VINI CARIDÁ

Human-Centered Data & AI

Vinicius Caridá, Ph.D.

Head of Digital Customer Service Platforms, PCP, WFM, Data and AI - Itaú Unibanco

MBA Professor - FIAP

Google Developer Expert – Machine Learning

Co-organizer TFUGSP and AWSUGSP

@vinicius caridá @vfcarida @vinicius caridá @vfcarida @vinicius caridá @vfcarida

VINI CARIDÁ
Human-Centered Data & AI

Zero to Hero
Machine Learning na AWS

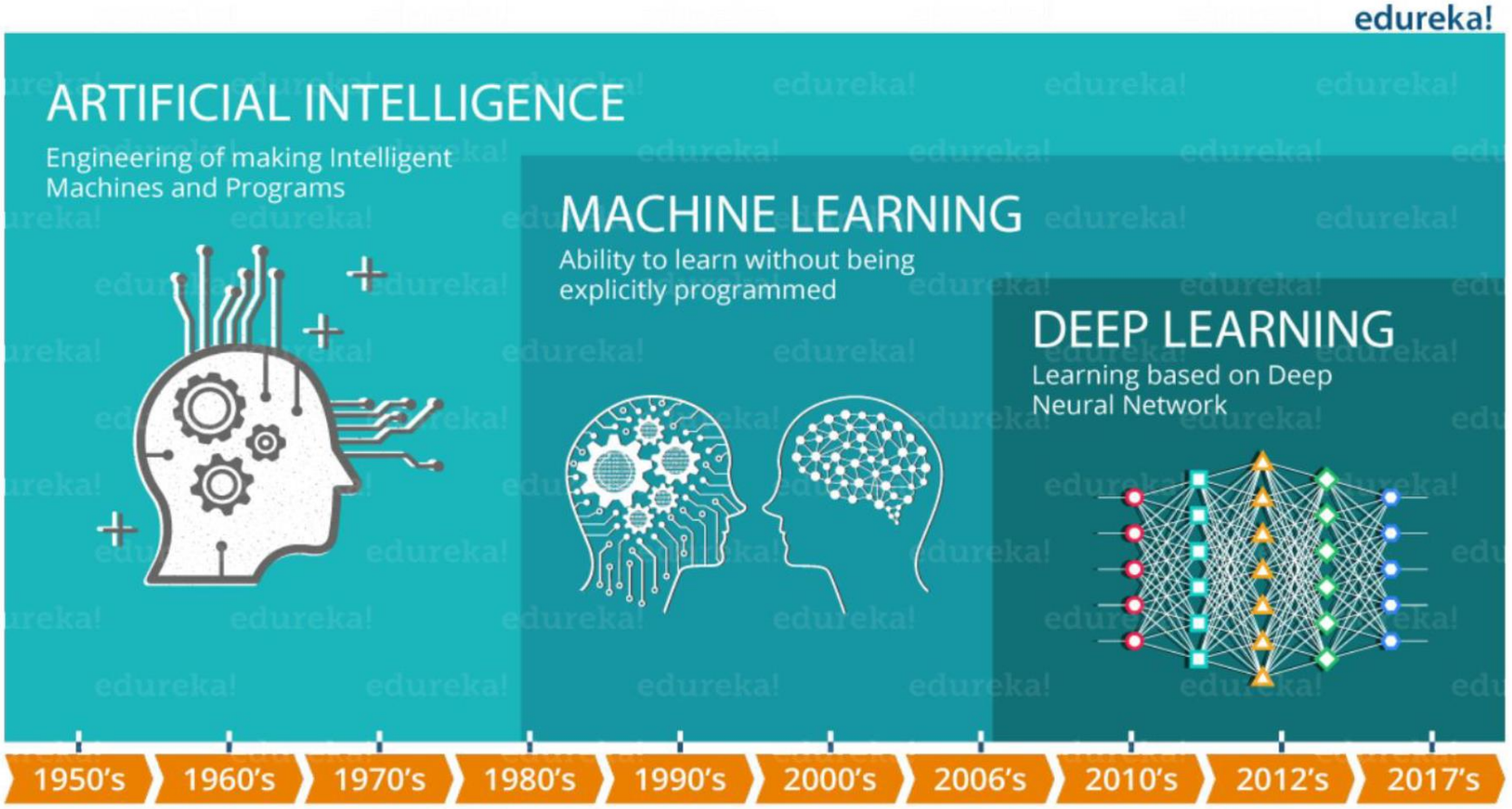
Parte 4/5

Vinicius F. Caridá

VINI CARIDÁ
Human-Centered Data & AI

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

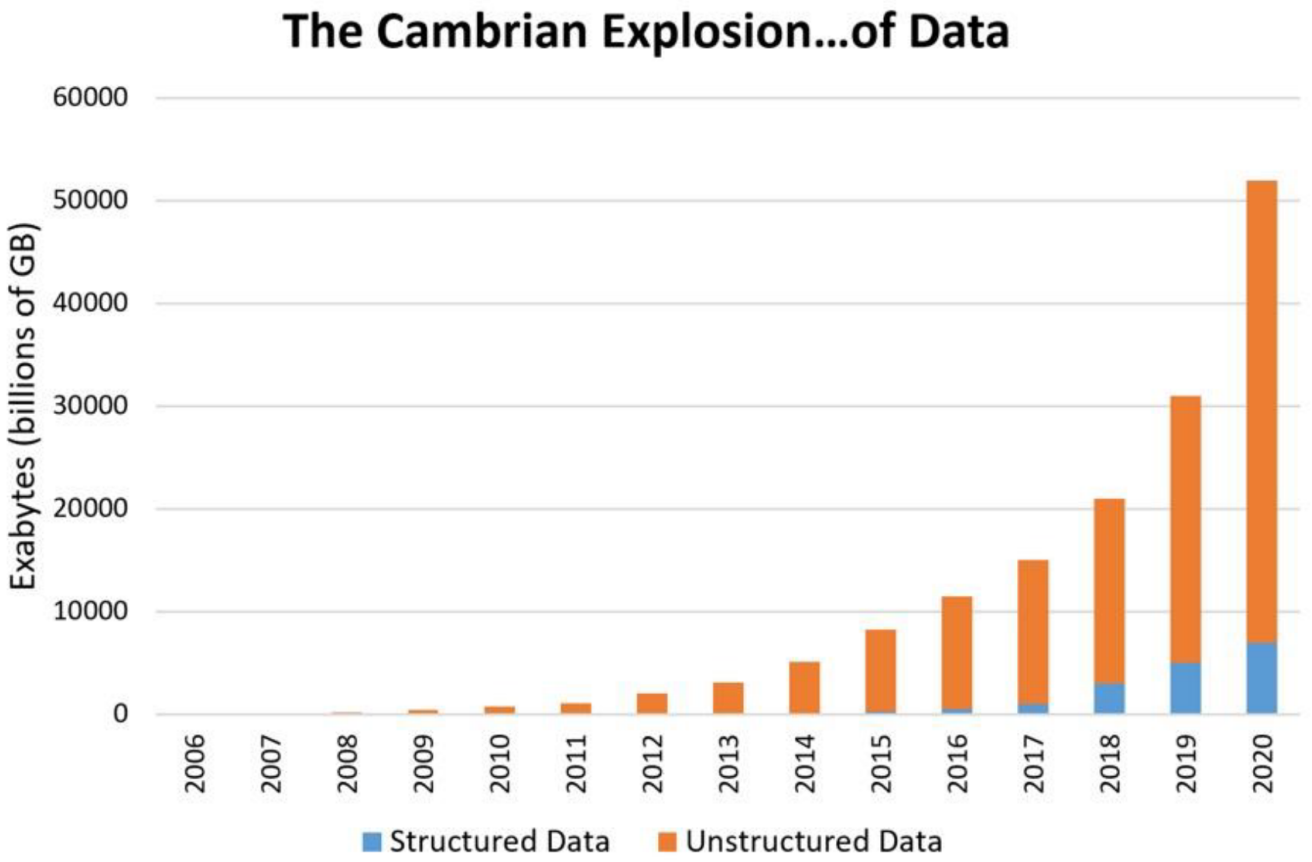


Figura 16 – Taxa de expansão na produção de dados estruturados e não-estruturados

Fonte: https://www.eetimes.com/author.asp?section_id=36&doc_id=1330462

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.

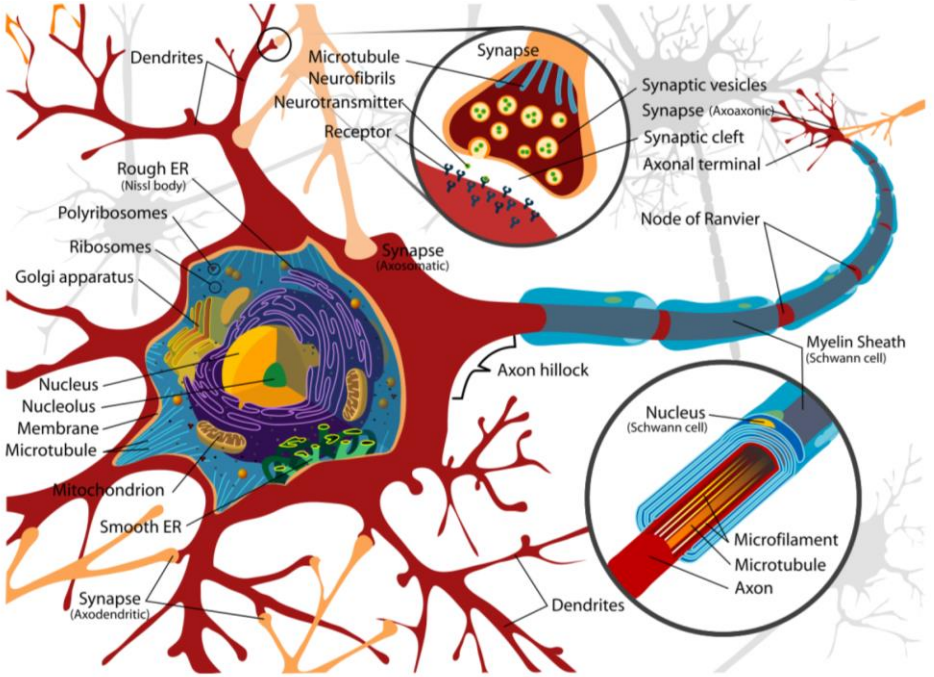


Figura 2 – Elementos constituintes de um neurônio

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]

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.

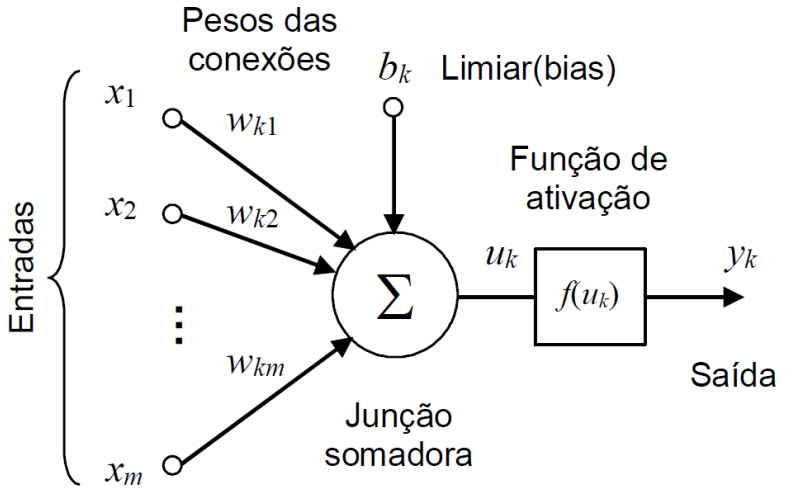


Figura 7 – Modelo matemático de um neurônio artificial

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

O problema do OU-exclusivo em MLP

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

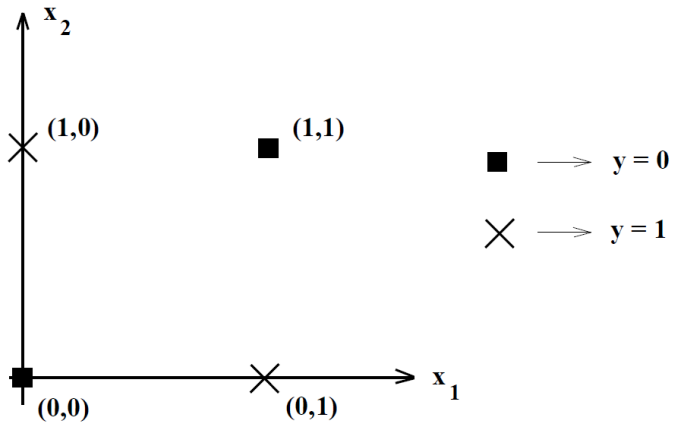


Figura 26 – O problema do OU-exclusivo

		AND	OR	XOR
0	0			
0	1			
1	0			
1	1			

O problema do OU-exclusivo em MLP

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

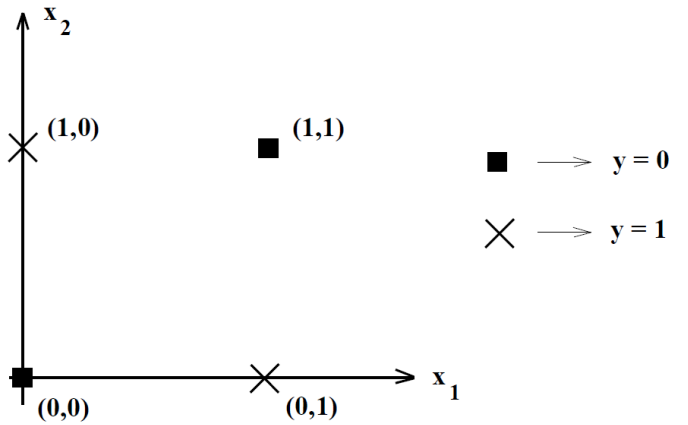


Figura 26 – O problema do OU-exclusivo

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

O problema do OU-exclusivo em MLP

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

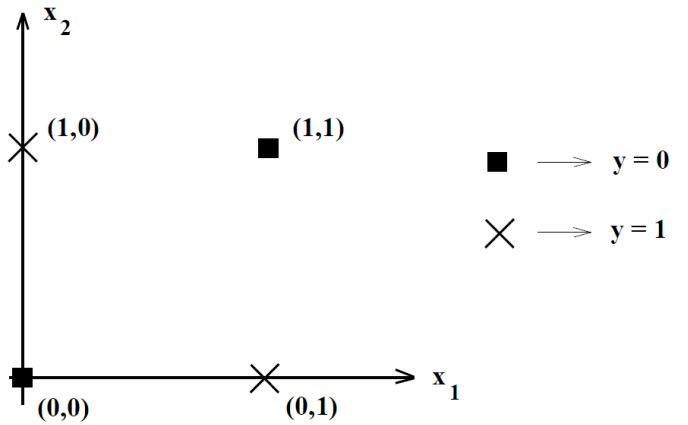


Figura 26 – O problema do OU-exclusivo

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

O problema do OU-exclusivo em MLP

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

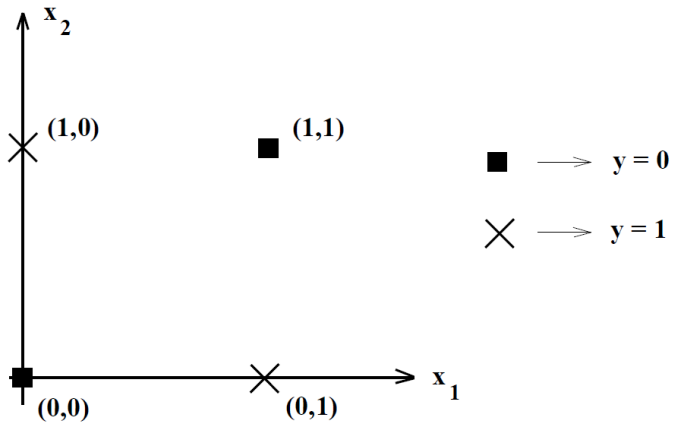
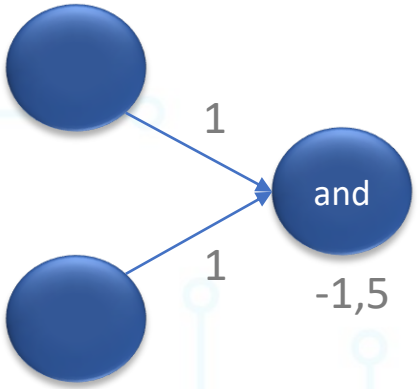


Figura 26 – O problema do OU-exclusivo

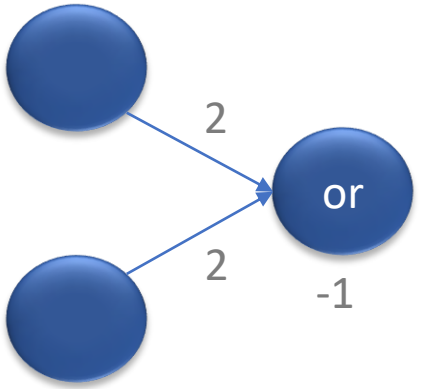
		AND	OR	XOR
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

Redes Neurais

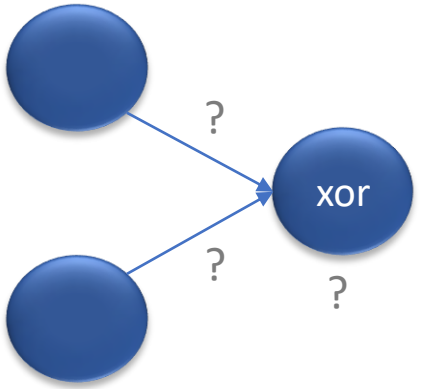
0	0	0
0	1	0
1	0	0
1	1	1



0	0	0
0	1	1
1	0	1
1	1	1



0	0	0
0	1	1
1	0	1
1	1	0



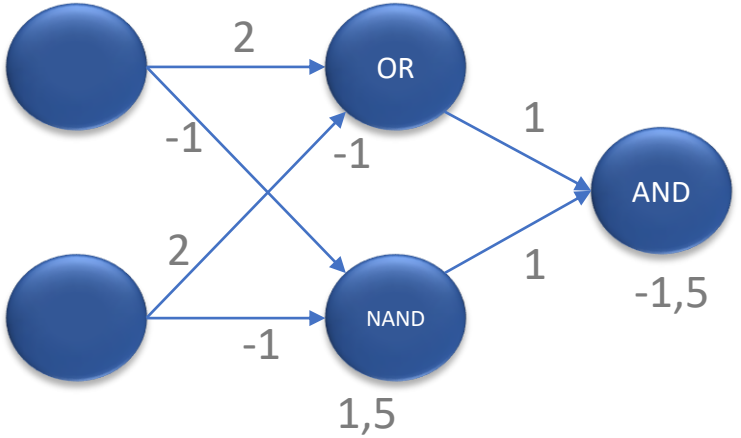
Função Degrau Padrão

Se >= 0 então 1

Se < 0 então 0

Redes Neurais

0	0	0
0	1	1
1	0	1
1	1	0

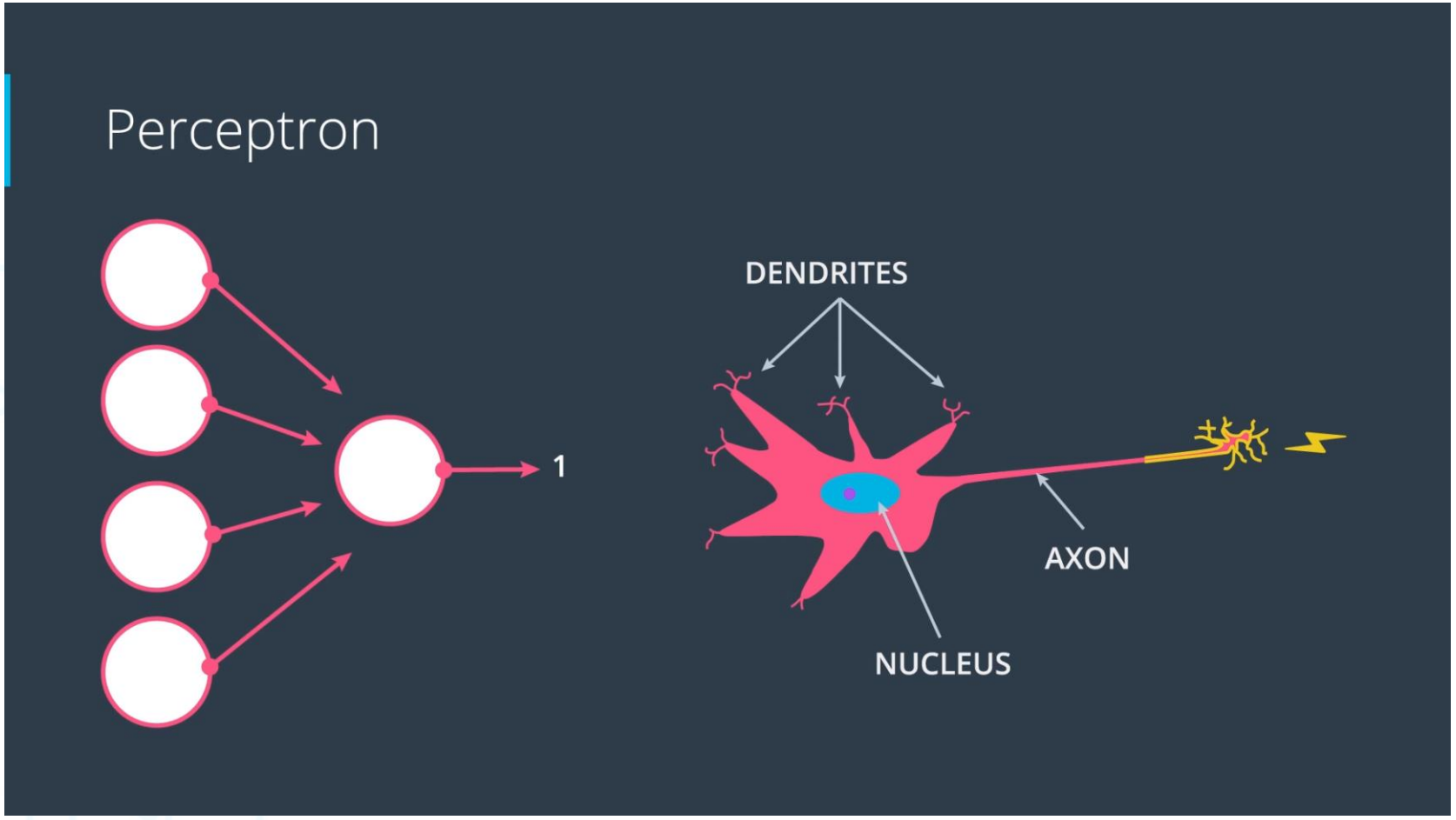


Função Degrau Padrão

Se >= 0 então 1

Se < 0 então 0

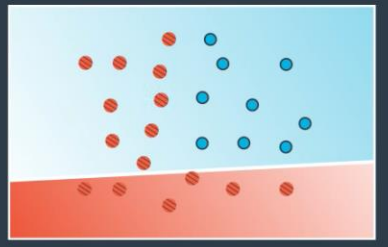
Redes Neurais

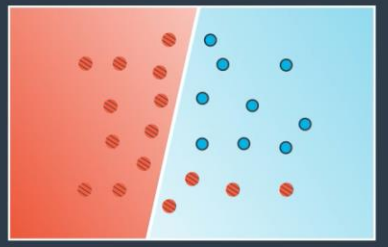


Redes Neurais - Arquitetura

Redes Neurais - Arquitetura

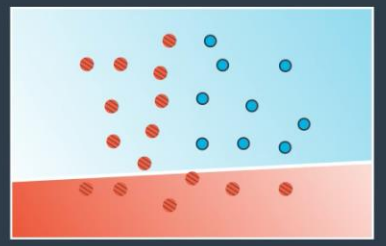
Combining Regions

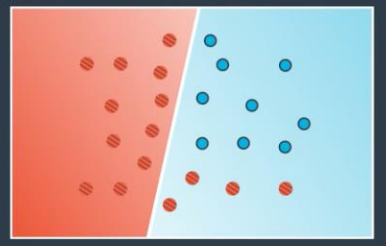


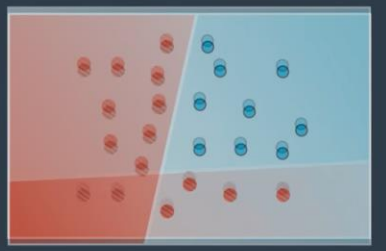


Redes Neurais - Arquitetura

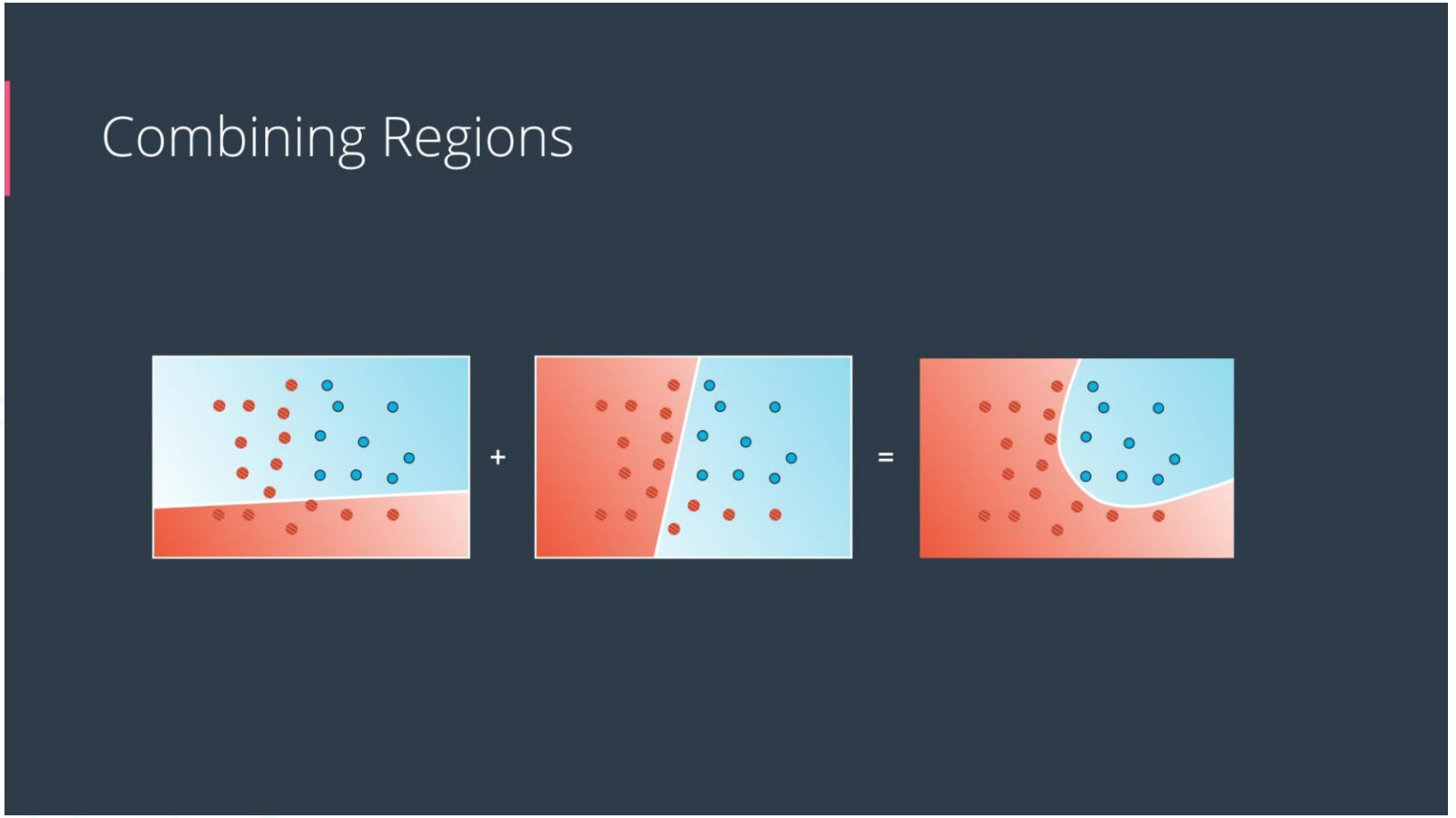
Combining Regions





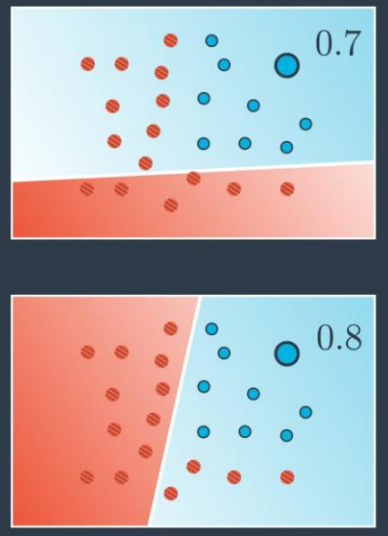


Redes Neurais - Arquitetura

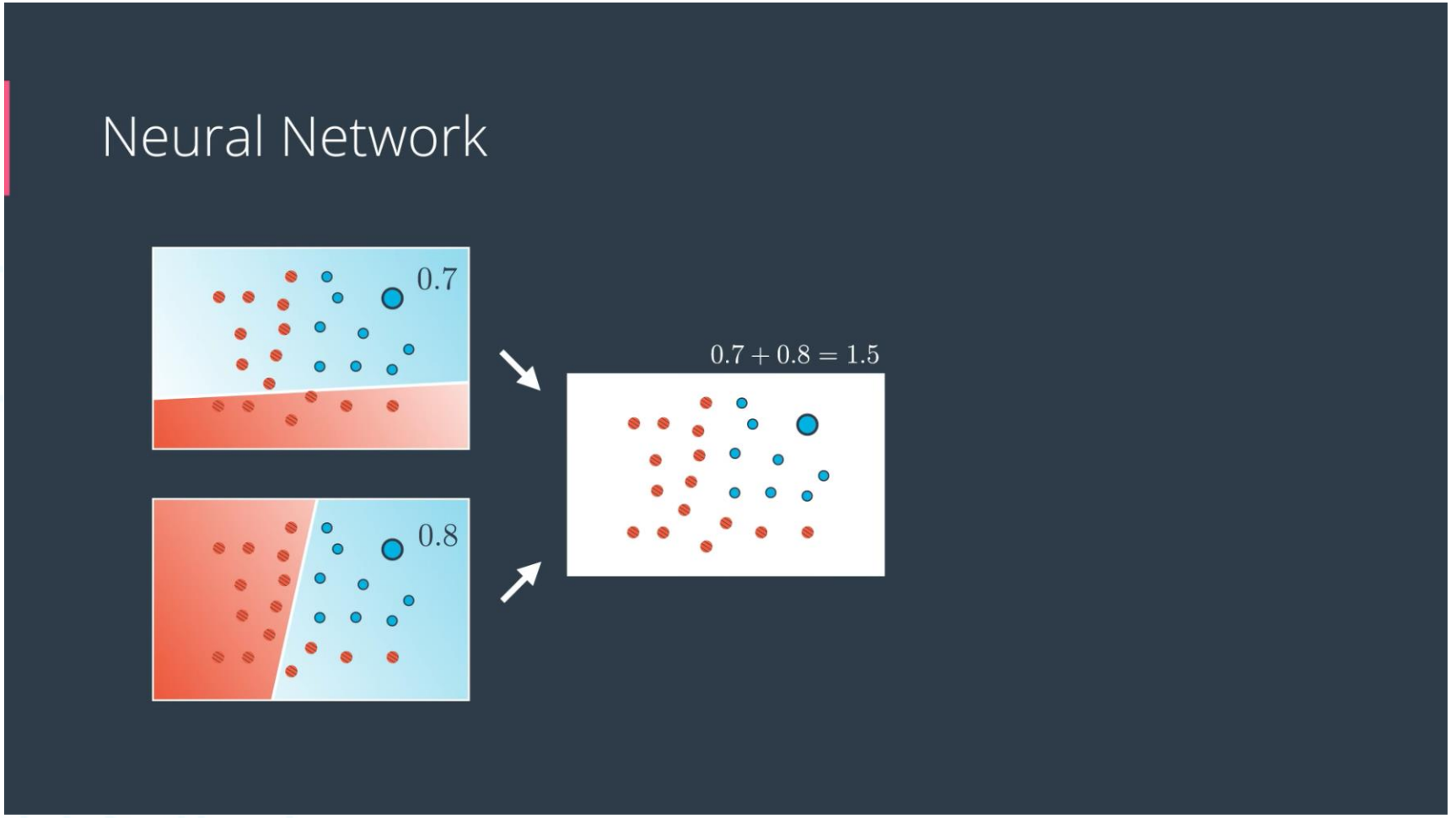


Redes Neurais - Arquitetura

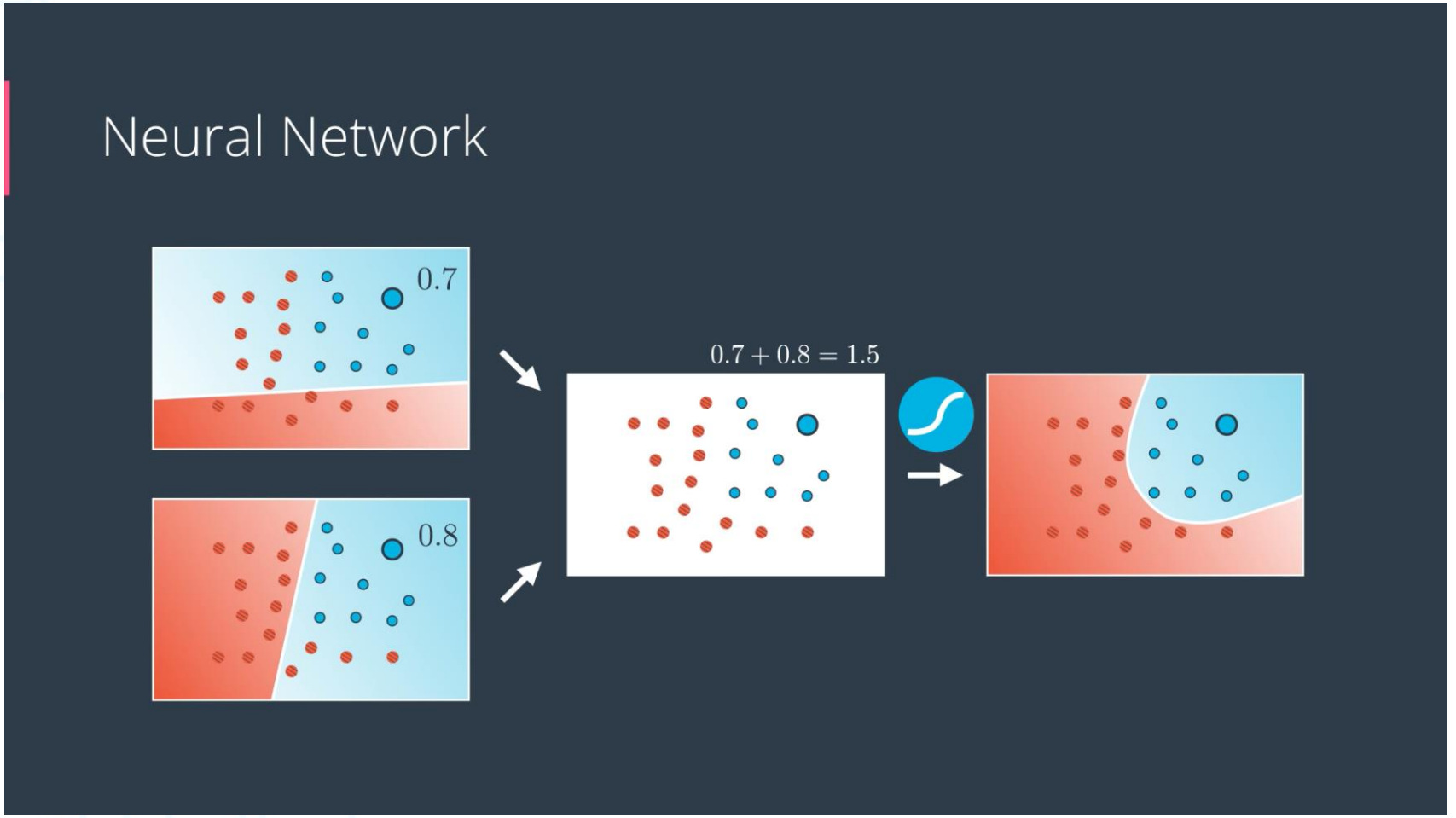
Neural Network



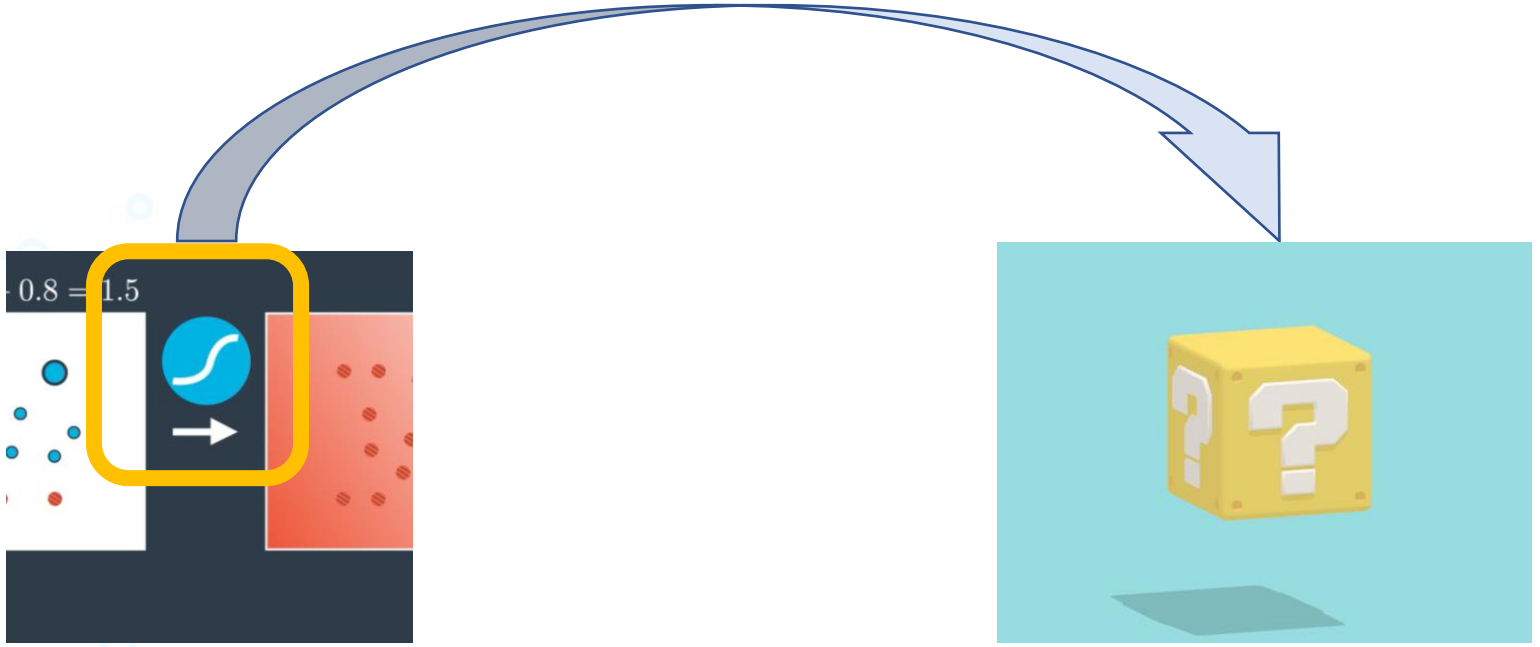
Redes Neurais - Arquitetura



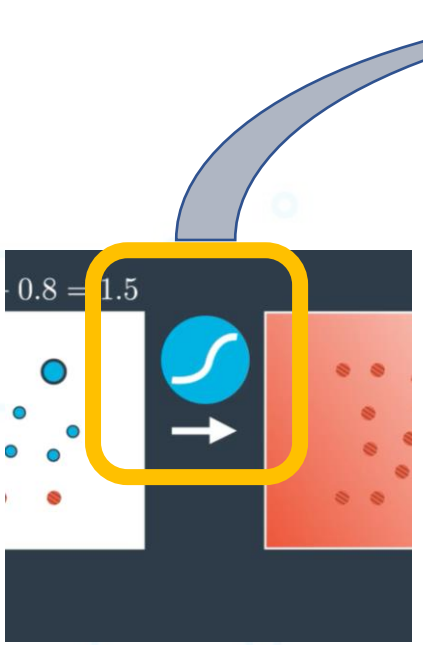
Redes Neurais - Arquitetura



Redes Neurais - Arquitetura



Redes Neurais - Arquitetura



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

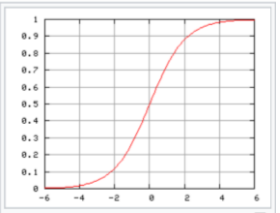
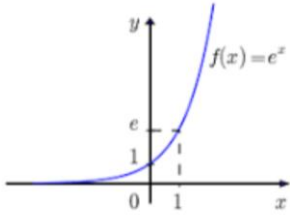
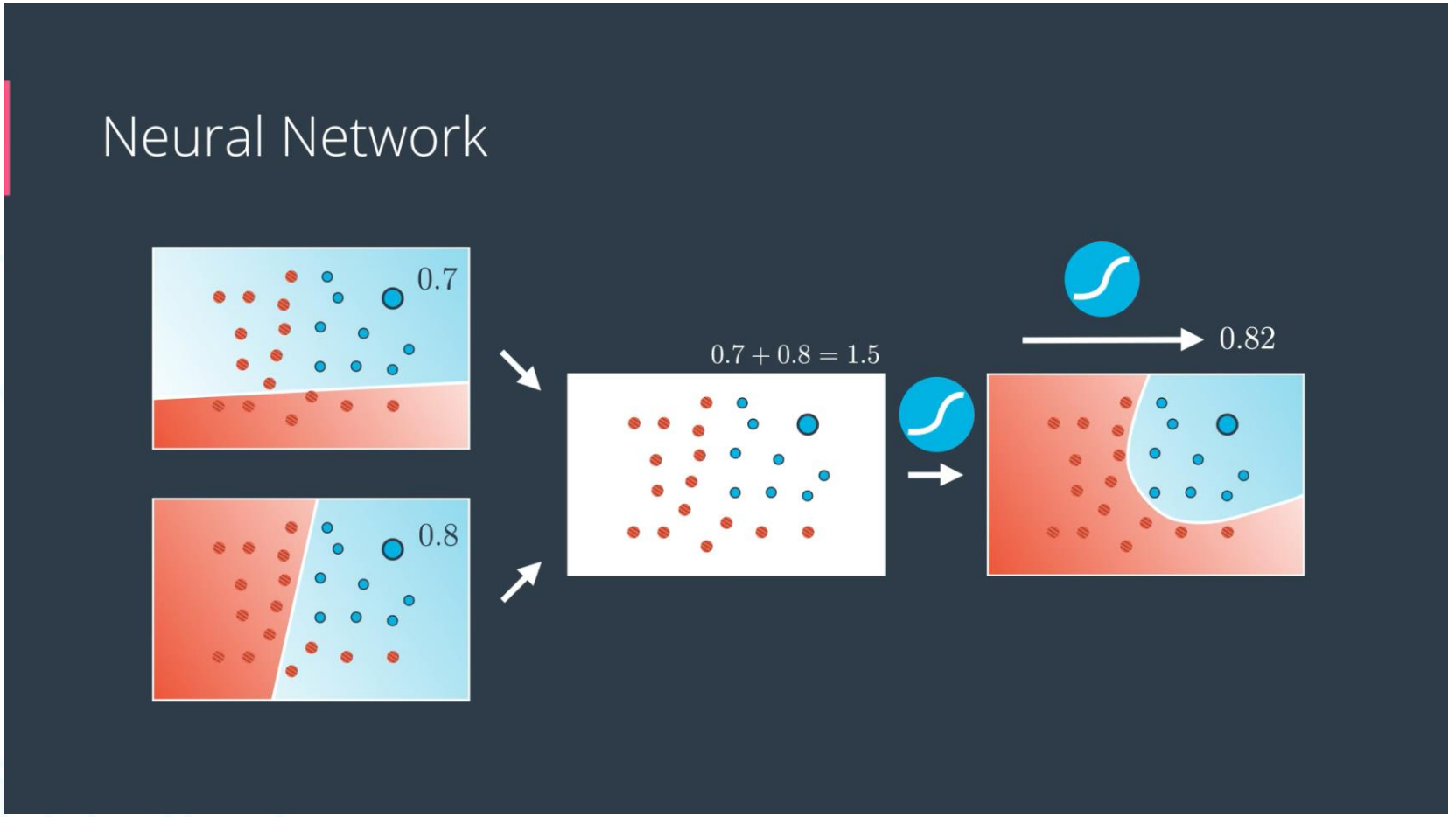


Gráfico da função sigmóide

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).

