

# Redes Generativas

## Algoritmos de

### *Deep Learning*



**Prof. Dr. Diego Renan Bruno**

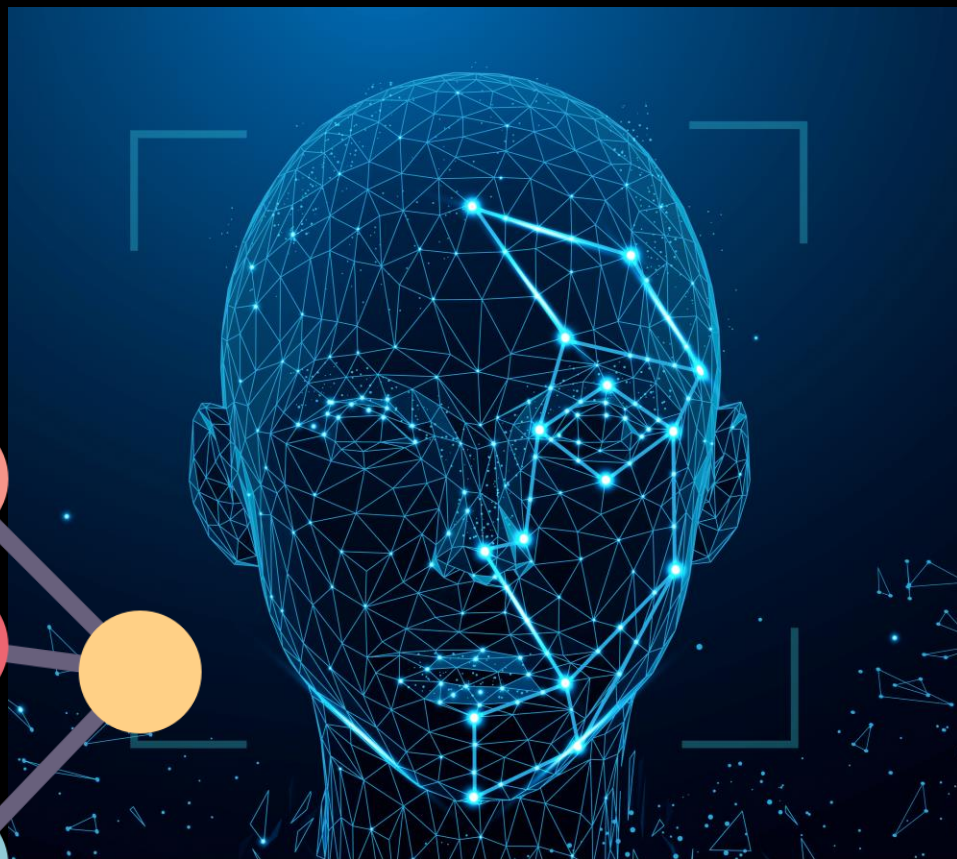
*Education Tech Lead na DIO*

*Doutor em Robótica e Machine Learning pelo ICMC-USP*



# Redes Generativas

*Machine Learning*



# Trabalhos Realizados



**Laboratório de Robótica Móvel**  
ICMC/USP - São Carlos

## CARINA 1



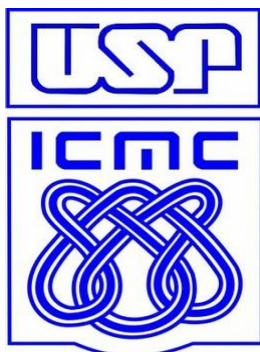
## CARINA 2



# Trabalhos Realizados



**Laboratório de Robótica Móvel**  
ICMC/USP - São Carlos

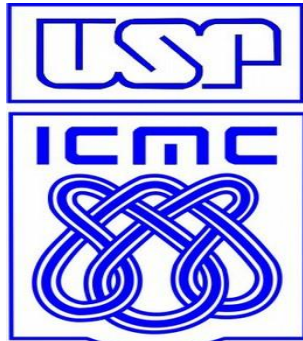




# Trabalhos Realizados



**Laboratório de Robótica Móvel**  
ICMC/USP - São Carlos



**SCANIA**



# O mundo da IA...

IA Geral



IA Restrita



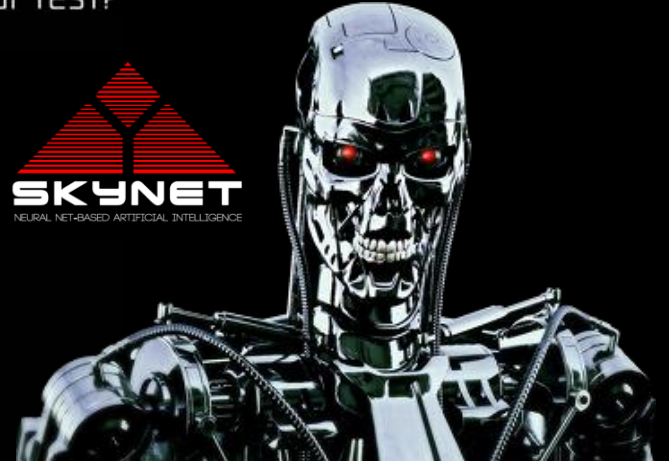
*Machine Learning*

ex machina

WHAT HAPPENS TO ME IF I FAIL YOUR TEST?



ARTIFICIAL INTELLIGENCE



# O que é Visão Computacional?



Sensoriamento:  
Imagens



Processamento  
De Imagens

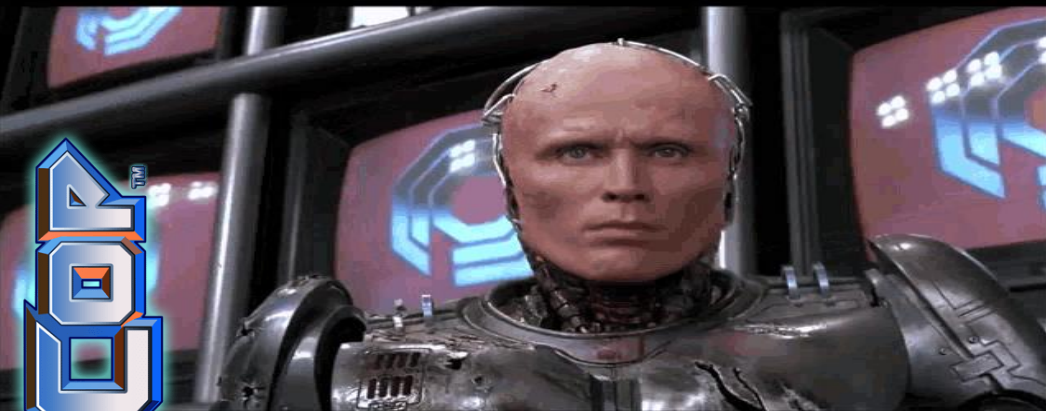


Análise: *Machine Learning*



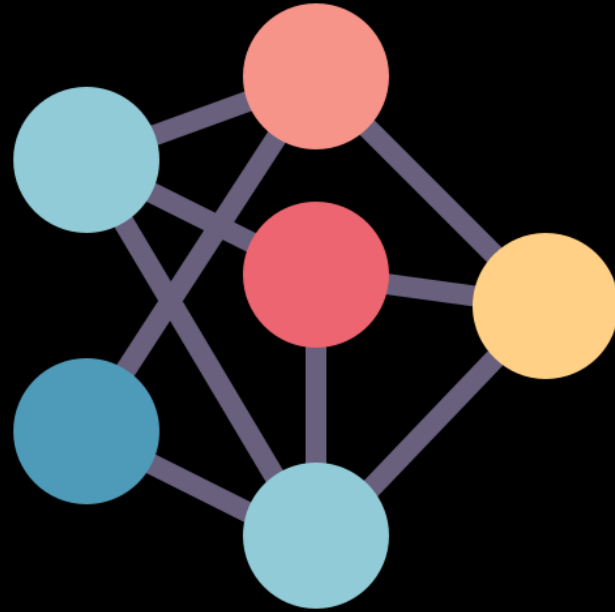


# Reconhecimento de Pessoas



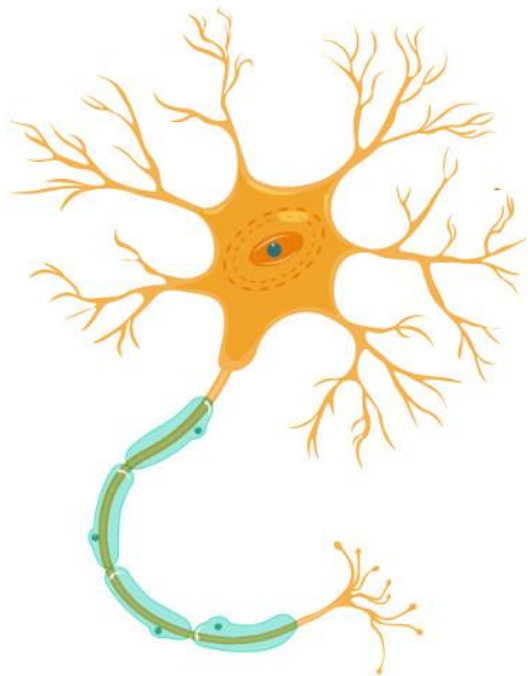


# *Redes Neurais Artificiais*

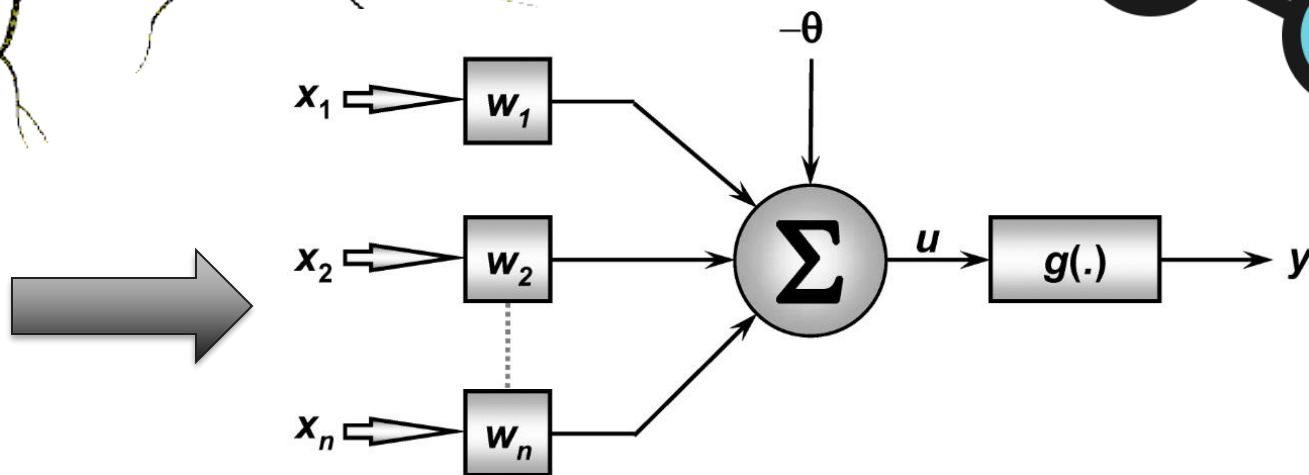
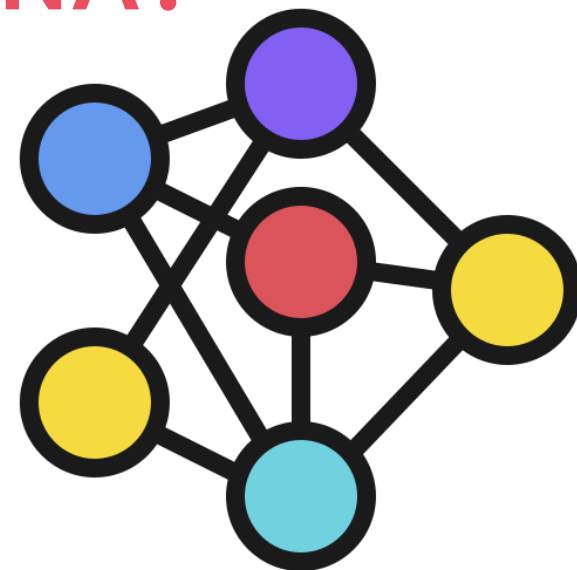
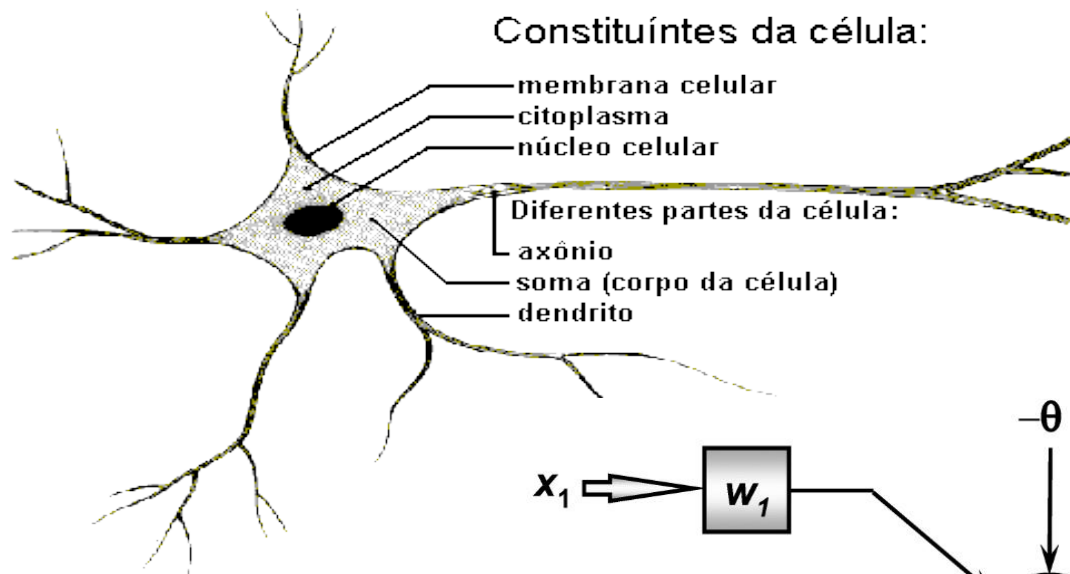


