Deep Learning

Redes Neurais Artificiais

Aula 3

Instruções Gerais

- ✓ Traga seu laptop

Instruções Gerais

- ✓ Traga seu laptop
- ✓ Use Software Livre

Instruções Gerais

- ✓ Traga seu laptop
- ✓ Use Software Livre
- ✓ Não converse por voz

Instruções Gerais

- ✓ Traga seu laptop
- ✓ Use Software Livre
- ✓ Não converse por voz
- ✓ Se não entender, pergunte!

Instruções Gerais

- ✓ Traga seu laptop
- ✓ Use Software Livre
- ✓ Não converse por voz
- ✓ Se não entender, pergunte!
- ✓ Se entender, explique!

Instruções Gerais

- ✓ Traga seu laptop
- ✓ Use Software Livre
- ✓ Não converse por voz
- ✓ Se não entender, pergunte!
- ✓ Se entender, explique!
- ✓ NÃO ENTRE EM PÂNICO

Bom dia!

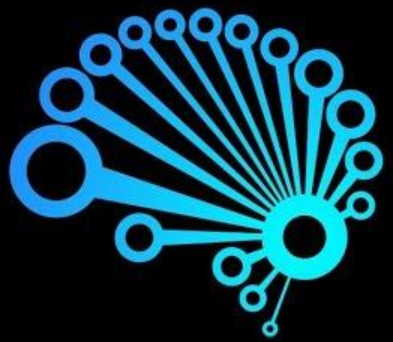
Eu sou Diego Dorgam

Alguma pergunta que você quer fazer?!

http://bit.ly/dl-unb03
https://t.me/DeepLearningUnB
@diegodorgam

O que Vamos Aprender?

1. Intuição
    - O que é o Neurônio
    - Funções de Ativação
    - Funcionamento das Redes Neurais
    - Aprendizagem nas Redes Neurais
2. Prática
    - Instalando o Keras
    - Construindo uma ANN

