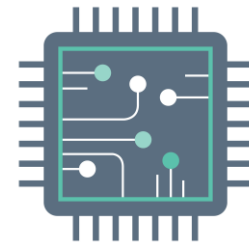
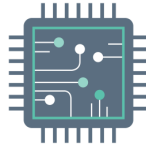


De l'informatique déterministe à l'intelligence artificielle

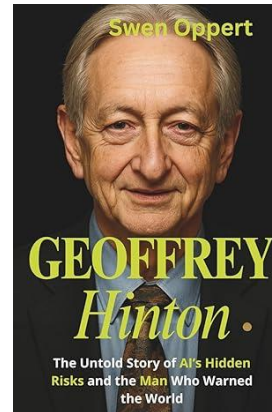
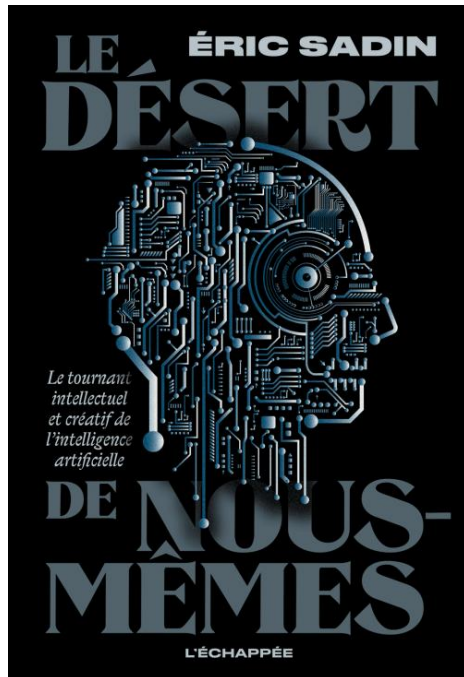
Julien VILLEMEJANE



« Ceci est un chat »



Motivations



Vocabulaire / Définition

Intelligence Artificielle

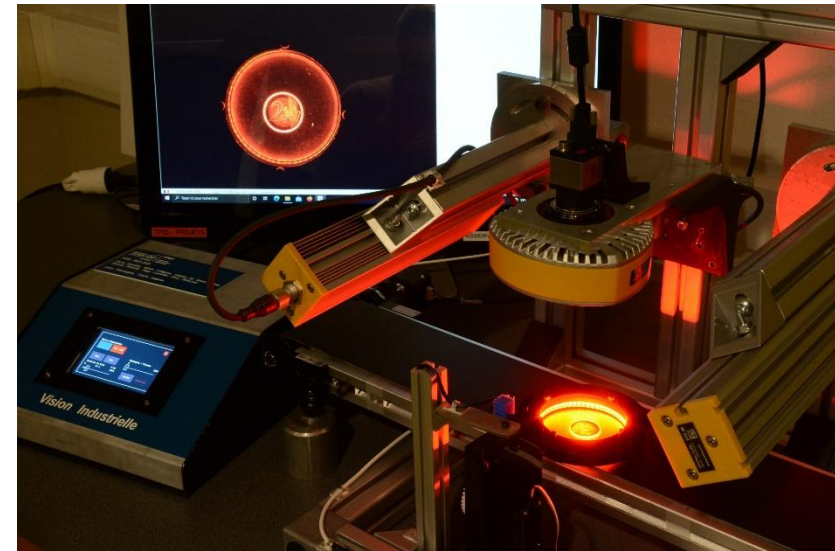
- Diagnostic médical
- Cybersécurité / Détection de fraudes
- Robotique autonome
- Analyse de risque
- Systèmes de recommandation Netflix/YouTube
- Reconnaissance vocale (Siri, Alexa)

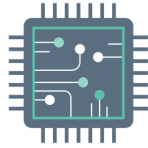
IA Génératives

- Génération de textes, d'images, de codes...

Informatique déterministe

Programmation séquentielle





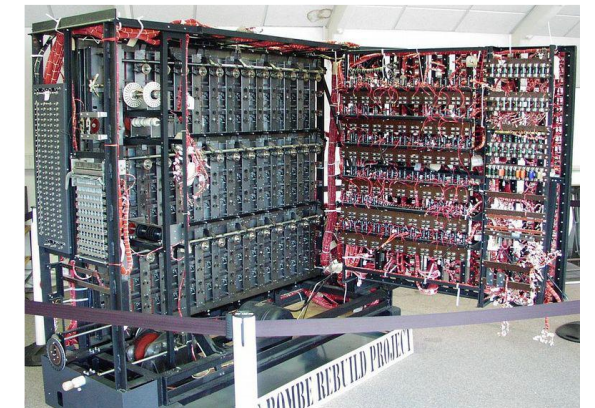
Informatique, pourquoi ?

traitement automatique de l'information par l'exécution de programmes informatiques par des machines

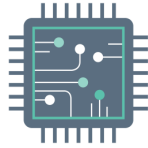
- Aide aux calculs
- Traitement de l'information
- Automatisation de calculs / de tâches
- Télécommunications
- Stockage de l'information



1652 / La PASCALINE

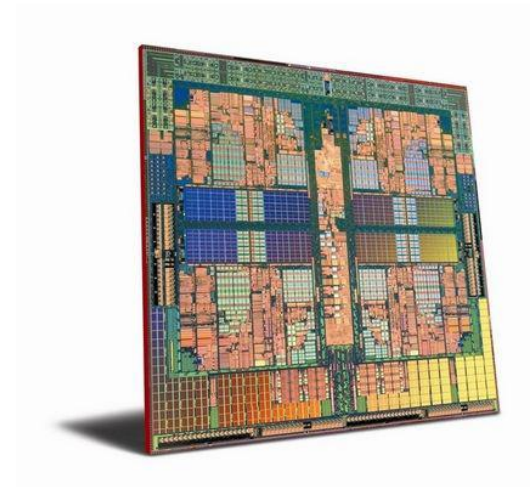
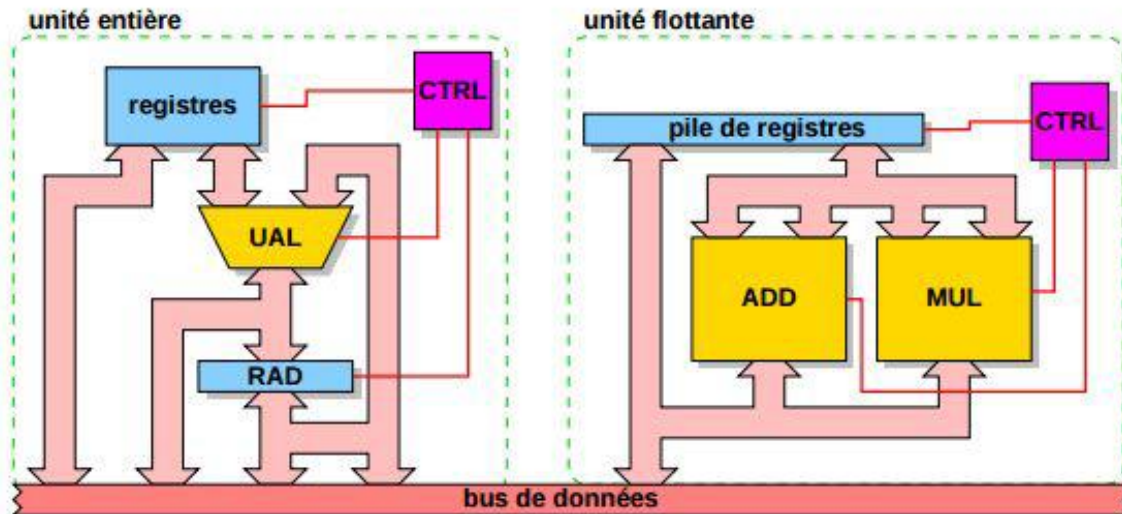


1940 / La BOMBE / Turing



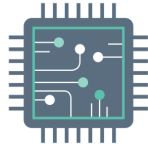
Informatique déterministe

Calculateur / Structure



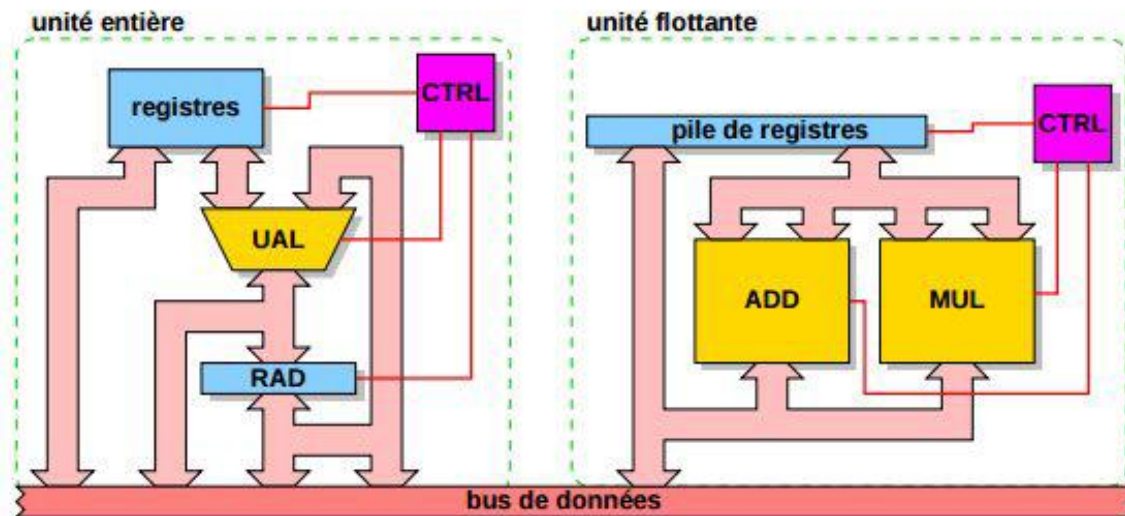
Processeur AMD Phenom – 4 coeurs

- **Exécution séquentielle** d'instructions
- Unité(s) de calculs **précablée(s)**
- Environ 2000 instructions

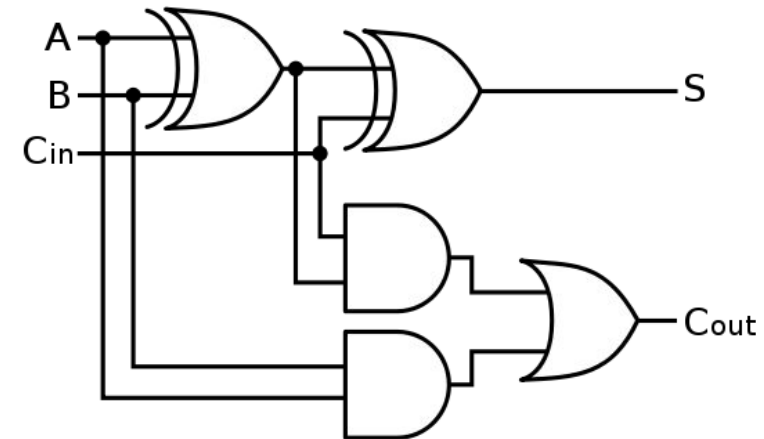
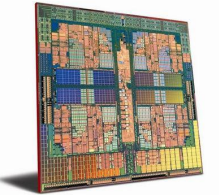