# O que são Redes Neurais?

## Redes Neurais

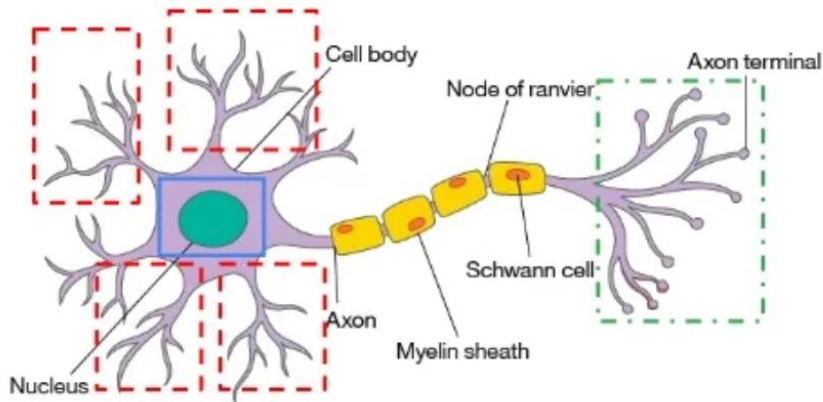


# Qual a estrutura de uma RNA?





# Redes Biológicas x Artificiais



neurônio



dendritos / pesos



núcleo / unidade



axônio+sinapse / saída

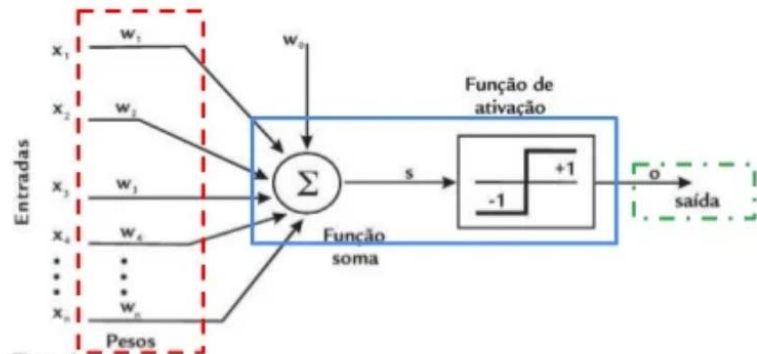
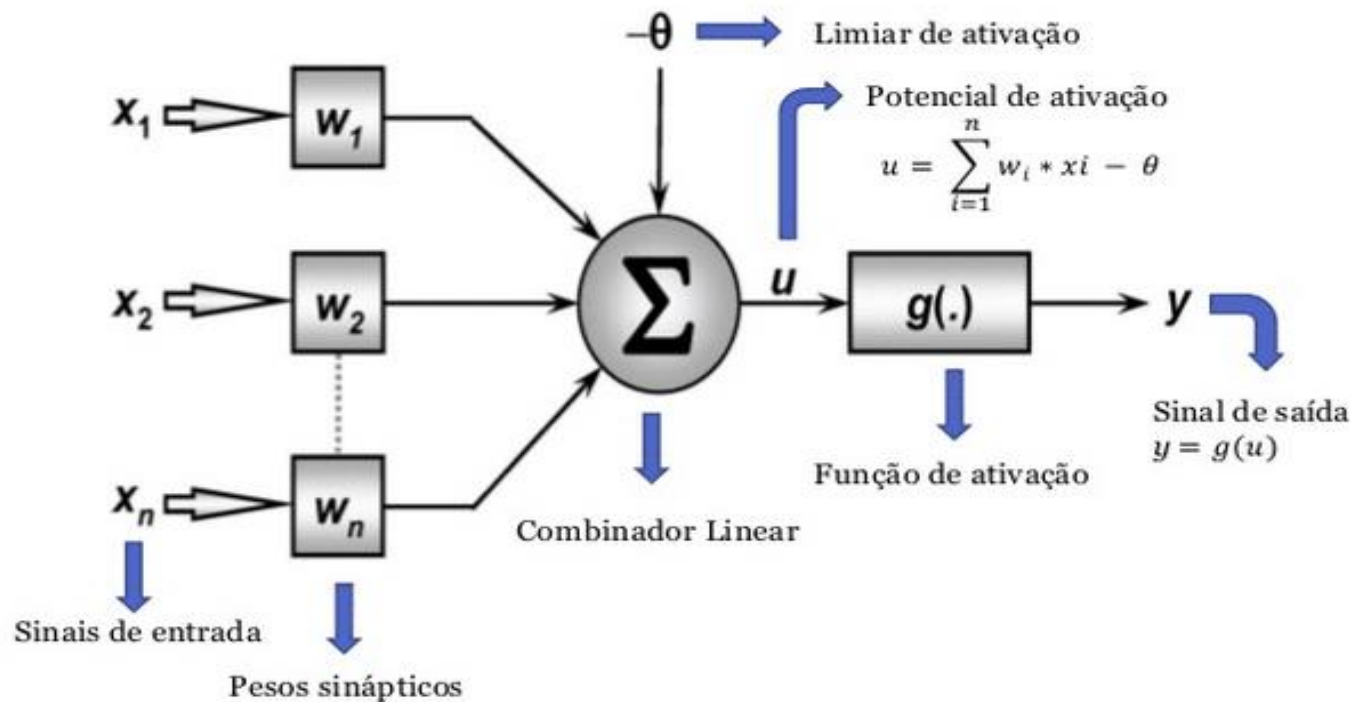


Figura 1

Modelo de um neurônio perceptron de Rosenblatt. Fonte: Adaptado de Medeiros (2006, p. 3).

neurônio artificial

# Neurônio Artificial

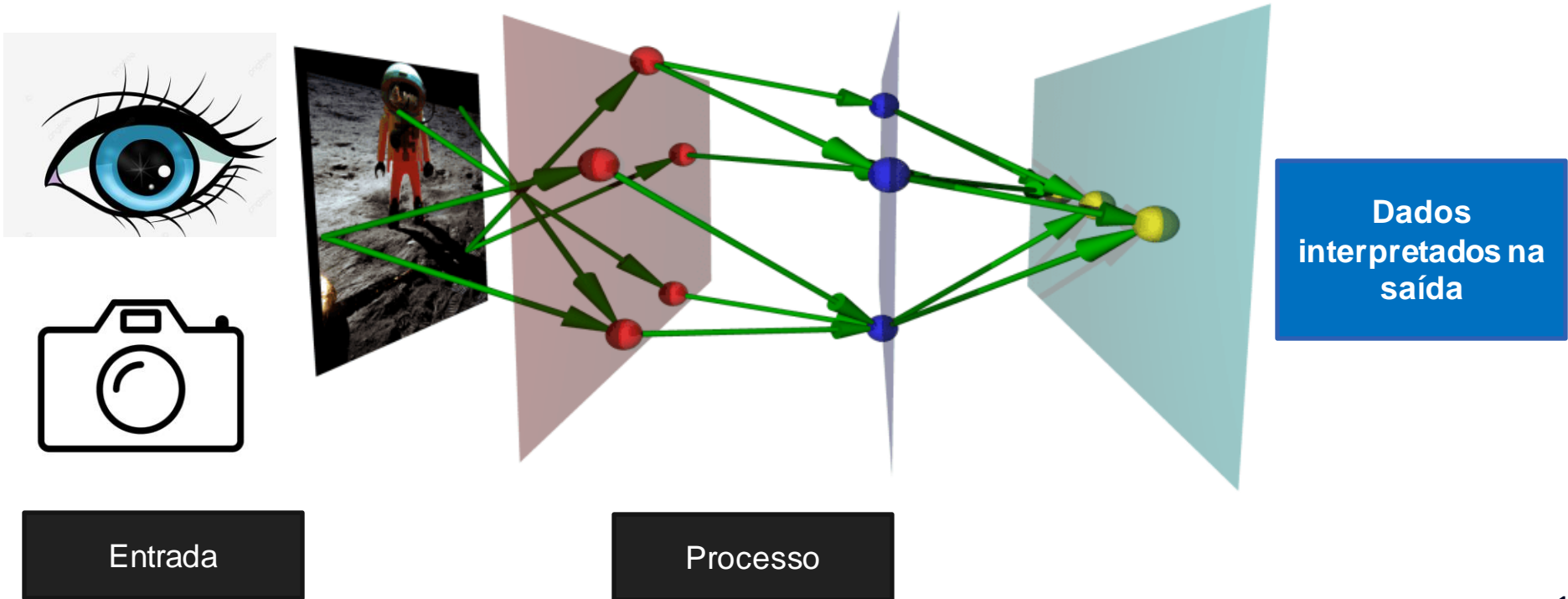


# Dados de entrada e saída





# Redes Neurais Biológicas x Artificiais



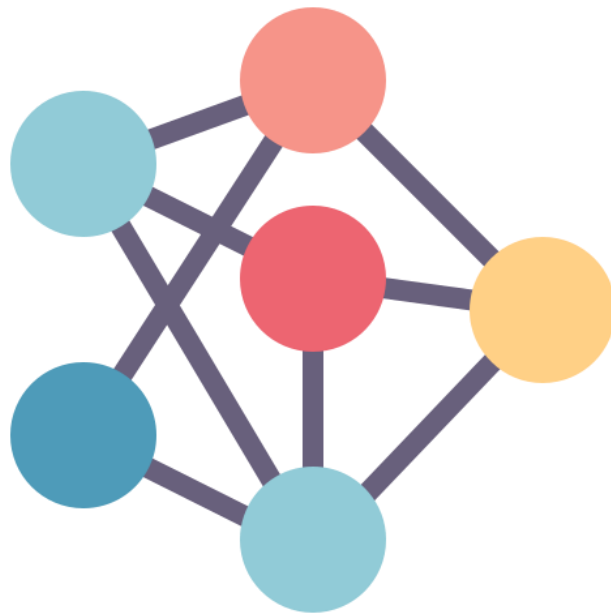
# Relação de entrada e saída



Imagem de Entrada

08	02	22	97	38	15	00	40	00	75	04	05	07	78	52	12	50	77	01	28
49	49	99	40	17	81	18	57	60	87	17	40	98	43	69	45	58	56	62	00
81	49	31	73	55	79	14	29	93	71	40	67	58	58	30	03	49	13	36	65
82	70	95	23	04	60	11	42	69	24	68	56	01	32	56	71	37	02	36	91
22	31	16	71	51	67	85	89	41	92	36	54	22	40	40	28	66	33	13	80
24	47	73	80	99	03	45	02	44	75	33	53	78	36	84	20	35	17	12	50
52	95	81	28	64	23	67	10	26	38	40	67	59	54	70	66	18	38	64	70
67	26	20	68	02	62	12	20	95	63	94	39	63	08	40	91	66	49	94	21
24	35	58	05	66	73	99	26	97	17	78	78	96	83	14	88	34	89	63	72
21	36	23	09	75	00	76	44	20	45	35	14	00	61	33	97	34	31	33	95
78	17	53	28	22	75	31	67	15	94	03	80	04	62	16	14	09	53	56	92
16	39	05	42	96	35	31	47	55	58	88	24	00	17	54	24	36	29	85	57
86	56	00	48	35	71	89	07	05	44	44	37	44	60	21	58	51	54	17	58
19	80	81	68	05	94	47	69	28	73	92	13	86	52	17	77	04	89	55	40
04	52	08	83	97	35	99	16	07	97	57	32	16	26	26	79	33	27	98	66
65	56	68	87	57	62	20	72	03	46	33	67	46	55	12	32	63	93	53	69
04	42	16	73	58	35	39	11	24	94	72	18	08	46	29	32	40	62	76	36
20	69	36	41	72	30	23	88	34	68	69	82	67	59	85	74	04	36	16	
20	73	35	29	78	31	90	01	74	31	49	71	48	58	81	16	23	57	05	54
01	70	54	71	83	51	54	69	16	92	33	48	61	43	52	01	89	27	67	48

