



Reconhecendo Imagens com Inteligência Artificial

Construa sua Própria Rede Neural com TensorFlow

Sandro Moreira

@sandro_moreira

About Me



@sandro_moreira

Organizador do Google Developers Group Rio Verde

Organizador do TensorFlow Goiás

Doutorando em Ciência da Computação (UFG)

Mestre em Engenharia Mecânica (UNESP)

Pesquisador no Centro de Excelência em Inteligência Artificial (CEIA)

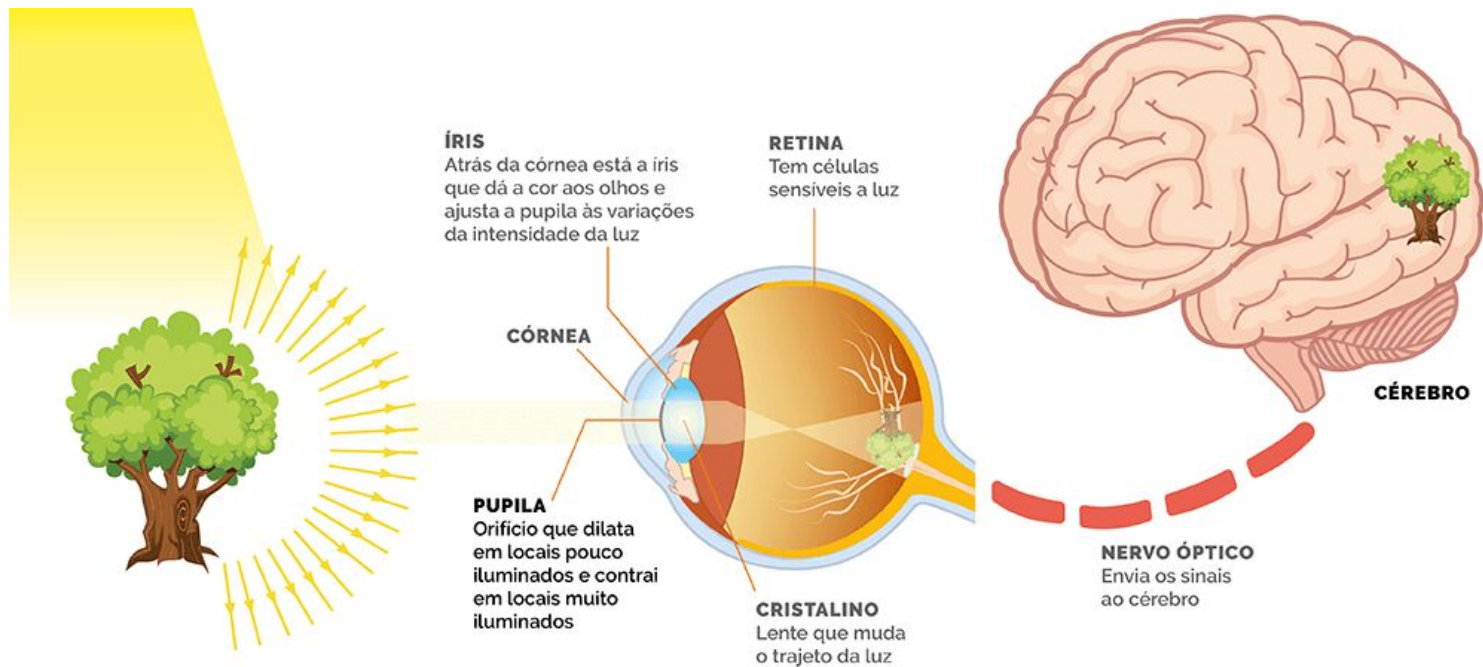
Professor na Faculdade de Engenharia de Software (UniRV)

Google Certified Educator Level 1

Mentor no Saturdays.AI (La Paz - BO)

Mentor de Machine Learning no Google for Startups

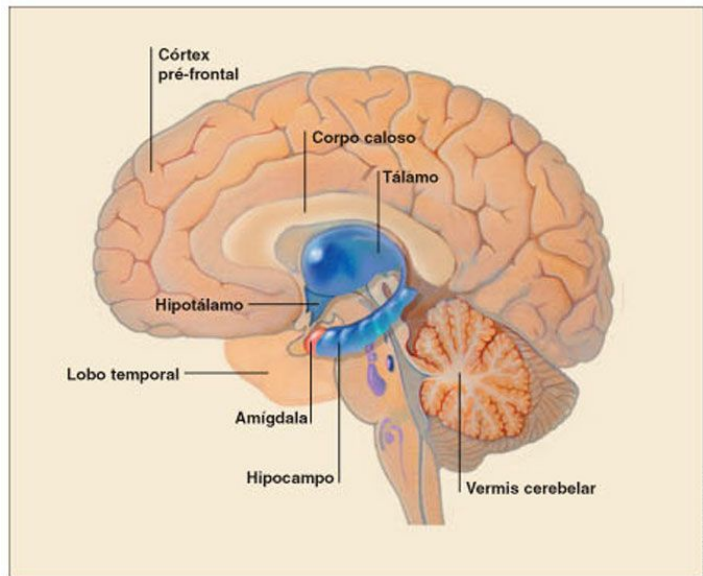
Como os humanos vêem?