Informatique déterministe

Calculateur / Structure

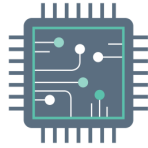


Processeur AMD Phenom – 4 coeurs



Additionneur 1 bit / ~30 transistors

- **Exécution séquentielle** d'instructions
- Unité(s) de calculs **précablée(s)**
- Environ 2000 instructions



Informatique déterministe

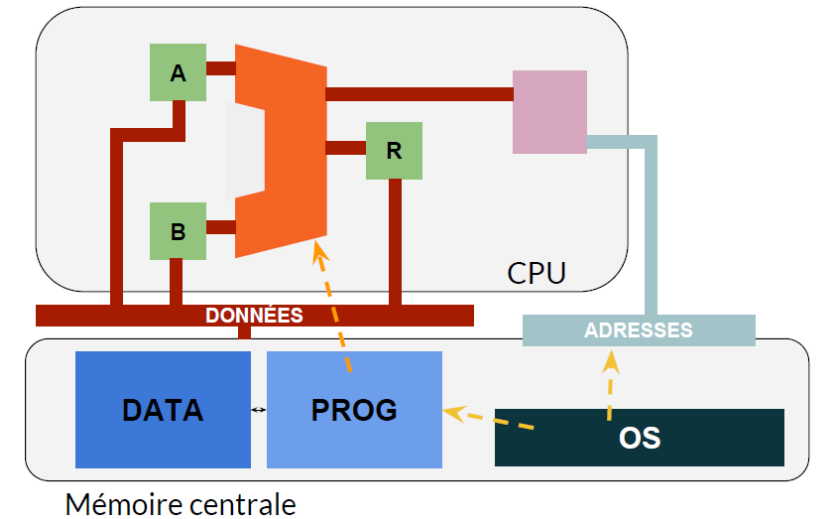
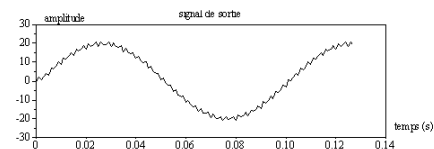
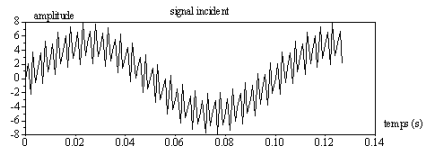
Programmes séquentiels

Suite d'instructions

Données

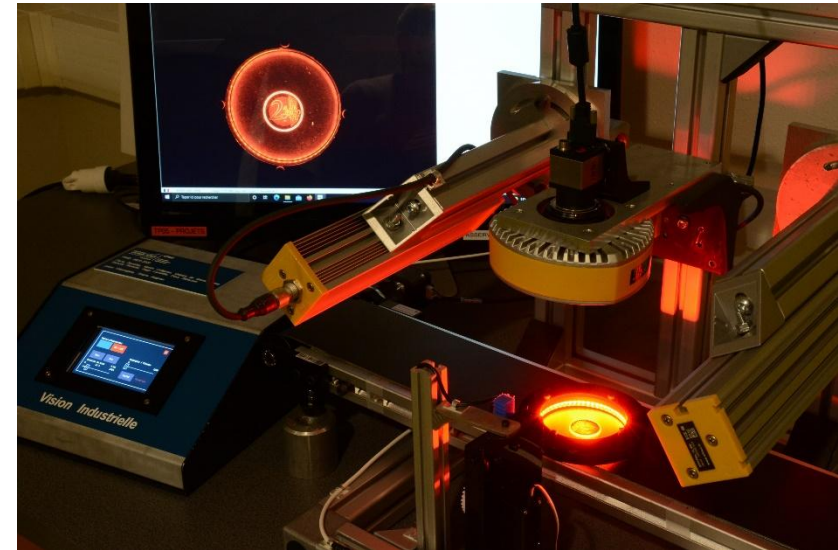
Calculateur

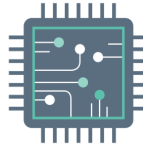
Résultats



Parallélisation

Structures parallèles



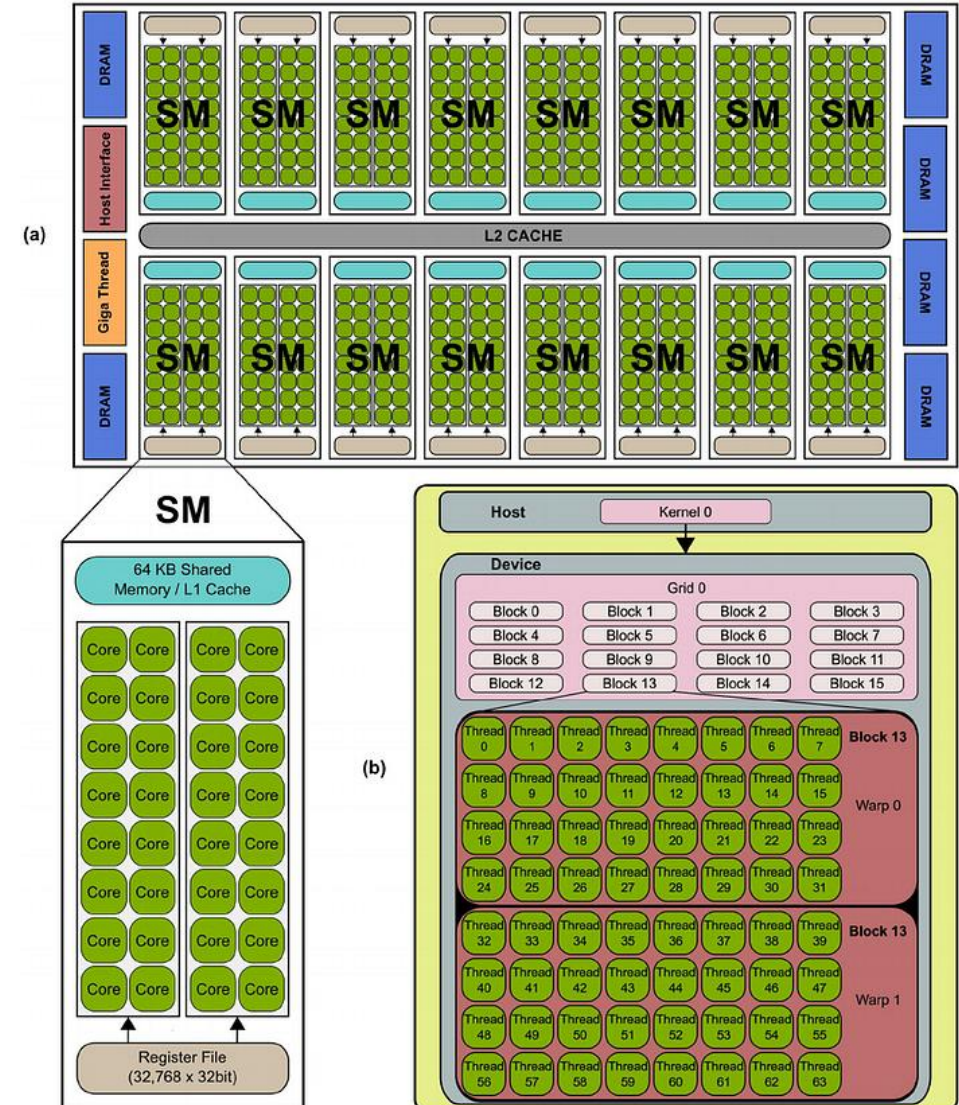
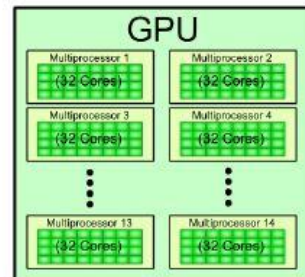
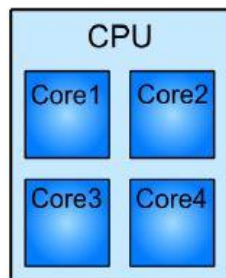


Parallélisation des tâches

Calculs vectoriels / matriciels

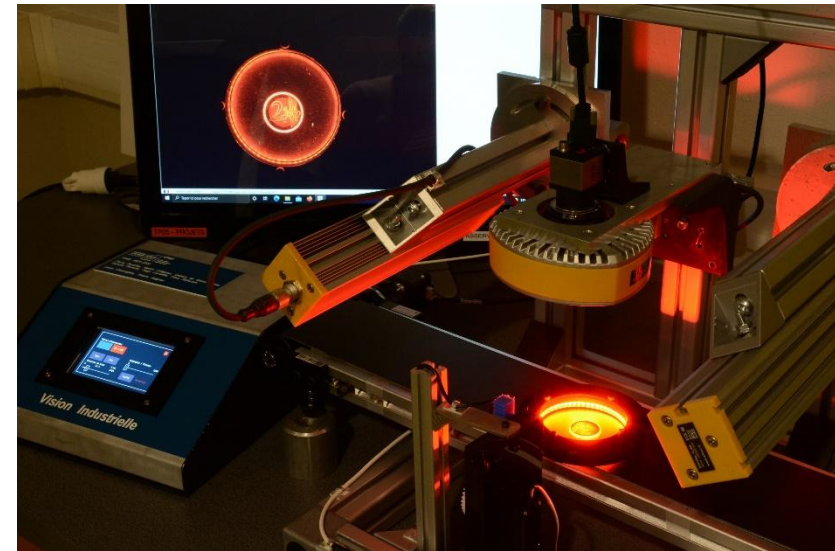
- **Exécution séquentielle** d'instructions sur chaque cœur
- Plus de **10000 cœurs** en parallèle (pour les plus récents)
- Environ 100 instructions / cœur

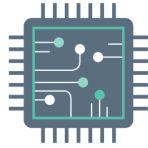
CPU/GPU Architecture Comparison



Intelligence artificielle

Définition

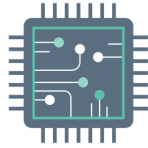




Définition

Sous-domaine de l'informatique dont l'objectif est de **faire accomplir à des machines des tâches nécessitant des capacités cognitives humaines**

