MINITEL

MACHINE LEARNING

04 NEURAL NETWORKS

THÉO GACHET

UNE SÉRIE DE FORMATIONS

PARTIE 1 :
SUPERVISED MACHINE LEARNING : REGRESSION AND CLASSIFICATION

Introduction to Machine Learning
Regression with multiple input
Classification

PARTIE 2 :
ADVANCED LEARNING ALGORITHMS

Neural Networks
Neural Networks training
Advice for applying Machine Learning
Decision Trees

PARTIE 3 :
UNSUPERVISED LEARNING, RECOMMENDERS, REINFORCEMENT LEARNING

Unsupervised Learning
Recommander Systems
Reinforcement Learning

04 NEURAL NETWORKS

NEURAL NETWORKS INTUITION

Neurons and the brain
Demand Prediction
Example : Recognizing Images

NEURAL NETWORK MODEL

Neural network layer
More complex neural networks
Inference : making predictions (forward propagation)
Lab : Neurons and Layers

TENSORFLOW IMPLEMENTATION

Inference in code
Data in TensorFlow
Building a neural network
Lab : Coffee Roasting in TensorFlow

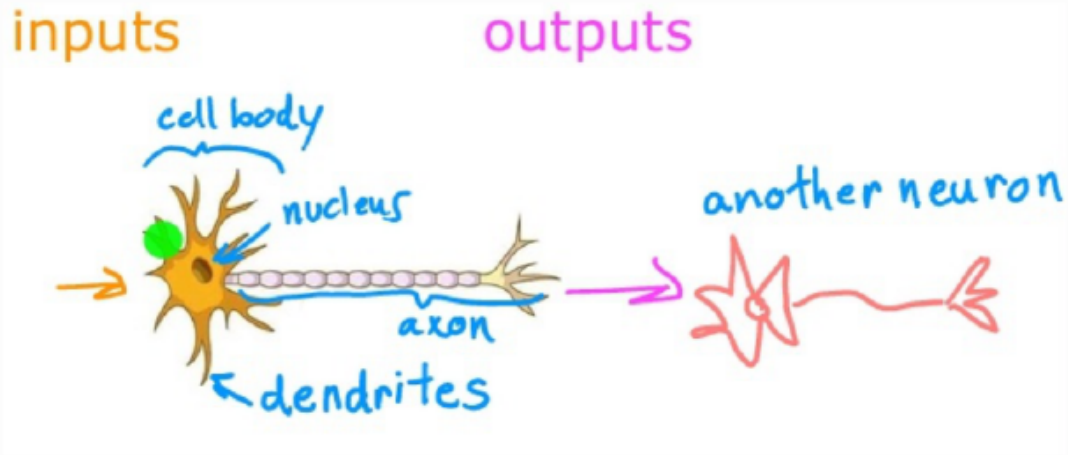
01

NEURAL NETWORKS INTUITION

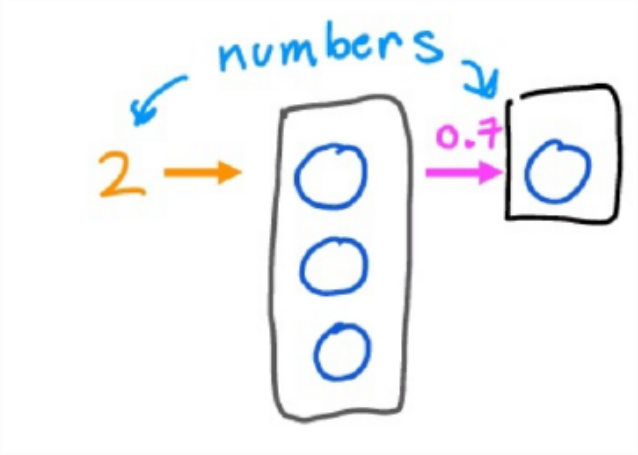
THÉO GACHET

Neurons and the Brain

Origins: Algorithms that try to mimic the brain

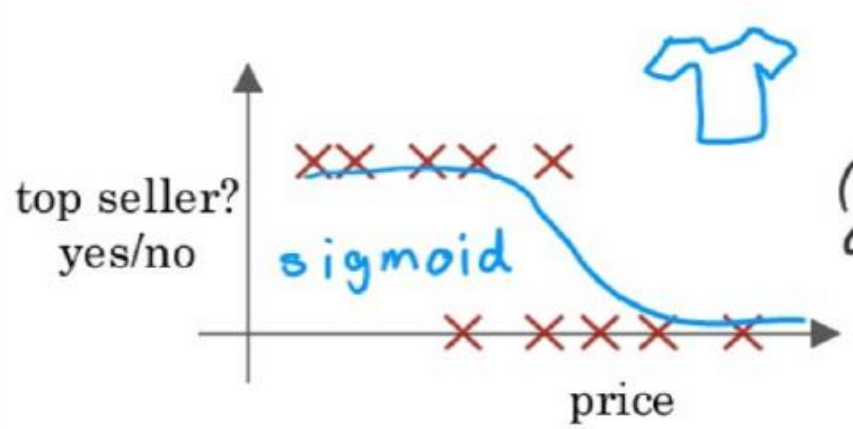


