

# AULA 005

## DEEP LEARNING



***Praticamente, tudo que se chama  
deep learning é movido por um  
algoritmo muito importante:  
Stochastic Gradient Descent (SGD).***

- Goodfellow et al (2016)

# MODELOS PARAMÉTRICOS

**Tem a capacidade de aprender padrões a partir de dados de input durante uma etapa conhecida como treinamento.**

# MODELOS PARAMÉTRICOS

**Esse modelo é representado não por todos os dados que o alimentam, mas por um número limitado de parâmetros.**

# MODELOS PARAMÉTRICOS

**Independente do tamanho do meus dados de treino,  
é possível generalizar, representar os mesmos por  
meio de parâmetros.**





**1 (avião)**

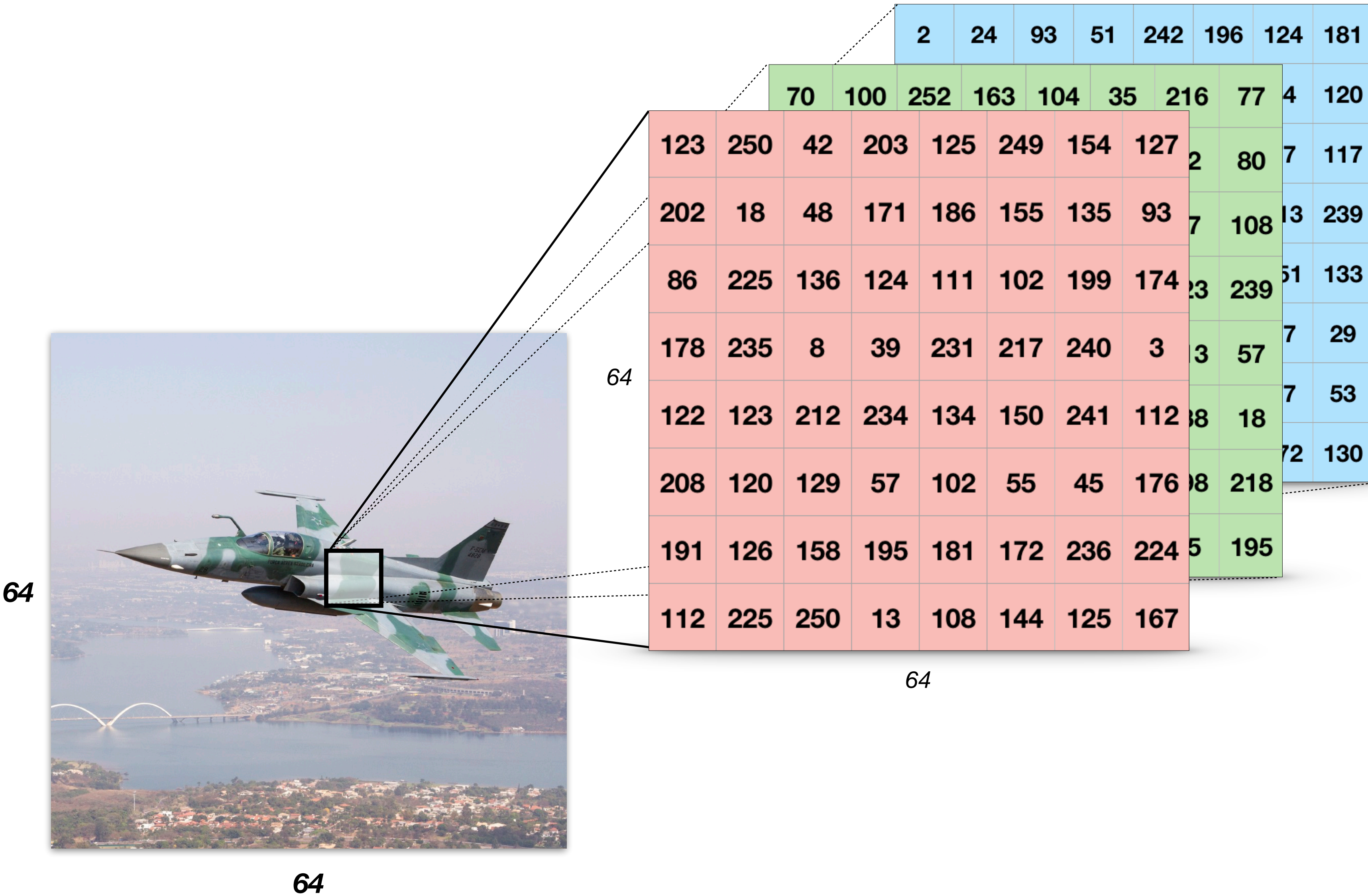


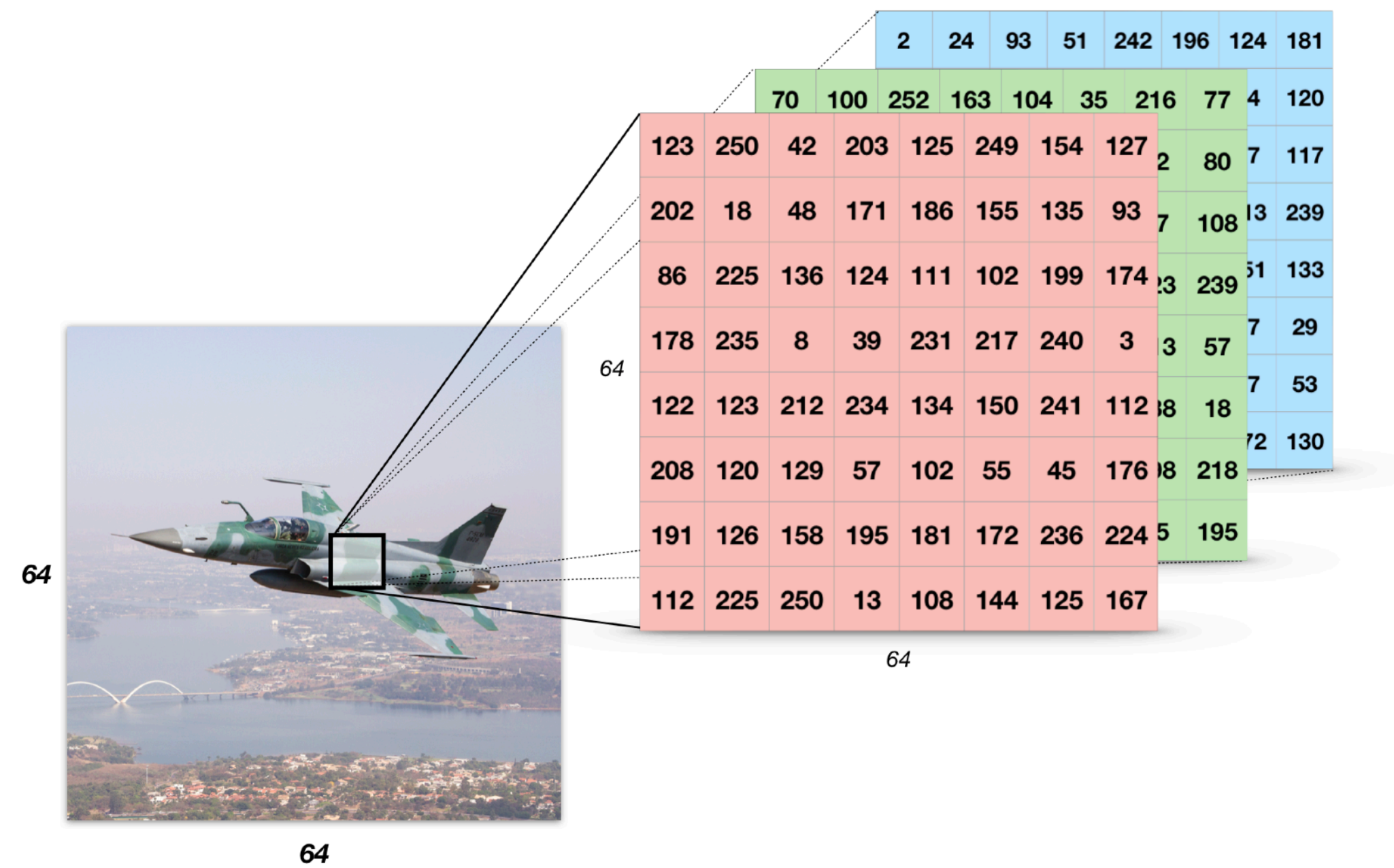