- systèmes à règles
- recherche et planification
- logique
- systèmes experts
- *machine learning*



Définition

Sous-domaine de l'informatique dont l'objectif est de **faire accomplir à des machines des tâches nécessitant des capacités cognitives humaines**

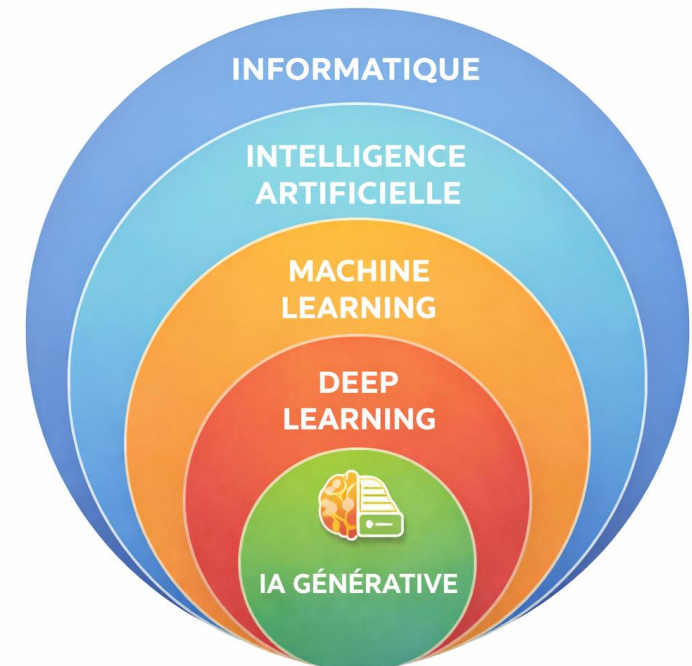
- systèmes à règles
- recherche et planification
- logique
- systèmes experts
- *machine learning*



L'intelligence artificielle ne fait pas nécessairement appel à des neurones ni à un système calqué sur le cerveau humain.

IA symbolique (non neuronale) dans les années 1950–1990
règles explicites (if / then), logique formelle, graphes, arbres de recherche

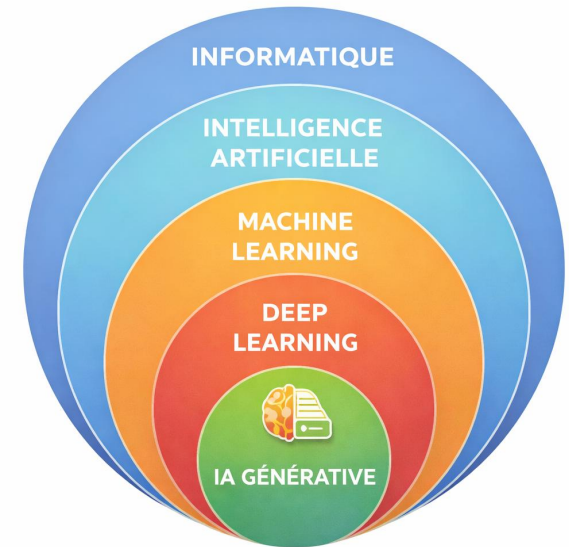
=/= IA connexionniste

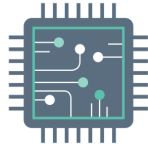




Machine Learning

Définition
Principe

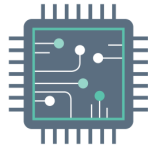




Définition

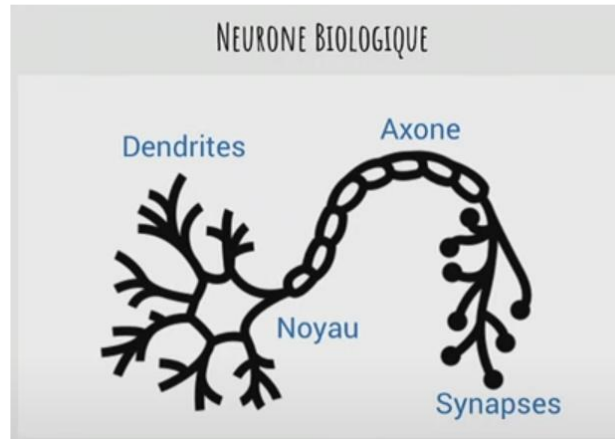
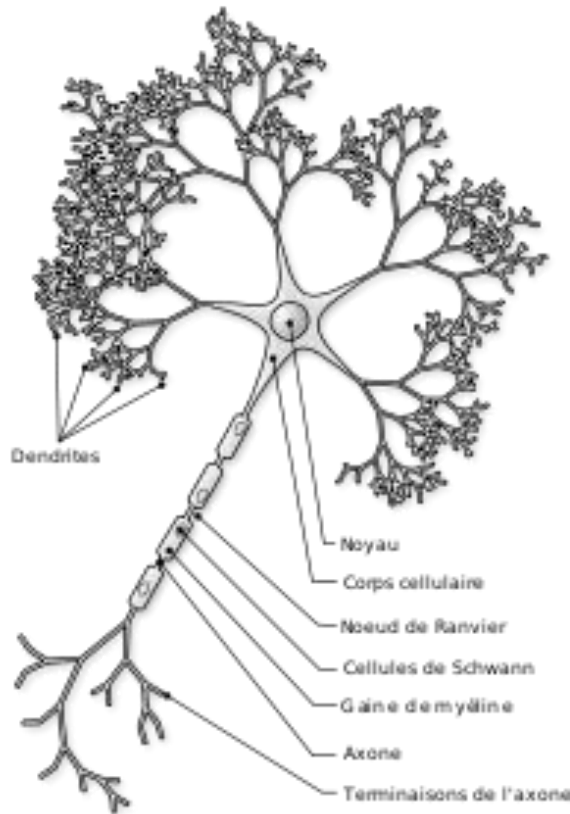
Sous-domaine de l'intelligence artificielle, dont l'objectif est de **donner aux ordinateurs la capacité d'« apprendre » à partir de données**, par le biais d'approches mathématiques et statistiques

- apprentissage automatique
- apprentissage machine
- apprentissage artificiel
- apprentissage statistique



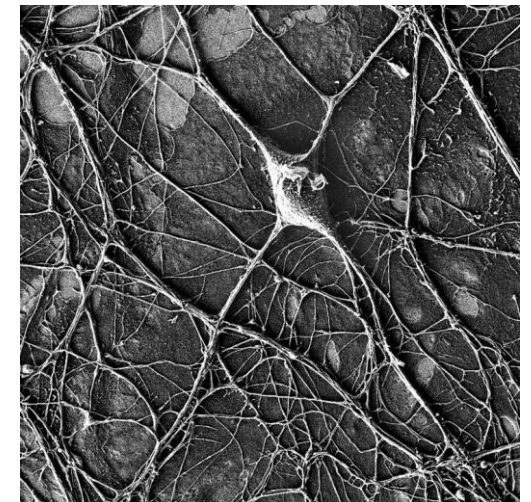
Intelligence humaine

Fonctionnement des neurones



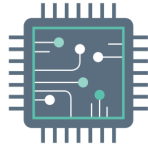
Hello TensorFlow : 3 ateliers
pour débuter avec TensorFlow 2.0
(Alexia Audevart et Philippe Antoine)

<https://youtu.be/hQ6pmoNZzU8>



*Issu d'une région du cerveau appelée
«substance noire compacte», ce
neurone dopaminergique intrigue
et fascine.*

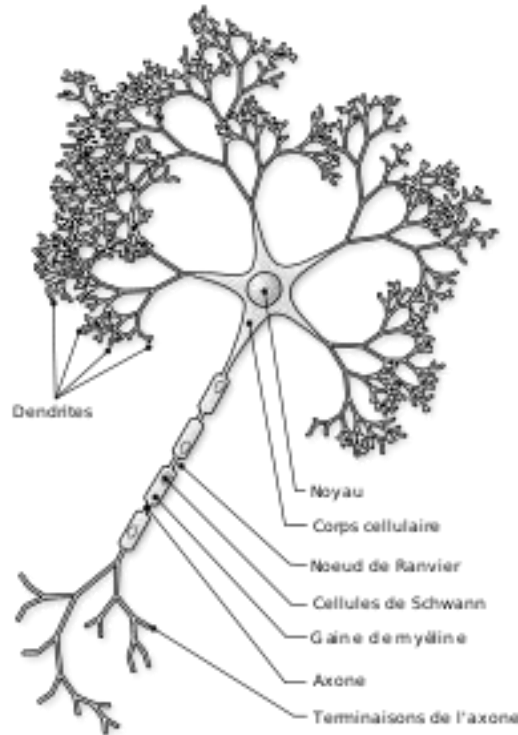
CHARLES DUCROT,
UNIVERSITÉ DE MONTRÉAL



Apprentissage et inférence

Phase d'apprentissage

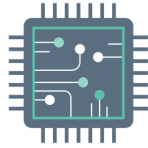
Phase d'inférence



- **Récolte de données sensorielles** : sons, images, mouvements, interactions sociales.
- **Exploration de motifs** : quelles actions produisent des résultats positifs (ex. sourire → attention des parents).
- **Renforcement (ou affaiblissement) des synapses** selon les expériences (plasticité neuronale)

Procédé **essentiellement expérimental et probabiliste**, avec beaucoup de “tentatives-erreurs”.

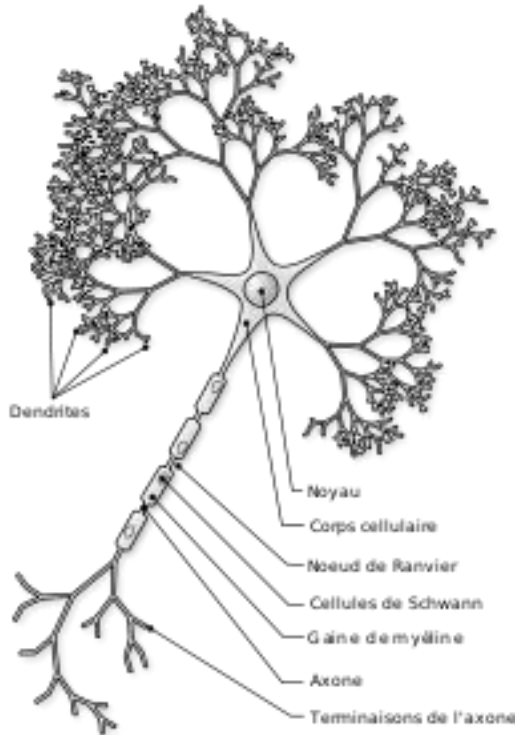
* Inférence = fait de **tirer une conclusion à partir d'informations disponibles**, selon des règles explicites ou implicites.