Dados gerados



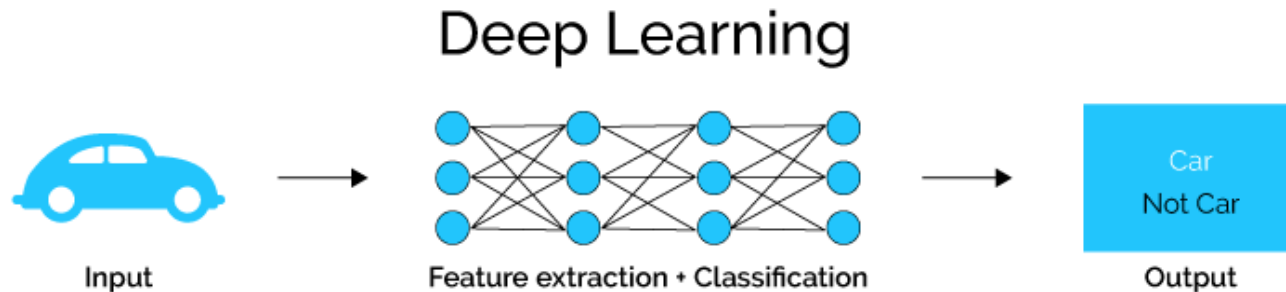
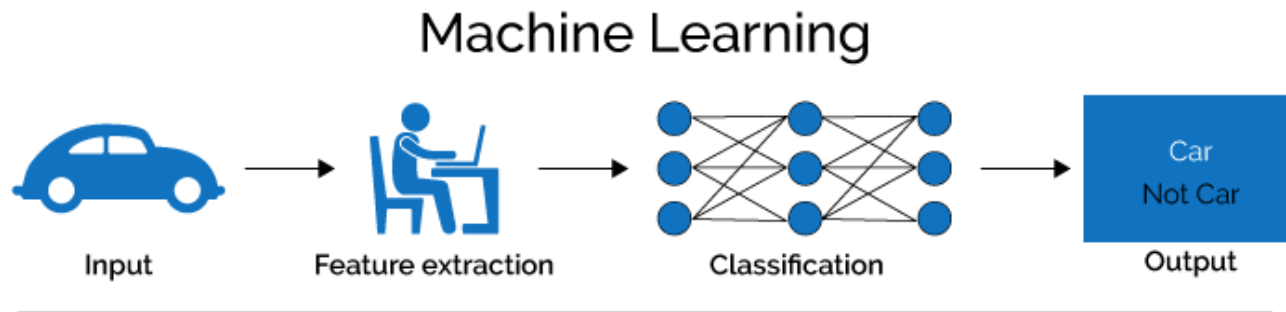
# Análise de Características (*Features*)



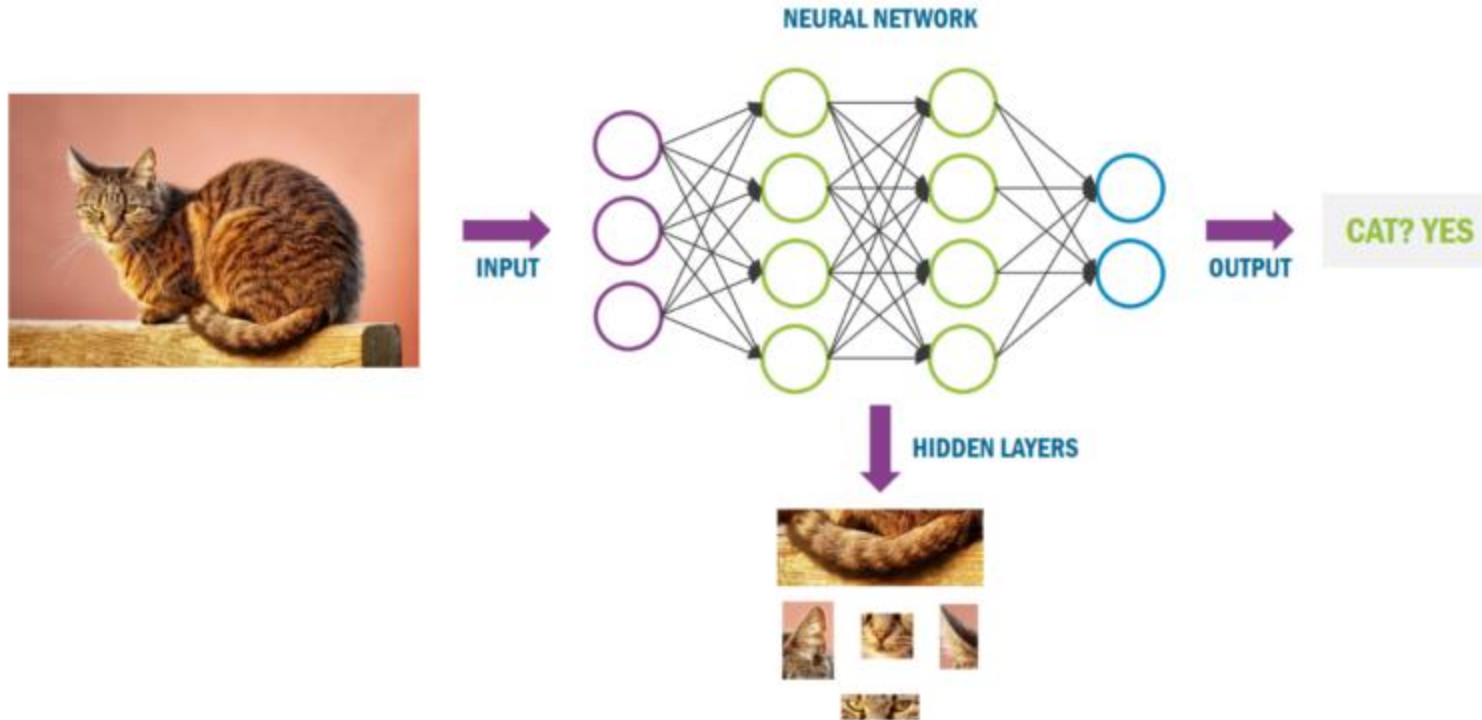


# Diferenças entre as redes *Deep*...

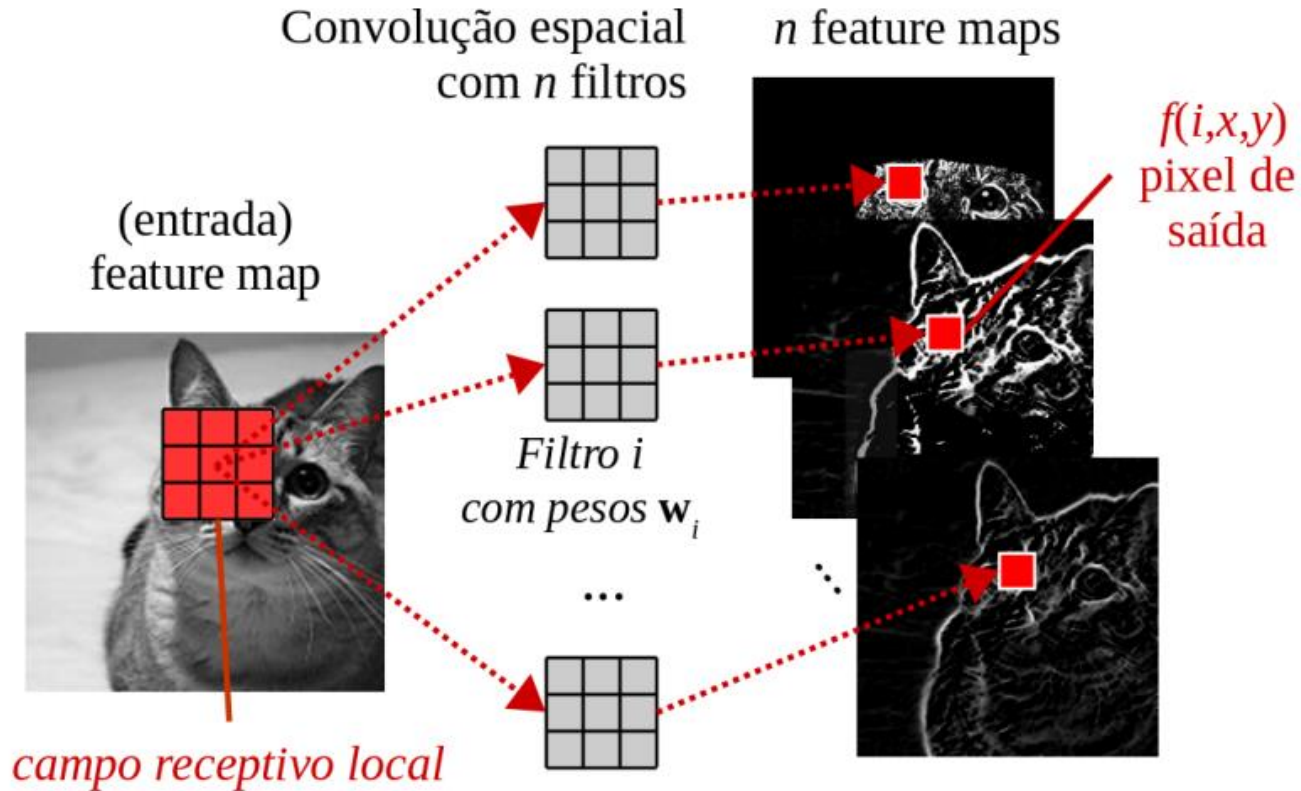
→ Extração de *Features*:



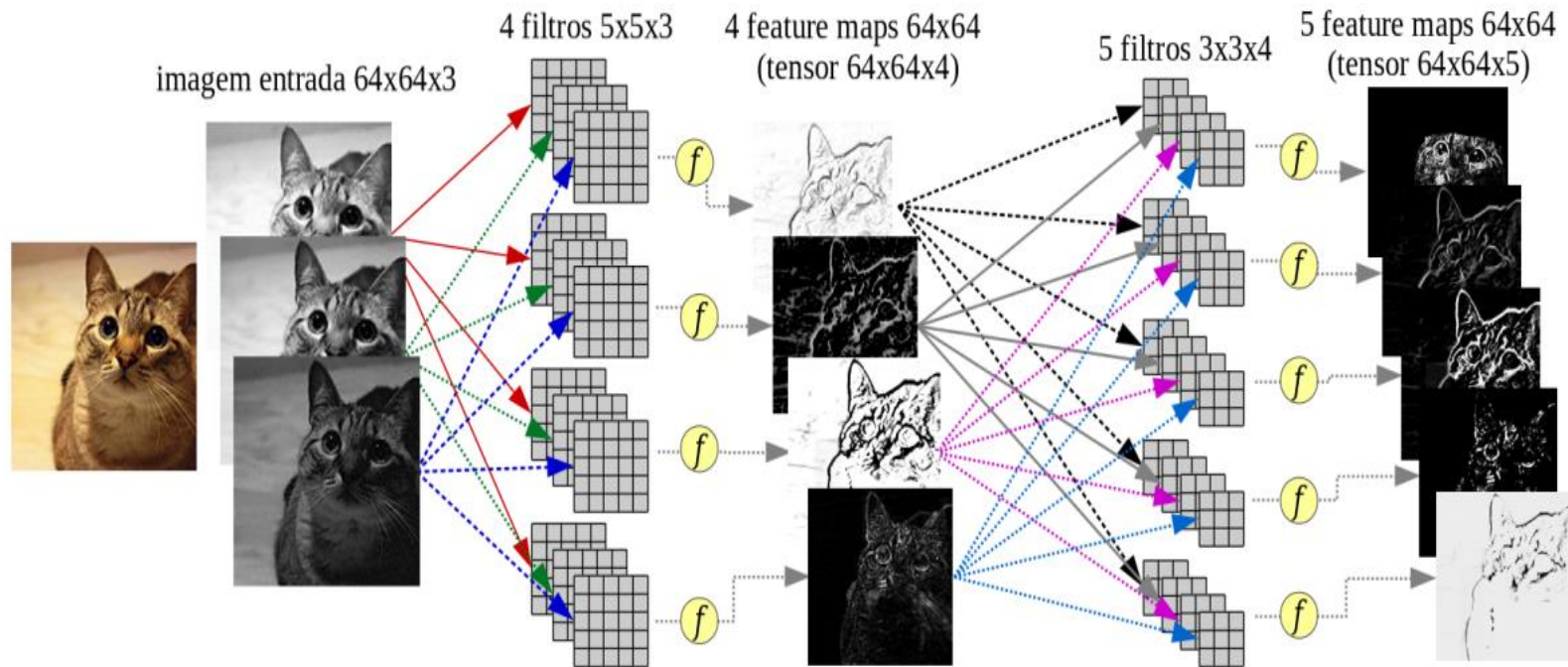
# Redes Neurais Artificiais



# Dados a serem interpretados

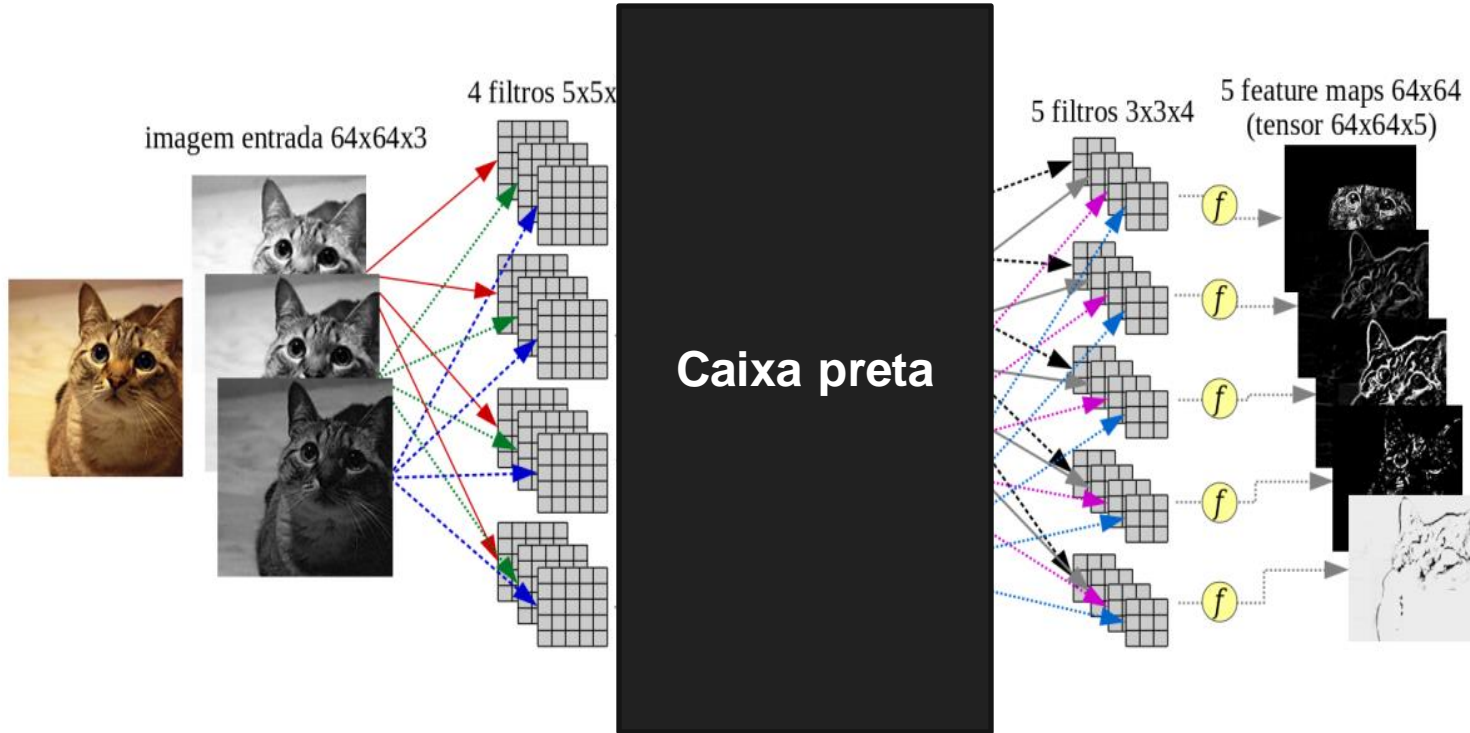


# Análise de características (features)





# Caixa preta gerada no treino

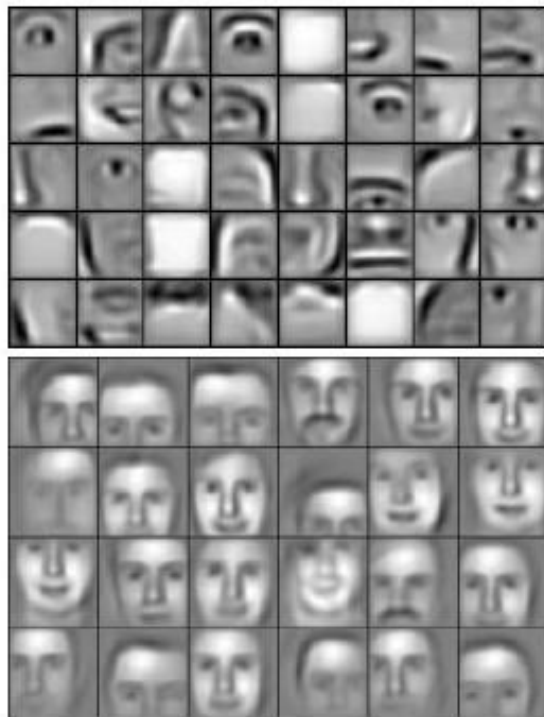


# Mas como são as *Features*?

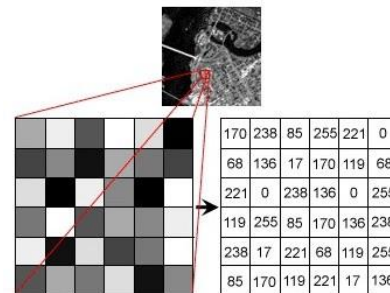
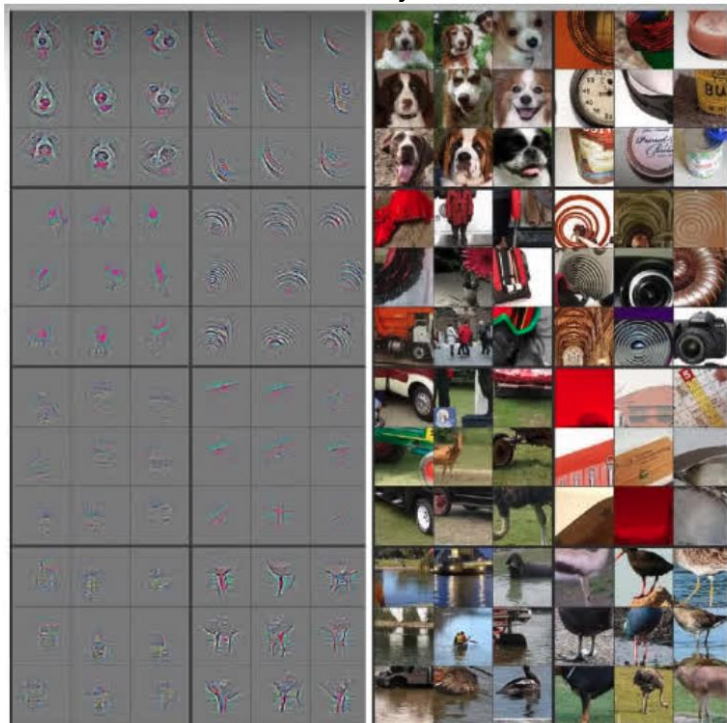


# Como são as *features*?

faces



Outros Objetos



$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

# Classificação de Objetos



# Classificação de objetos



CAT

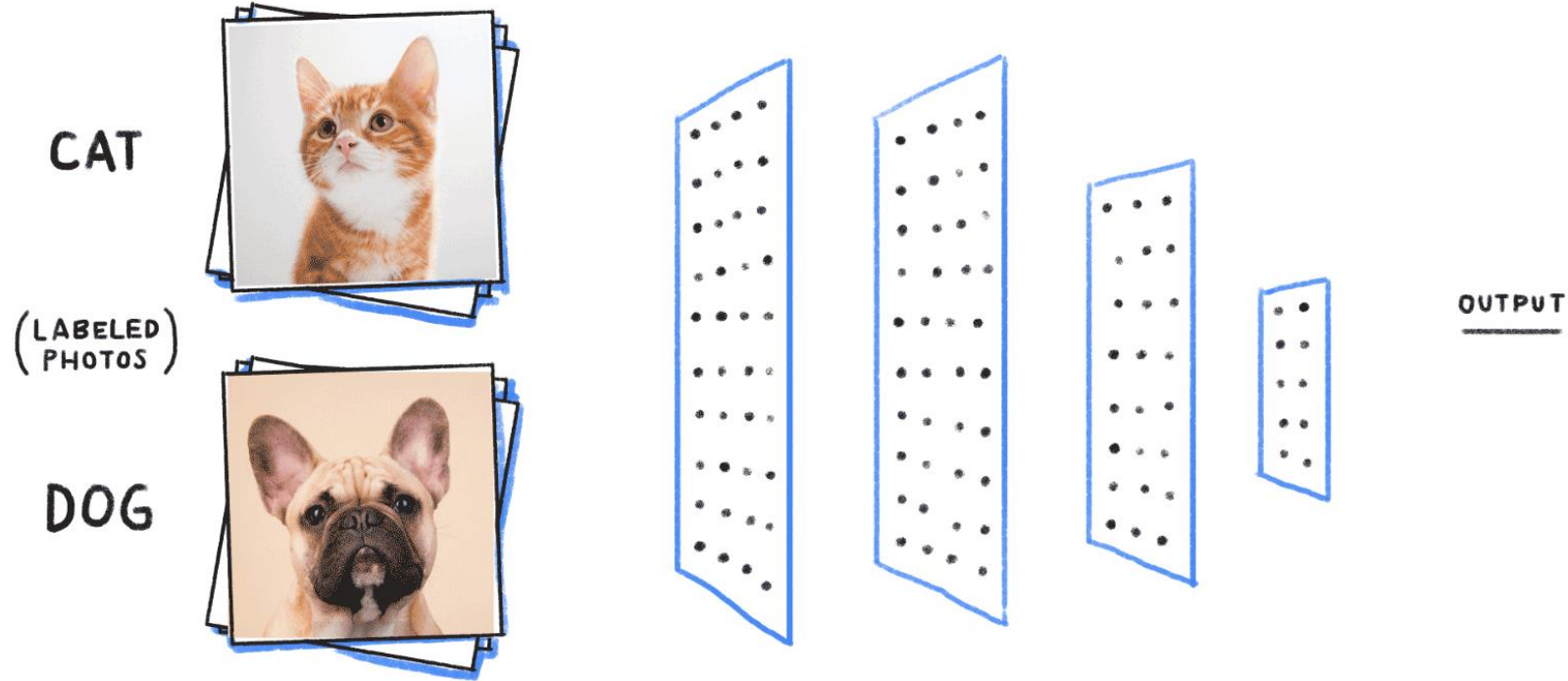
Aqui temos duas  
classes



NOT CAT



# Dados a serem interpretados



# Dados a serem interpretados

Entrada

Rótulo

Classificação



CAT



NOT CAT



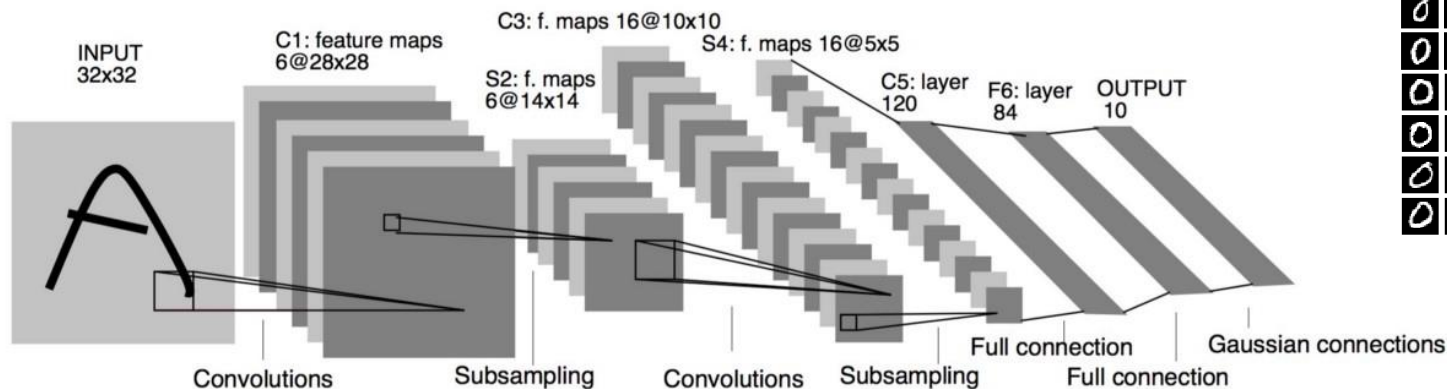
?