**0 (não-avião)**



**1 (avião)**







$$x = \begin{bmatrix} 123 \\ 250 \\ \vdots \\ 70 \\ 100 \\ \vdots \\ 130 \end{bmatrix}$$

$$y = 1$$

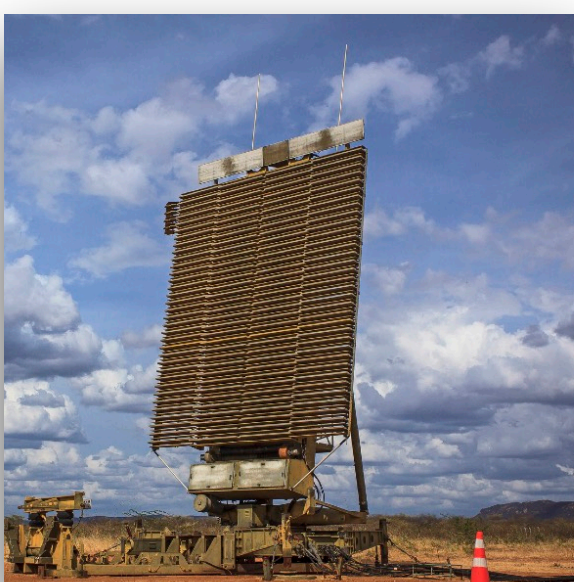
$$n_x = 64 \times 64 \times 3 = 12288$$

$$m = 1$$





$$x^{(1)} = \begin{bmatrix} 12 \\ 122 \\ \vdots \\ 163 \end{bmatrix}$$



$$x^{(2)} = \begin{bmatrix} 231 \\ 146 \\ \vdots \\ 0 \end{bmatrix}$$



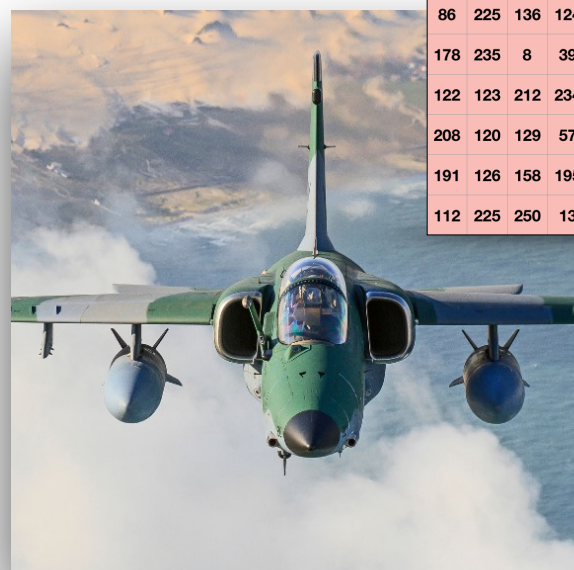
$$x^{(3)} = \begin{bmatrix} 22 \\ 13 \\ \vdots \\ 89 \end{bmatrix}$$

$$X = \begin{bmatrix} 12 & 231 & 22 \\ 122 & 146 & 13 \\ \vdots & \vdots & \vdots \\ 163 & 0 & 89 \end{bmatrix}$$

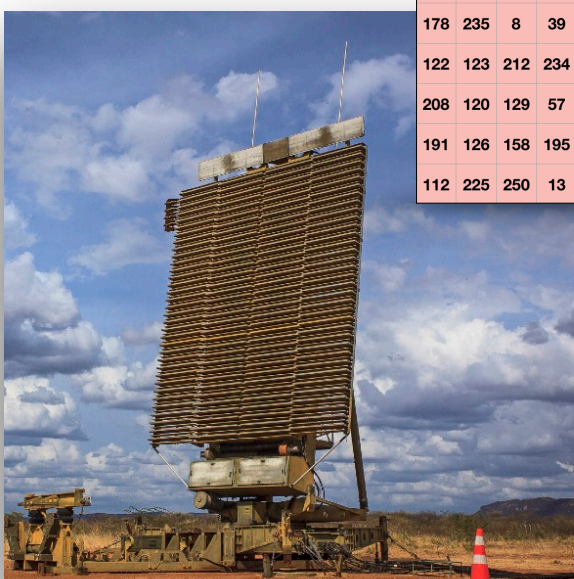
$$Y = [1, 0, 1]$$

$$n_x = 12288$$

$$m = 3$$

[illegible]