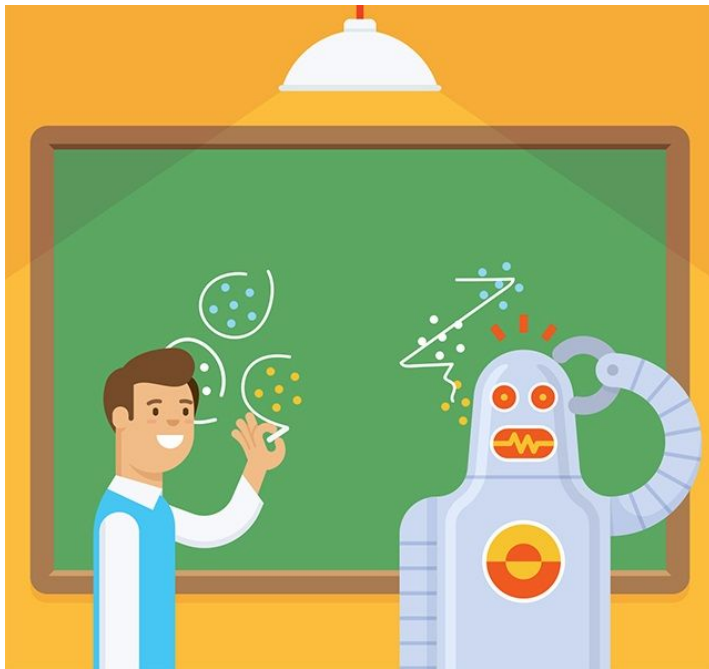
Como os humanos reconhecem?

"No cérebro, a área envolvida nesse processo é o hipocampo, que é a região que mais concentra neurônios associados à consolidação das memórias."

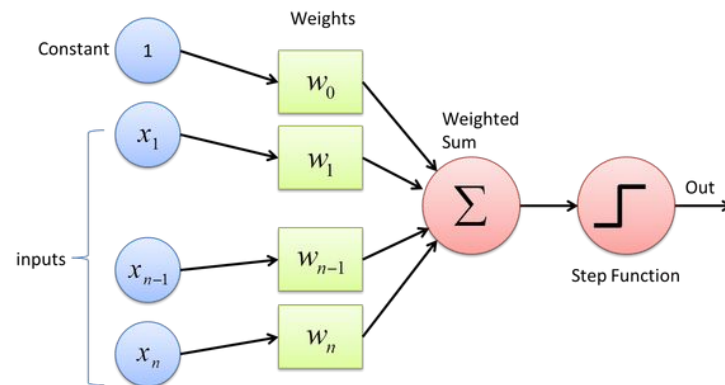
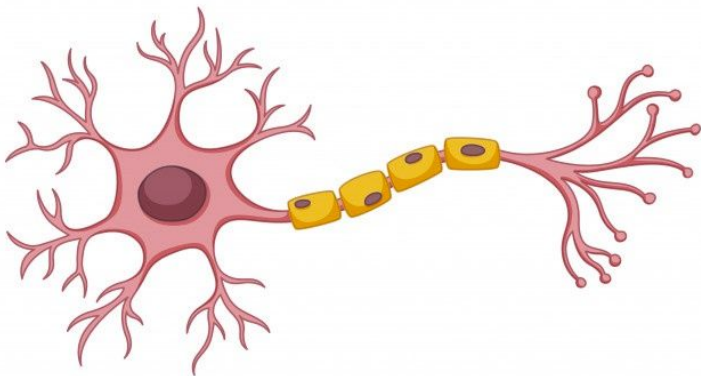
<https://www.brainlatam.com/blog/memoria-e-atencao--993>



Como ensinar uma máquina a reconhecer?

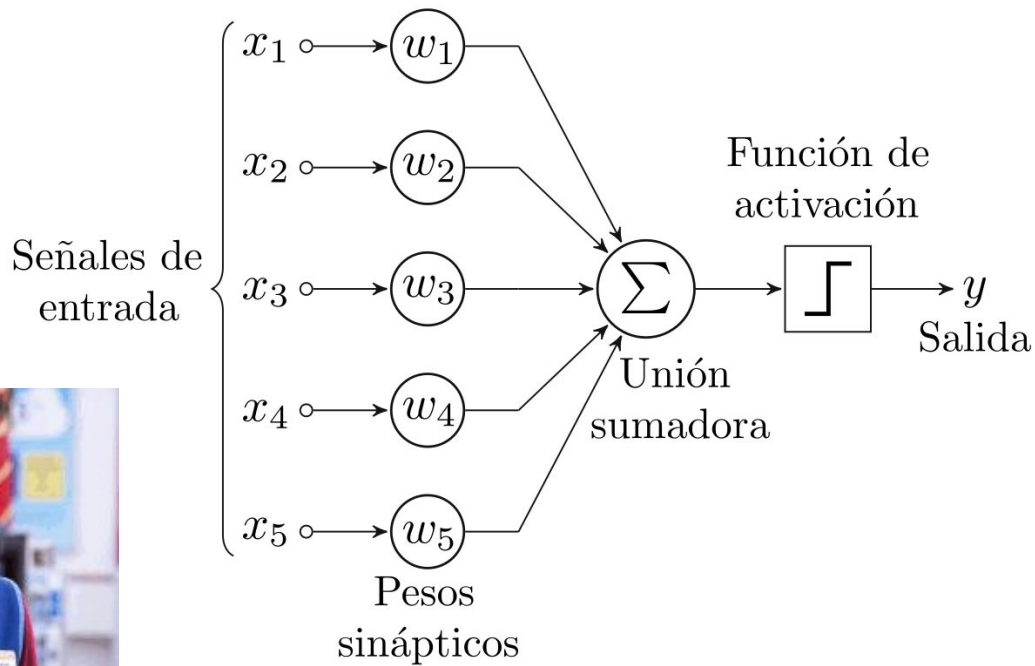


Imitando humanos...