Biological Neuron



Simplified mathematical model of a neuron

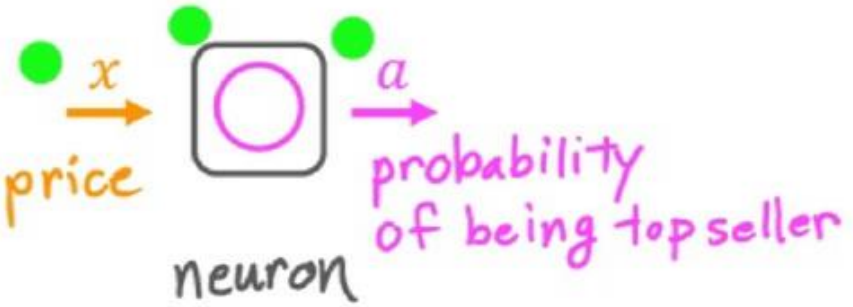
Demand Prediction

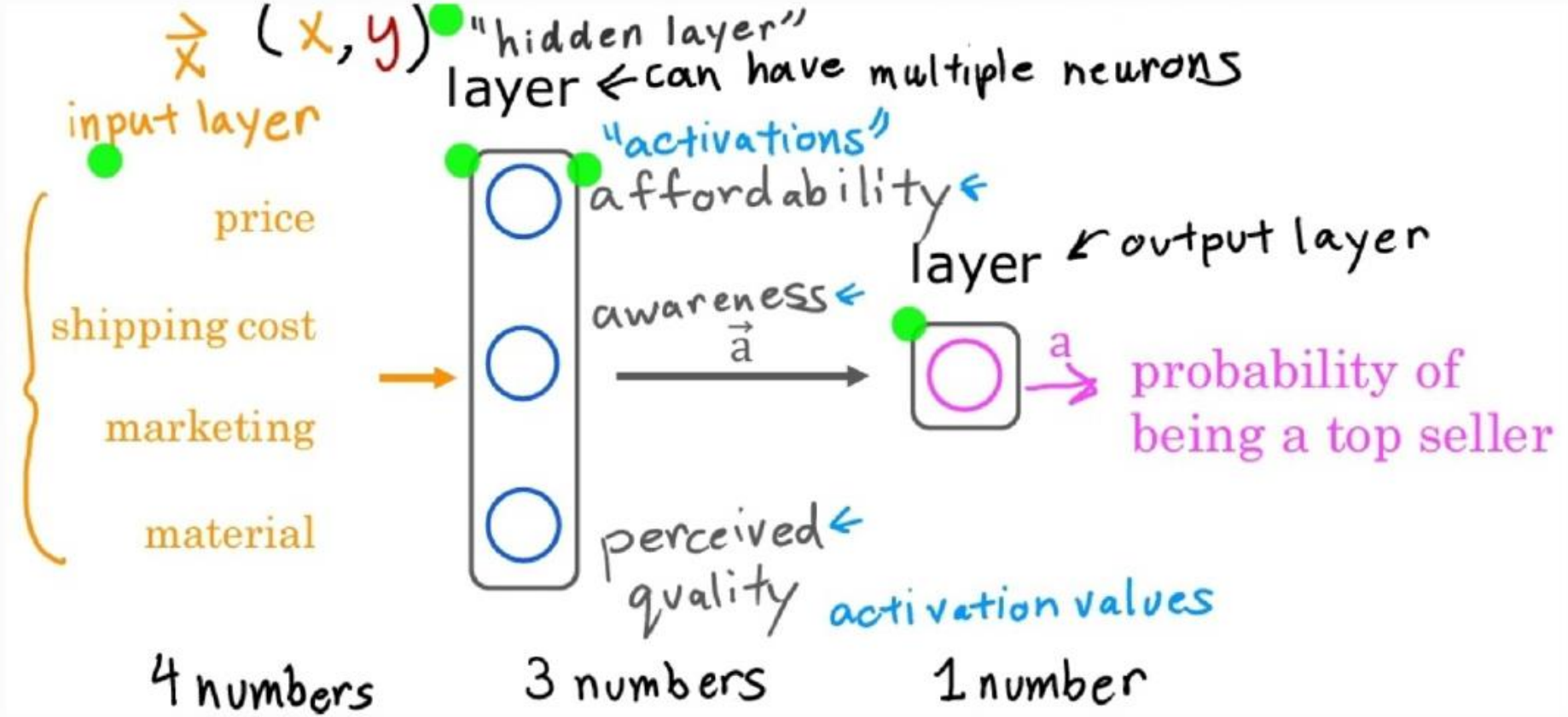


$x = price$ — input

$a = f(x) = \frac{1}{1 + e^{-(wx+b)}}$ — output

activation





By adding hidden layers to the NN architecture, you get a multilayer perceptron.

QUESTIONS ?

SUR UN CONCEPT ? UNE IDÉE ?
SUR UN DÉTAIL DU CODE ?
(ENVIE D'UNE PAUSE ?)

N'HÉSITEZ PAS !

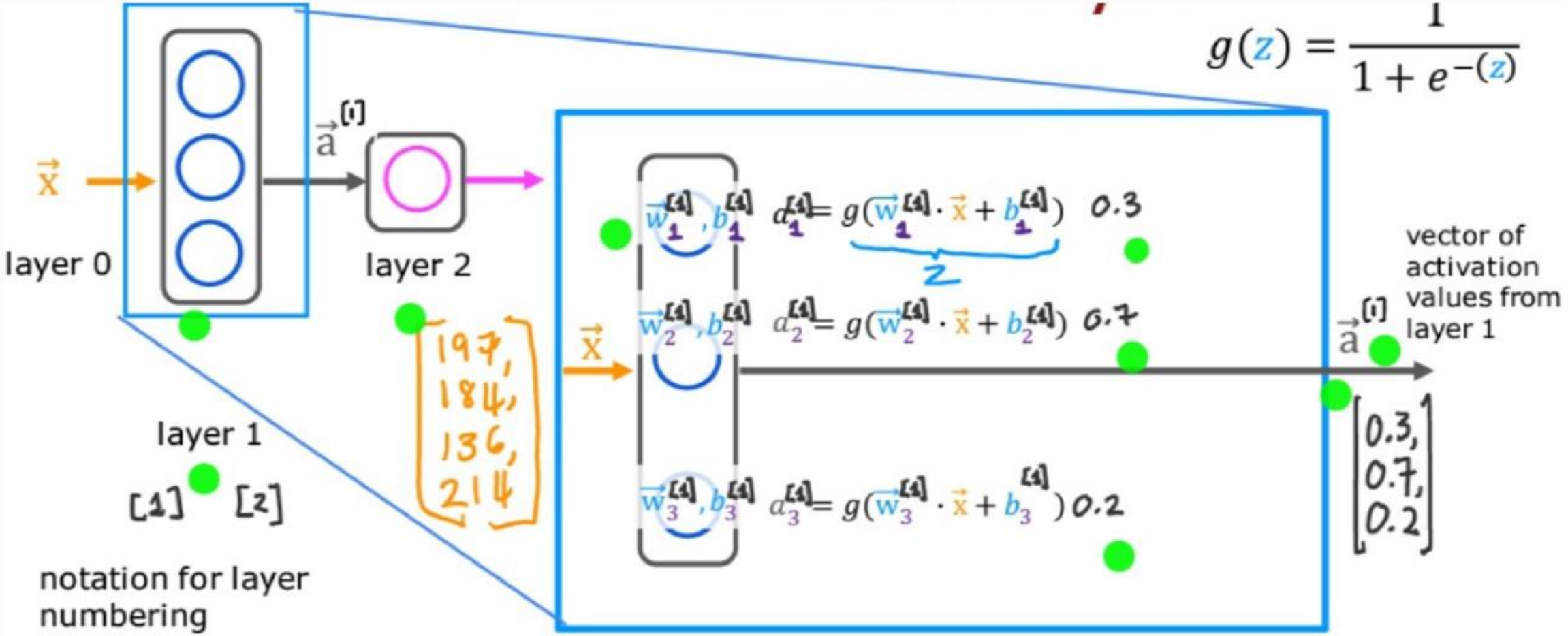
IL N'Y A PAS DE QUESTION BÊTE, SI VOUS AVEZ UN DOUTE, D'AUTRES ONT SÛREMENT LE MÊME

