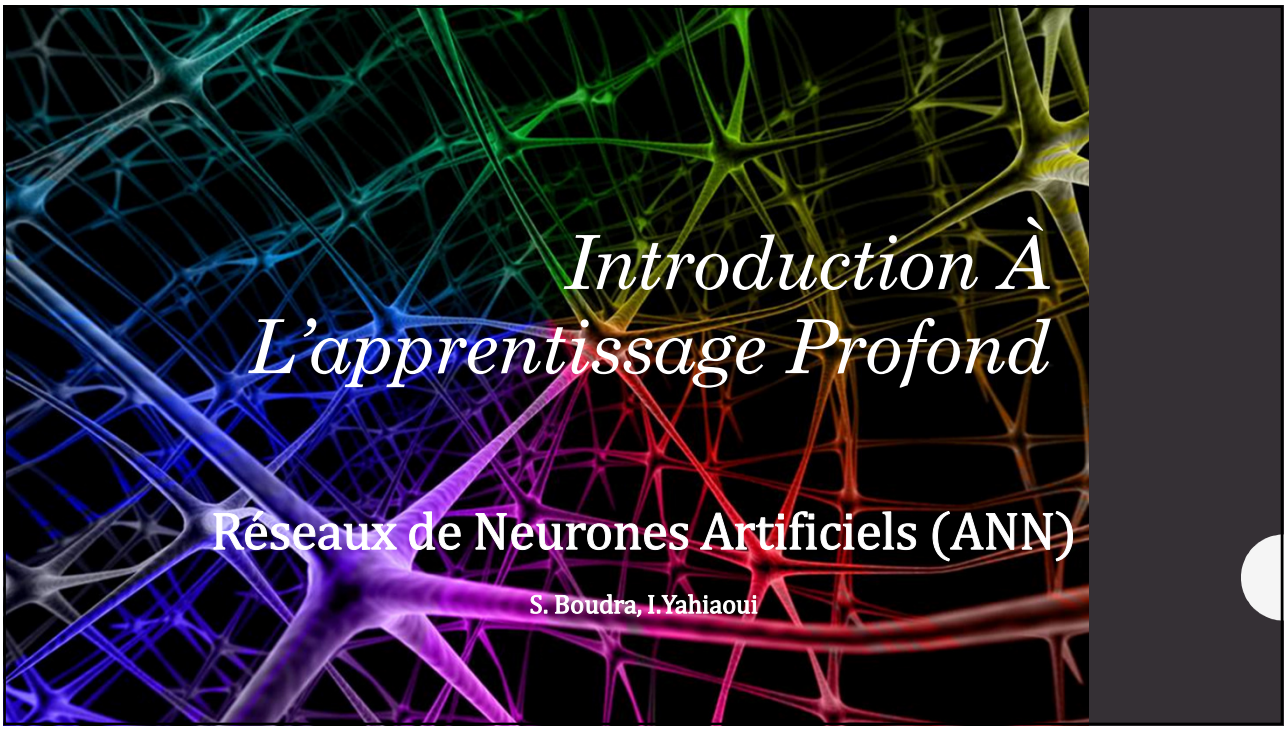


1

Ce qu'on va voir

1. Introduction aux réseaux de neurones et à l'apprentissage profond
2. Fonctionnement et protocole d'entraînement des ANN
3. Evaluation des réseaux de neurones
4. Réseaux de neurones à Convolution (CNN)
5. Réseaux antagonistes génératifs (GAN)

2

Contexte

Intelligence Artificielle — Imiter l'intelligence et le comportement humain

Apprentissage automatique *(Machine Learning)* — Techniques qui permettent à la machine d'apprendre à partir de données sans être explicitement programmée

Apprentissage Profond *(Deep Learning)* — Techniques d'apprentissage automatiques à base de « **réseaux de neurones** »

3

Types d'apprentissage

Supervisé

Donnée «étiquetées» «labelisées» «annotées»

$x_1 \rightarrow y_1$; $predire\ (\hat{y}_1)$
$x_2 \rightarrow y_2$; $predire\ (\hat{y}_2)$
....
$x_N \rightarrow y_N$; $predire\ (\hat{y}_N)$