Modelos matemáticos inspirados em Neurônios Biológicos

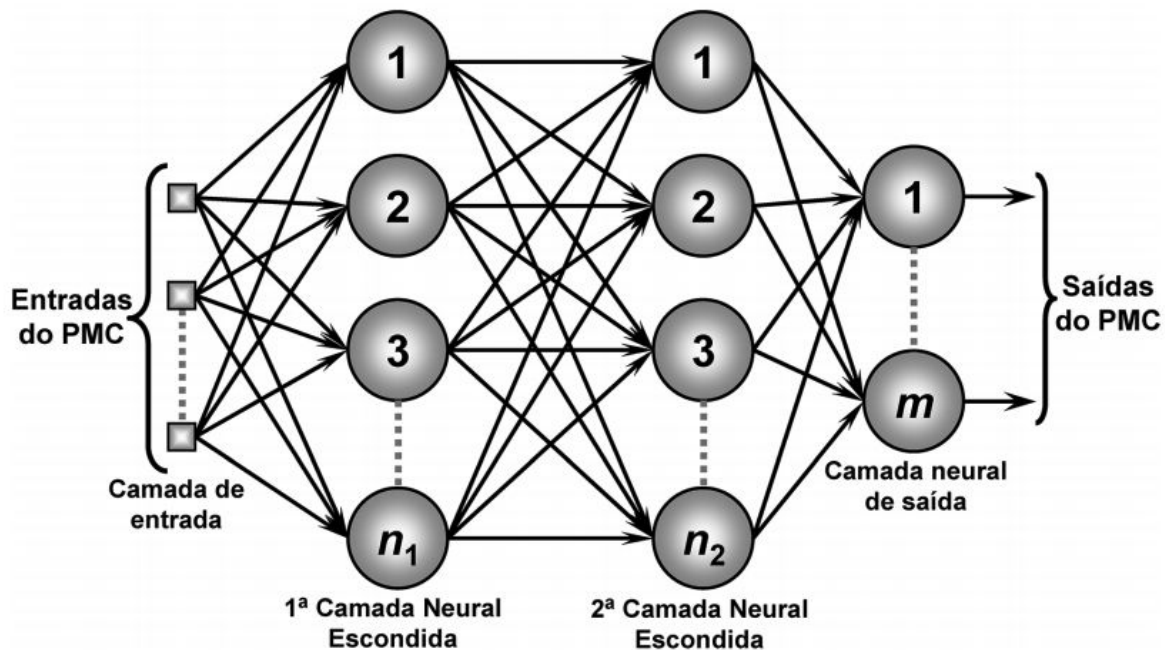
Basicamente, um neurônio artificial



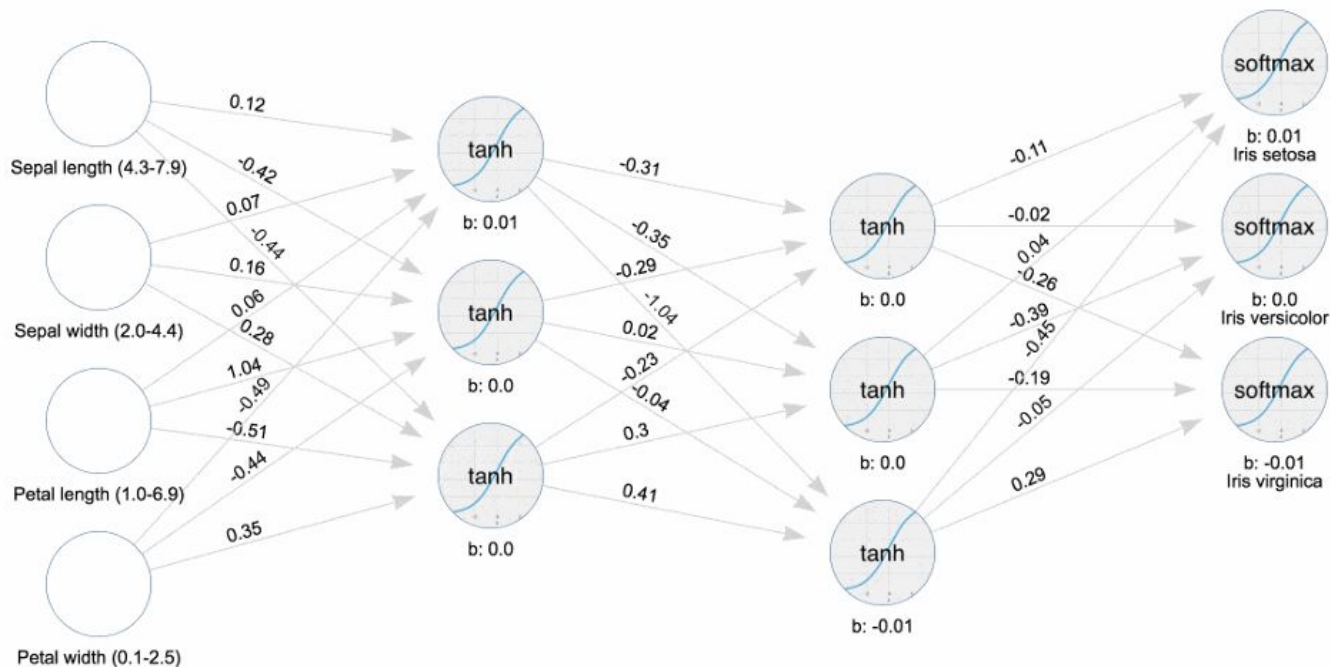
$$\begin{cases} u = \sum_{i=1}^n w_i \cdot x_i - \theta \\ y = g(u) \end{cases}$$