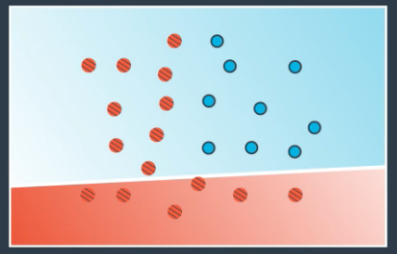


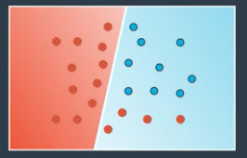
Redes Neurais - Arquitetura



Redes Neurais - Arquitetura

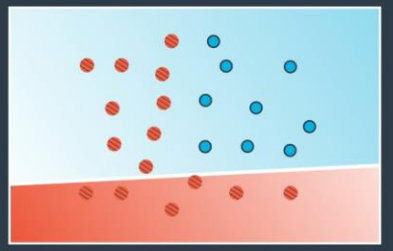
Combining Regions

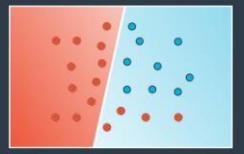


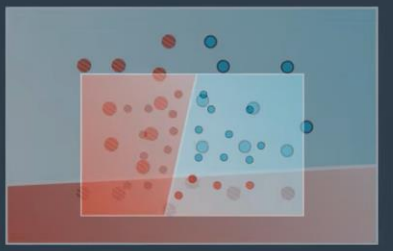


Redes Neurais - Arquitetura

Combining Regions

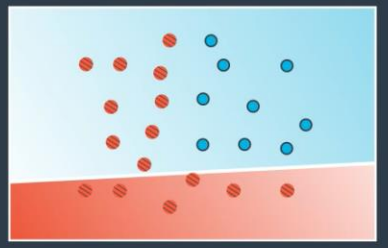


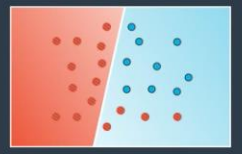


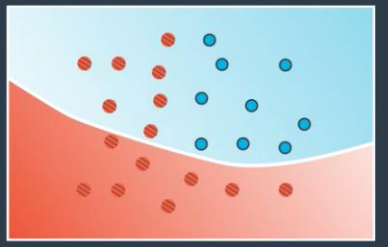


Redes Neurais - Arquitetura

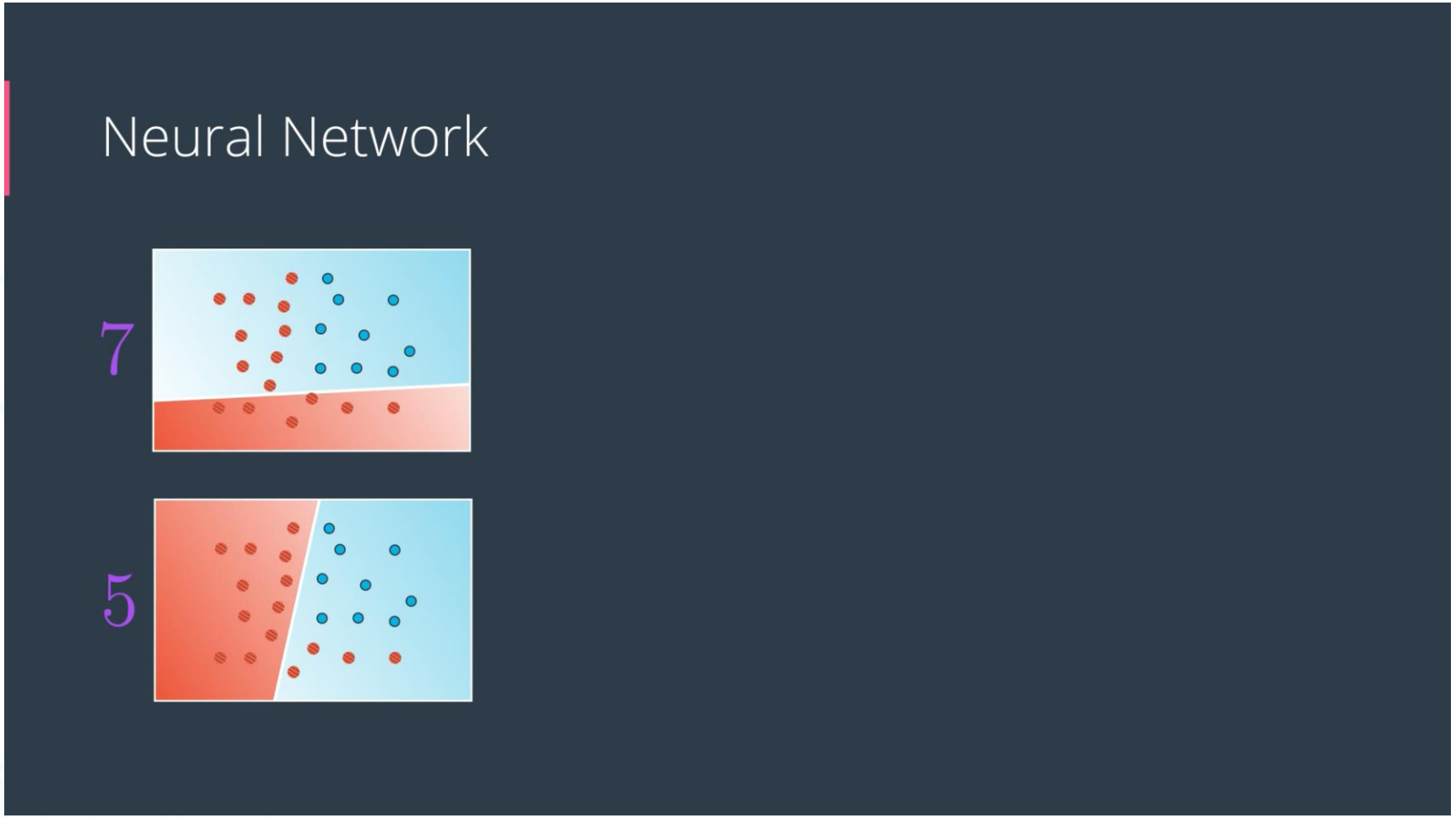
Combining Regions



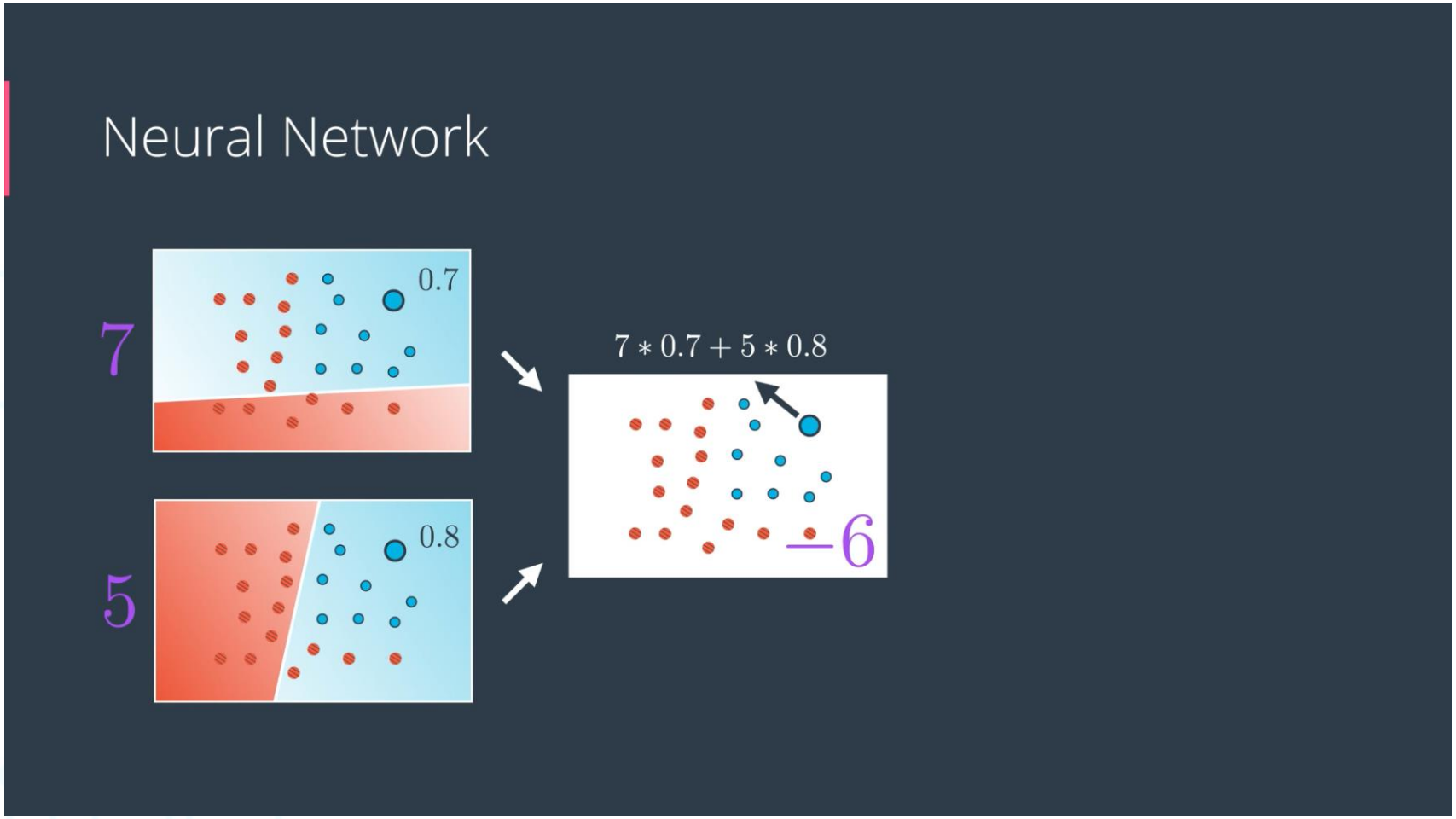




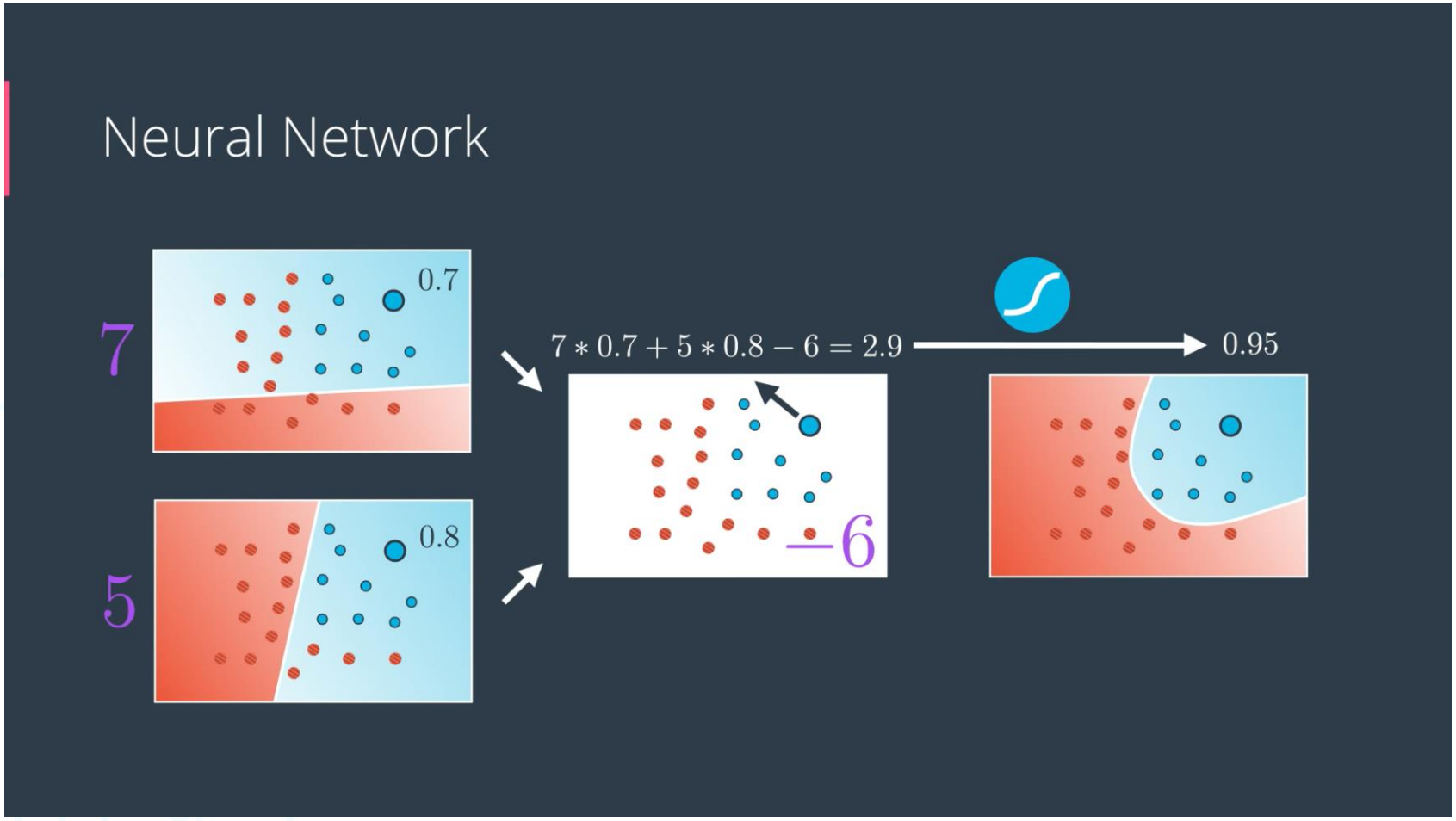
Redes Neurais - Arquitetura



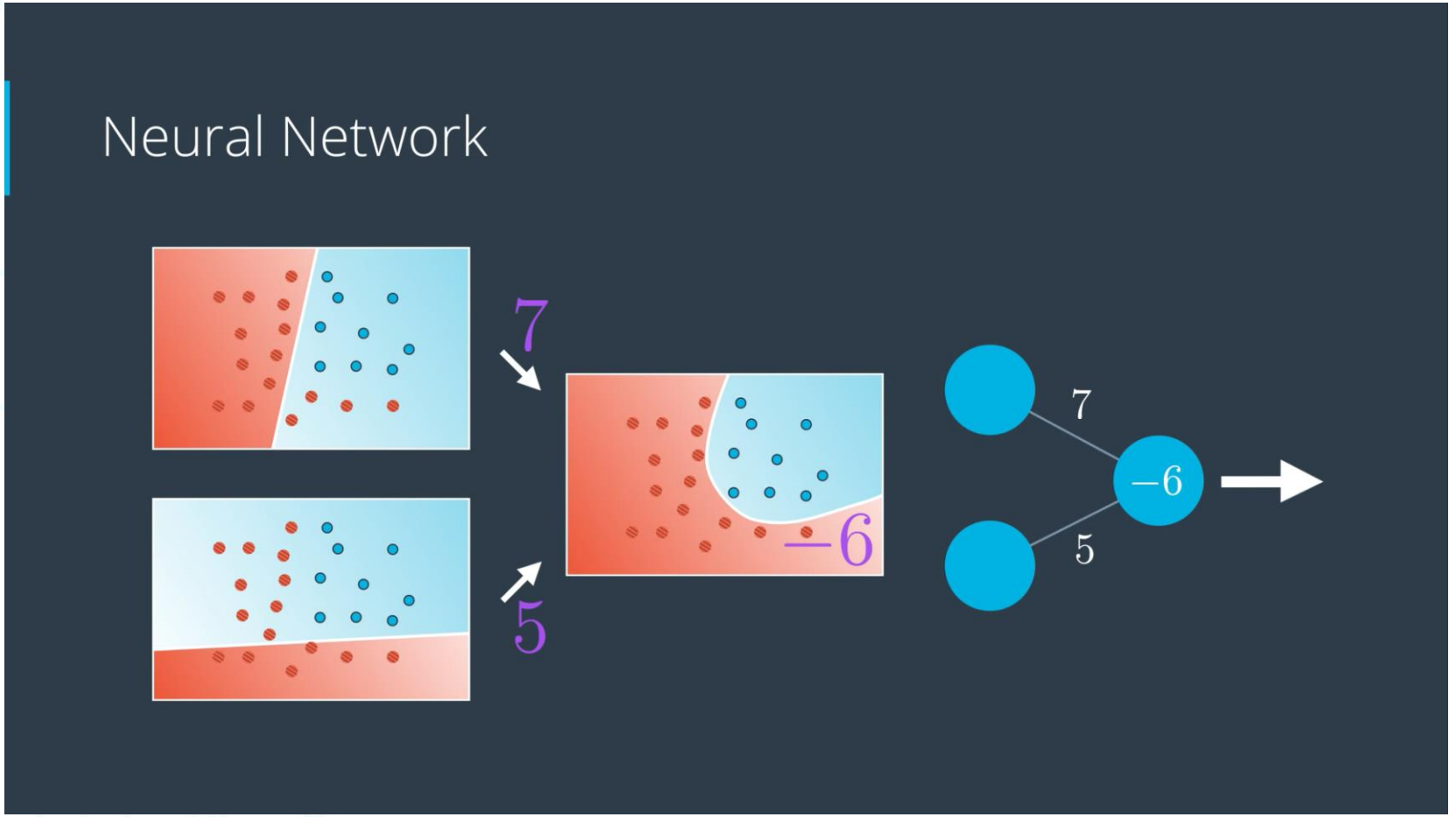
Redes Neurais - Arquitetura



Redes Neurais - Arquitetura

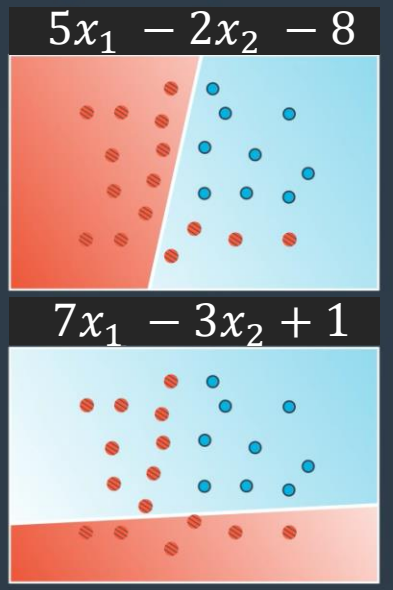


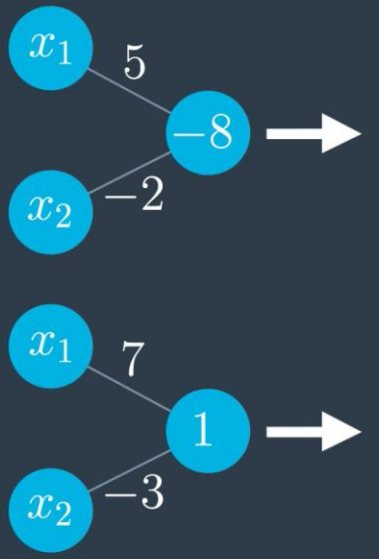
Redes Neurais - Arquitetura



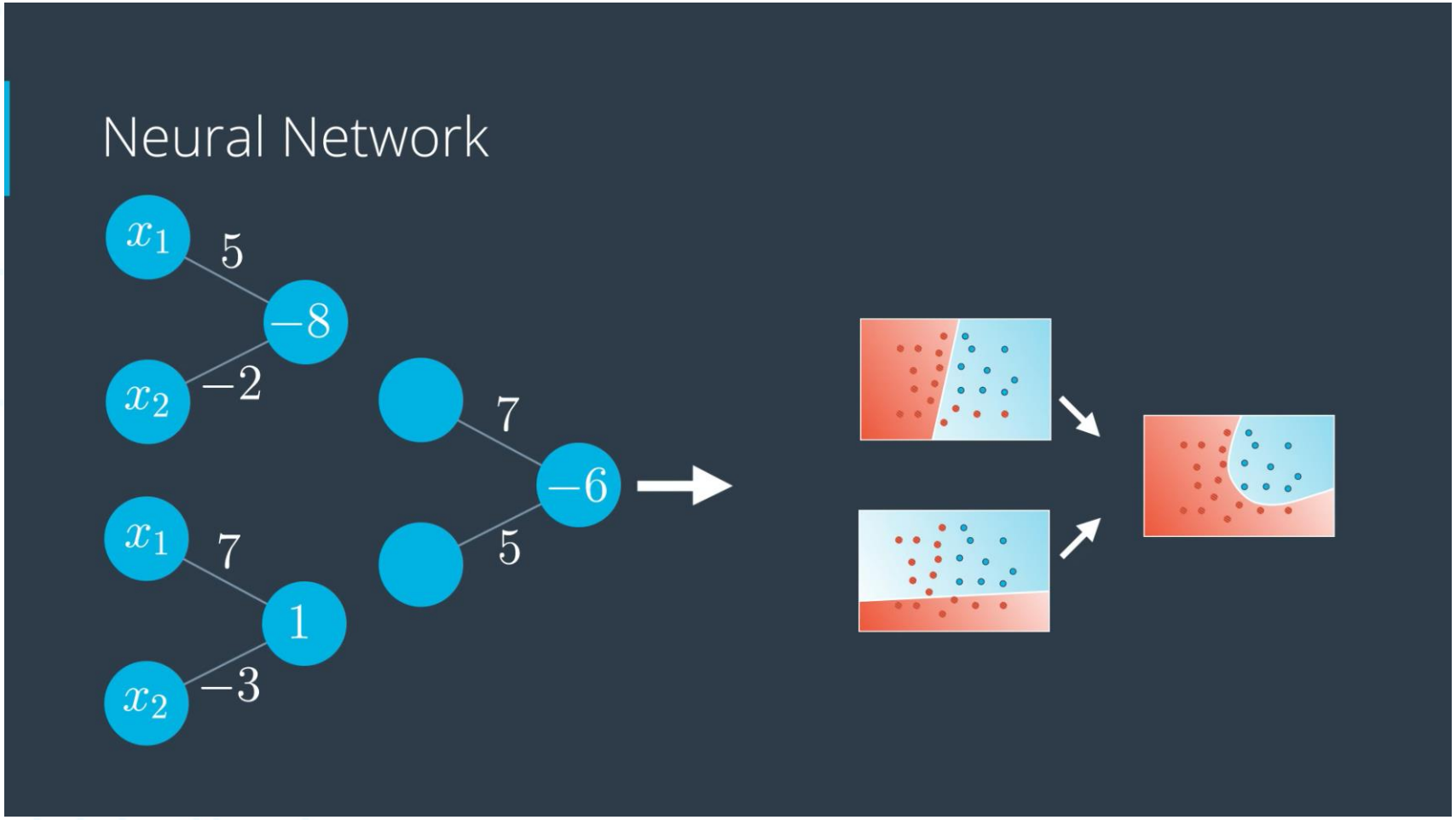
Redes Neurais - Arquitetura

Neural Network

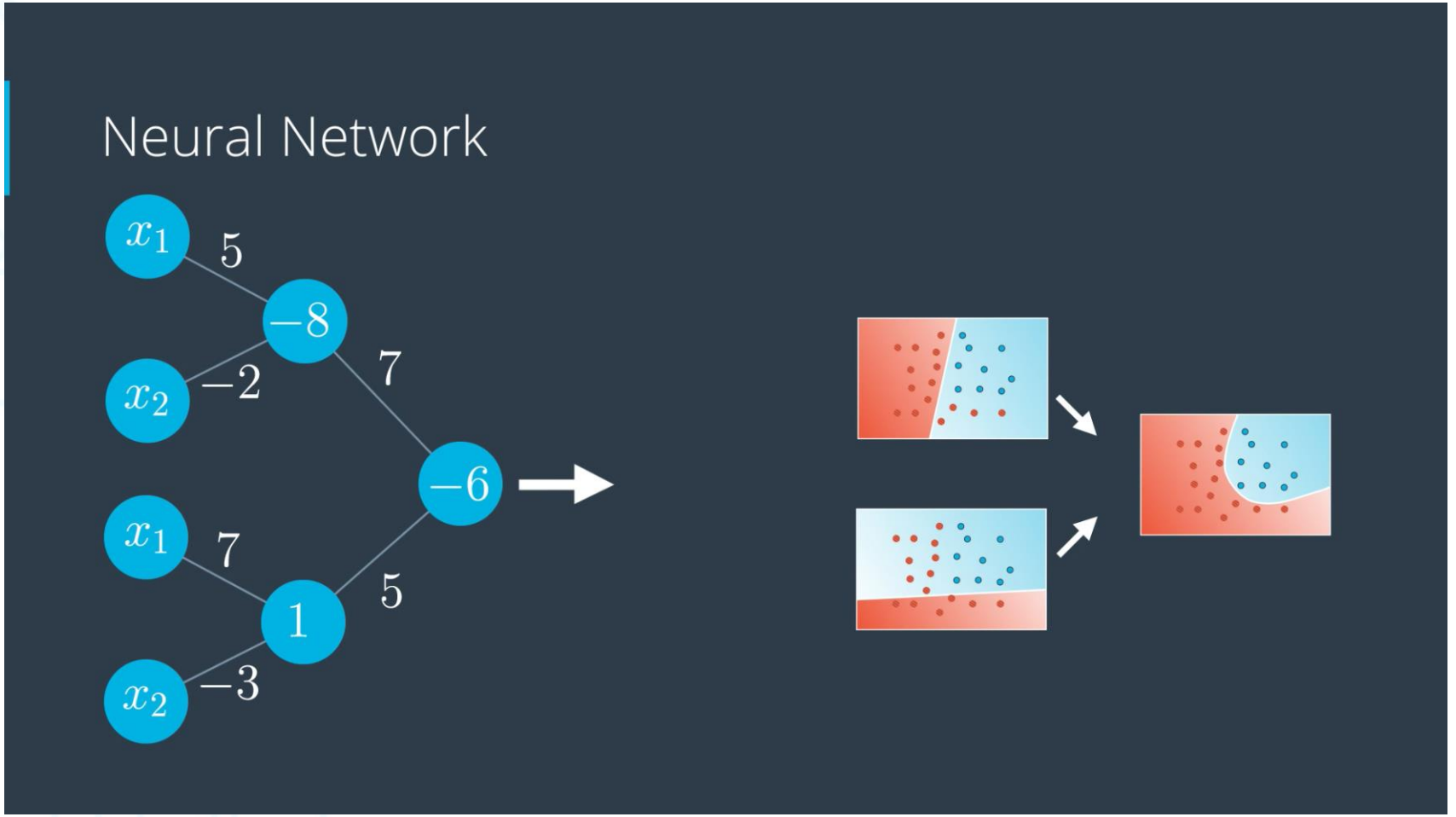




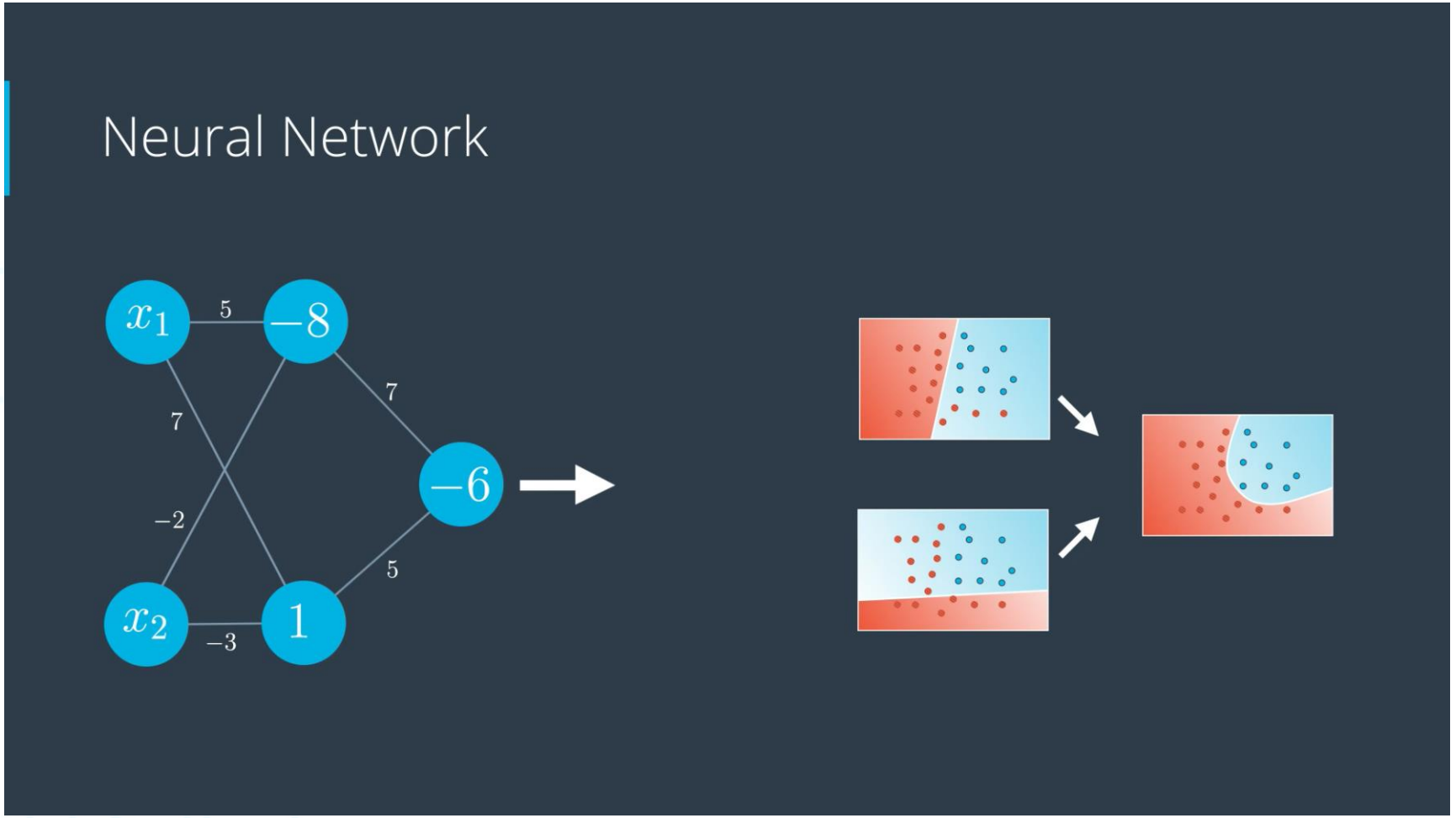
Redes Neurais - Arquitetura



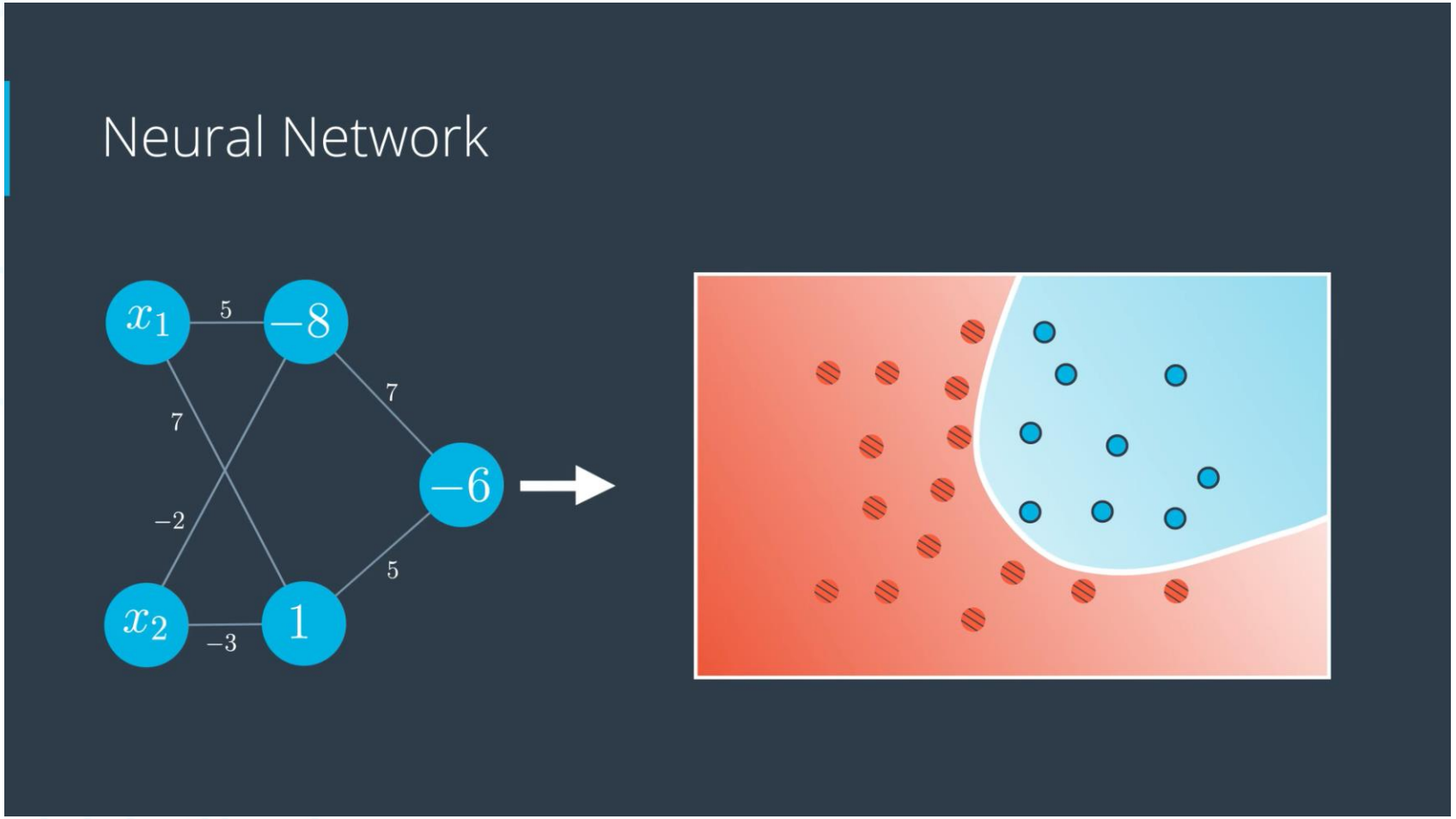
Redes Neurais - Arquitetura



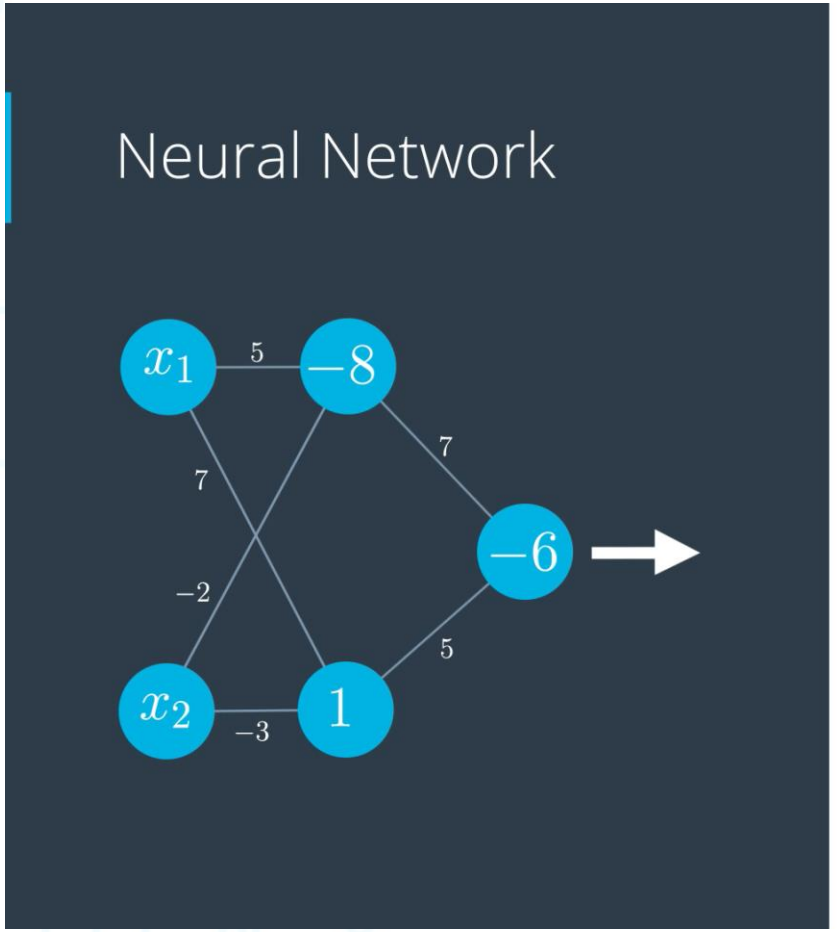
Redes Neurais - Arquitetura

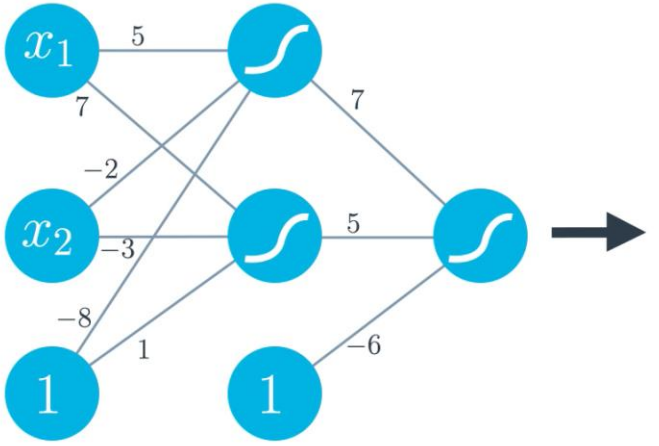


Redes Neurais - Arquitetura

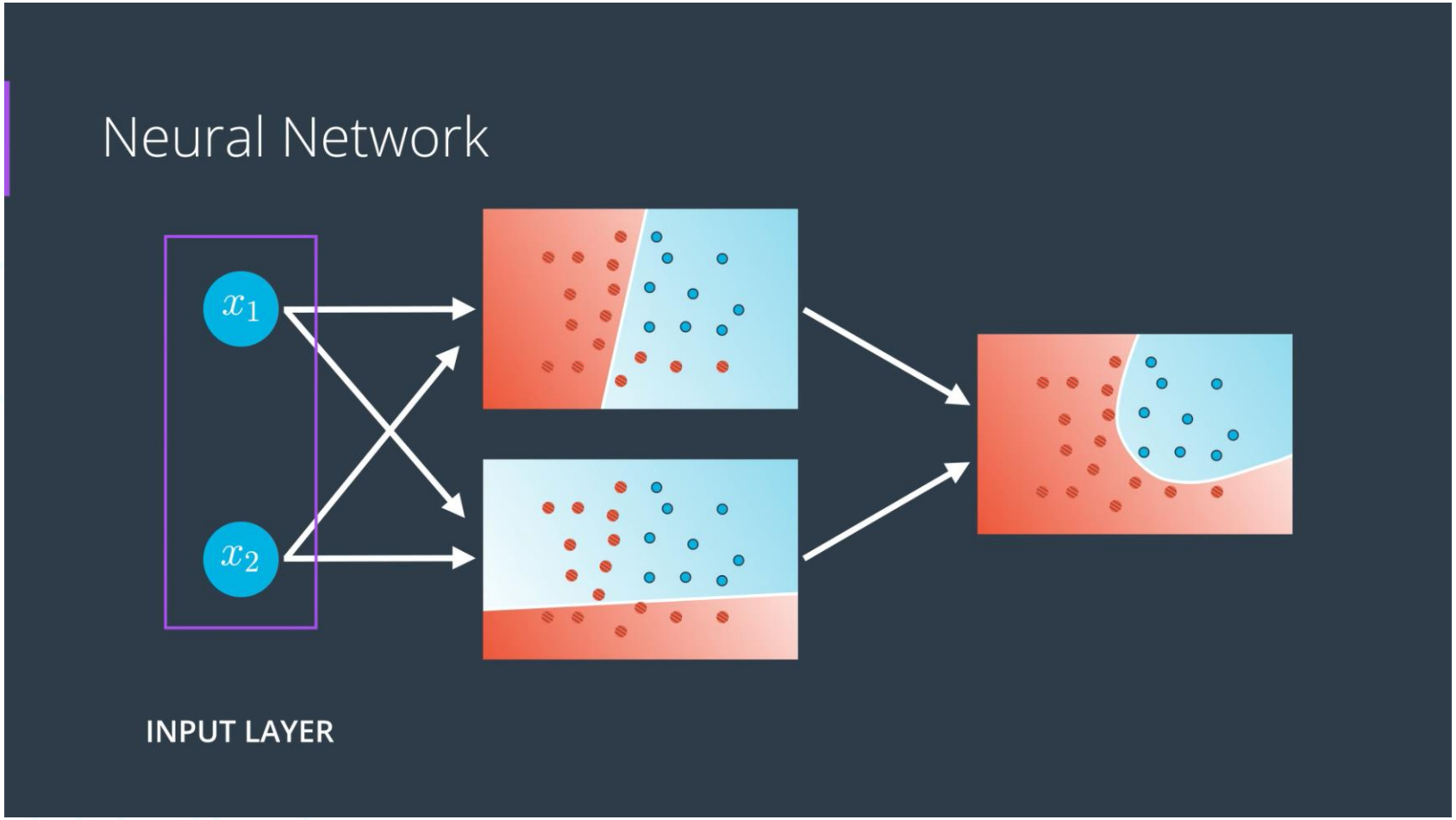


Redes Neurais - Arquitetura

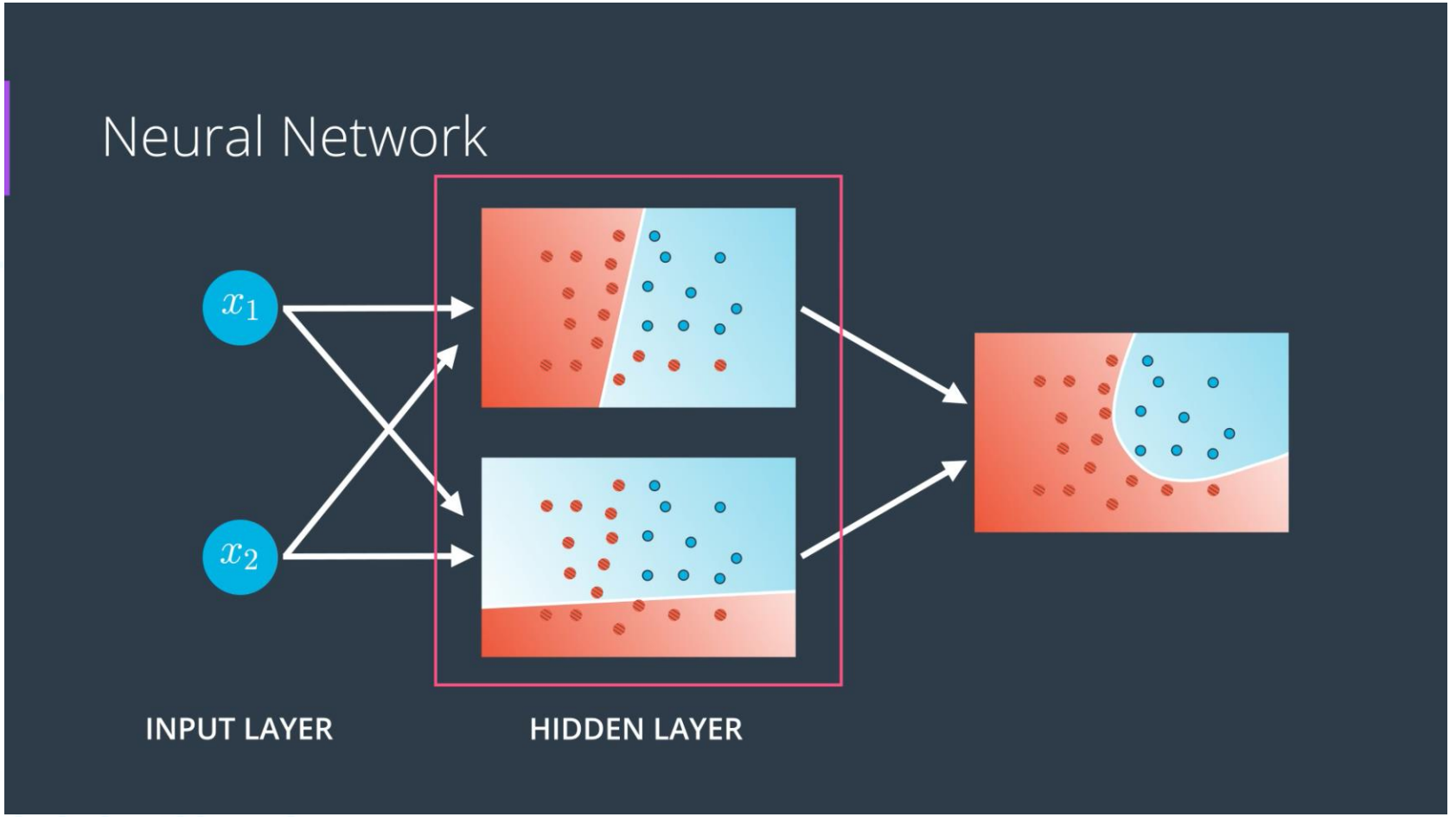




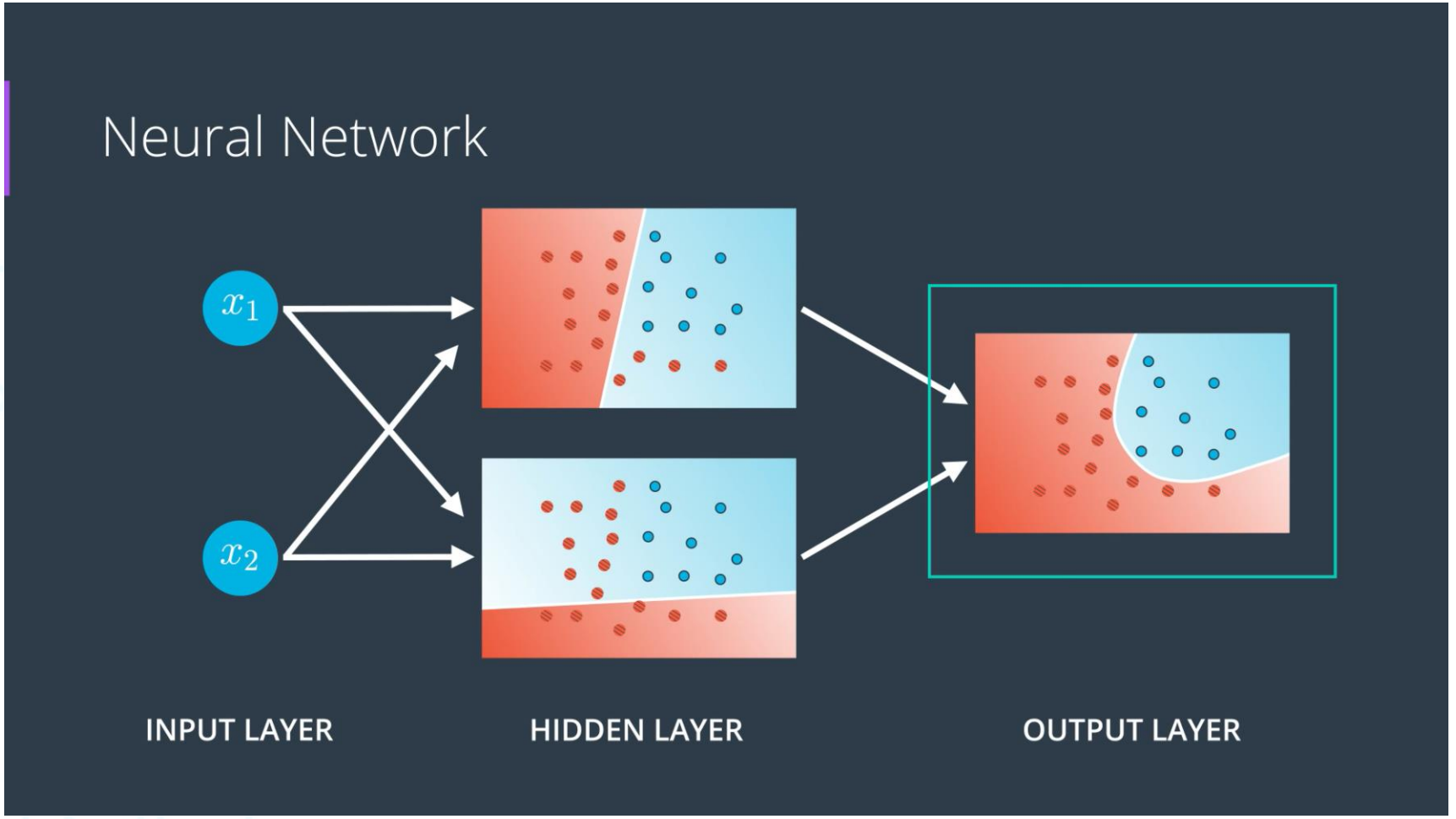
Redes Neurais – Multiple Layers



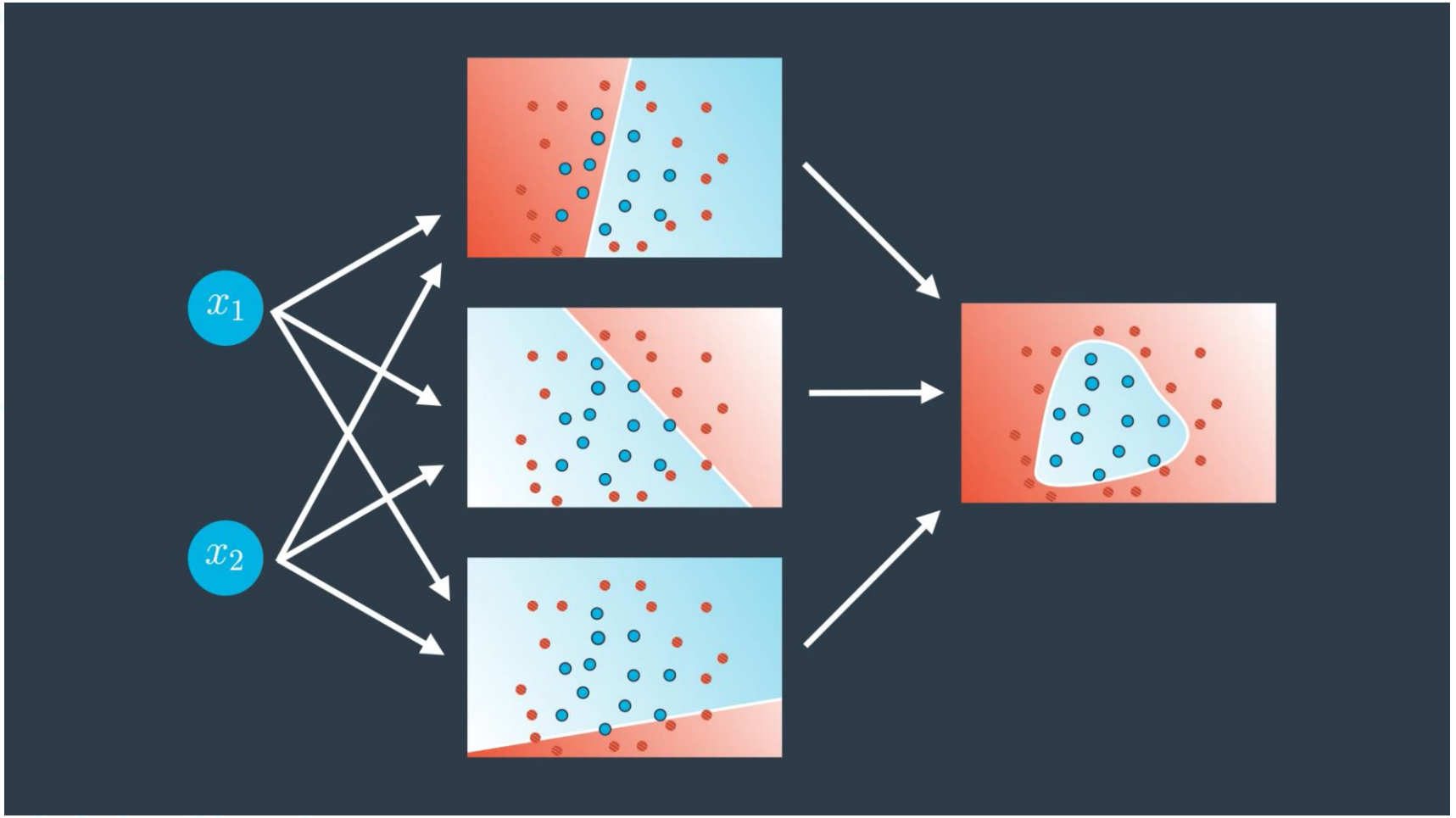
Redes Neurais – Multiple Layers



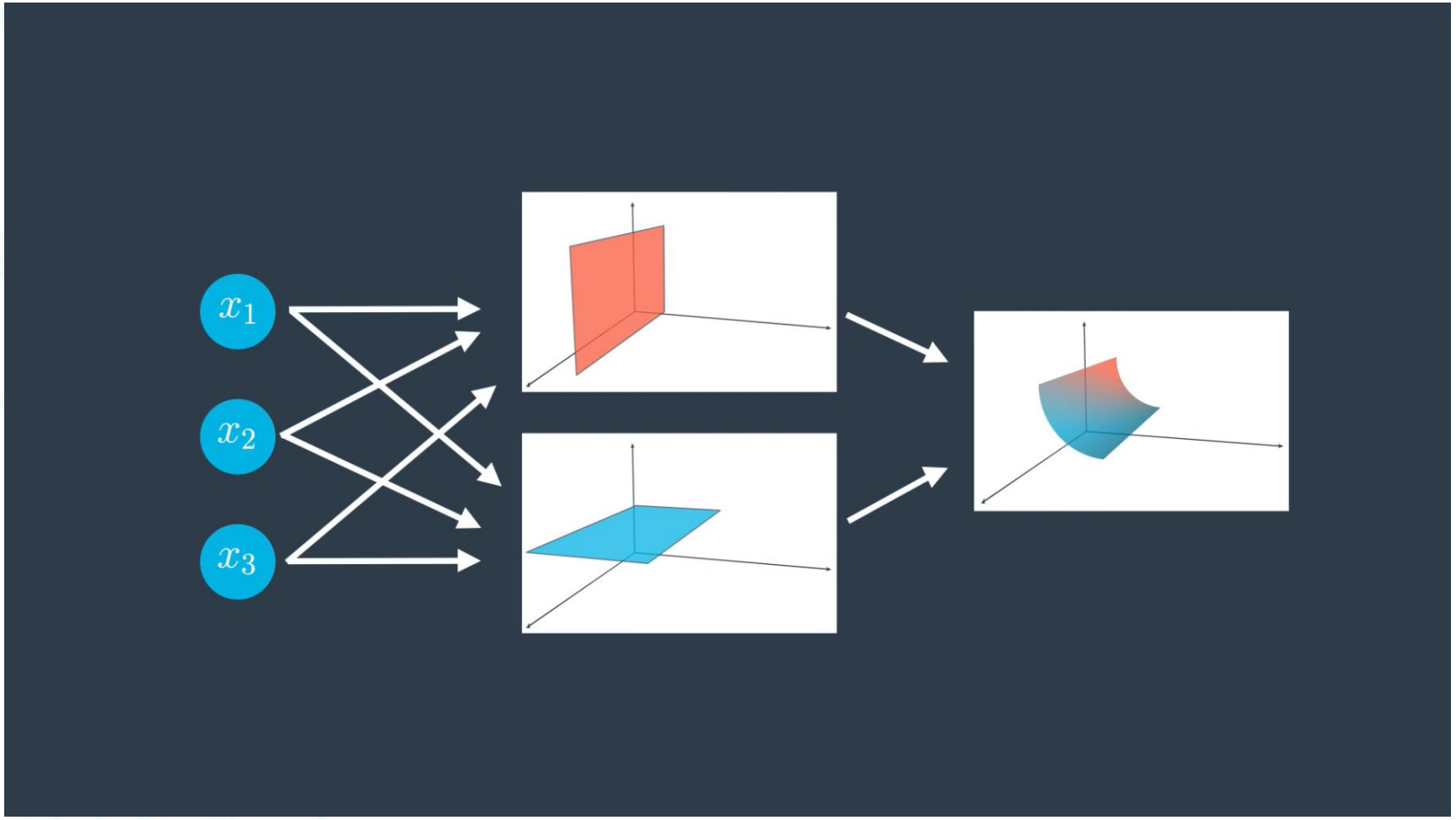
Redes Neurais – Multiple Layers