$$x^{(1)} = \begin{bmatrix} 12 \\ 122 \\ \vdots \\ 163 \end{bmatrix}$$



					2	24	93	51	242	196	124	161	
			70	100	252	163	104	35	216	77	4	120	
123	250	42	203	125	249	154	217						117
										2	80	7	120
202	18	48	147	186	155	135	93					13	239
86	225	136	124	111	102	199	179			13	239	1	133
178	235	8	39	231	217	240	3		3	57	7	29	
128	123	212	234	134	150	241	112	18		18			
													2 130
202	120	129	57	102	55	45	176	18					
191	126	158	195	181	172	236	224	5		195			
112	225	250	13	108	144	125	167						

$$x^{(2)} = \begin{bmatrix} 231 \\ 146 \\ \vdots \\ 0 \end{bmatrix}$$

$$X = \begin{bmatrix} 12 & 231 & 22 \\ 122 & 146 & 13 \\ \vdots & \vdots & \vdots \\ 163 & 0 & 89 \end{bmatrix}$$

$$Y = [1,0,1]$$

[illegible]

$$x^{(3)} = \begin{bmatrix} 22 \\ 13 \\ \vdots \\ 89 \end{bmatrix}$$

$$n_x = 12288$$

$$m = 3$$

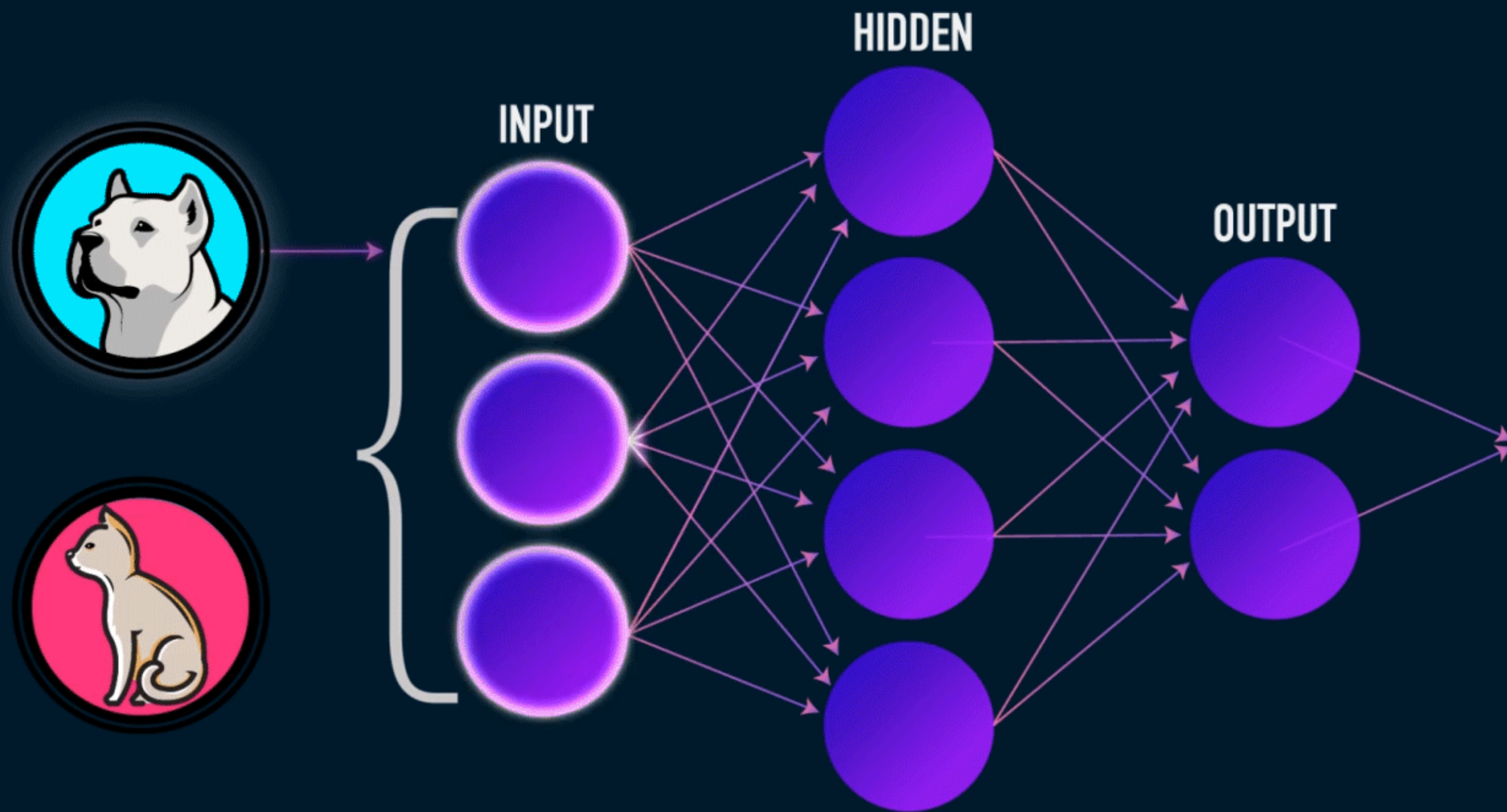
# Para um conjunto de $m$ amostras de treinamento

**$m$ -amostras:**  $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$

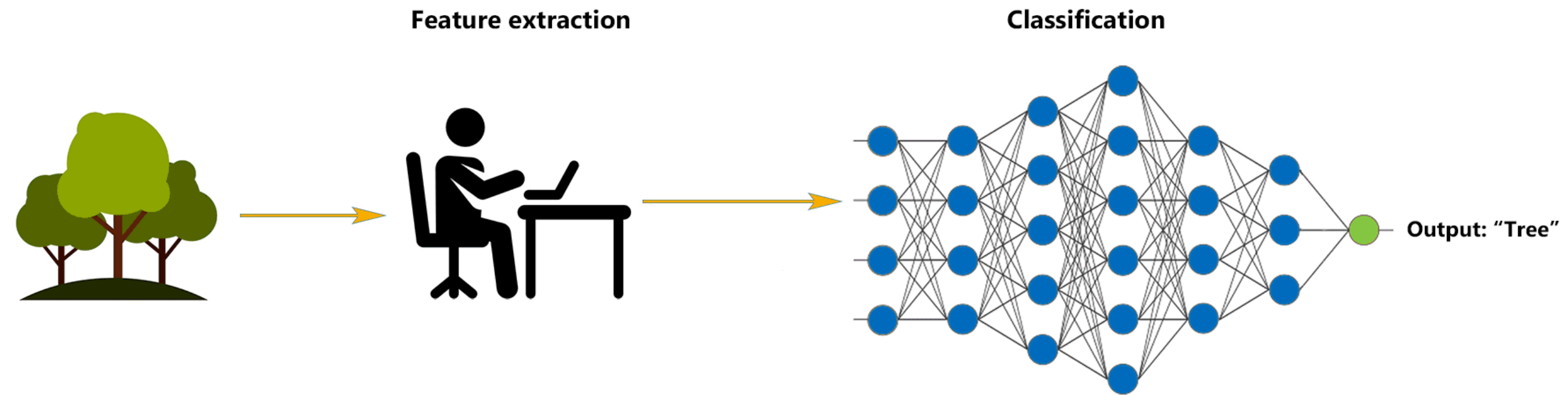
$$X = \left[ \begin{array}{c|c|c|c} | & | & & | \\ x^{(1)} & x^{(2)} & \dots & x^{(m)} \\ | & | & & | \end{array} \right]$$