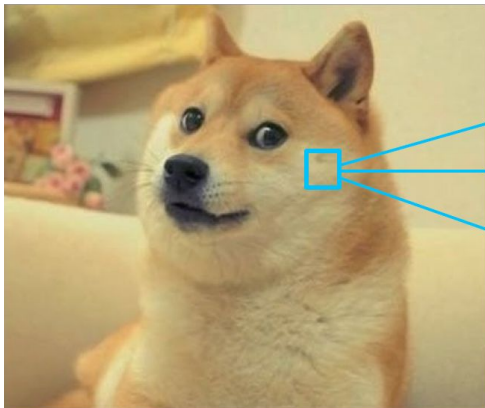
Precisamos de uma rede de neurônios



Precisamos de uma rede de neurônios



Uma imagem é uma matriz



R
[[186 233]
[91 162]]

G
[[109 201]
[65 132]]

B
[[123 144]
[13 128]]

Precisamos de uma rede de neurônios

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} * \begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix} + \begin{bmatrix} 4 & 3 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 8 & 7 \\ 10 & 9 \end{bmatrix}$$

entradas * pesos + bias = prediction

Precisamos de uma rede de neurônios

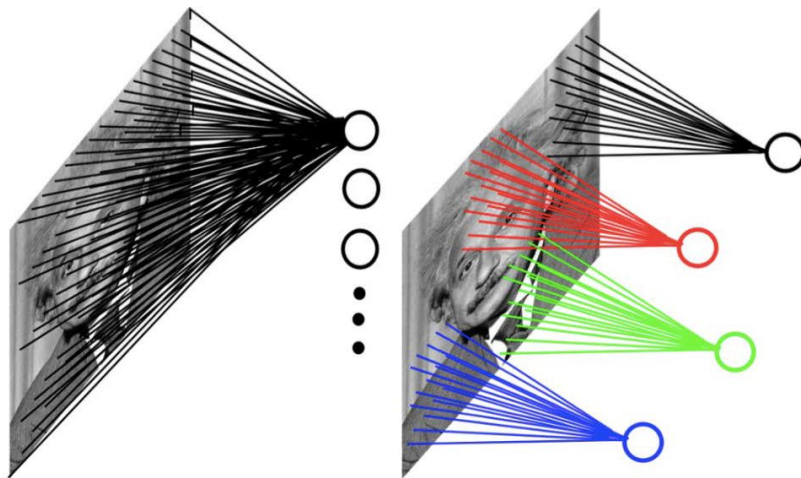
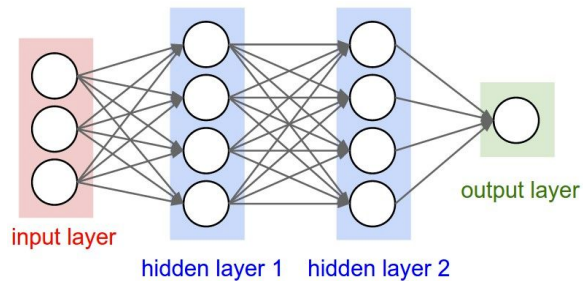
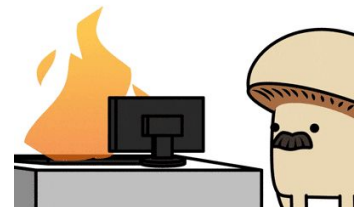
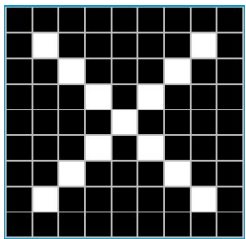


Imagem com 200x200px

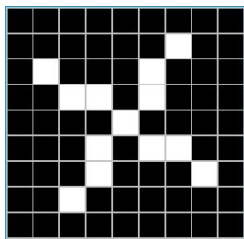
$40.000 \times 4 = 160$ mil neurônios cada um com seus parâmetros



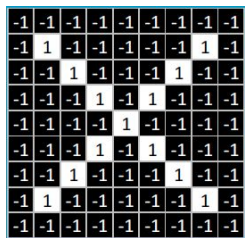
E as dificuldades não param por aí...



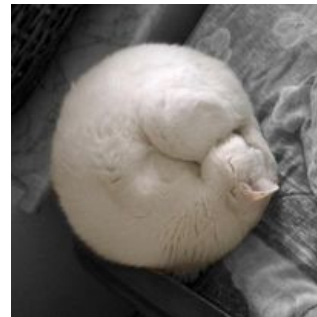
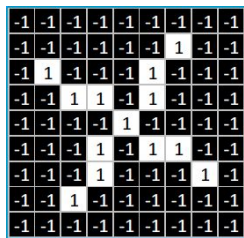
=



?