Classification/Régression

Non-Supervisé

Donnée «non-étiquetées»

$x_1, x_2, \ldots, x_N$

Regrouper les données selon leurs similarité

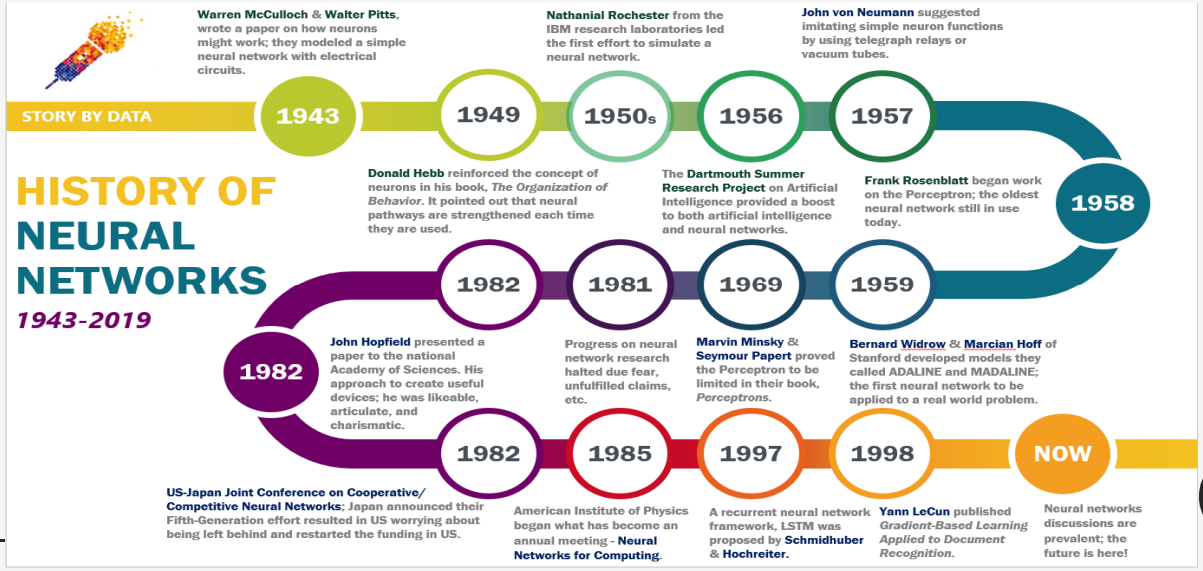
Regroupement/Clustering

Autres

Semi-supervisé,
Par renforcement,

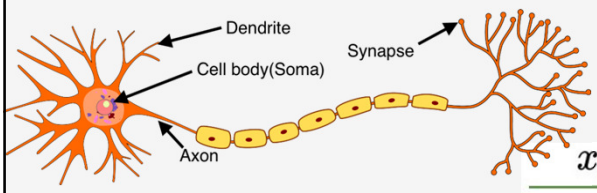
4

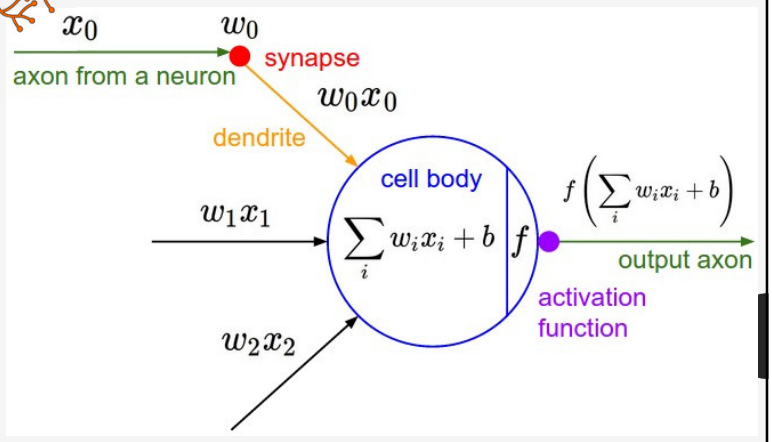
Un peu d'histoire



5

Neurone biologique vs. Neurone artificiel



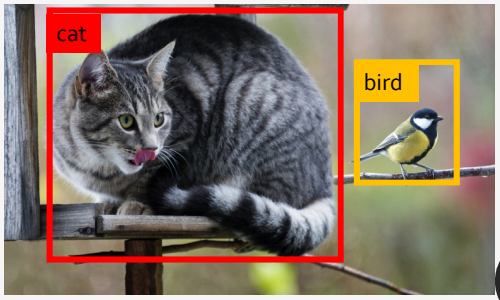


6

Applications

7

- Classification (1/0) Localisation des Objets Détection des Objets