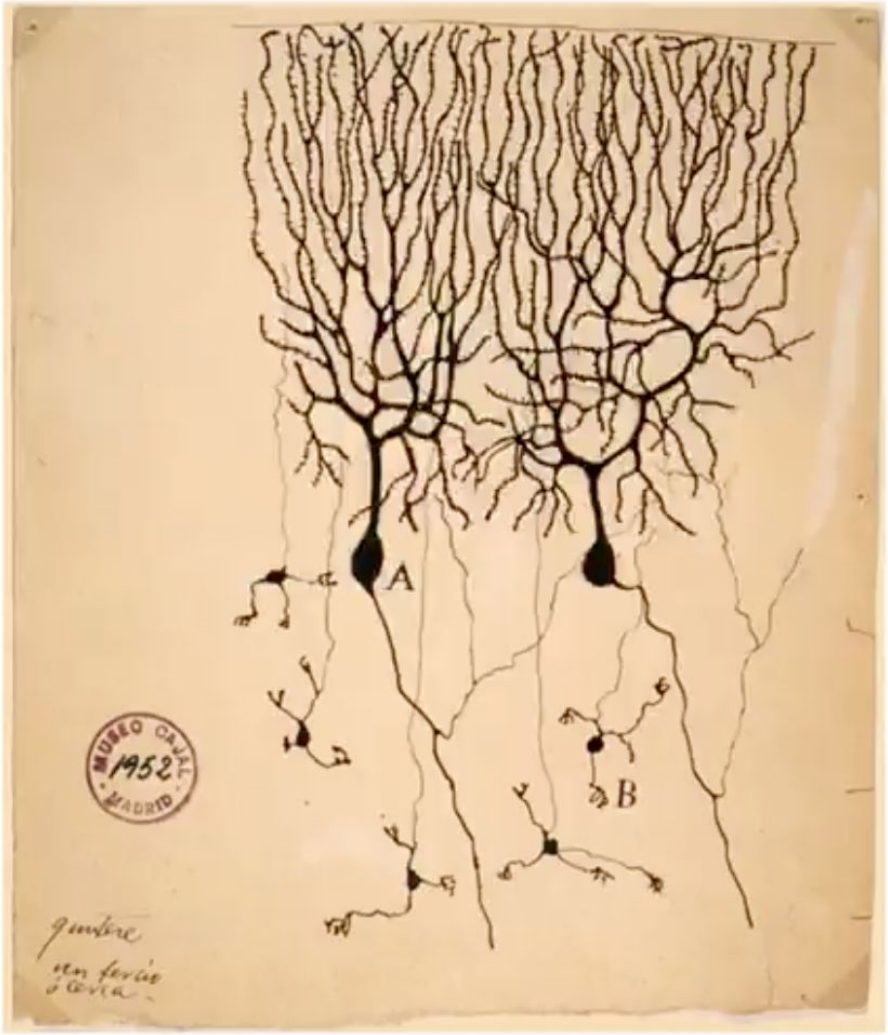


SUPER
DATASCIENCE
MAKING THE COMPLEX SIMPLE

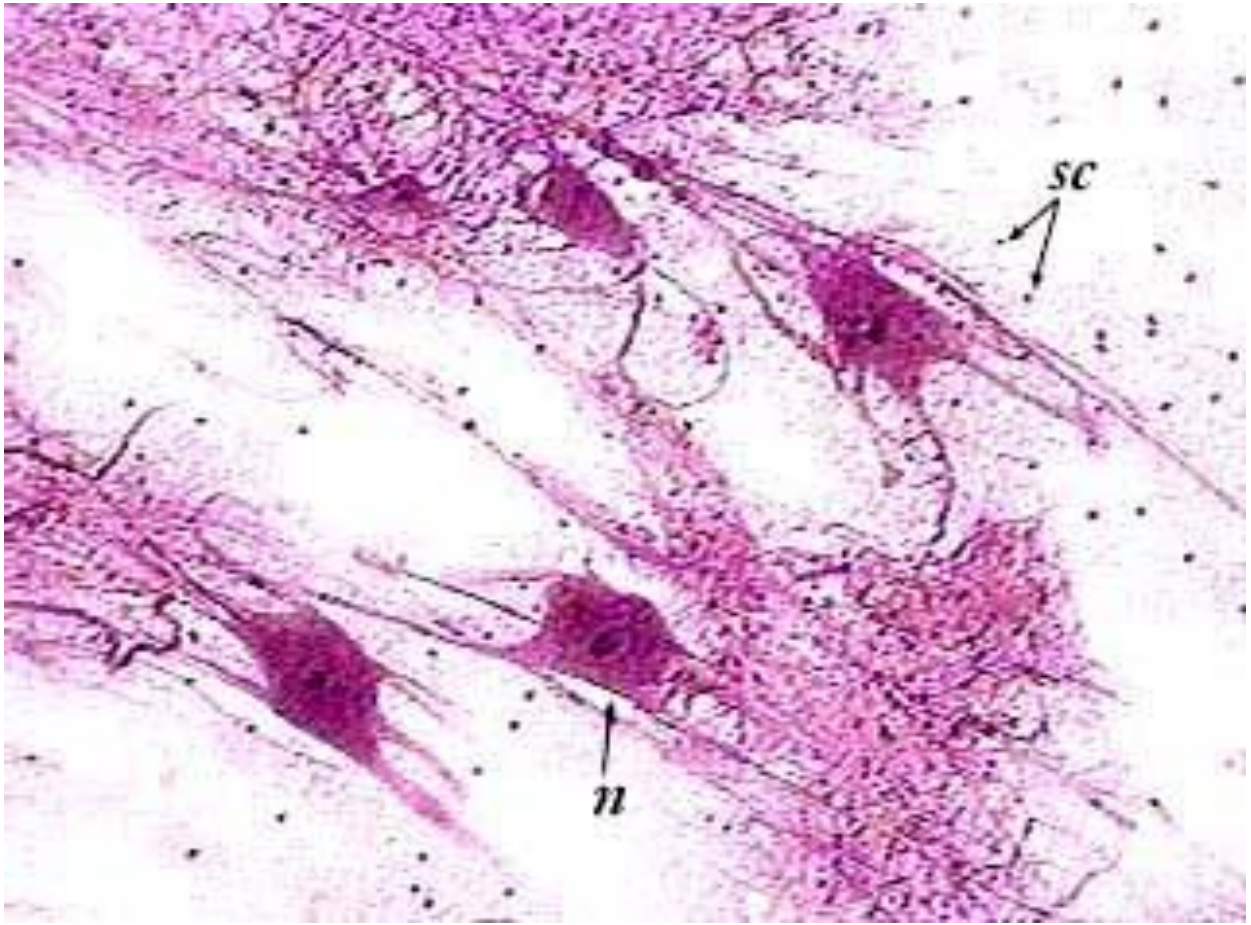

The Super
Data Science
PODCAST
w/ Kirill
Eremenko
Data | Career | Success

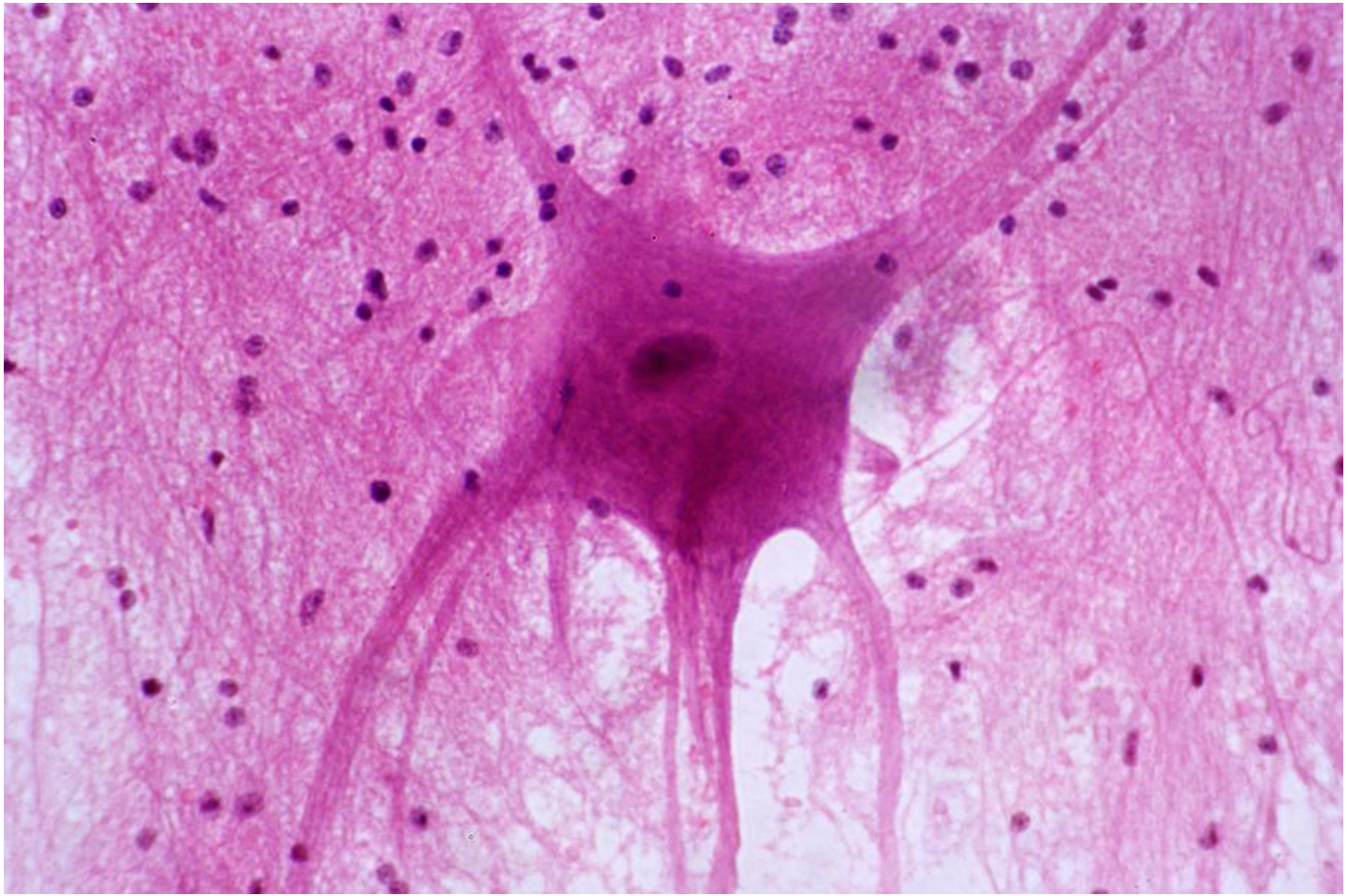
O que é O Neurônio

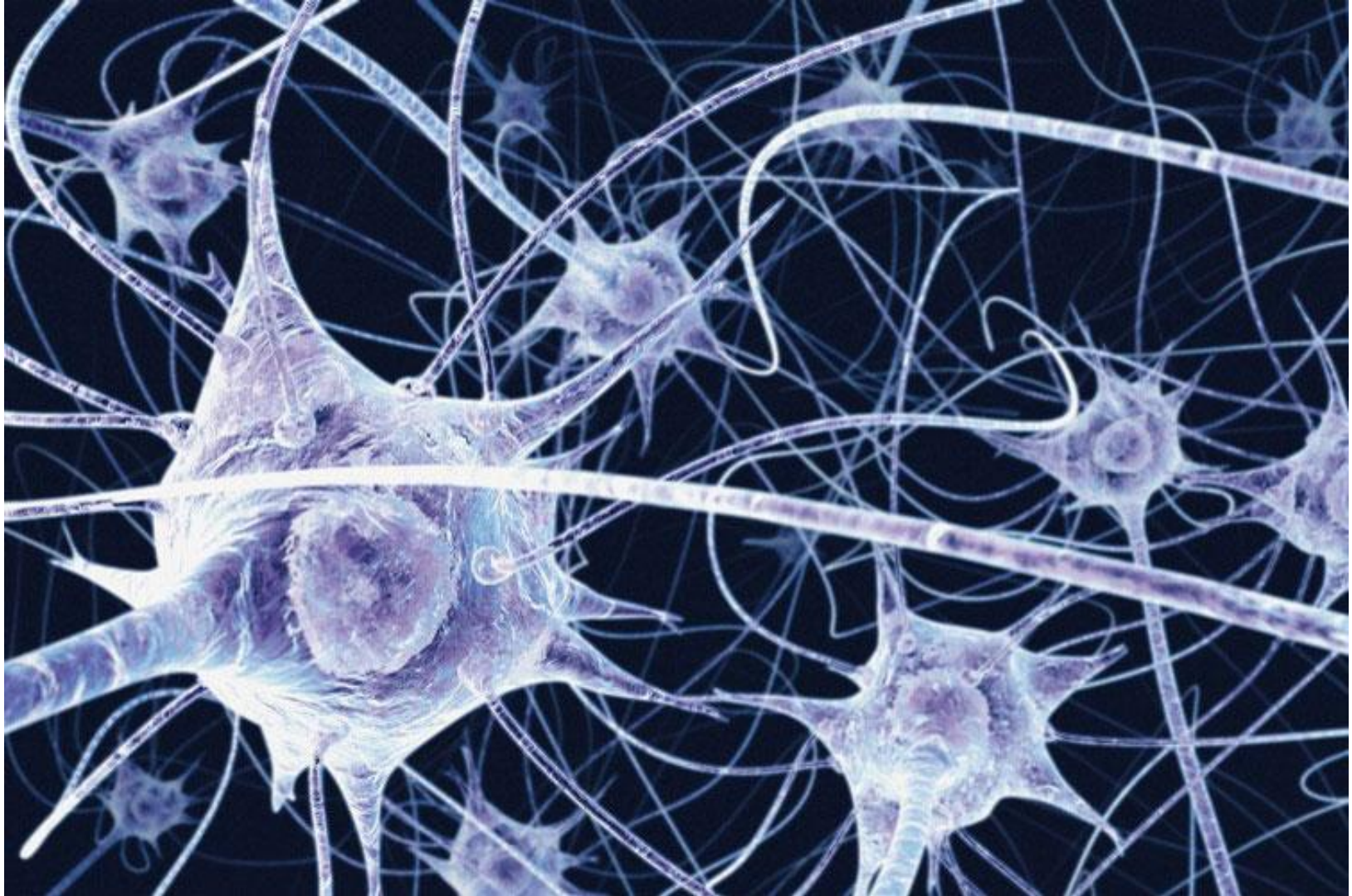
Redes Neurais Artificiais



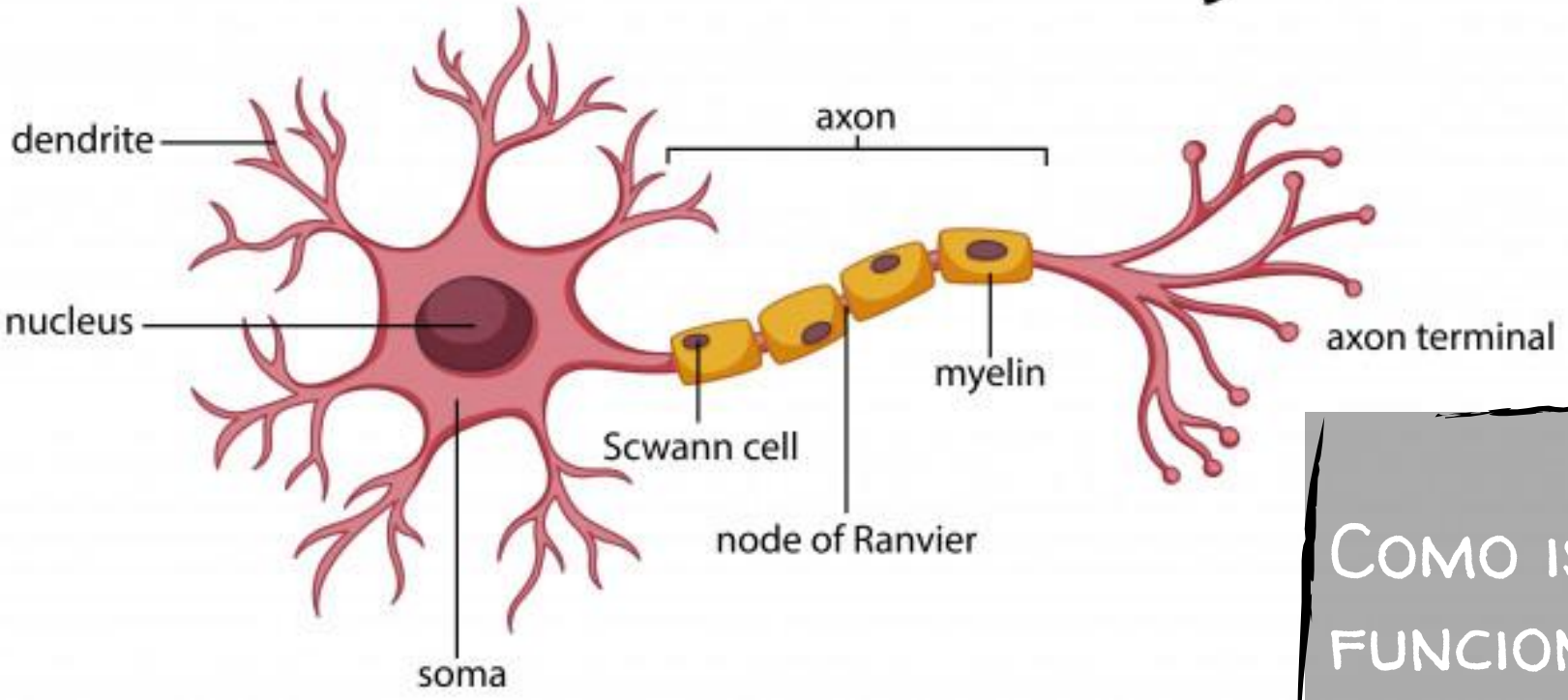