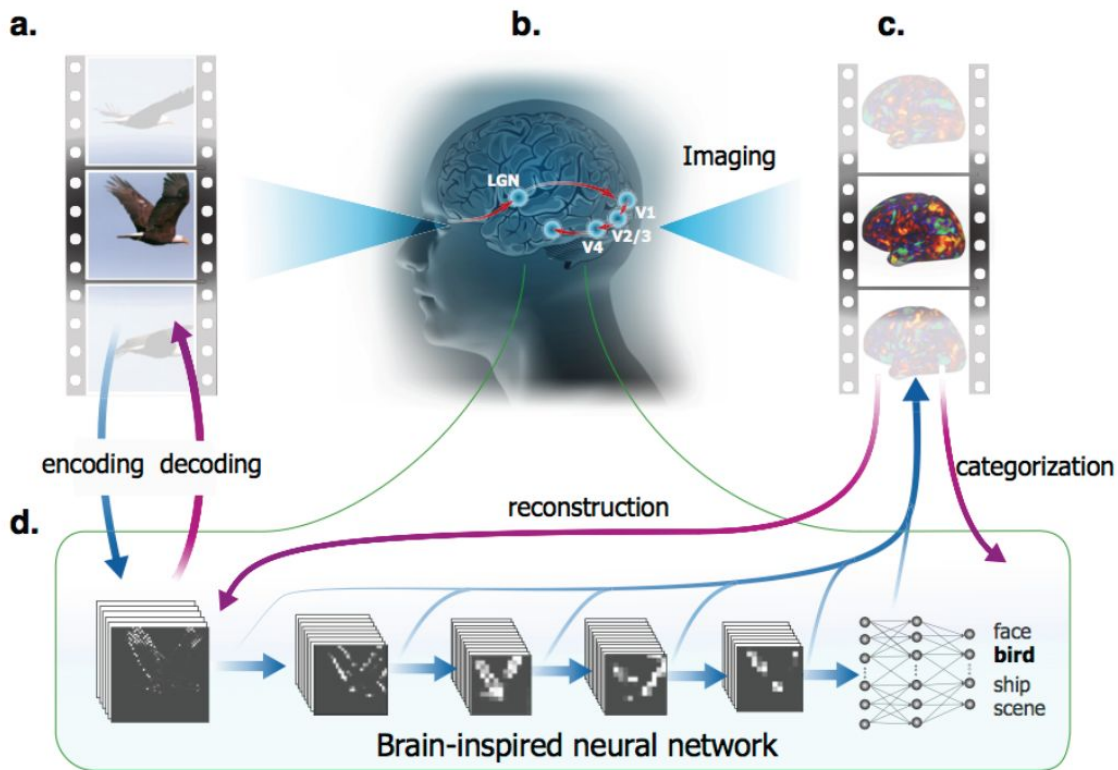
≠



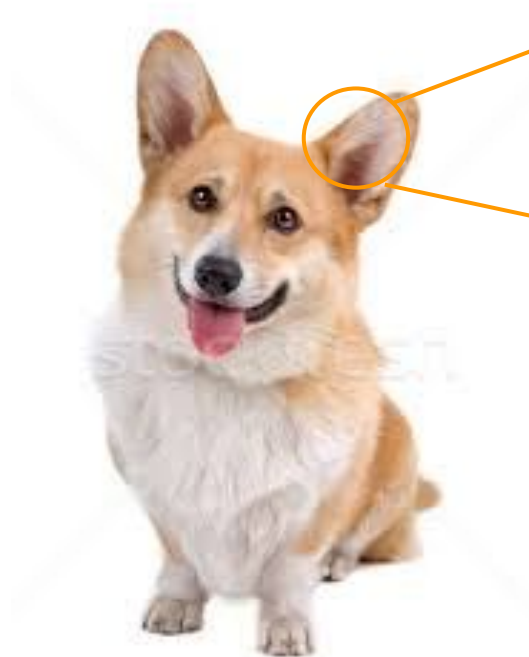
Redes Neurais Convolucionais

Convolution Neural Networks

Inspiradas no Cortex Visual Humano



Redes Convolucionais



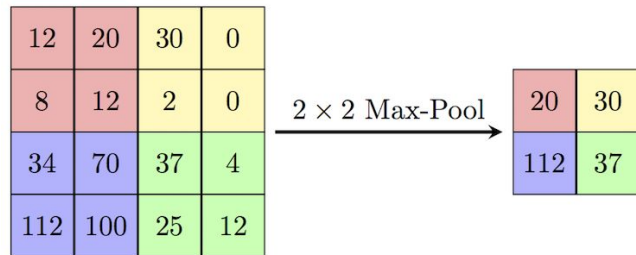
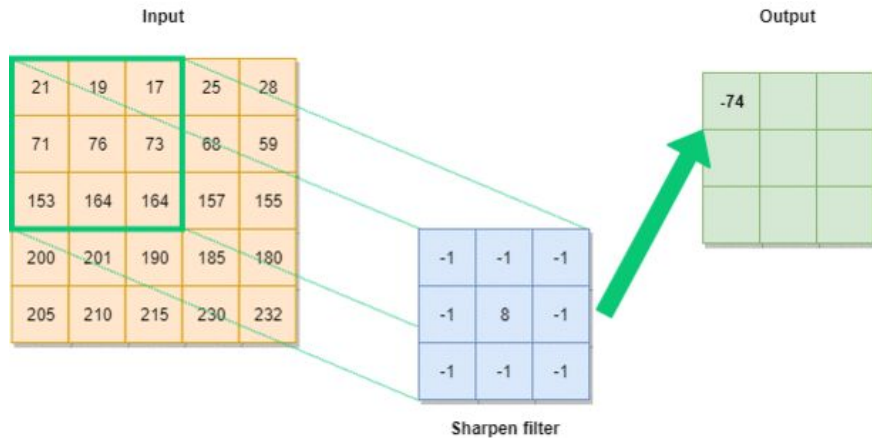
0	64	128
48	192	144
142	226	168

