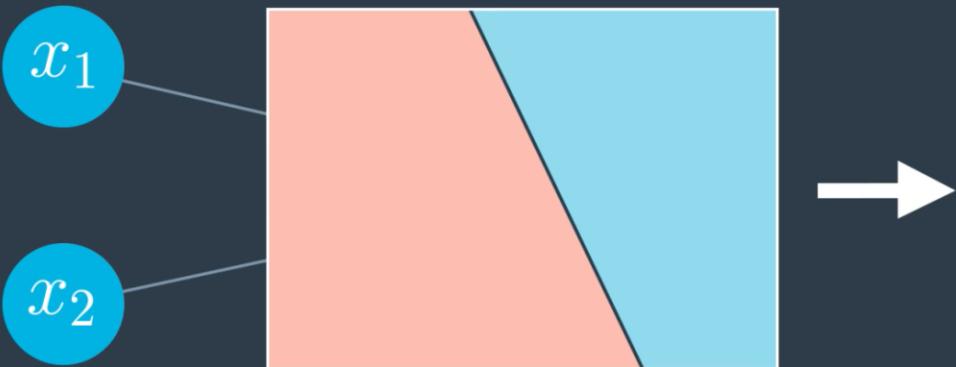
PROBABILITY 

# Redes Neurais – Multi-Class classification



# Redes Neurais – Feedforward

Perceptron

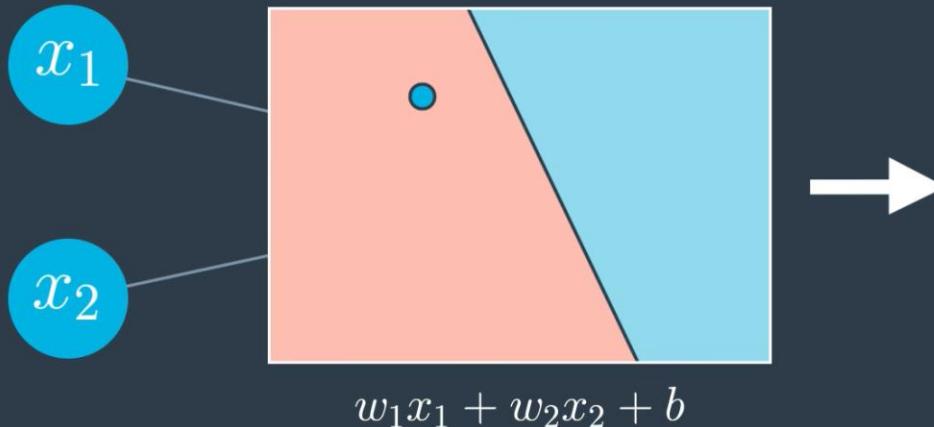


# Redes Neurais – Feedforward

## Perceptron

$$x = (x_1, x_2)$$

$$y = 1$$

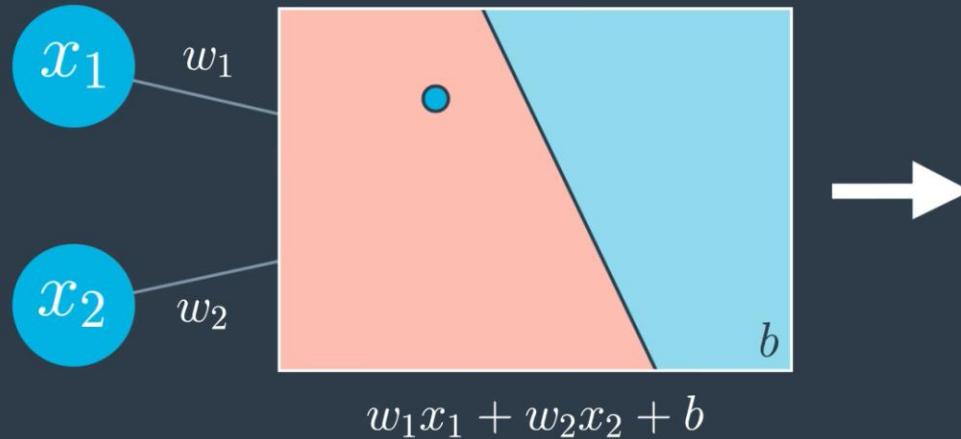


# Redes Neurais – Feedforward

## Perceptron

$$x = (x_1, x_2)$$

$$y = 1$$