$$Y = [y^{(1)}, y^{(1)}, \dots, y^{(m)}]$$



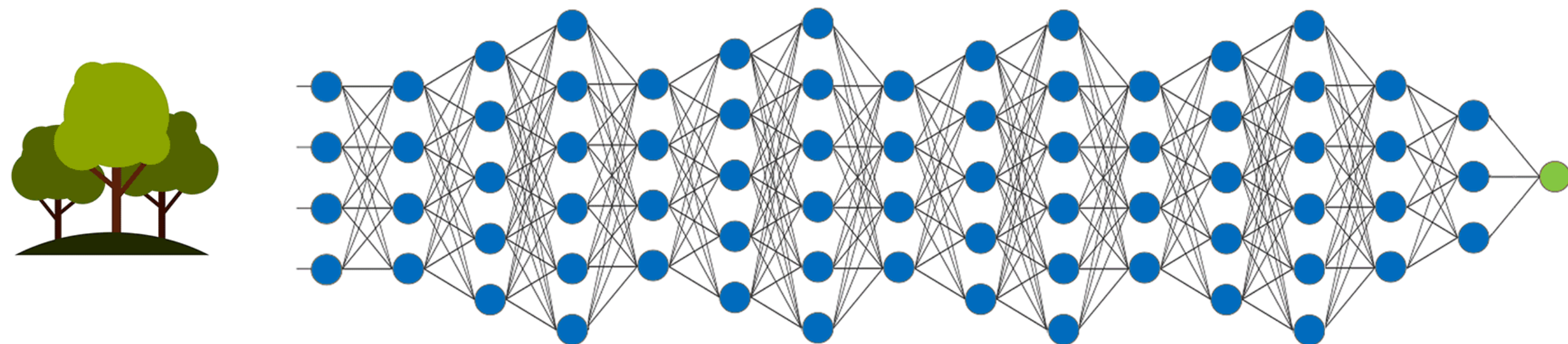


# Machine Learning

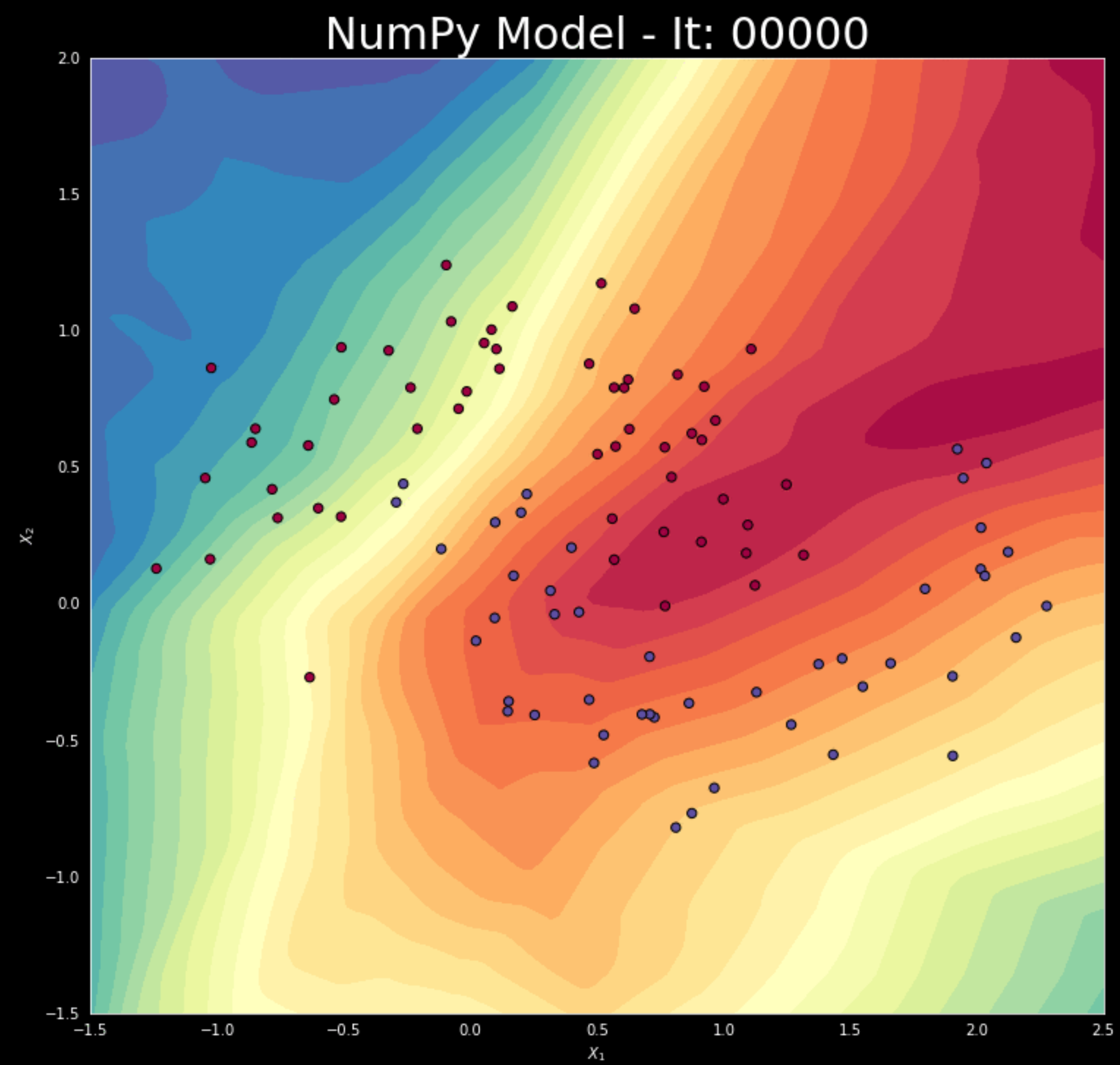


# Deep Learning

Feature extraction + Classification

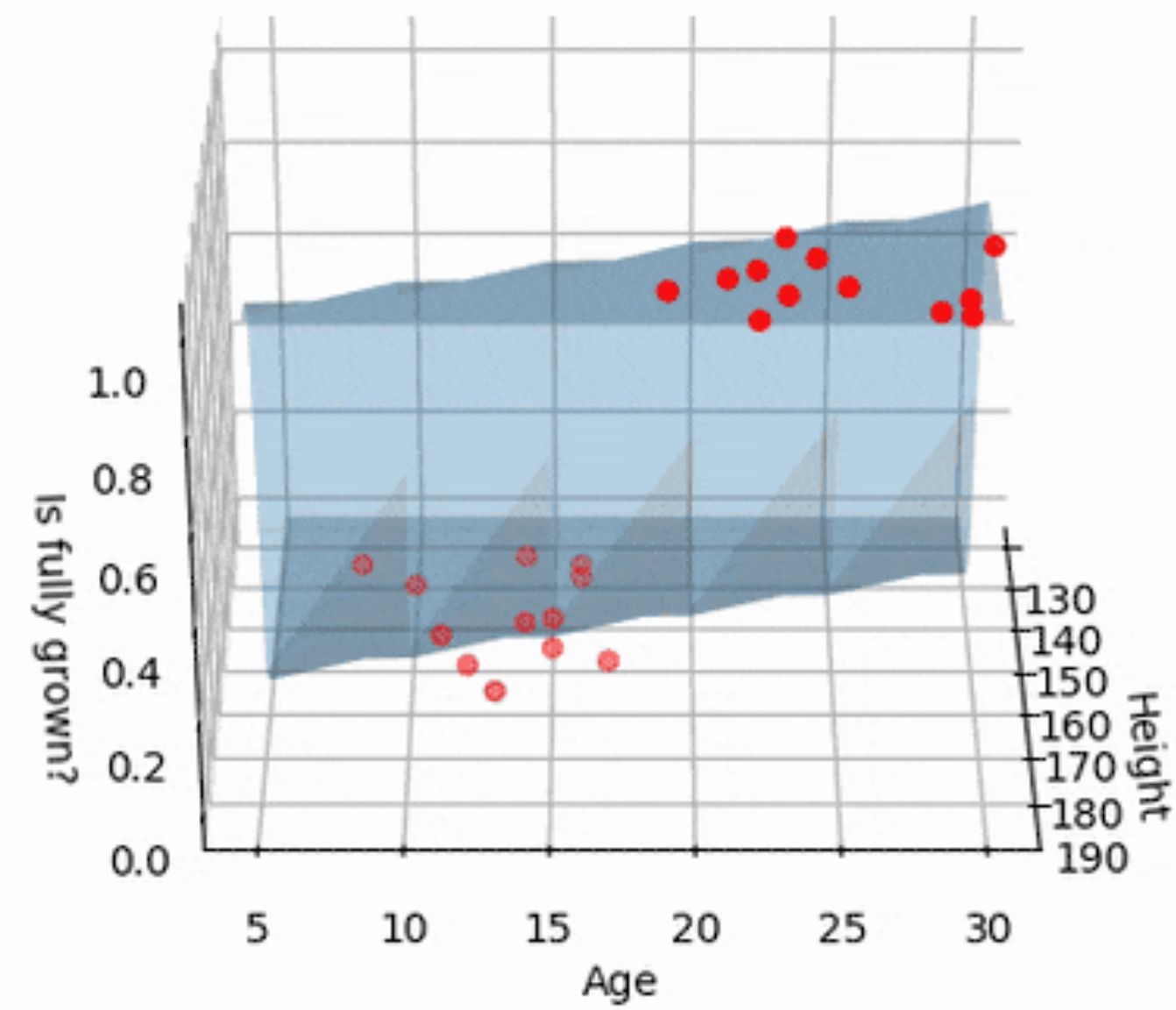




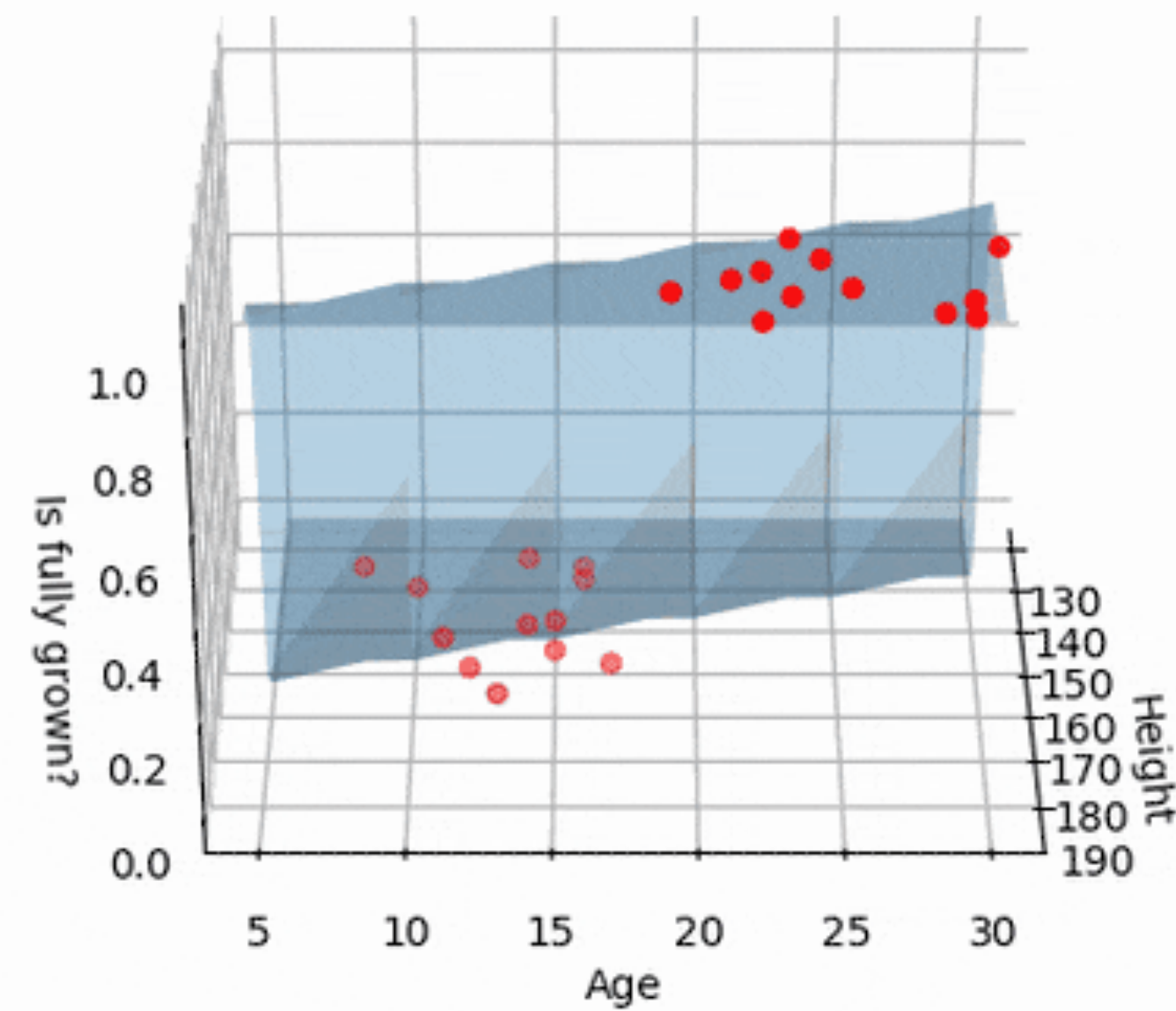




# REGRESSÃO LOGÍSTICA



# REGRESSÃO LOGÍSTICA



$$\hat{y} = P(y = 1 | x)$$