O valor atual do pixel é 192
Considera valores ao redor

-1	0	-2
.5	4.5	-1.5
1.5	2	-3

Filtro Convolutacional

Filtros Convolucionais e Pooling

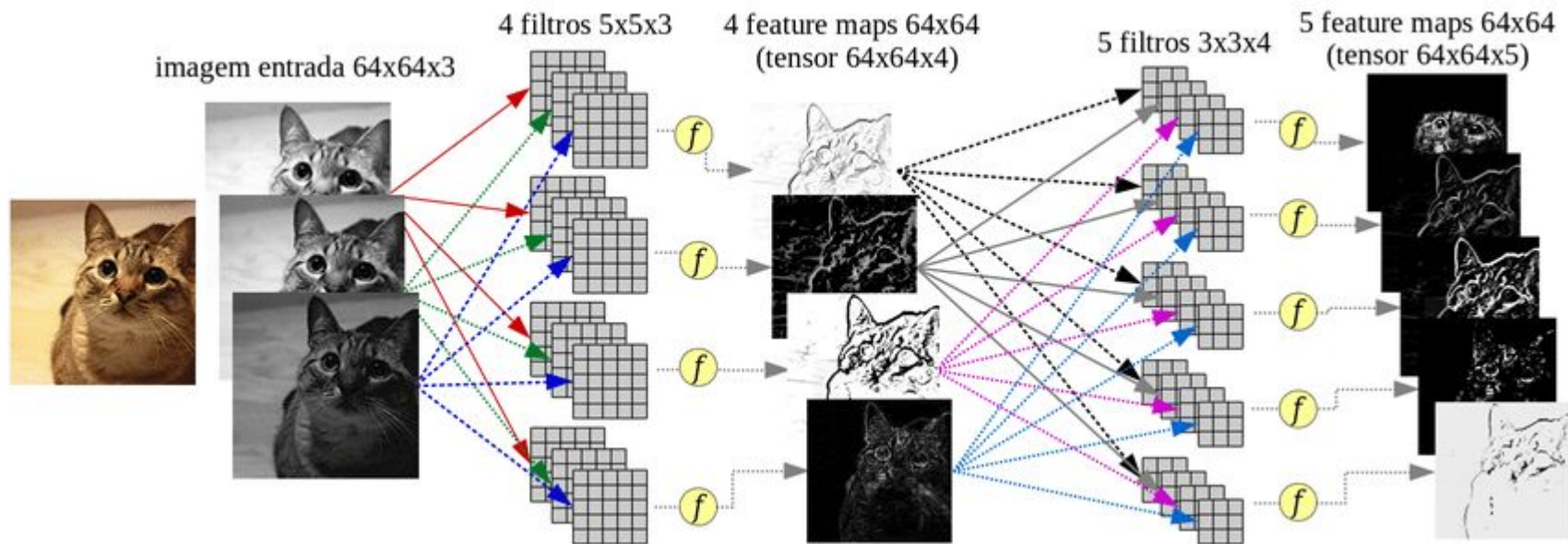


Filtros Convolucionais

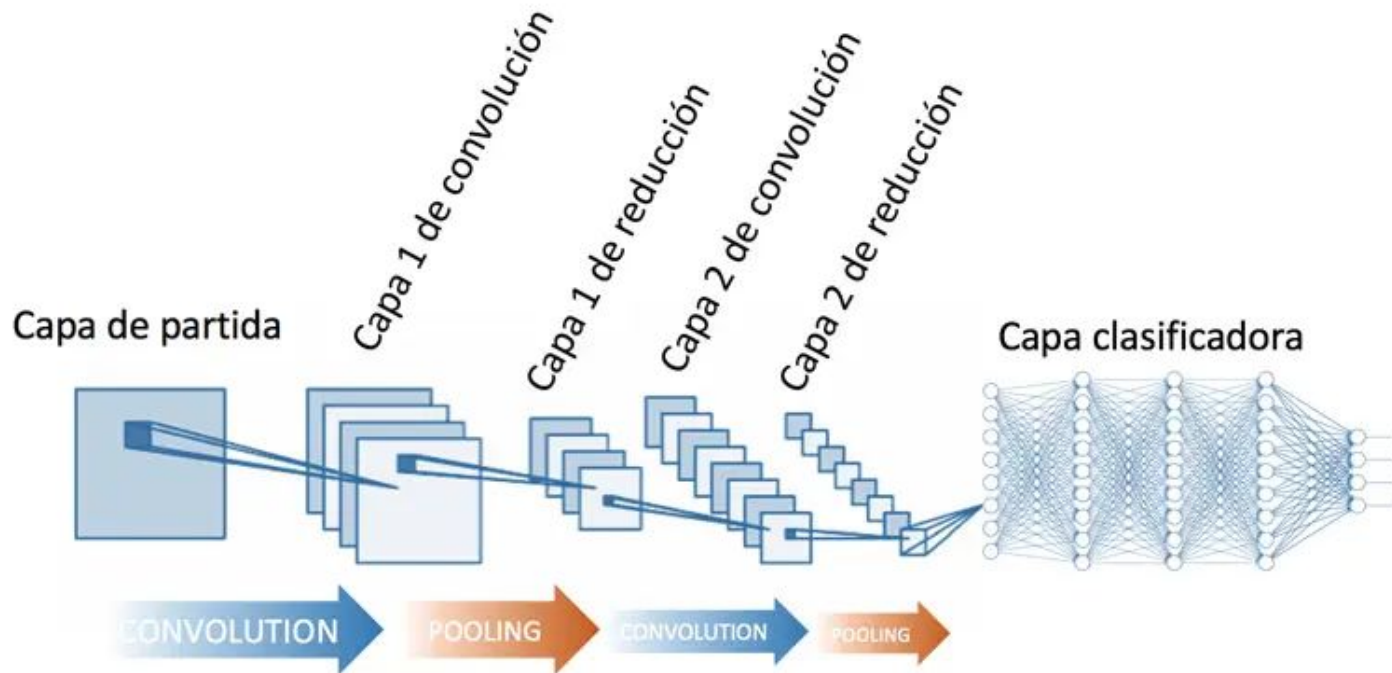


Input

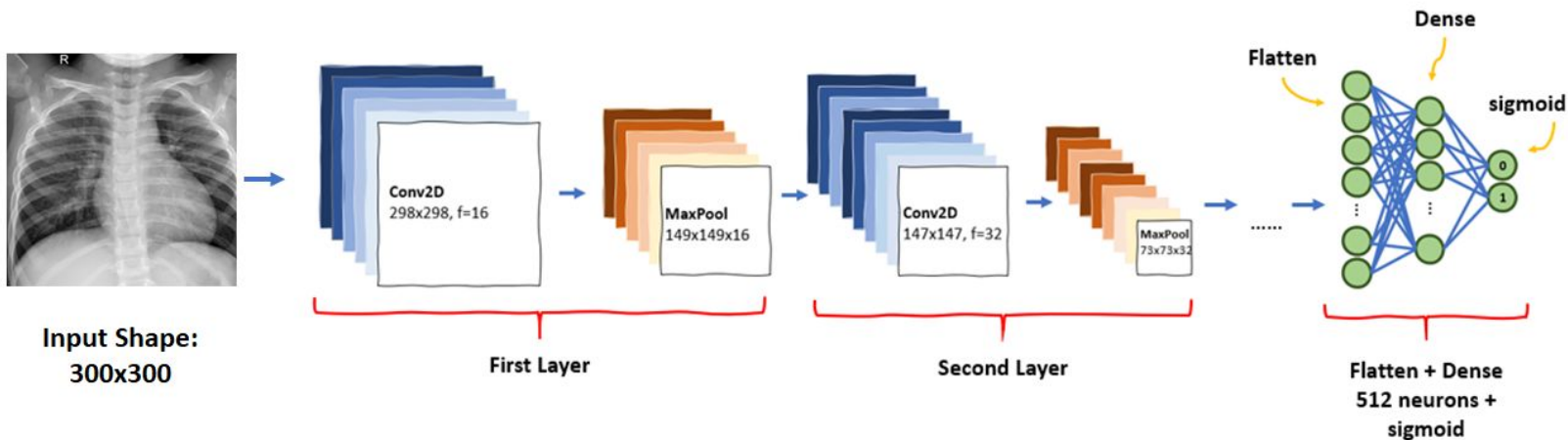
Um imagem, múltiplos mapas de características



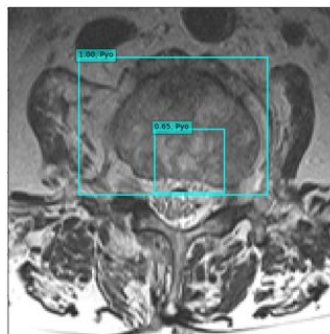
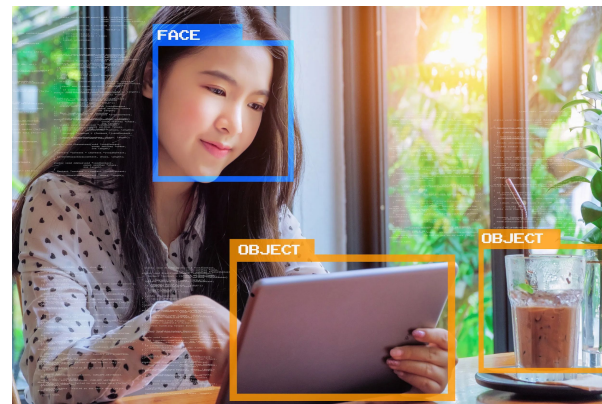
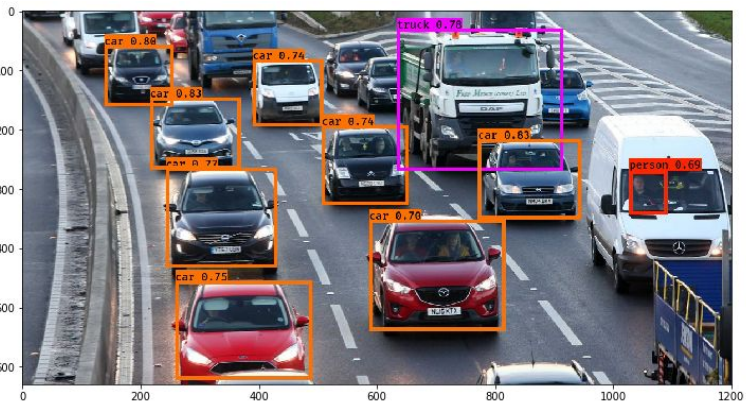
Um imagem, múltiplos mapas de características



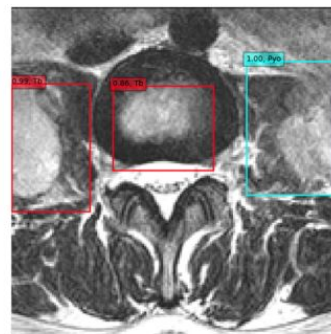
Aplicações de uma CNN



Aplicações de uma CNN



(a)



(b)

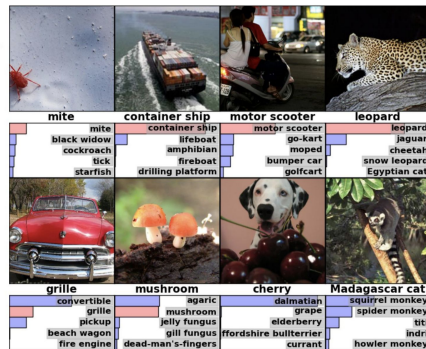
Uma CNN com TensorFlow

```
model = tf.keras.model.Sequential([  
    tf.keras.layers.Conv2D(64, (3,3), activation='relu', input_shape=(28,28,1)),  
    tf.keras.layers.MaxPooling2D(2,2),  
    tf.keras.layers.Flatten(),  
    tf.keras.layers.Dense(128, activation=tf.nn.relu),  
    tf.keras.layers.Dense(10, activation=tf.nn.softmax)  
])
```

Redes Pré-Treinadas e a Transferência de Aprendizado

IMAGENET

- 1,000 object classes (categories).
- Images:
 - 1.2 M train
 - 100k test.