Apprentissage et inférence

Phase d'apprentissage

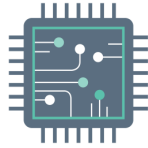
Phase d'inférence



- En fonction des **motifs prédéfinis**, **prédictions** et **prise de décision** rapidement
 - Reconnaître un visage familier dans la foule
 - Comprendre le sens d'une phrase incomplète
 - Savoir qu'un objet chaud peut brûler

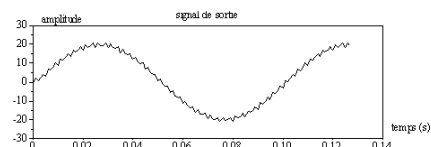
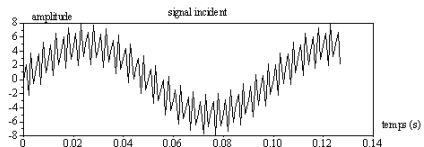
Procédé ne nécessitant **pas de modification des connexions synaptiques** (*en tout cas immédiate*)

* *Inférence* = fait de **tirer une conclusion** à partir d'**informations disponibles**, selon des règles explicites ou implicites.



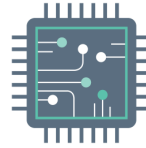
Machine Learning

Informatique Déterministe



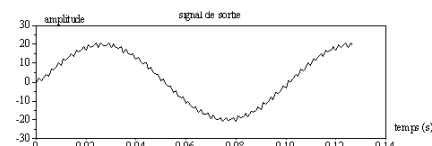
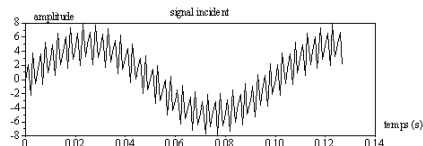
Machine Learning





Machine Learning

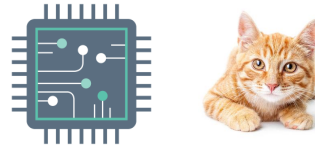
Informatique Déterministe



Machine Learning

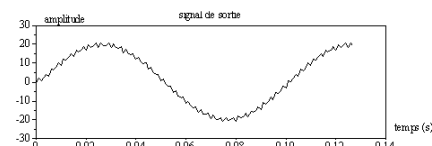
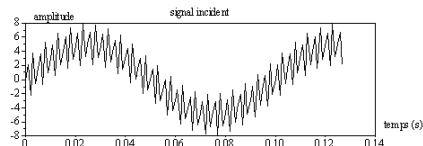
Phase d'apprentissage





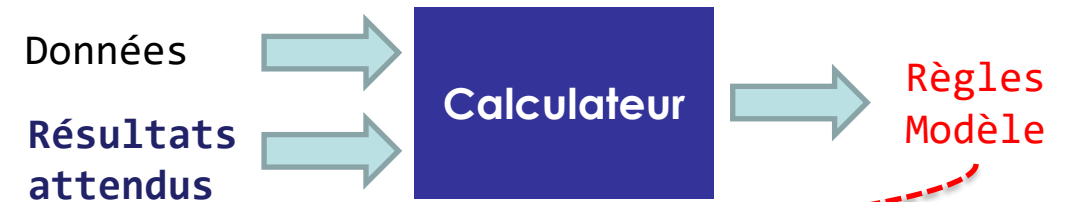
Machine Learning

Informatique Déterministe



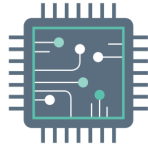
Machine Learning

Phase d'apprentissage



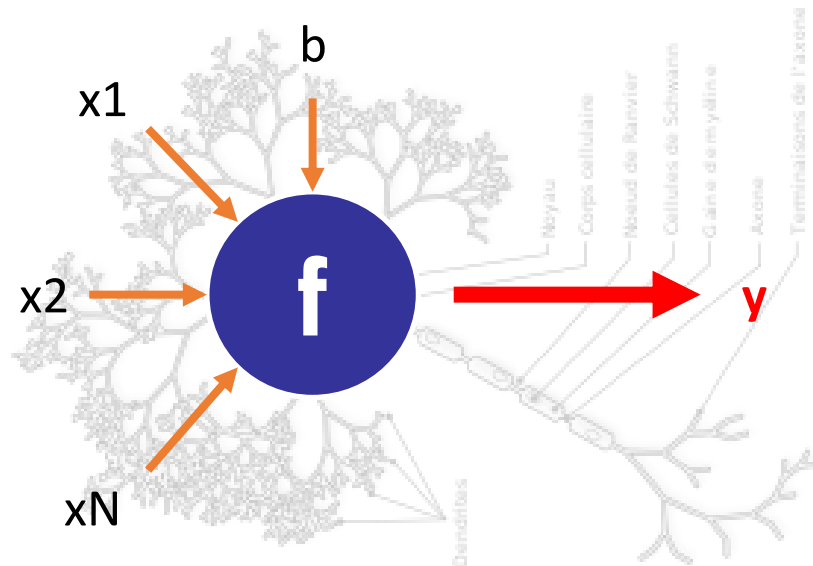
Phase d'inférence





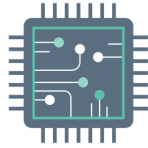
Neurone artificiel

Système à neurones



$$f = \sum_{k=1}^n w_k \cdot x_k + b$$

$$y = \begin{cases} 1 & \text{si } f \geq 0 \\ 0 & \text{sinon} \end{cases}$$

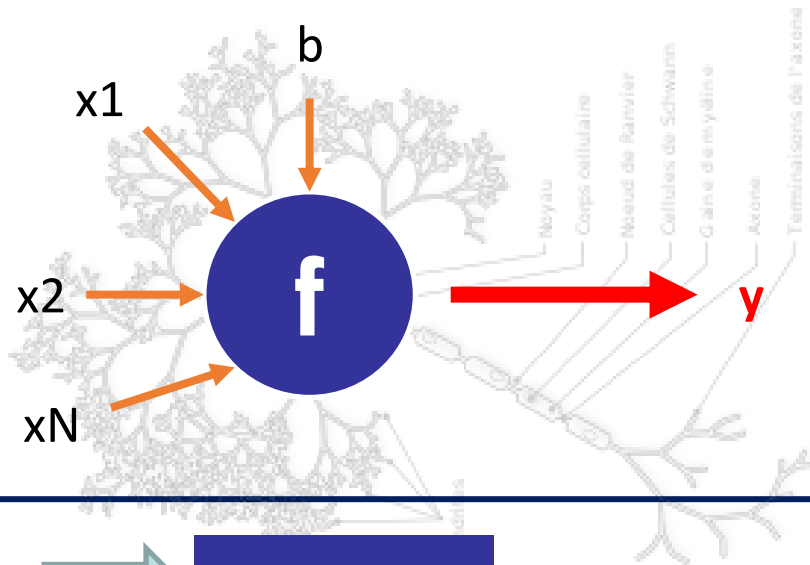


Neurone artificiel

Système à neurones

Phase d'apprentissage

Phase d'inférence



Données

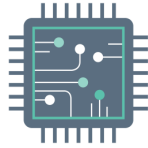
Résultats
attendus

Calculateur

Règles
Modèle

$$f = \sum_{k=1}^n w_k \cdot x_k + b$$

$$y = \begin{cases} 1 & \text{si } f \geq 0 \\ 0 & \text{sinon} \end{cases}$$

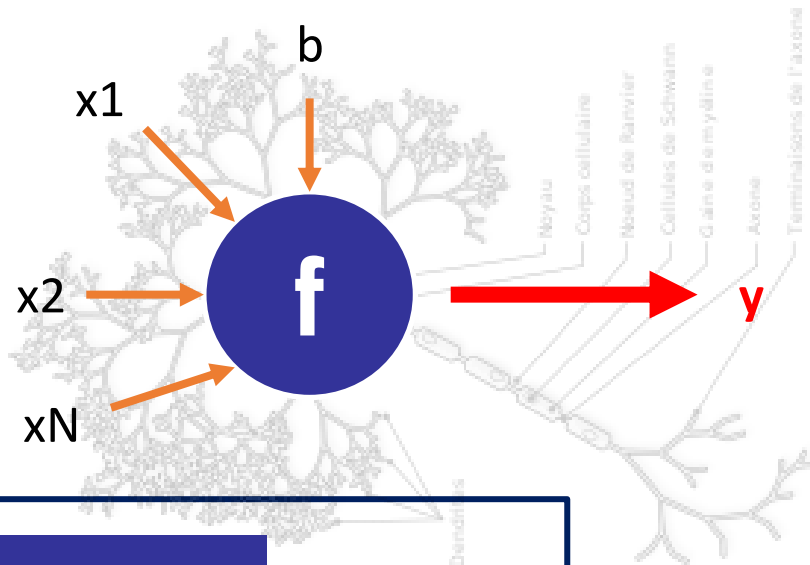


Neurone artificiel

Système à neurones

Phase d'apprentissage

plusieurs « epochs »



Phase d'inférence

$$f = \sum_{k=1}^n w_k \cdot x_k + b$$

$$y = \begin{cases} 1 & \text{si } f \geq 0 \\ 0 & \text{sinon} \end{cases}$$

1 epoch = 1 cycle d'apprentissage (c)

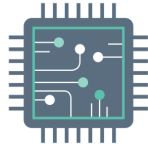
$$w_k(c + 1) = w_k(c) + \alpha \cdot (y_v - y) \cdot x_k$$