8

Véhicules autonomes

https://bit.ly/34QcUb9

9

Eviter les obstacles

https://bit.ly/3iQ4BkW

10

Generative Adversial Network (GAN)

- Karras, Tero, et al. "Progressive growing of gans for improved quality, stability, and variation." *arXiv preprint arXiv:1710.10196* (2017).

11

FaceApp

https://www.faceapp.com/

12

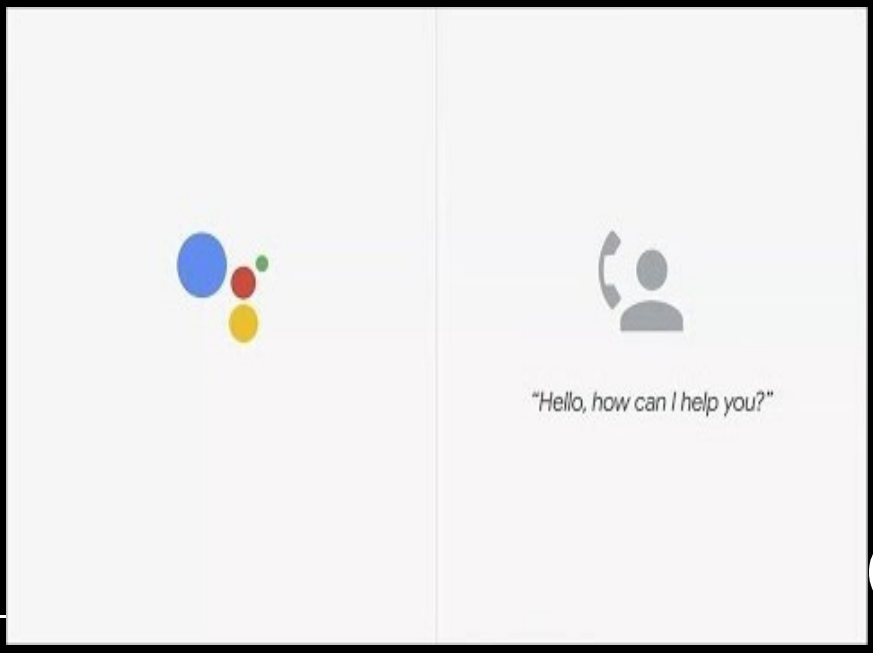
Générer de la music

ITERATION 2000

https://bit.ly/3jPmkdy

13

Google Duplex

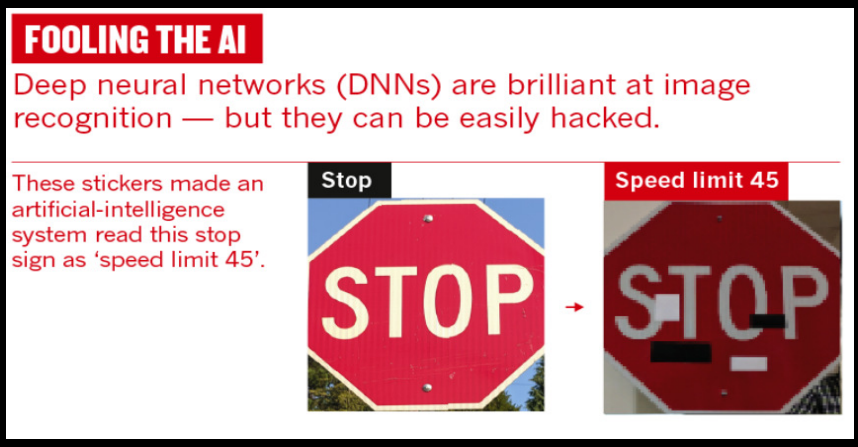


https://bit.ly/3iVb4L7

14

Faciles à hacker !