# 1. SCORE FUNCTION

*A score function é uma função que vai receber o nosso input (no exemplo usado aqui, uma imagem) e mapear os dados para as classes de labels.*

$$f(x, \theta, b) = \hat{y} = \theta^T x + b$$



Lembra que estamos falando de classificação de imagens, e da probabilidade dessa imagem ser um avião ou não?

**O problema é que a nossa score function, do jeito que está, pode retornar qualquer valor.**

$$0 \leq \hat{y} \leq 1$$

# FUNÇÕES DE ATIVAÇÃO

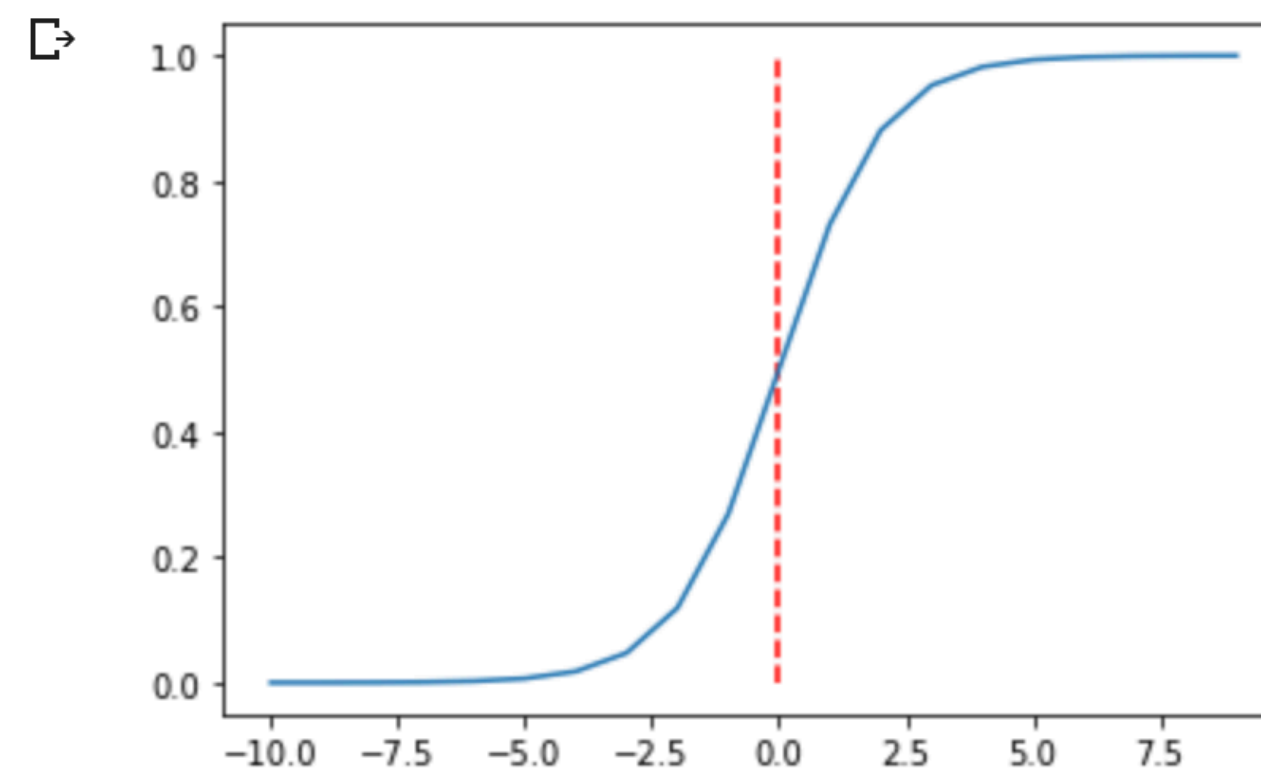
## FUNÇÃO SIGMÓIDE

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

```
[ ] # importar pacotes necessários
import numpy as np
import matplotlib.pyplot as plt

# criar um set entre -10 e 10 e aplicar a função sigmóide
x = np.arange(-10, 10)
y = 1 / (1 + np.exp(-x))

# plotar a curva sigmoidal
plt.plot(x, y)
plt.vlines(0, 0, 1, colors='r', linestyle='dashed')
plt.show()
```