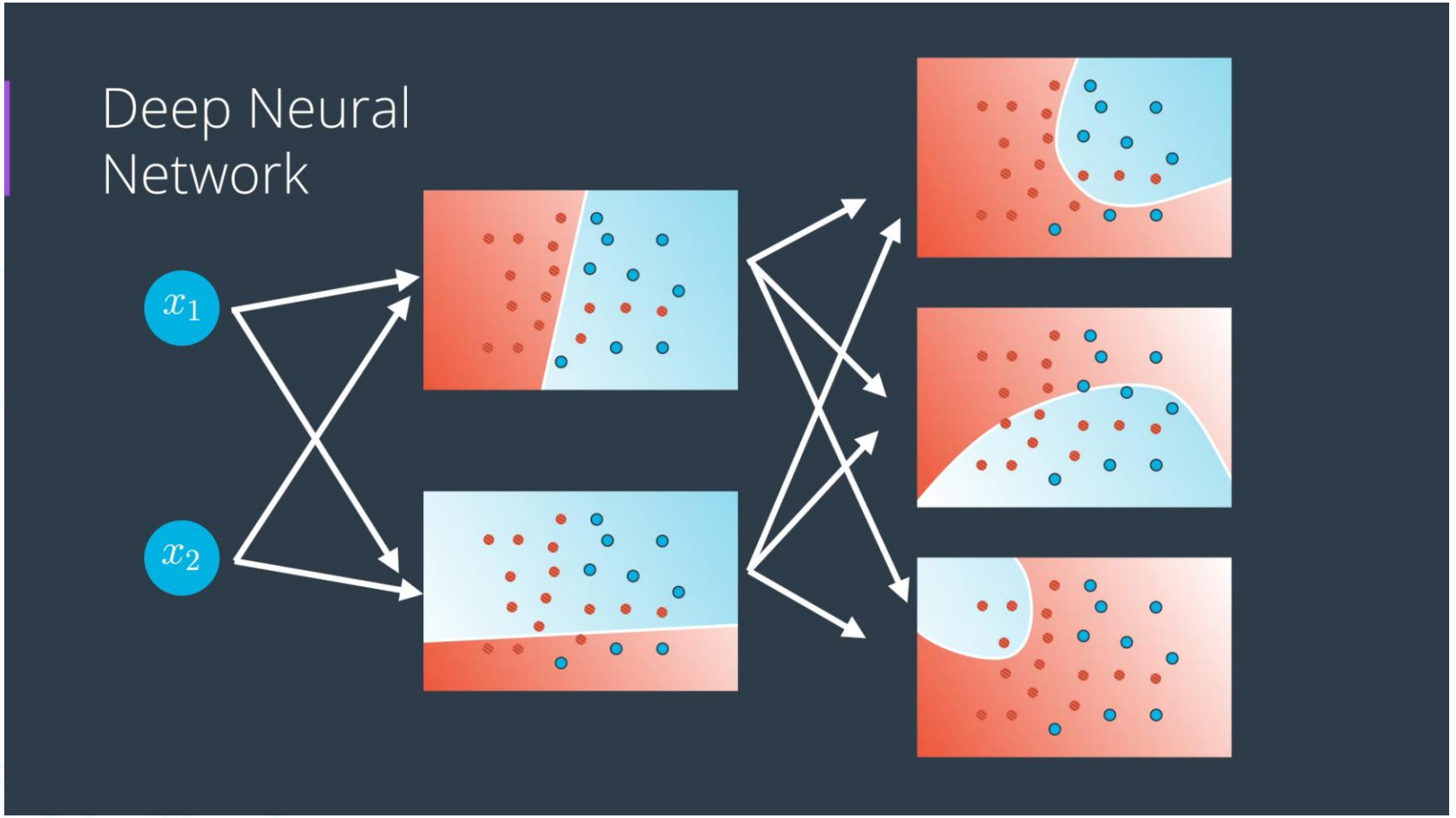
Redes Neurais – Multiple Layers



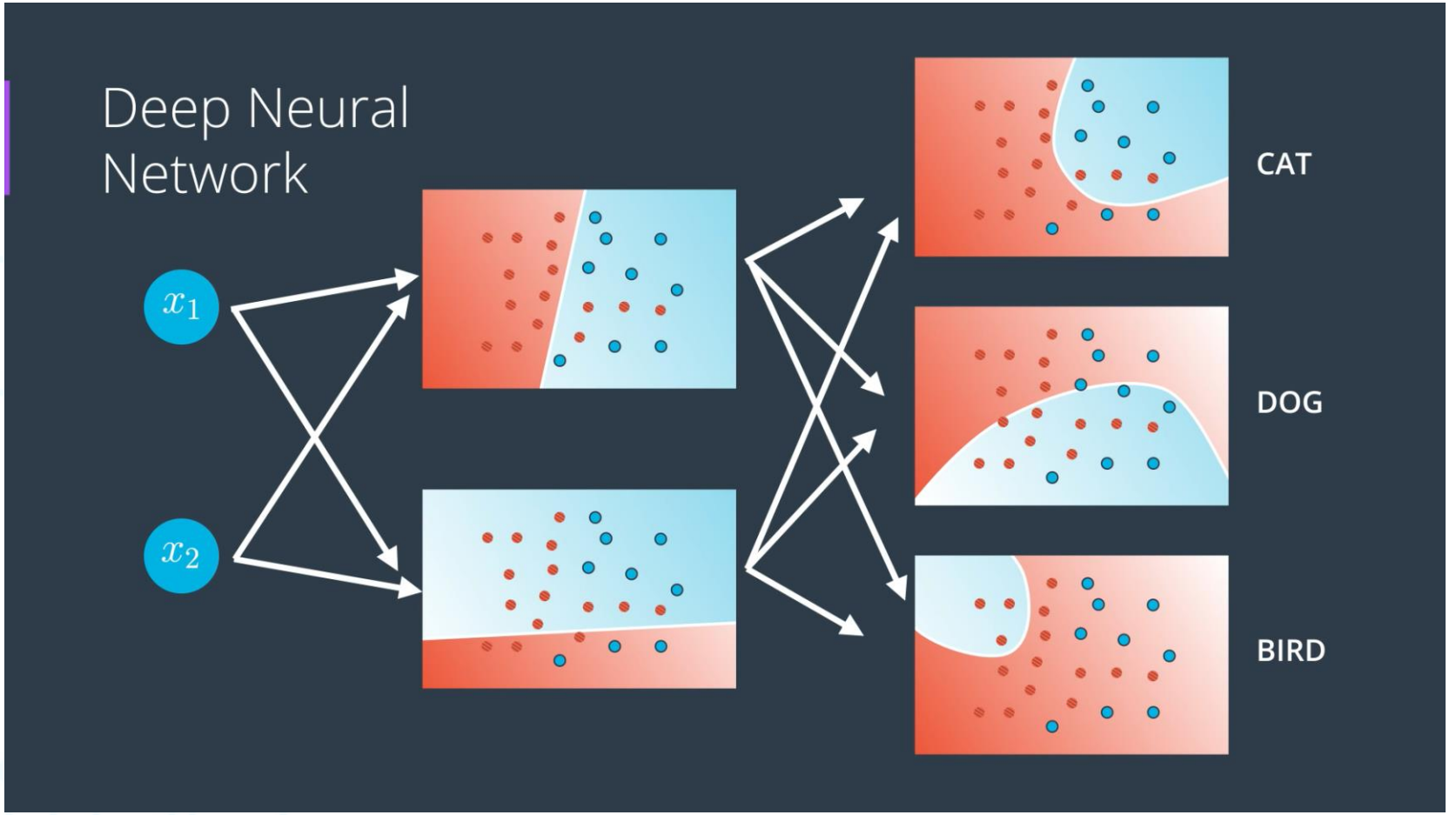
Redes Neurais – Multiple Layers



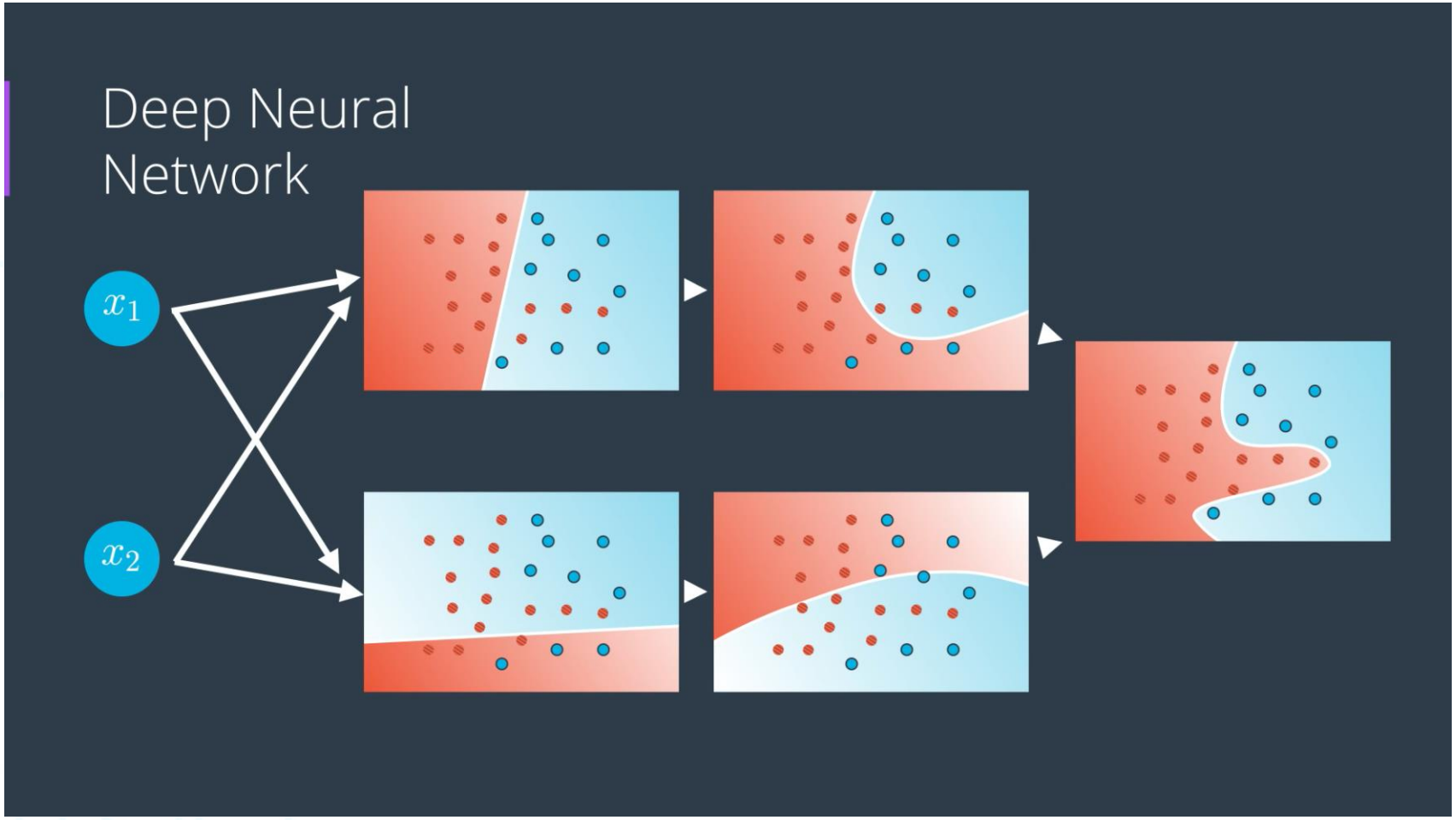
Redes Neurais – Multiple Layers



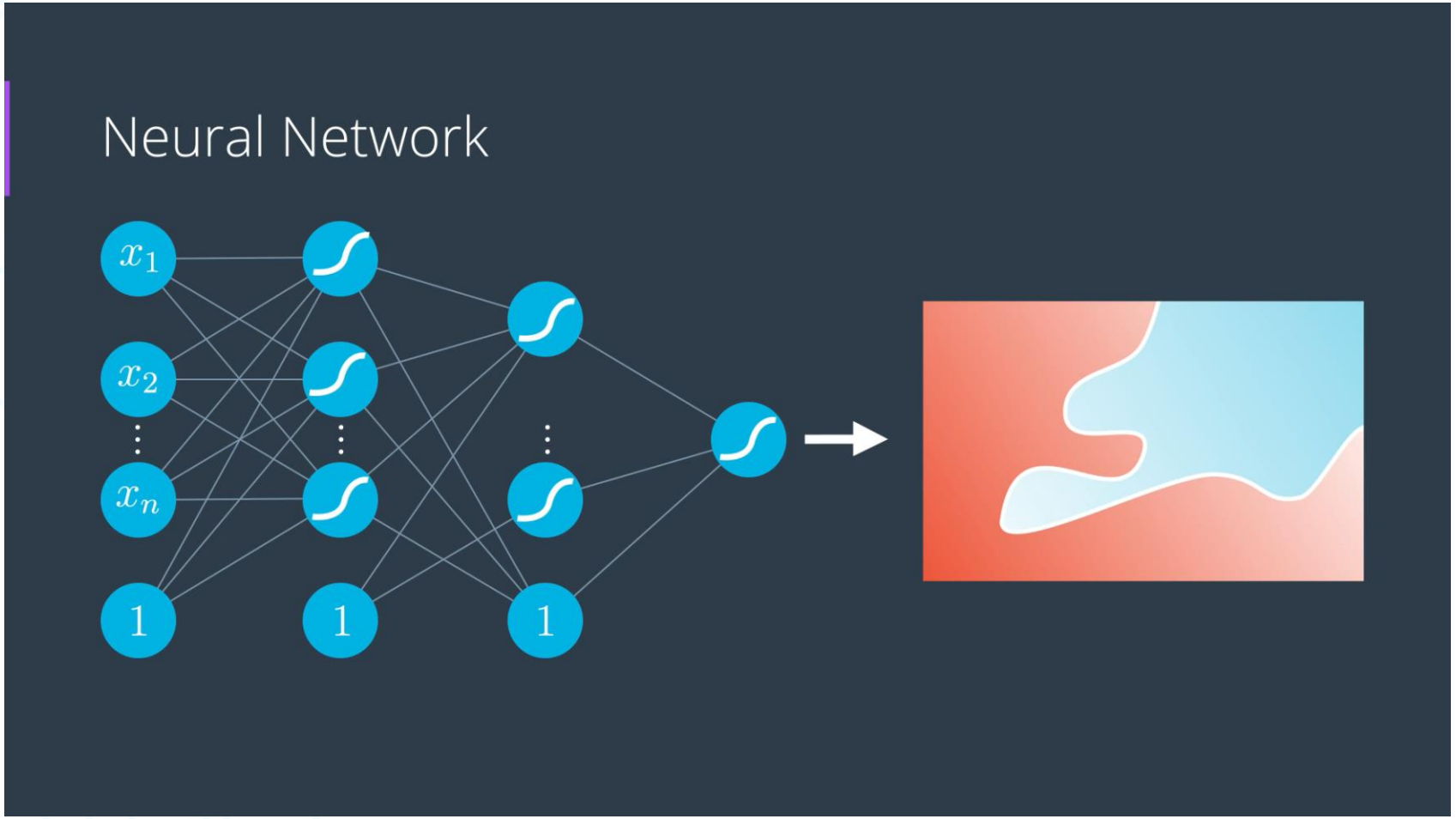
Redes Neurais – Multiple Layers



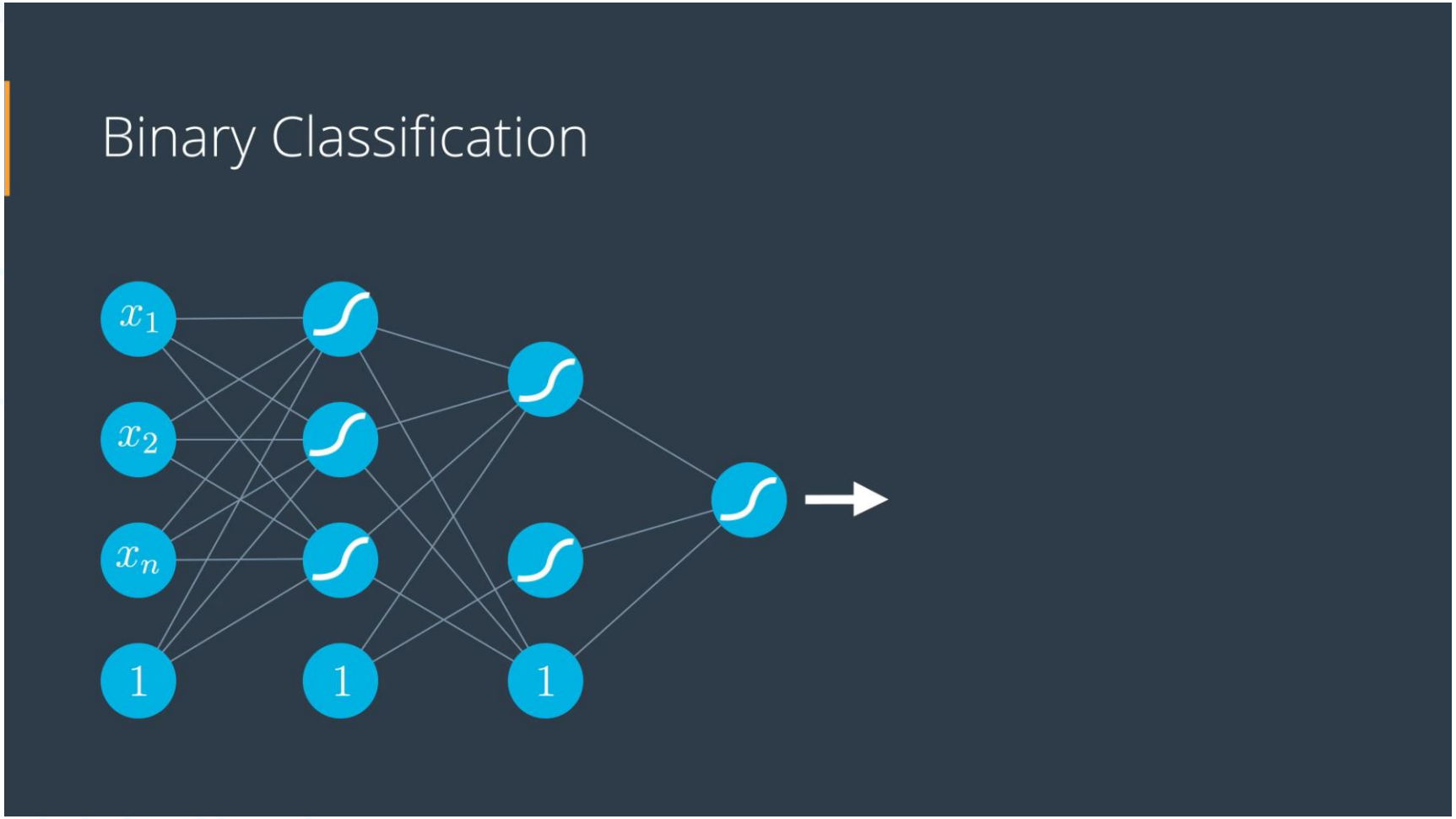
Redes Neurais – Multiple Layers



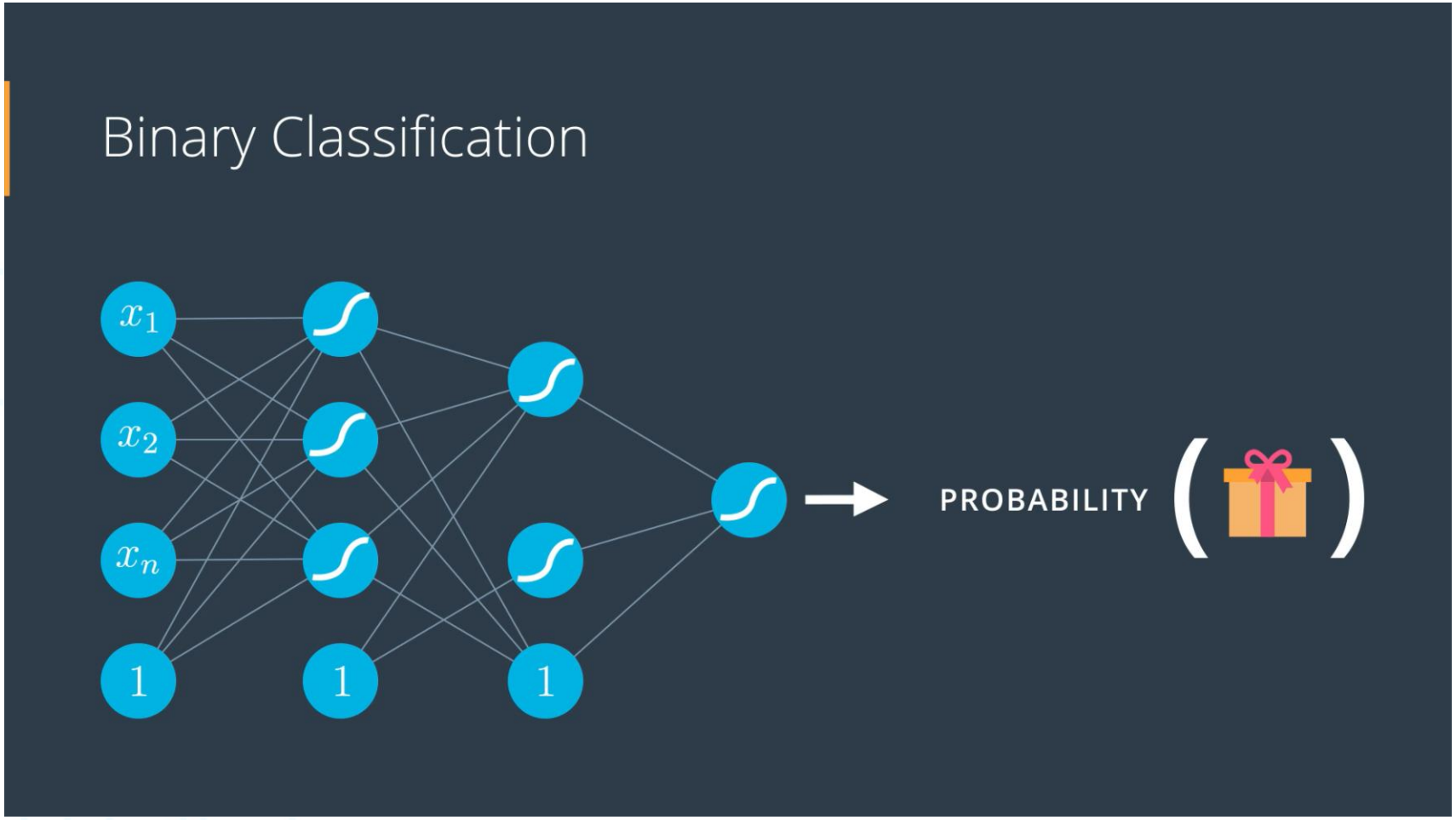
Redes Neurais – Multiple Layers



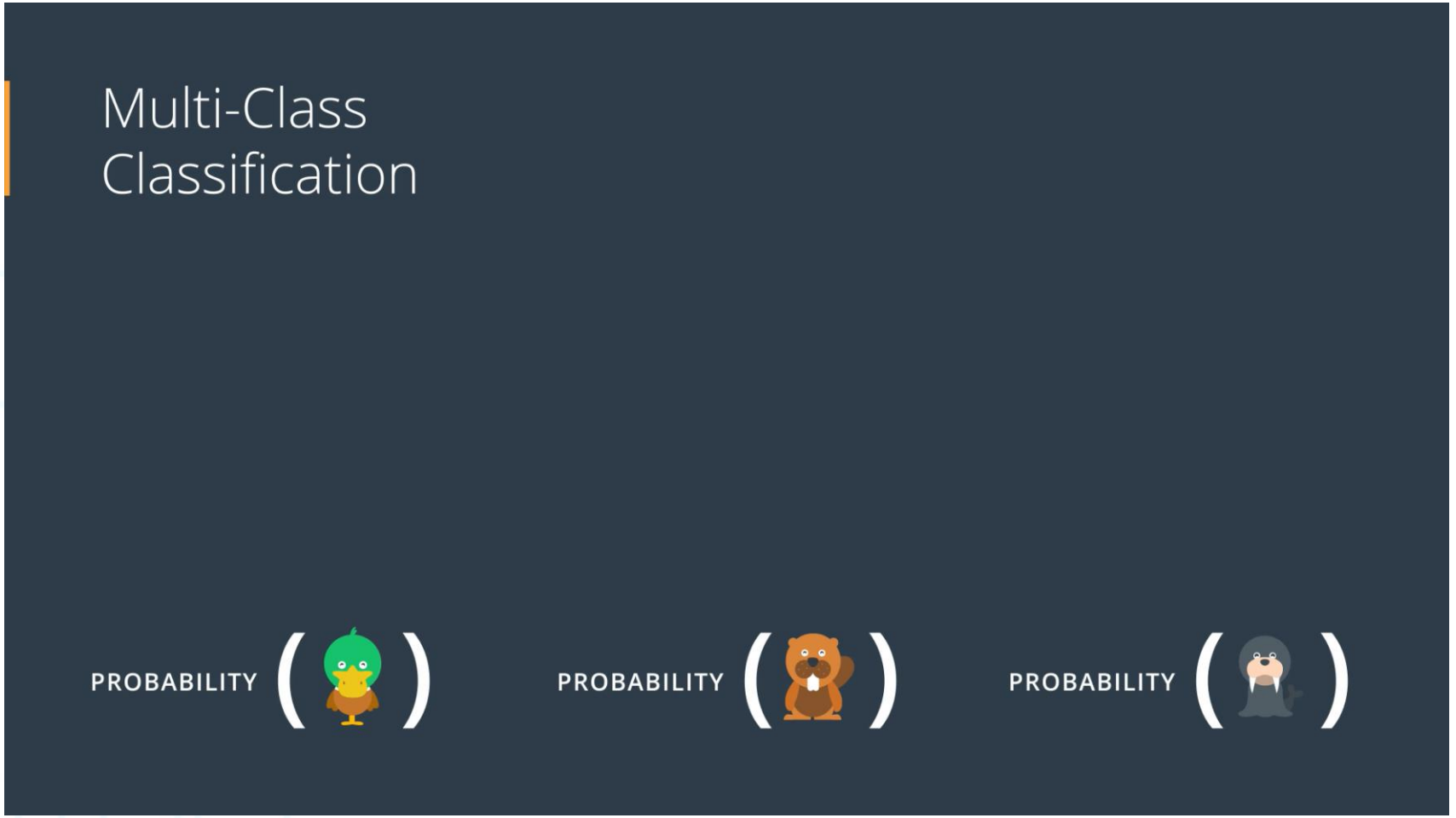
Redes Neurais – Multi-Class classification



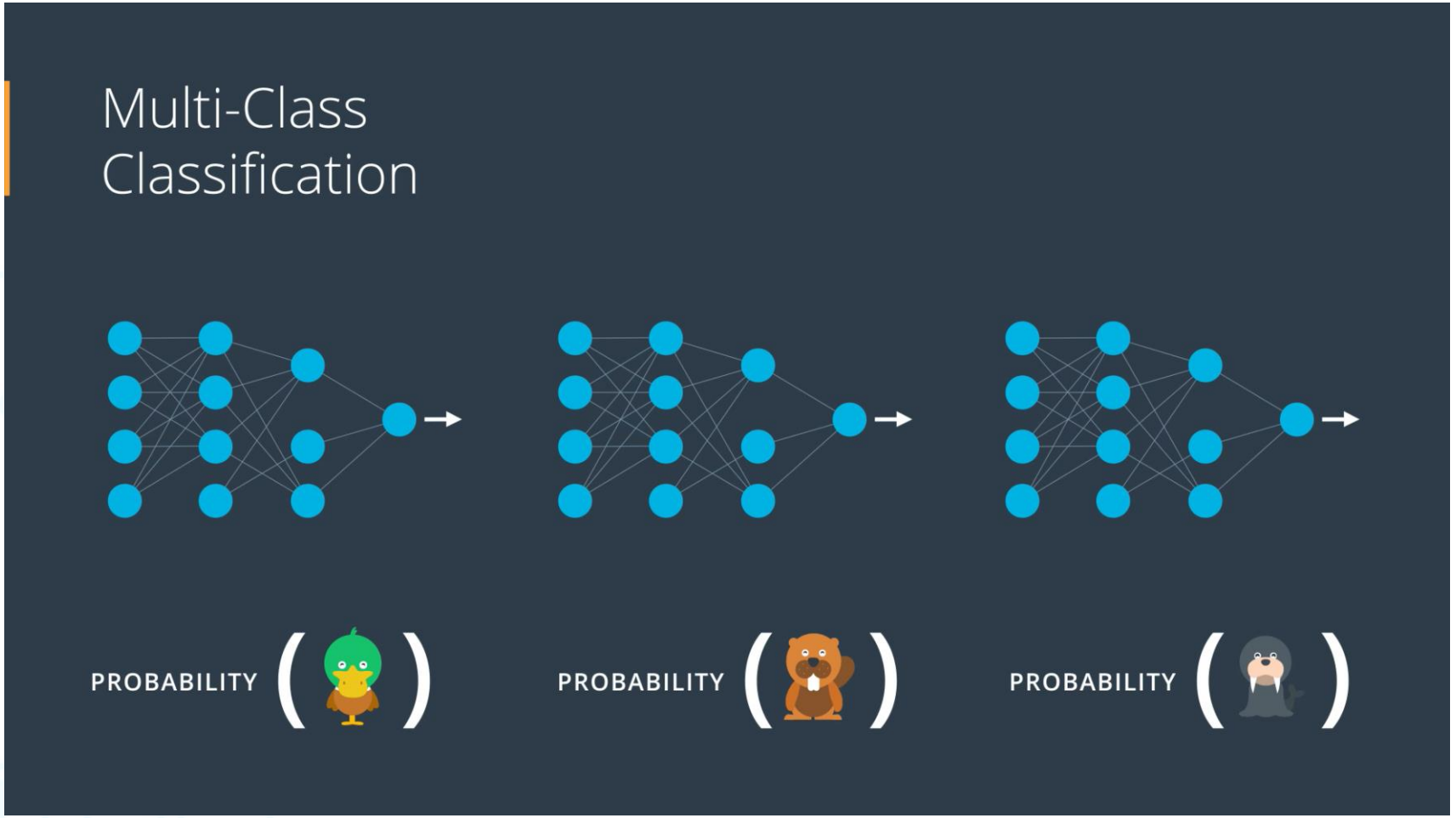
Redes Neurais – Multi-Class classification



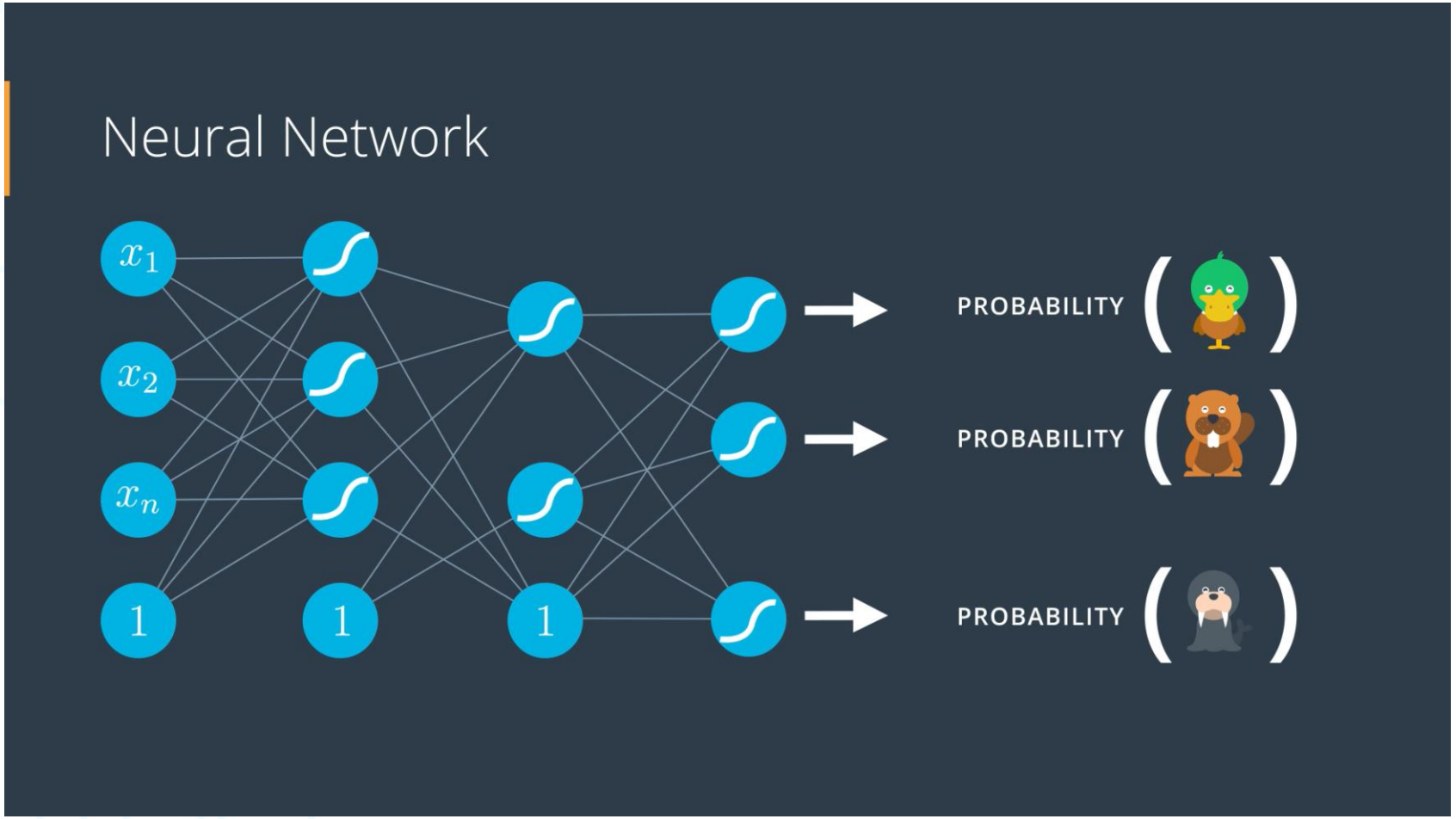
Redes Neurais – Multi-Class classification



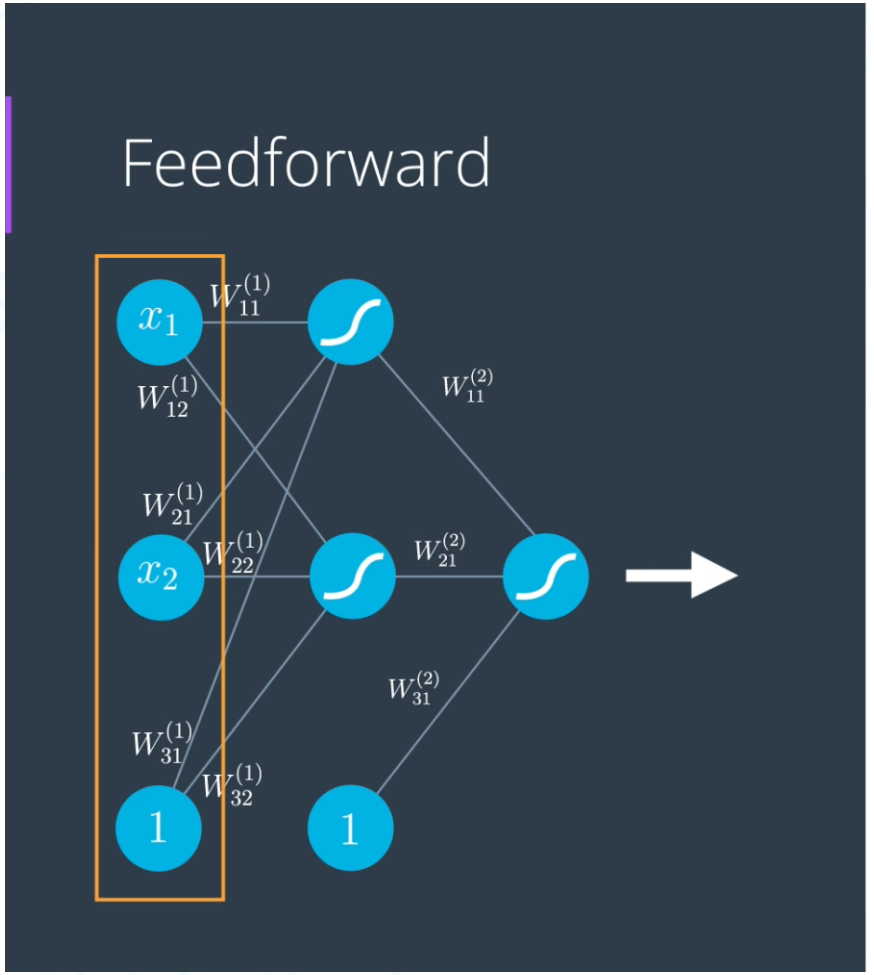
Redes Neurais – Multi-Class classification



Redes Neurais – Multi-Class classification

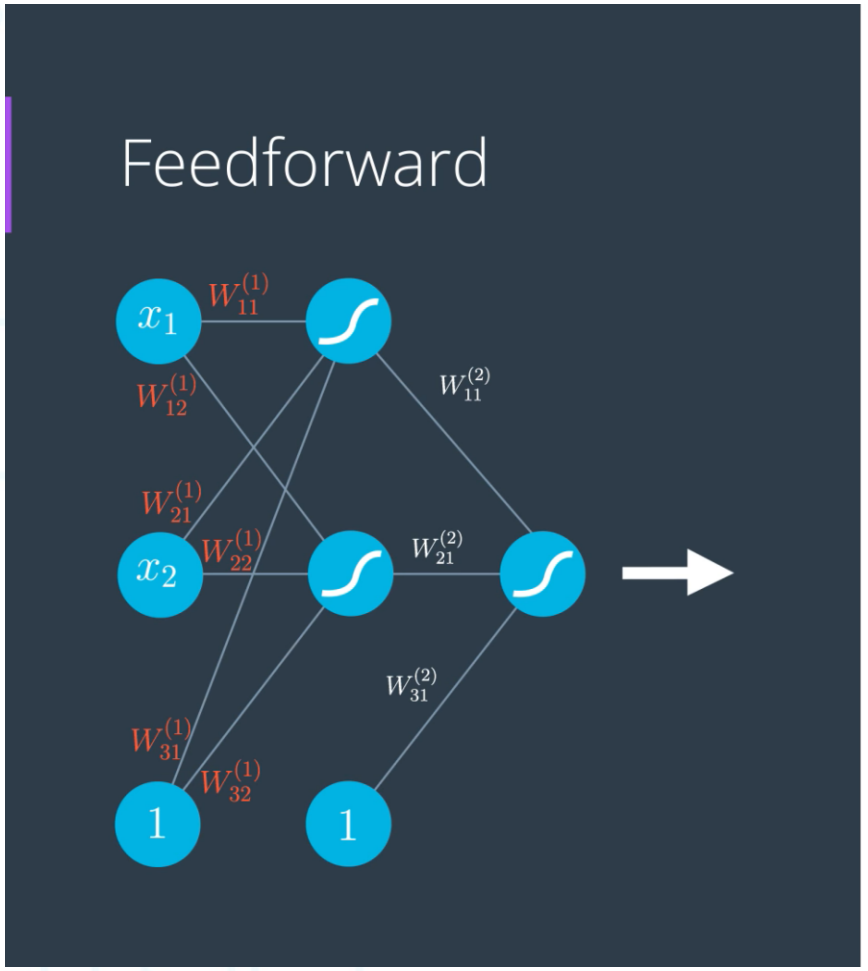


Redes Neurais – Feedforward



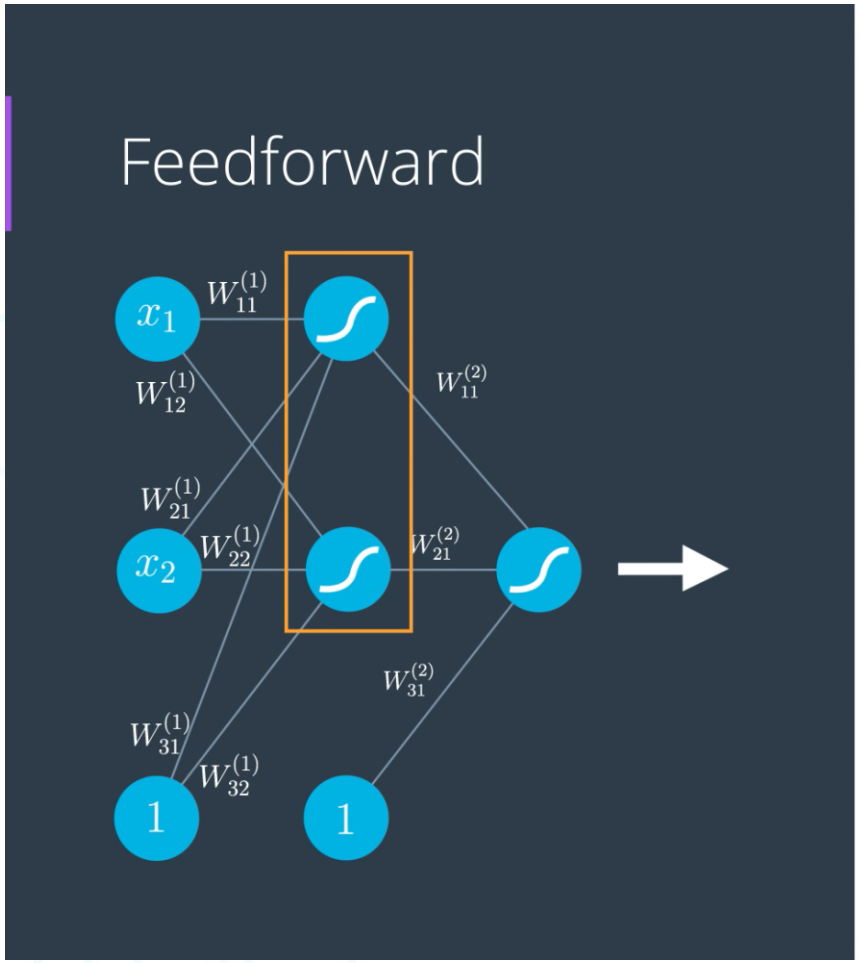
$$\begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Redes Neurais – Feedforward



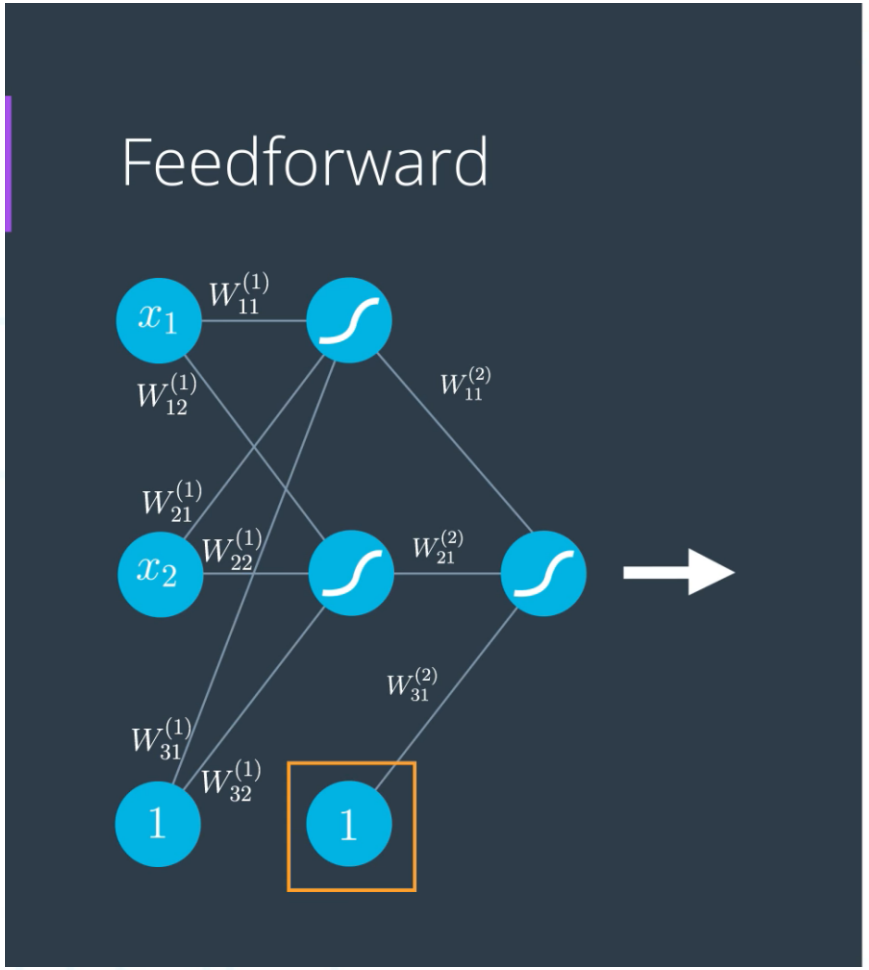
$$\begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Redes Neurais – Feedforward



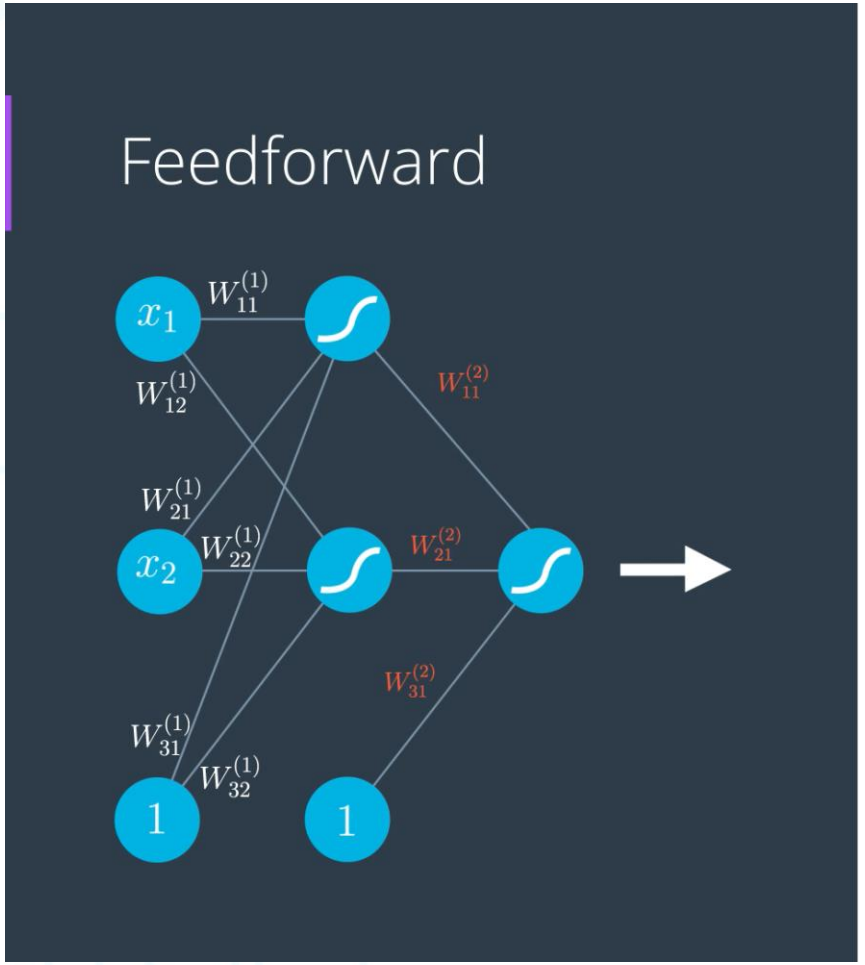
$$\sigma \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Redes Neurais – Feedforward



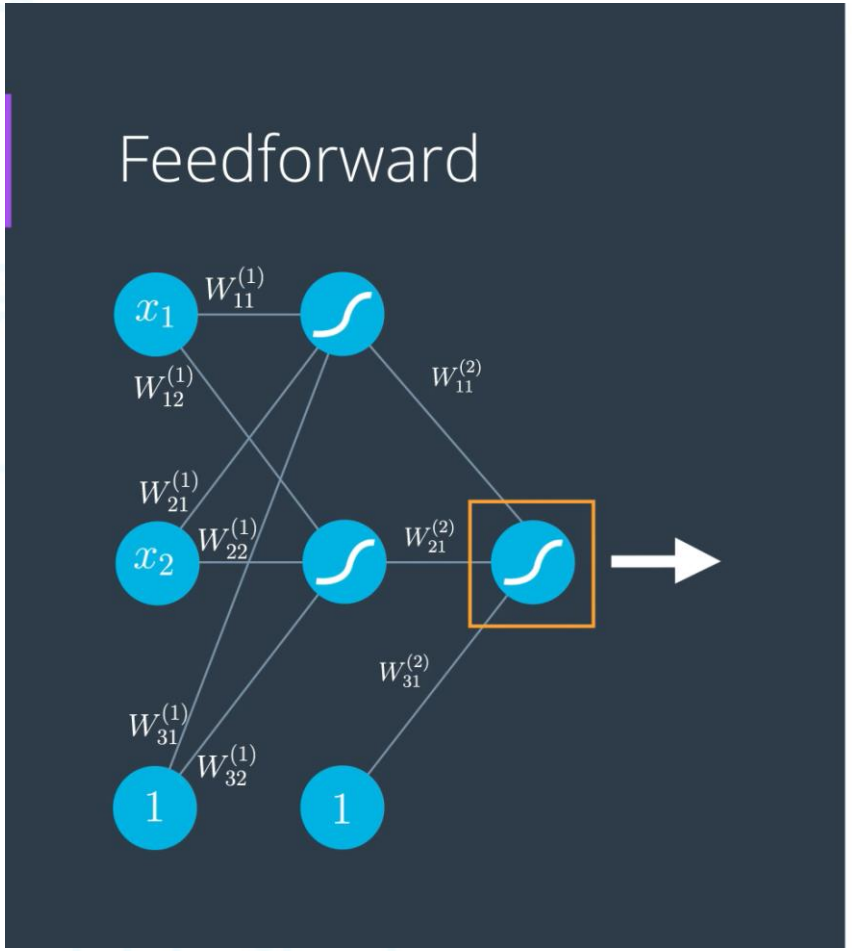
$$\sigma \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Redes Neurais – Feedforward



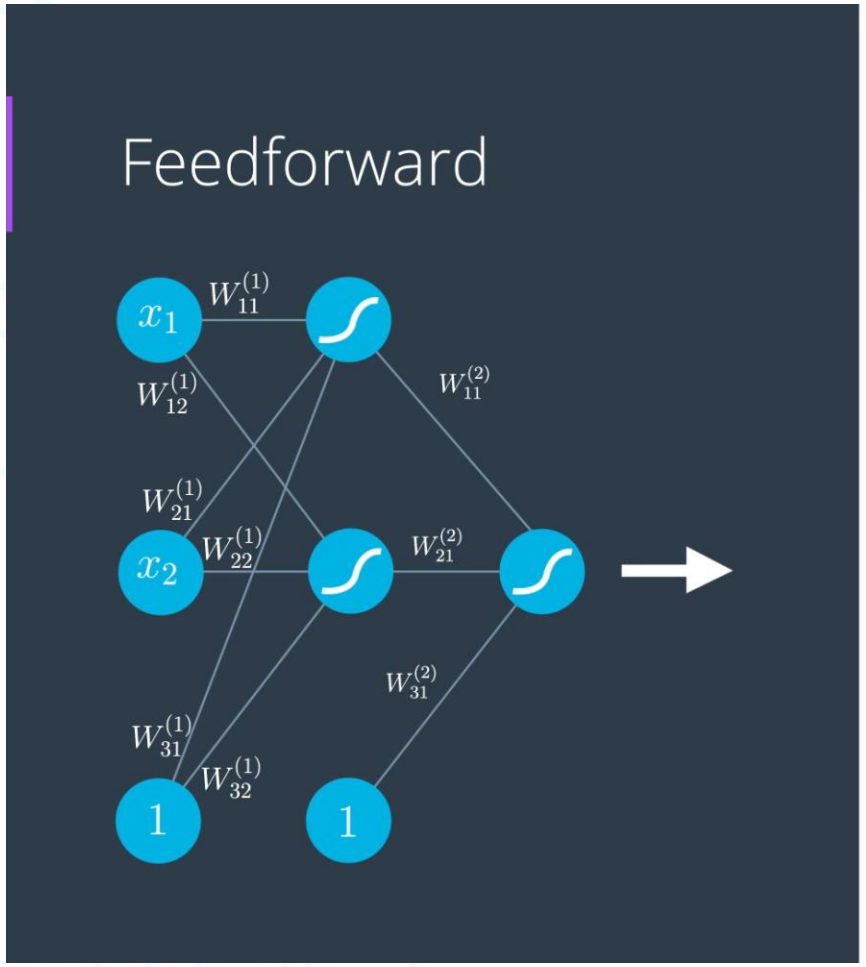
$$\begin{pmatrix} W_{11}^{(2)} \\ W_{21}^{(2)} \\ W_{31}^{(2)} \end{pmatrix} \sigma \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Redes Neurais – Feedforward