Santiago Ramón y Cajal, 1899


sc
n

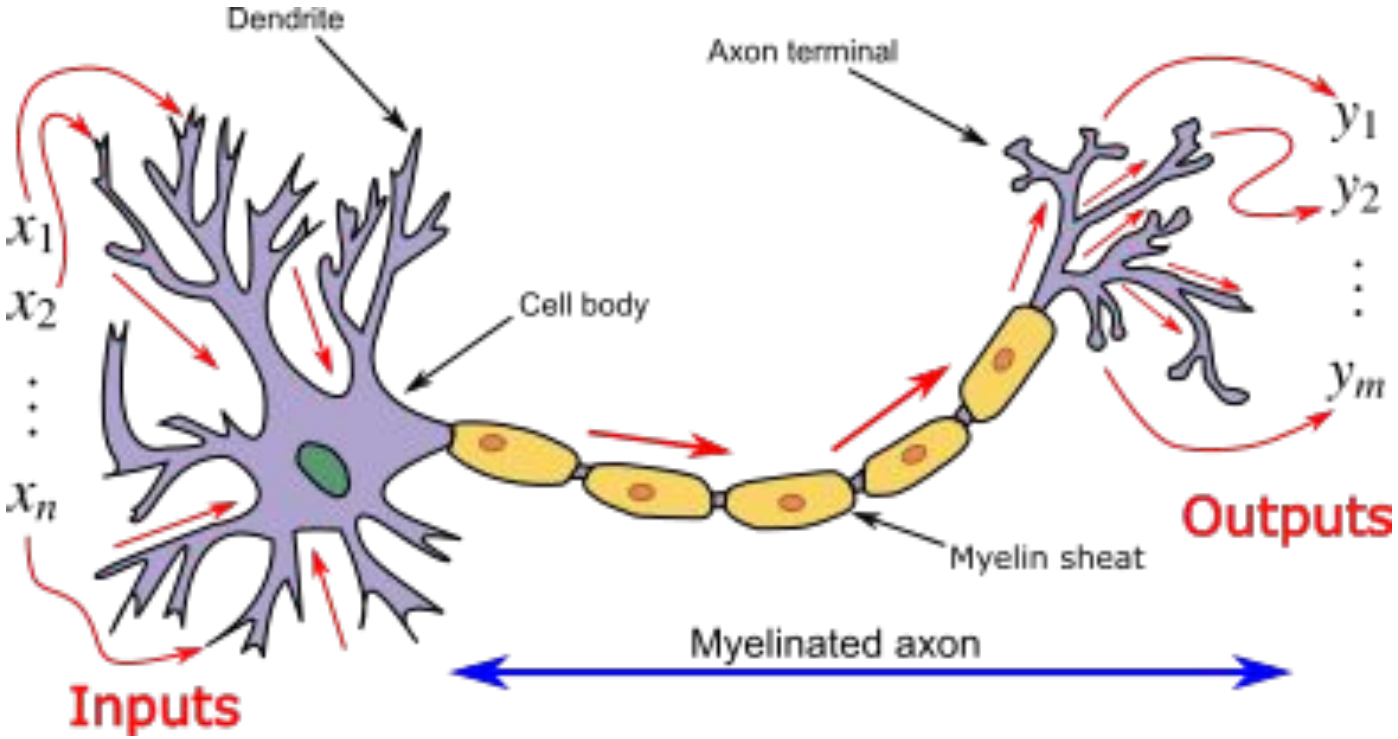


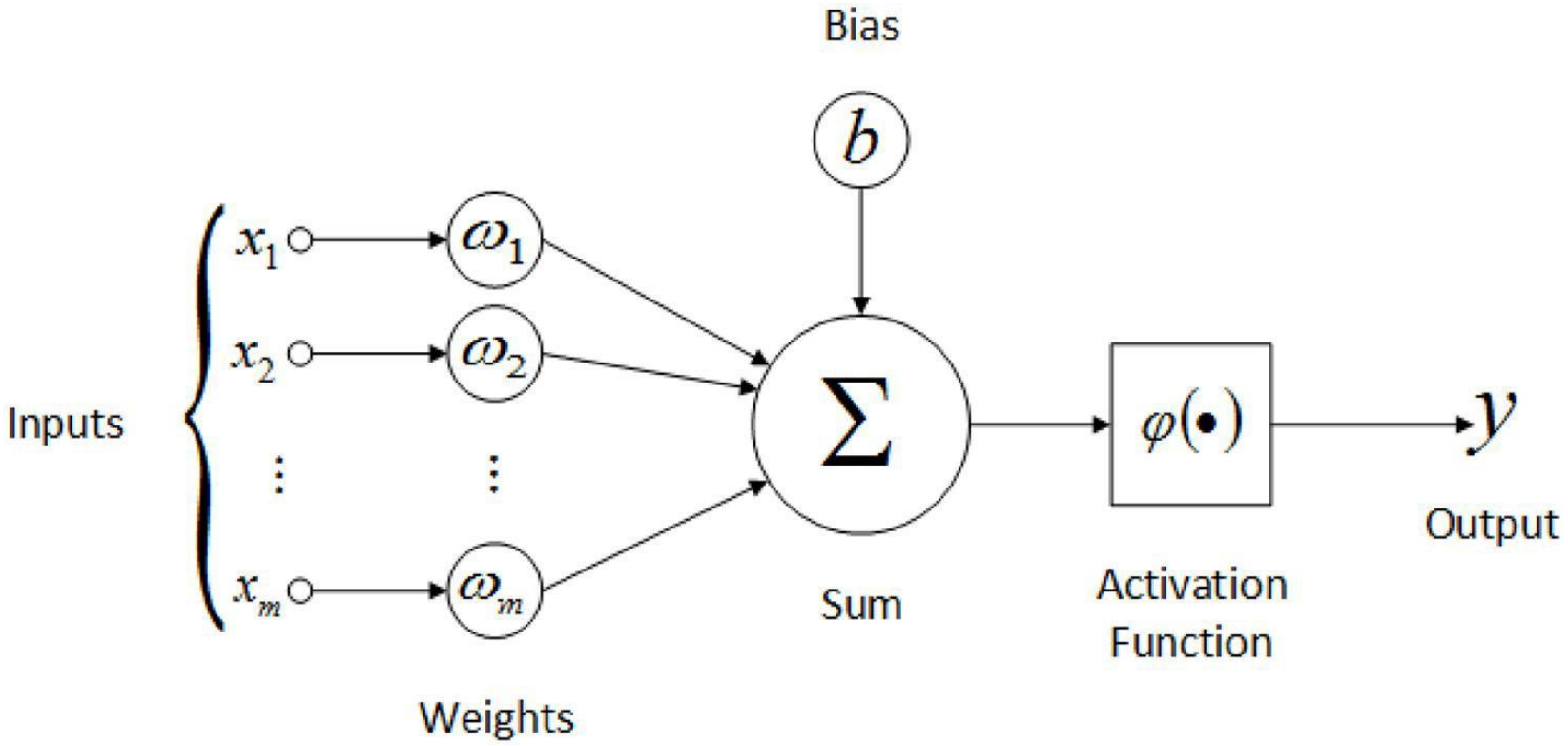


Neuron Anatomy

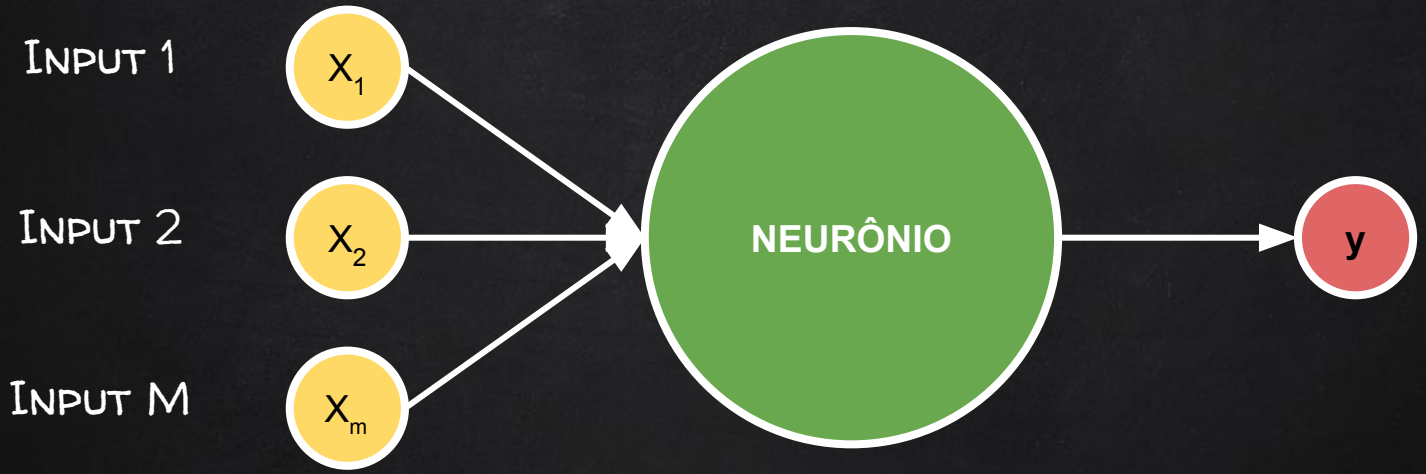


Como isso funciona?


Dendrite
Axon terminal
x_1
x_2
x_n
Cell body
y_1
y_2
y_m
Outputs
Myelin sheat
Myelinated axon
Inputs


Bias
b
Inputs
x_1
x_2
x_m
ω_1
ω_2
ω_m
Σ
$\varphi(\bullet)$
y
Output
Sum
Activation Function
Weights