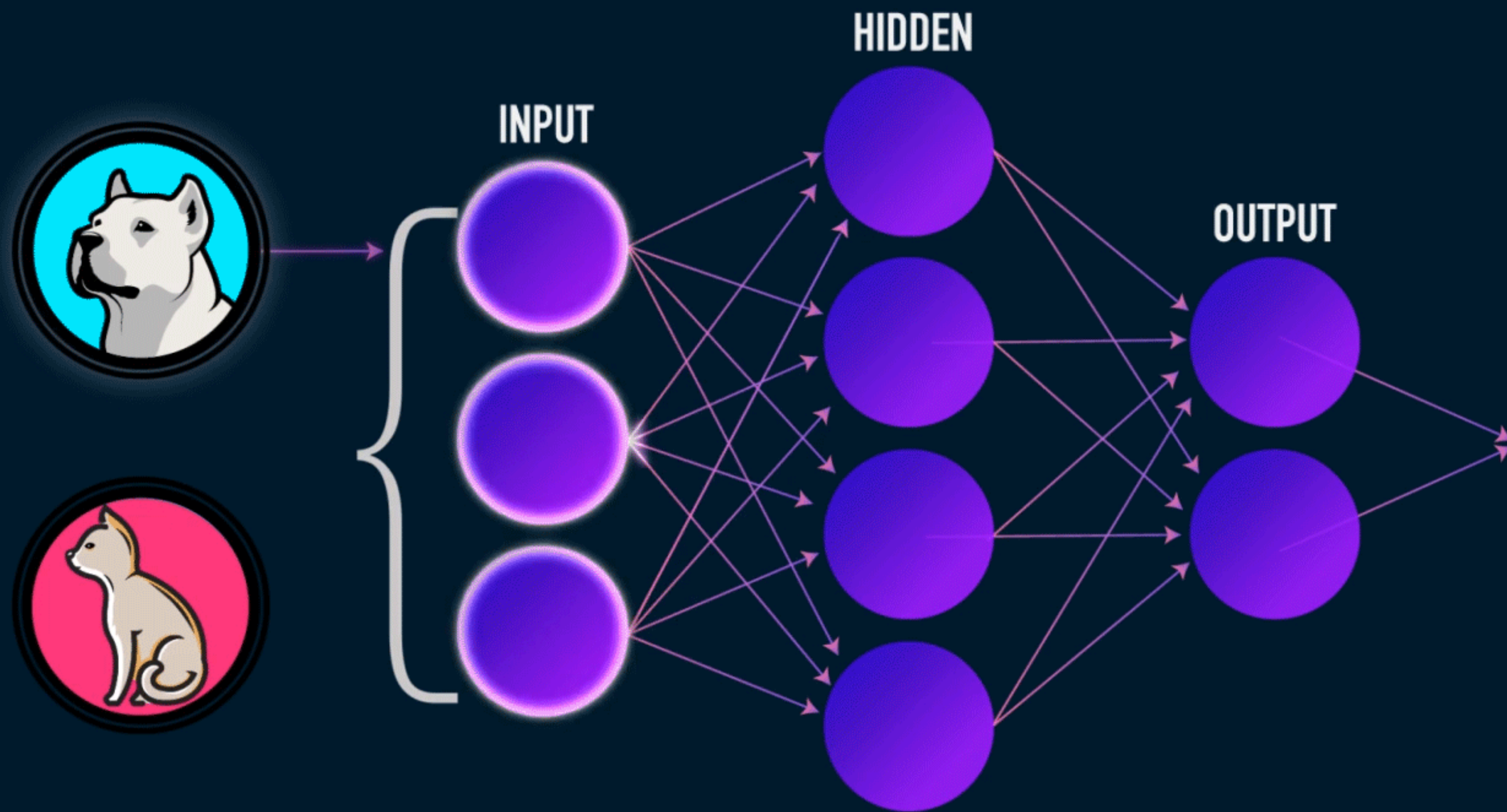


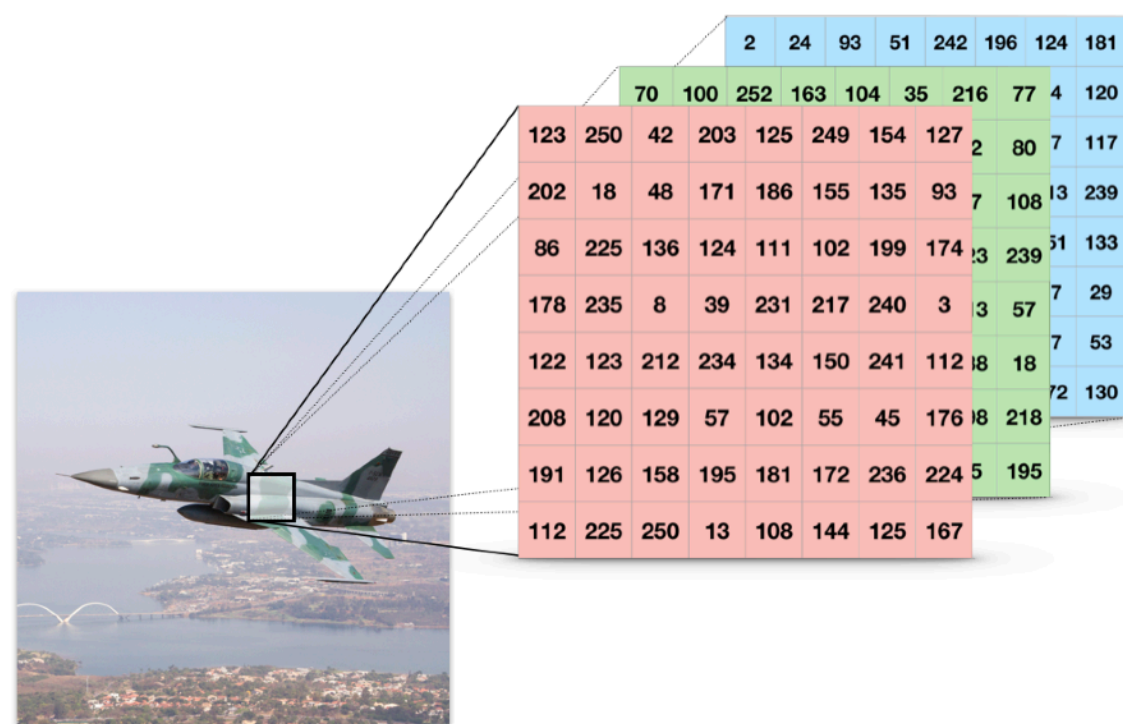
# 1. SCORE FUNCTION

*A score function é uma função que vai receber o nosso input (no exemplo usado aqui, uma imagem) e mapear os dados para as classes de labels.*