α : vitesse d'apprentissage

y_v : valeur de sortie attendue

y : valeur fournie par le neurone





Neurone artificiel

Système à neurones

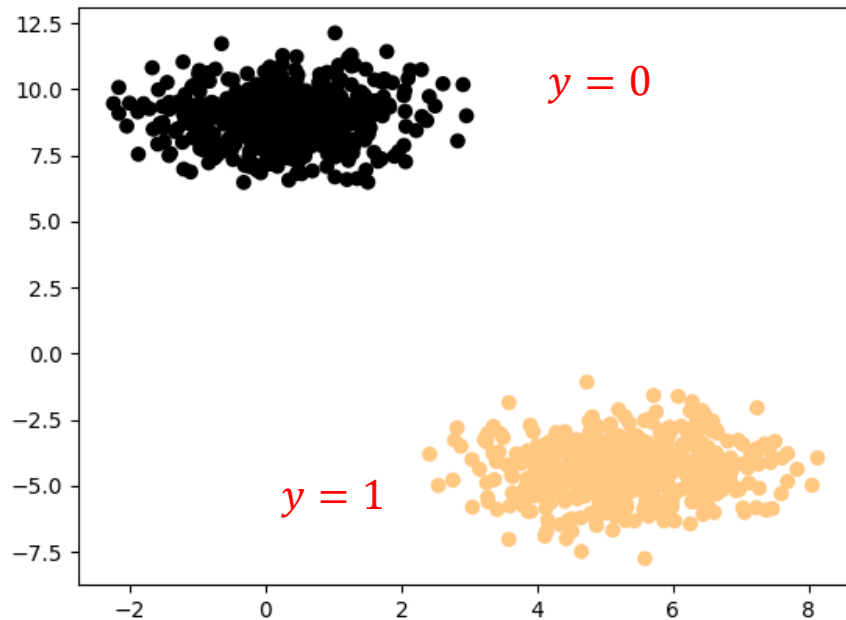
Phase d'apprentissage

Phase d'inférence

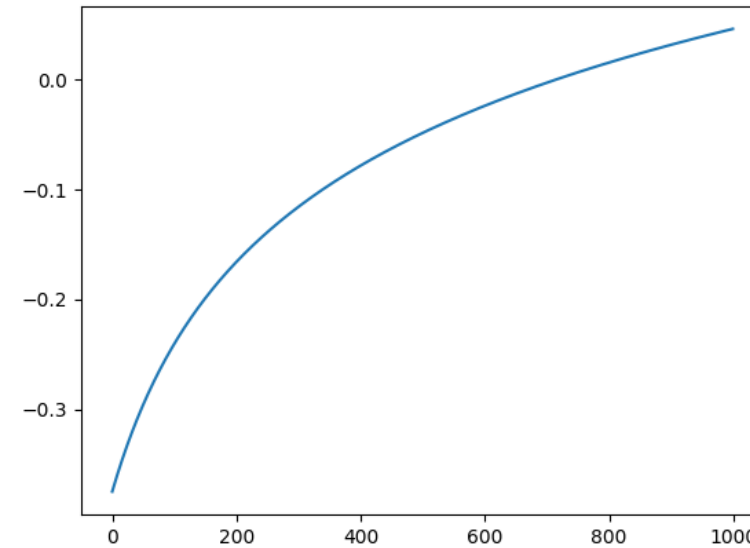
Epochs= 1000

$\alpha = 0,002$

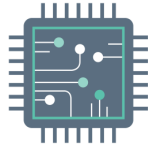
Données d'entrée



Evolution du premier parametre W



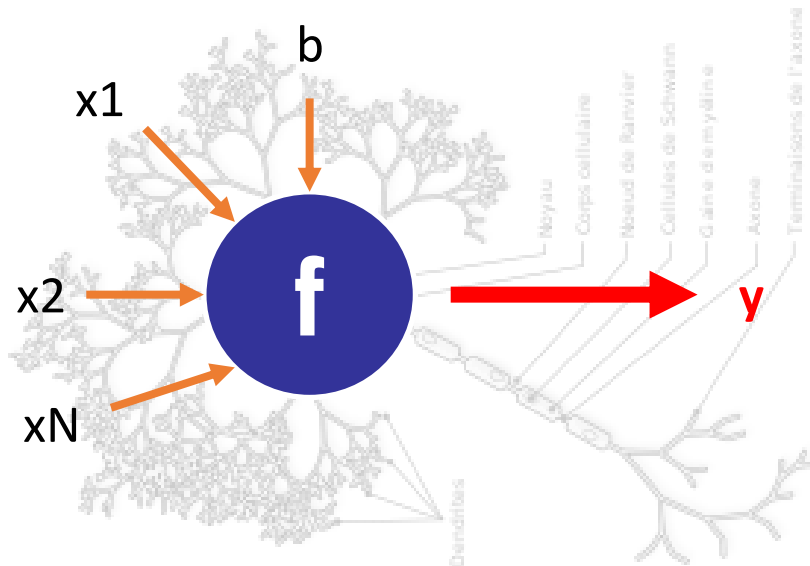
$$w_k(c + 1) = w_k(c) + \alpha \cdot (y_v - y) \cdot x_k$$



Neurone artificiel

Système à neurones

Phase d'apprentissage



Phase d'inférence

Etape 1 : *Forward Propagation*

$$f = \sum_{k=1}^n \boxed{w_k} \cdot x_k + \boxed{b}$$

$$y = \begin{cases} 1 & \text{si } f \geq 0 \\ 0 & \text{sinon} \end{cases}$$

Etape 2 : *Cost Function / Loss*

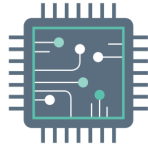
$$w_k(c + 1) = w_k(c) + \alpha \cdot (\boxed{y_v} - \boxed{y}) \cdot x_k$$

α : vitesse d'apprentissage

y_v : valeur de sortie attendue

y : valeur fournie par le neurone

Etape 3 :
Mise à jour



Neurone artificiel

Système à neurones

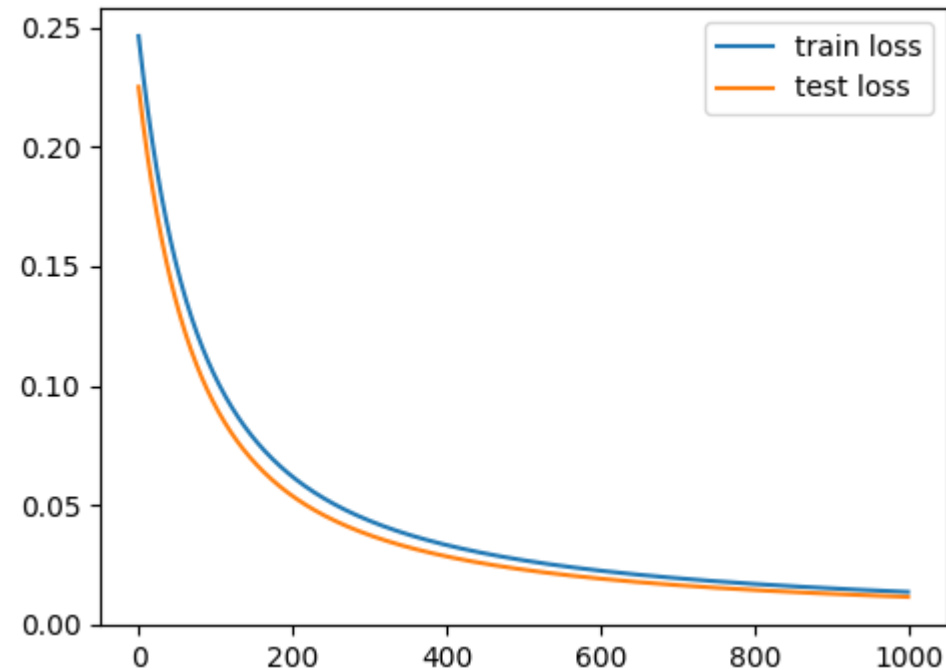
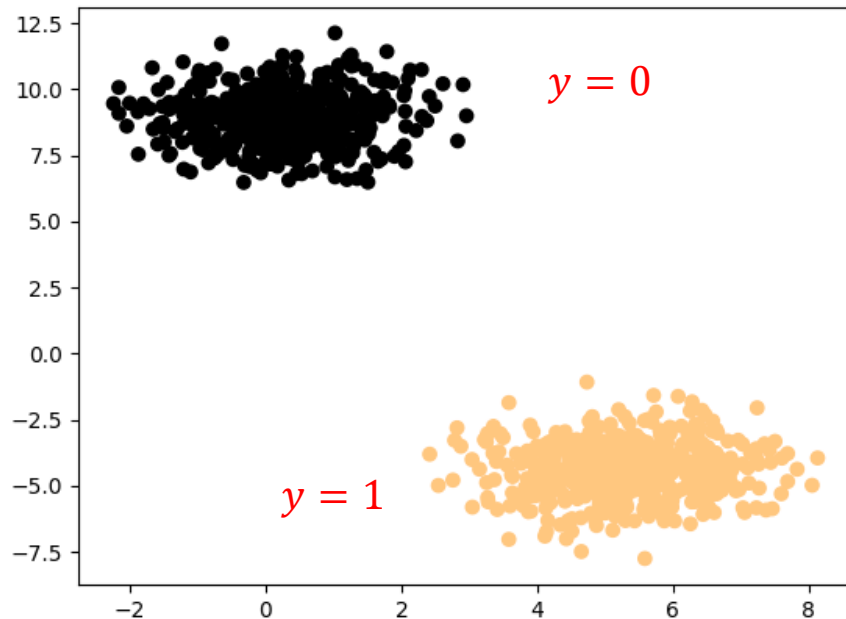
Phase d'apprentissage

Phase d'inférence

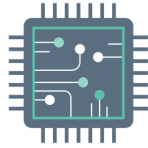
Epochs= 1000

$\alpha = 0,002$

Données d'entrée



Cost Function = erreur commise par le modèle



Neurone artificiel

Système à neurones

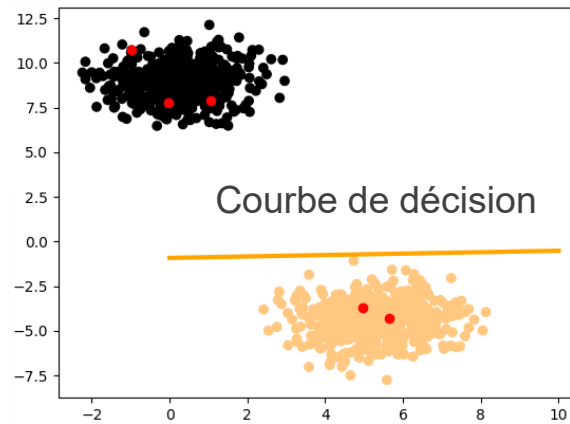
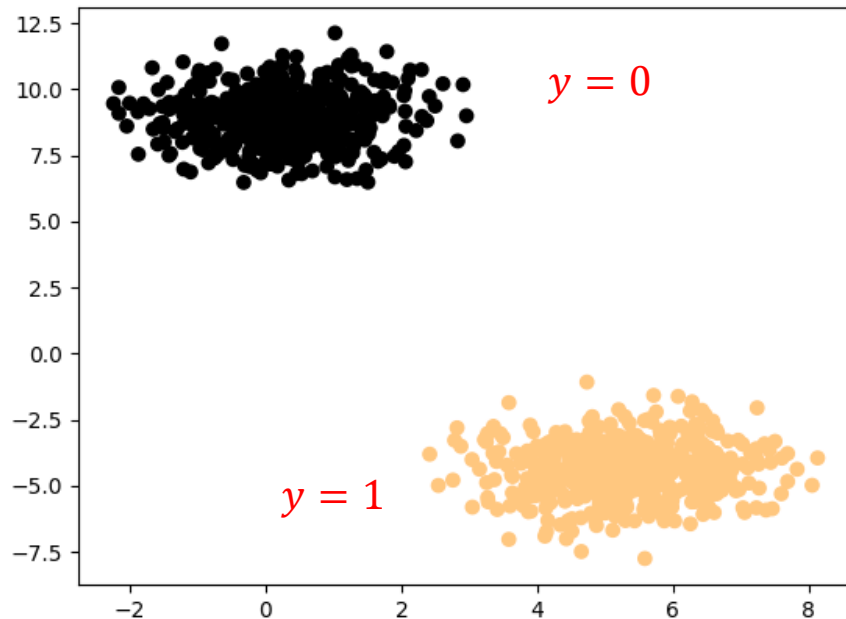
Phase d'apprentissage