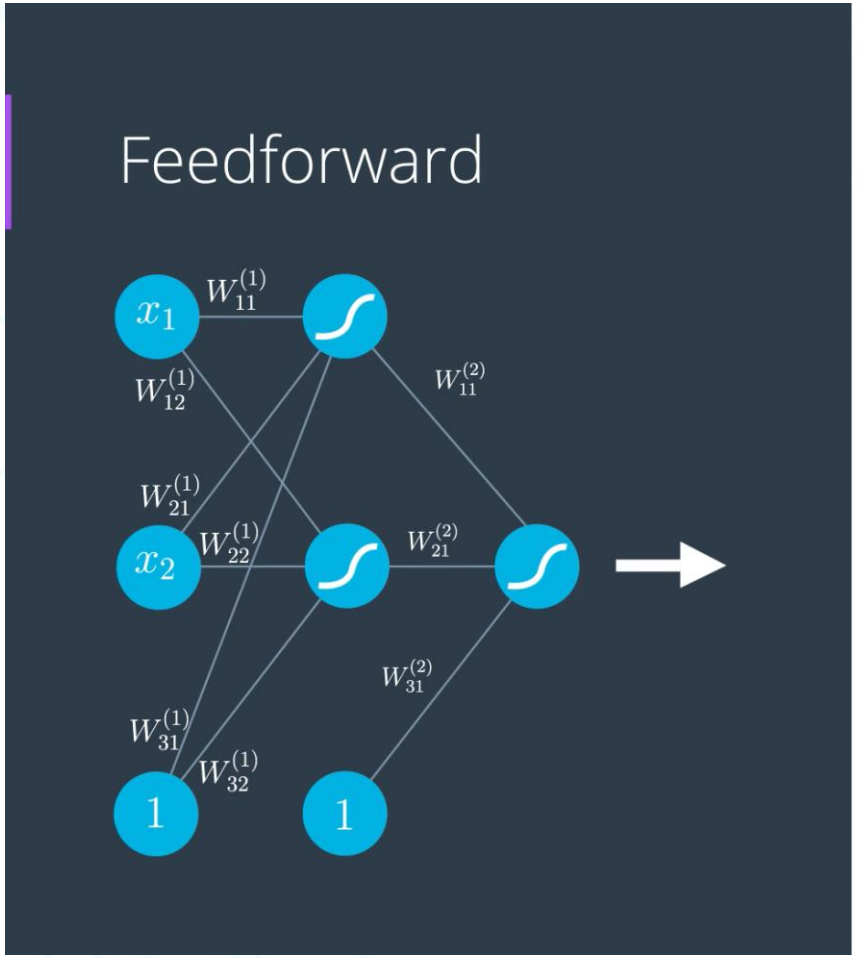
$$\sigma \begin{pmatrix} W_{11}^{(2)} \\ W_{21}^{(2)} \\ W_{31}^{(2)} \end{pmatrix} \sigma \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Redes Neurais – Feedforward



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

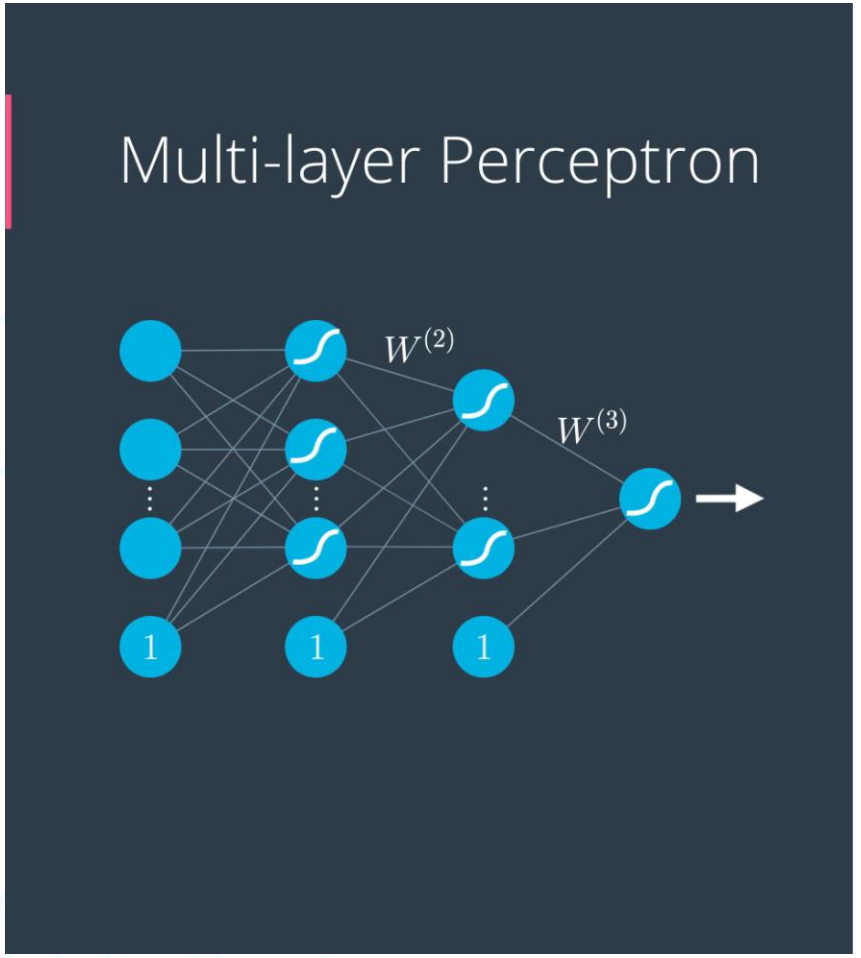
Redes Neurais – Feedforward



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

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

Redes Neurais – Error Function

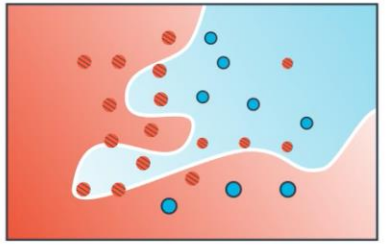


PREDICTION

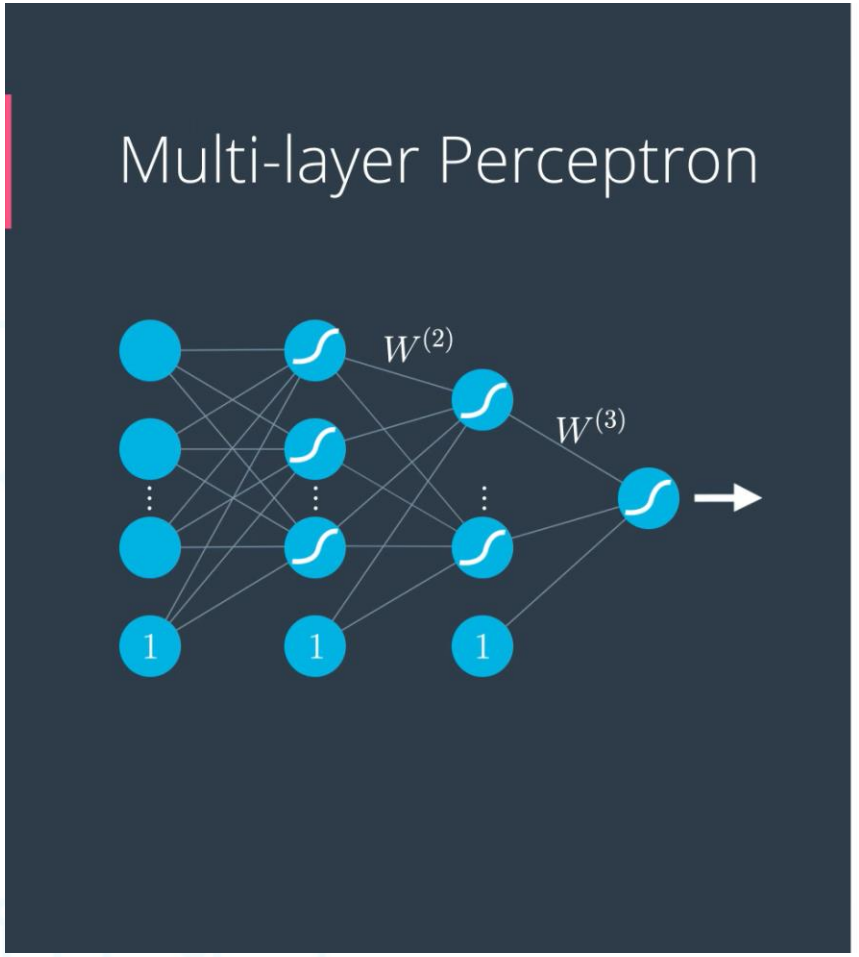
$$\hat{y} = \sigma \circ W^{(3)} \circ \sigma \circ W^{(2)} \circ \sigma \circ W^{(1)}(x)$$

ERROR FUNCTION

$$E(W) = -\frac{1}{m}\sum_{i=1}^{m} y_i ln(\hat{y}_i) + (1 - y_i) ln(1 - \hat{y}_i)$$



Redes Neurais – Error Function

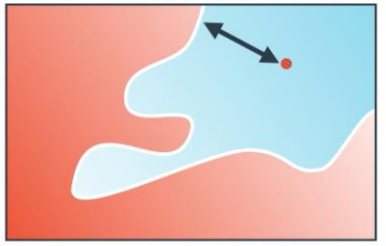


PREDICTION

$$\hat{y} = \sigma \circ W^{(3)} \circ \sigma \circ W^{(2)} \circ \sigma \circ W^{(1)}(x)$$

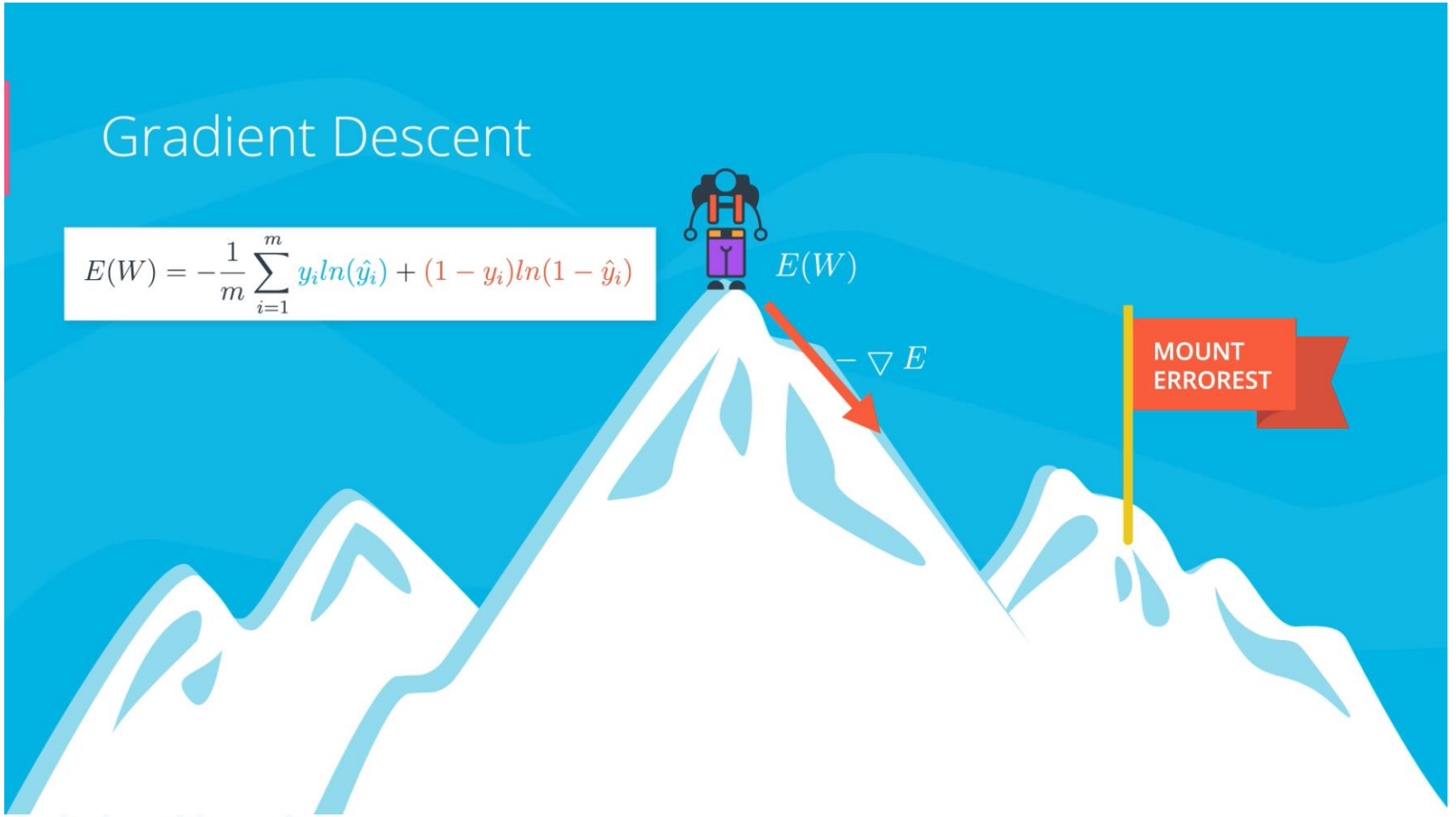
ERROR FUNCTION

$$E(W) = -\frac{1}{m}\sum_{i=1}^{m} y_i ln(\hat{y}_i) + (1 - y_i) ln(1 - \hat{y}_i)$$

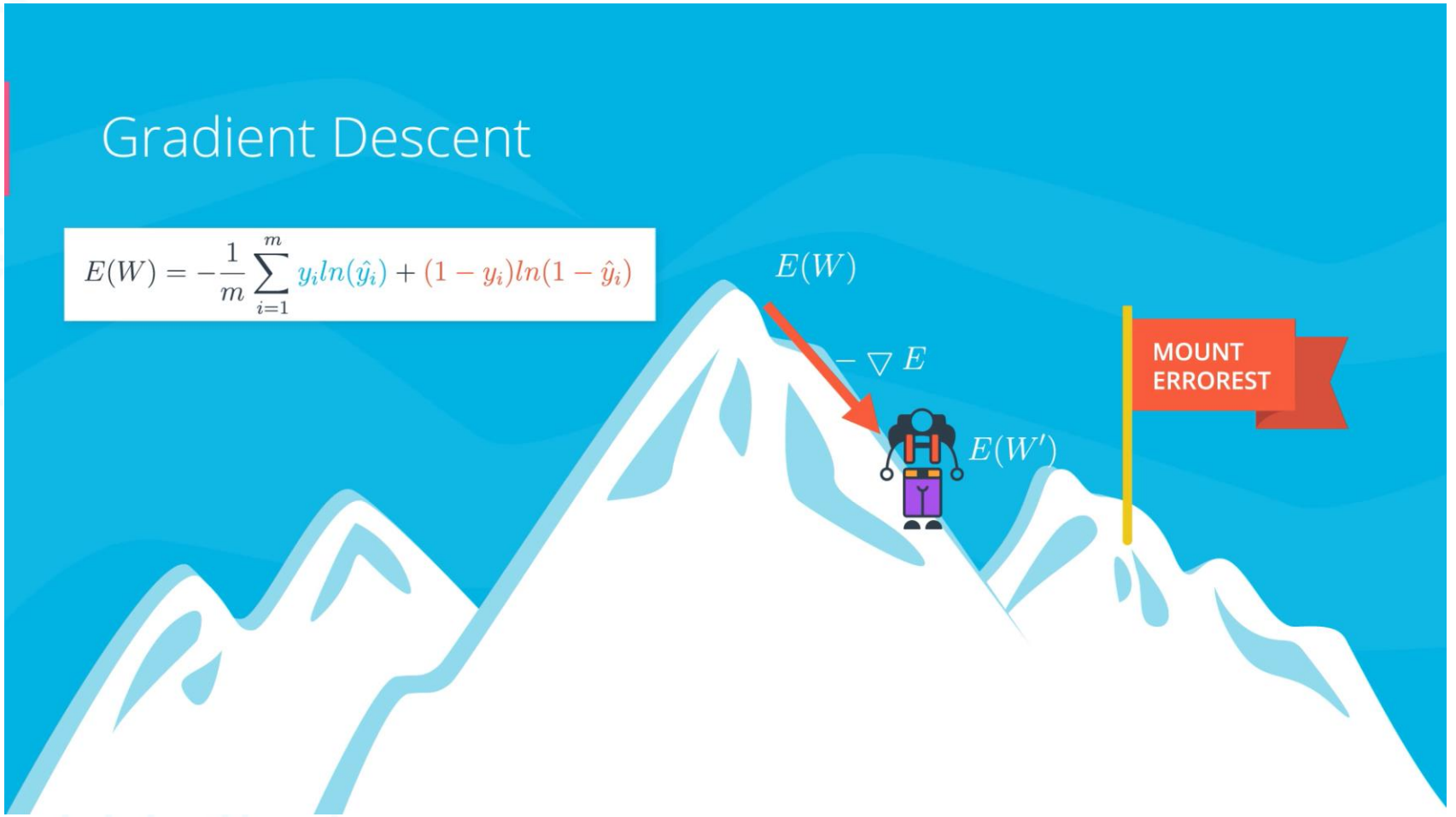


Redes Neurais – Backpropagation

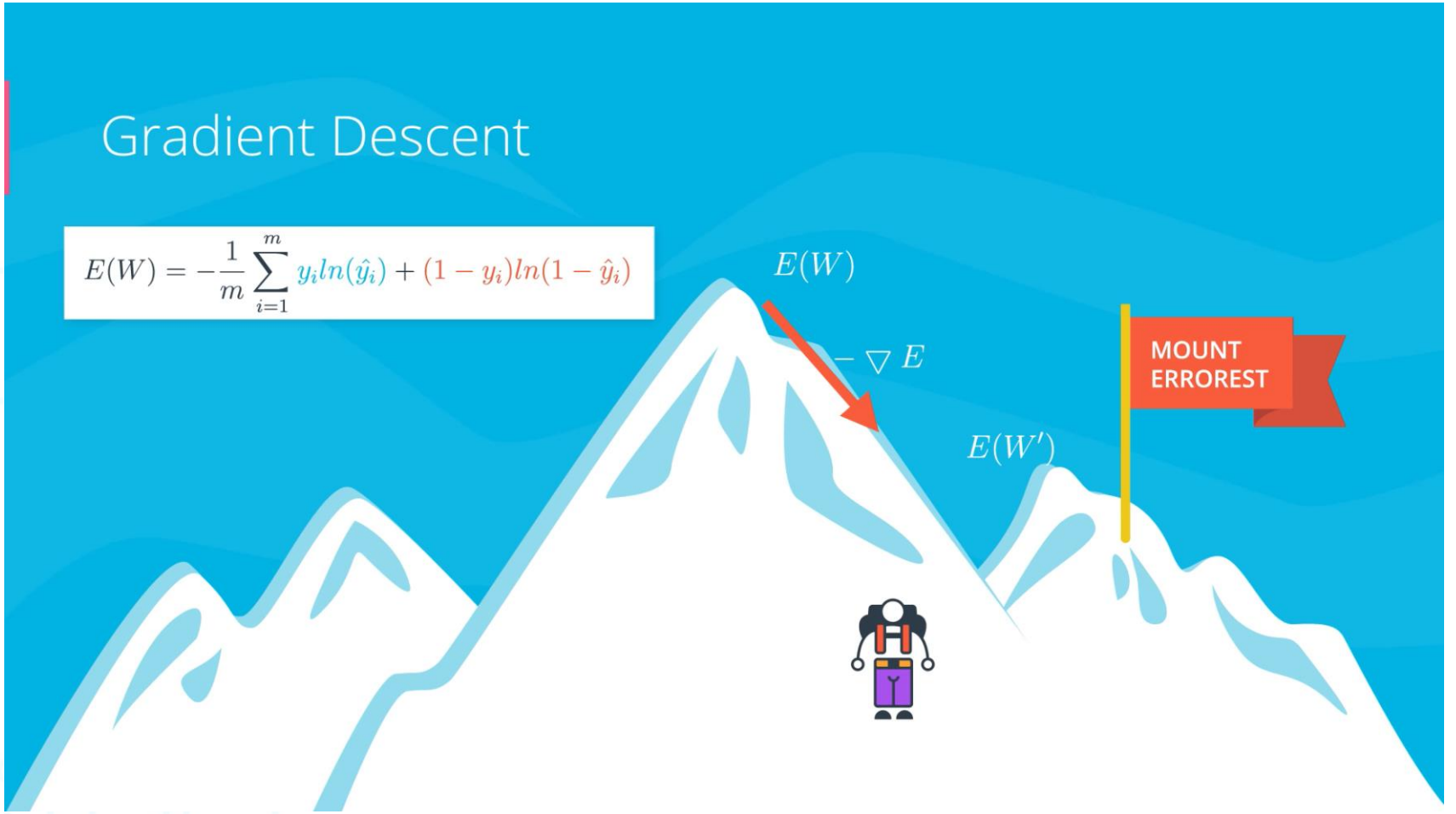
Redes Neurais – Backpropagation



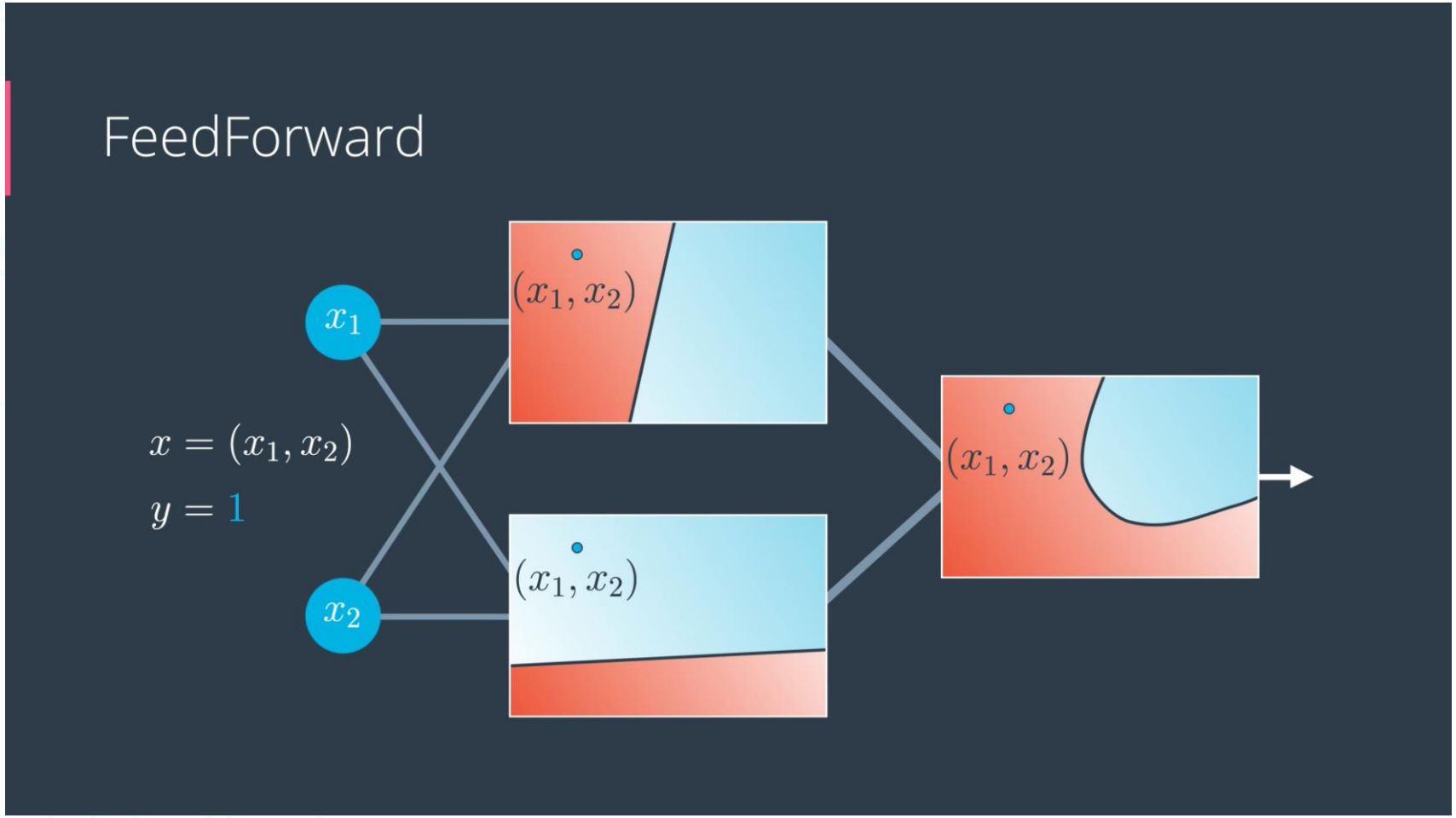
Redes Neurais – Backpropagation



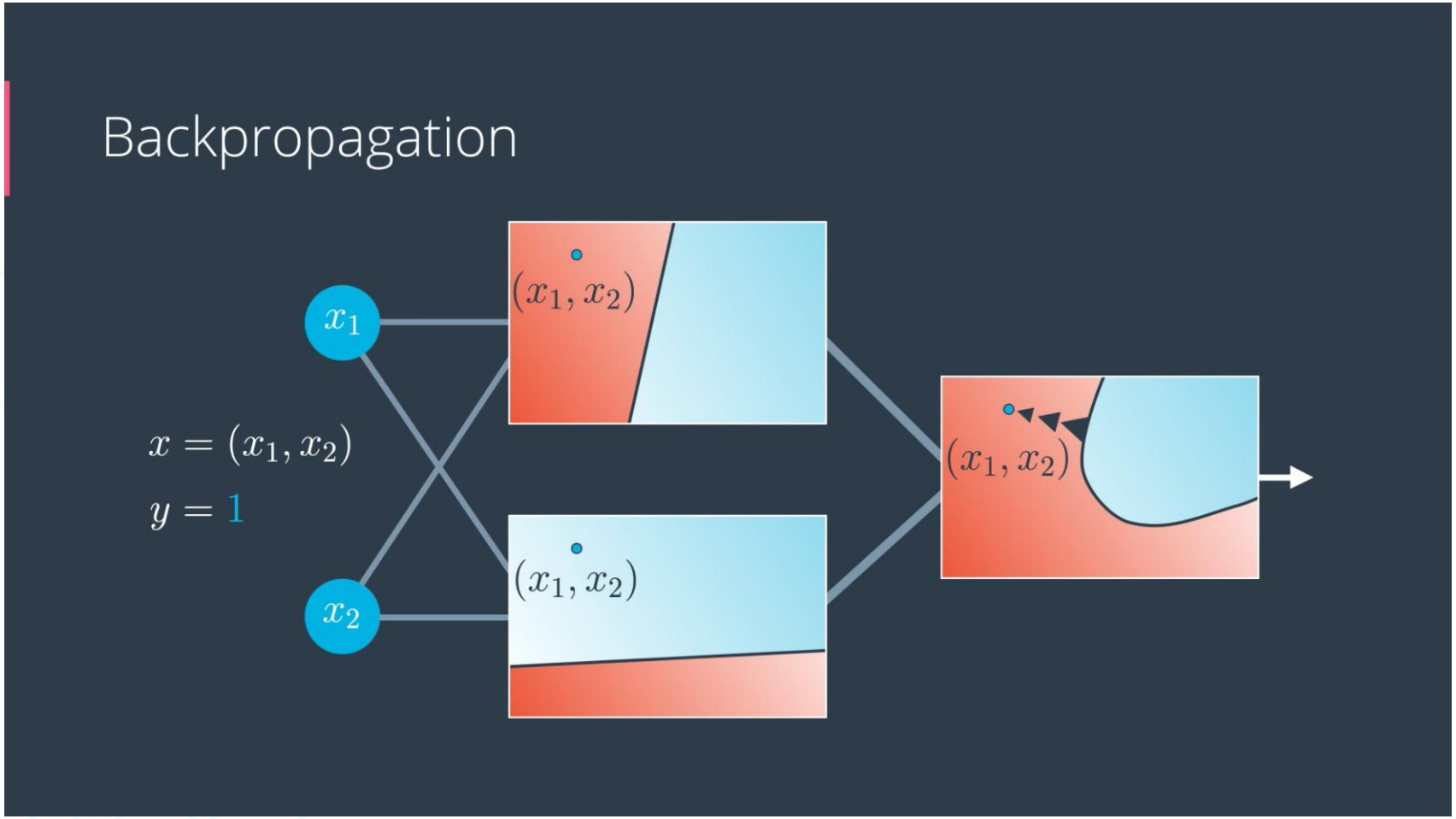
Redes Neurais – Backpropagation



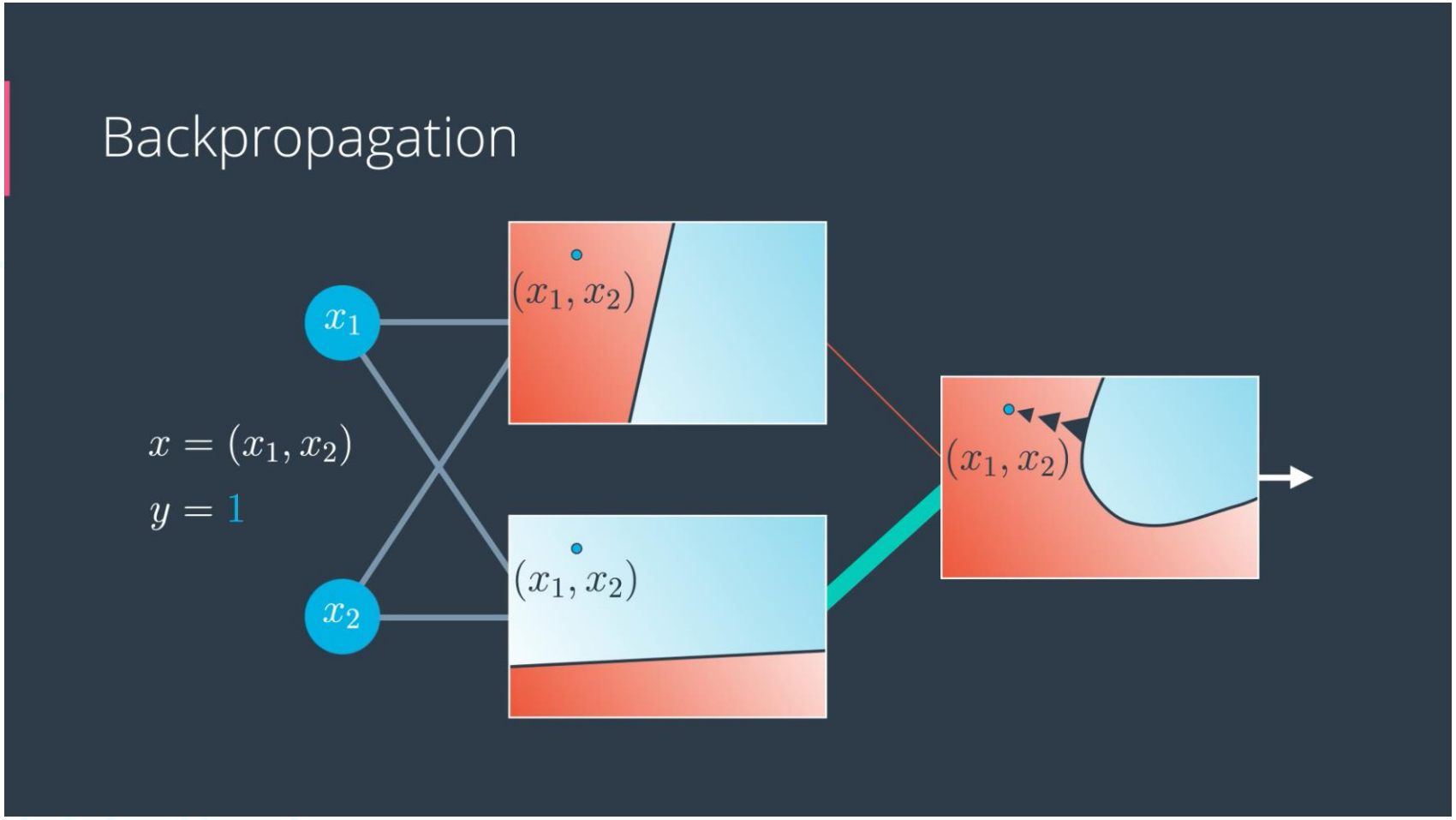
Redes Neurais – Backpropagation



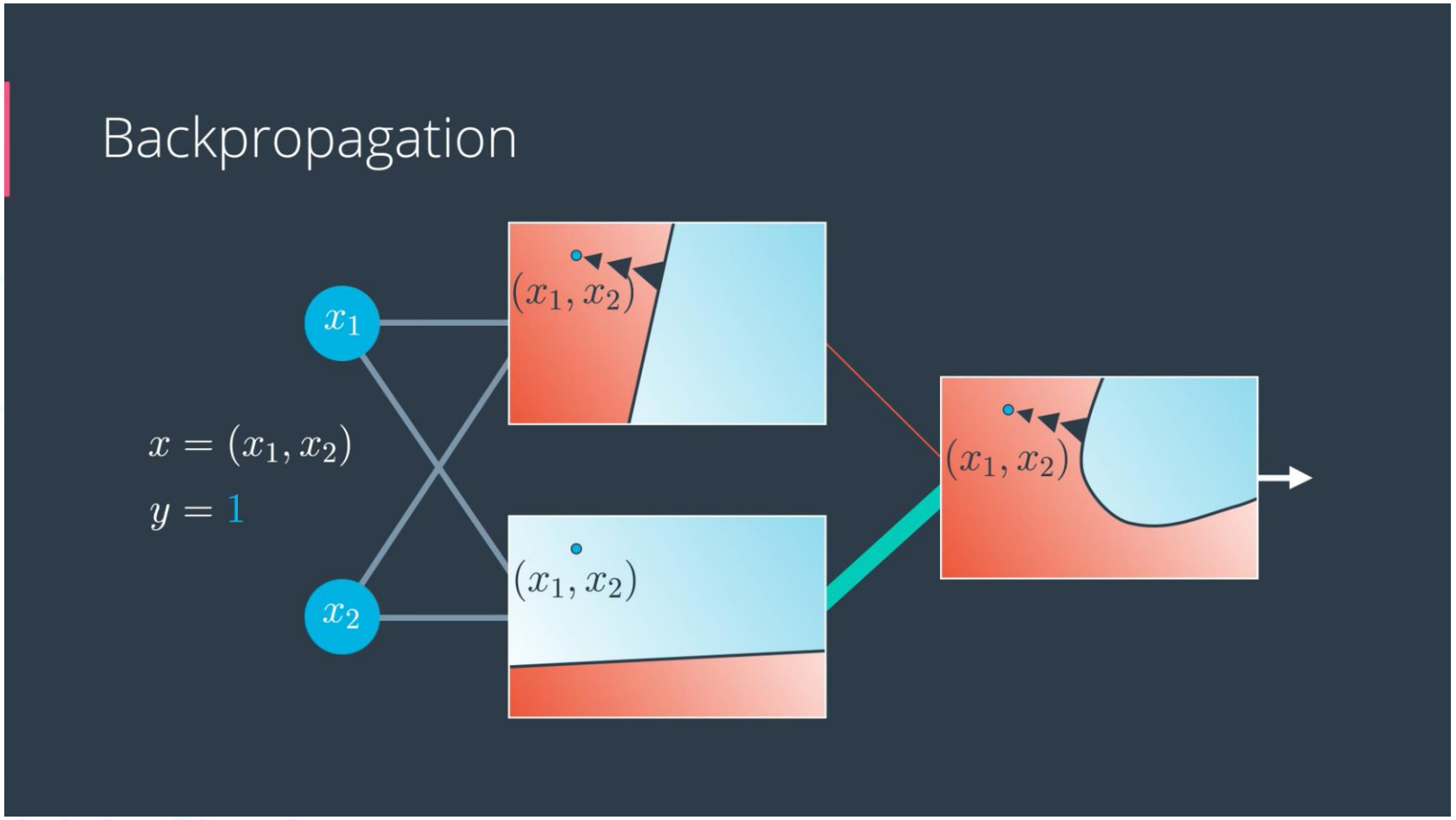
Redes Neurais – Backpropagation



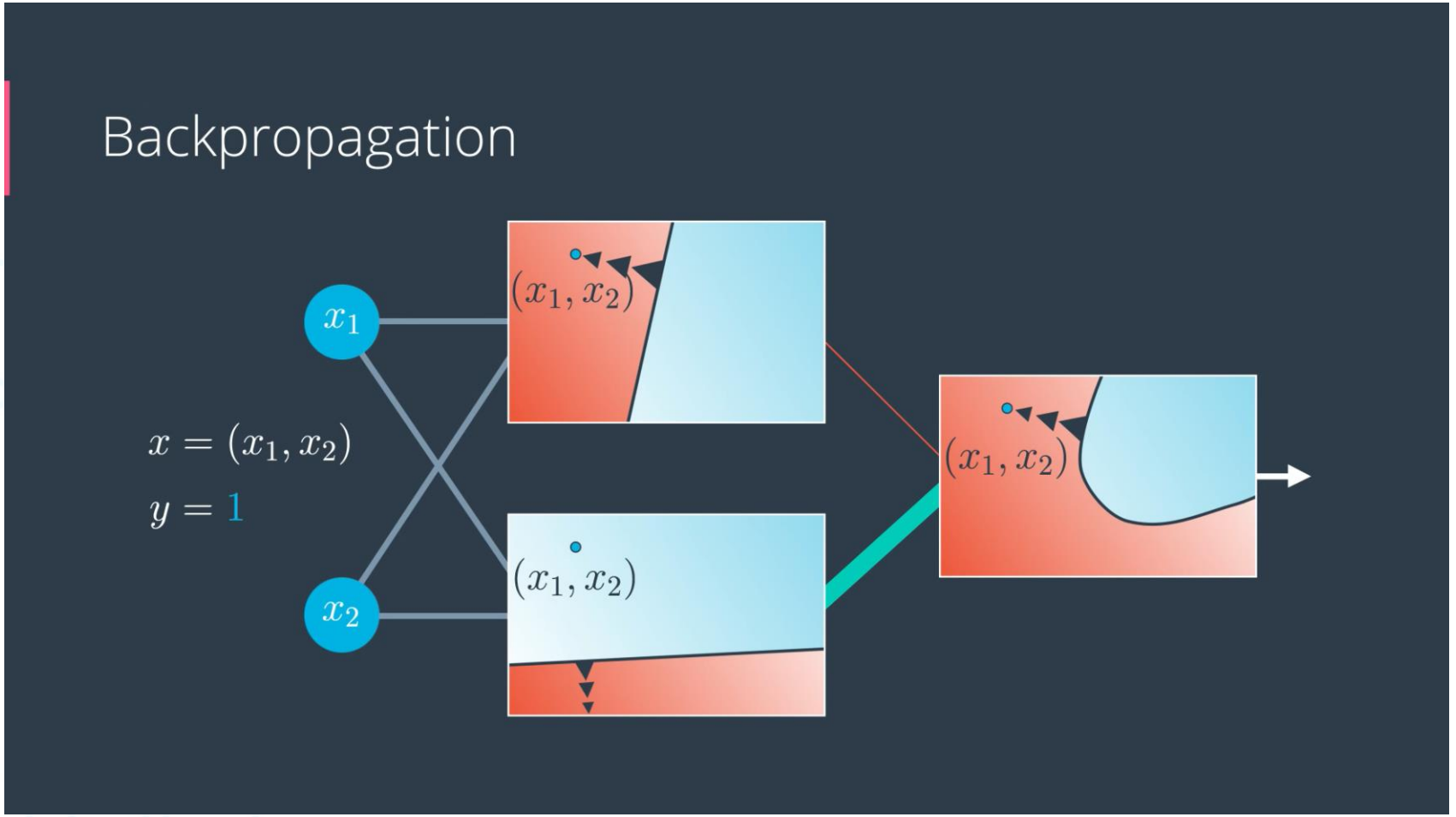
Redes Neurais – Backpropagation



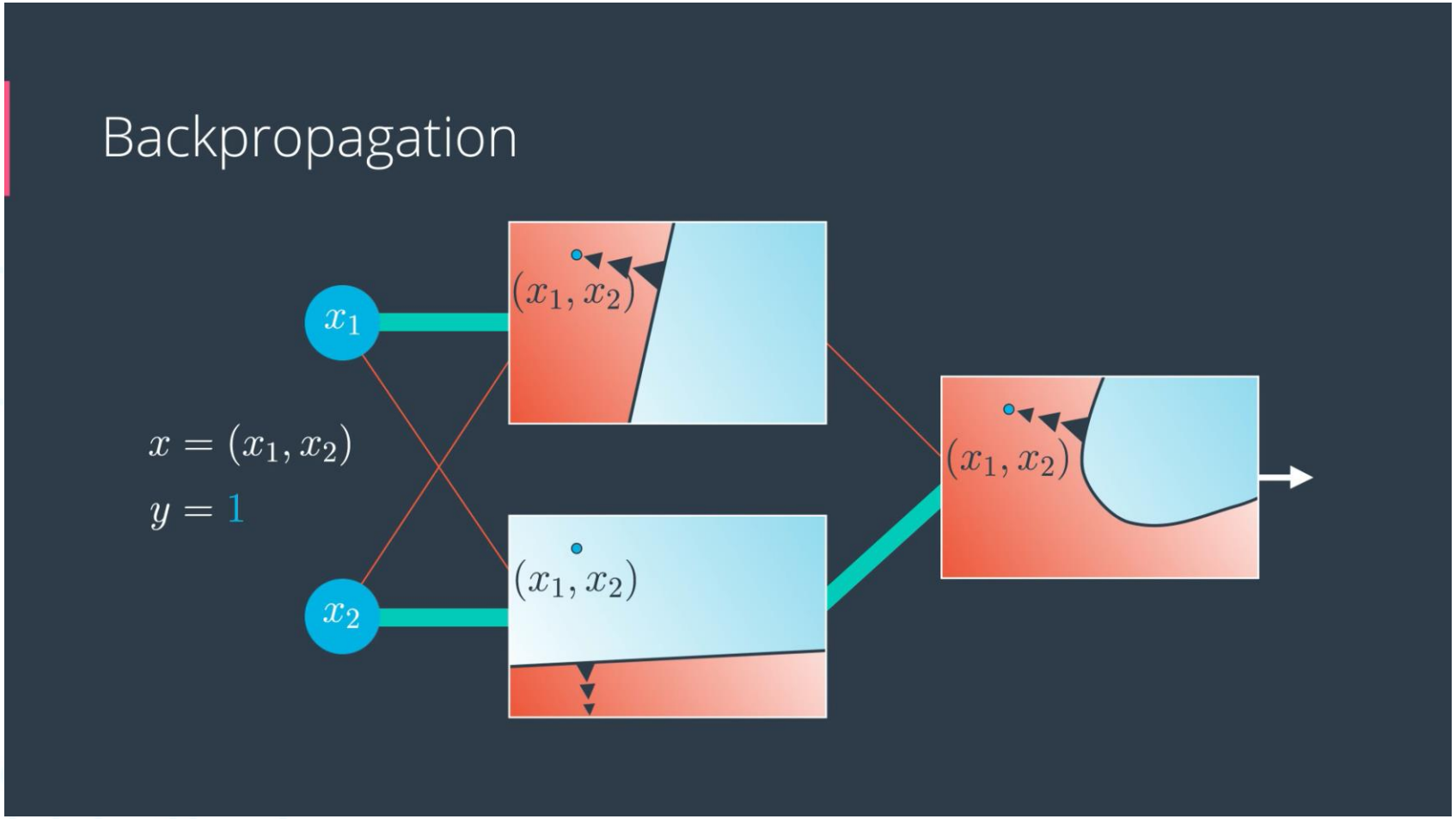
Redes Neurais – Backpropagation



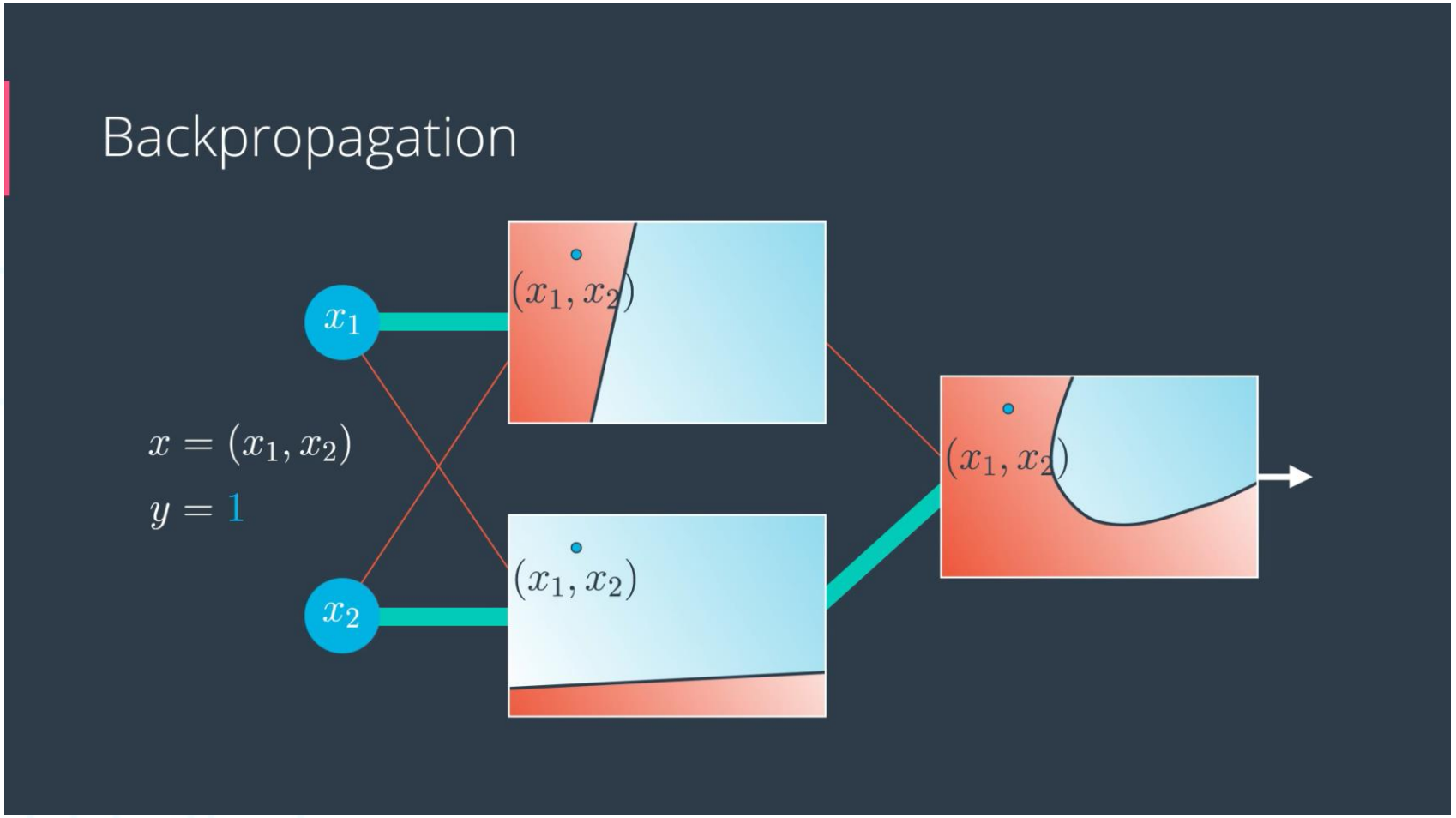
Redes Neurais – Backpropagation



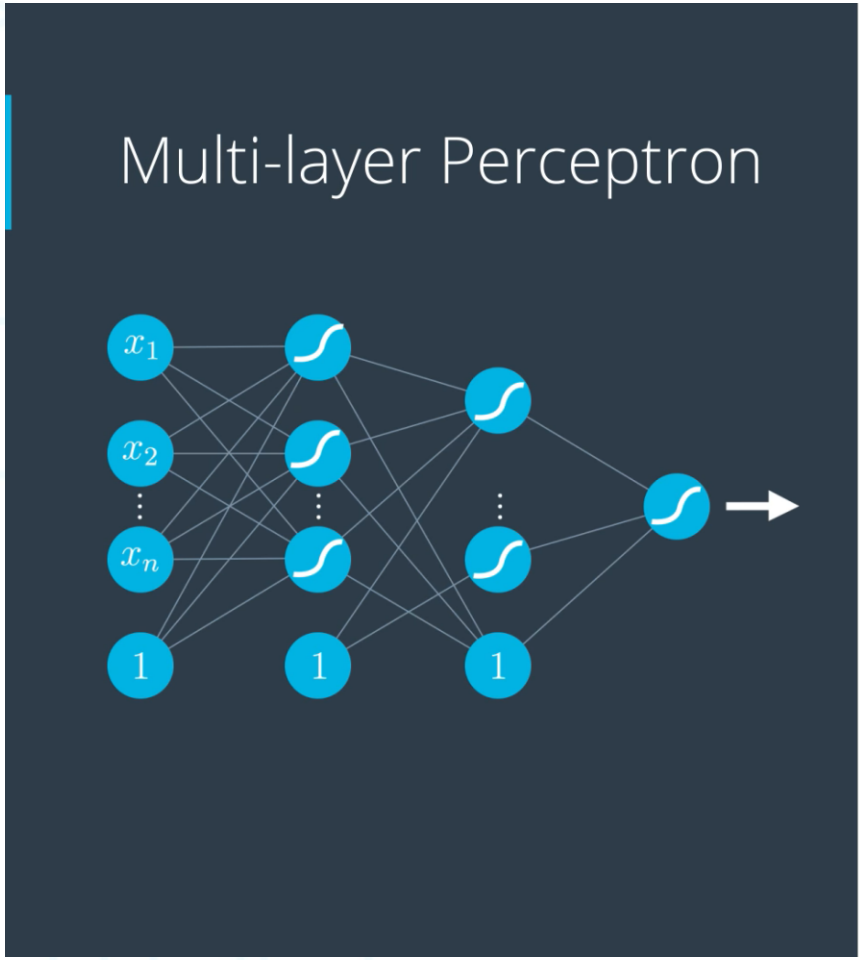
Redes Neurais – Backpropagation



Redes Neurais – Backpropagation



Redes Neurais – Backpropagation



PREDICTION

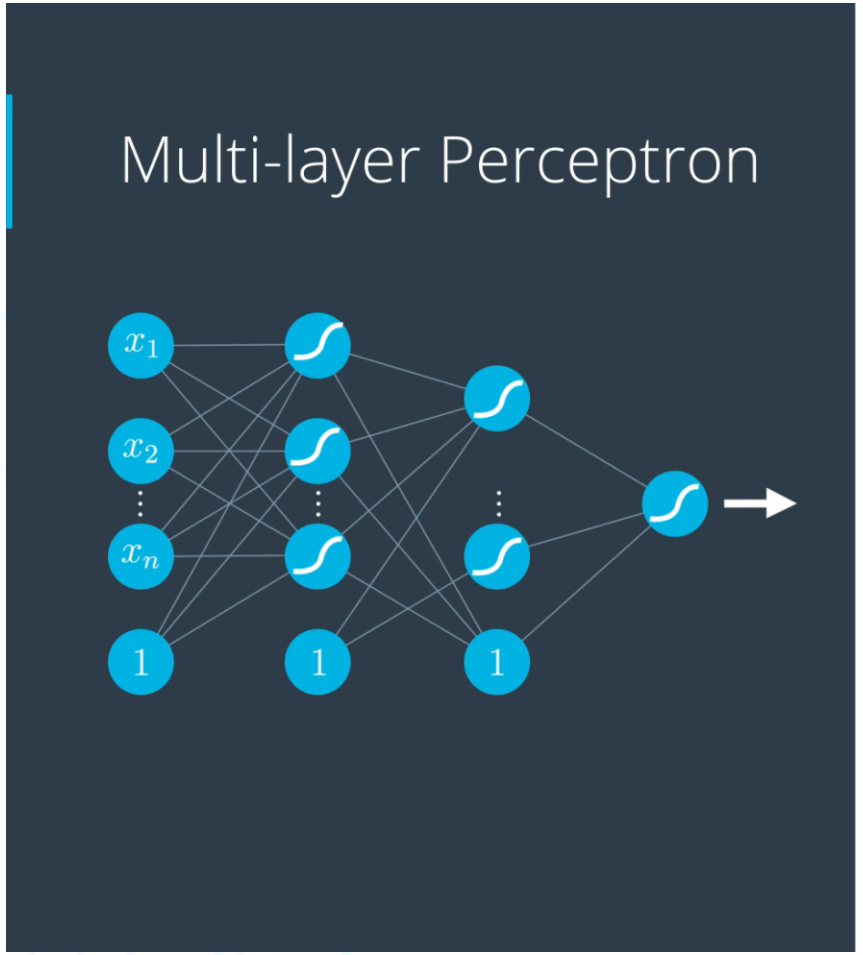
$$\hat{y} = \sigma W^{(3)} \circ \sigma W^{(2)} \circ \sigma \circ W^{(1)}(x)$$