- Eykholt, Kevin, et al. "Robust physical-world attacks on deep learning visual classification." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.



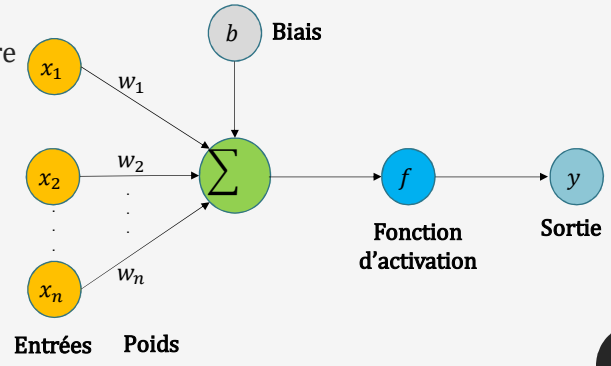
15

Neurone artificiel: Perceptron

- Le principe repose sur la sommation des entrées pondérées et le calcul du seuil d'activation.
- Un perceptron implémente la fonction linéaire
- suivante:

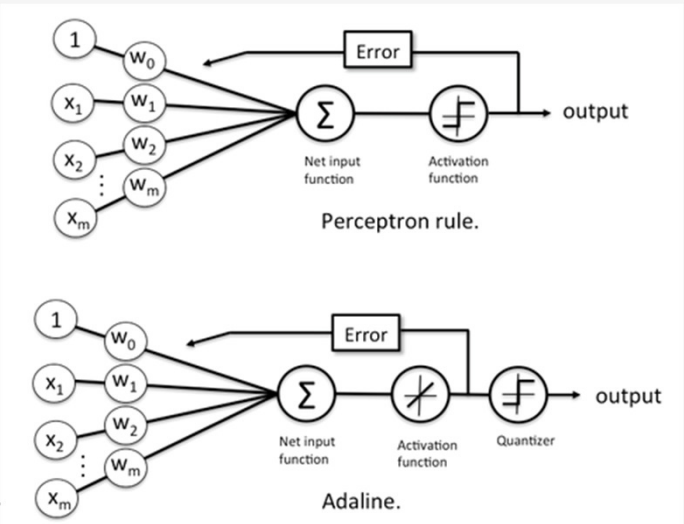
- $y = f(w_1 \times x_1 + w_2 \times x_2 + \cdots w_n \times x_n + b)$
- $\Rightarrow \mathcal{Y} = f(\mathcal{W}.\mathcal{X} + b)$

où f (v) $= \begin{cases} 1 \; si \; v \geq 0 \\ 0 \; si \; v < 0 \end{cases}$



16

Perceptron vs. Adaline



En commun:

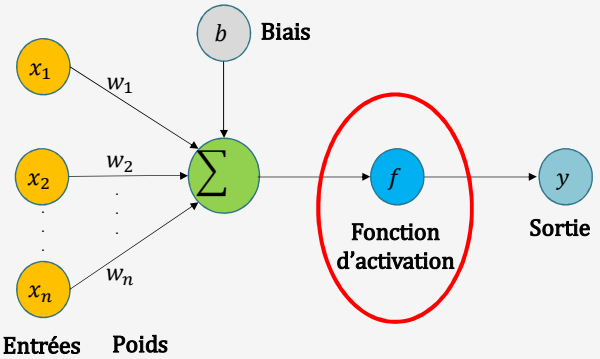
- Classifieur binaire (sortie 1/0)
- Séparation linaire entre les deux classes
- Apprentissage itératif.