Phase d'inférence

Epochs= 1000

$\alpha = 0,002$

Données d'entrée

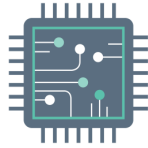


Données

Règles
Modèle

Ordinateur

Résultats



Machine Learning

Machine Learning

Phase d'apprentissage

Données d'entrée

Ajustement des paramètres du modèle

Minimisation de la fonction de coût

Époque : phase d'entraînement

Phase d'apprentissage

Application du modèle

Entrée → sortie prédite

Paramètres figés (modèle entraîné)

Cerveau Humain

Phase d'inférence

Stimuli sensoriels

Renforcement ou affaiblissement des connexions synaptiques

Maximisation du succès adaptatif (récompense, attention, survie)

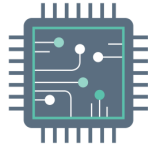
Jeune âge, enfance, apprentissage continu

Phase d'inférence

Application des connaissances

Stimulus → réponse appropriée

Connexions neuronales stables pour ce pattern



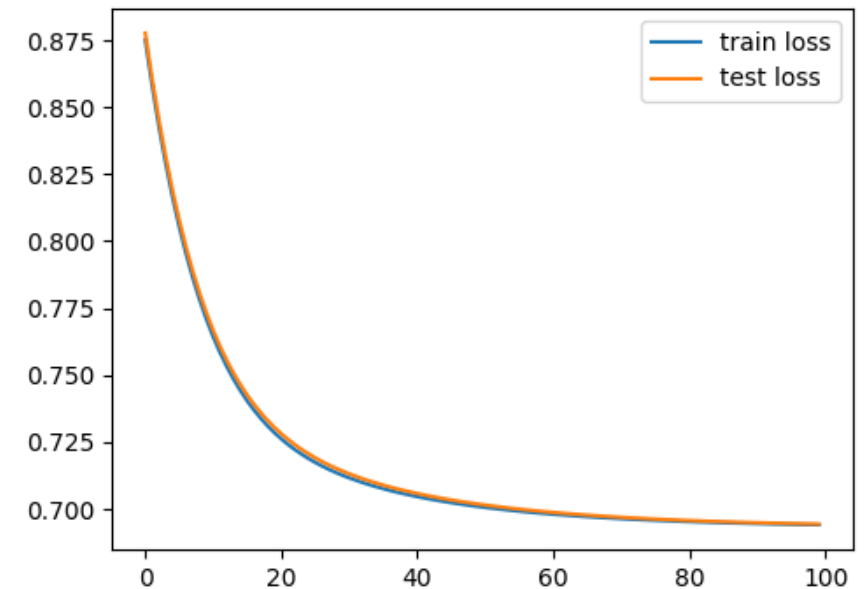
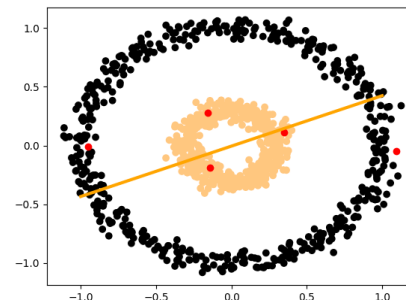
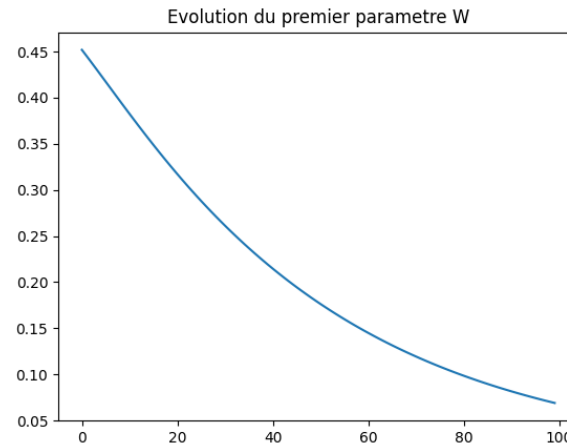
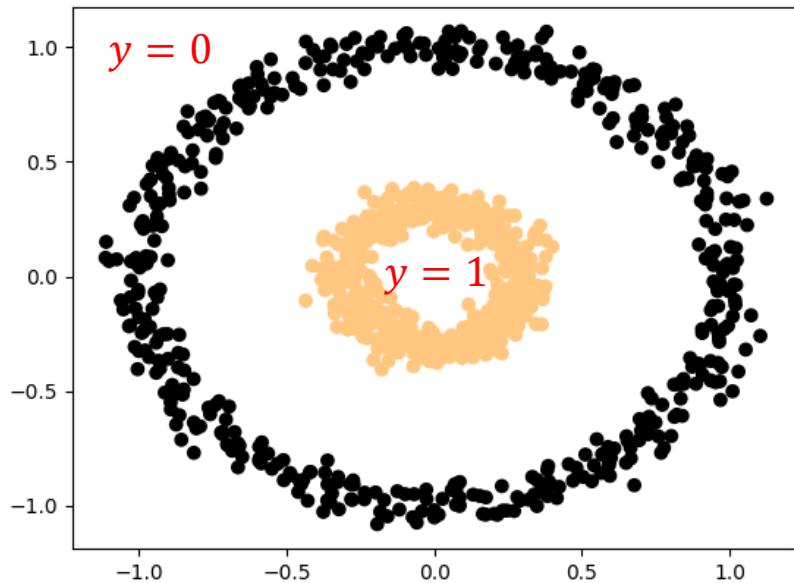
Neurone artificiel