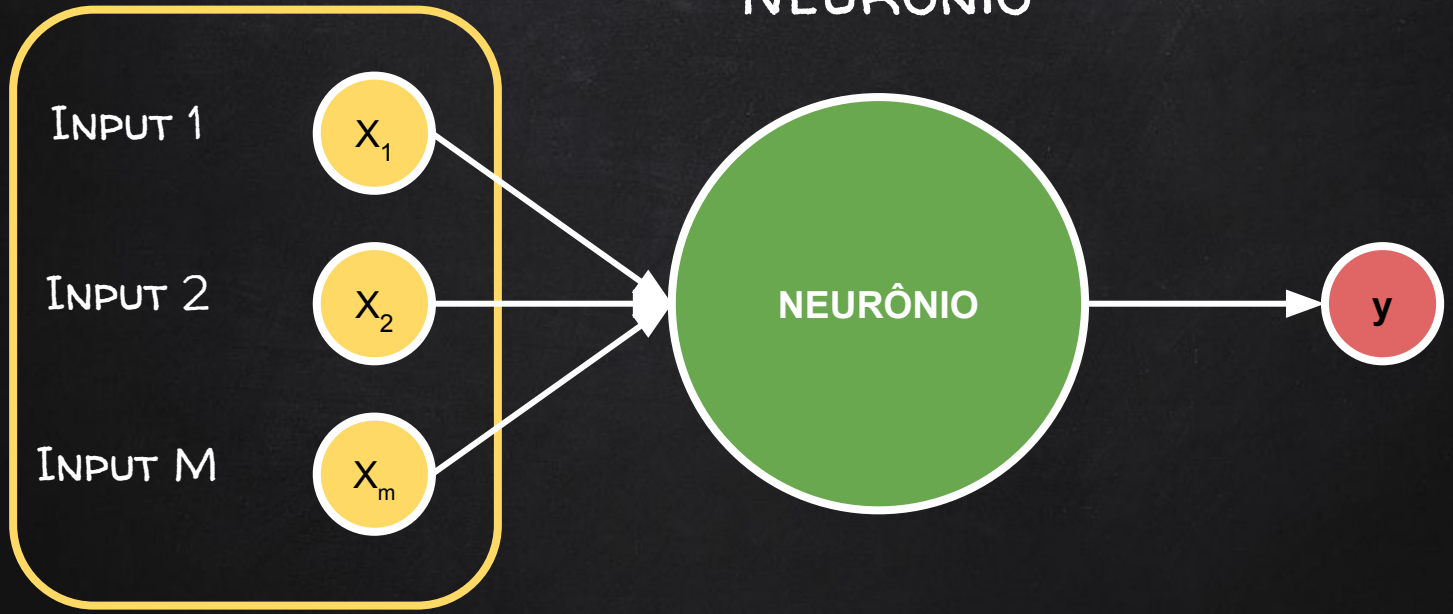
Neurônio



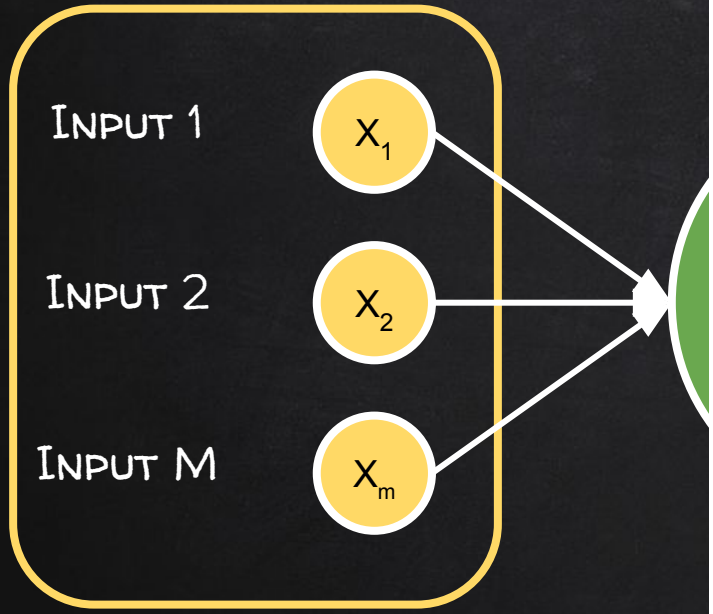
Variáveis Independentes

Variável dependente

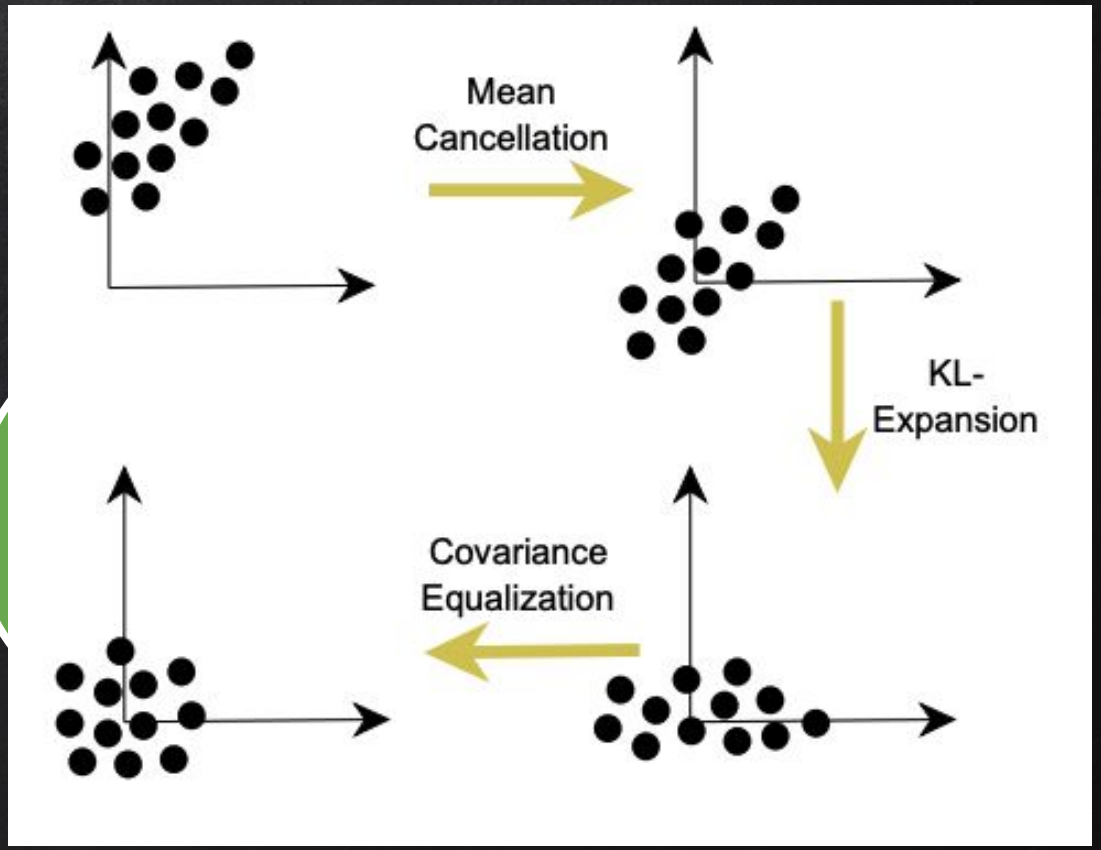
Neurônio



Padronização
Normalização



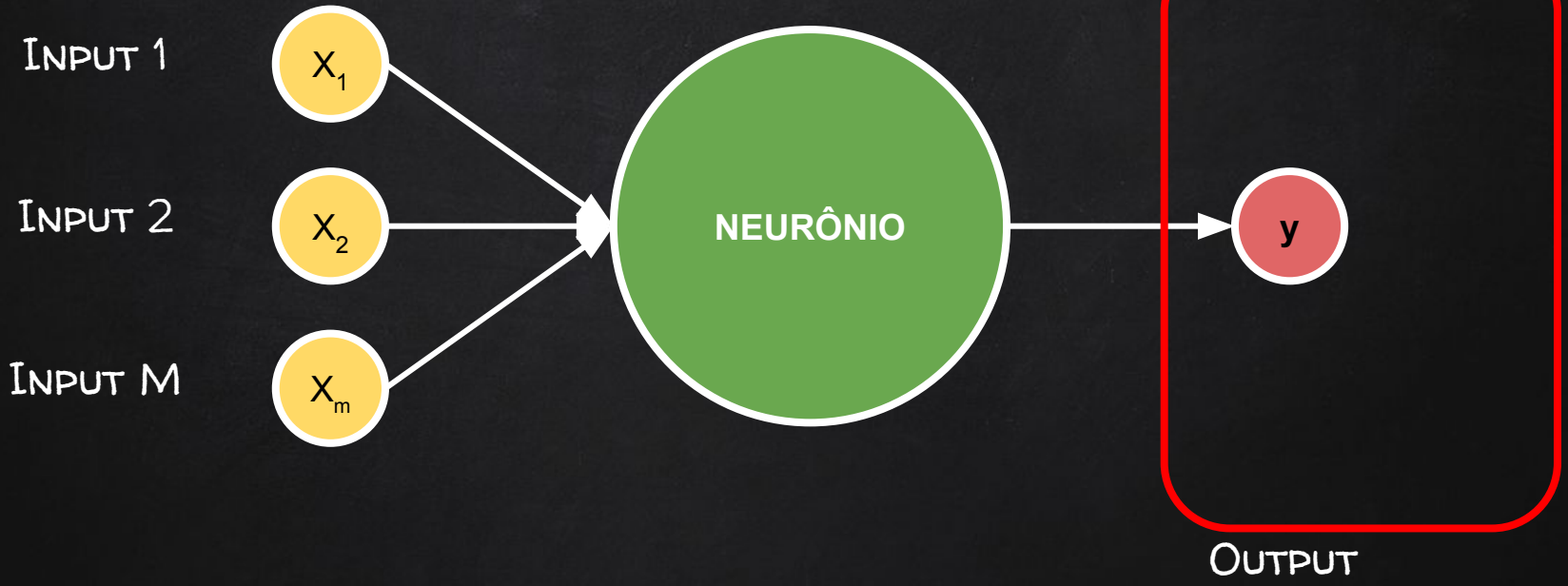
Padronização
Normalização



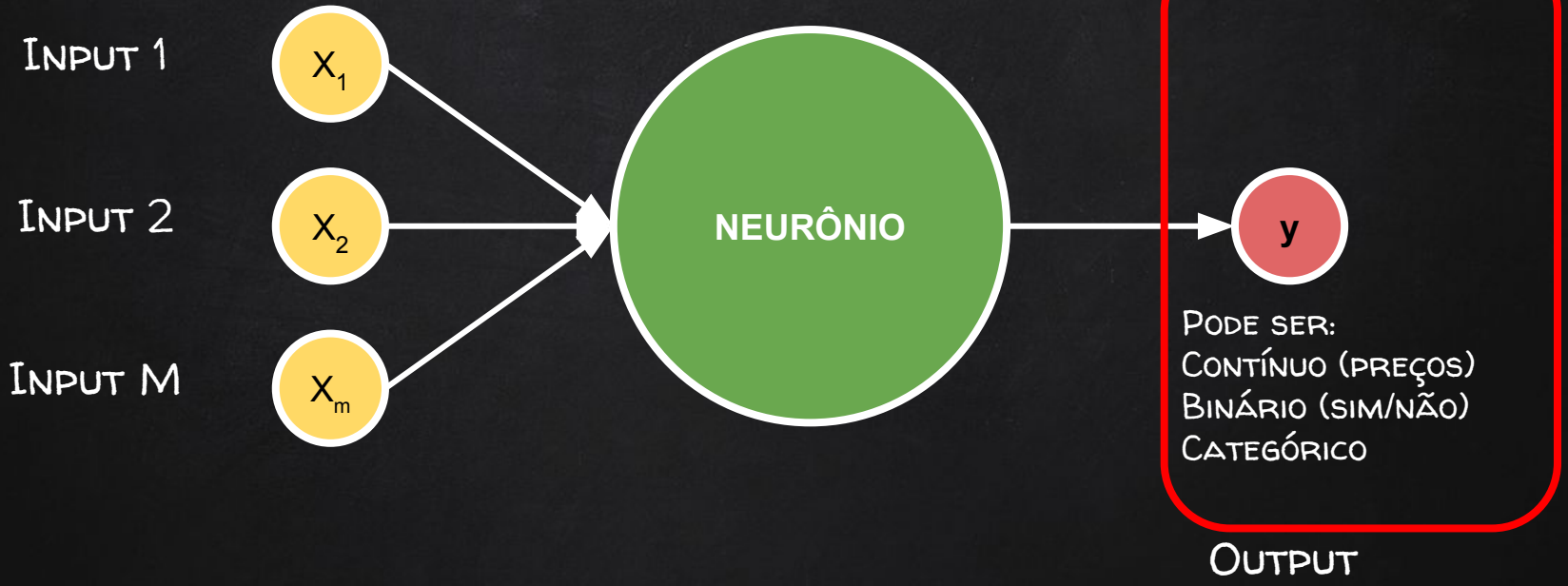
Efficient BackProp - Yann Lecun

http://yann.lecun.com/exdb/publis/pdf/lecun-98b.pdf

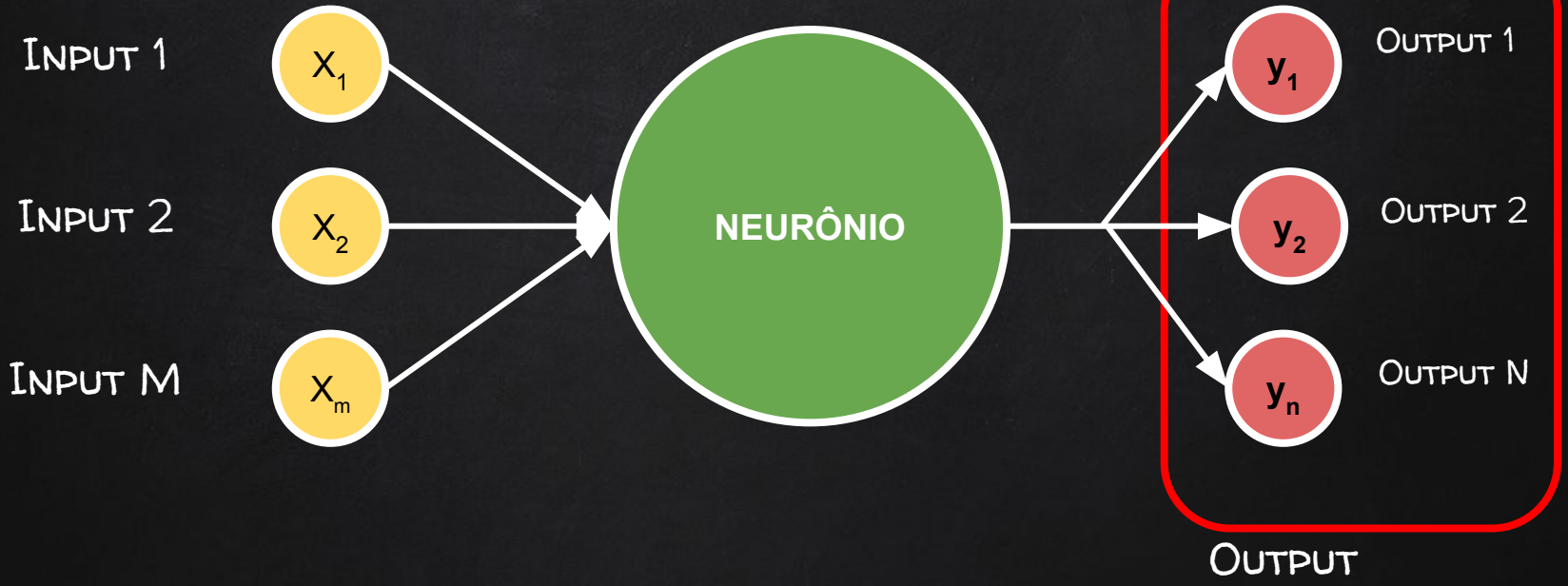
Neurônio



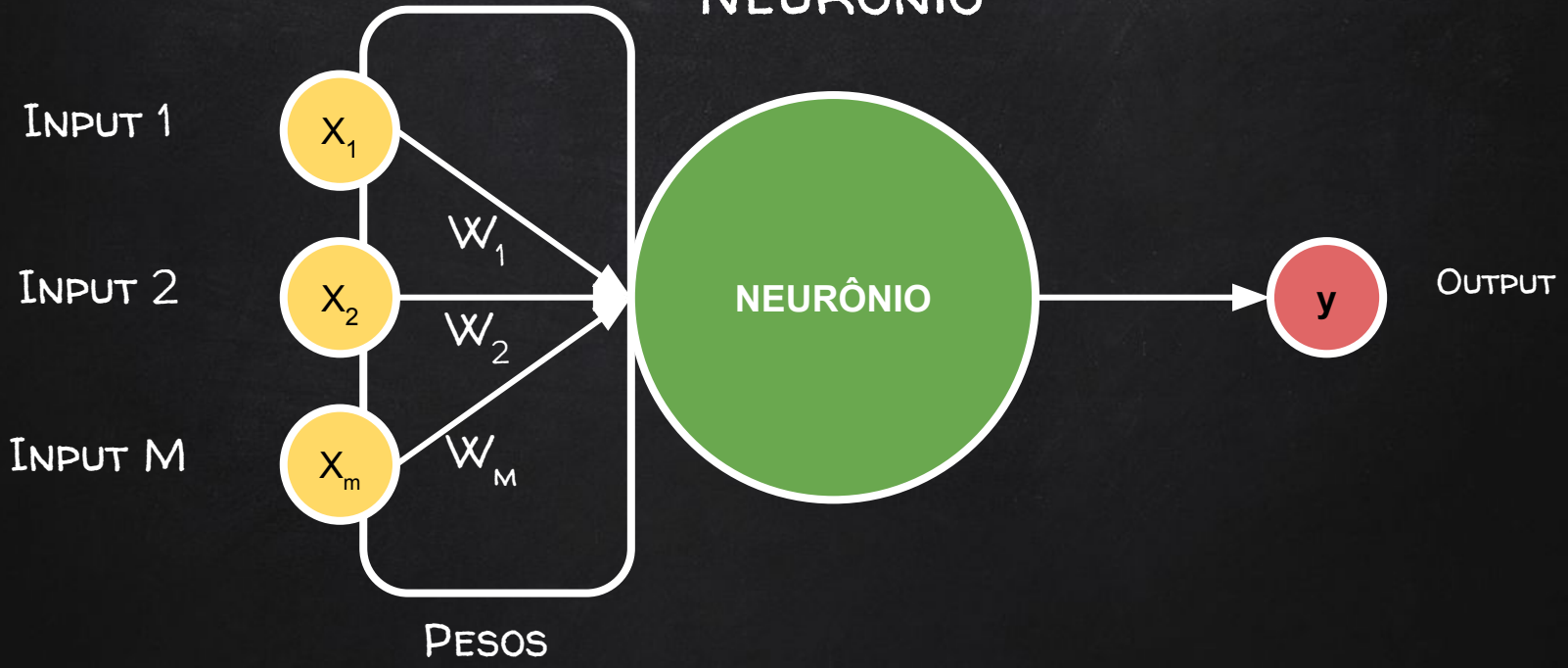
Neurônio


Input 1
Input 2
Input M
X_1
X_2
X_m
NEURÔNIO
y
Pode ser:
Contínuo (preços)
Binário (sim/não)
Categórico
Output