CAT

# Classificação de objetos

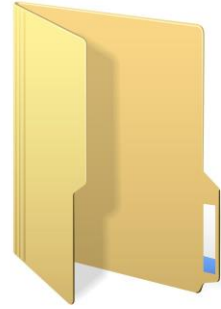
## Mnist Dataset



0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

Aqui temos 9 classes

# ***DATASET – Base de treino***



Classe “gatos”

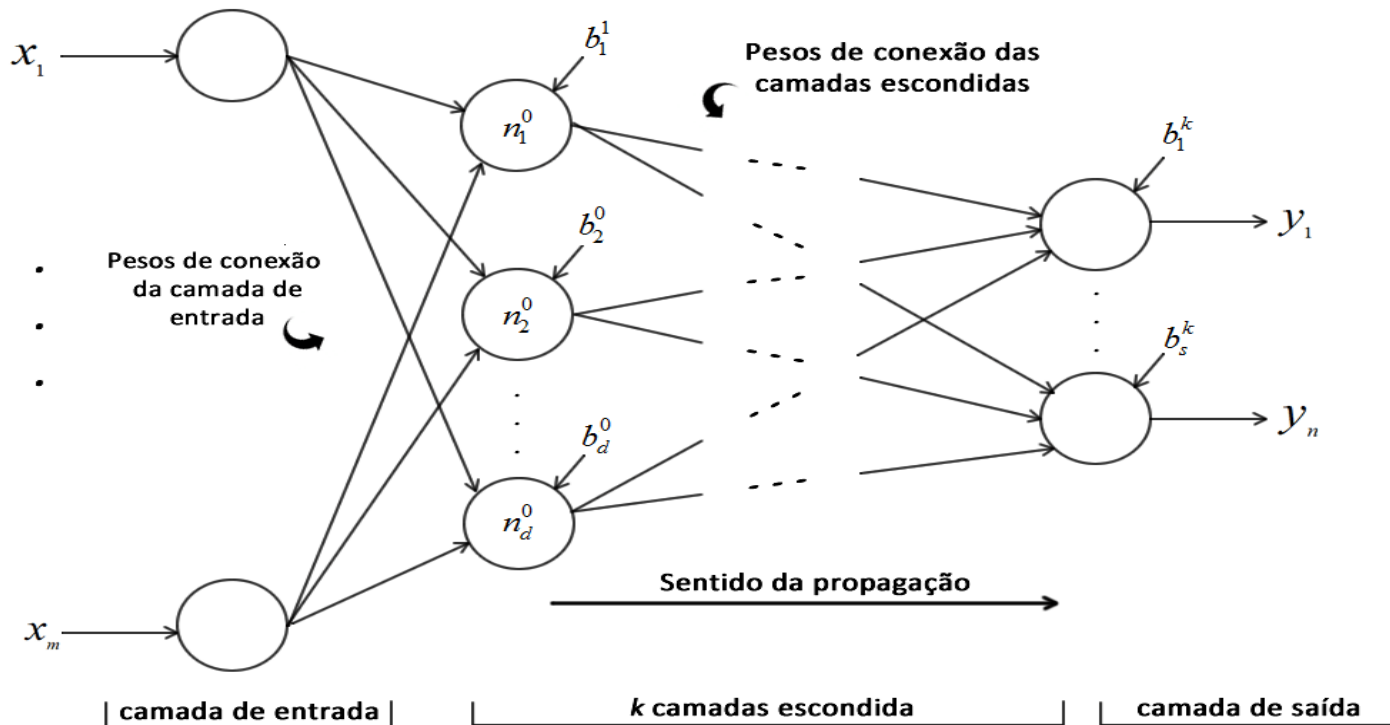
Mas o que gera um  
Treinamento?





# Dados de aprendizado

## Pesos gerados no treinamento



# Dados de aprendizado

Arquivos de pesos

Protótipo - Rede Neural Artificial

Efetuar treinamento Erro Total: 0,051603300438941

Entrada 01

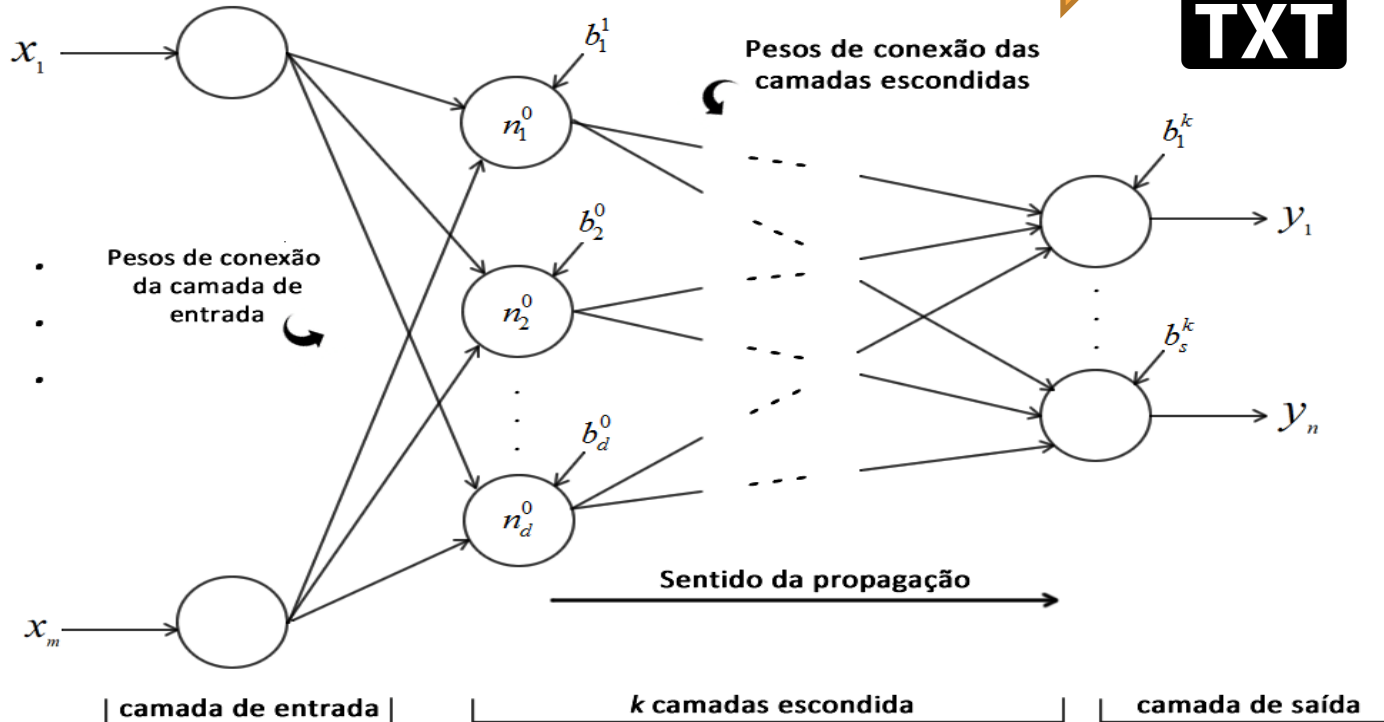
E01	E02	E03	E04	E05	E06	E07	E08
0	1	3	30	0	2000	2	3

Run



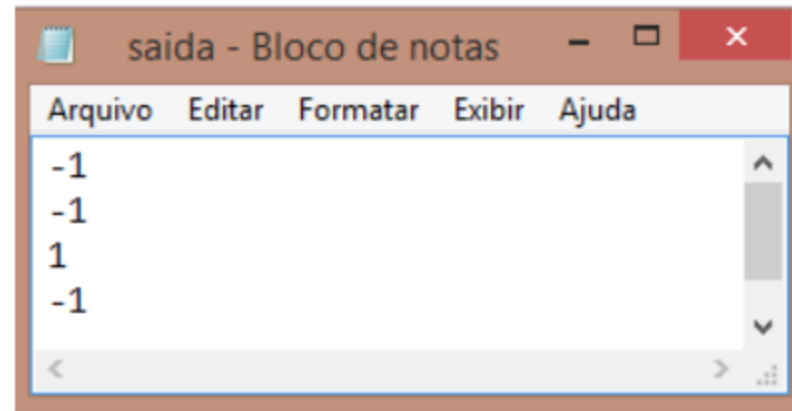
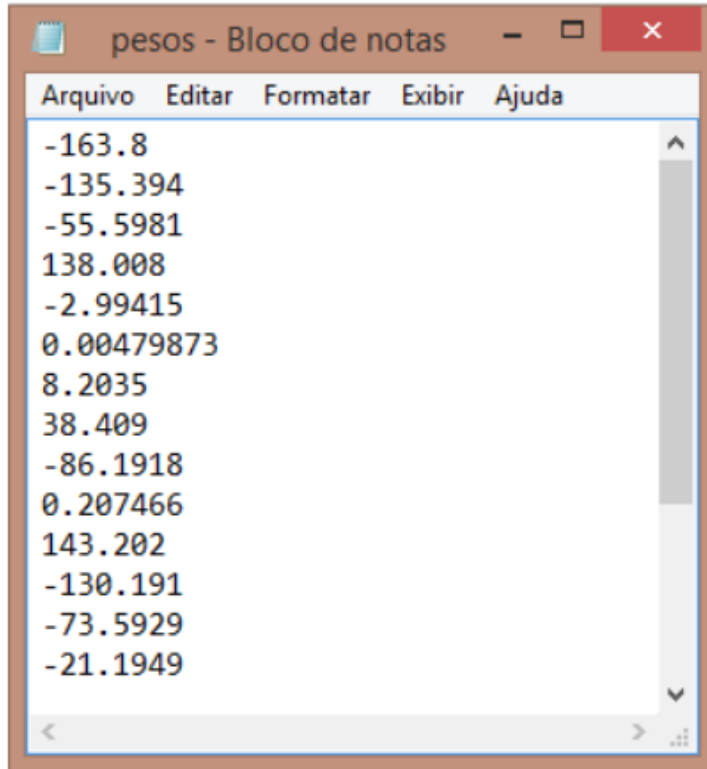
# Modelo de treinamento