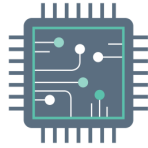
Système à neurones

Phase d'apprentissage

Phase d'inférence

Données d'entrée

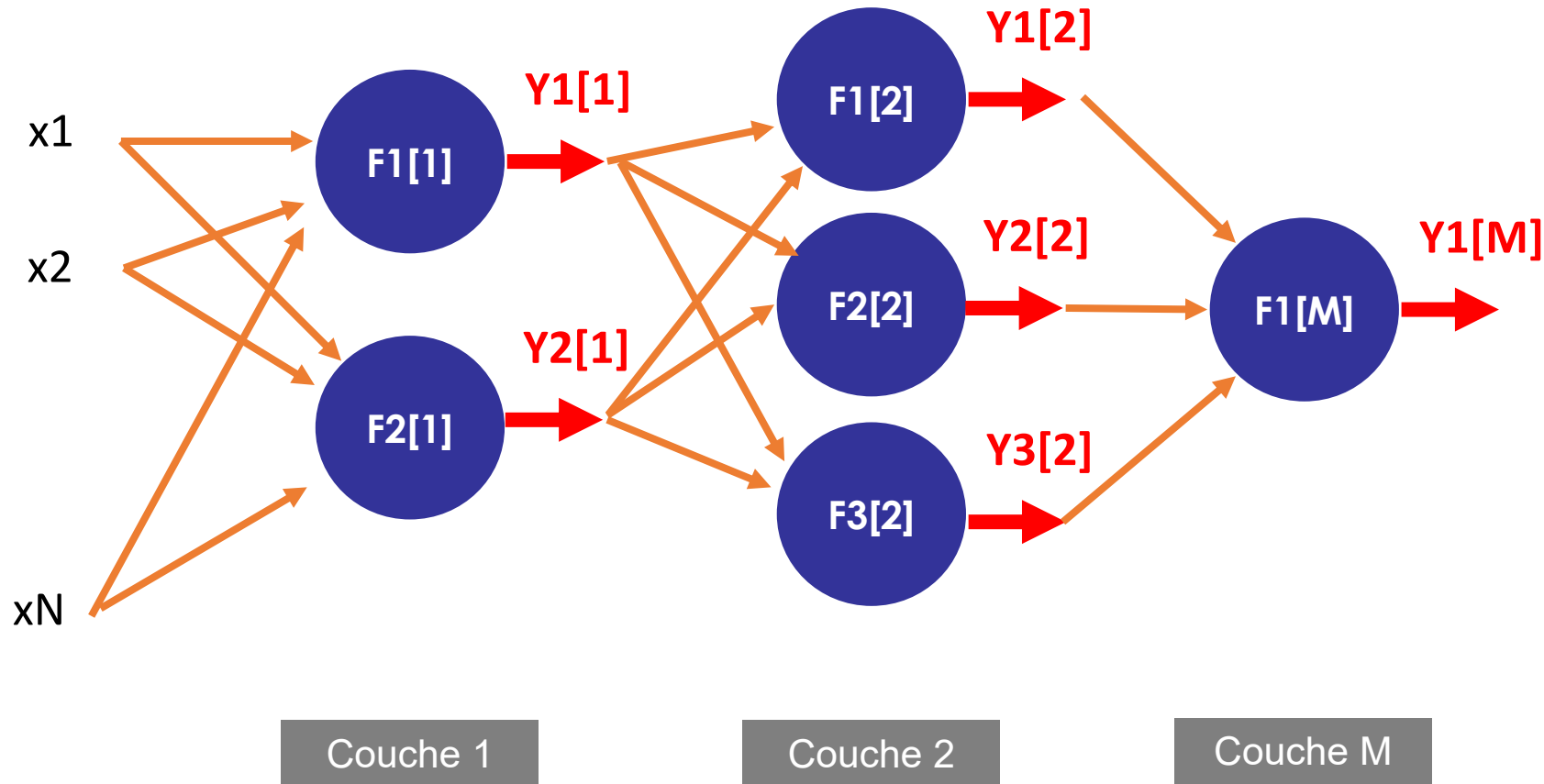




Réseau de neurones

Réseau de neurones

Phase d'apprentissage



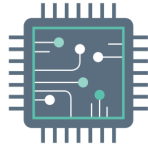
Etape 1 : *Forward Propagation*

- Calcul des $f_i[k]$
- Activation : calcul des $y_i[k]$

Etape 2 : *Cost Function / Loss*

- Comparaison de Y à la valeur attendue

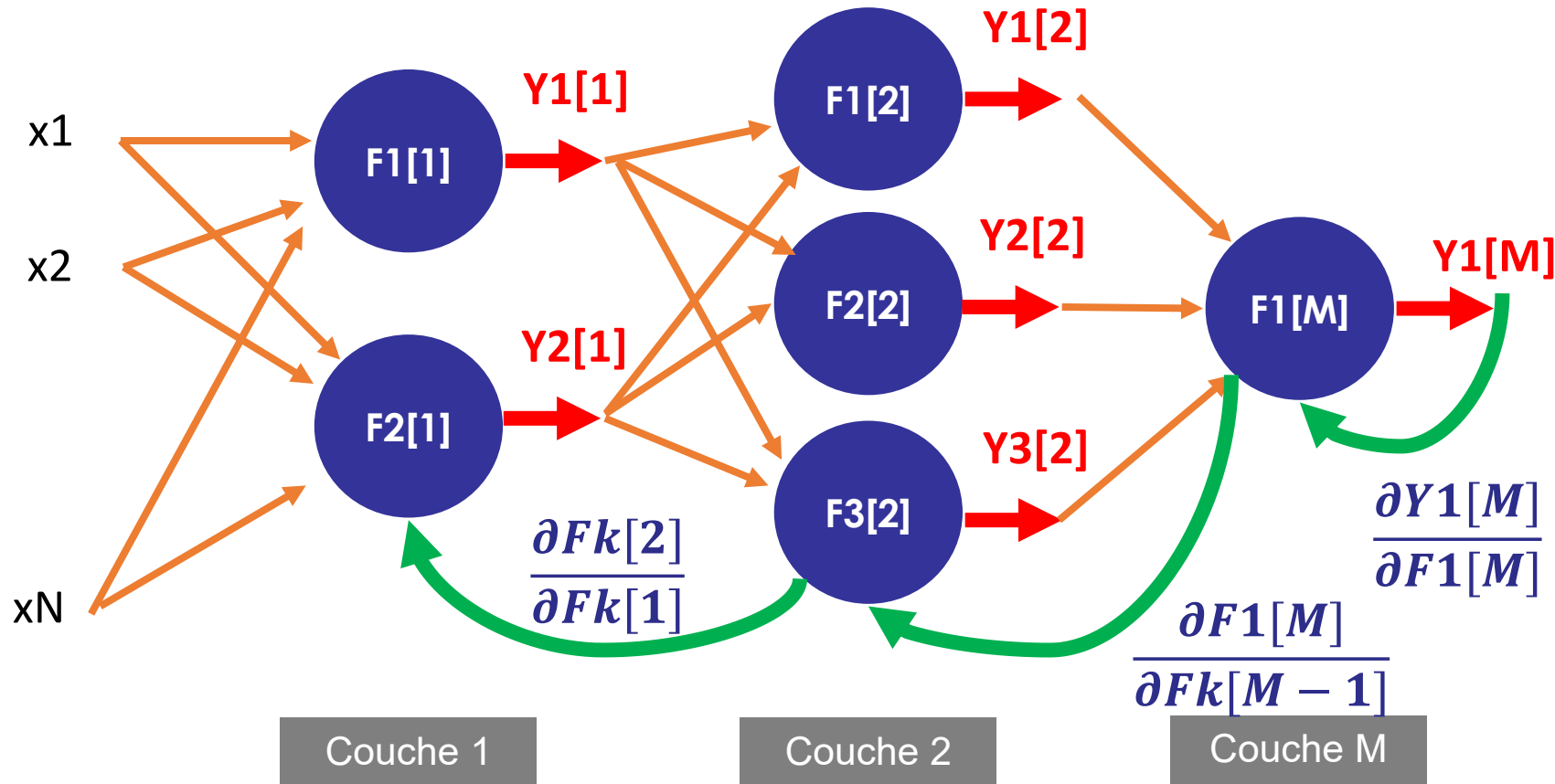




Réseau de neurones

Réseau de neurones

Phase d'apprentissage



Etape 1 : *Forward Propagation*

- Calcul des $f_i[k]$
- Activation : calcul des $y_i[k]$

Etape 2 : *Cost Function / Loss*

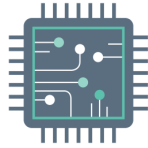
- Comparaison de Y à la valeur attendue

Etape 3 : *Back Propagation*

- Chaîne de gradients

Etape 4 : *Descente de gradients*

- Calcul des paramètres W pour minimiser l'erreur en sortie



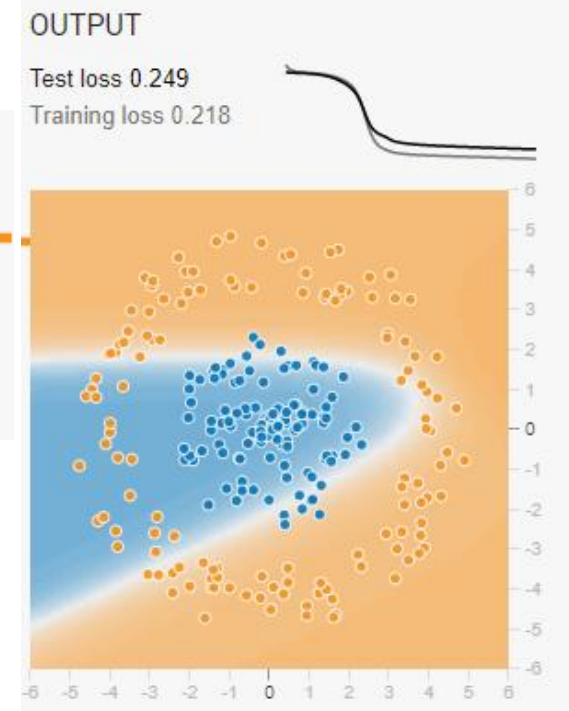
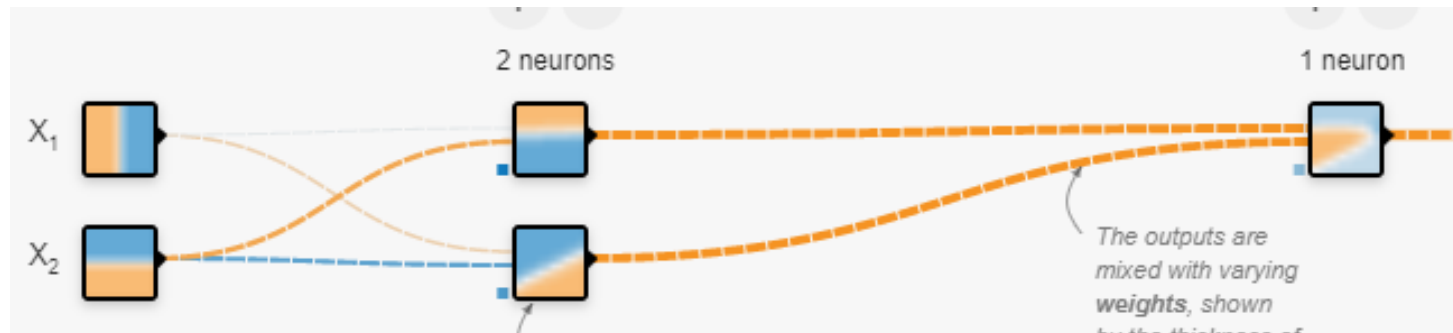
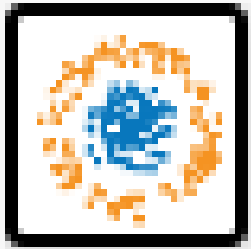
Réseau de neurones

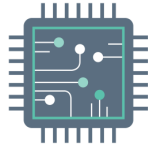
Réseau de neurones

Phase d'apprentissage

Epochs= 400

$\alpha = 0,01$





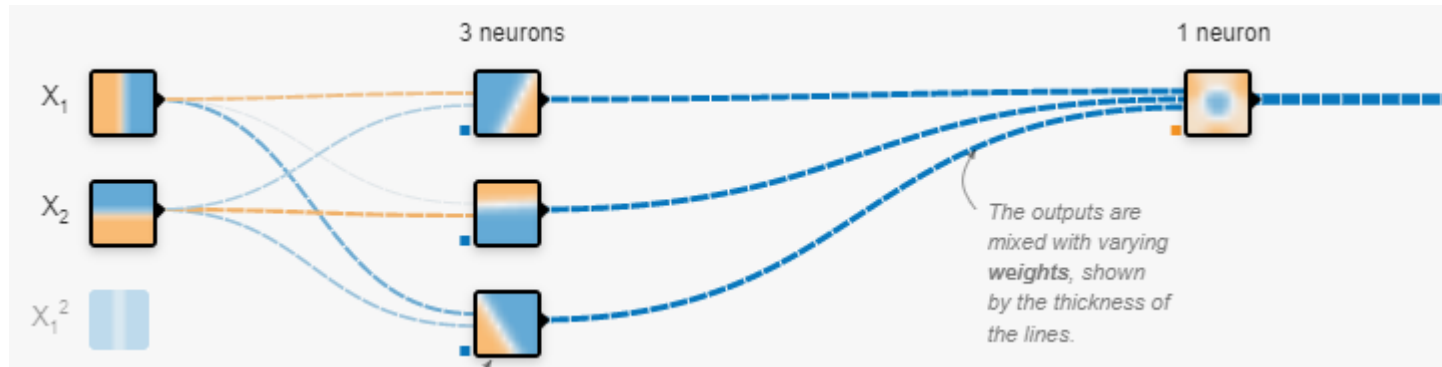
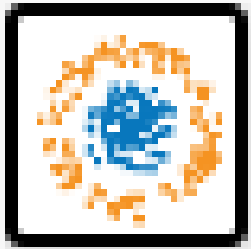
Réseau de neurones