ERROR FUNCTION

$$E(W) = -\frac{1}{m}\sum_{i=1}^{m} y_i ln(\hat{y}_i) + (1 - y_i) ln(1 - \hat{y}_i)$$

GRADIENT OF THE ERROR FUNCTION

Redes Neurais – Backpropagation



PREDICTION

$$\hat{y} = \sigma W^{(3)} \circ \sigma W^{(2)} \circ \sigma \circ W^{(1)}(x)$$

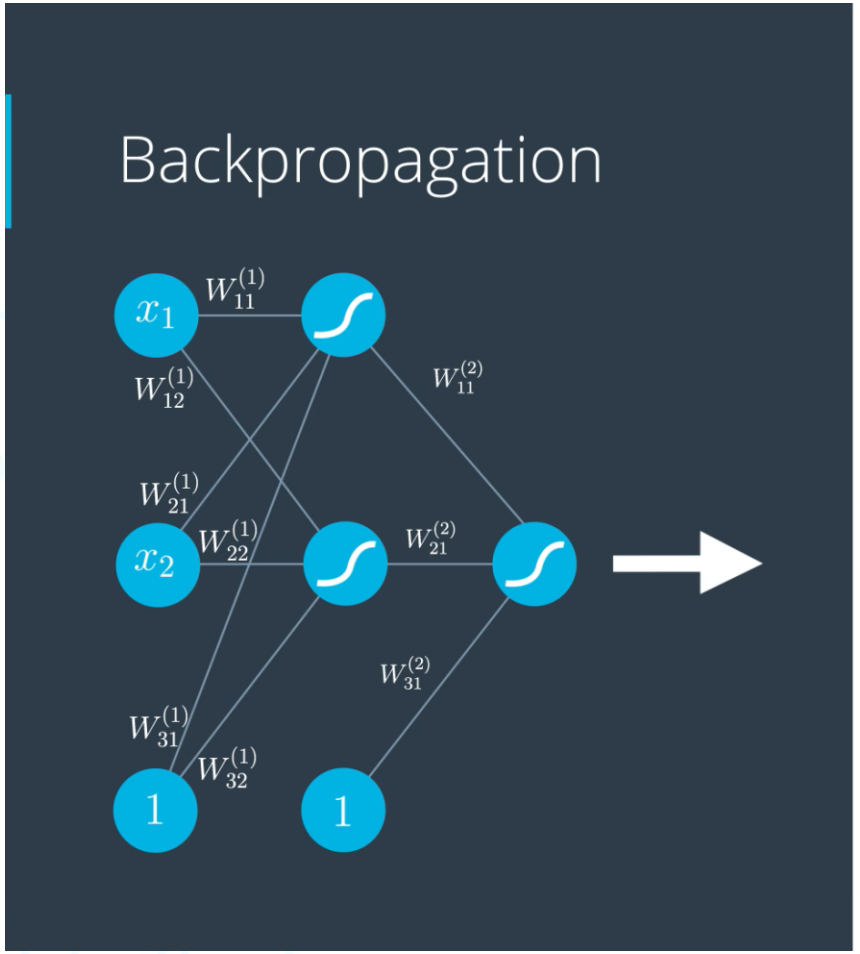
ERROR FUNCTION

$$E(W) = -\frac{1}{m}\sum_{i=1}^{m} y_i ln(\hat{y}_i) + (1 - y_i) ln(1 - \hat{y}_i)$$

GRADIENT OF THE ERROR FUNCTION

$$\nabla E = (..., \frac{\partial E}{\partial w_j^{(i)}}, ...)$$

Redes Neurais – Backpropagation



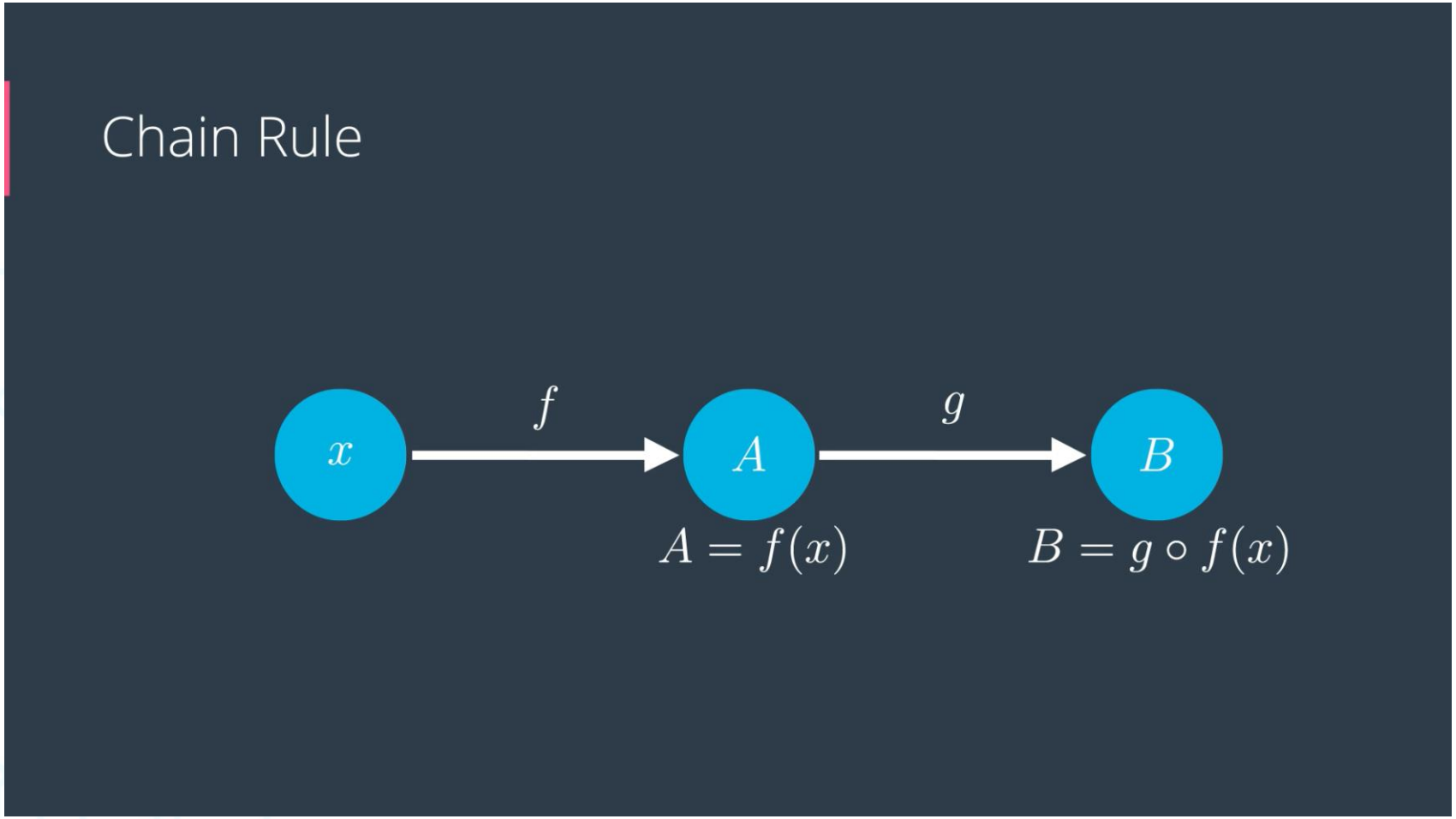
$$\hat{y} = \sigma W^{(2)} \circ \sigma \circ W^{(1)}(x)$$

$$W^{(1)} = \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{pmatrix} \quad W^{(2)} = \begin{pmatrix} W_{11}^{(2)} \\ W_{21}^{(2)} \\ W_{31}^{(2)} \end{pmatrix}$$

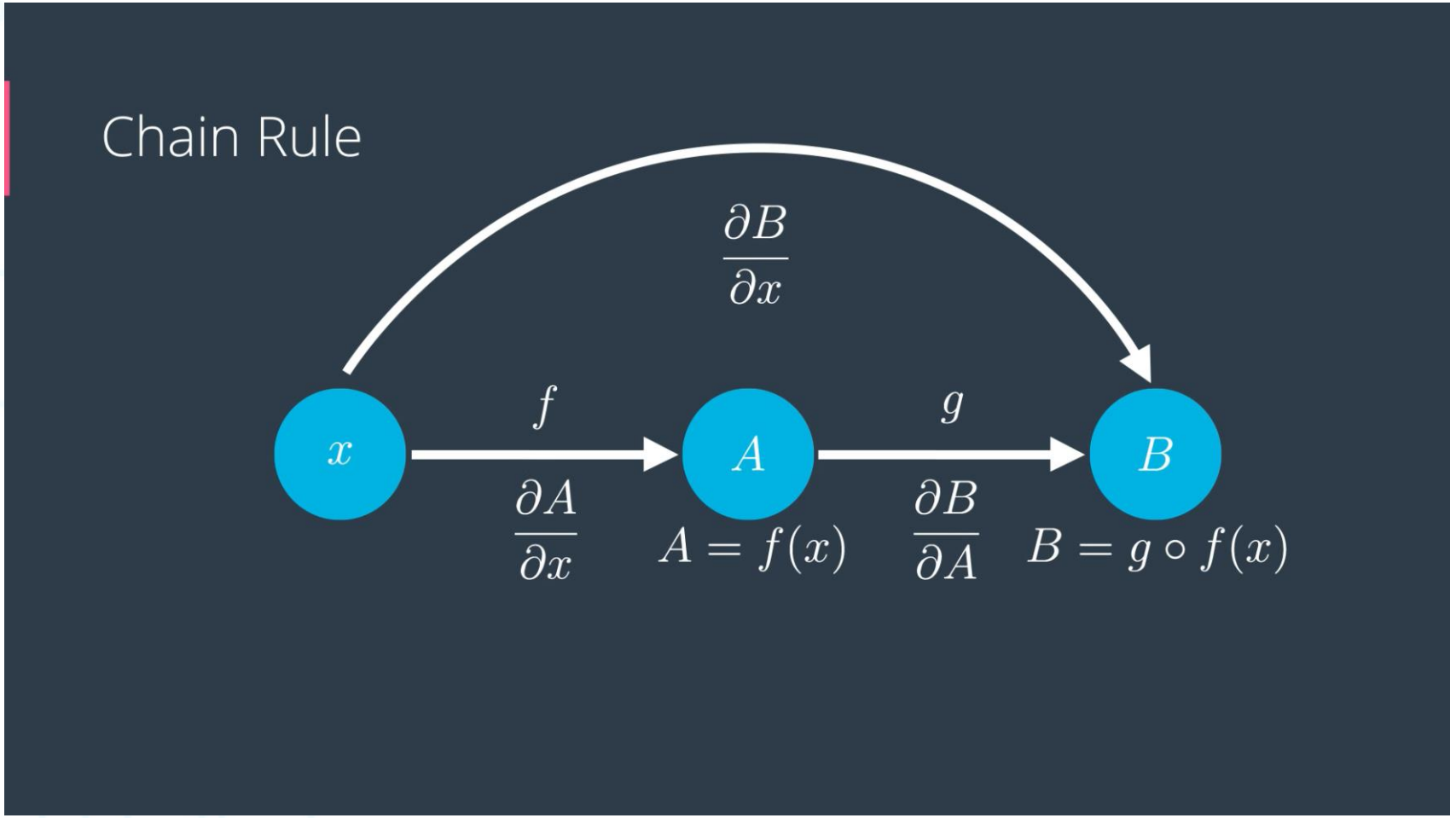
$$\nabla E = \begin{pmatrix} \frac{\partial E}{\partial W_{11}^{(1)}} & \frac{\partial E}{\partial W_{12}^{(1)}} & \frac{\partial E}{\partial W_{11}^{(2)}} \\ \frac{\partial E}{\partial W_{21}^{(1)}} & \frac{\partial E}{\partial W_{22}^{(1)}} & \frac{\partial E}{\partial W_{21}^{(2)}} \\ \frac{\partial E}{\partial W_{31}^{(1)}} & \frac{\partial E}{\partial W_{32}^{(1)}} & \frac{\partial E}{\partial W_{31}^{(2)}} \end{pmatrix}$$

$$W_{ij}^{\prime(k)} \leftarrow W_{ij}^{(k)} - \alpha \frac{\partial E}{\partial W_{ij}^{(k)}}$$

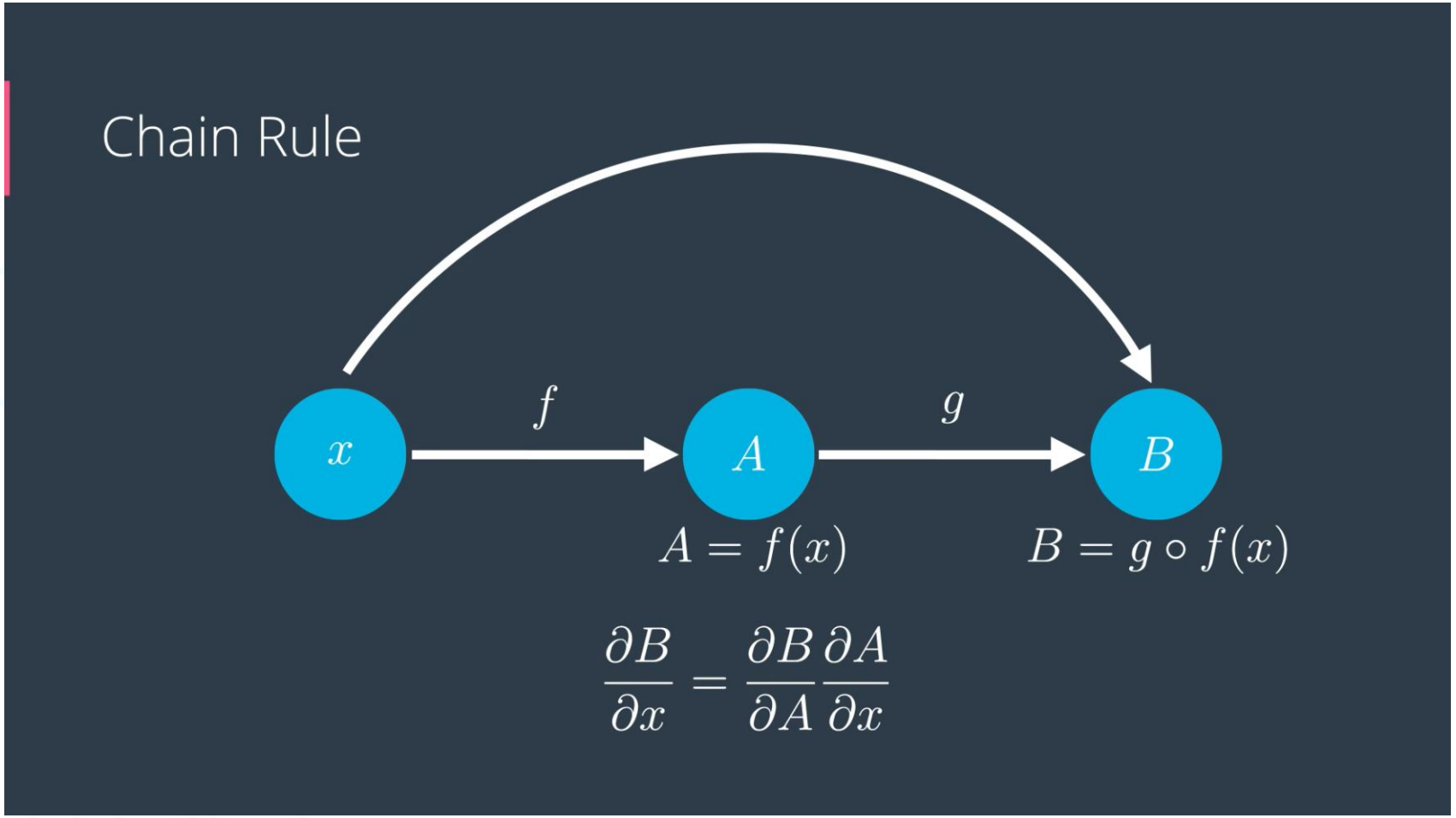
Redes Neurais – Backpropagation



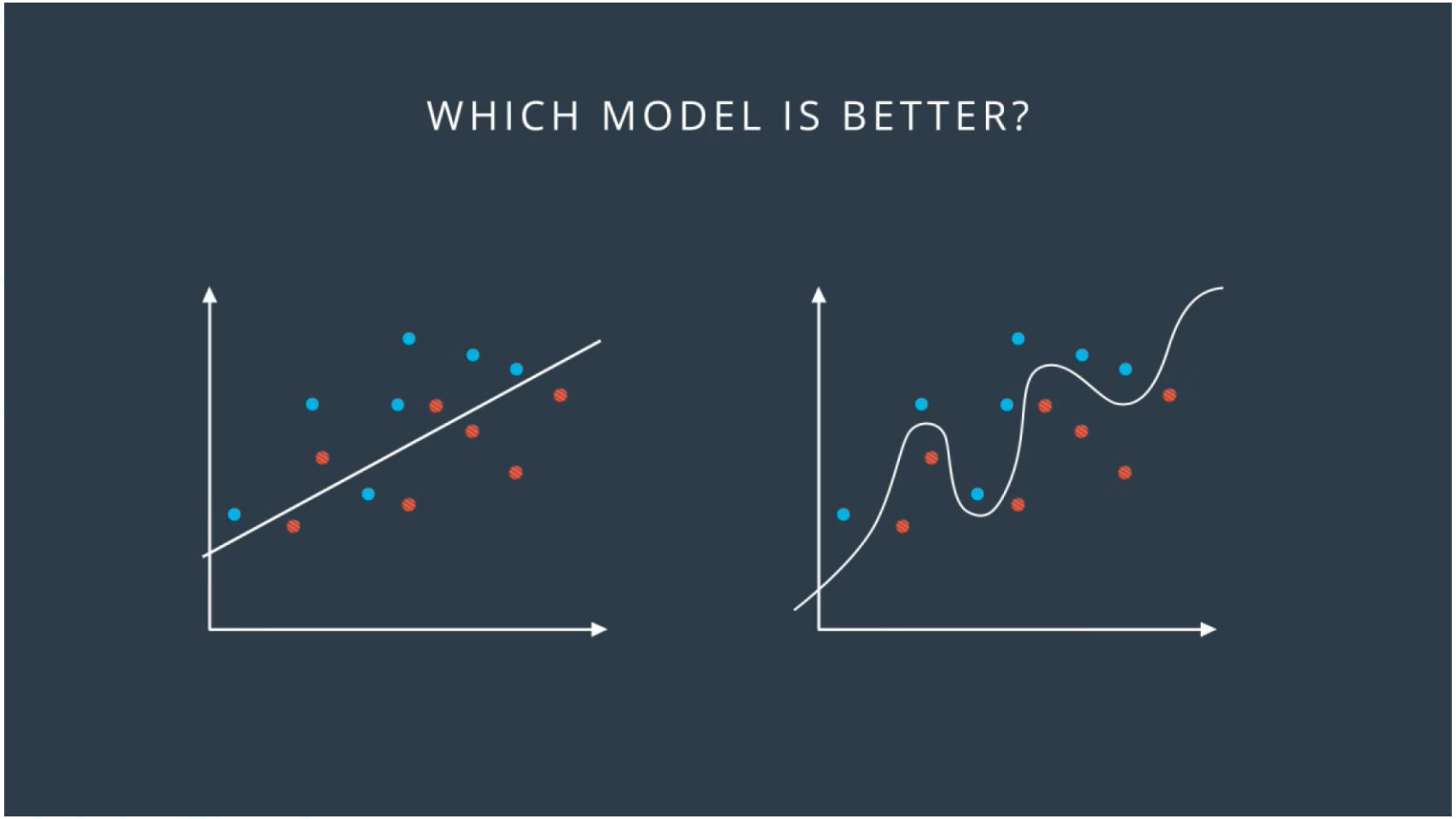
Redes Neurais – Backpropagation



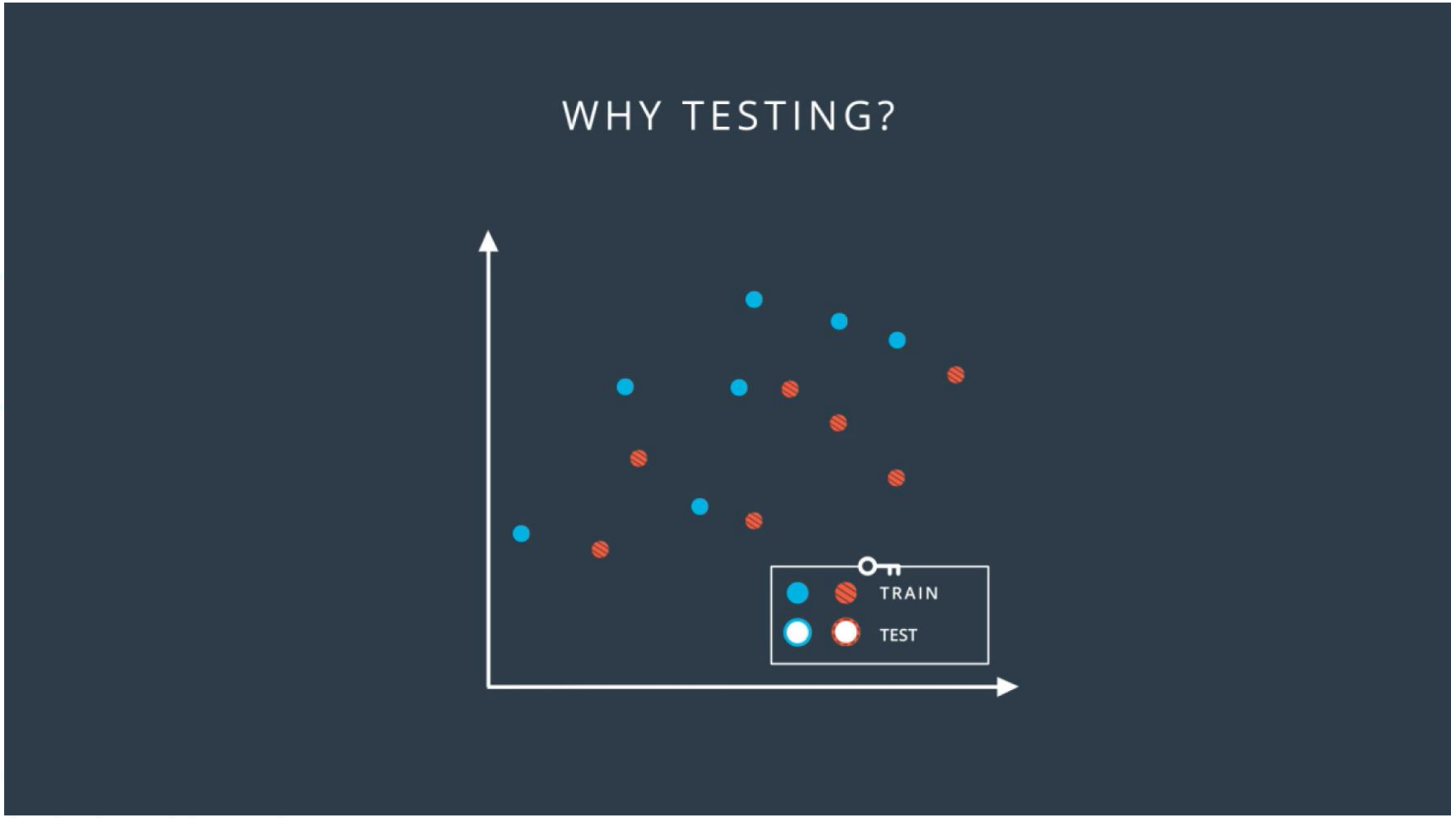
Redes Neurais – Backpropagation



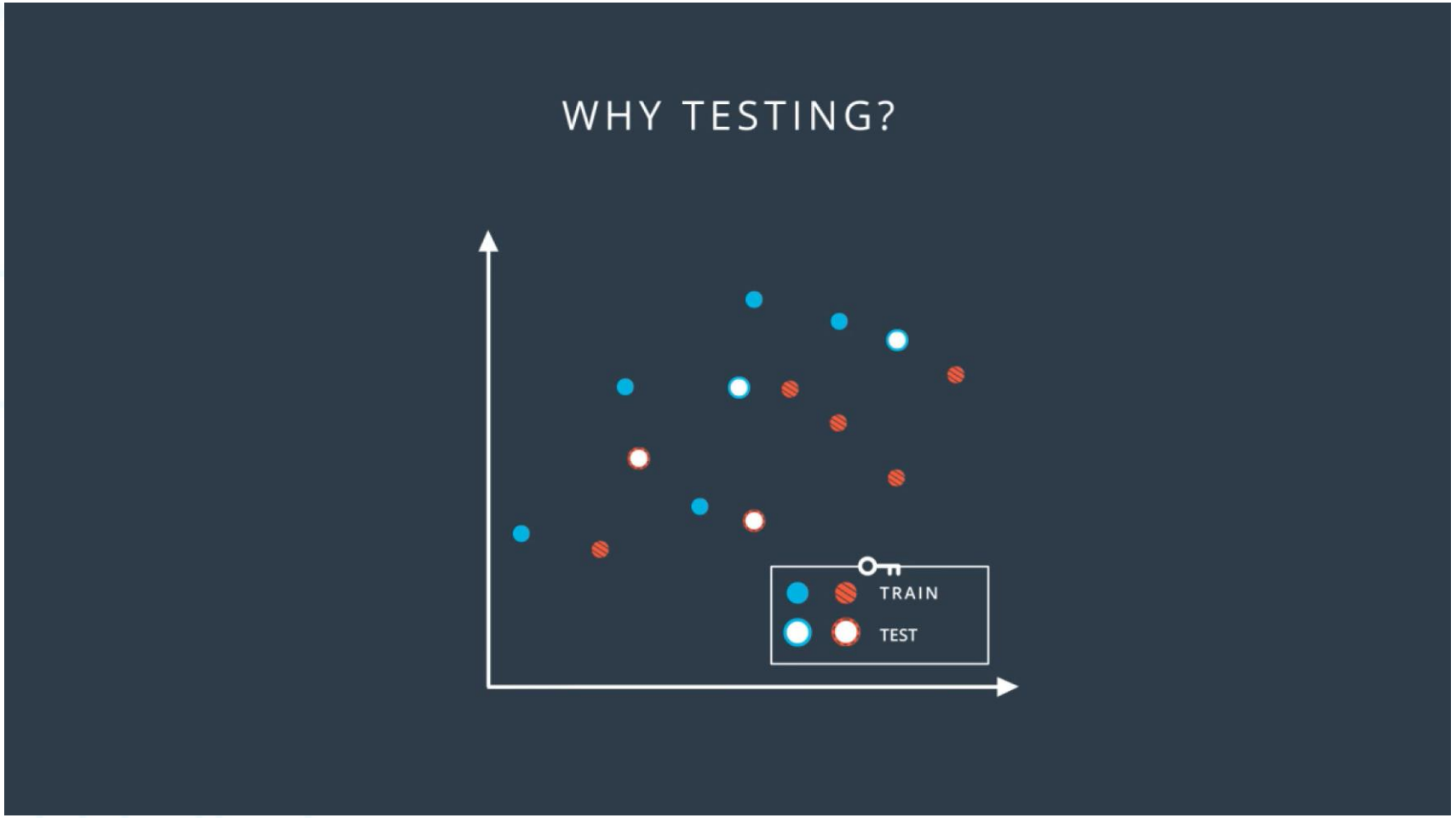
Redes Neurais – Testing



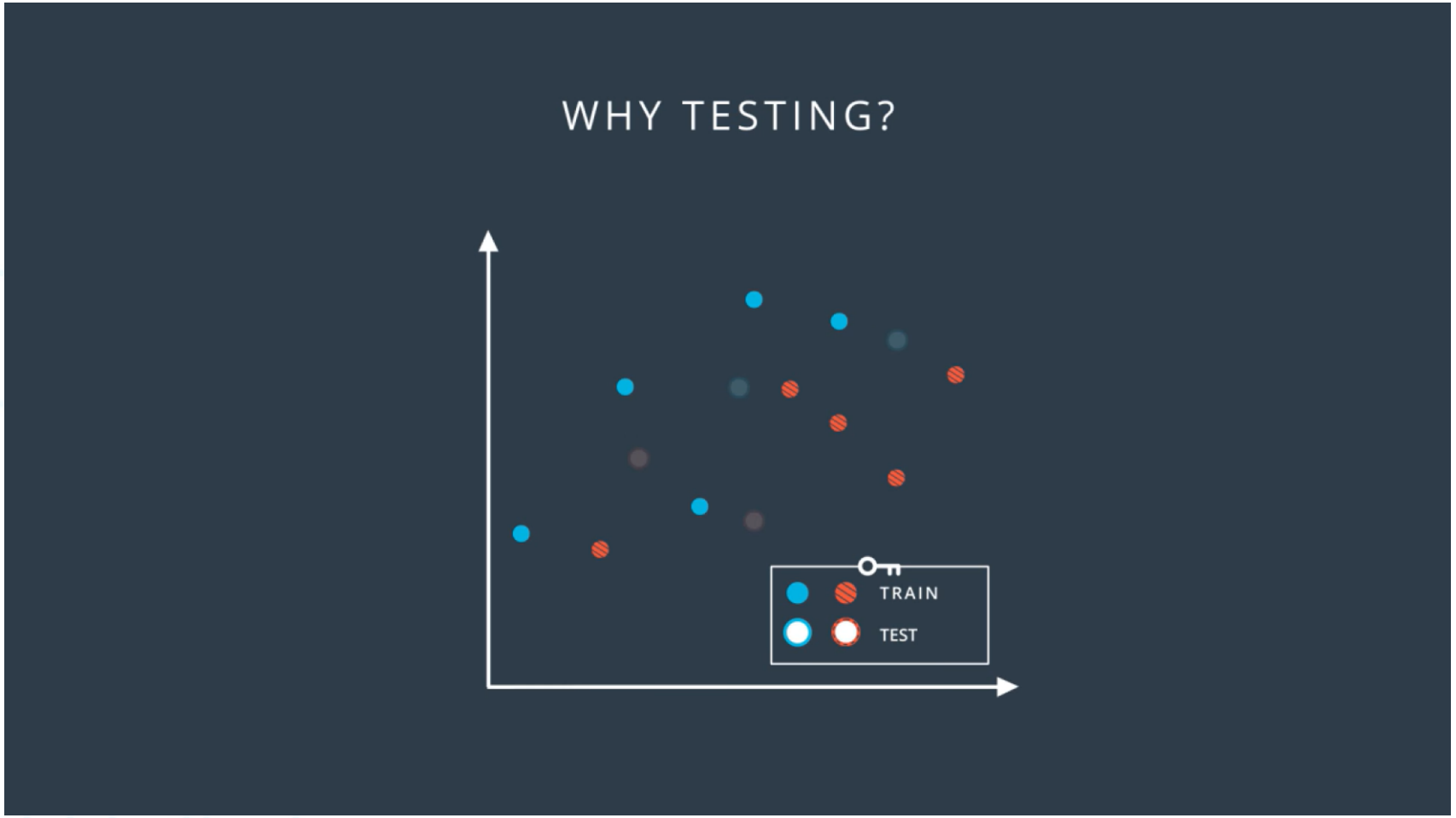
Redes Neurais – Testing



Redes Neurais – Testing

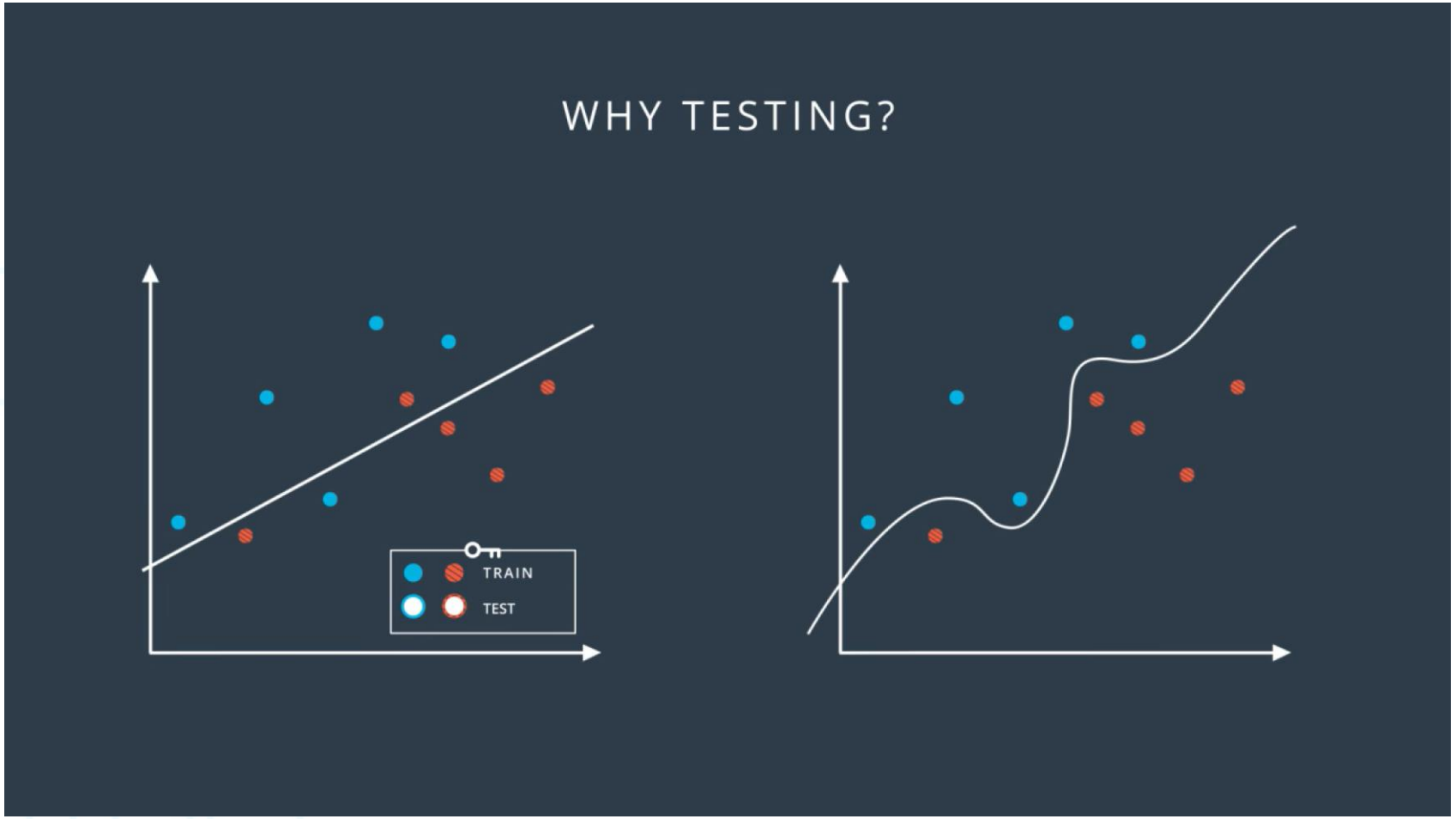


Redes Neurais – Testing

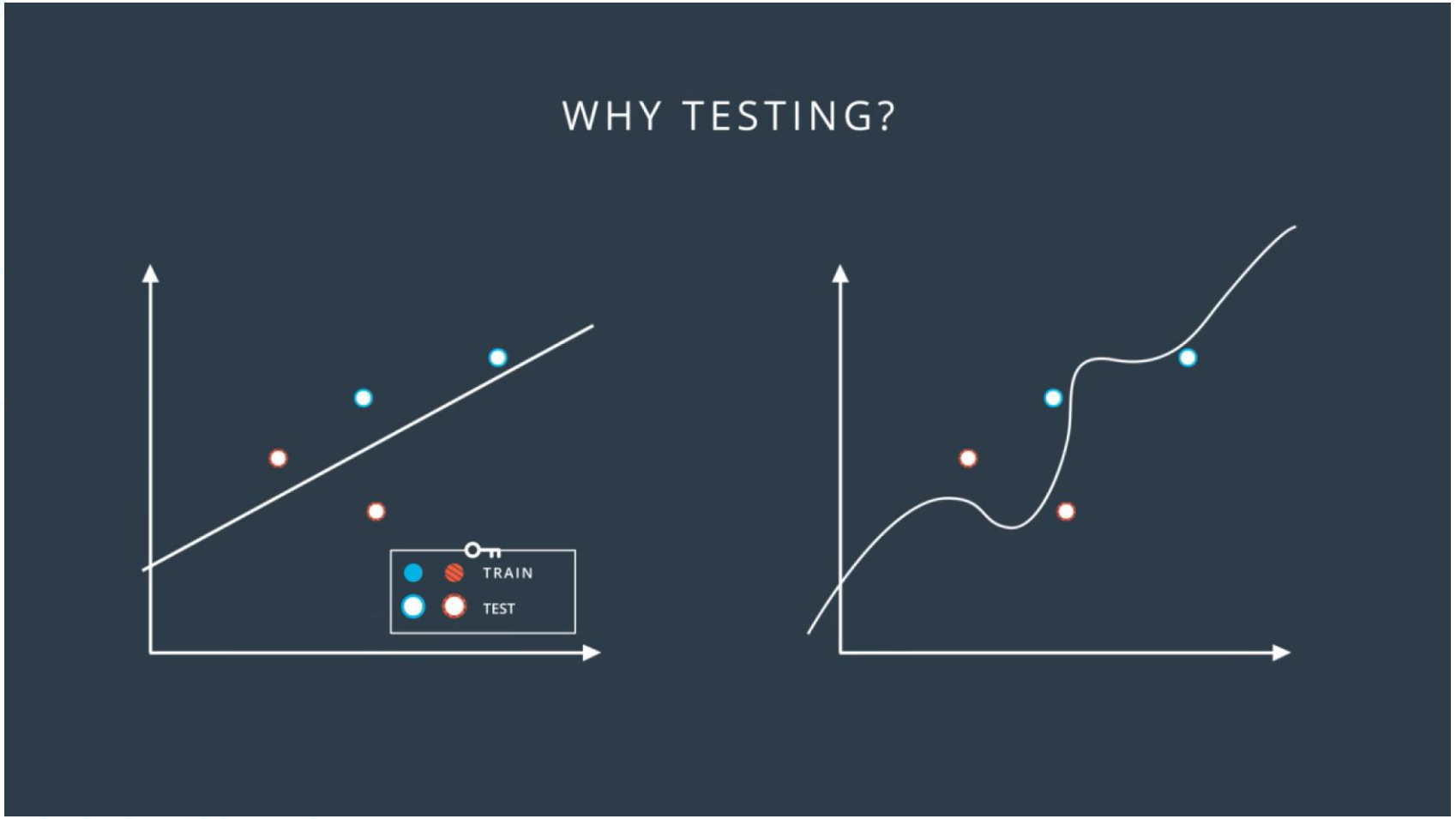


Redes Neurais – Testing

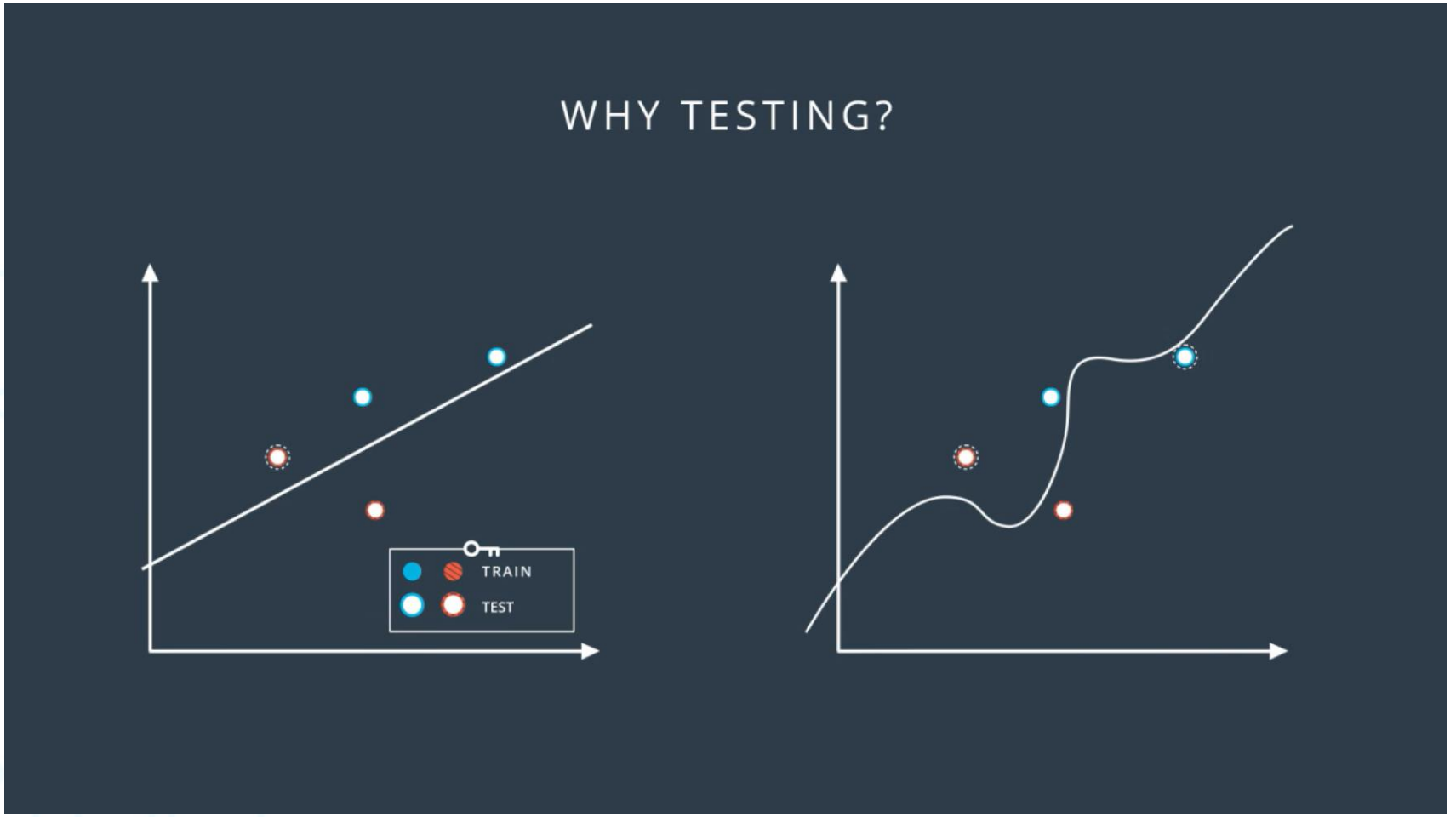
Redes Neurais – Testing



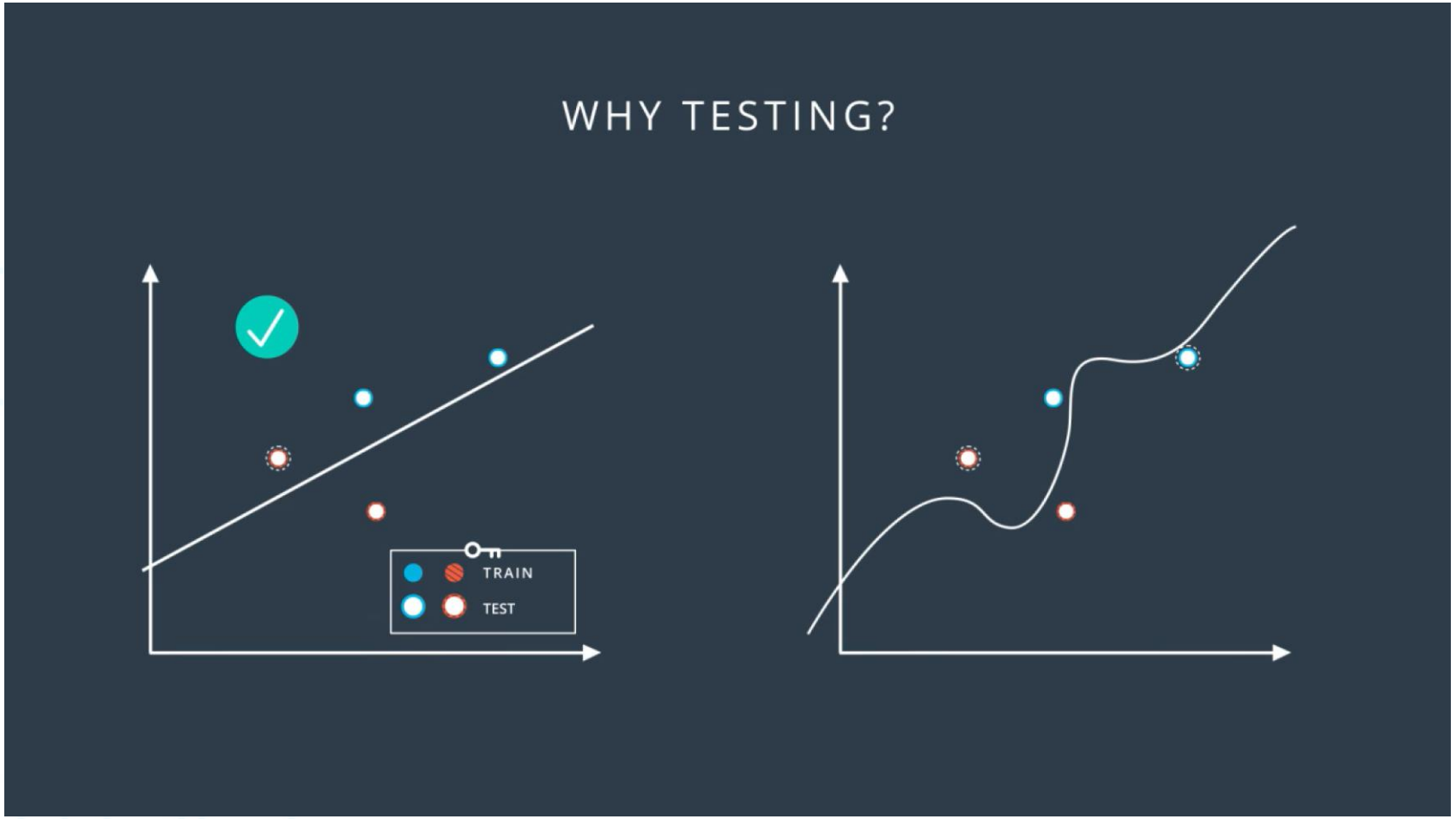
Redes Neurais – Testing



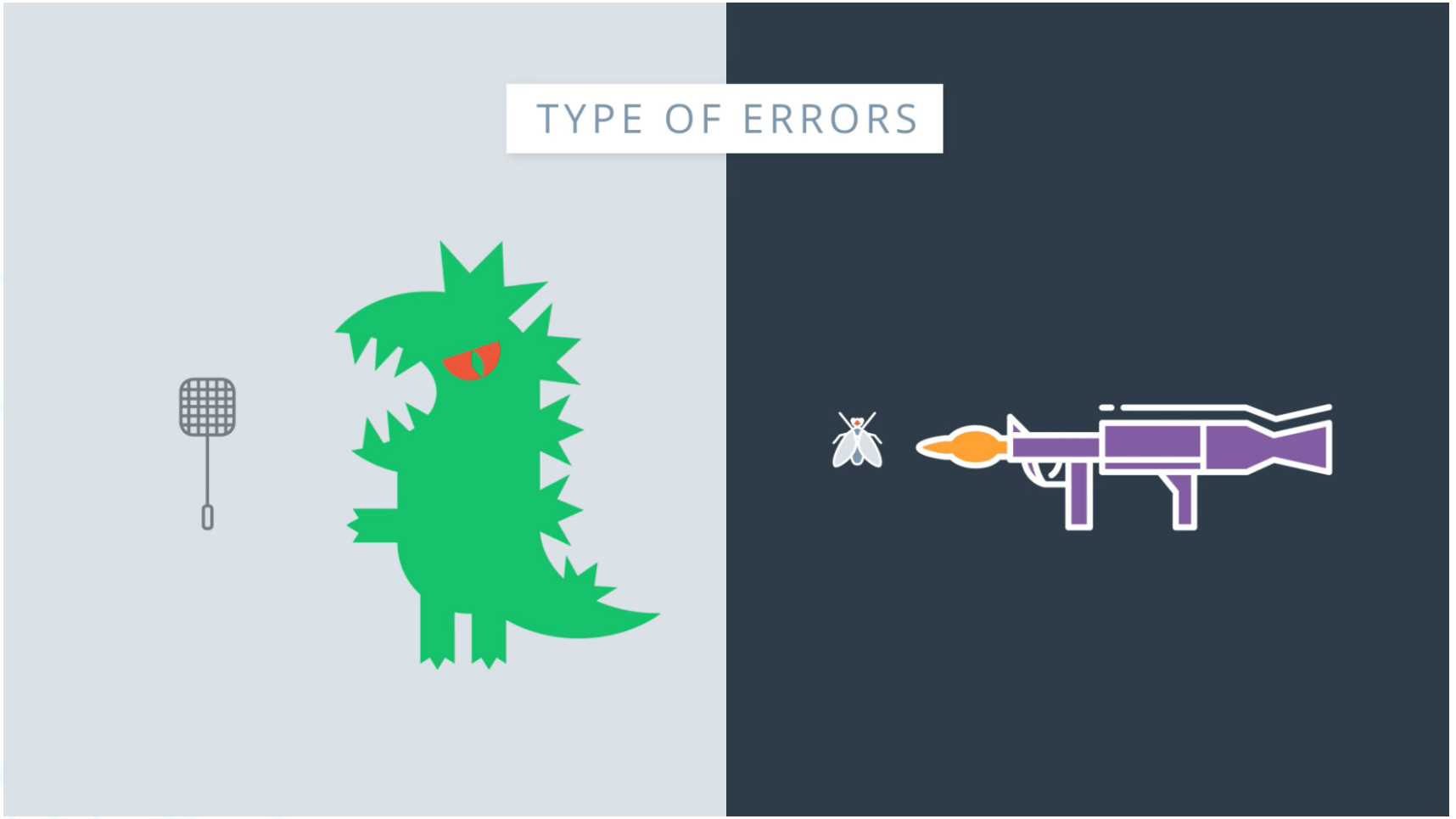
Redes Neurais – Testing



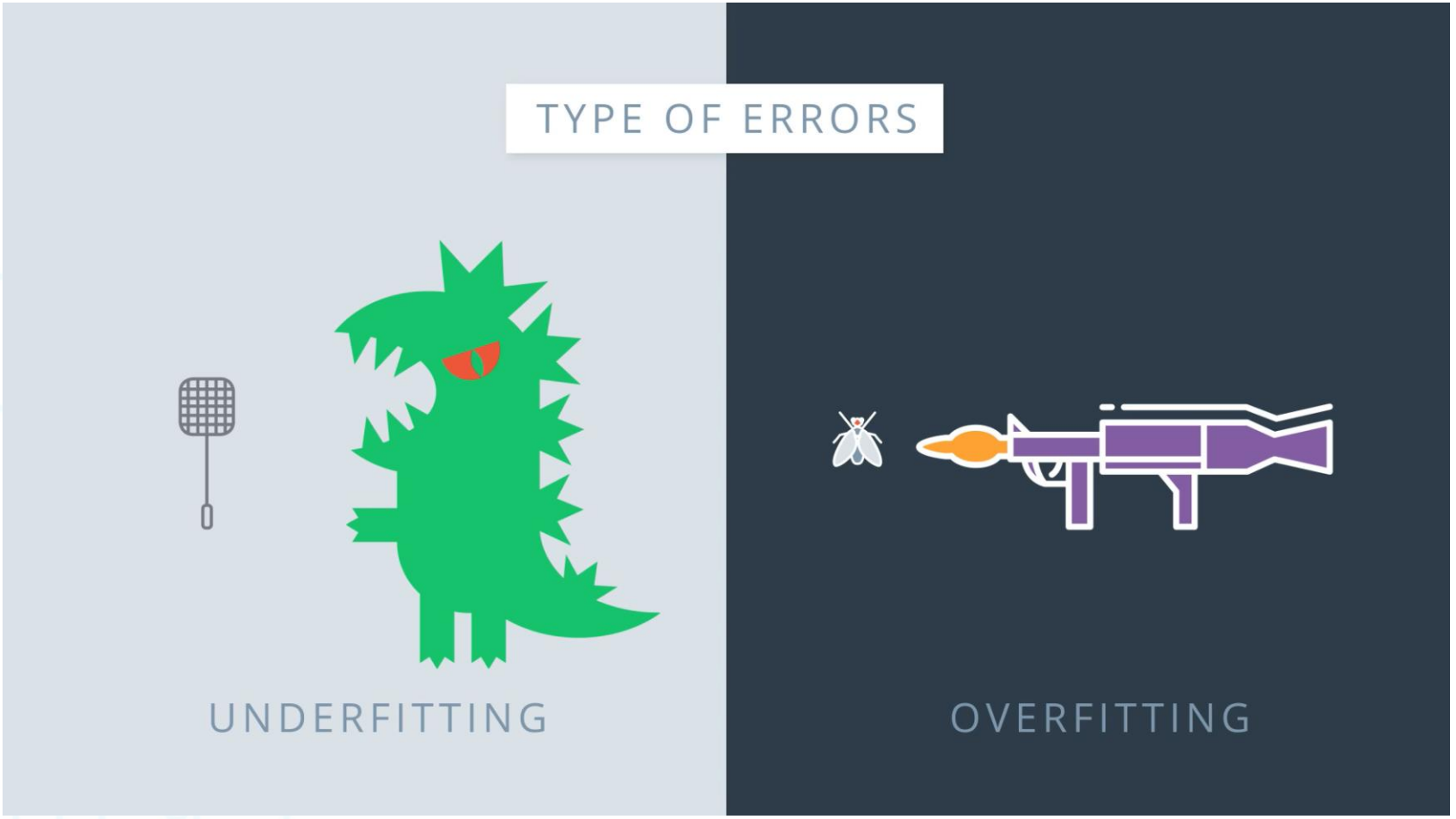
Redes Neurais – Testing



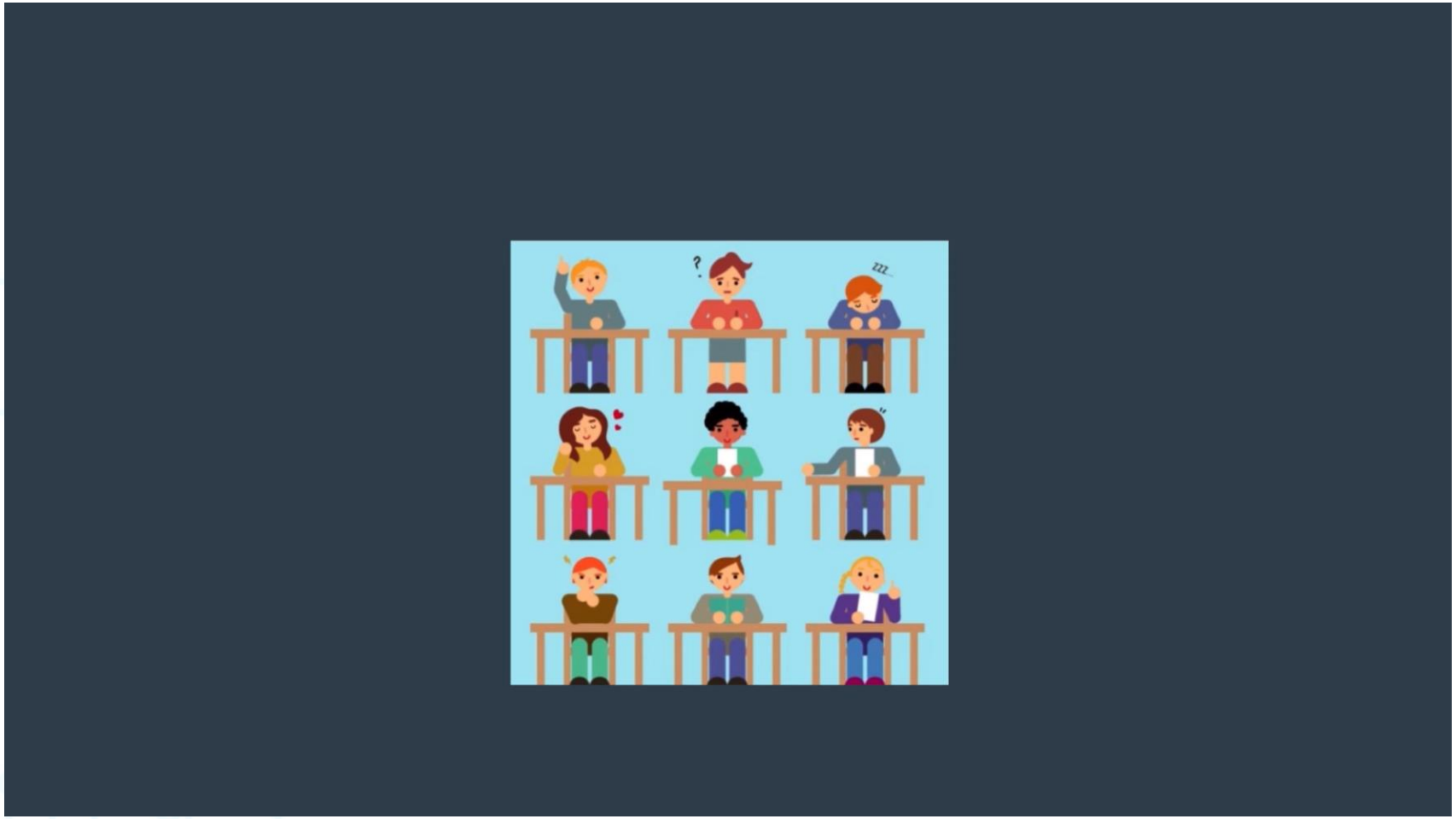
Redes Neurais – Overfitting and Underfitting