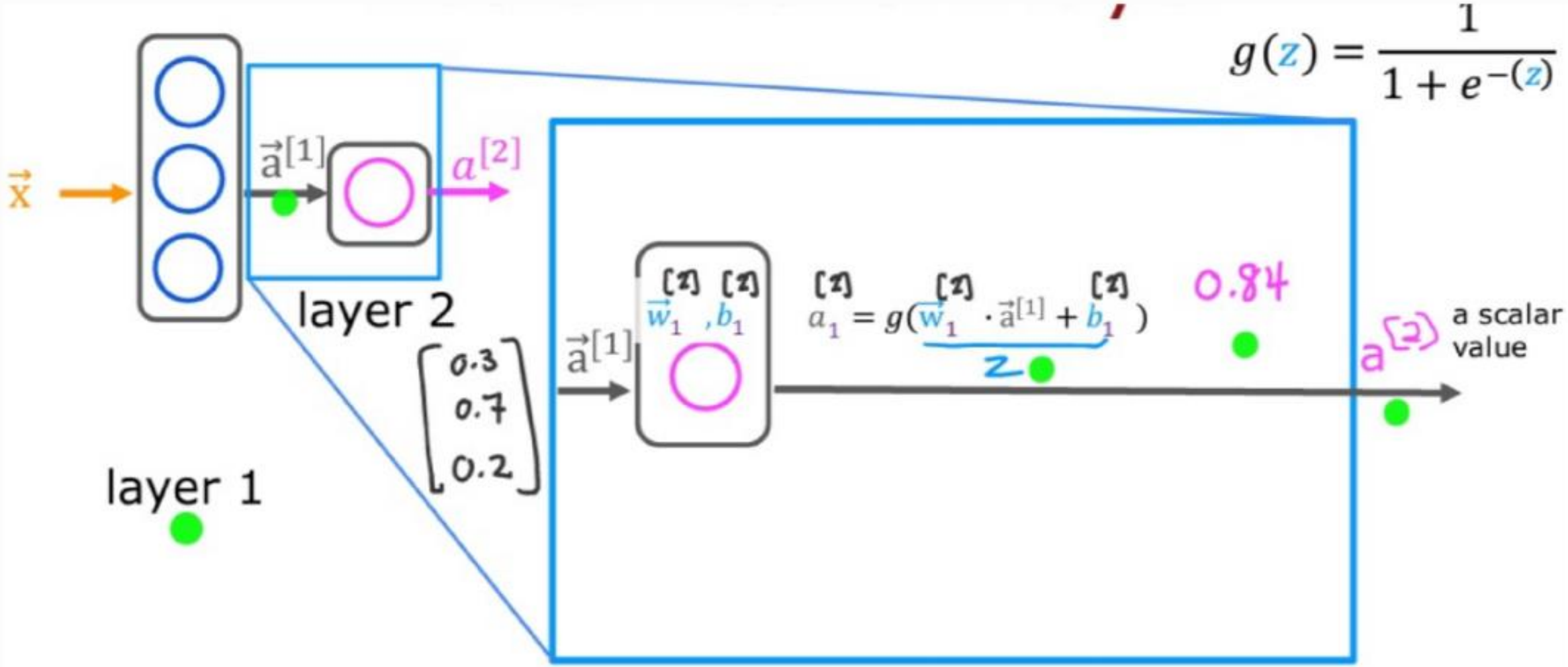
02

NEURAL NETWORK MODEL

THÉO GACHET

Neural Network layer




$g(z) = \frac{1}{1+e^{-(z)}}$
$\vec{x}$
$\vec{a}^{[1]}$
$a^{[2]}$
layer 2
layer 1
0.3
0.7
0.2
$\vec{w}_1, b_1$
$a_1 = g(\vec{w}_1 \cdot \vec{a}^{[1]} + b_1)$
z
0.84
a scalar value

output Predict category : is $a^{[2]} \geq 0.5$? $\rightarrow \hat{y} \in \{0,1\}$

activation function

activation value of layer l, unit j $\rightarrow$ $a_j^{[l]} = g(\vec{w}_j^{[l]} \cdot \vec{a}^{[l-1]} + b_j^{[l]})$