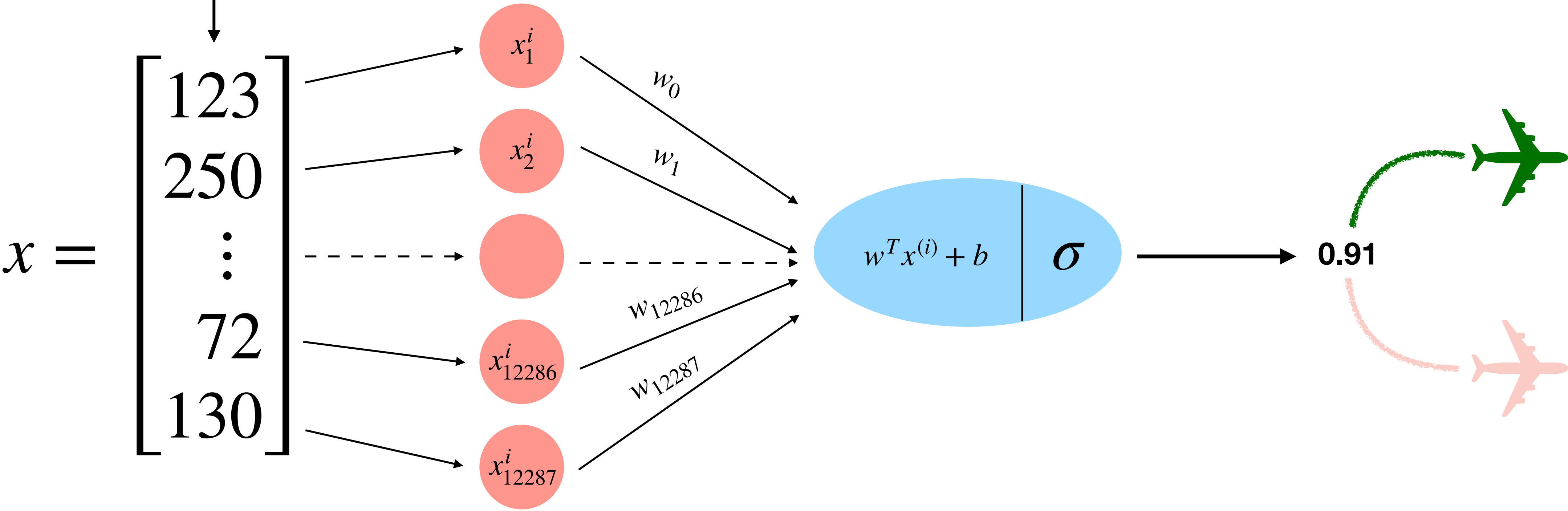
$$f(x, \theta, b) = \hat{y} = \theta^T x + b$$

$$\hat{y} = \sigma(\theta^T x + b) = \frac{1}{1 + e^{-(\theta^T x + b)}}$$

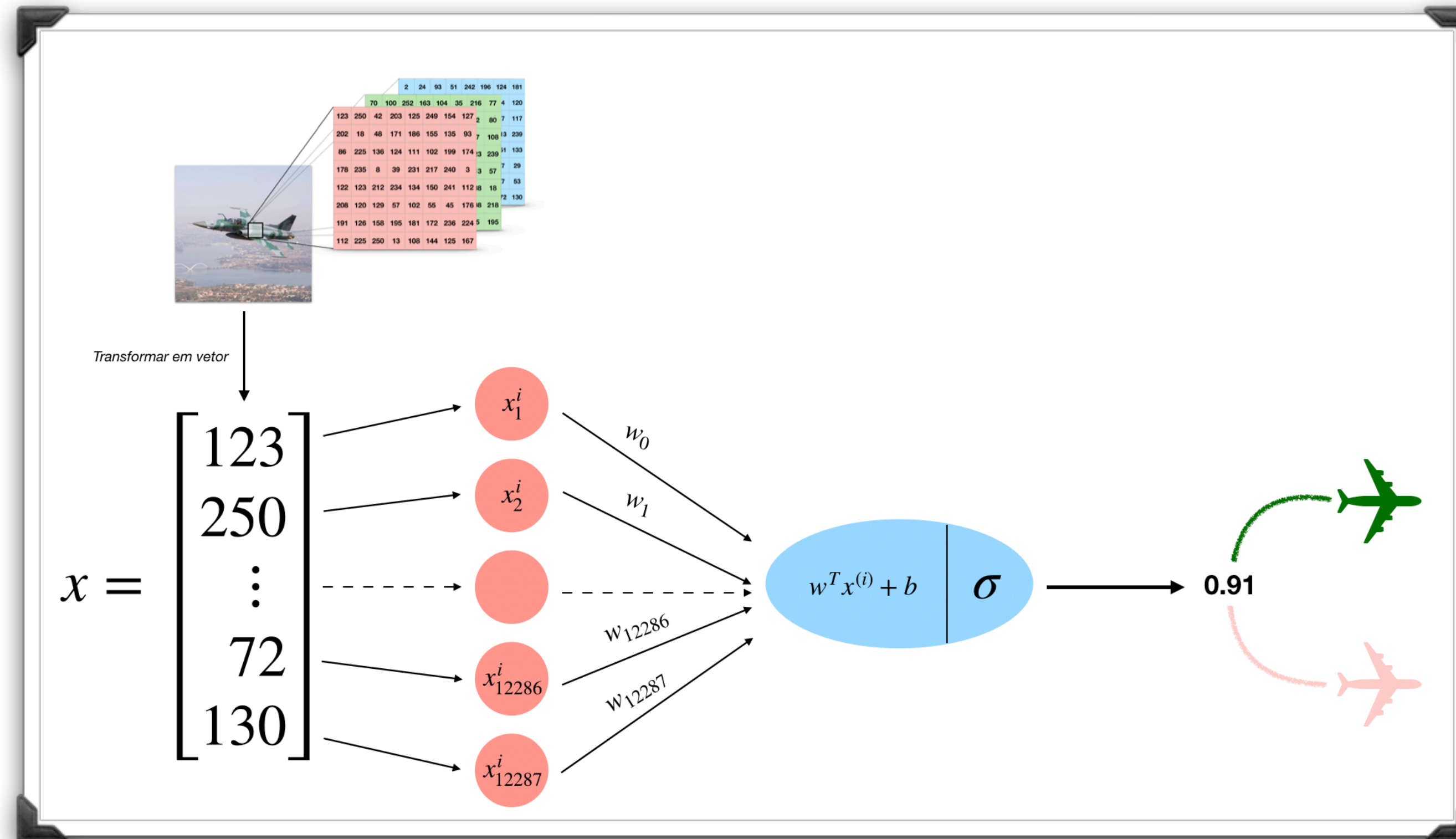




Transformar em vetor







1. Transformar a imagem em um vetor.
2. Multiplicar o valor de cada pixel de  $x$  pelo seu peso  $w$ .
3. Obter o valor de  $z$ .
4. Obter a probabilidade de ser um avião. Ou seja, obter um valor entre 0 e 1.
5. Classificar a imagem com o label “*avião*”.