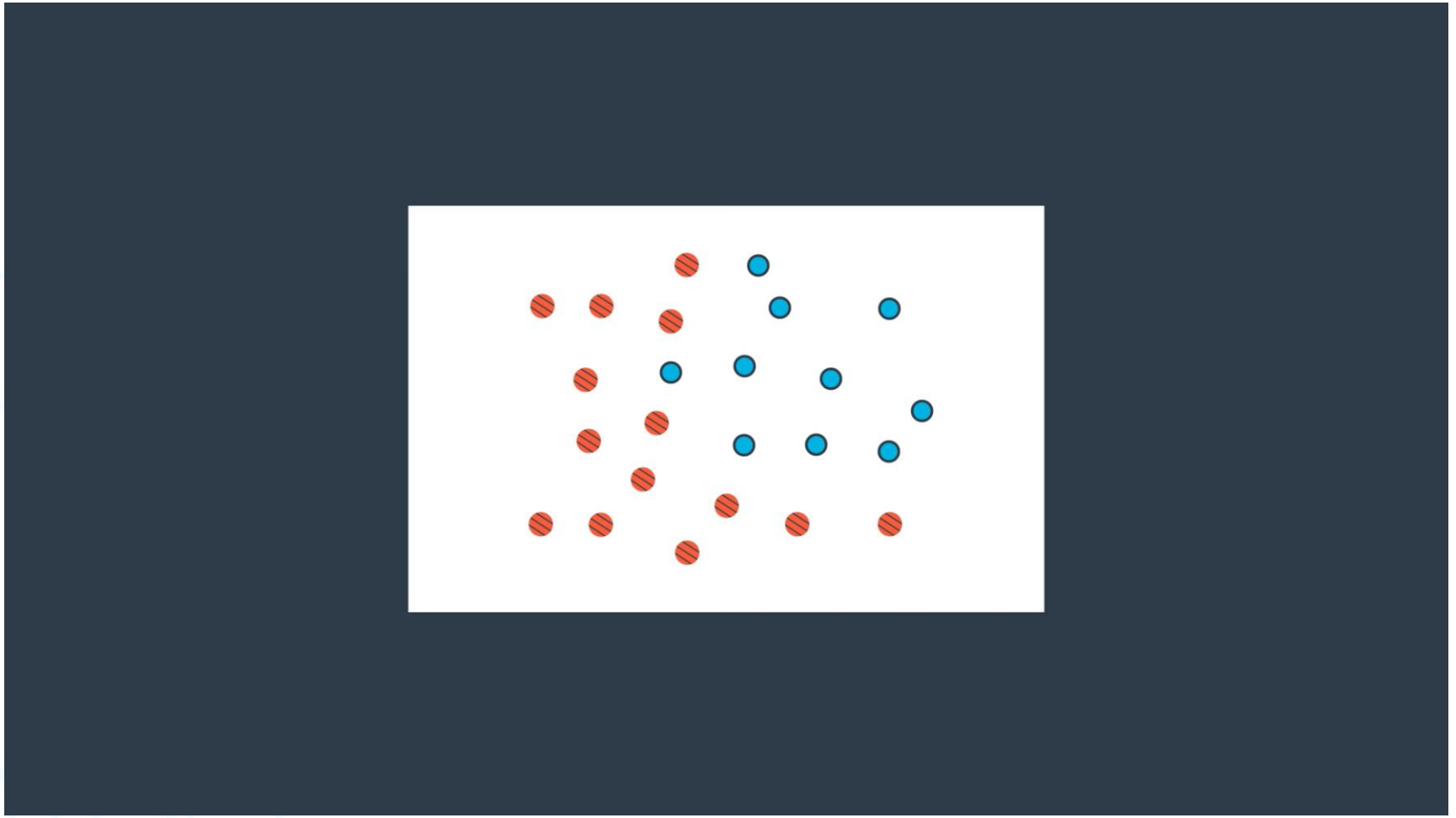
Redes Neurais – Overfitting and Underfitting



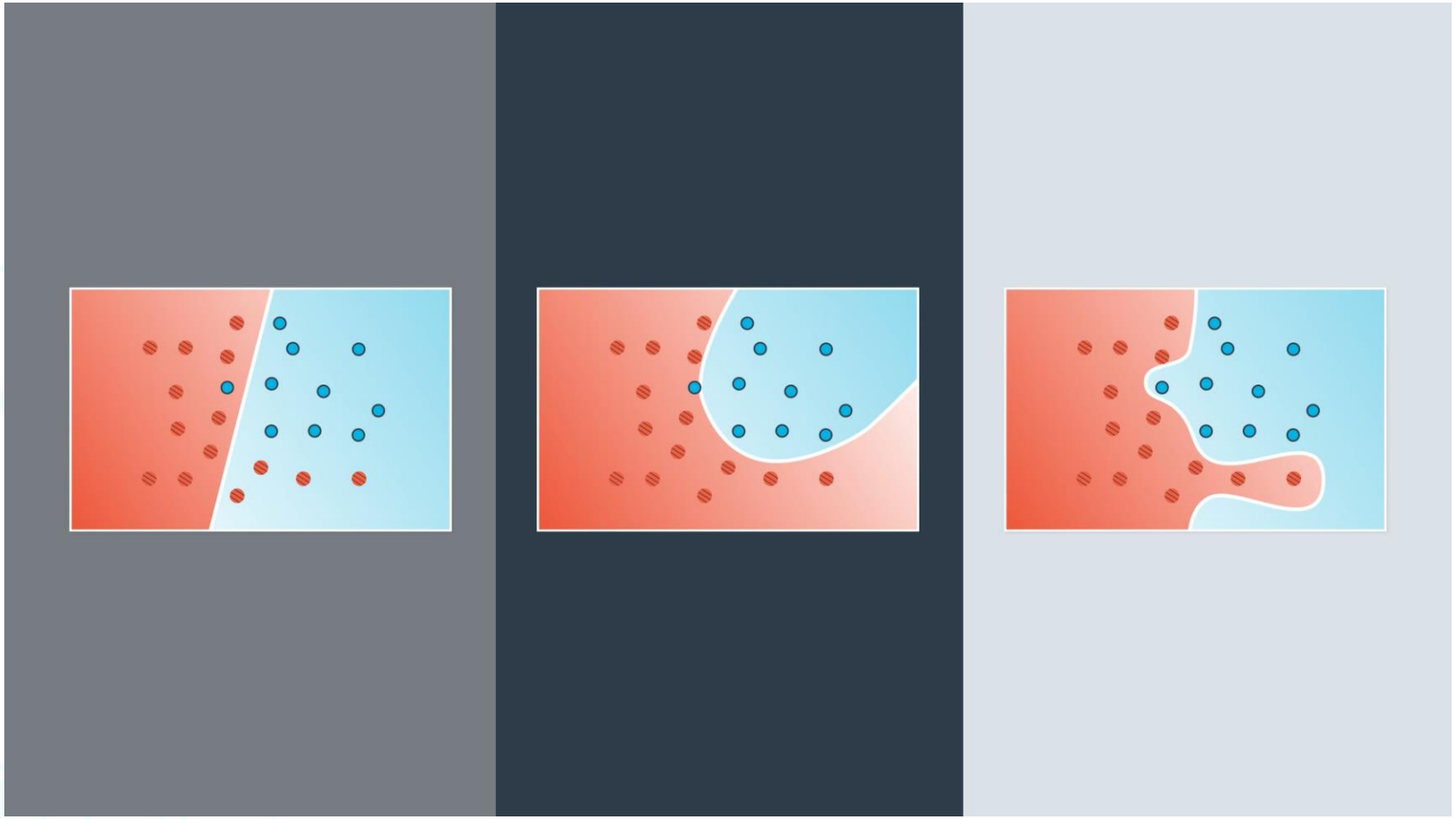
Redes Neurais – Overfitting and Underfitting



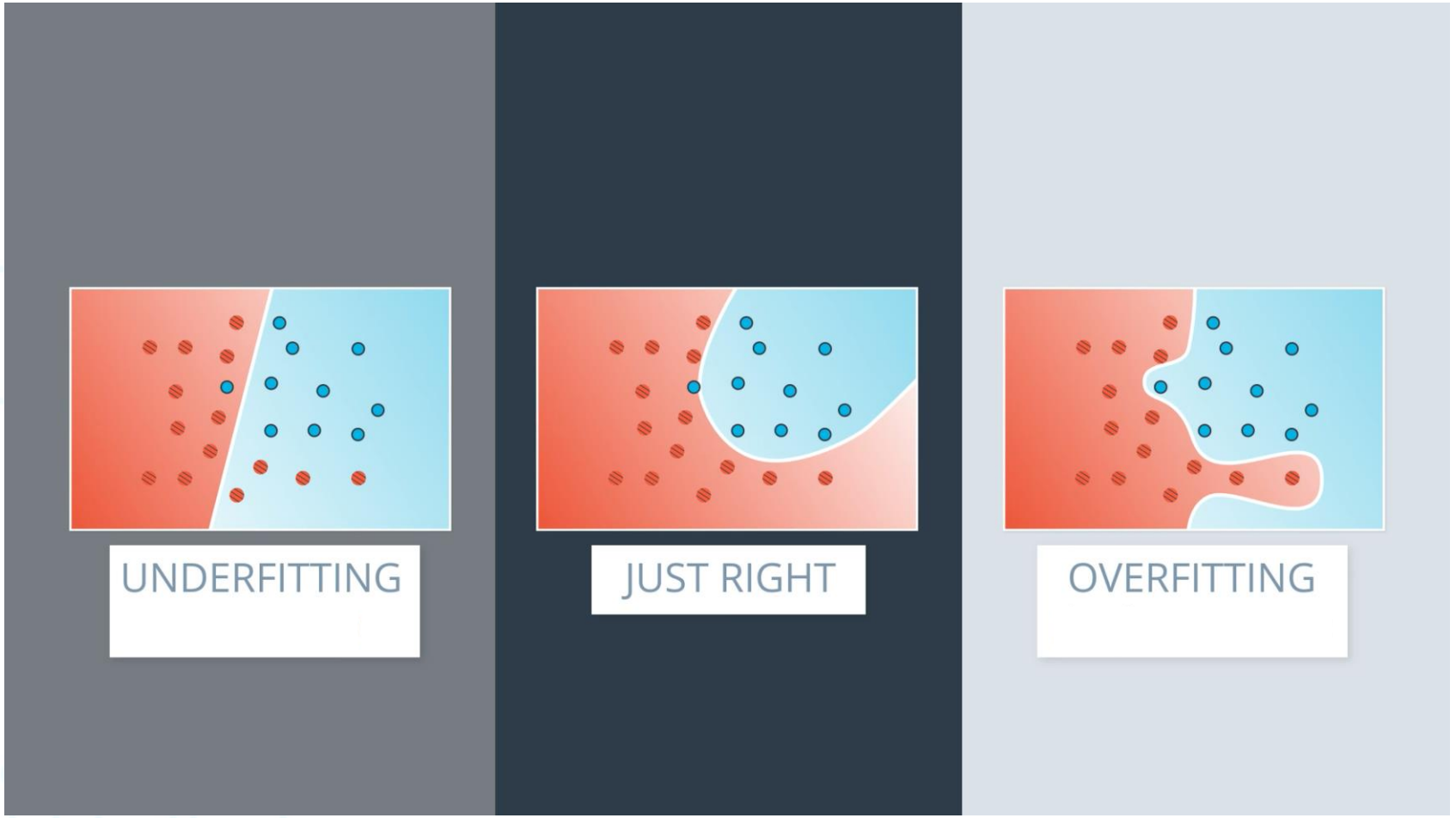
Redes Neurais – Overfitting and Underfitting



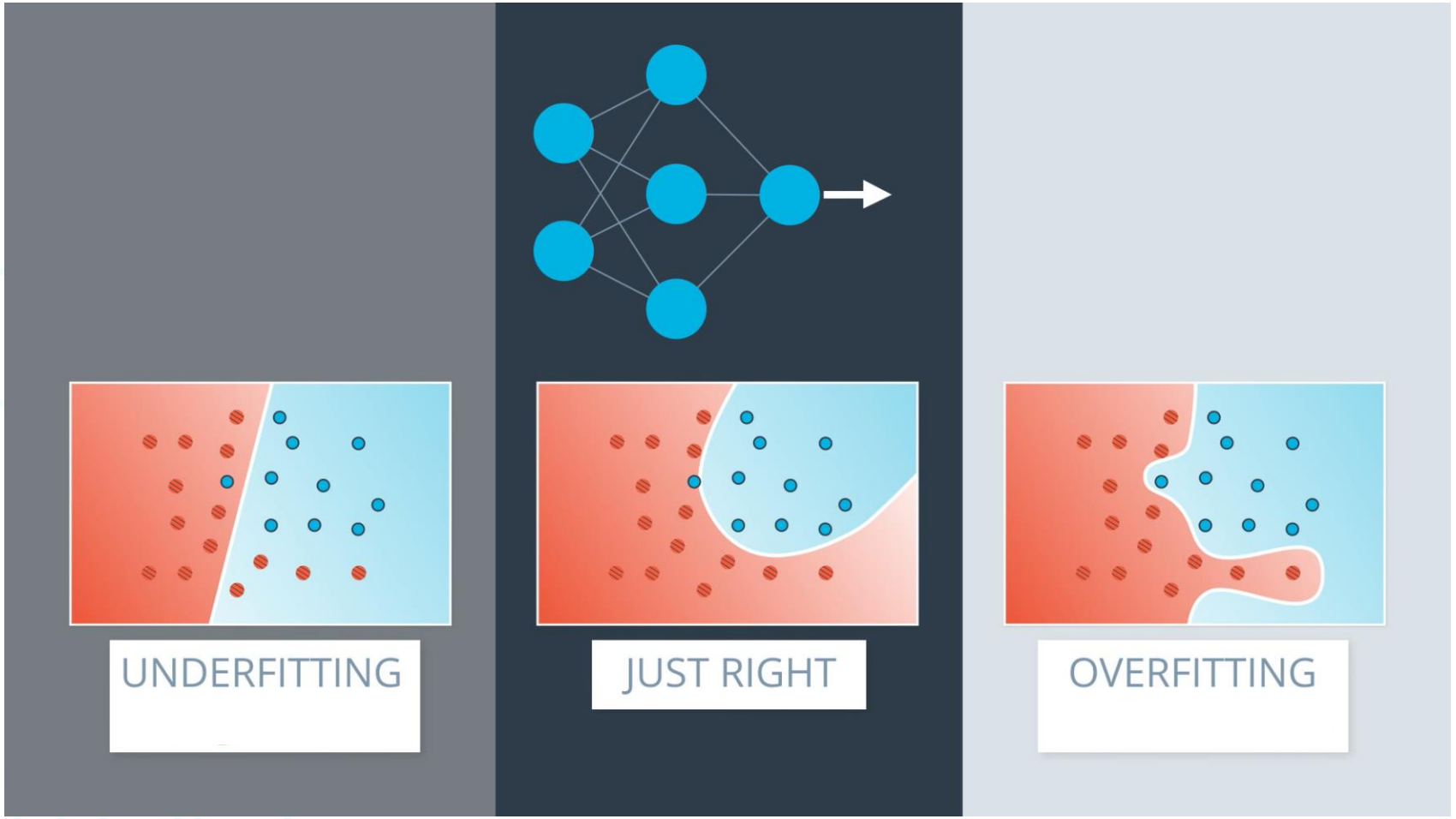
Redes Neurais – Overfitting and Underfitting



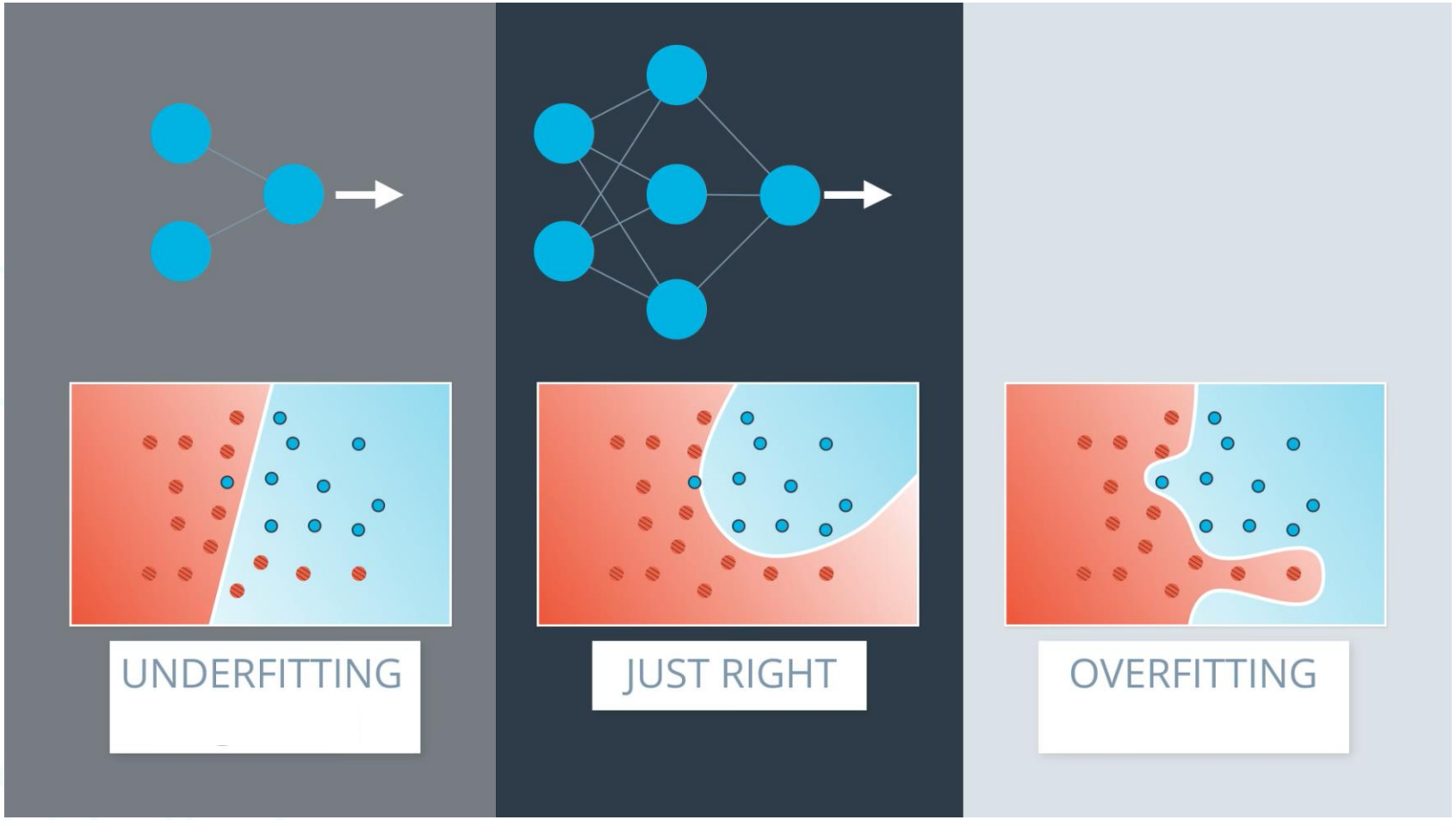
Redes Neurais – Overfitting and Underfitting



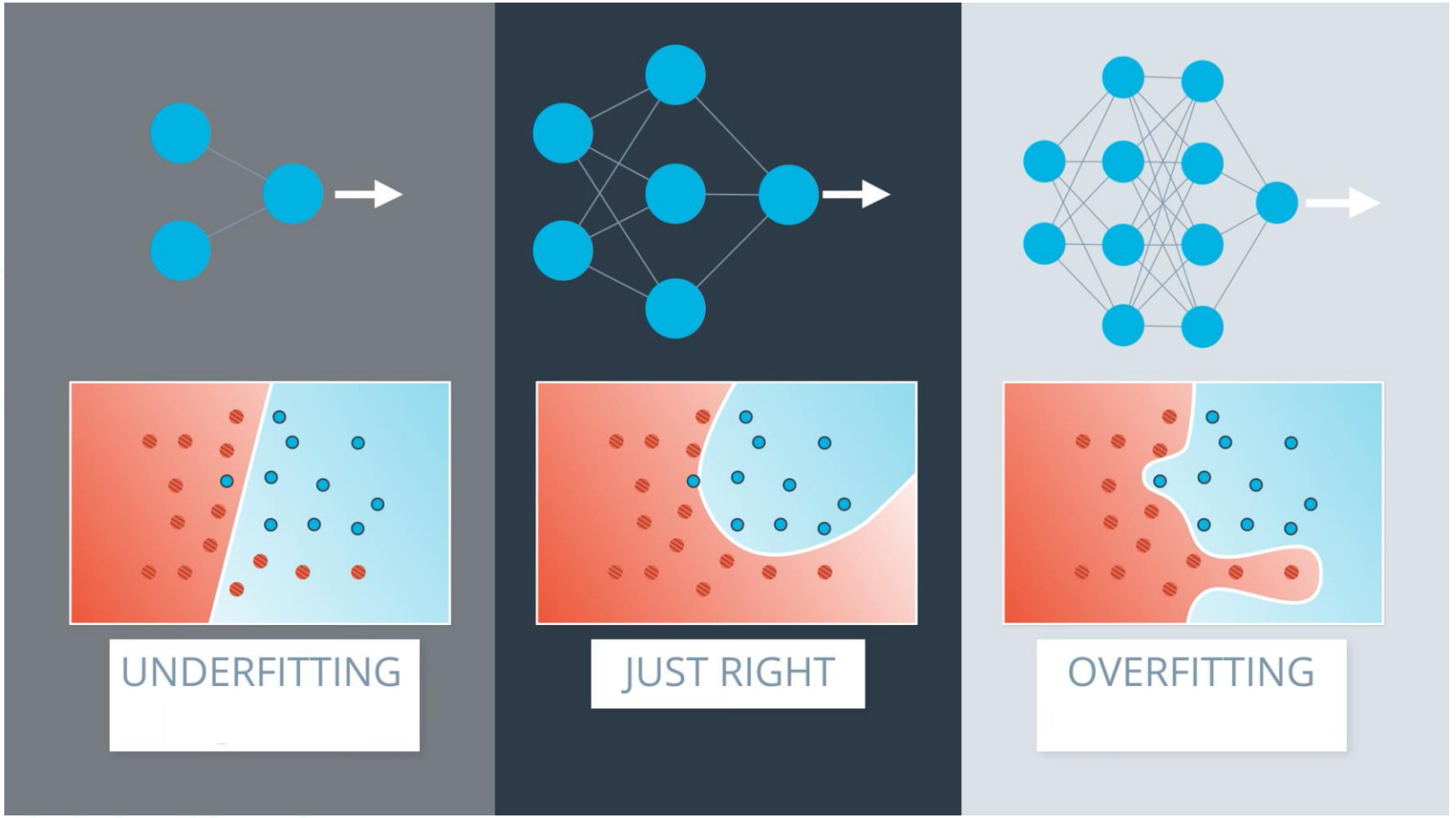
Redes Neurais – Overfitting and Underfitting