QUESTIONS ?

SUR UN CONCEPT ? UNE IDÉE ?
SUR UN DÉTAIL DU CODE ?
(ENVIE D'UNE PAUSE ?)

N'HÉSITEZ PAS !

IL N'Y A PAS DE QUESTION BÊTE, SI VOUS AVEZ UN DOUTE, D'AUTRES ONT SÛREMENT LE MÊME

LAB-01

NEURONS AND LAYERS

QUESTIONS ?

SUR UN CONCEPT ? UNE IDÉE ?
SUR UN DÉTAIL DU CODE ?
(ENVIE D'UNE PAUSE ?)

N'HÉSITEZ PAS !

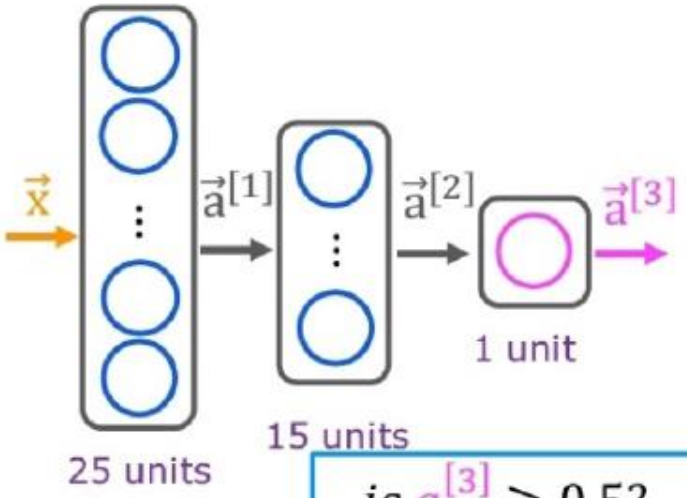
IL N'Y A PAS DE QUESTION BÊTE, SI VOUS AVEZ UN DOUTE, D'AUTRES ONT SÛREMENT LE MÊME