Réseau de neurones

Phase d'apprentissage

Epochs= 170

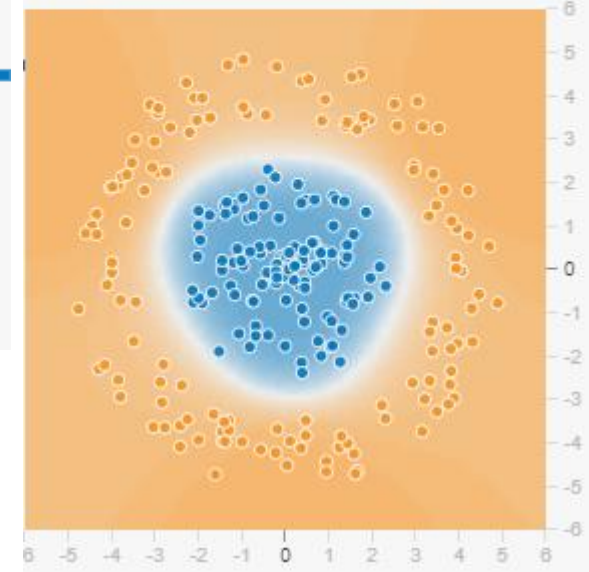
$\alpha = 0,01$



OUTPUT

Test loss 0.021

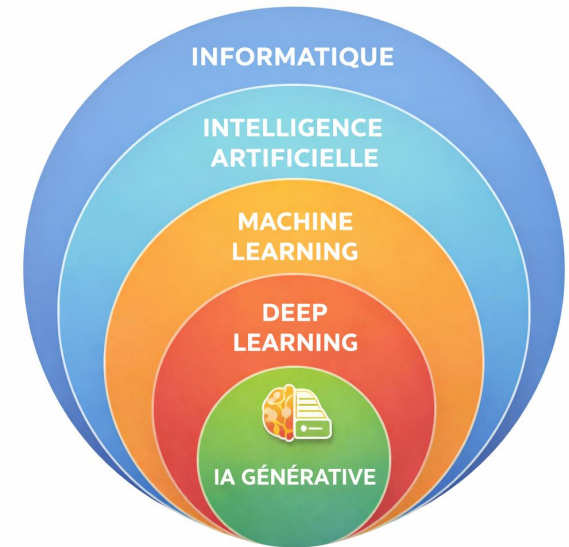
Training loss 0.020

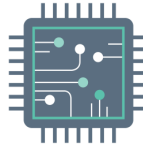




Machine Learning

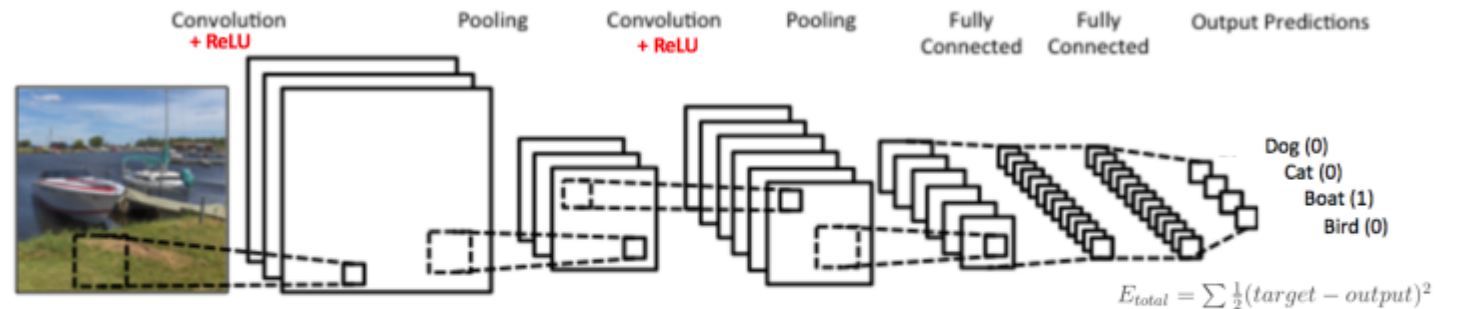
Réseaux plus complexes
Résumé





Réseau de neurones

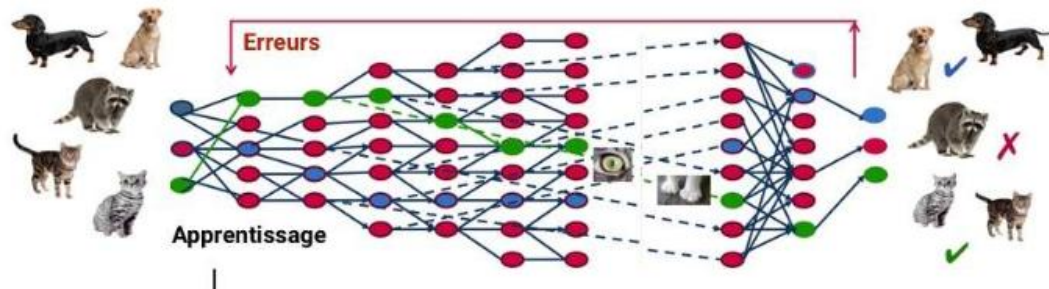
Réseau de neurones complexes

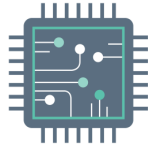


Feature Extraction from Image

Classification

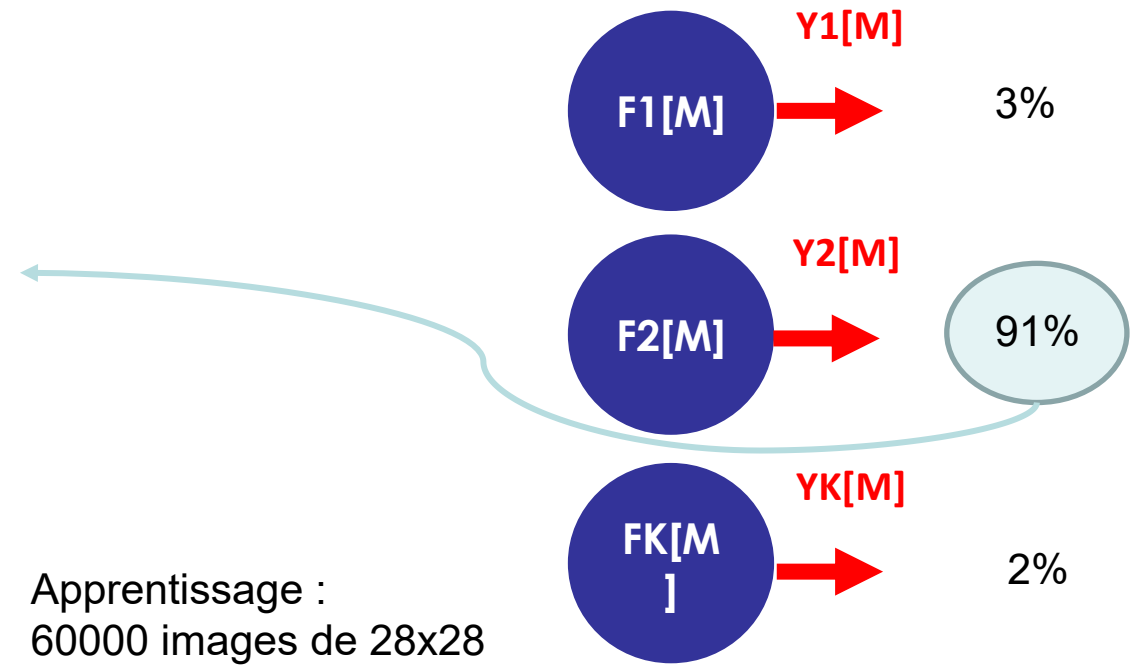
CNN classifier using 1D, 2D and 3D feature vectors
Site de MathWorks



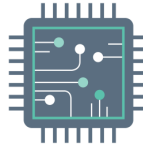


Réseau de neurones

Réseau de neurones complexes



Couche M



Réseau de neurones

Etapes de conception / mise en œuvre

Phase d'apprentissage

Préparation des données

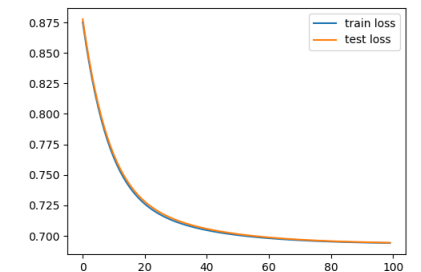
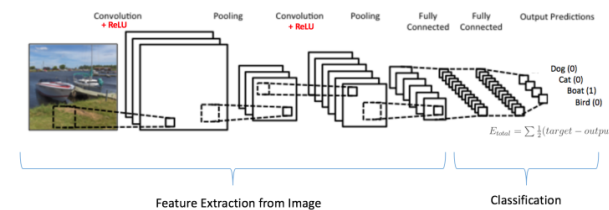
- Images de mêmes dimensions
- 2 sets différents pour entraînement et test
- Classification (résultat attendu)

Conception du modèle de réseau

- Dimensions données d'entrée
- Etages de convolution
- Etages de classification / Couches du réseau

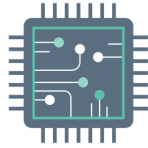
Entraînement du modèle

- Calcul des paramètres du modèle



Vérification avec données test

- Vérification du bon apprentissage













Réseau de neurones

Machine Learning : Quantité de Données d'Animaux Nécessaires

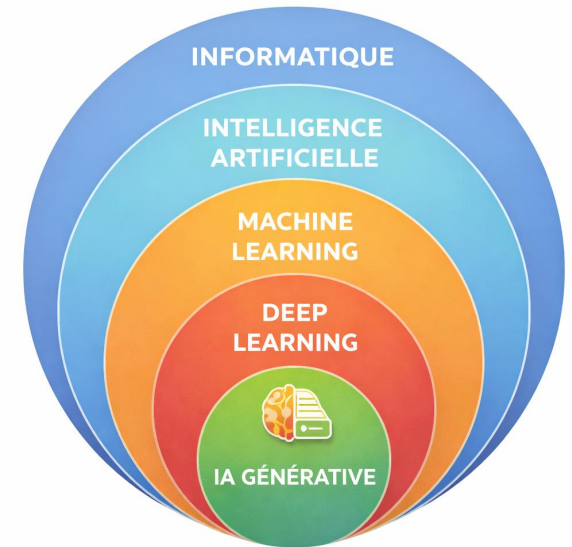
Pour distinguer 5 catégories d'animaux

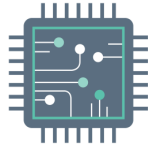


IMAGES		Modèle simple (SVM, CNN léger) 500 images par catégorie	 2500 ▶	~70-80%
		CNN standard (ResNet, EfficientNet) 2.000 images par catégorie	 10 000 ▶	~85-90%
		CNN avancé / haute précision 10.000 images par catégorie	 50 000 ▶	✦ 95%+
		Transfer Learning 200-500 images par catégorie	 1.000-2.500 ▶	~85-90%
		Transfer Learning 100-300 images au catégorie	 500-1.500 ▶	~85-90%
	+	Transfer Learning avancé + augmentation de données 100-300 images par catégorie, soit 500-1.500 images au total		~85-90% / ~85-90%



IA génératives





Définition

Système **capable de créer du contenu**, ressemblant à des données déjà acquises par le biais du *machine learning* (probabiliste)

a) Collecte des données

- Des milliards de phrases provenant de livres, articles, forums, code source, sites web...
- Données nettoyées pour enlever spam, contenus sensibles ou dangereux

b) Conversion en tokens

- découpage du texte en **tokens** : petites unités (mots ou morceaux de mots)

c) Apprentissage statistique

- Modèle apprenant **la probabilité qu'un token en suive un autre**

ChatGPT 5 environ **479 teraoctets** de texte brut, code ou metadonnées