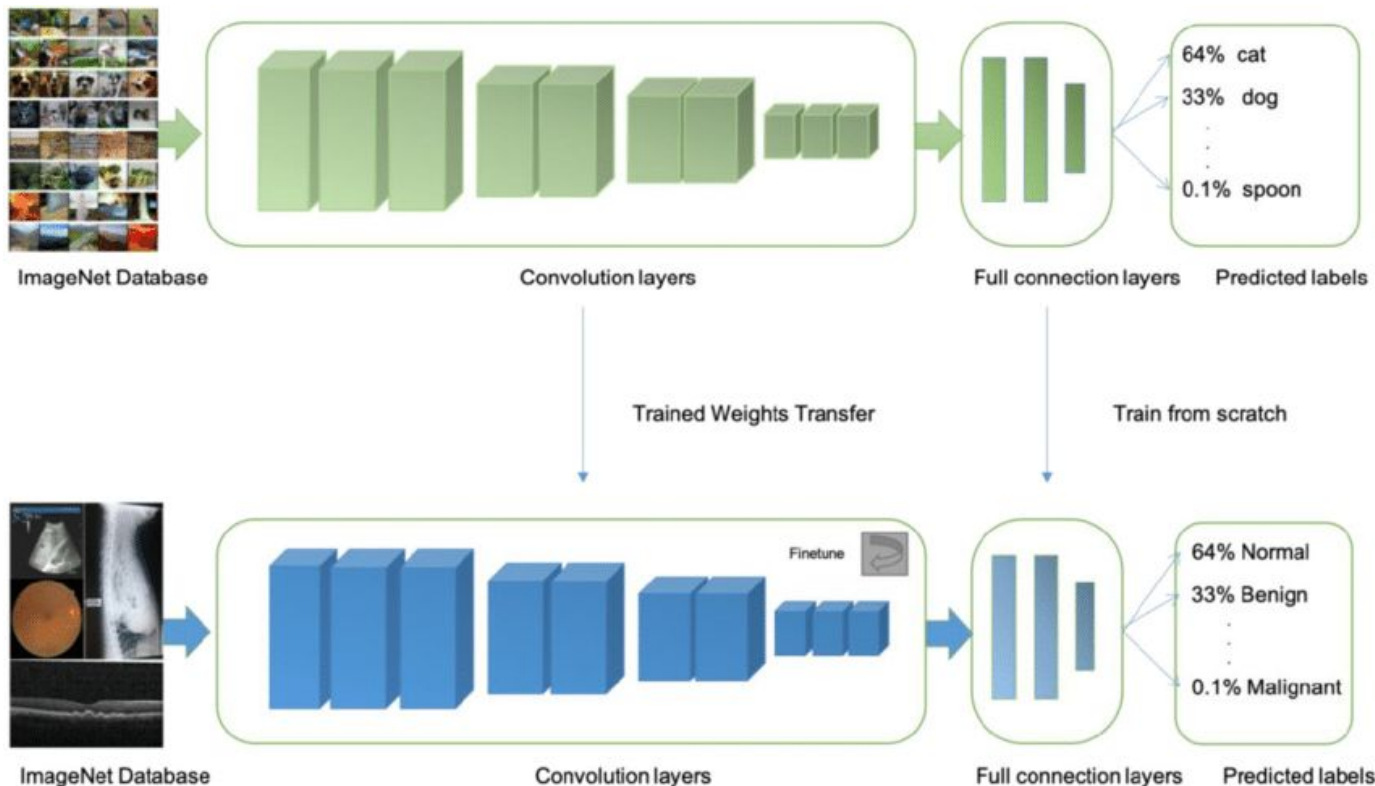


Treinar uma CNN pode ser algo demorado

Uma arquitetura de rede muito eficiente descoberta em 2017, chamada de **Xception**, precisou de **60 GPUs Nvidia K80 em paralelo** para conseguir ser treinada no dataset **Imagenet** a uma taxa de 28 steps por segundo.

Treinar redes assim muito frequentemente pode se tornar algo problemático nos ambientes de produção que precisam de deployments ágeis.

Redes Pré-Treinadas e a Transferência de Aprendizado



Let's Code...



colab



TensorFlow



python™



Links Recomendados

- <https://ai.google>
- <http://playground.tensorflow.org>
- <https://www.tensorflow.org/guide>
- <https://keras.io>
- <http://deeplearning.stanford.edu/tutorial>
- <http://www.saturdays.ai>
- Deep Learning Brasil - <https://www.facebook.com/groups/333175140356771/>

<https://www.youtube.com/sandromoreirago>

Dúvidas?



Sandro Moreira

@sandro_moreira

moreira.sandro@gmail.com

Esta apresentação e notebook Colab produzidos nesse workshop encontram-se disponíveis em: github.com/smoreira