03

TENSORFLOW IMPLEMENTATION

THÉO GACHET

Inference in code



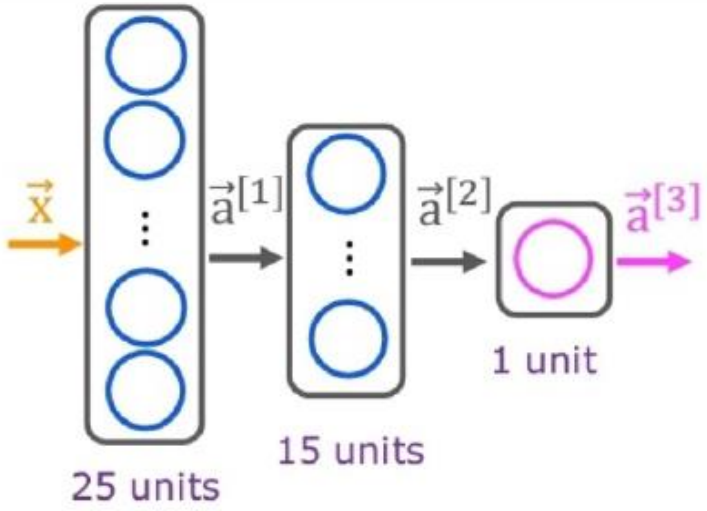

```
x = np.array([[0.0,...245,...240...0]])
layer_1 = Dense(units=25, activation='sigmoid')
a1 = layer_1(x)

layer_2 = Dense(units=15, activation='sigmoid')
a2 = layer_2(a1)

layer_3 = Dense(units=1, activation='sigmoid')
a3 = layer_3(a2)
```

$is\ a_1^{[3]} \geq 0.5?$

$\hat{y} = 1$ ✗ $\hat{y} = 0$ ○

```
if a3 >= 0.5:
  yhat = 1
else:
  yhat = 0
```

Building a Neural Network




```
model = Sequential([
    Dense(units=25, activation="sigmoid"),
    Dense(units=15, activation="sigmoid"),
    Dense(units=1, activation="sigmoid")])

model.compile(...)

x = np.array([[0..., 245, ..., 17],
              [0..., 200, ..., 184])
y = np.array([1,0])

model.fit(x,y)

model.predict(x_new)
```

model.compile : defines a loss function and specifies a compile optimization

model.fit : runs gradient descent and fits the weights to the data

```
model.compile(
    loss = tf.keras.losses.BinaryCrossentropy(),
    optimizer = tf.keras.optimizers.Adam(learning_rate=0.01),
)

model.fit(
    Xt,Yt,
    epochs=10,
)
```

Matrix Multiplication (NumPy)

Transpose A^T : `AT = A.T`

Multiply $A^T W$: `Z = np.matmul(AT, W) = AT @ W`

QUESTIONS ?

SUR UN CONCEPT ? UNE IDÉE ?
SUR UN DÉTAIL DU CODE ?
(ENVIE D'UNE PAUSE ?)

N'HÉSITEZ PAS !

IL N'Y A PAS DE QUESTION BÊTE, SI VOUS AVEZ UN DOUTE, D'AUTRES ONT SÛREMENT LE MÊME

LAB-02

COFFEE ROASTING IN TENSORFLOW

QUESTIONS ?

SUR UN CONCEPT ? UNE IDÉE ?
SUR UN DÉTAIL DU CODE ?
(ENVIE D'UNE PAUSE ?)

N'HÉSITEZ PAS !

IL N'Y A PAS DE QUESTION BÊTE, SI VOUS AVEZ UN DOUTE, D'AUTRES ONT SÛREMENT LE MÊME

04 NEURAL NETWORKS

NEURAL NETWORKS INTUITION

Neurons and the brain
Demand Prediction
Example : Recognizing Images

NEURAL NETWORK MODEL

Neural network layer
More complex neural networks
Inference : making predictions (forward propagation)
Lab : Neurons and Layers

TENSORFLOW IMPLEMENTATION

Inference in code
Data in TensorFlow
Building a neural network
Lab : Coffee Roasting in TensorFlow

QUESTIONS ?

SUR UN CONCEPT ? UNE IDÉE ?
SUR UN DÉTAIL DU CODE ?
(ENVIE D'UNE PAUSE ?)

N'HÉSITEZ PAS !

IL N'Y A PAS DE QUESTION BÊTE, SI VOUS AVEZ UN DOUTE, D'AUTRES ONT SÛREMENT LE MÊME

MINITEL

MACHINE LEARNING

04 NEURAL NETWORKS

THÉO GACHET