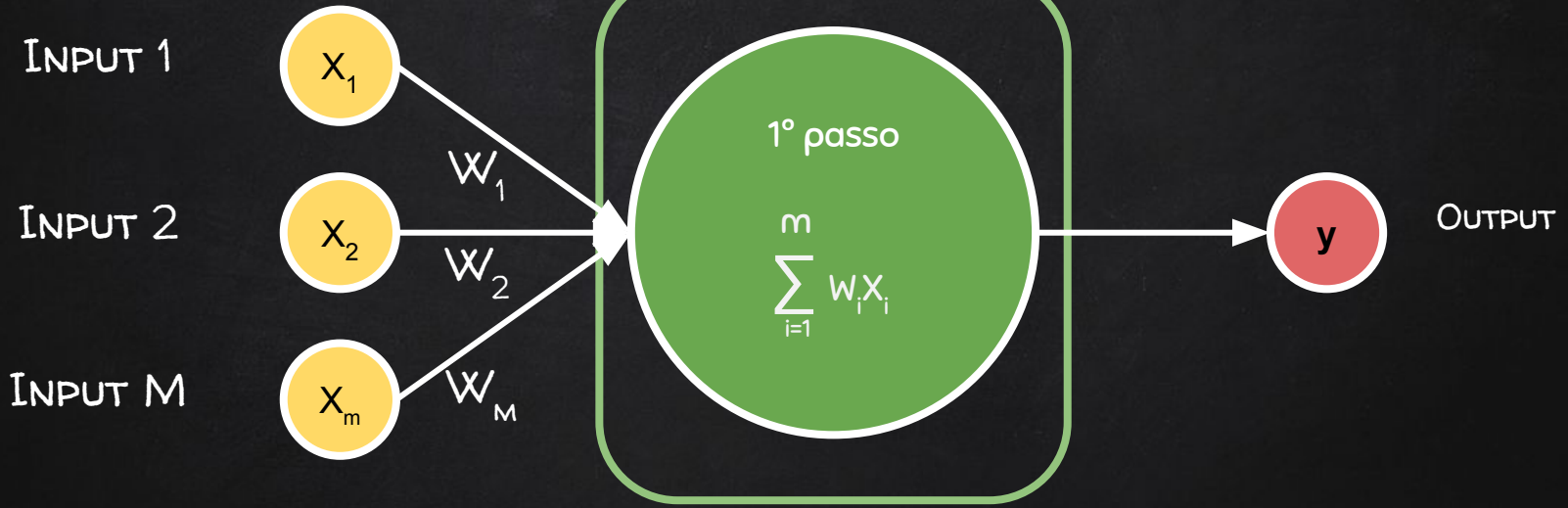
Neurônio

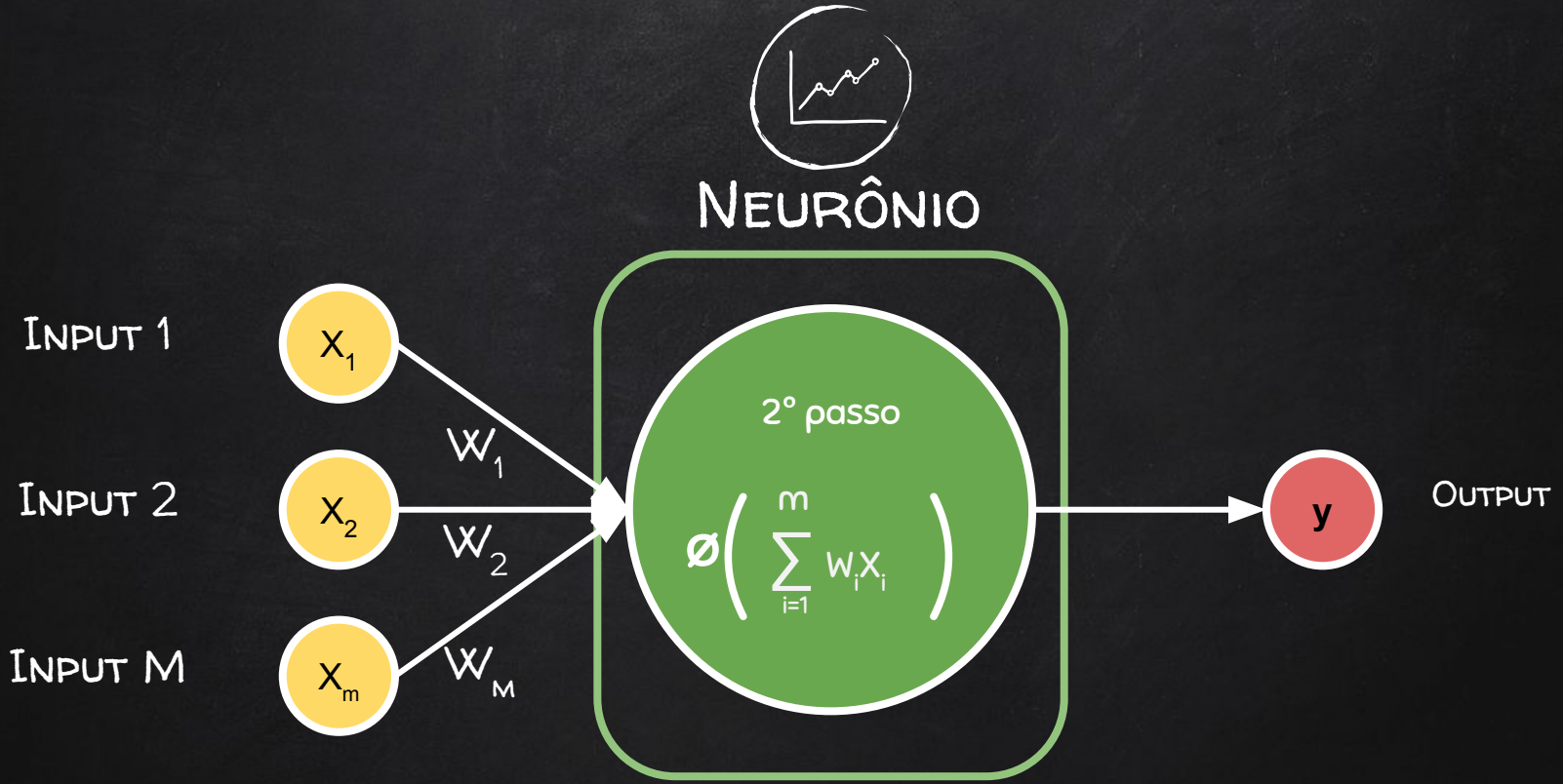

Input 1
Input 2
Input M
x_1
x_2
x_m
NEURÔNIO
y_1
y_2
y_n
Output 1
Output 2
Output N
Output

Neurônio

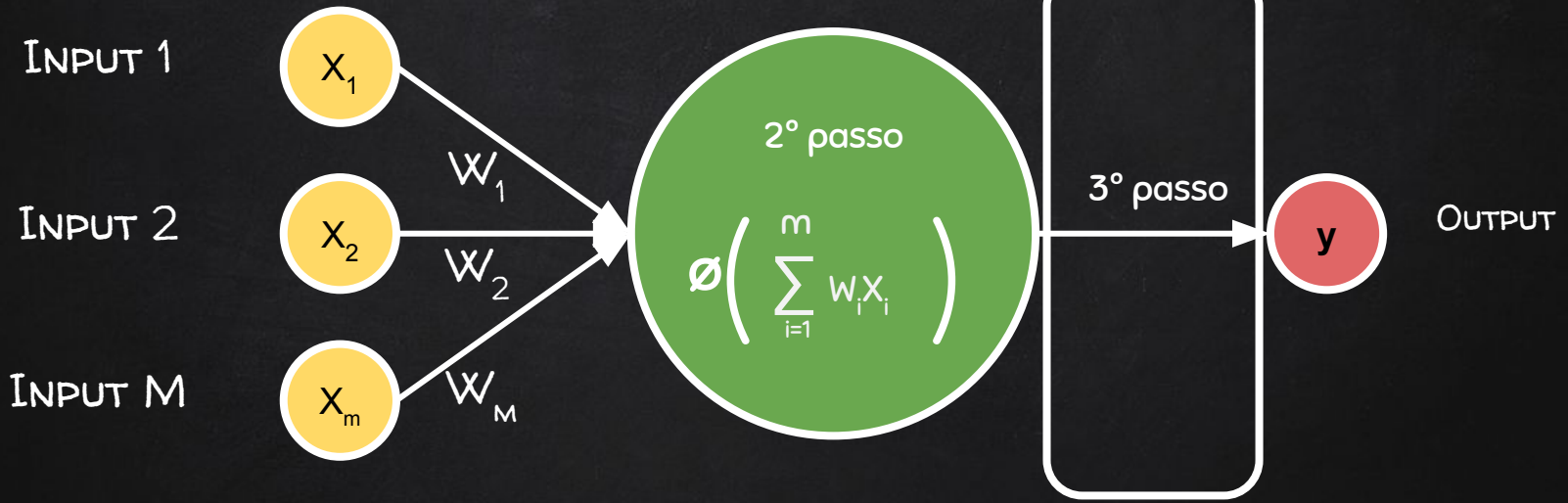

Input 1
Input 2
Input M
x_1
x_2
x_m
w_1
w_2
w_M
NEURÔNIO
y
Output
Pesos

Neurônio


Input 1
Input 2
Input M
x_1
x_2
x_m
w_1
w_2
w_M
1º passo
$\sum_{i=1}^{m} w_i x_i$
y
Output


Neurônio
Input 1
Input 2
Input M
x_1
x_2
x_m
w_1
w_2
w_M
2° passo
$\phi\left(\sum_{i=1}^{m} w_i x_i\right)$
y
Output

Neurônio

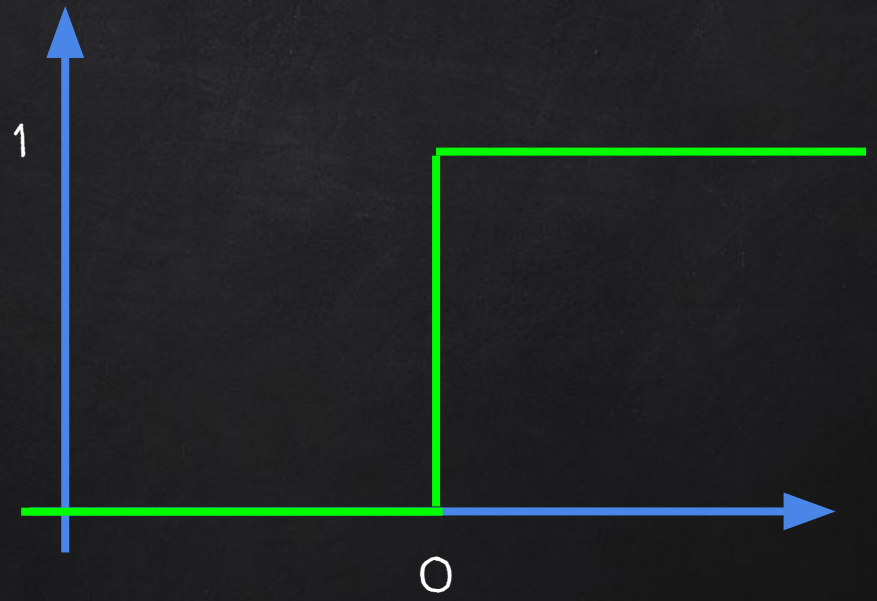

Input 1
Input 2
Input M
x_1
x_2
x_m
w_1
w_2
w_M
2° passo
$\phi\left(\sum_{i=1}^{m} w_i x_i\right)$
3° passo
y
Output

Gostaram?

Funções de Ativação

Redes Neurais Artificiais

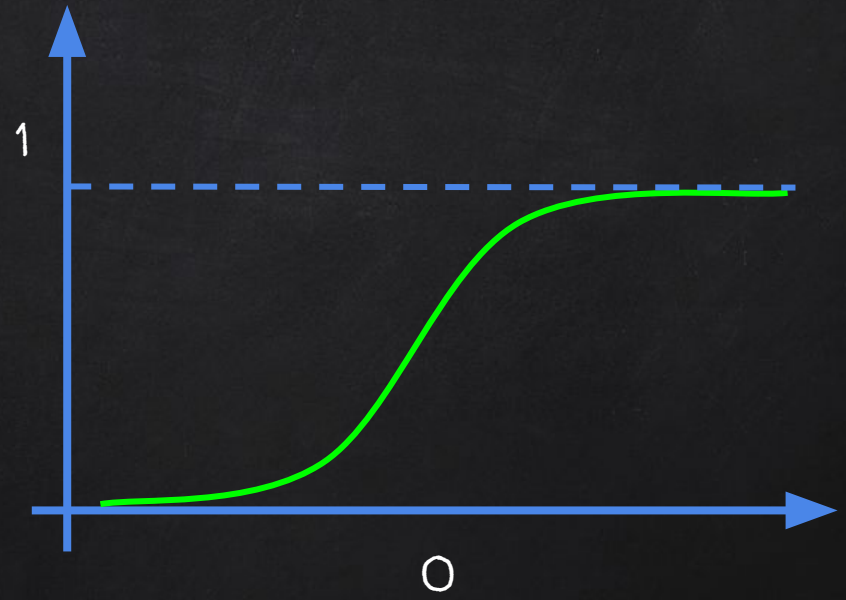
Threshold Function



$$\phi(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ 0 \text{ if } x < 0 \end{cases}$$

$$\sum_{i=1}^{m} w_i x_i$$

Sigmoid Function



$$\phi(x) = \frac{1}{1 + e^{-x}}$$

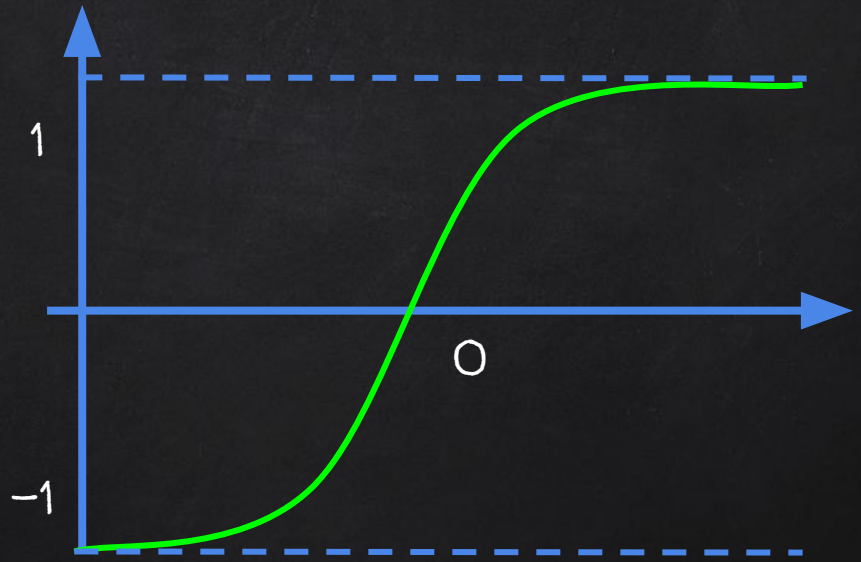
$$\sum_{i=1}^{m} w_i x_i$$

Rectifier Function

$$\phi(x) = \max(x, 0)$$

$$\sum_{i=1}^{m} w_i x_i$$

Hyperbolic Tangent (tanh)