Arquivos de pesos



# Modelo de treinamento

Pesos gerados em uma rede



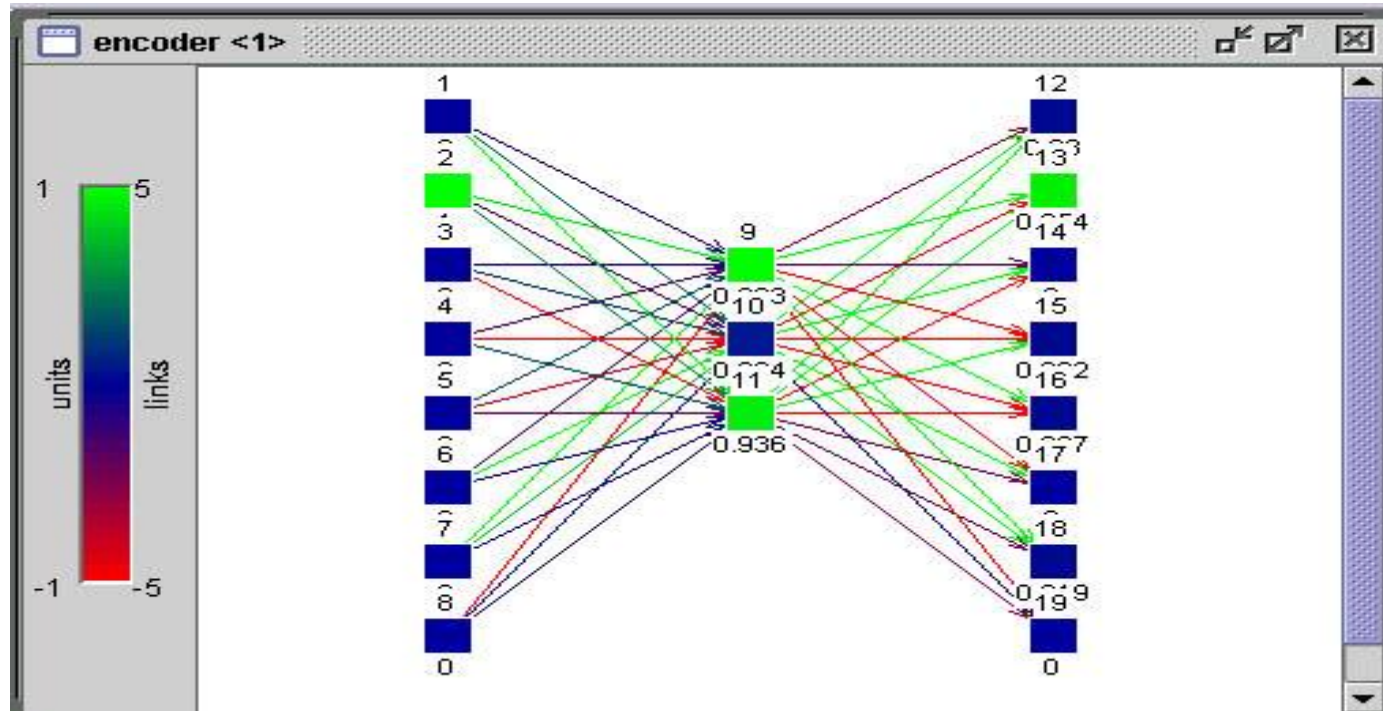
# Algoritmo Neural





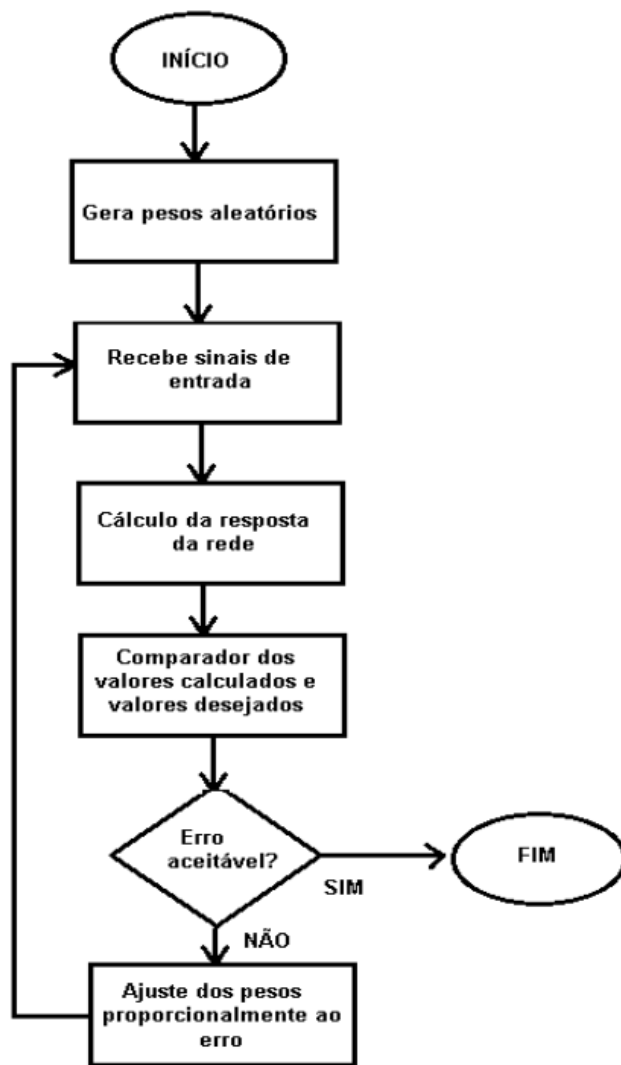
# Modelo de treinamento

## Relação dos pesos



# Algoritmo

```
31 def __init__(self, settings):
32     self.file = None
33     self.fingerprints = set()
34     self.logdups = True
35     self.debug = debug
36     self.logger = logging.getLogger(__name__)
37     if path:
38         self.file = open(os.path.join(path, 'requests.log'),
39                         'a')
40         self.file.seek(0)
41         self.fingerprints.update(re.sub(r'[\s\S]+', ' '), self.file)
42
43 @classmethod
44 def from_settings(cls, settings):
45     debug = settings.getbool('debug', False)
46     return cls(job_dir(settings), debug)
47
48 def request_seen(self, request):
49     fp = self.request_fingerprint(request)
50     if fp in self.fingerprints:
51         return True
52     self.fingerprints.add(fp)
53     if self.file:
54         self.file.write(fp + os.linesep)
55
56 def request_fingerprint(self, request):
57     return request_fingerprint(request)
```



# Importando Modelos de RNA



Classify ImageNet classes with ResNet50

```
from tensorflow.keras.applications.resnet50 import ResNet50
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import preprocess_input, decode_predictions
import numpy as np

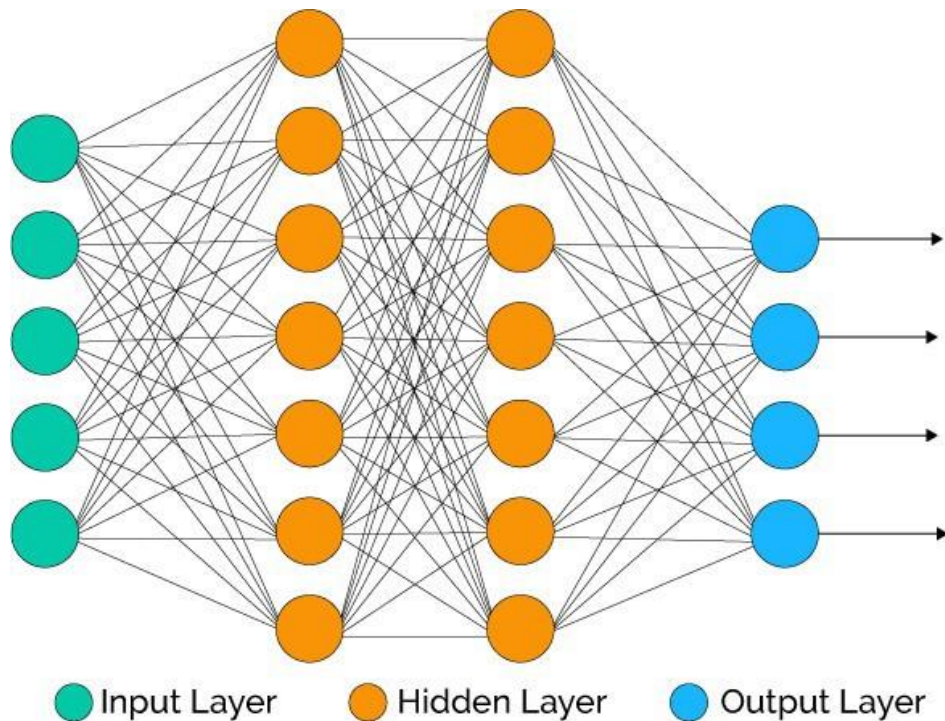
model = ResNet50(weights='imagenet')

img_path = 'elephant.jpg'
img = image.load_img(img_path, target_size=(224, 224))
x = image.img_to_array(img)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
```



Keras

# Exemplo de RNA no COLAB



colab

[https://colab.research.google.com/github/storopoli/ciencia-de-dados/blob/master/notebooks/Aula\\_18\\_a\\_Redes\\_Neuralis\\_com\\_TensorFlow.ipynb#scrollTo=6zmMUxg8pfqE](https://colab.research.google.com/github/storopoli/ciencia-de-dados/blob/master/notebooks/Aula_18_a_Redes_Neuralis_com_TensorFlow.ipynb#scrollTo=6zmMUxg8pfqE)

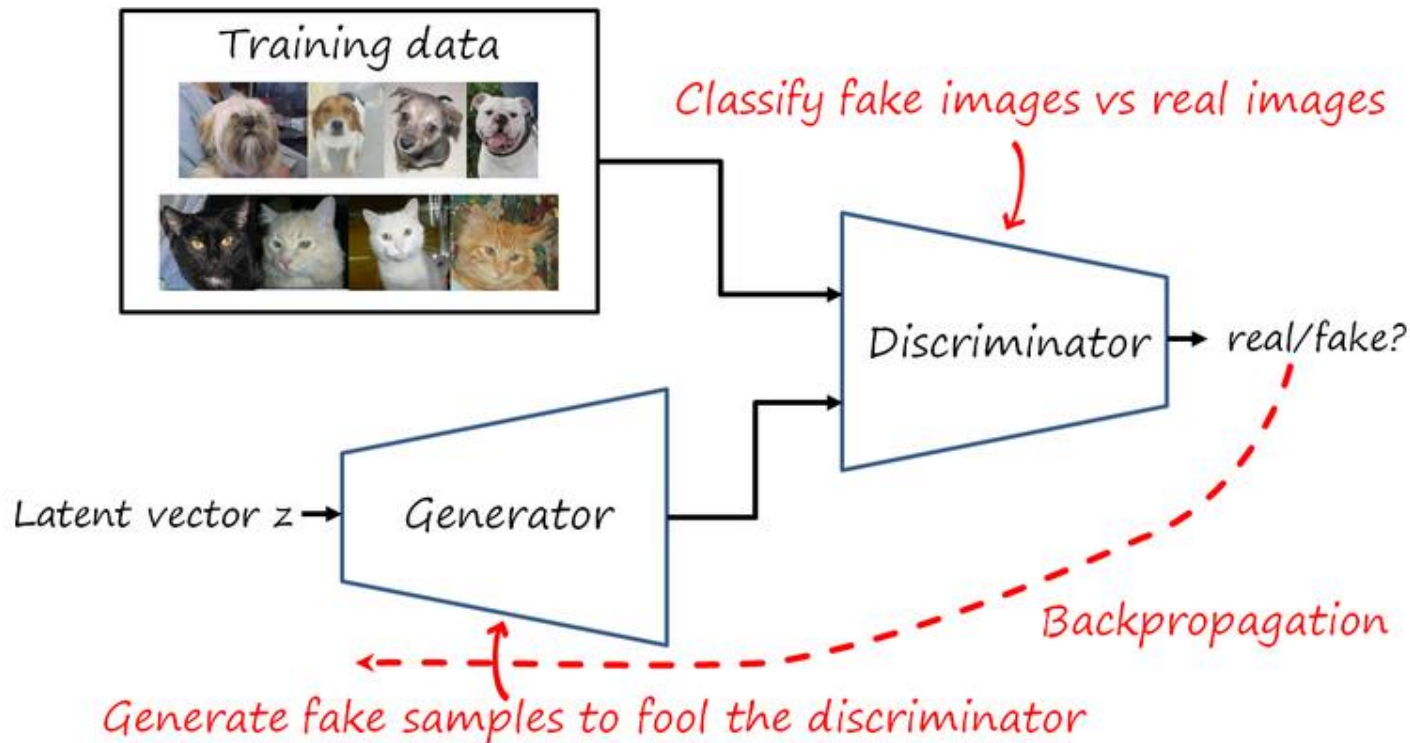
# Redes Adversárias





# Redes adversárias

## DeepFake



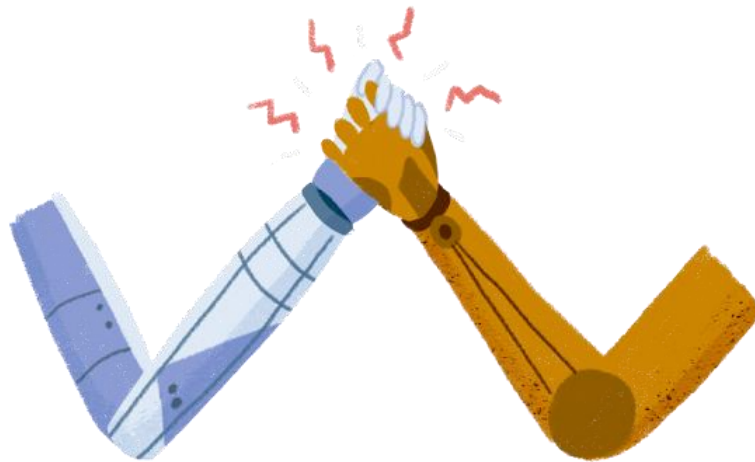
# Redes adversárias

Imagine que a tarefa seja gerar uma imagem inspirada no estilo artístico de Picasso. As equipes de desenvolvimento de IA podem coletar todas as pinturas de Picasso e treinar uma GAN para identificar as cores, características e pinceladas individuais representativas das obras do artista.