Différence: processus d'apprentissage

- Perceptron utilise des valeurs de classe 1/0 **(valeurs discrètes)** pour calculer l'erreur.
- Adaline utilise des valeurs linéaires **(valeurs continues)** pour calculer l'erreur.

17

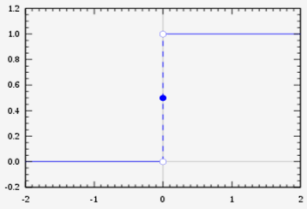
Régression vs. Classification



- **Classification:** Prédire y (valeur discrète
- Fonction d'activation:

fonction « seuil »

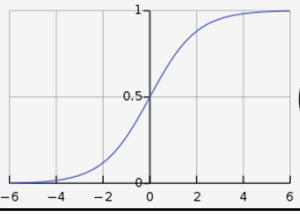
- $f(v) = \begin{cases} 1 \; si \; v \geq 0 \\ 0 \; si \; v < 0 \end{cases}$



- **Régression:** Approximer y (valeur continue, probabilité)
- Fonction d'activation:

fonction « sigmoid »

- $f(v) = \frac{1}{1+e^{-v}}$

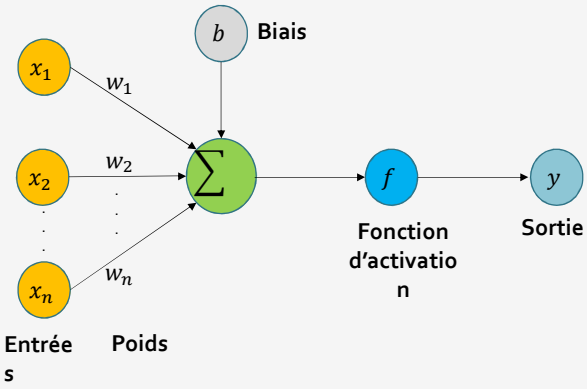


18

Que peut implémenter un perceptron?

- **Les fonctions logiques:**

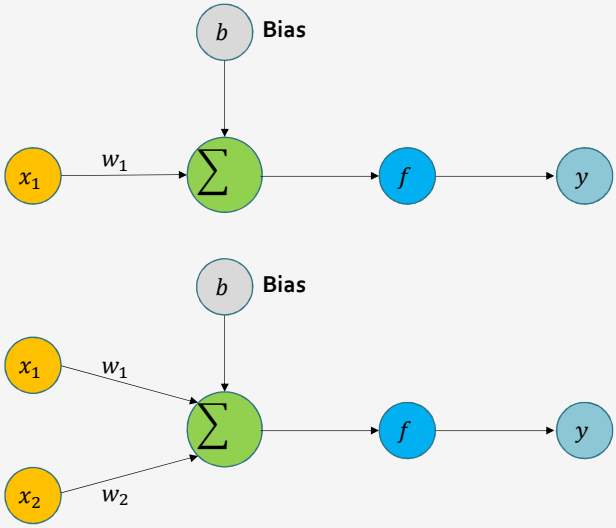
x_1	x_2	x_1 ET x_2	x_1 OU x_2	NON x_1
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0



19

Perceptron: le NON et le ET logiques

- Un perceptron modélise le NON logique par:
  - $y = f(w_1 x_1 + b)$
  - Une entrée x_1 et une sortie y
  - Pour $w_1 = -1$ *and* $b = 0.5$
  - $\Rightarrow y = f(-1 \times x_1 + 0.5)$
  - NON(1)=0; NON(0)=1
- Un perceptron le ET logique par:
  - $y = f(w_1 x_1 + w_2 x_2 + b)$
  - Deux entrées x_1, x_2, une sortie y
  - pour $w_1 = 1, w_2 = 1, b = -1.5$
  - $\Rightarrow y = f(x_1 + x_2 - 1.5)$
  - ET(1,1) =1, ET(0,1)=0