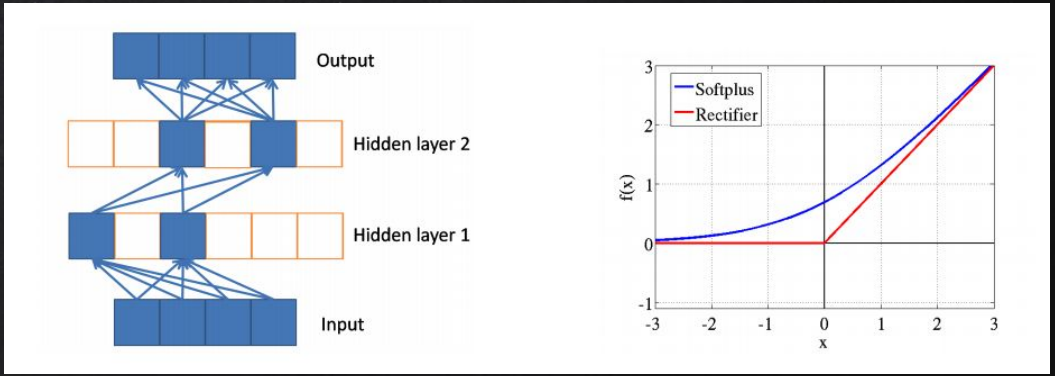
$$\phi(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

$$\sum_{i=1}^{m} w_i x_i$$

Leitura Opcional

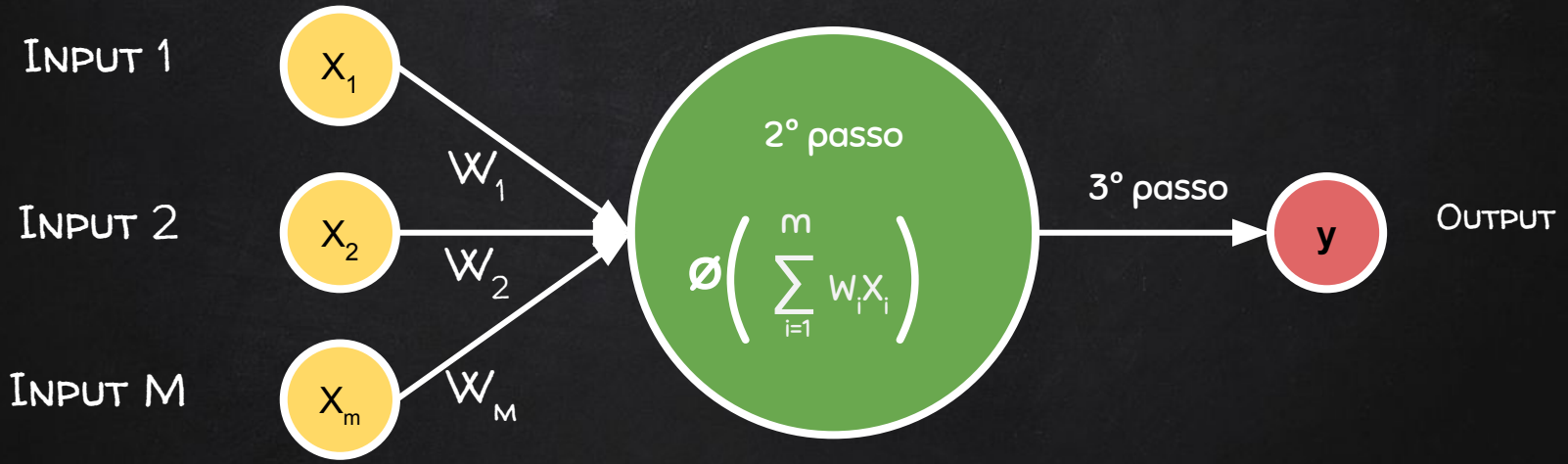
DEEP SPARSE RECTIFIER NEURAL NETWORKS

Por Xavier Glorot(2011)



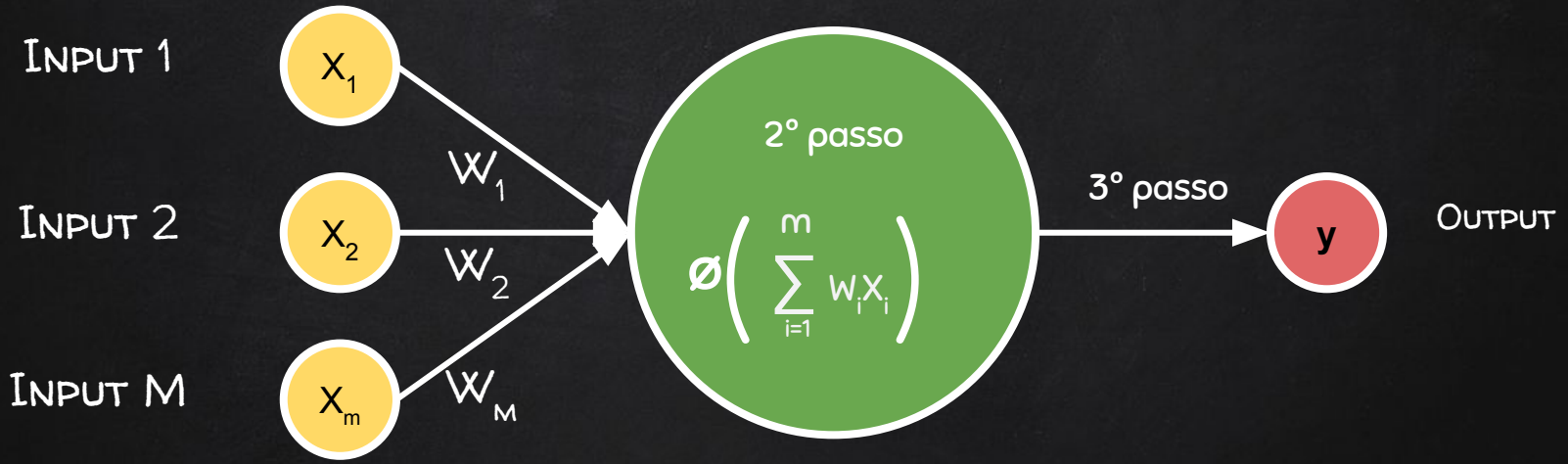
http://proceedings.mlr.press/v15/glorot11a/glorot11a.pdf

Exercício



Se a Variável Dependente
é binária (y=0 ou y=1) ??

Exercício

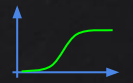


Se a Variável Dependente é binária (y=0 ou y=1) ??

- Threshold

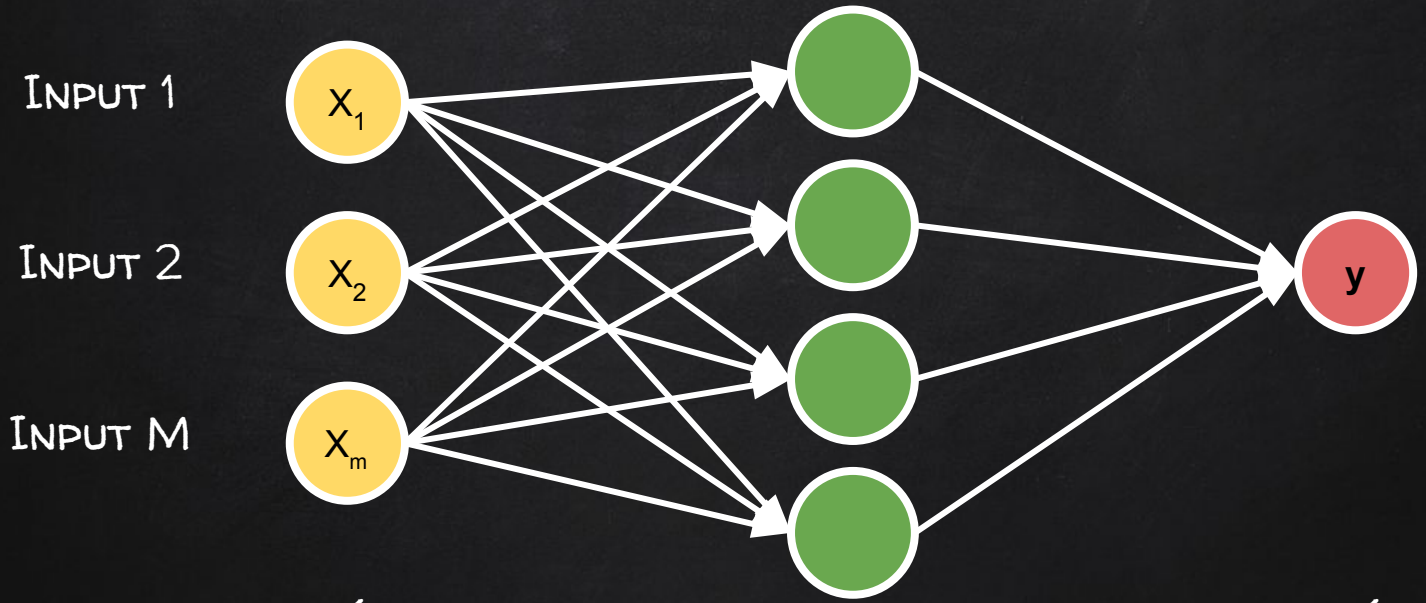
$$y = \phi\left(\sum_{i=1}^{m} w_i x_i\right)$$

- Sigmoid

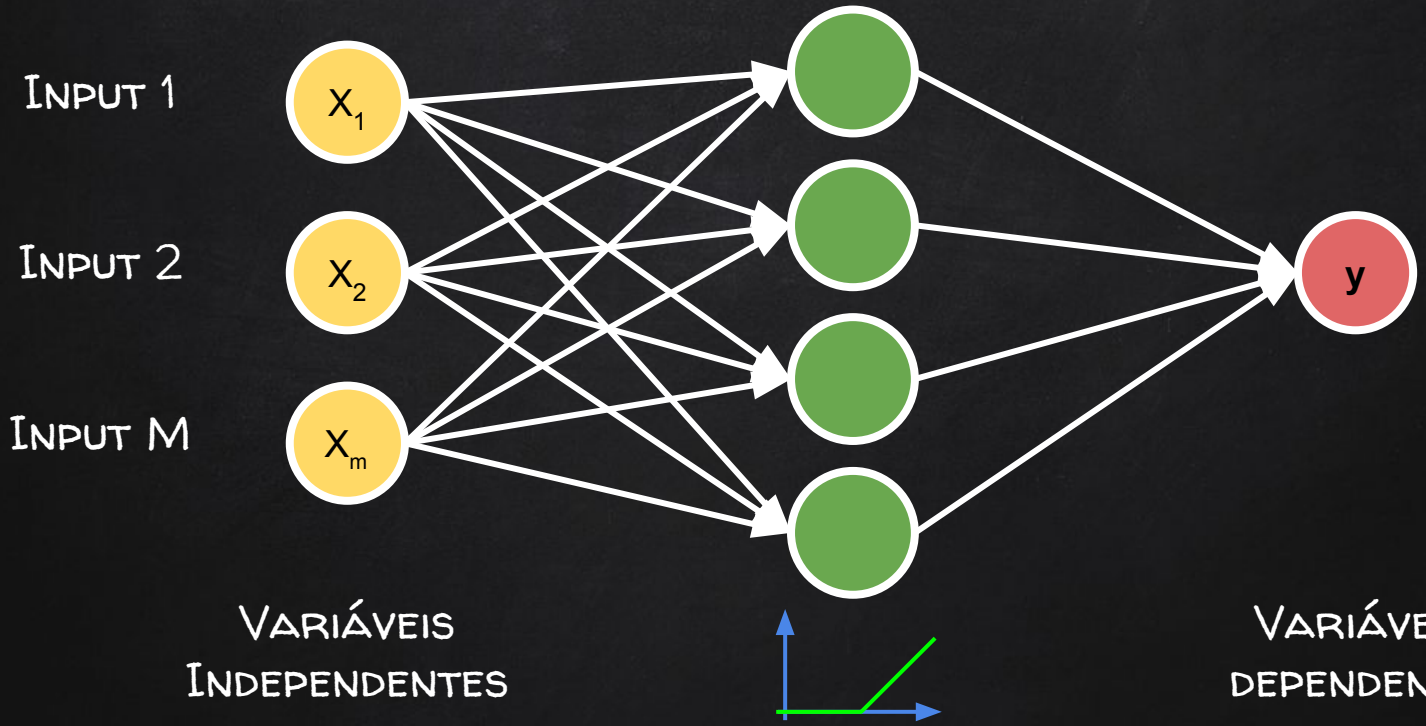


$$P(y=1) = \phi\left(\sum_{i=1}^{m} w_i x_i\right)$$

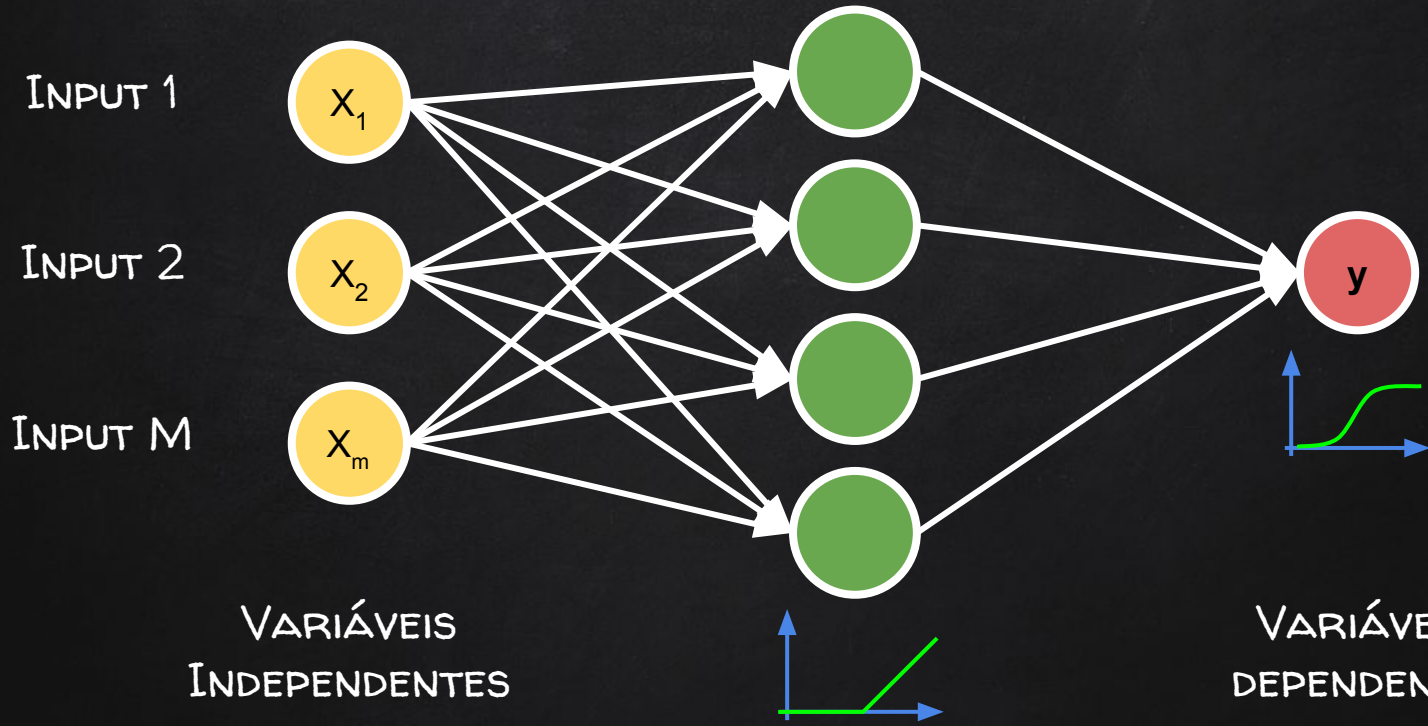
Exemplo


Input 1
Input 2
Input M
x_1
x_2
x_m
y
Variáveis Independentes
Variável dependente

Exemplo


Input 1
Input 2
Input M
x_1
x_2
x_m
y
Variáveis Independentes
Variável dependente

Exemplo


Input 1
Input 2
Input M
x_1
x_2
x_m
y
Variáveis Independentes
Variável dependente

Entendeu?

Como Funcionam?