# Redes adversárias

O sistema de imitação aplica os conhecimentos que tem para produzir milhares de novas imagens no estilo de Picasso, usando características de obras de arte existentes, enquanto outro sistema de IA avalia a semelhança entre as criações e o estilo de Picasso e gera uma classificação. Os resultados não convincentes são retornados ao sistema de imitação para serem aprimorados.



# Redes adversárias

Após trocar informações milhões de vezes, o sistema de imitação fica cada vez melhor na criação de pinturas no estilo de Picasso.



PABLO PICASSO



IA

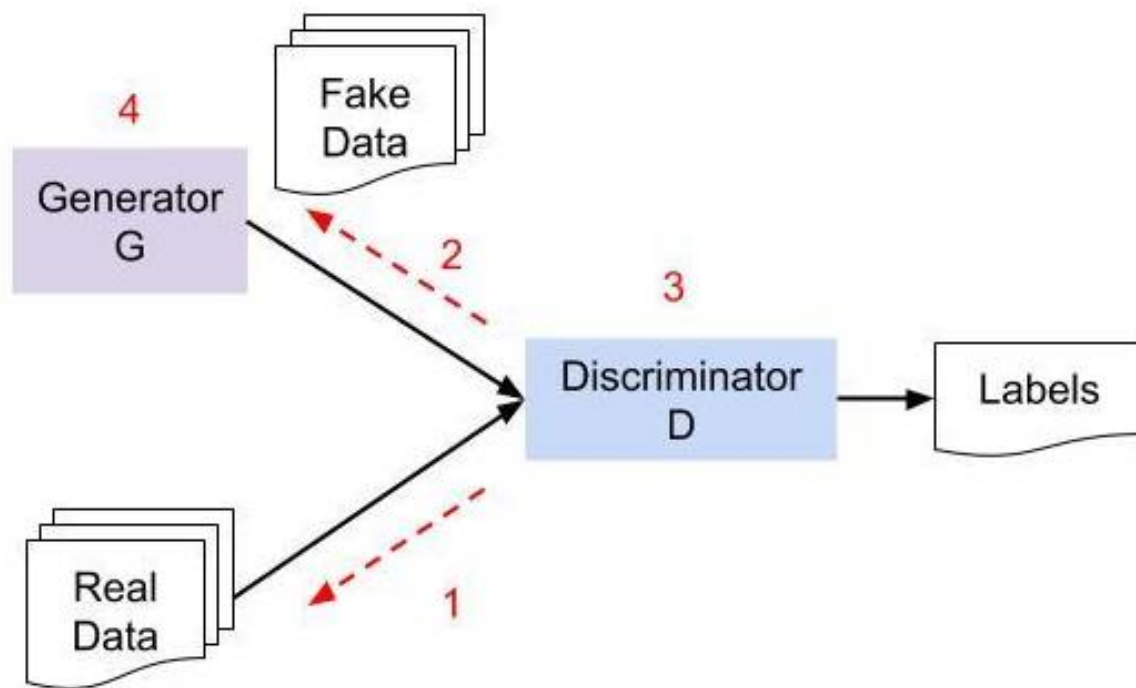
# Redes adversárias

As GANs vão além da simples memorização do que já foi feito: elas criam conteúdo novo. Por isso, elas são consideradas um marco importante pela comunidade de pesquisa de IA. Designers e arquitetos já exploram o potencial desses sistemas para gerar modelos 3D de carros e edifícios com base no estudo de fotos em 2D.