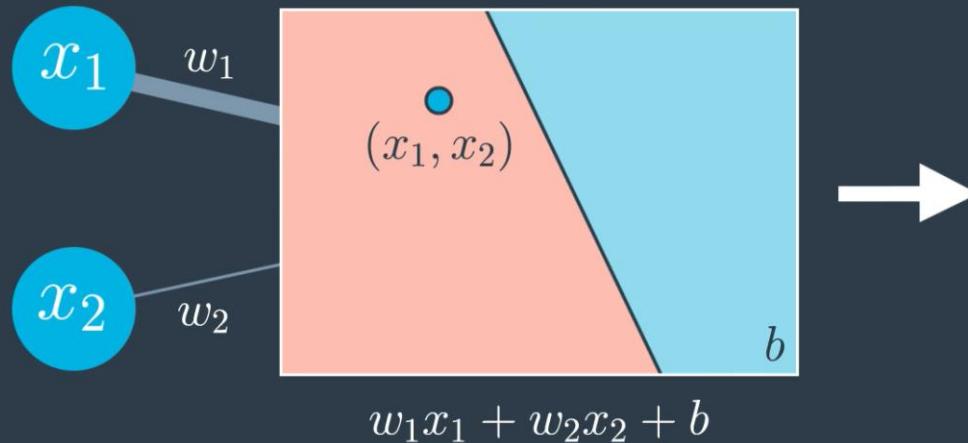


# Redes Neurais – Feedforward

## Perceptron

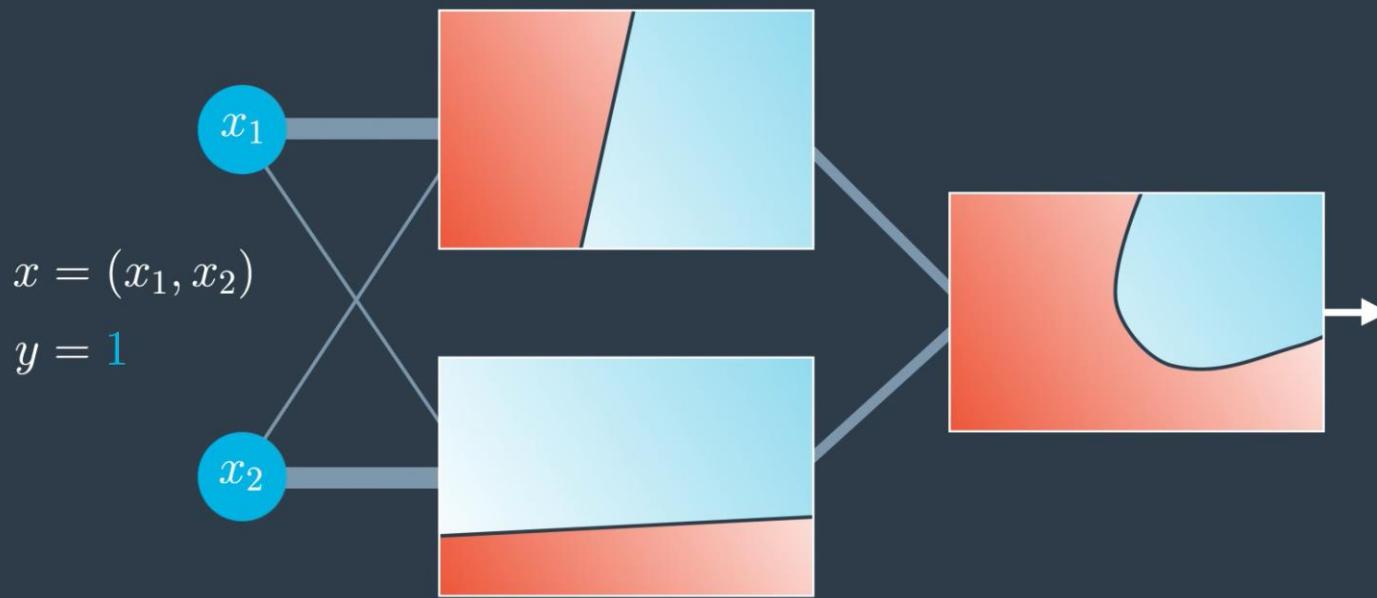
$$x = (x_1, x_2)$$

$$y = 1$$



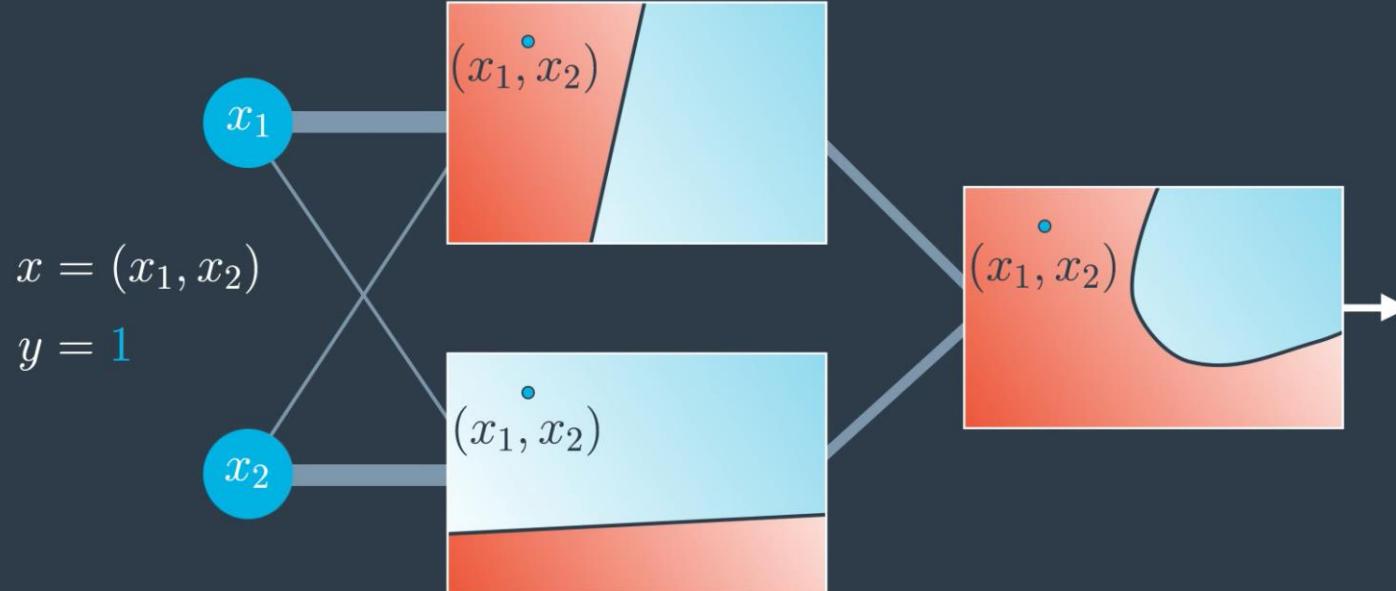
# Redes Neurais – Feedforward

Neural Network



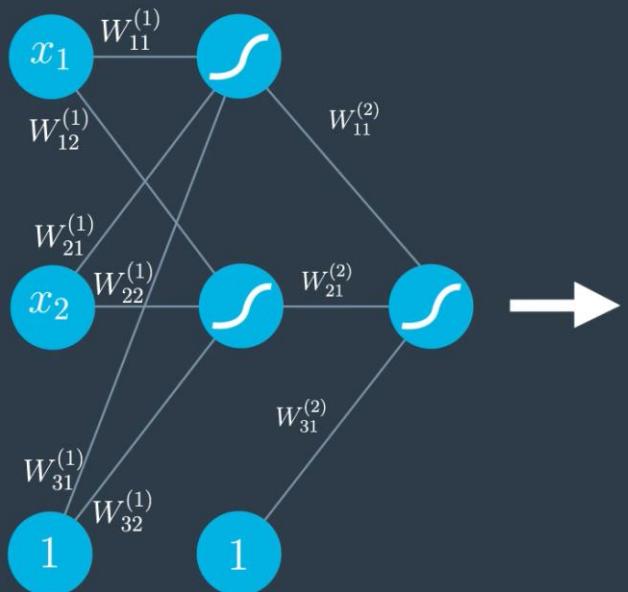
# Redes Neurais – Feedforward

Neural Network



# Redes Neurais – Feedforward

Feedforward



$$\hat{y} = \sigma \begin{pmatrix} W_{11}^{(2)} \\ W_{21}^{(2)} \\ W_{31}^{(2)} \end{pmatrix} \sigma \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

$$\hat{y} = \sigma \circ W^{(2)} \circ \sigma \circ W^{(1)}(x)$$



# Thanks !



Vinicius Fernandes Caridá

[vfcarida@gmail.com](mailto:vfcarida@gmail.com)



@vinicius caridá



@vfcarida



@vinicius caridá



@vfcarida



@vinicius caridá



@vfcarida