Redes Neurais Artificiais

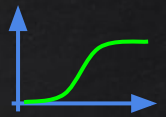
Preço do Imóvel

Quartos x_1

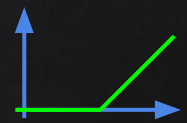
Metros^2 x_2

Idade x_3

Distância x_4



y



Variável dependente

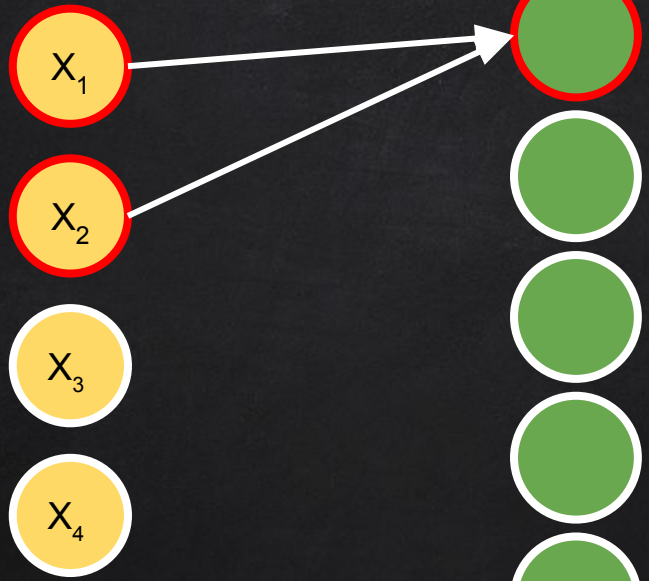
Preço do Imóvel

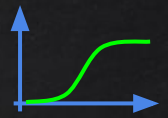
Quartos

Metros^2

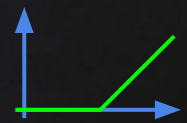
Idade

Distância



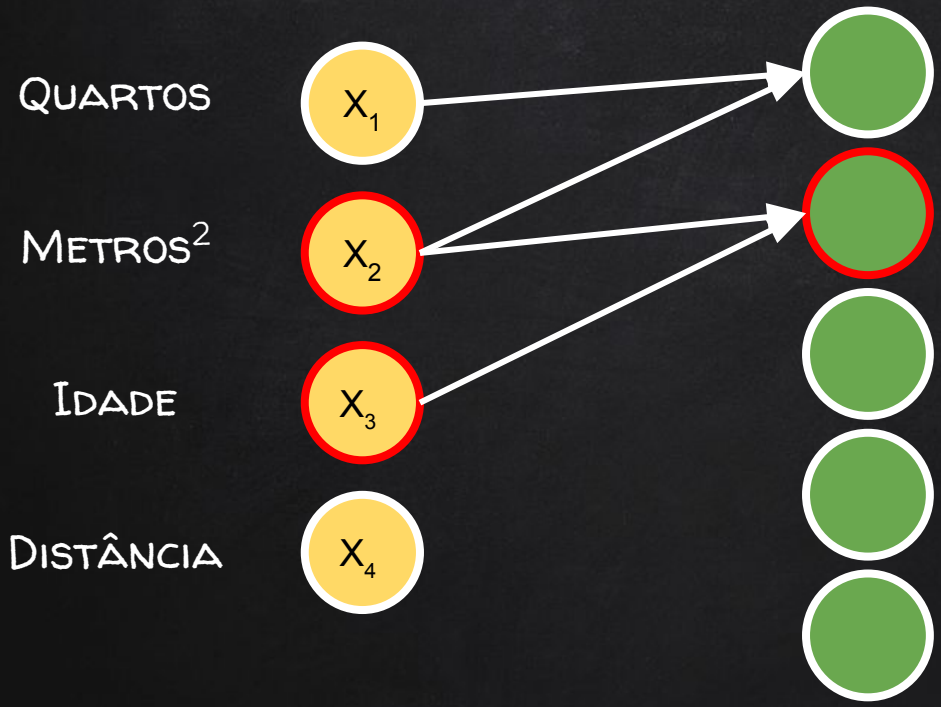


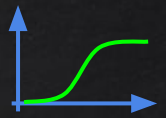
y

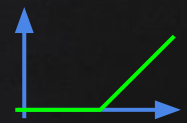


Variável dependente

Preço do Imóvel

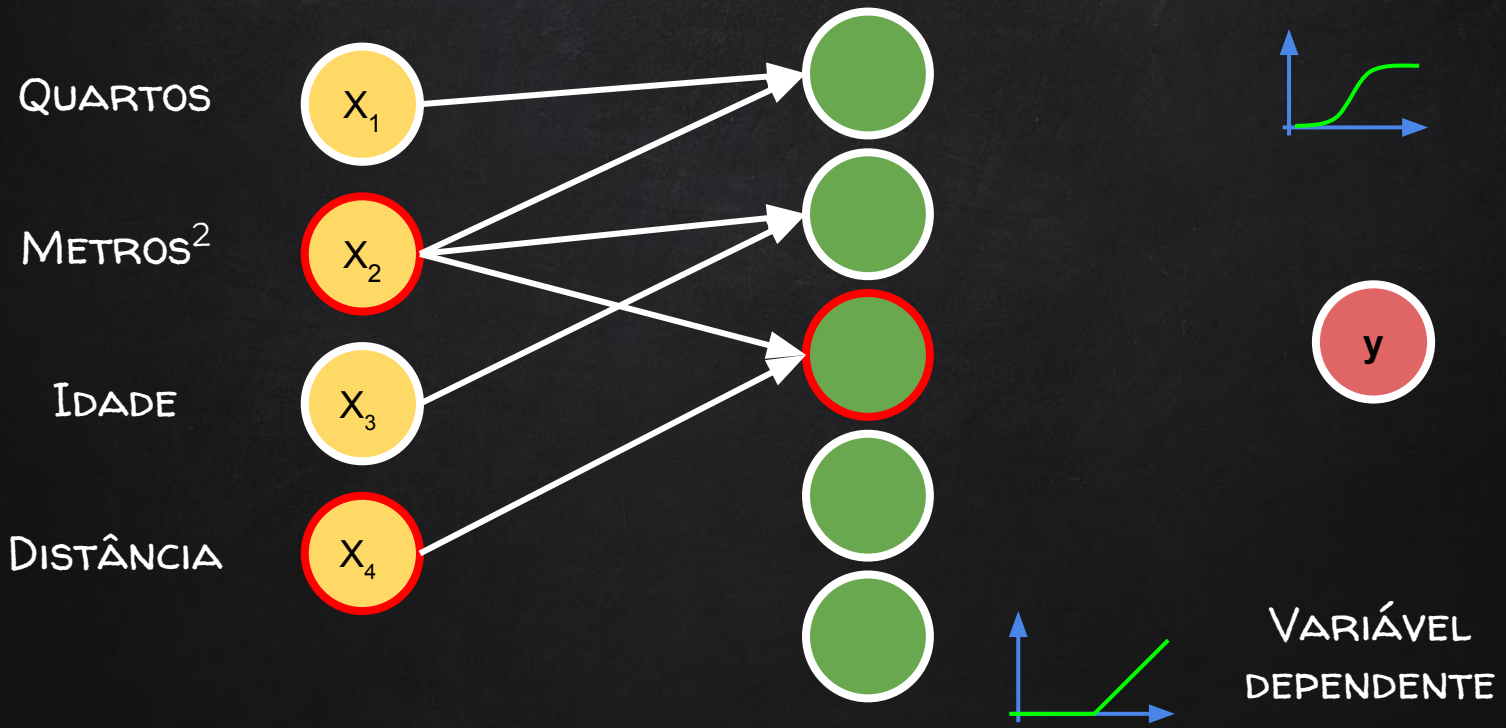






Variável dependente

Preço do Imóvel



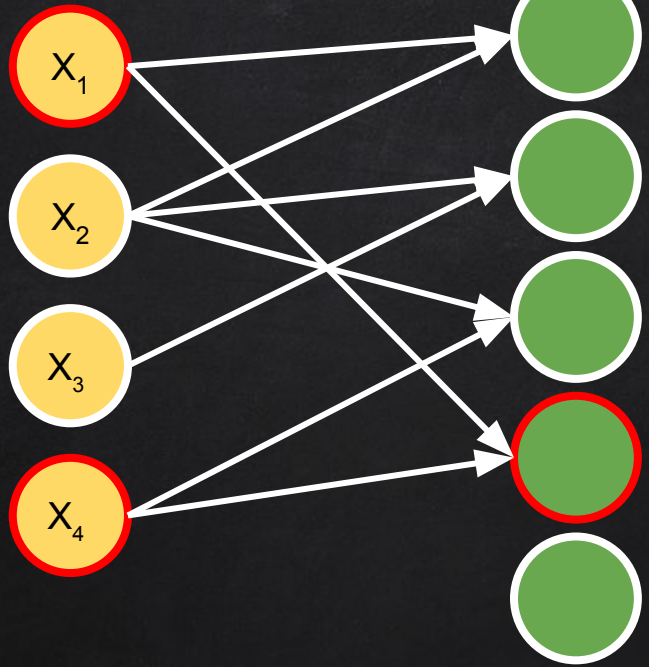
Preço do Imóvel

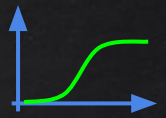
Quartos

Metros^2

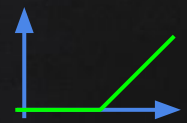
Idade

Distância





y



Variável dependente

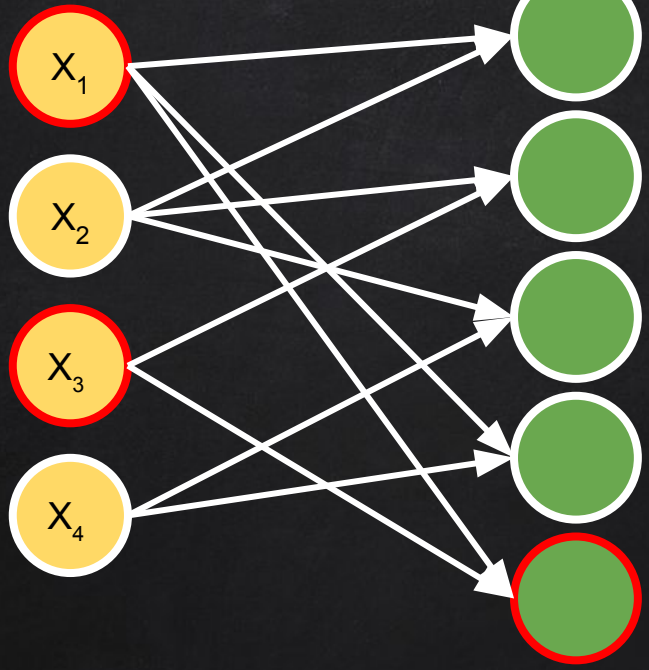
Preço do Imóvel

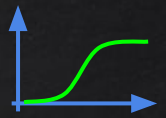
Quartos

Metros^2

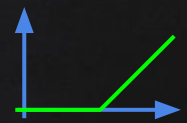
Idade

Distância



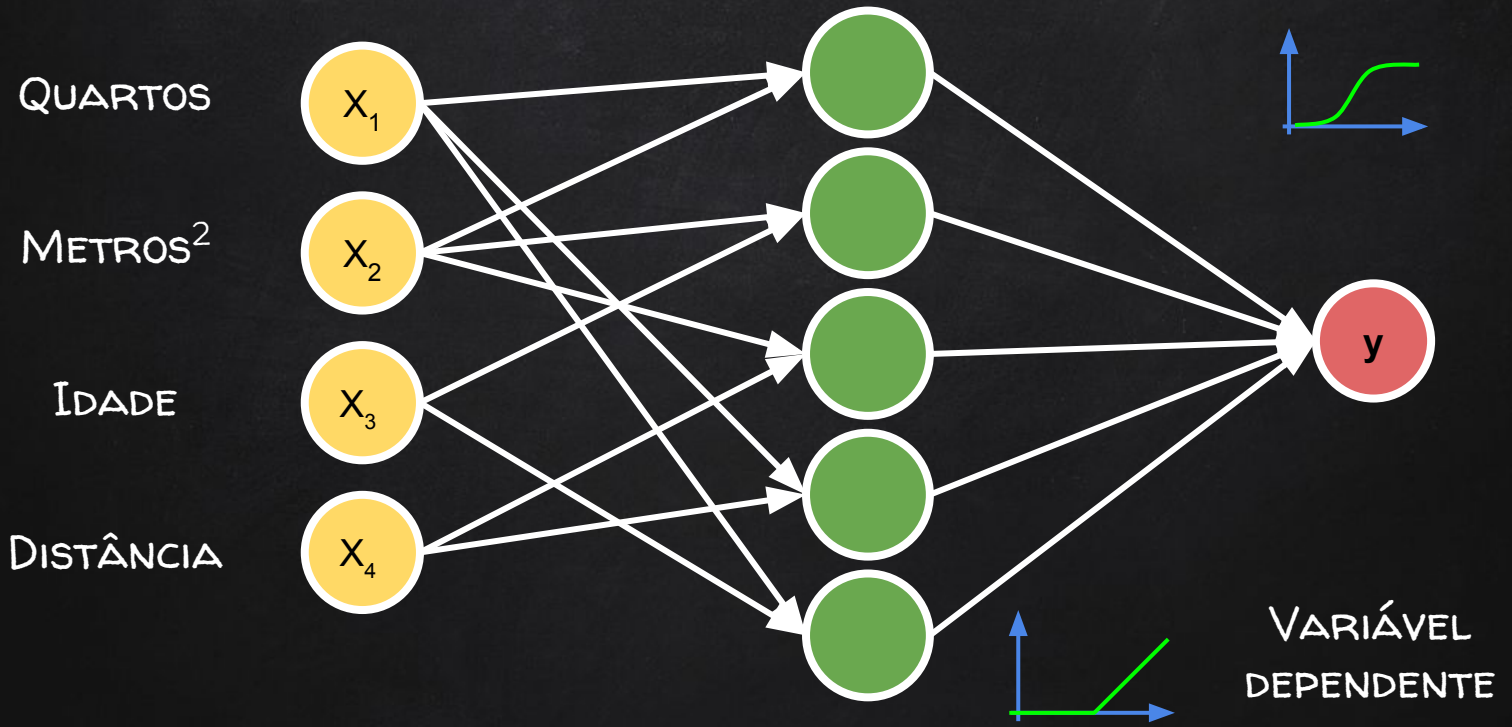


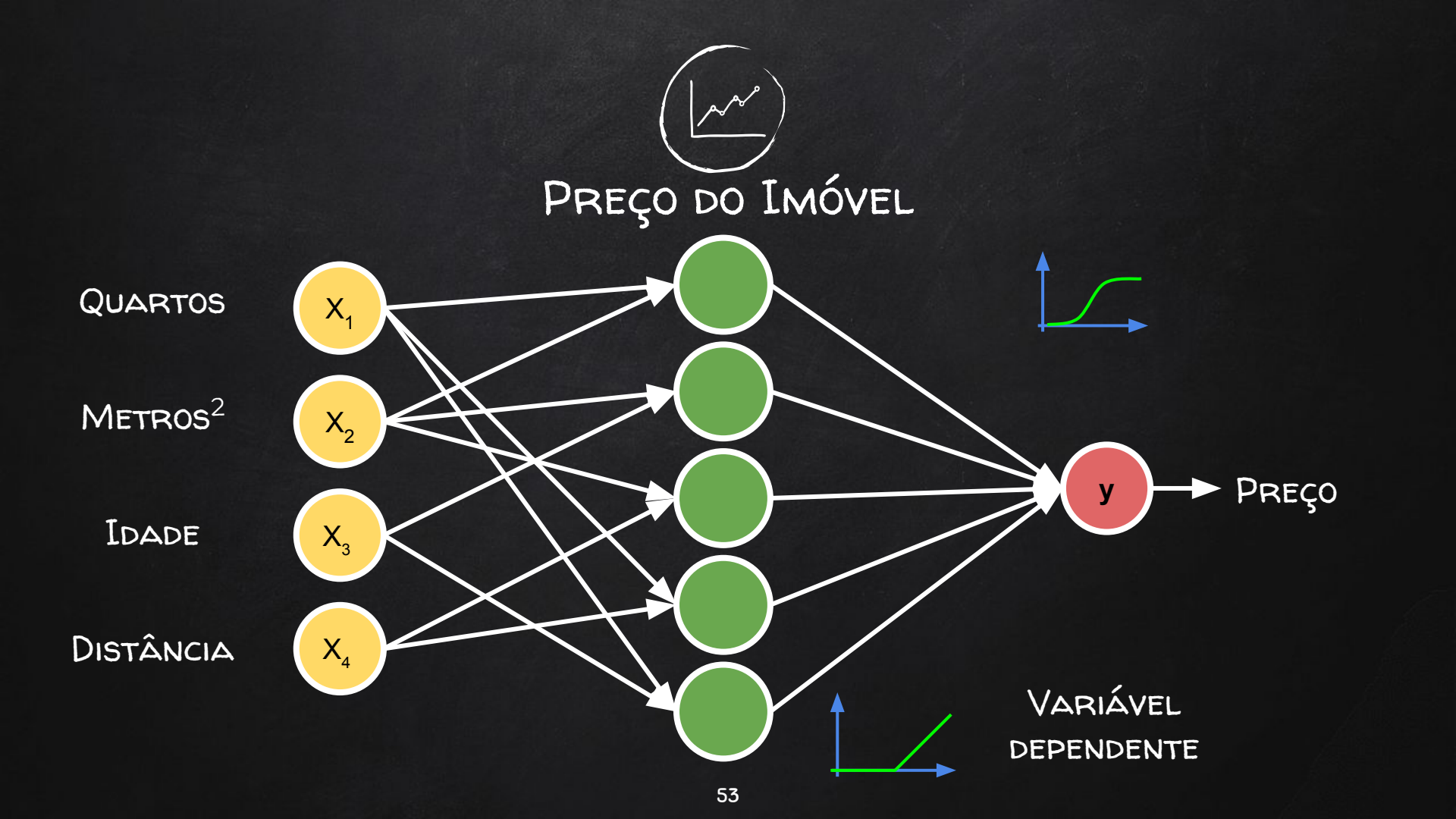
y



Variável dependente

Preço do Imóvel

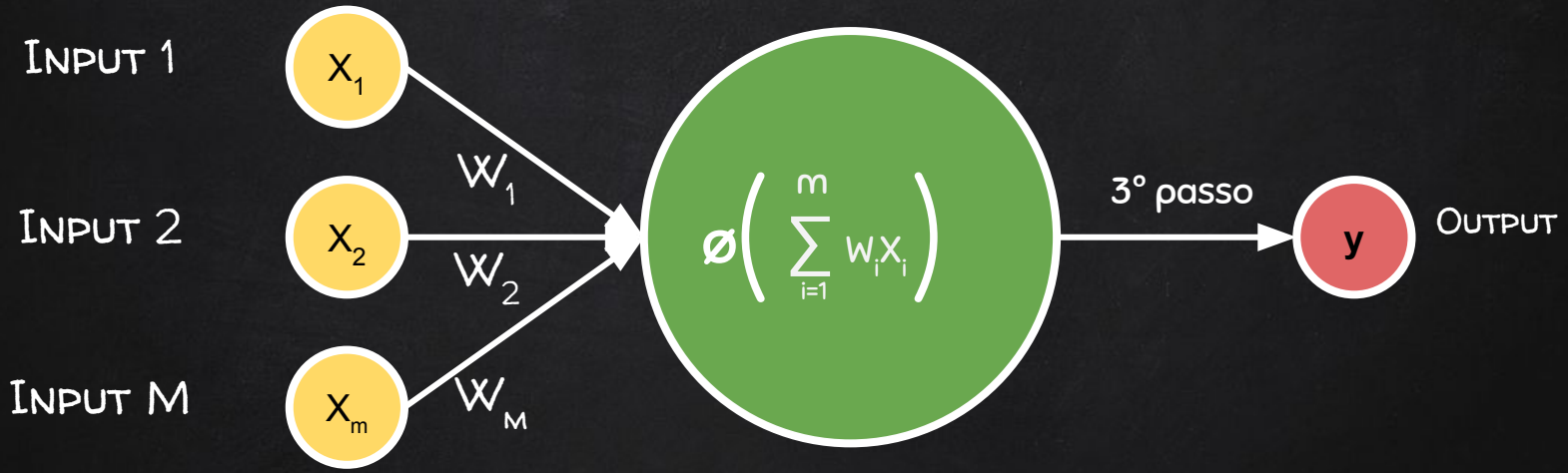


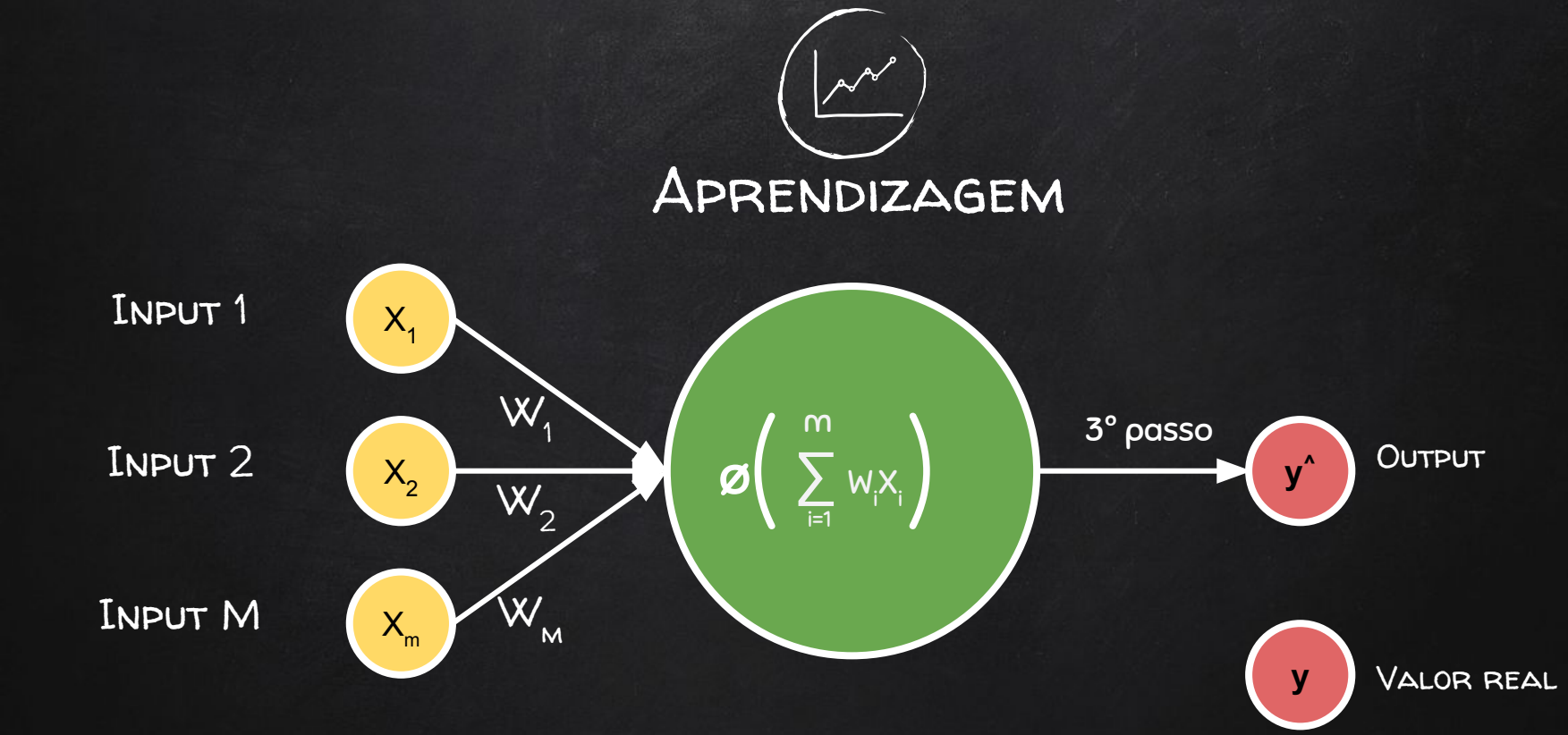

Preço do Imóvel
Quartos
Metros²
Idade
Distância
X1
X2
X3
X4
y
Preço
Variável dependente

Como Aprendem?

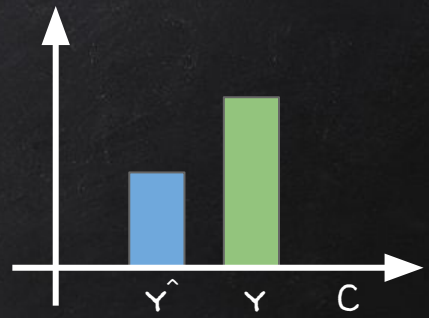
Redes Neurais Artificiais

Aprendizagem




Aprendizagem
Input 1
Input 2
Input M
x_1
x_2
x_m
w_1
w_2
w_M
$\phi\left(\sum_{i=1}^{m} w_i x_i\right)$
3° passo
y^
Output
y
Valor real