20

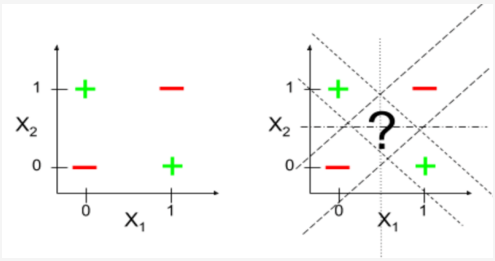
Problème du XOR logique

- Problème:
- ***Non separable linéairement***

- Solution:

➔*Besoin d'une couche intermédiaire*

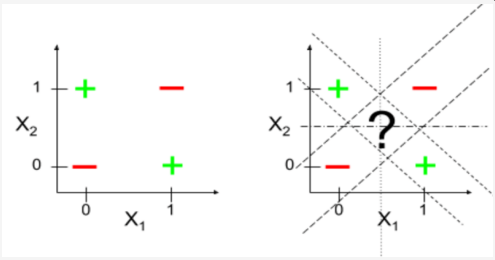
x_1	x_2	$x_1 \text{XOR} x_2$
0	0	0
0	1	1
1	0	1
1	1	0



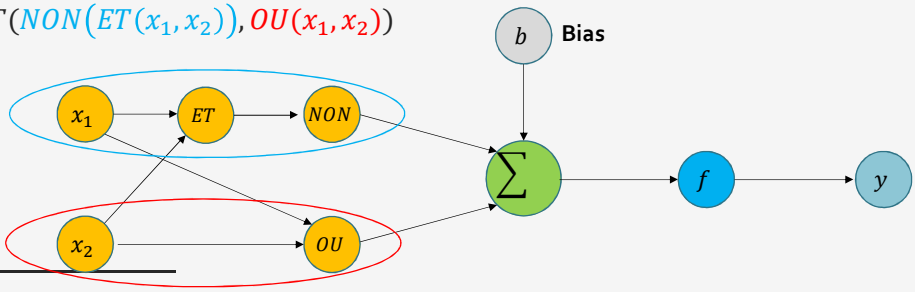
21

Problème du XOR logique

x_1	x_2	$x_1 \text{XOR} x_2$
0	0	0
0	1	1
1	0	1
1	1	0

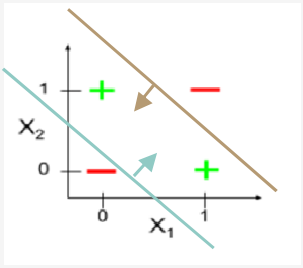


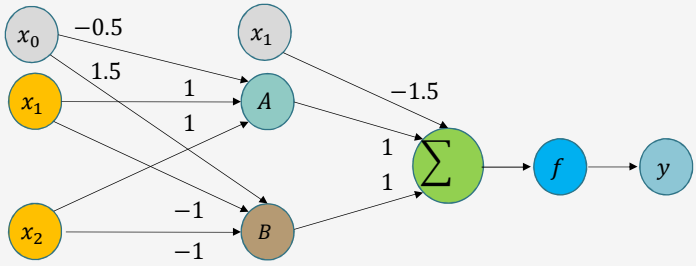
- $XOR(x_1, x_2) = ET(NON(ET(x_1, x_2)), OU(x_1, x_2))$