Redes Neurais – Overfitting and Underfitting

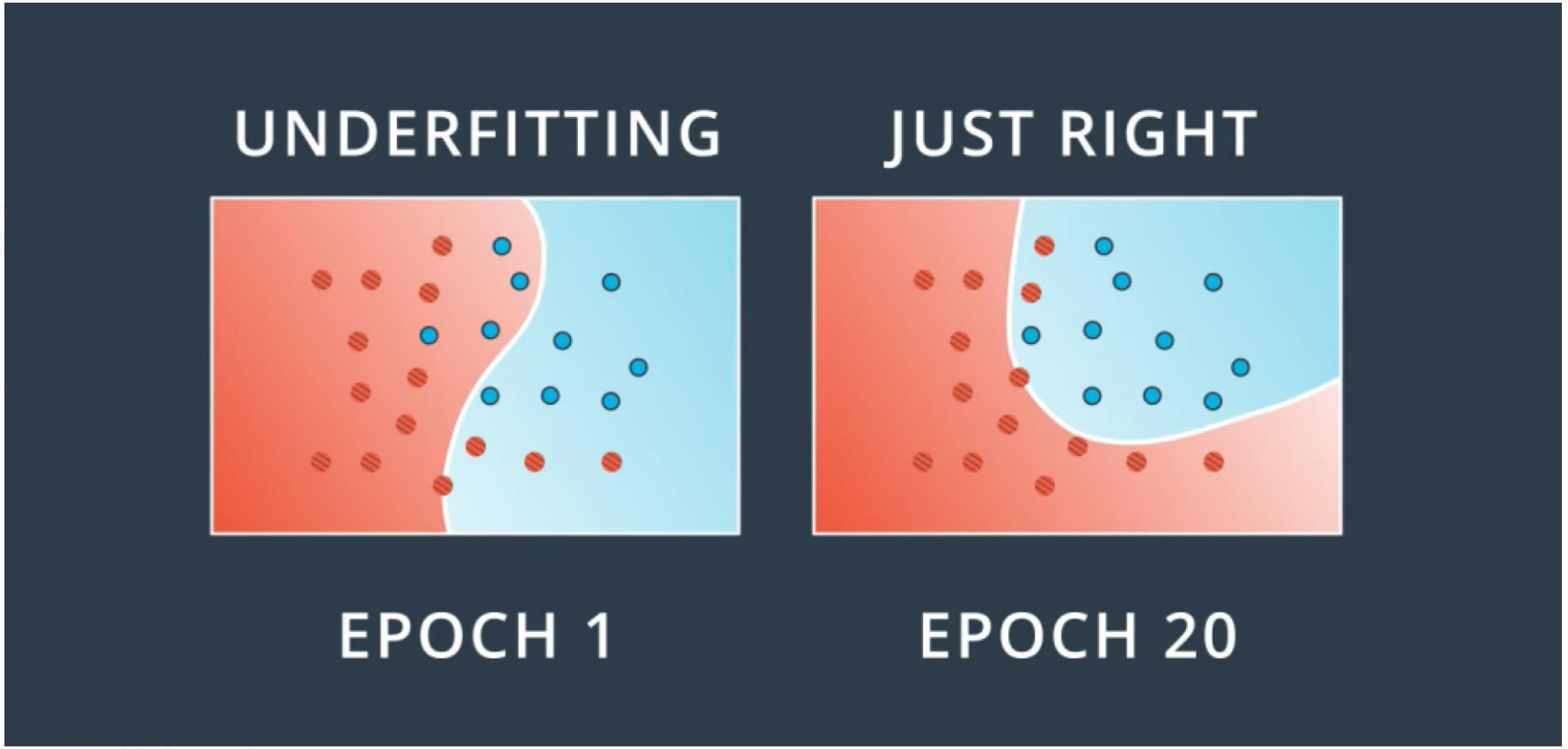


Redes Neurais – Overfitting and Underfitting

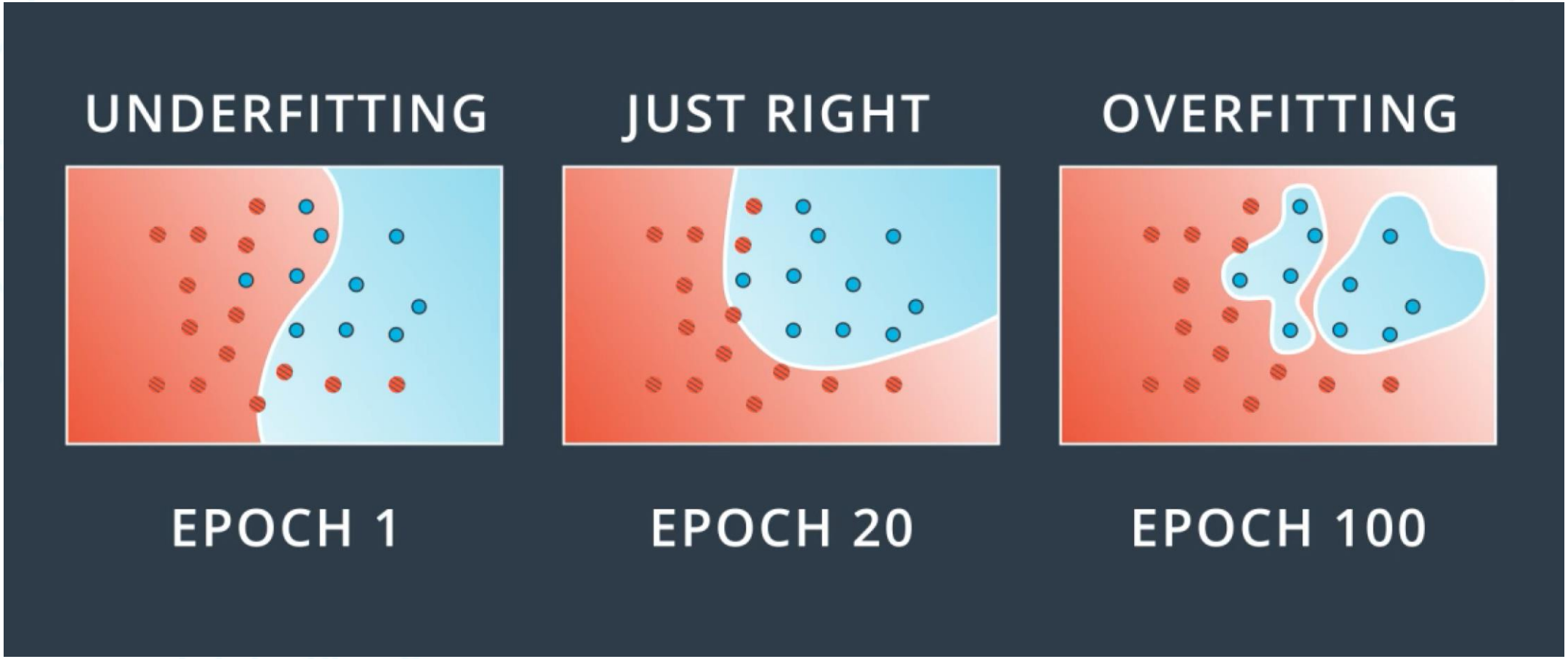


Redes Neurais – Model Complexity Graph

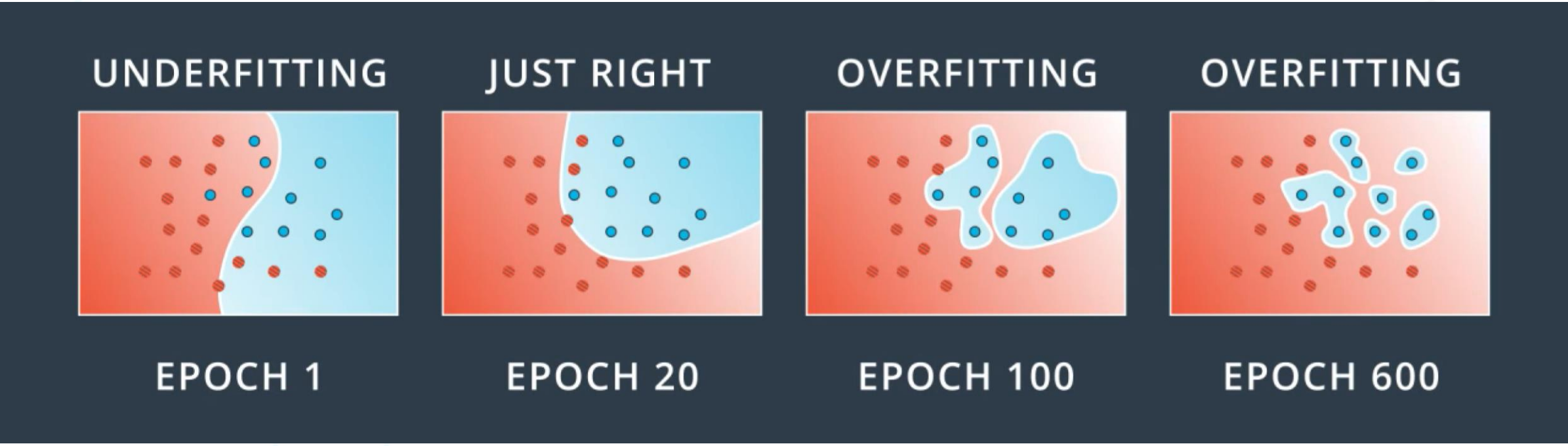
Redes Neurais – Model Complexity Graph



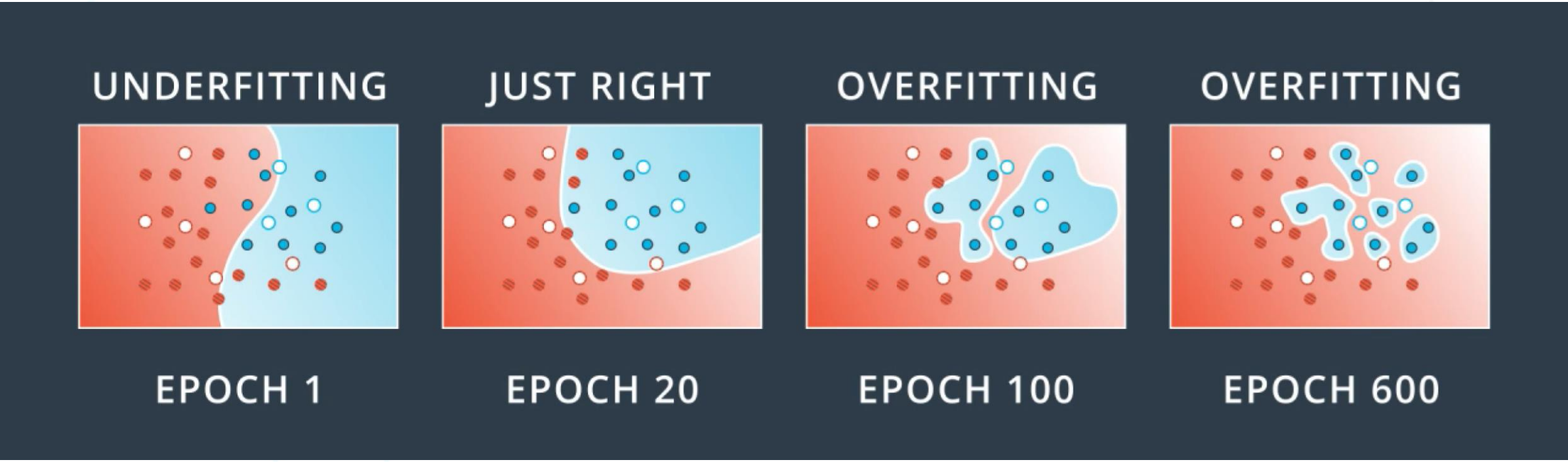
Redes Neurais – Model Complexity Graph



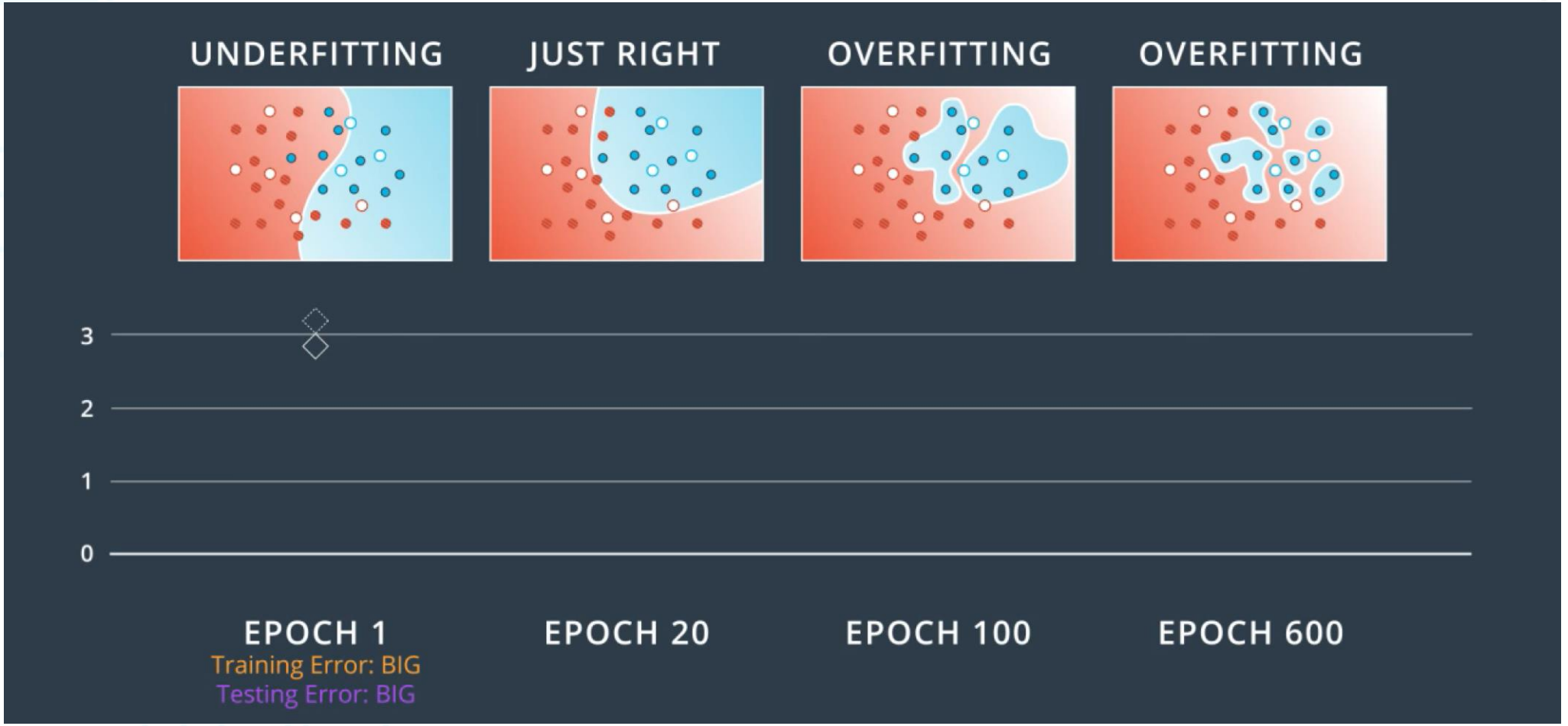
Redes Neurais – Model Complexity Graph



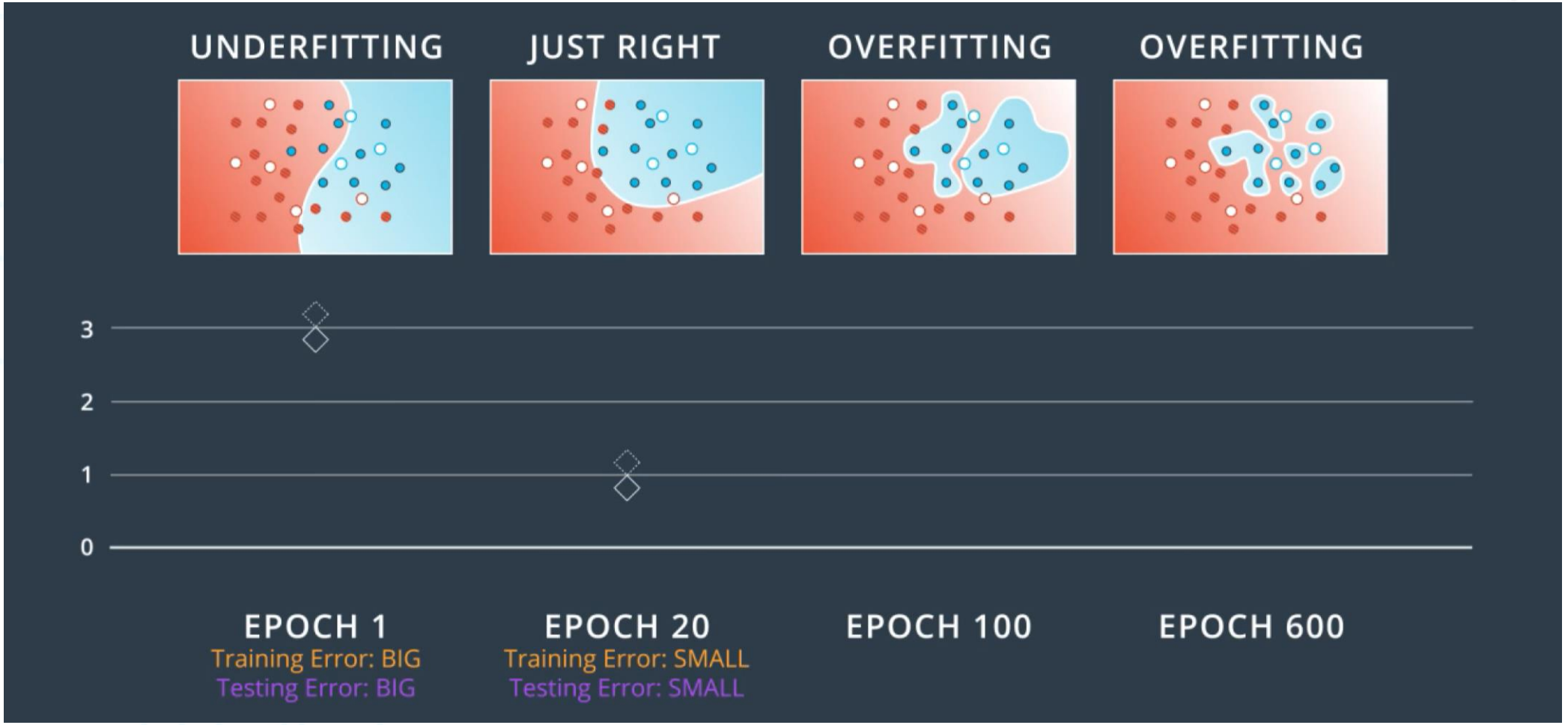
Redes Neurais – Model Complexity Graph



Redes Neurais – Model Complexity Graph

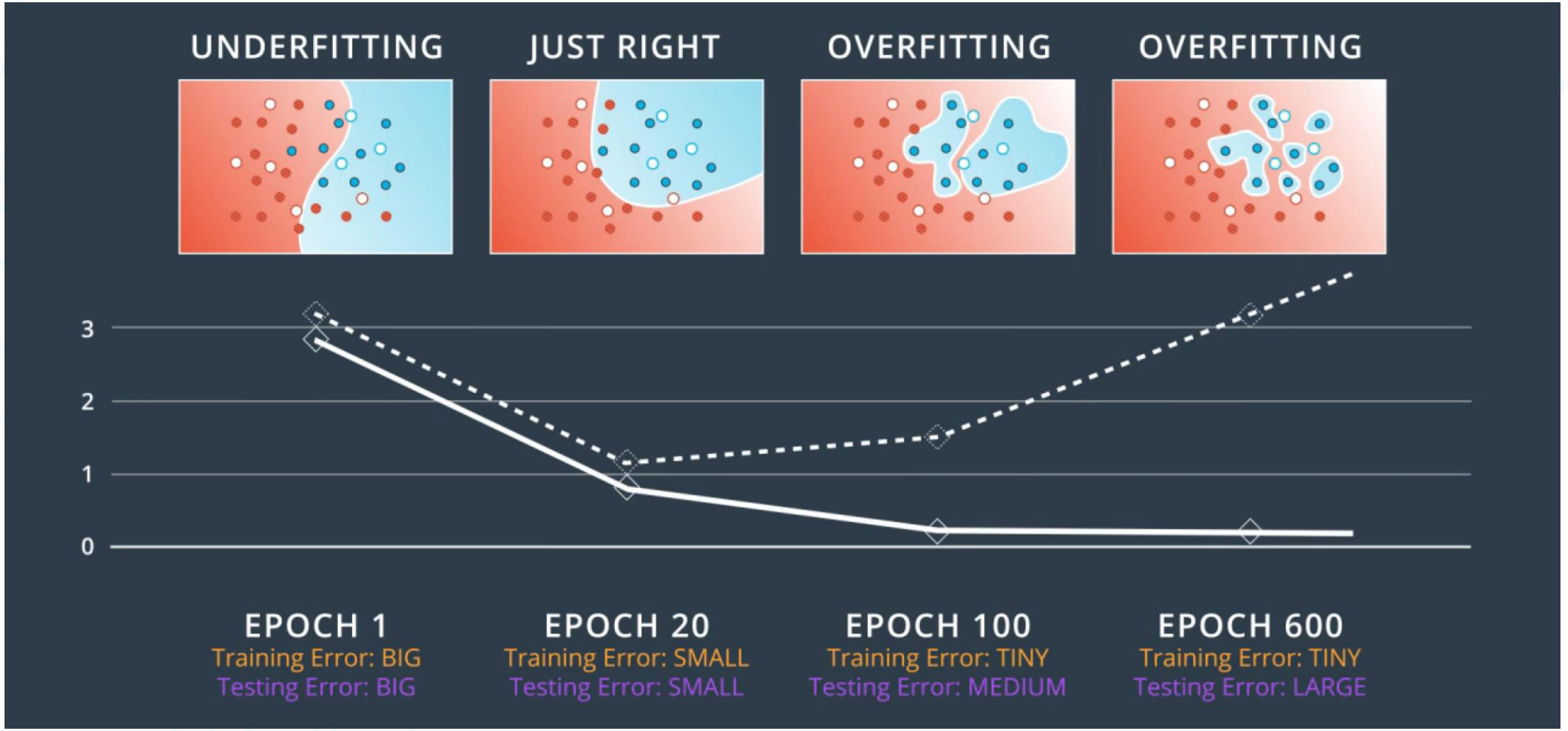


Redes Neurais – Model Complexity Graph

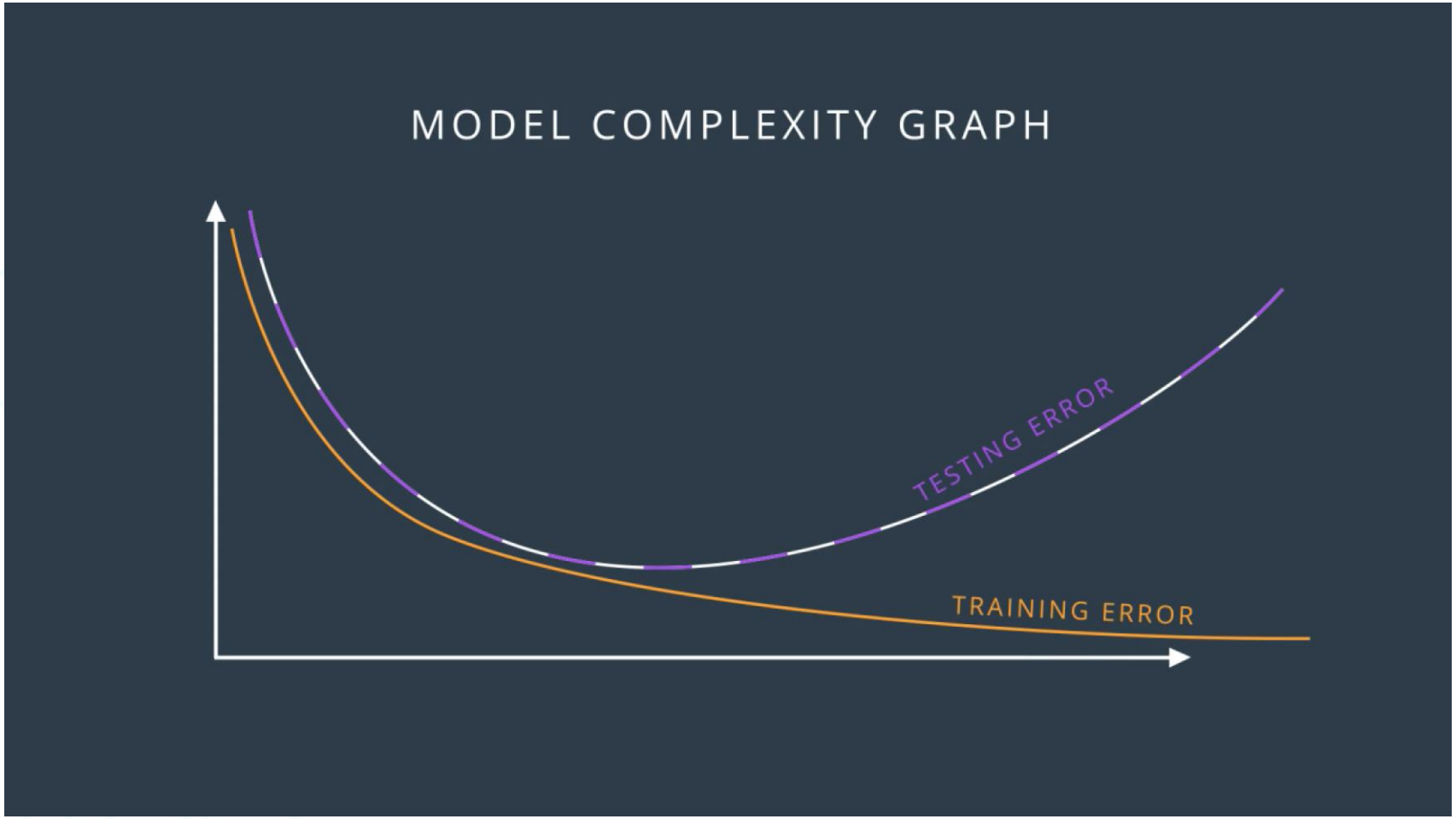


Redes Neurais – Model Complexity Graph

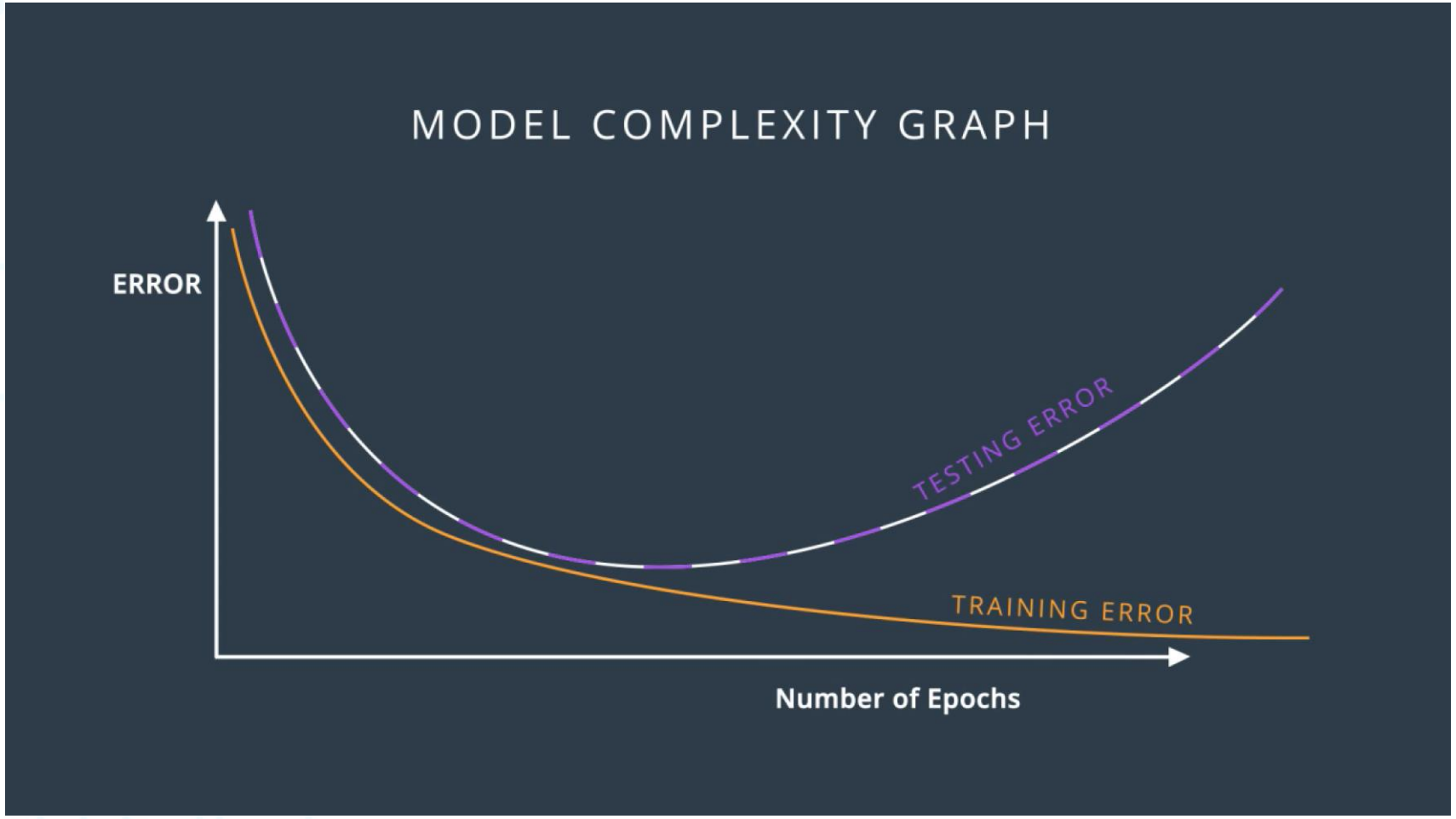
Redes Neurais – Model Complexity Graph



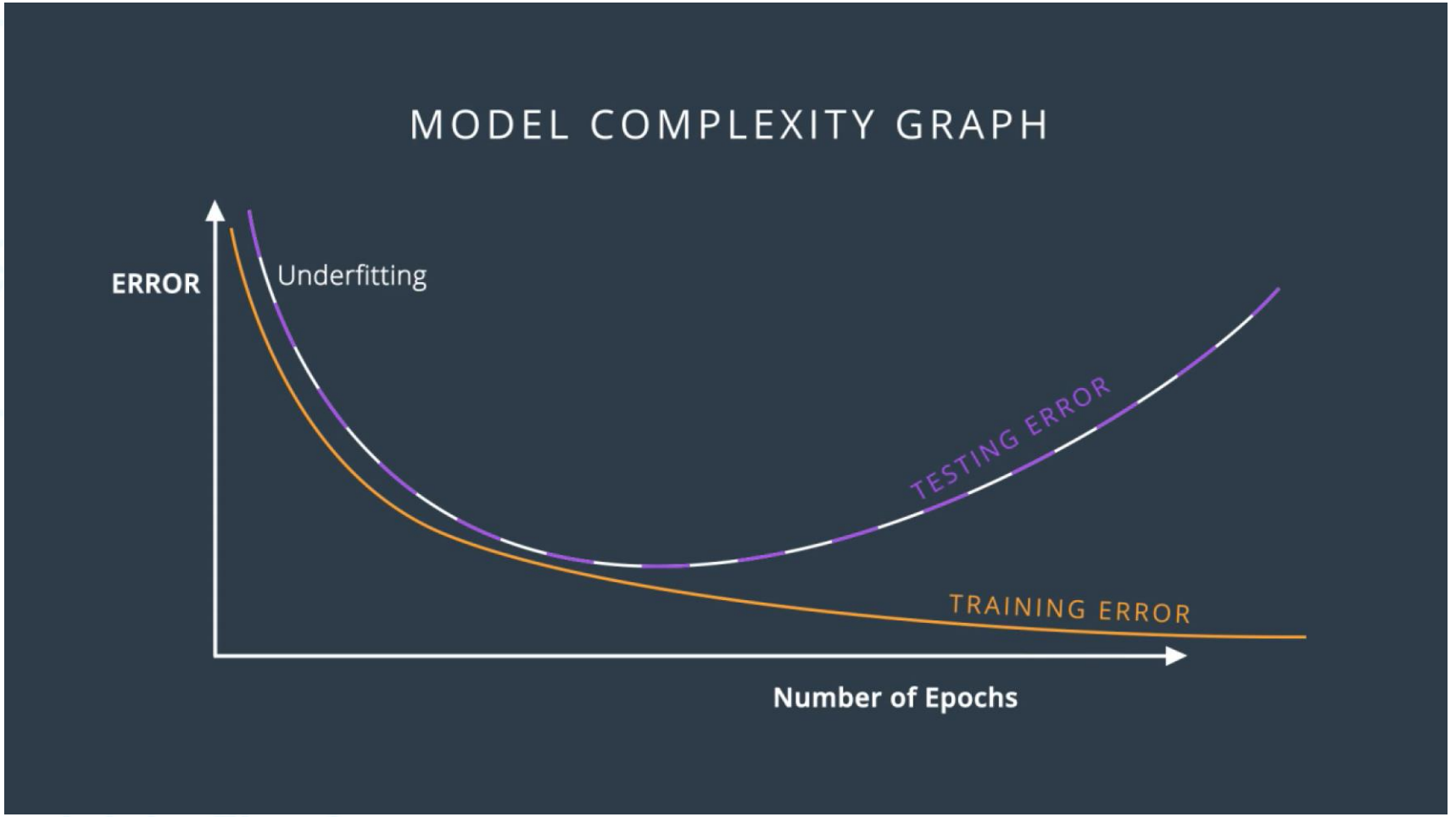
Redes Neurais – Model Complexity Graph



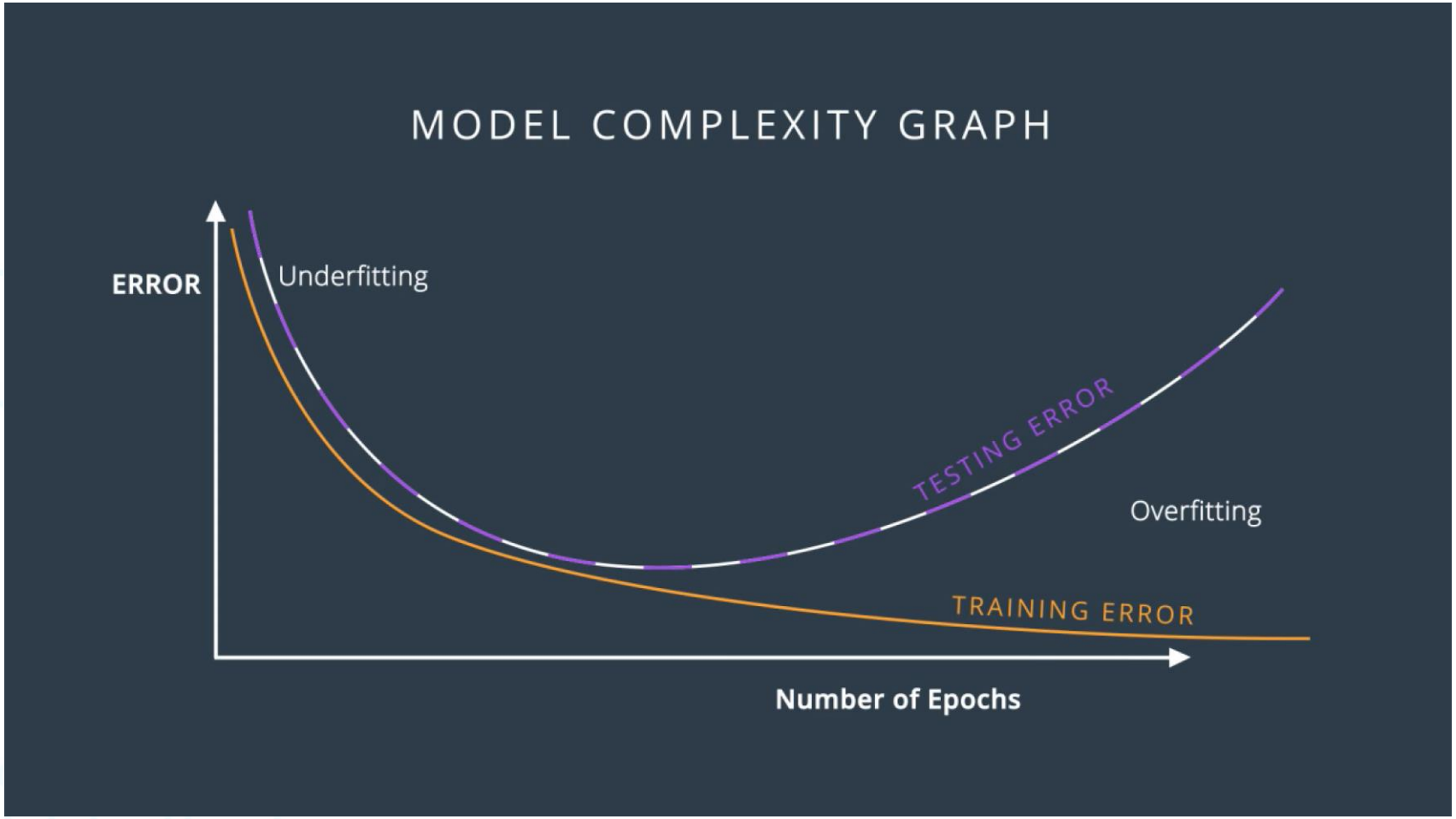
Redes Neurais – Model Complexity Graph



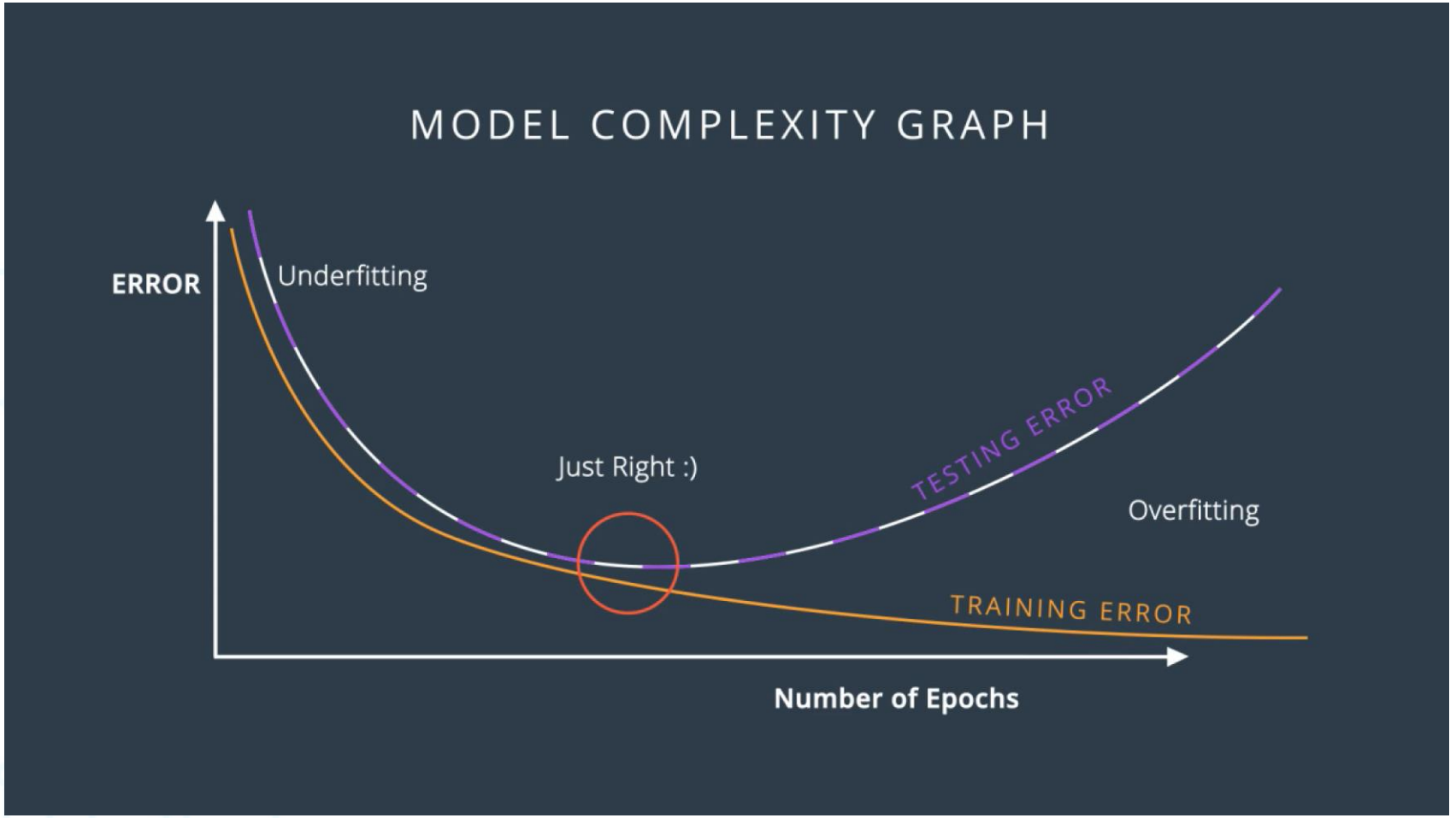
Redes Neurais – Model Complexity Graph



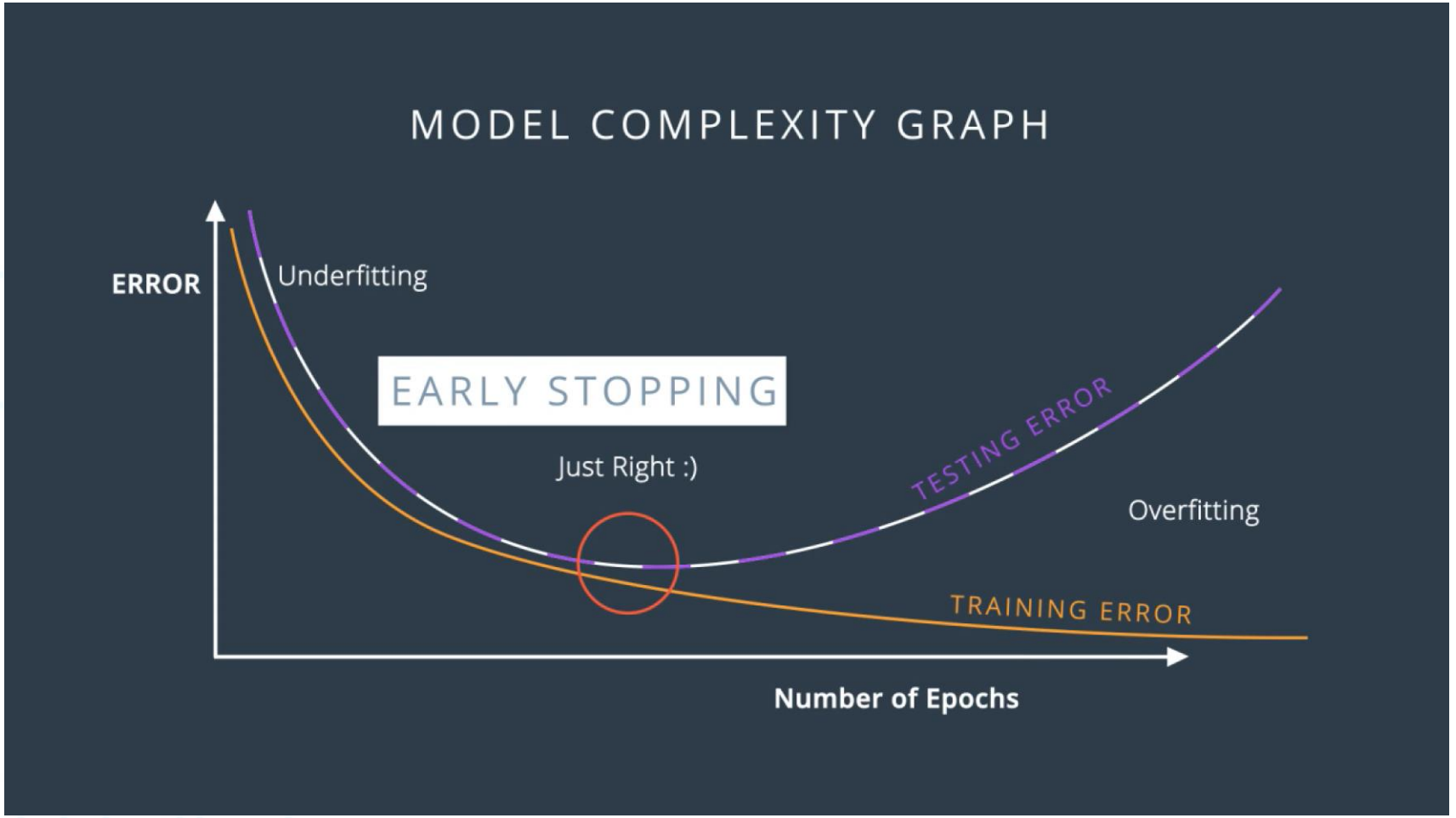
Redes Neurais – Model Complexity Graph



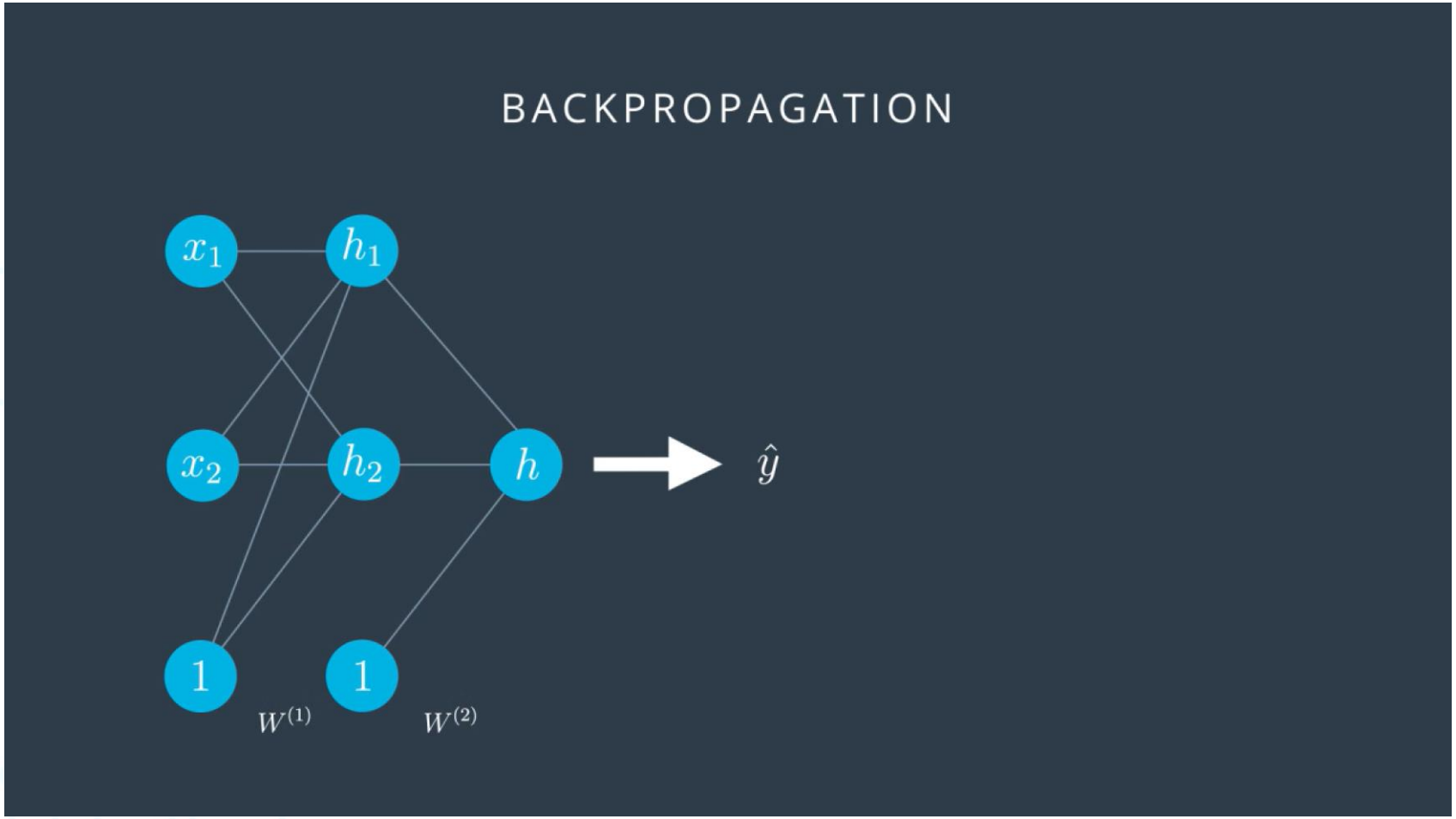
Redes Neurais – Model Complexity Graph



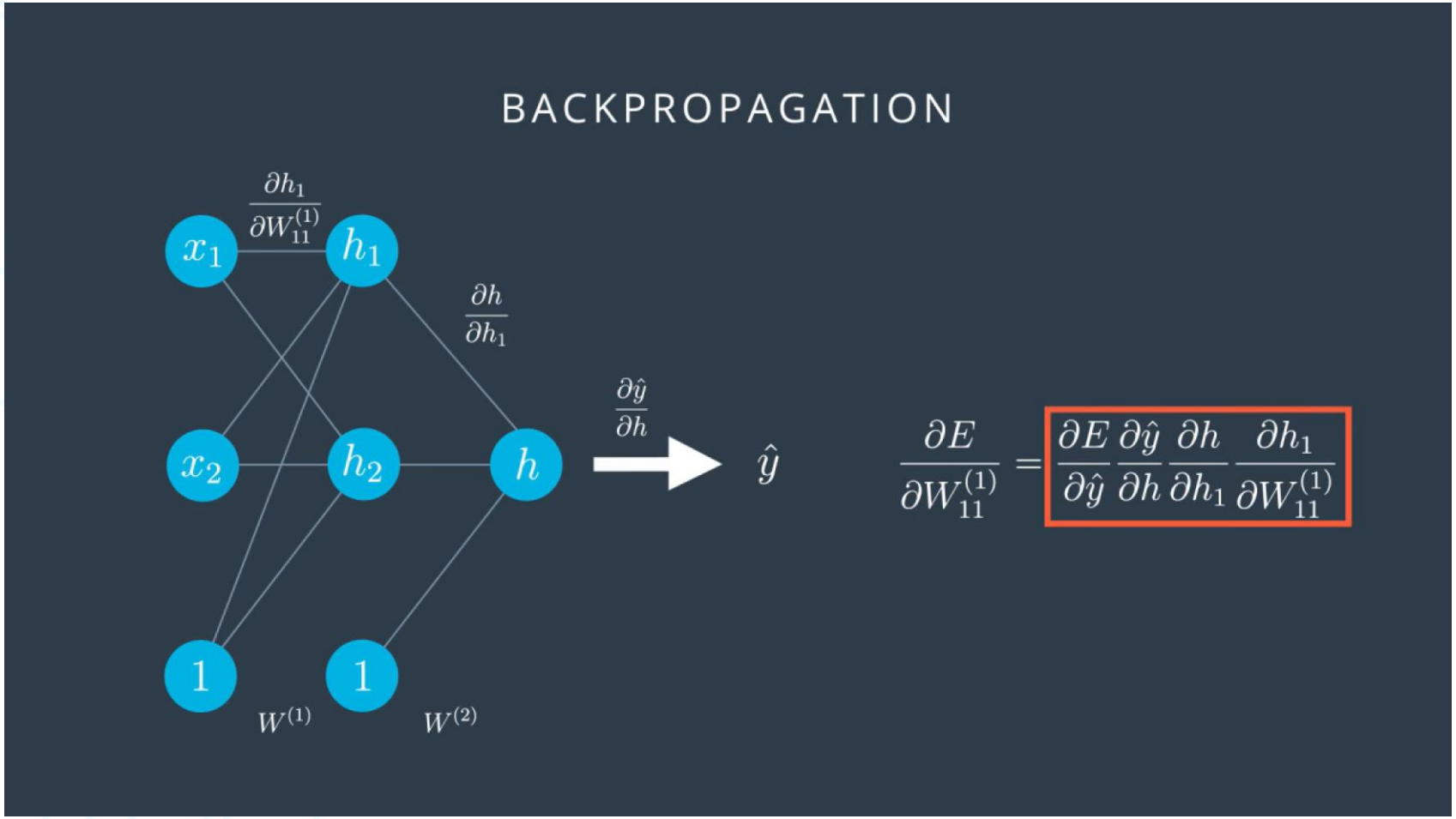
Redes Neurais – Model Complexity Graph



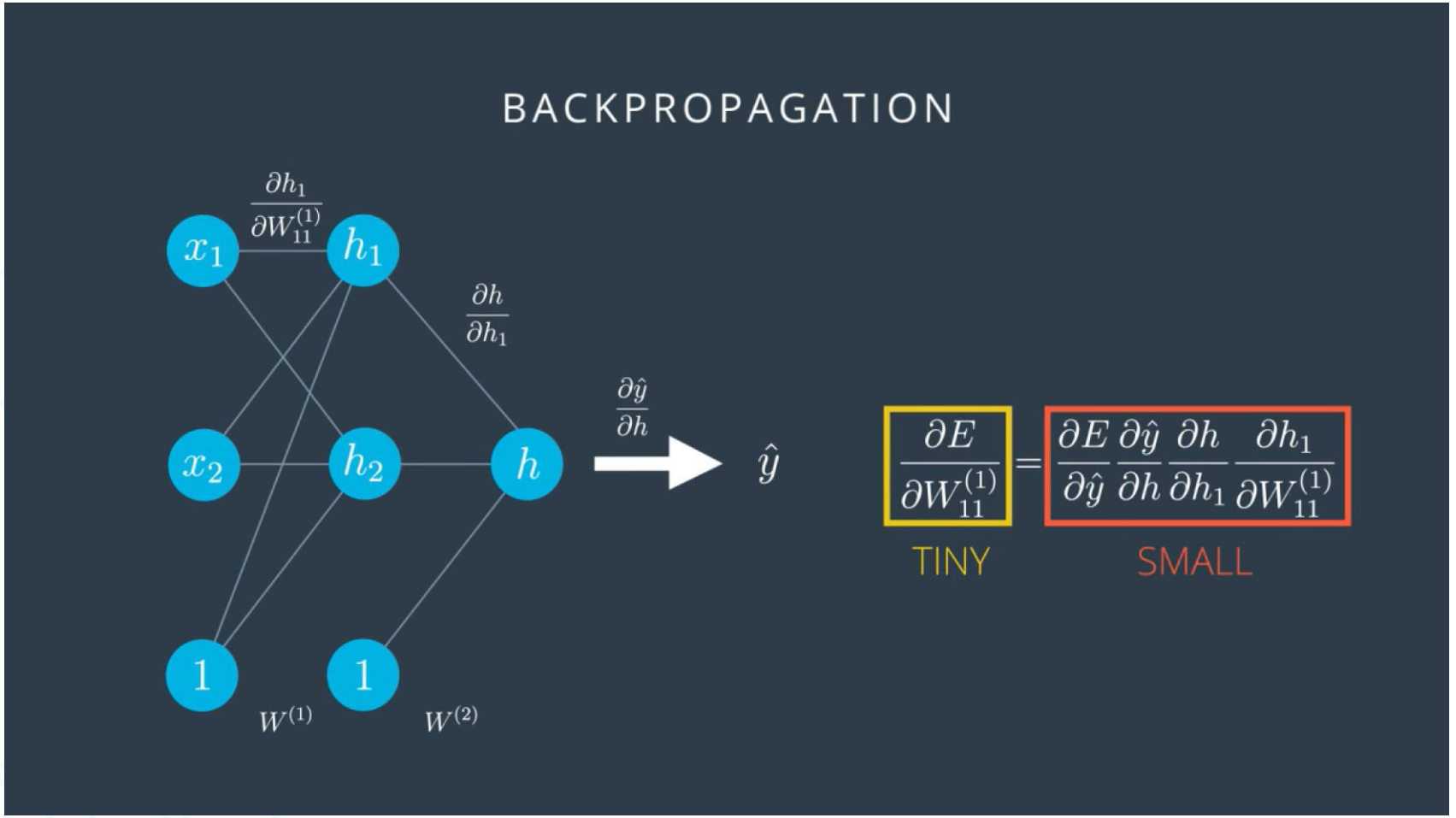
Vanishing Gradient



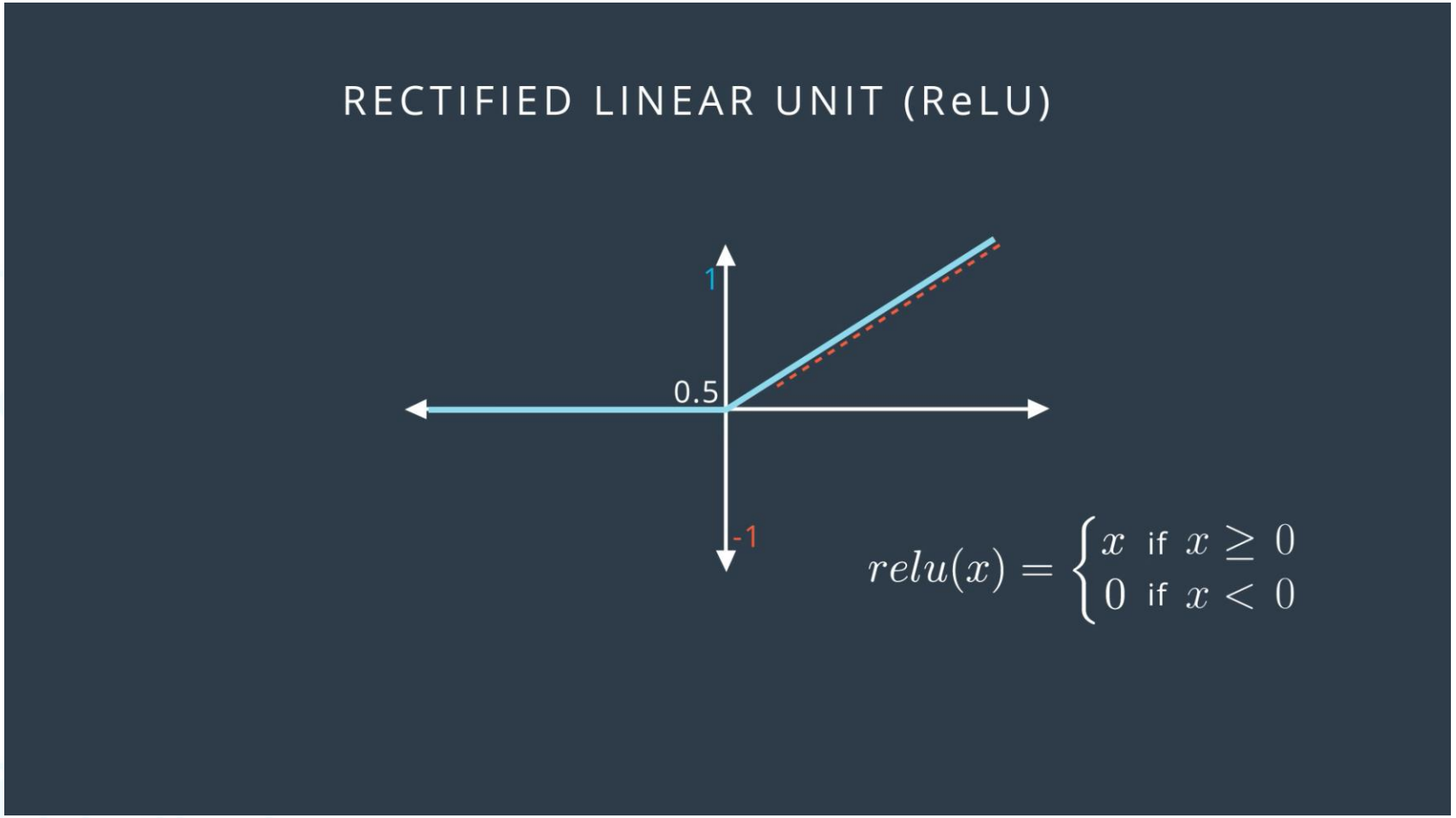
Vanishing Gradient



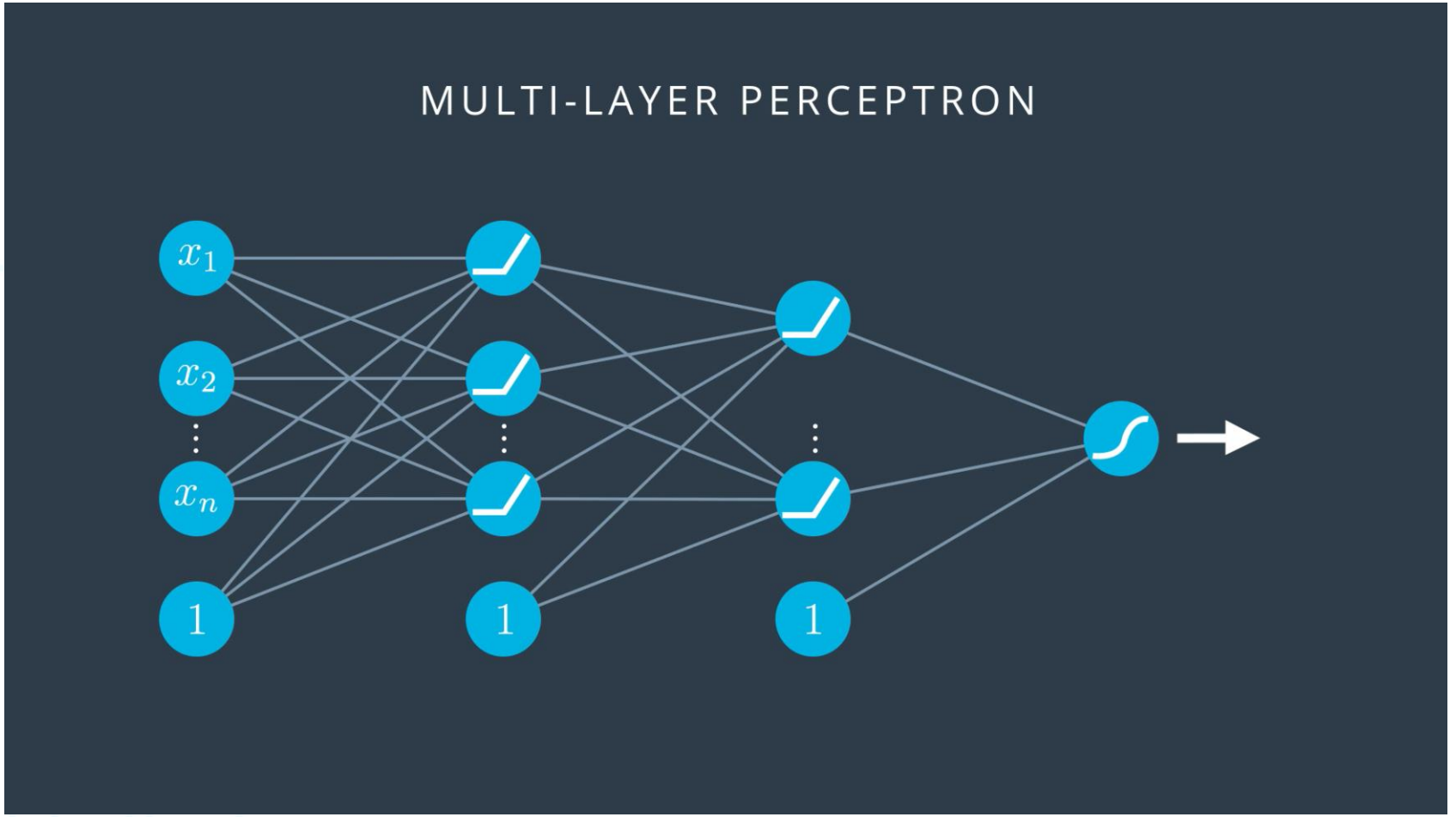
Vanishing Gradient