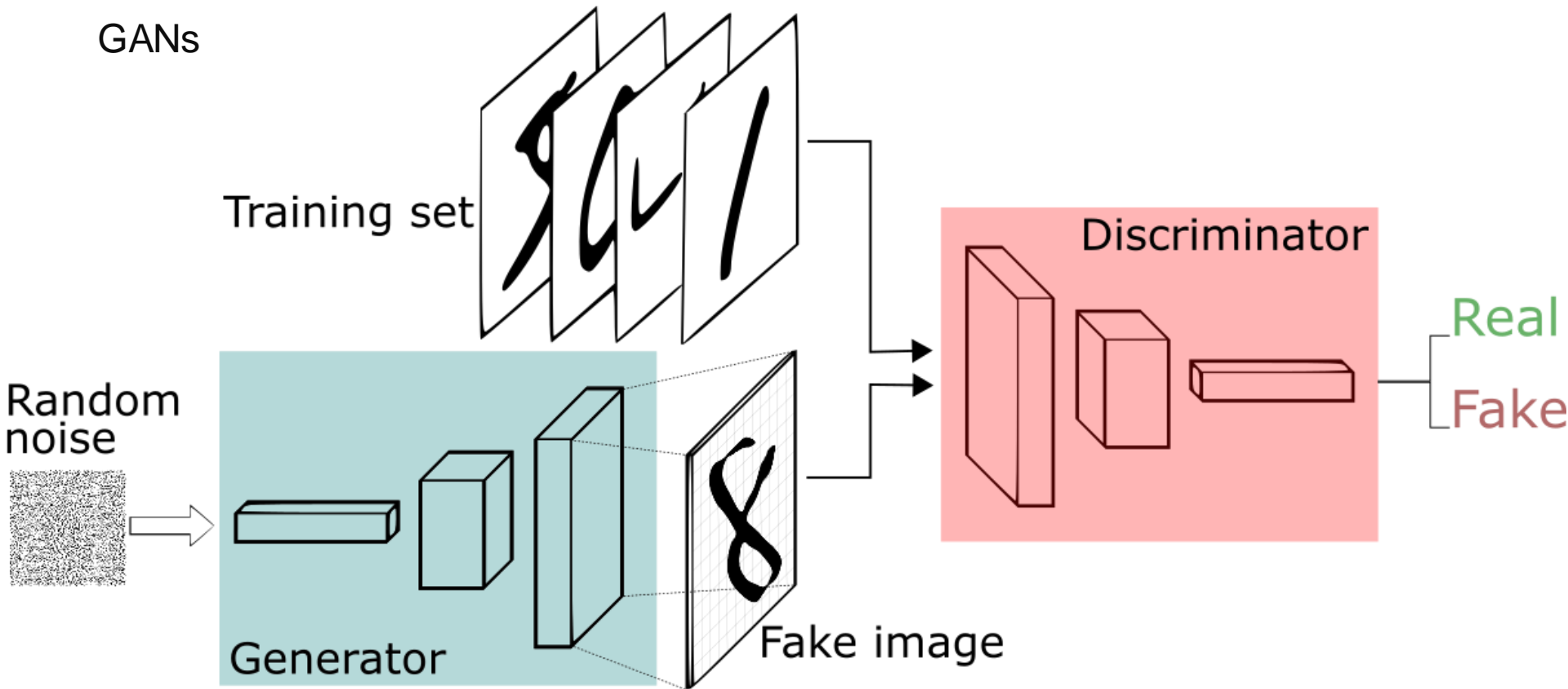
# Redes adversárias

GANs



# Redes adversárias

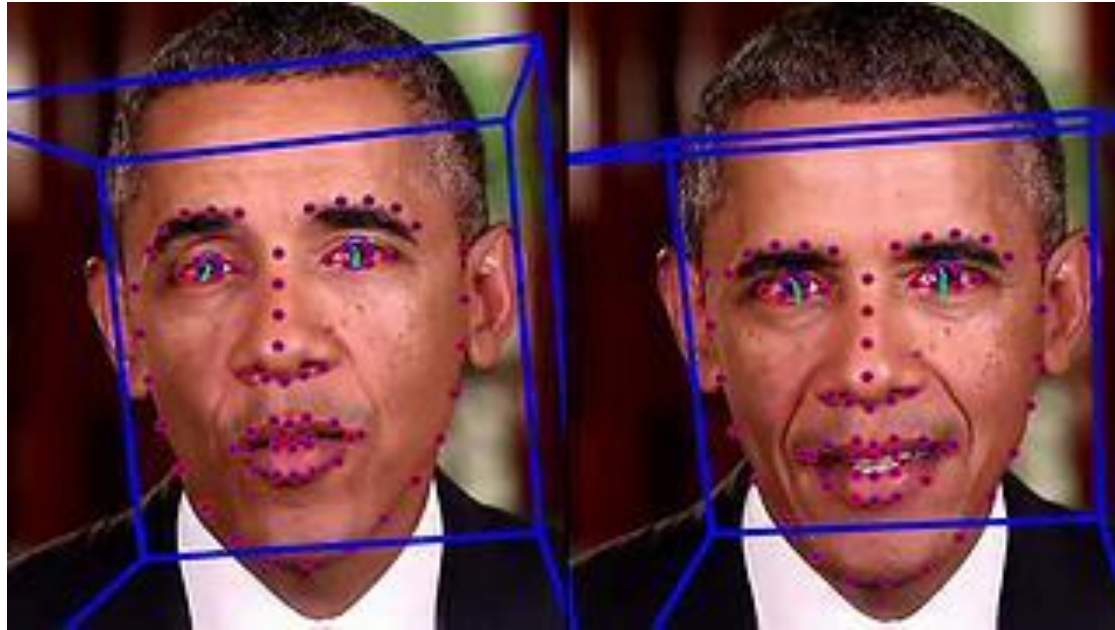
GANs





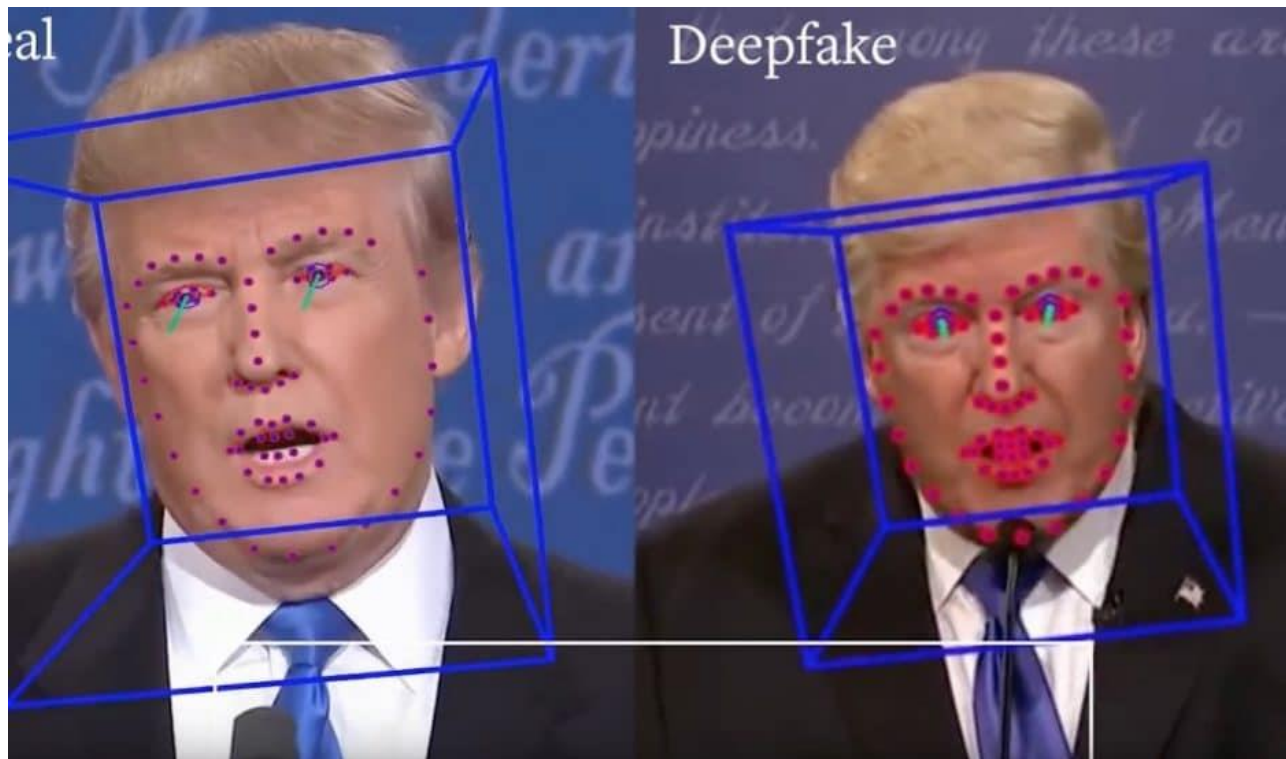
# Redes adversárias

GANs



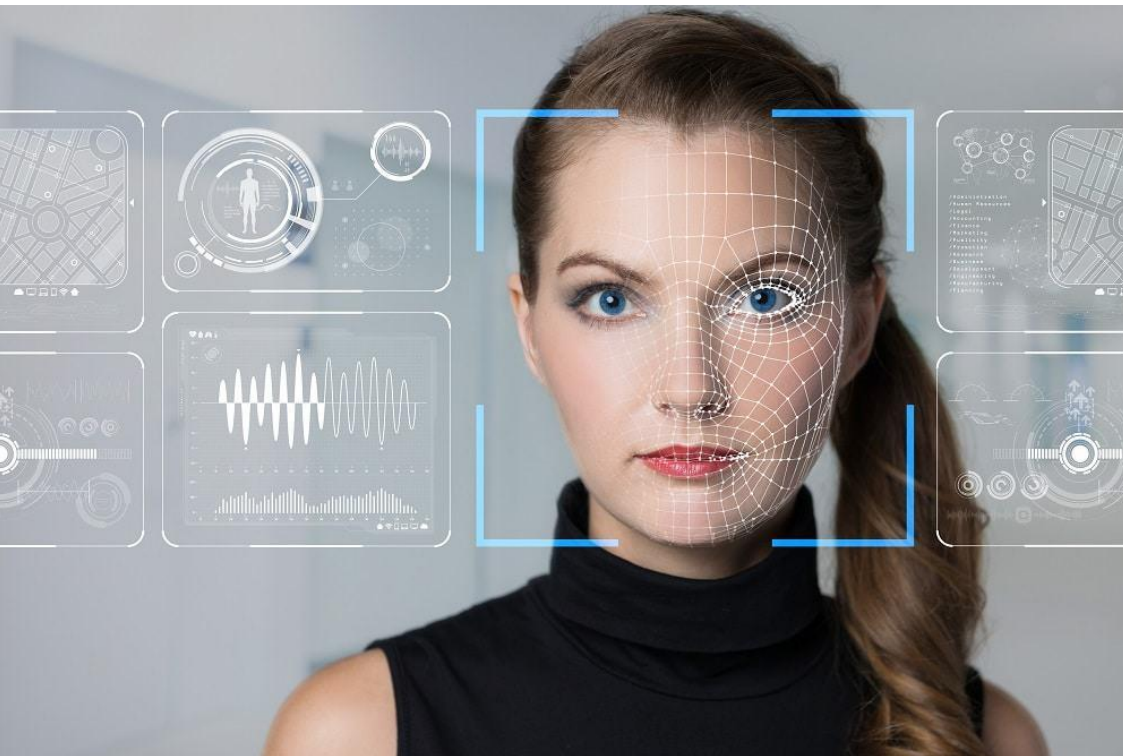
# Redes adversárias

GANs



# Redes adversárias

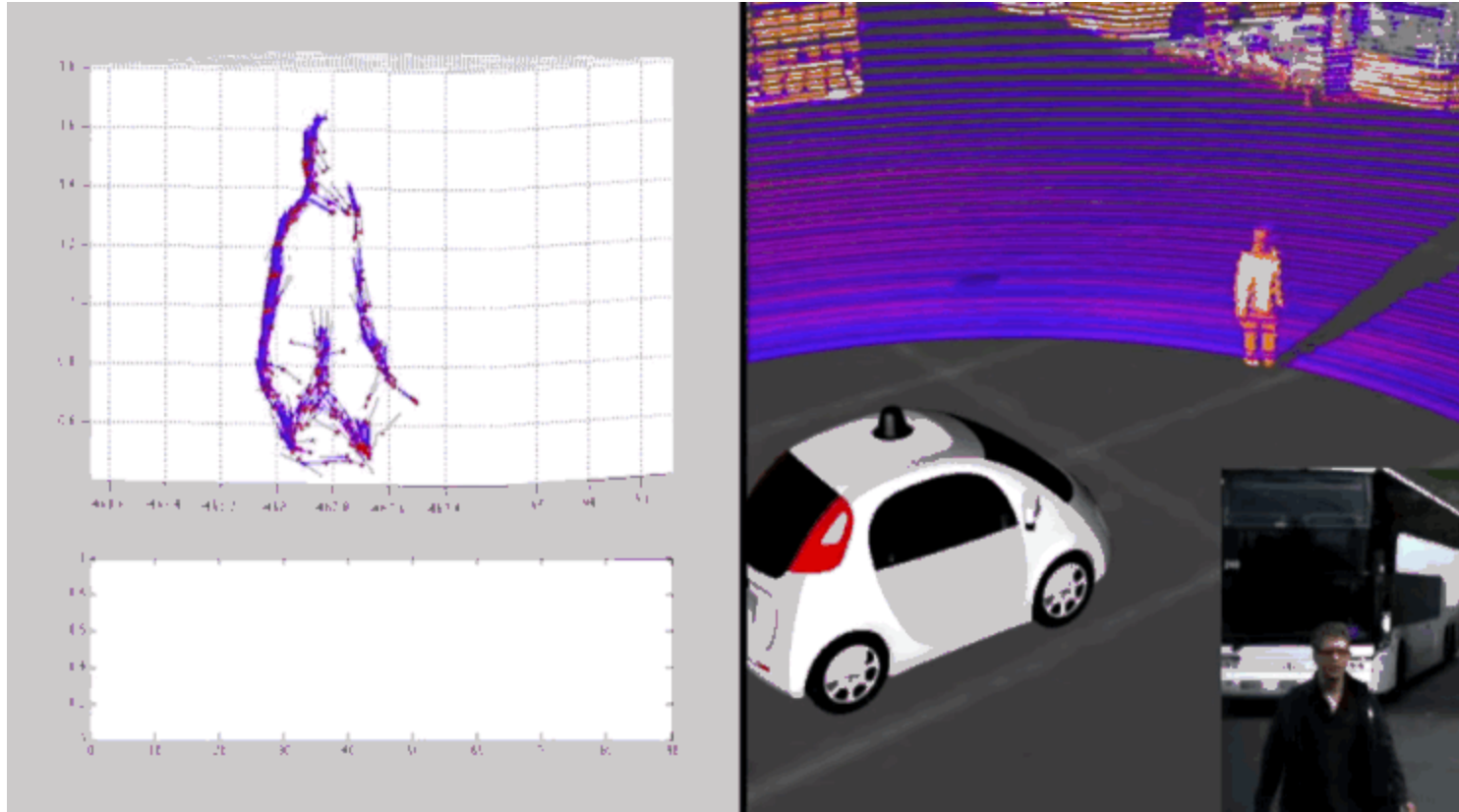
## Problemas



- Desinformação:
- Fraude e extorsão:
- Violação de privacidade:
- Uso indevido de imagens e vozes:
- Dificuldade na detecção:

# Como surgiram?

## Aplicações em veículos autônomos



# Como surgiram?

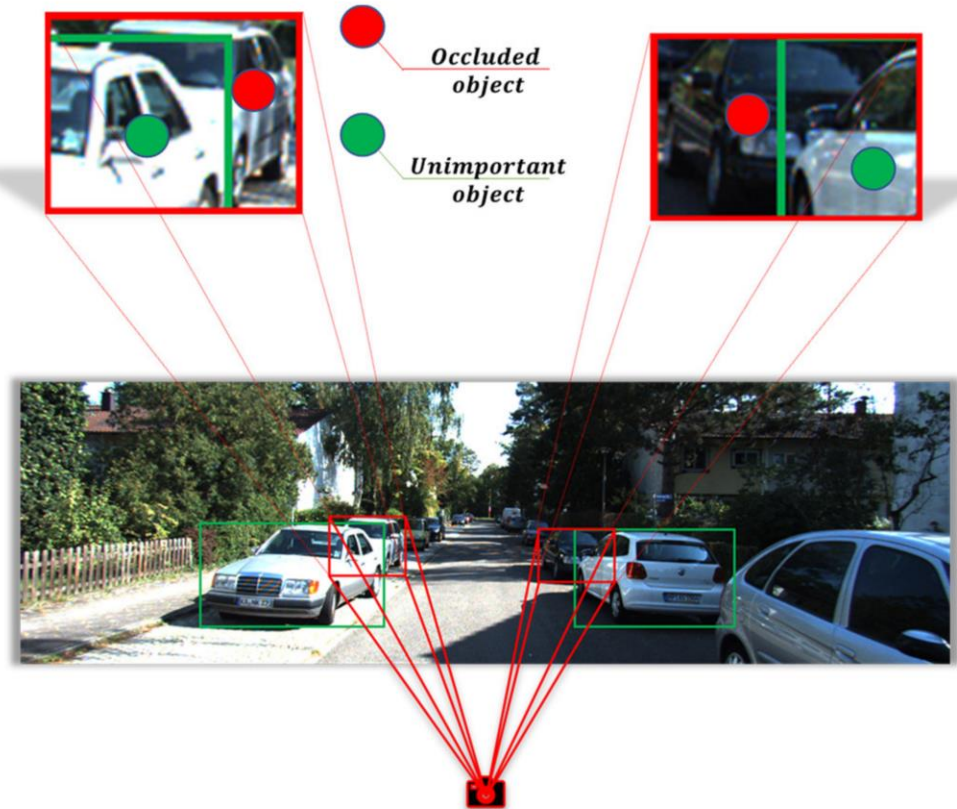
## Segmentação





# Como surgiram?

## Oclusões



# Como surgiram?

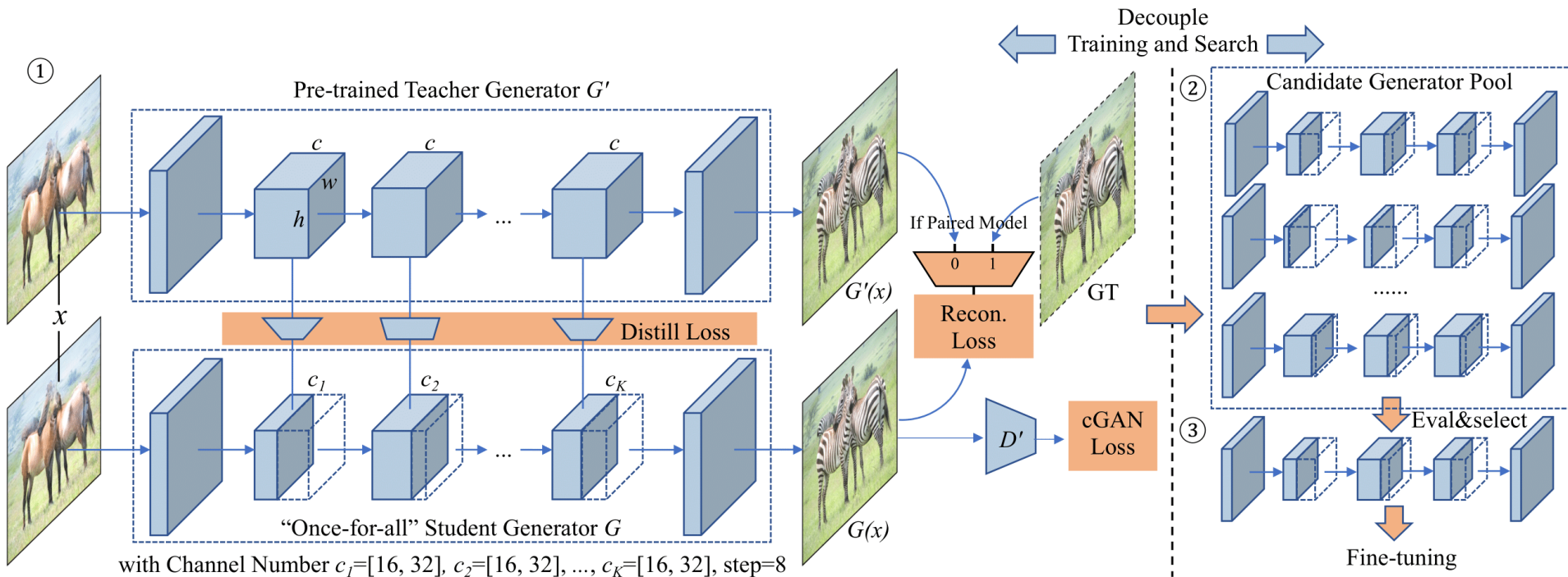
## Gerações de imagens





# Como surgiram?

## Gerações de imagens



# Projeto prático:



## Gerações de imagens

[https://colab.research.google.com/github/lexfridman/mit-deep-learning/blob/master/tutorial\\_gans/tutorial\\_gans.ipynb](https://colab.research.google.com/github/lexfridman/mit-deep-learning/blob/master/tutorial_gans/tutorial_gans.ipynb)



# Obrigado!

*Machine Learning*

Prof. Dr. Diego Bruno