Aprendizagem



Input 1

Input 2

Input M

x_1

x_2

x_m

w_1

w_2

w_M

$$\phi\left(\sum_{i=1}^{m} w_i x_i\right)$$

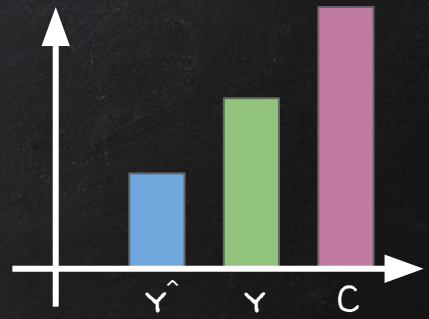
3° passo

$\hat{y}$ Output

$C = \frac{1}{2}(\hat{Y} - Y)^2$

y Valor real

Aprendizagem



Input 1

Input 2

Input M

x_1

x_2

x_m

w_1

w_2

w_M

$$\phi\left(\sum_{i=1}^{m} w_i x_i\right)$$

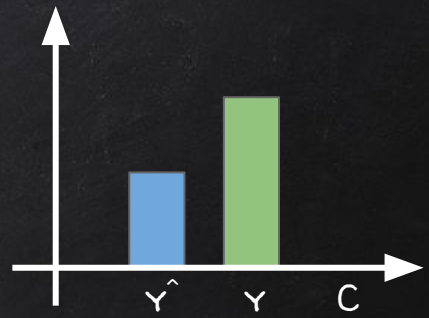
3° passo

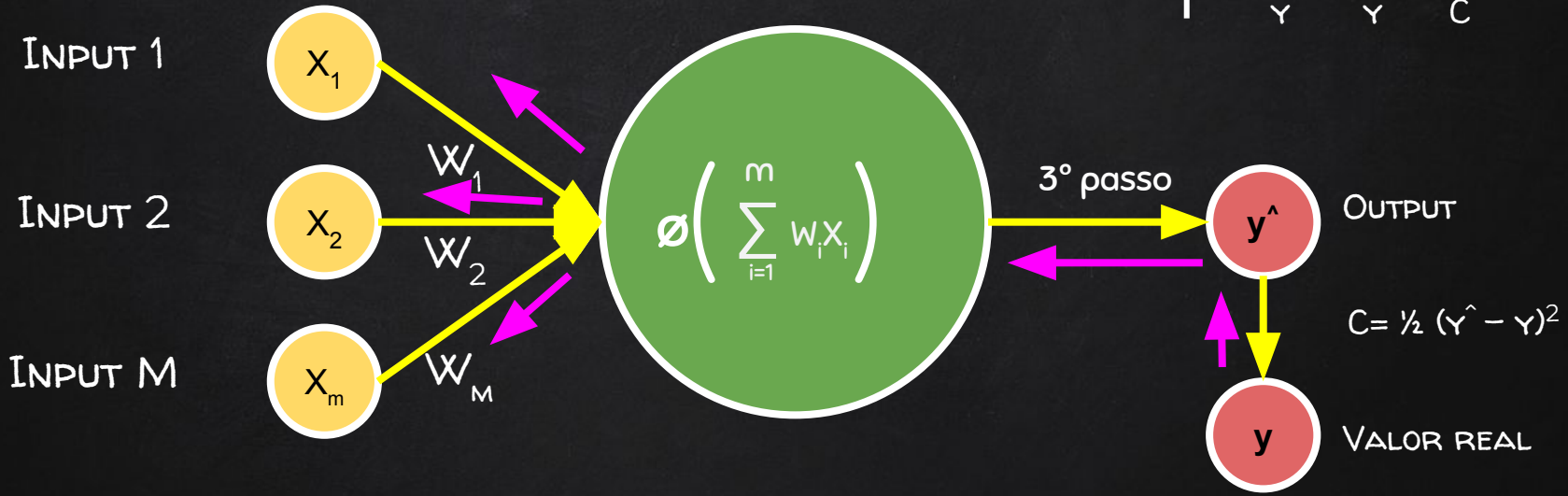
$\hat{y}$ Output

$C = \frac{1}{2}(\hat{Y} - Y)^2$

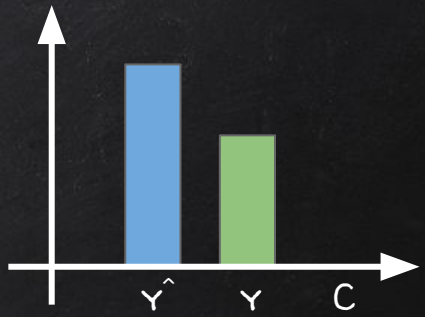
y Valor real

Aprendizagem





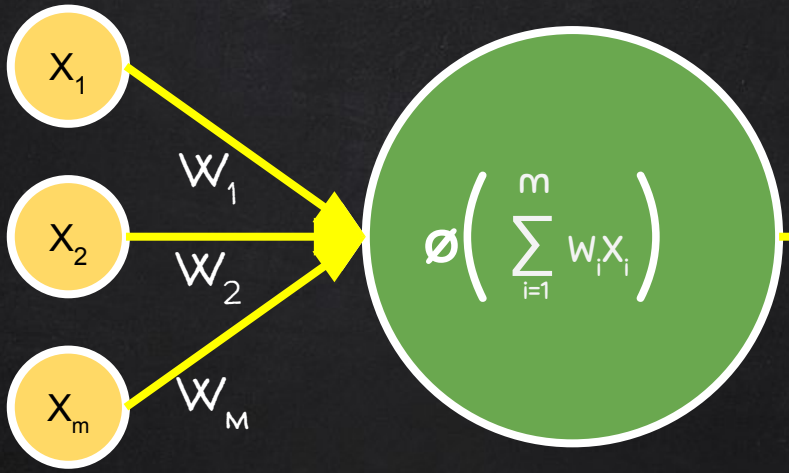
Aprendizagem



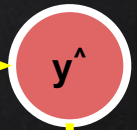
Input 1

Input 2

Input M



3° passo

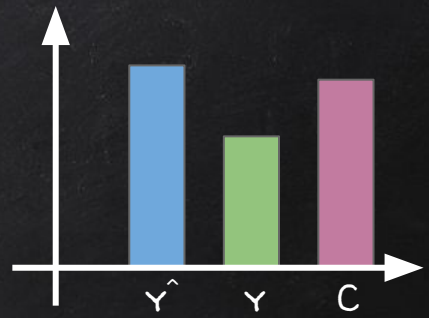


Output

$C = \frac{1}{2}(\hat{Y} - Y)^2$

Valor real

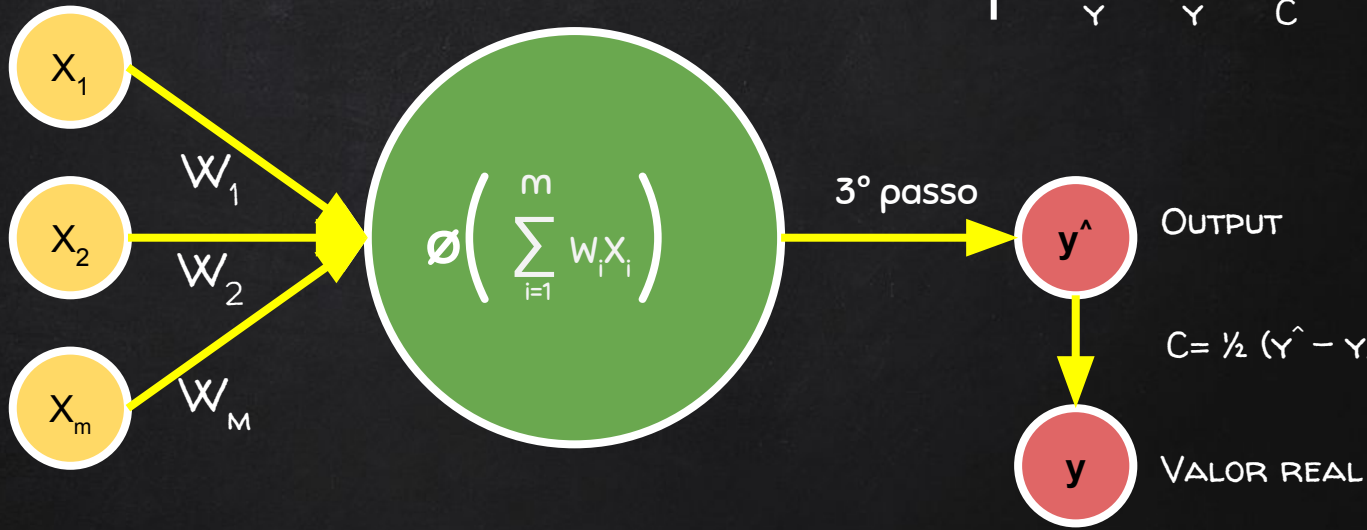
Aprendizagem



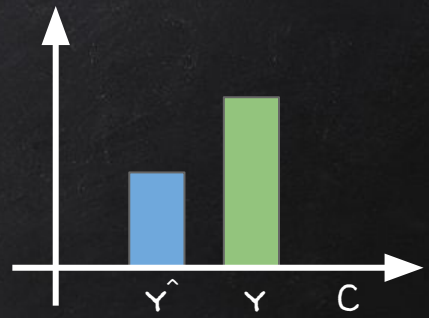
Input 1

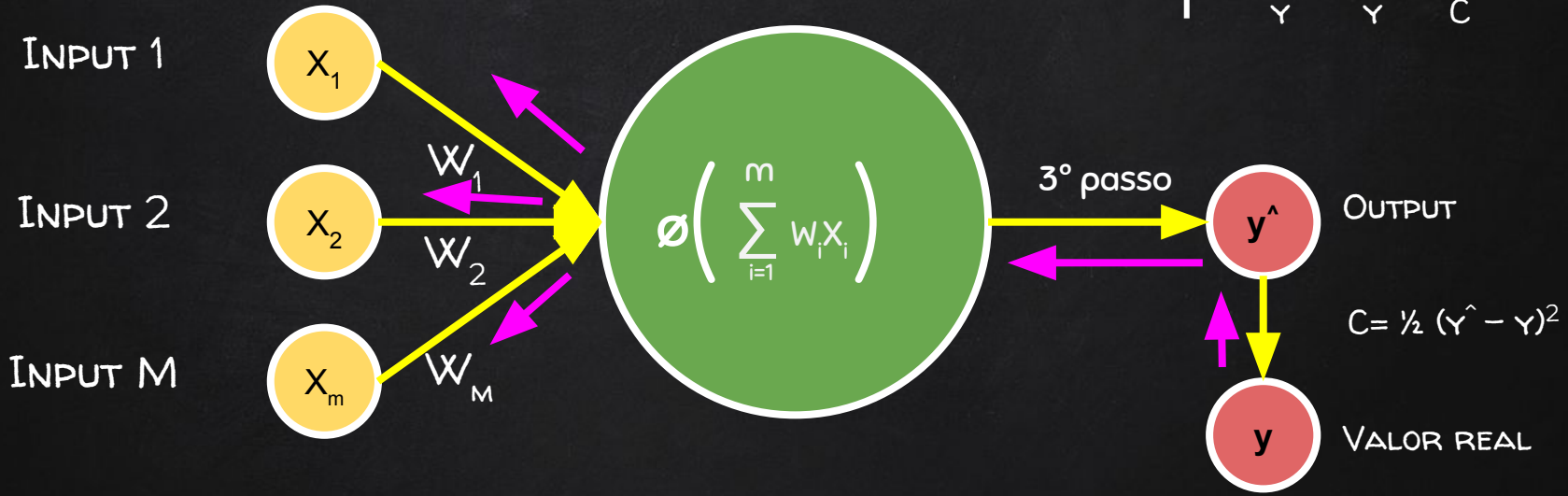
Input 2

Input M

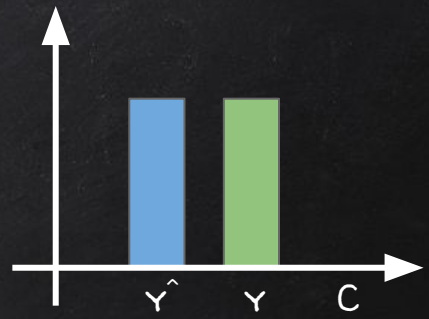


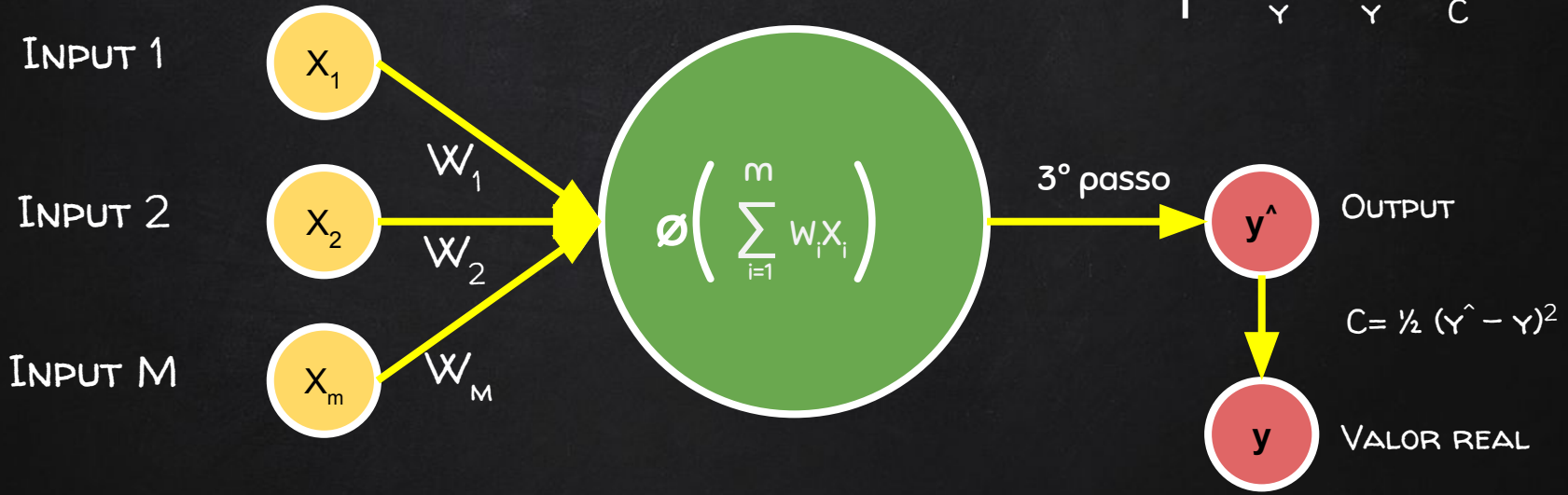
Aprendizagem





Aprendizagem





Épocas

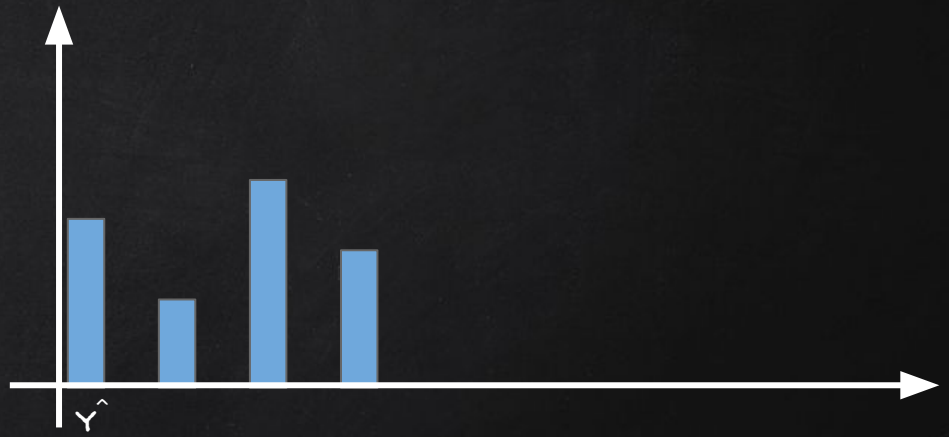
Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%

Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%

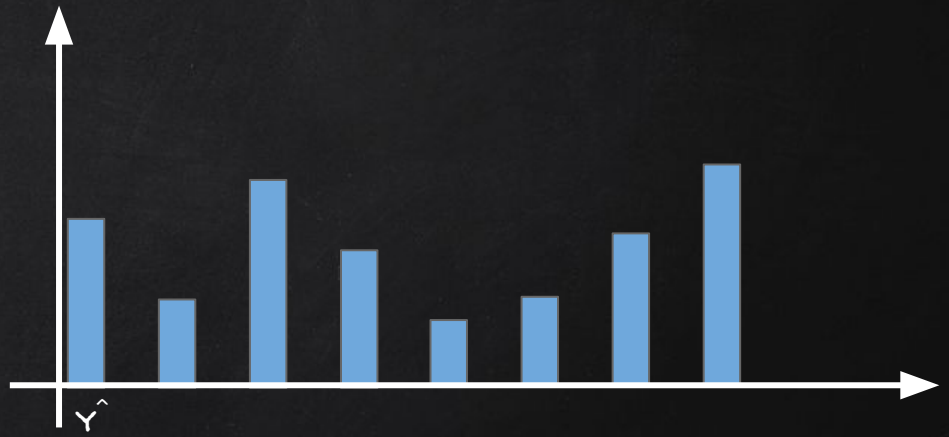
Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%

Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%

Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%

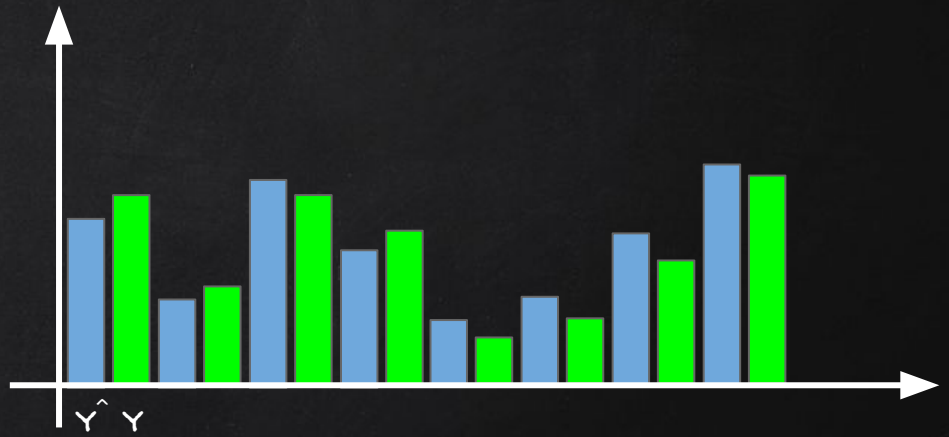


Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%



Épocas

Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%



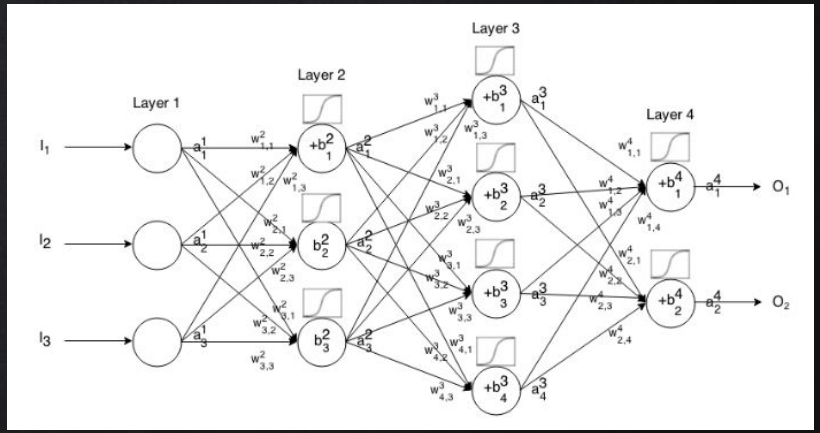
Épocas

Row ID	Study Hrs	Sleep Hrs	Quiz	Exam
1	12	6	78%	93%
2	22	6.5	24%	68%
3	115	4	100%	95%
4	31	9	67%	75%
5	0	10	58%	51%
6	5	8	78%	60%
7	92	6	82%	89%
8	57	8	91%	97%

$$C = \sum \tfrac{1}{2} (\hat{Y} - Y)^2$$

Aprendizagem

A list of cost functions used in neural networks, alongside applications CrossValidated (2015)

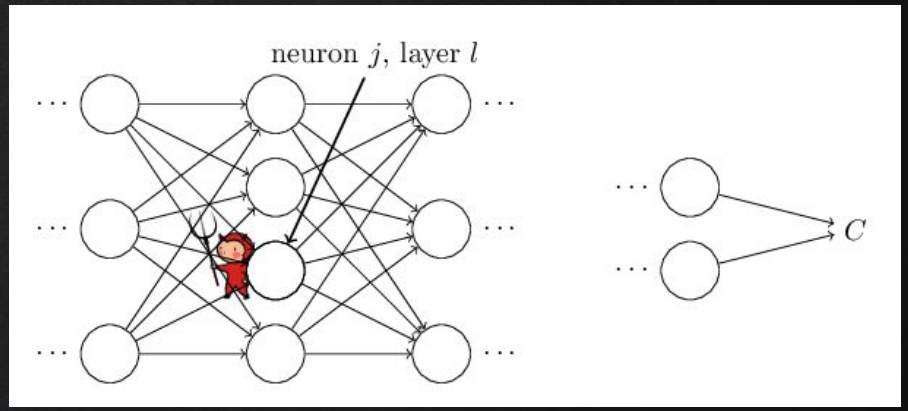


https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-used-in-neural-networks-alongside-applications

Aprendizagem

How the backpropagation algorithm works

Michael Nielsen (2015)



http://neuralnetworksanddeeplearning.com/chap2.html

Should a self-driving car kill the baby or the grandma?
Depends on where you're from.

https://www.technologyreview.com/s/612341/a-global-ethics-study-aims-to-help-ai-solve-the-self-driving-trolley-problem/

Prática
Construindo sua ANN

O cliente vai deixar o banco?

Pré-processamento

https://github.com/deeplearningunb/building-ann

Vamos Exercitar?

Vamos Exercitar?

```
54  # Adding the input layer and the first hidden layer
55  classifier.add(Dense(units = 6, kernel_initializer = 'uniform', activation = 'relu', input_dim = 11))
56
57  # Adding the second hidden layer
58  classifier.add(Dense(units = 6, kernel_initializer = 'uniform', activation = 'relu'))
59
60  # Adding the output layer
61  classifier.add(Dense(units = 1, kernel_initializer = 'uniform', activation = 'sigmoid'))
62
63  # Compiling the ANN
64  classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
65
66  # Fitting the ANN to the Training set
67  classifier.fit(X_train, y_train, batch_size = 10, epochs = 100)
```

1. Crie uma branch com seu nome
2. **Troque a função de ativação**
3. Compile a rede
4. Commit do resultado (NA SUA BRANCH)

Obrigado!

Dúvidas?

http://bit.ly/dl-unb03
https://t.me/DeepLearningUnB
@diegodorgam

Credits

Special thanks to all the people who made and released these awesome resources for free:

- ✘ Presentation template by SlidesCarnival
- ✘ Photographs by Unsplash