22

Problème du XOR logique

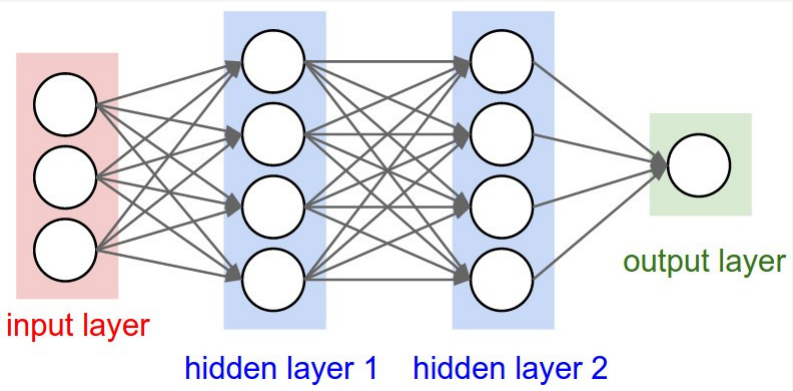




- $f(1 \times 0 + 1 \times 0 - 0.5) = 0$
- $f(1 \times 0 + 1 \times 1 - 0.5) = 1$
- $f(1 \times 1 + 1 \times 0 - 0.5) = 1$
- $f(1 \times 1 + 1 \times 1 - 0.5) = 1$
- $f(-1 \times 0 - 1 \times 0 + 1.5) = 1$
- $f(-1 \times 0 - 1 \times 1 + 1.5) = 1$
- $f(-1 \times 1 - 1 \times 0 + 1.5) = 1$
- $f(-1 \times 1 - 1 \times 1 + 1.5) = 0$
- $f(1 \times 0 + 1 \times 1 - 1.5) = 0$
- $f(1 \times 1 + 1 \times 1 - 1.5) = 1$
- $f(1 \times 1 + 1 \times 0 - 1.5) = 0$

23

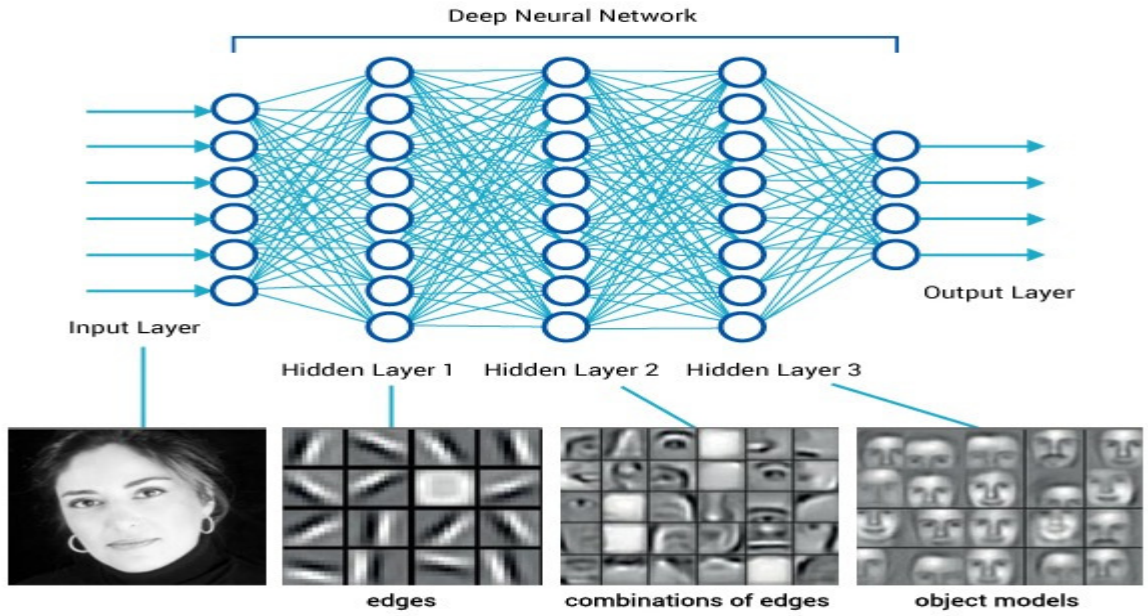
Réseaux de neurones



- **Type de couche:**
- Couche d'entrée: les données
- Couches cachées: couches intermédiaires, nécessaires pour des calculs complexes non séparables linéairement.
- Couche de sortie: étiquettes des classes.

24

Réseaux Profonds



25

Références

- https://medium.com/analytics-vidhya/brief-history-of-neural-networks-44c2bf72eec
- https://www.youtube.com/watch?v=kNPGXgzxoHw
- https://towardsdatascience.com/perceptrons-logical-functions-and-the-xor-problem-37ca5025790a
- https://medium.com/@lucaspereira0612/solving-xor-with-a-single-perceptron-34539f395182
- https://www.udemy.com/course/deeplearning/

26