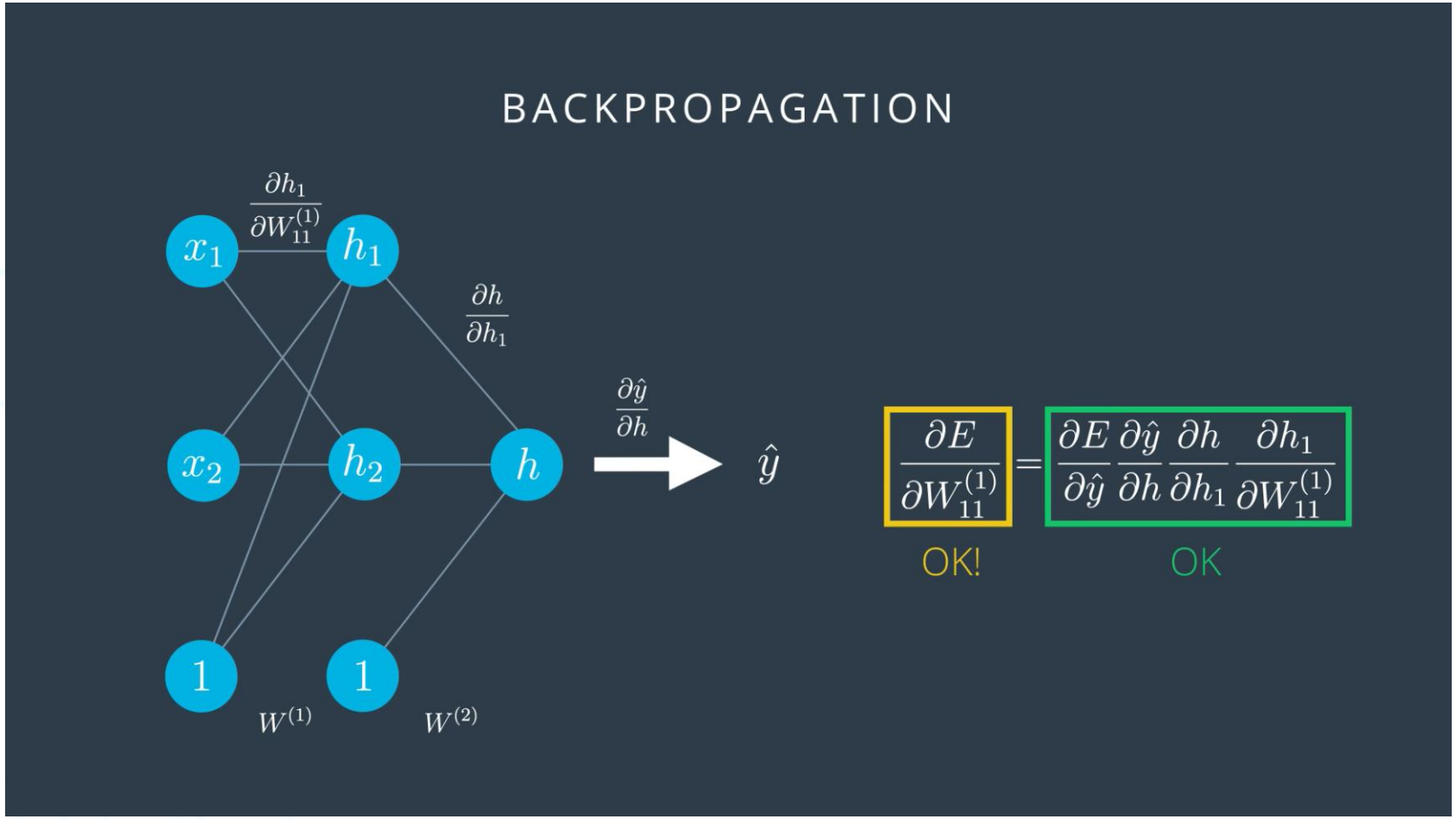
Funções de Ativação



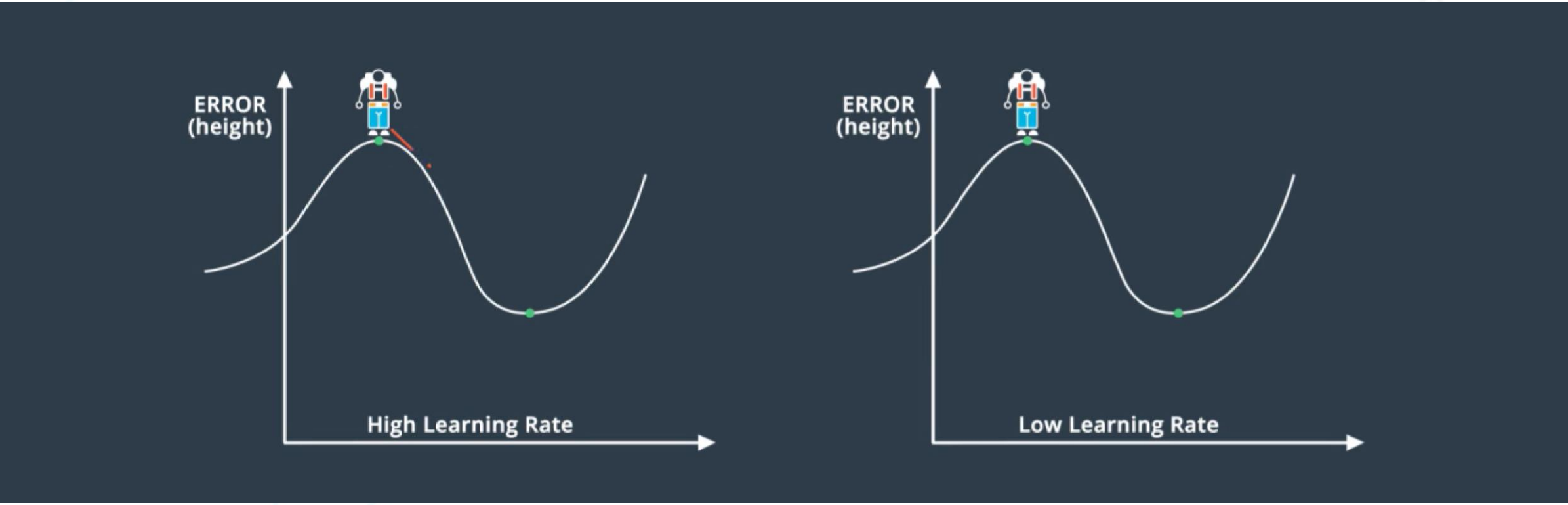
Funções de Ativação



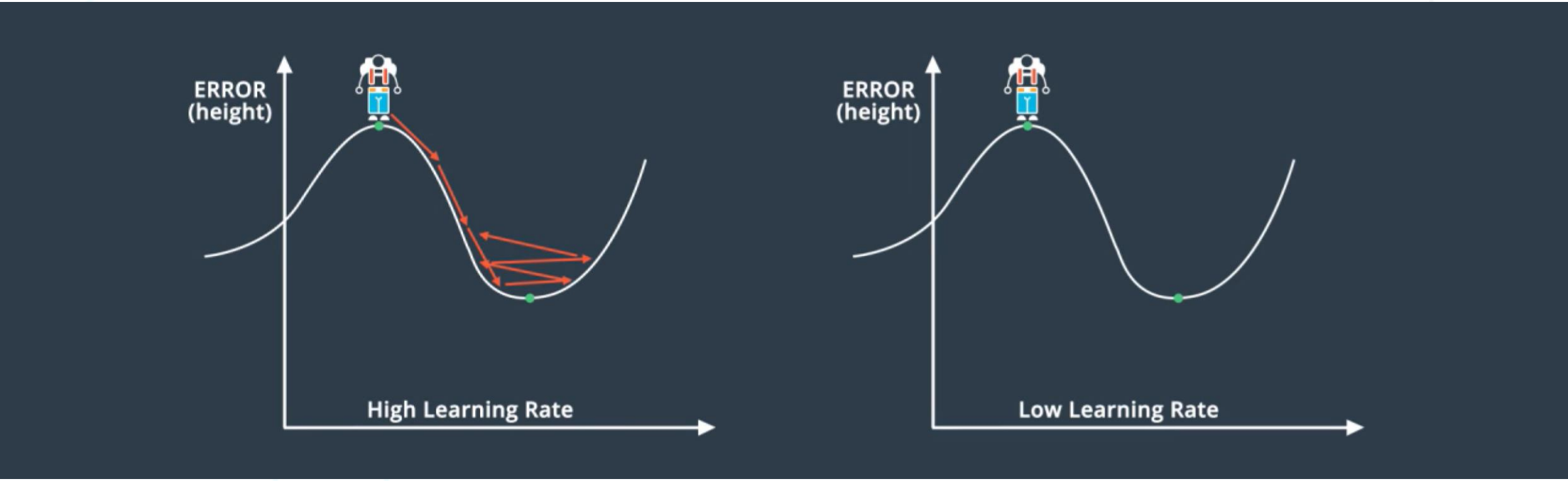
Funções de Ativação



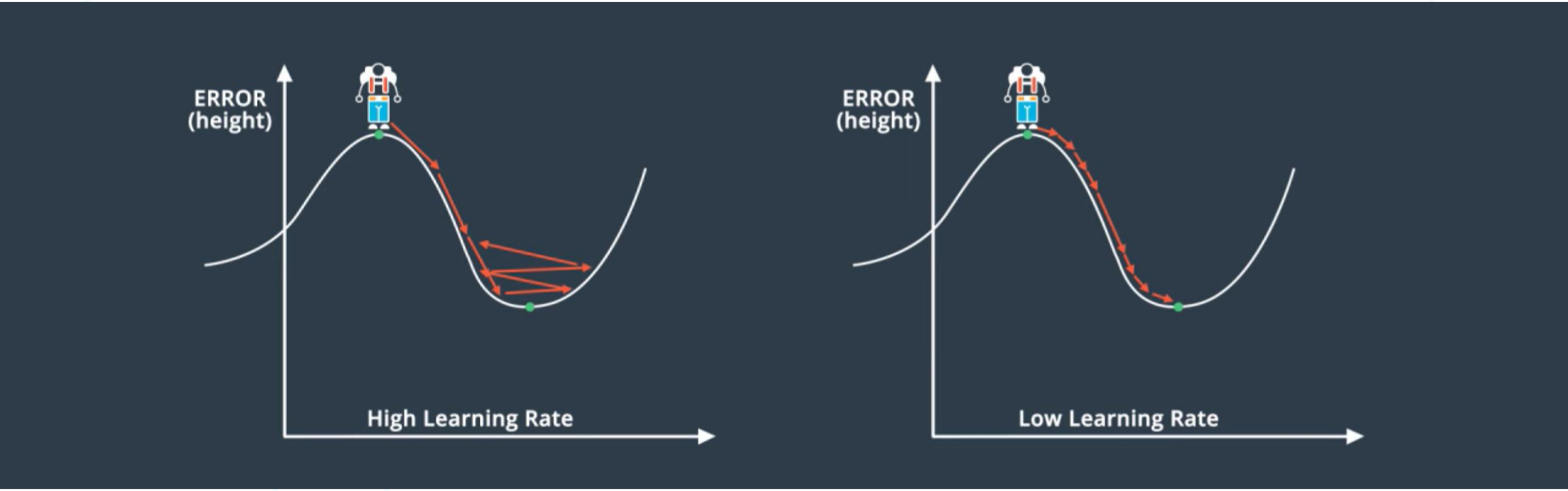
Learning Rate



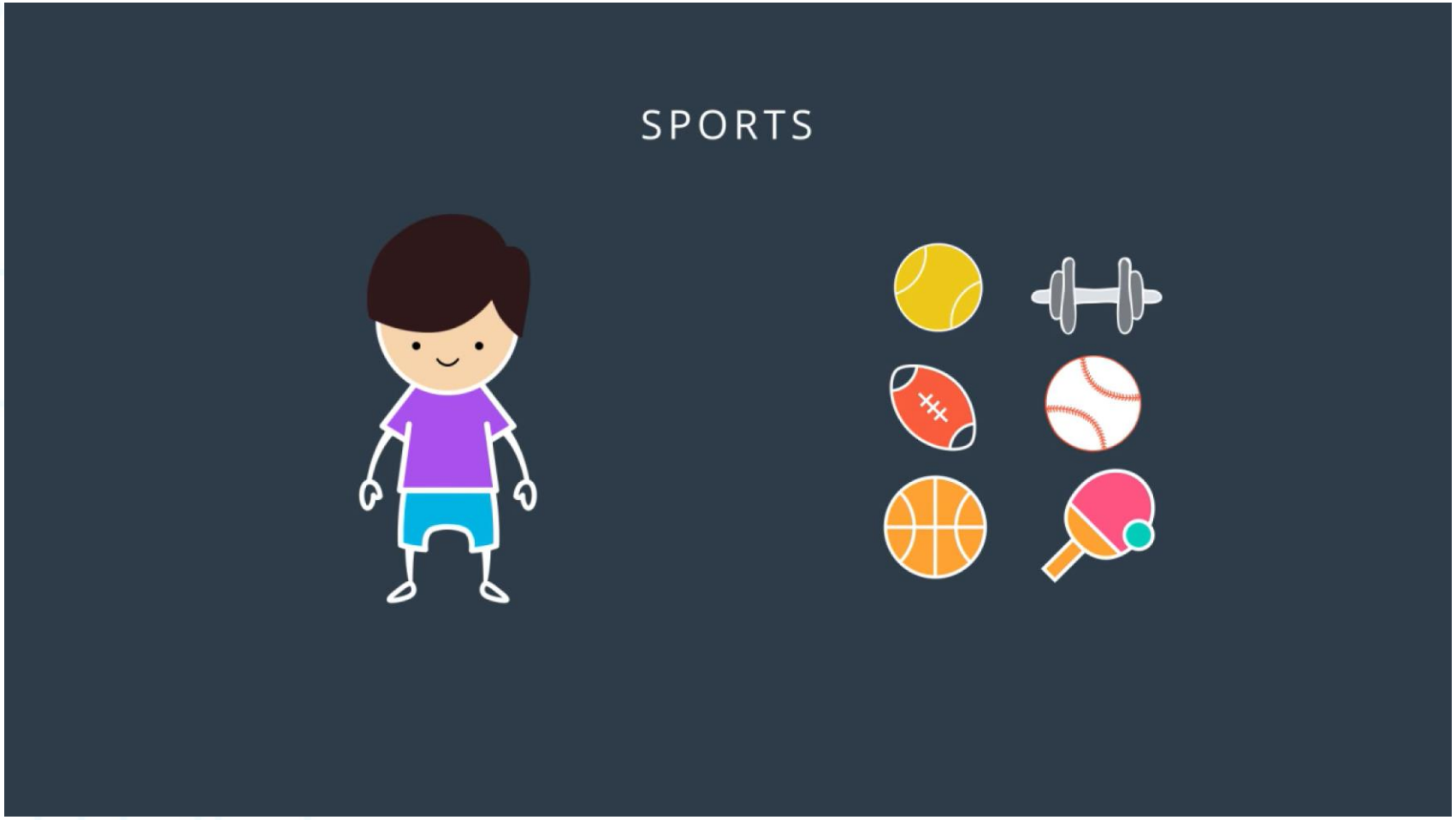
Learning Rate



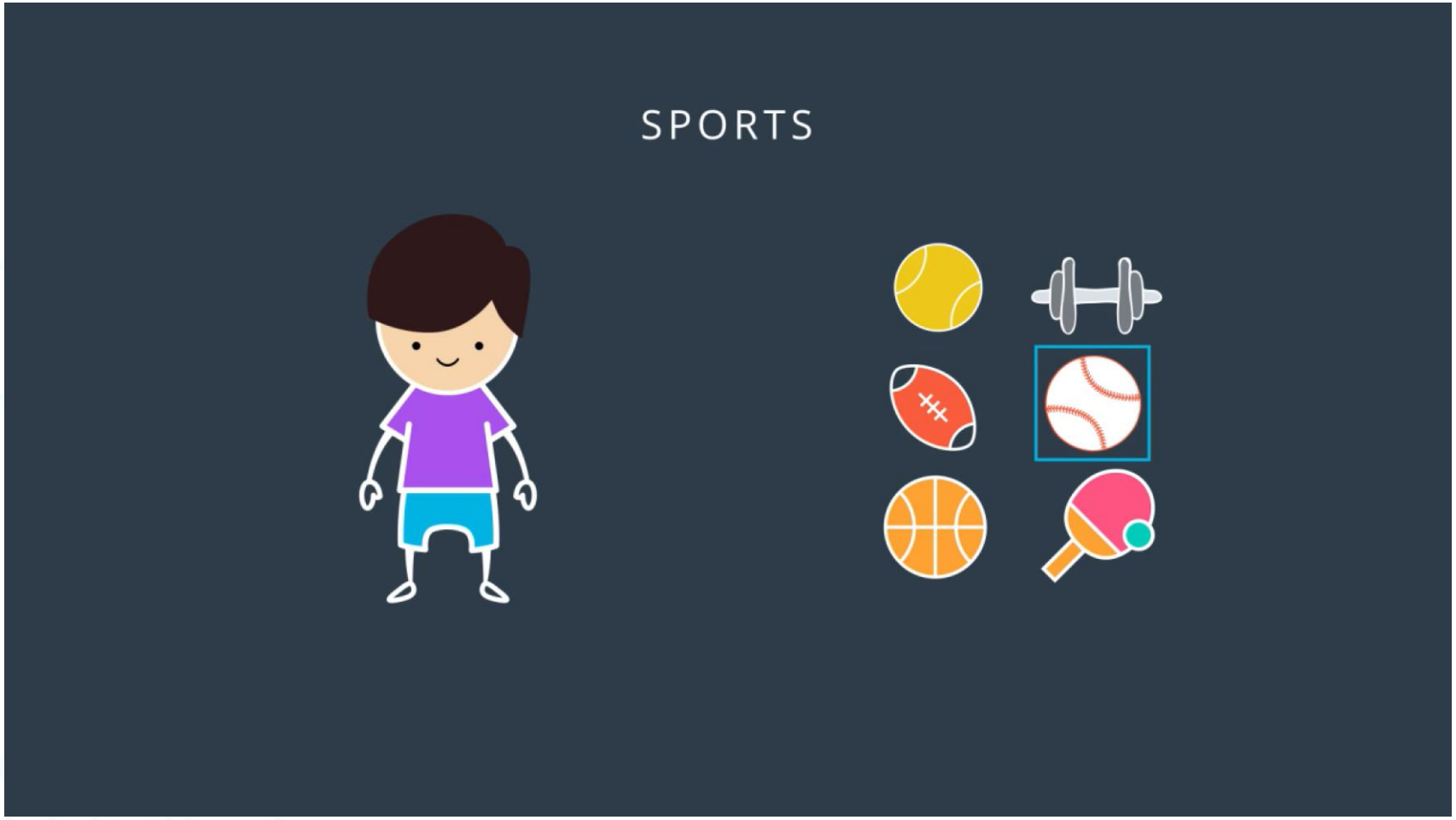
Learning Rate



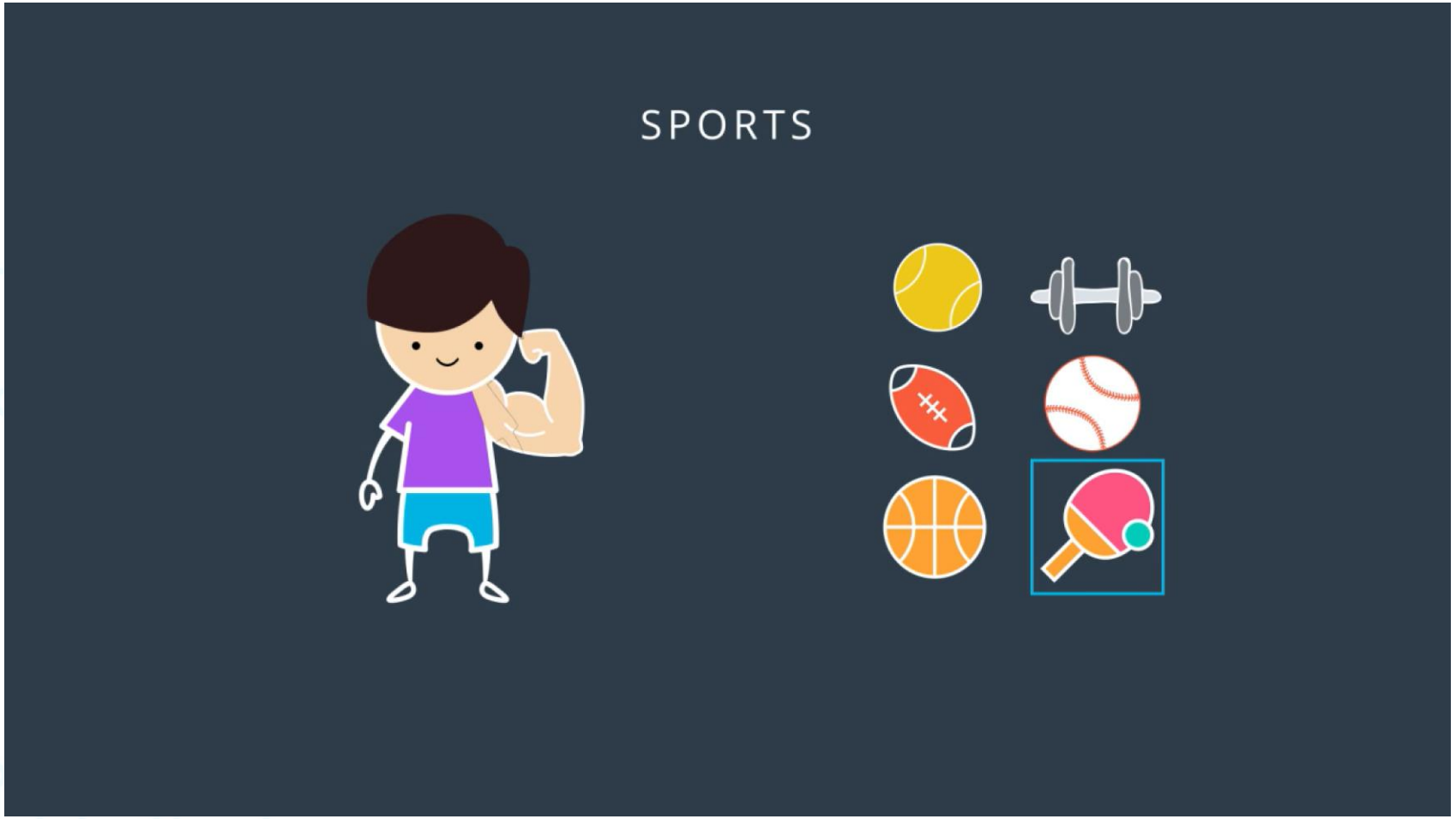
Redes Neurais – Dropout



Redes Neurais – Dropout

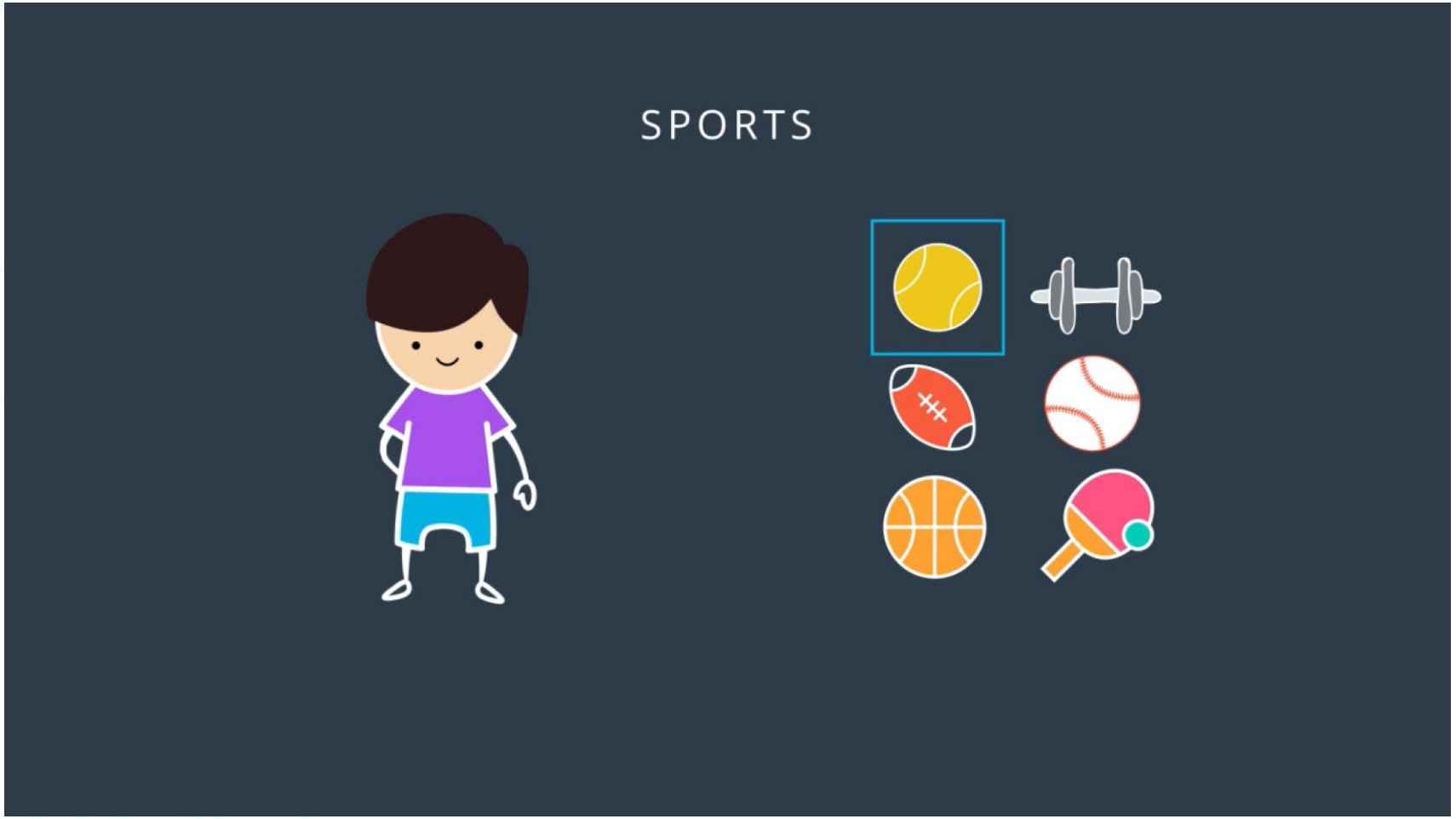


Redes Neurais – Dropout

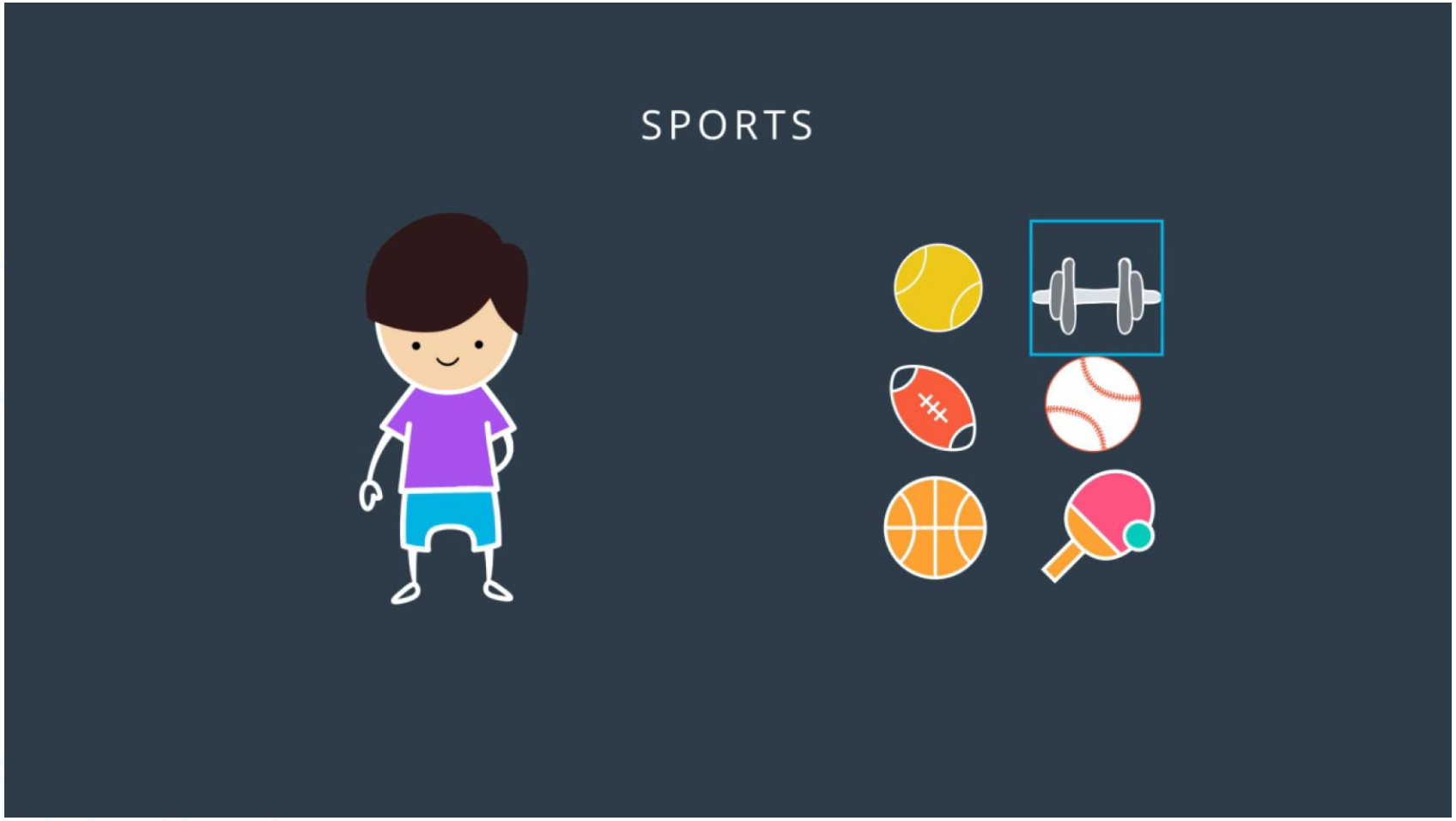


Redes Neurais – Dropout

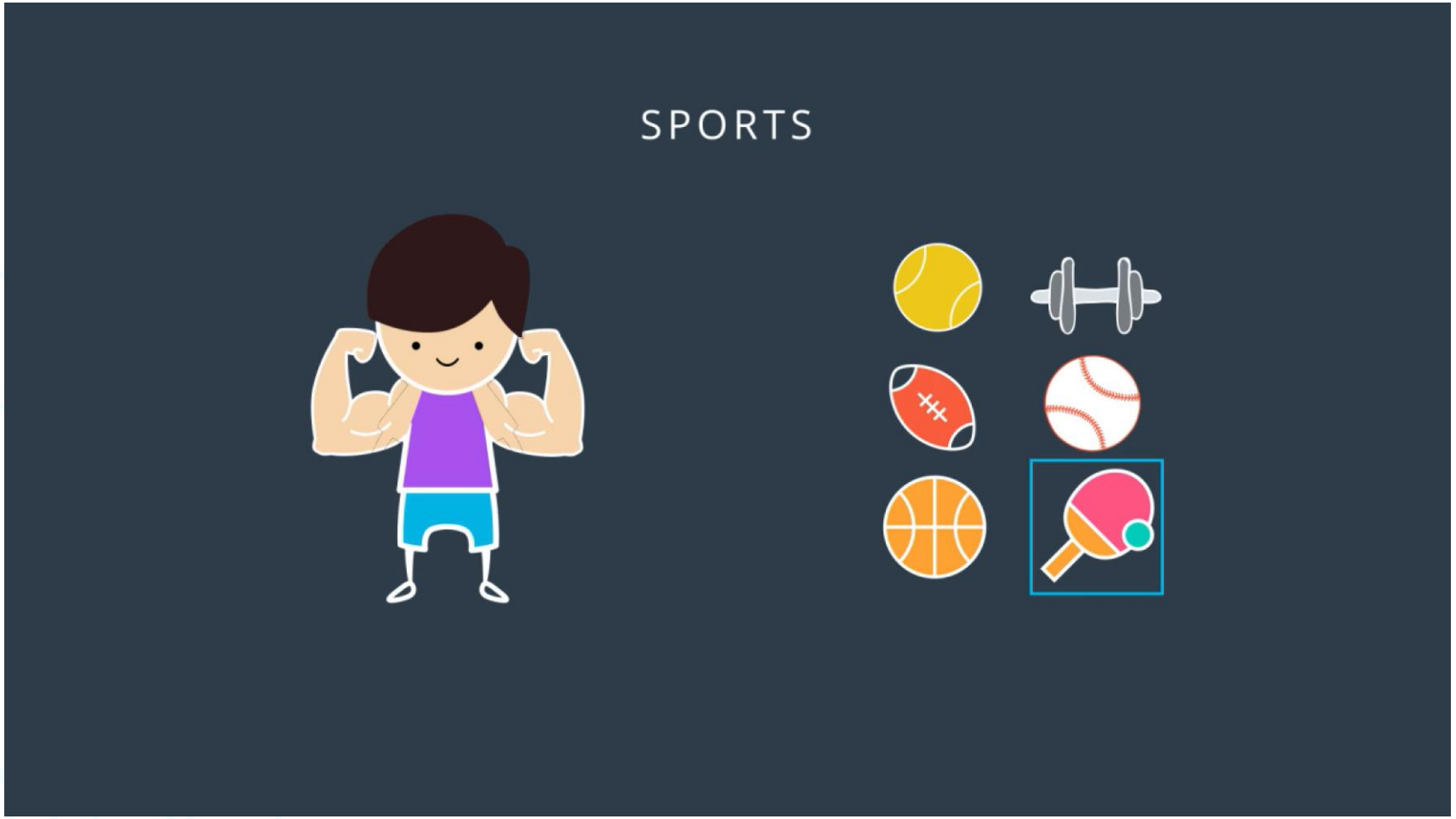
Redes Neurais – Dropout



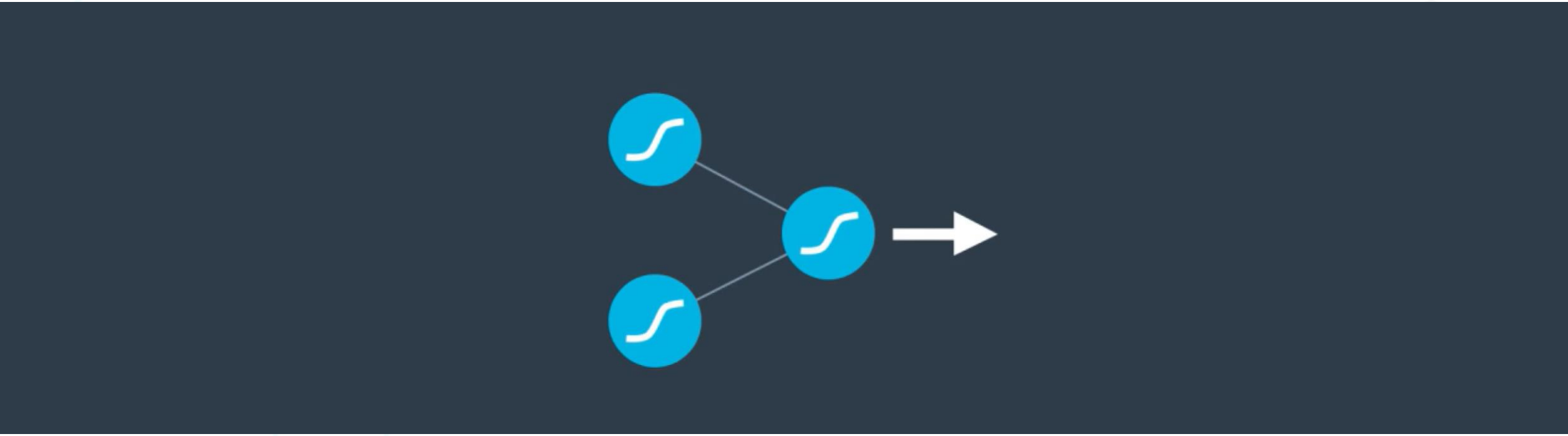
Redes Neurais – Dropout



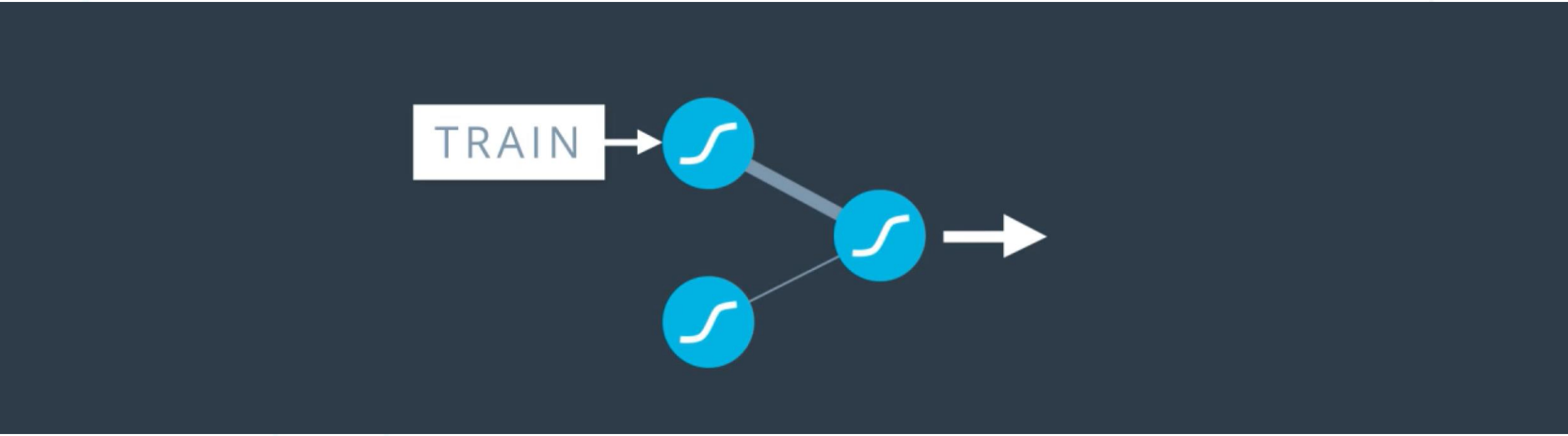
Redes Neurais – Dropout



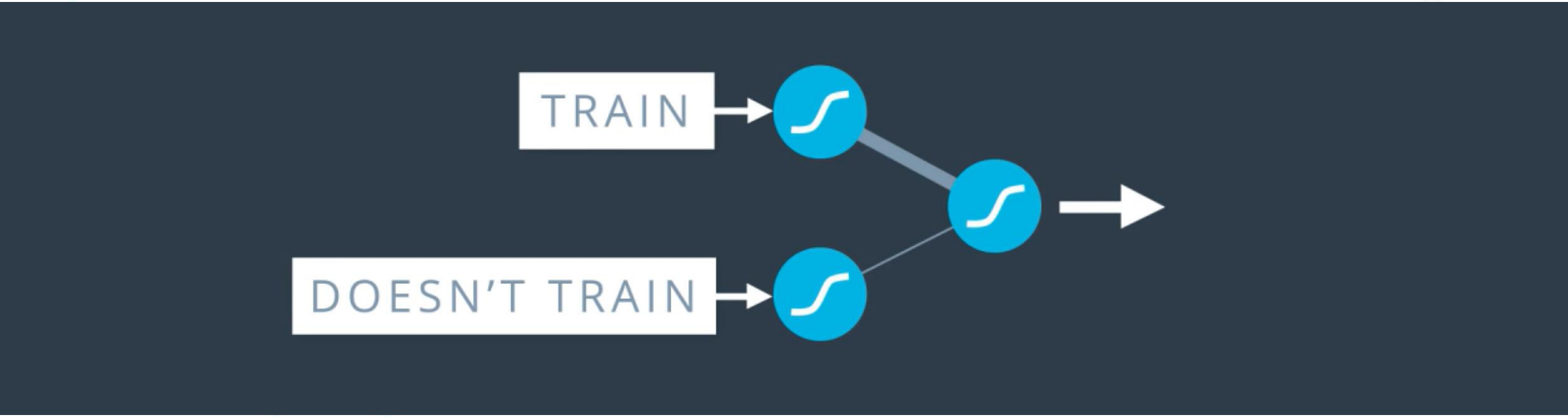
Redes Neurais – Dropout



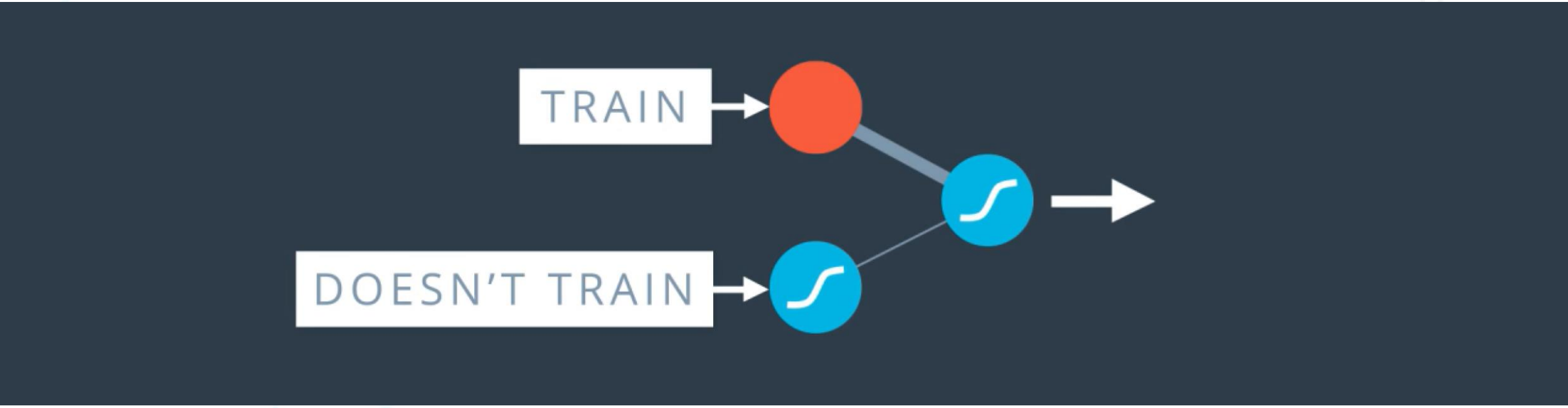
Redes Neurais – Dropout



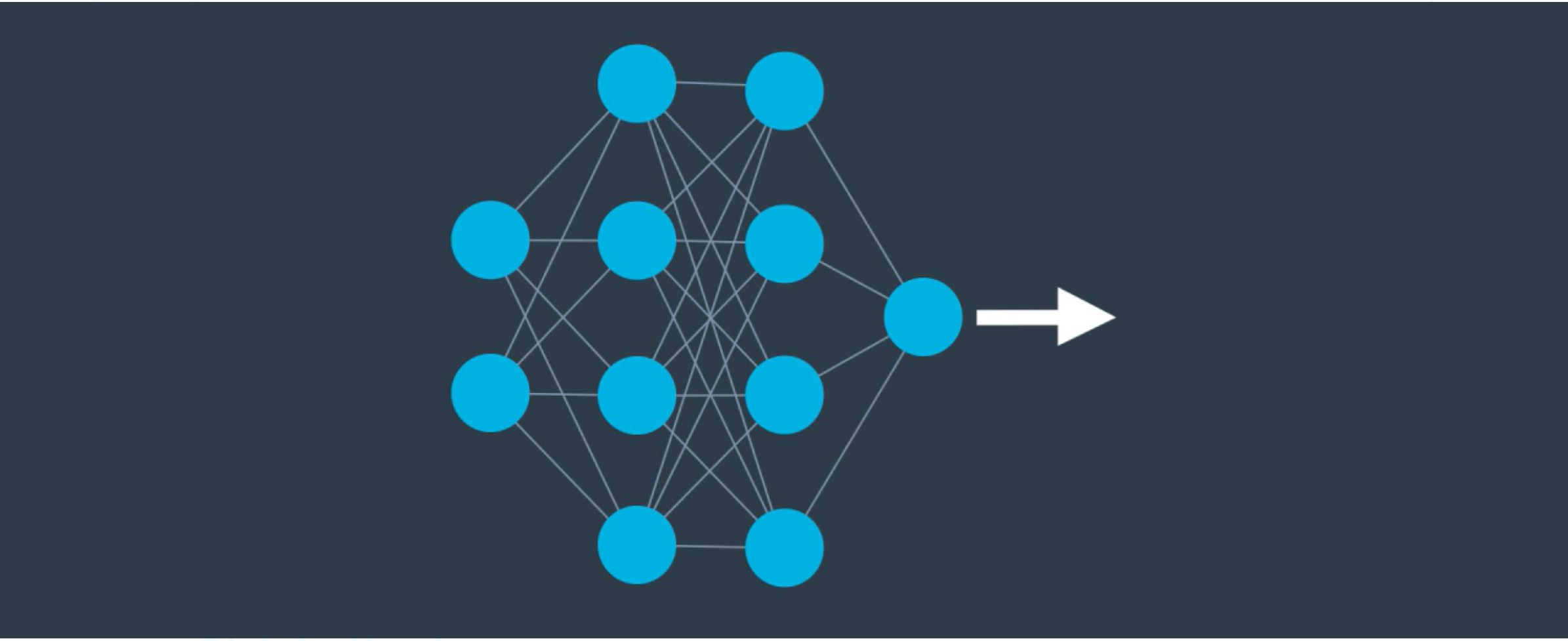
Redes Neurais – Dropout



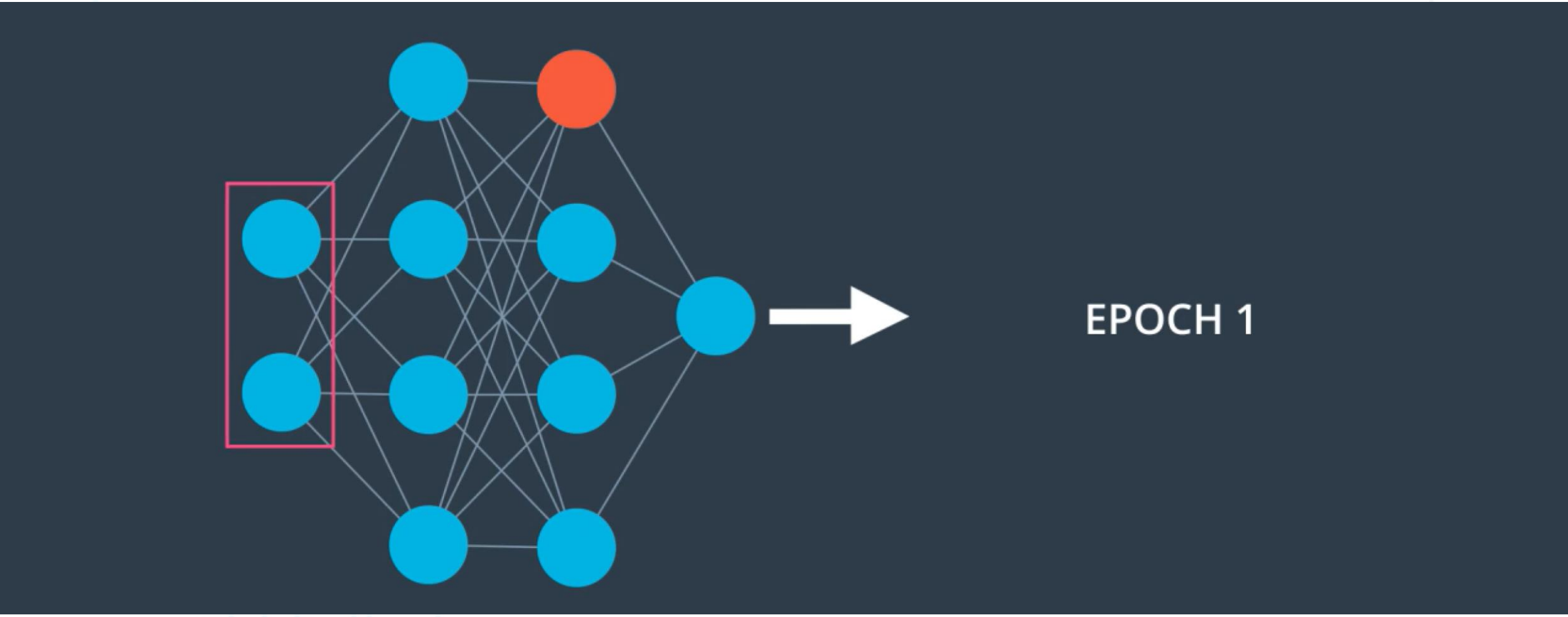
Redes Neurais – Dropout



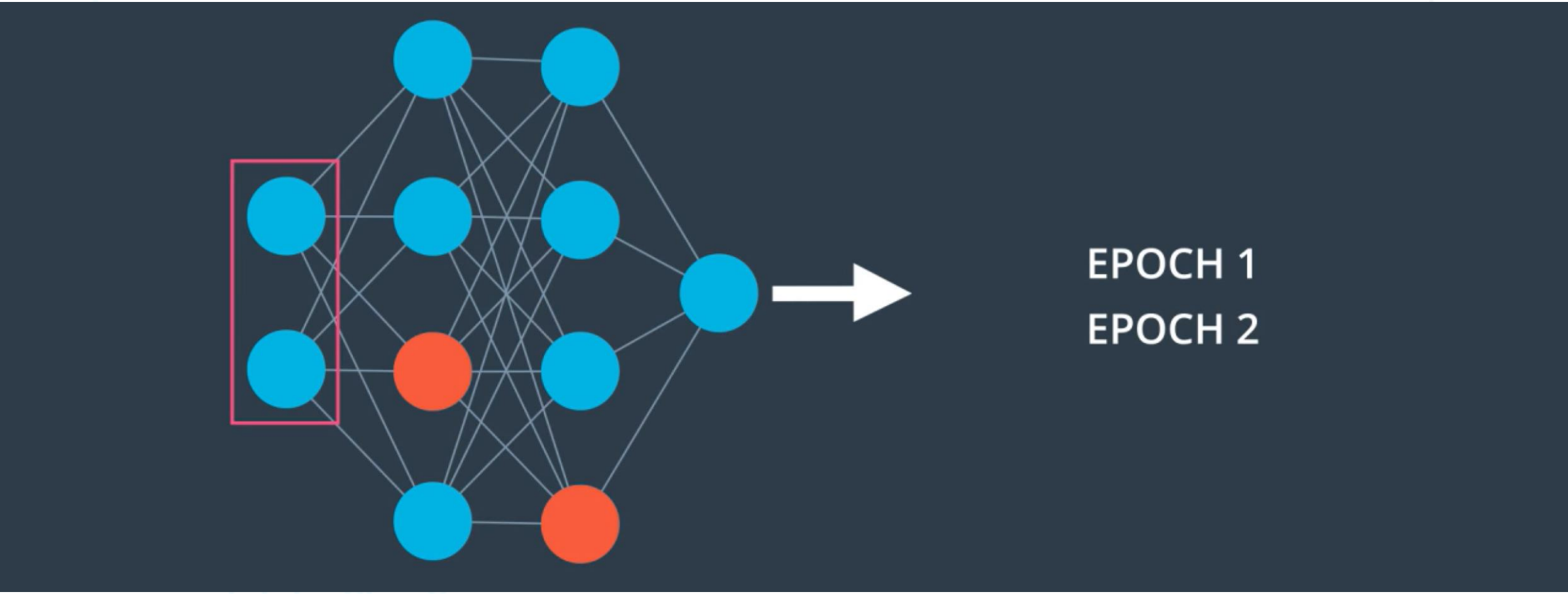
Redes Neurais – Dropout



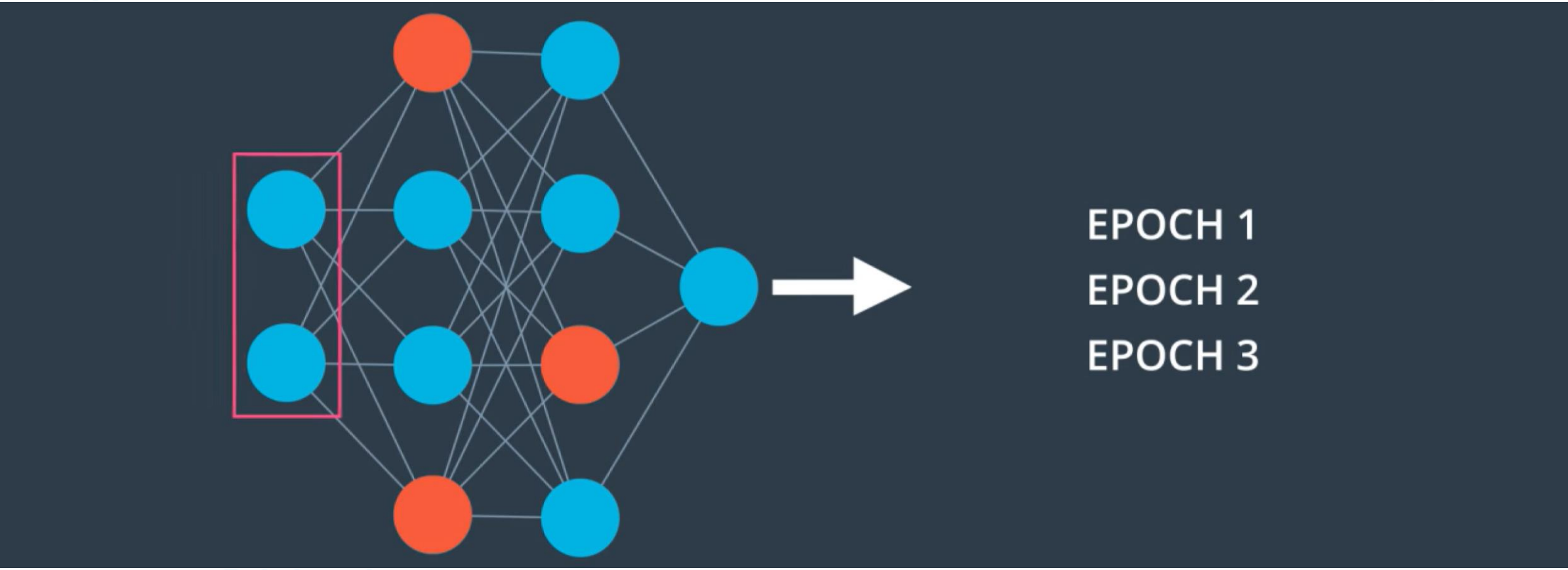
Redes Neurais – Dropout



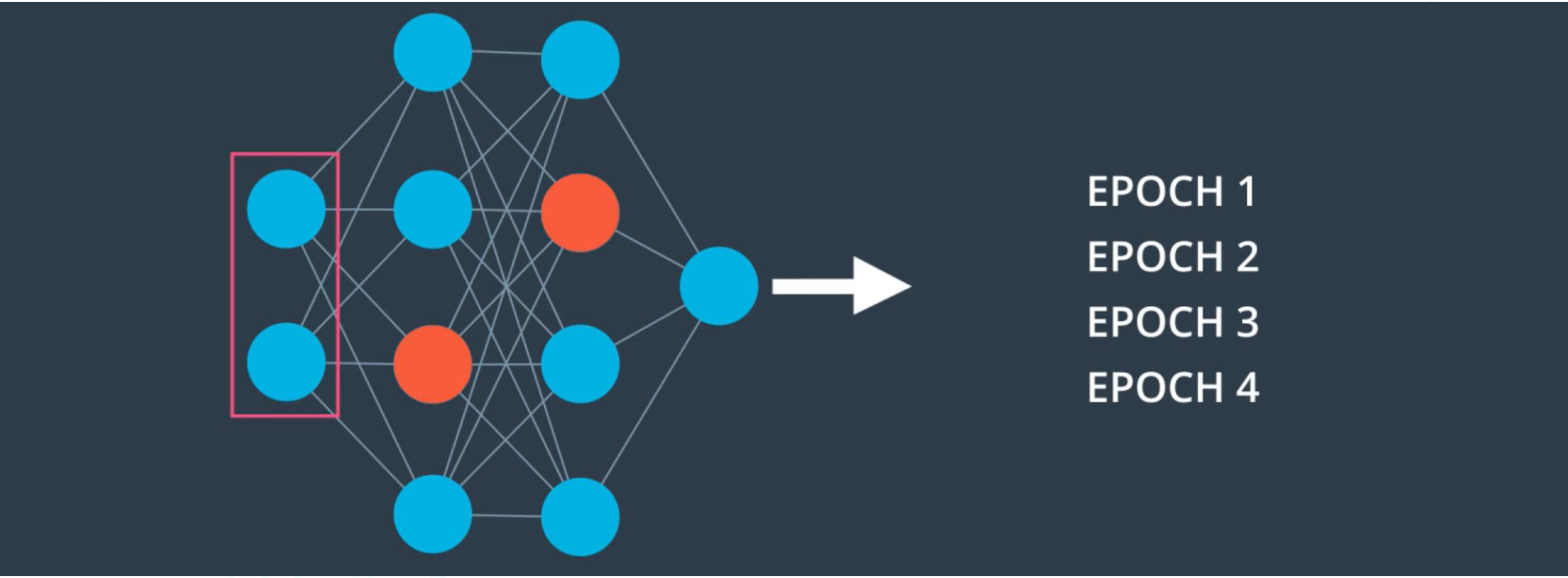
Redes Neurais – Dropout



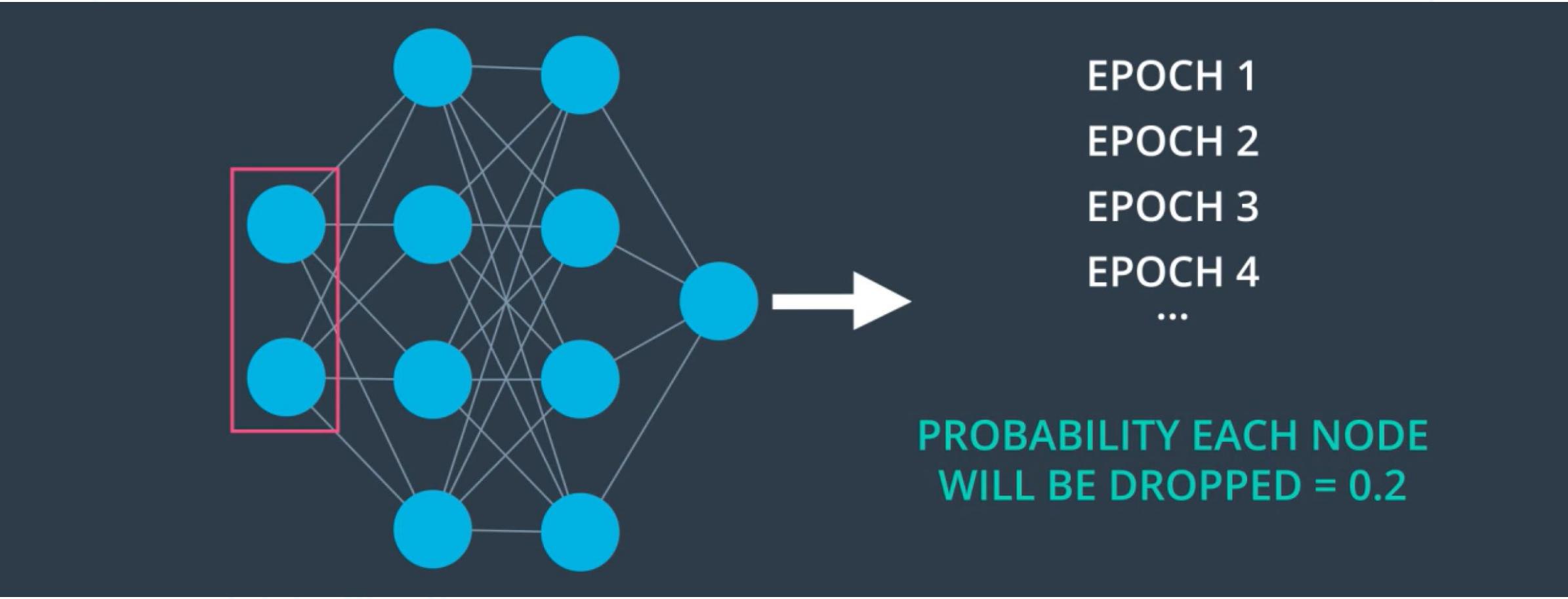
Redes Neurais – Dropout



Redes Neurais – Dropout



Redes Neurais – Dropout



Thanks !

Vinicius Fernandes Caridá

vfcarida@gmail.com


@vinicius caridá


@vfcarida


@vinicius caridá


@vfcarida


@vinicius caridá


@vfcarida