

Introduction au connexionnisme

Licence de Sciences Cognitives (L3)

Didier Puzenat



1 Prologue...

- Organisation des cours et TD
- Continuité pédagogique
- L'IA selon Wikipédia
- L'IA dans la littérature, le cinéma, et la recherche

2 Introduction

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Planning et contrôle des connaissances

Intitulé : « UE Compléments 3 - Cognition Artificielle »

- CM, lundi 14h 15h45, salle L231 ;
- TD 1 et 2, mardi 14h 15h45 et 16h 17h45, salle H116 ;
- TD 3 et 4, jeudi 14h 15h45 et 16h 17h45, salle H114 ;

Contrôle des connaissances :

- CM : « écrit sur site » de 1 heure,
ou « dossier à distance » si (re)confinement.
- TD (et DA TD) : « dossier à distance »
→ reprend les projets faits en TD (code, réponses, etc.).

Codes Apogée : 3NCSD015 (CM) et 3NCSD025 (TP).

Cognition artificielle = programmation !

Langage : Python

En cas de besoin, (re)lire :

« Apprendre à programmer avec Python 3 »

gratuit, en français, assez complet (473 pages),

téléchargeable ici :

http://inforef.be/swi/download/apprendre_python3_5.pdf



Gérard Swinnen

**Apprendre à
programmer avec**

Python 3



Si vous passez beaucoup de temps dans les TCL

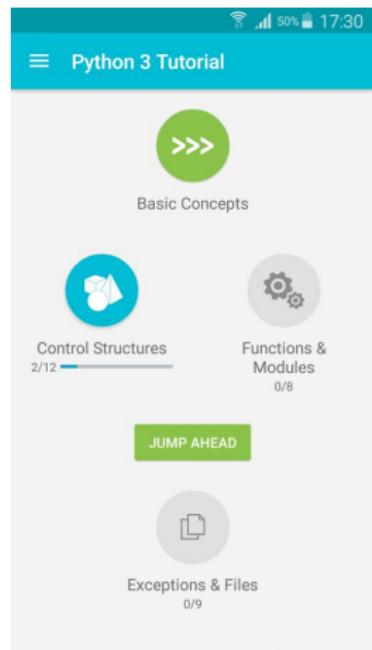
Il existe des applications Android pour apprendre très progressivement Python, par exemple :



Learn Python

SoloLearn Enseignement

PEGI 3



Mais ça ne remplace pas :

- un bon livre;
- la pratique !

1 Prologue...

- Organisation des cours et TD
- Continuité pédagogique
- L'IA selon Wikipédia
- L'IA dans la littérature, le cinéma, et la recherche

2 Introduction

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Moodle : au centre de la continuité pédagogique



- support du cours
- sondages et questionnaires

Moodle : au centre de la continuité pédagogique



- support du cours
- sondages et questionnaires
- cours en lui-même :
 - en visio via **BigBlueButton** (BBB) ?
 - via des « capsules vidéo » plus un forum ?
 - autrement ? (zoom, discord, teams, etc.)

Moodle : au centre de la continuité pédagogique



- support du cours
- sondages et questionnaires
- cours en lui-même :
 - en visio via **BigBlueButton** (BBB) ?
 - via des « capsules vidéo » plus un forum ?
 - autrement ? (zoom, discord, teams, etc.)
- aussi pour les TD ?

Moodle : au centre de la continuité pédagogique

The image shows two views of a Moodle course. On the left, a desktop browser window displays the course '3NCSD015-CM-Compléments 3 - Cognition artificielle (Didier Puzenat)'. The sidebar includes links for Participants, Badges, Compétences, Notes, Généralités, Support de cours, and sections 2 through 5. The main content area shows a 'Support de cours' section with five subsections: Section 2, Section 3, Section 4, and Section 5. On the right, a mobile application interface for the same course is shown, featuring tabs for Contenus, Participants, and Notes, with 'Toutes les sections' selected. The mobile view also lists 'Généralités', 'Questionnaire pour préparer un enseignement à distance', and 'Essai de visio durant le premier CM' under the 'Support de cours' section.

Clients : web, android, (a priori iphone)

Moodle : questionnaire COVID

Questionnaire pour préparer un enseignement à distance

Continuer

Mode: Le nom du participant sera enregistré et affiché avec ses réponses

1. Connexion internet à votre domicile*

Fibre optique

ADSL

4G

Mobile non 4G

2. Type de connexion(s) au domicile

Ordinateur

Tablette

Smartphone

Webcam

Casque

Microphone

3. Outils numériques

Ordinateur

Tablette

Smartphone

Webcam

4. Espace de travail personnel*

← Questionnaire pour préparer un en...

Mode

Le nom du participant sera enregistré et affiché avec ses réponses

1. Connexion internet à votre domicile *

2. Type de connexion(s) au domicile

Fibre optique

ADSL

4G

Mobile non 4G

3. Outils numériques

Ordinateur

Tablette

Smartphone

Webcam

À faire : questionnaire pour gérer au mieux le cours en distanciel

Moodle : essai de la visio avec BBB depuis le web

The screenshot shows a Moodle course page titled "Introduction à l'apprentissage automatique". A video player window is open, displaying the BigBlueButton interface. The interface includes a large blue "b" logo, the text "Welcome To BigBlueButton", and a message stating "BigBlueButton is an open source web conferencing system designed for online learning". Below this, there are several features listed with icons:

- CHAT: Send public and private messages.
- BREAKOUT ROOMS: Group users into breakout rooms for team collaboration.
- POLLING: Poll your users anytime.
- AUDIO: Communicate using high quality audio.
- SCREEN SHARING: Share your screen.
- EMOJIS: Express yourself.
- MULTI-USER WHITEBOARD: Draw together.

At the bottom of the video player, there is a link: "For more information visit bigbluebutton.org →".

Moodle : essai de la visio avec BBB depuis le web

The screenshot shows a Moodle course page titled "Bonne continuation". On the left, there's a sidebar with course navigation and a message from the teacher. The main content area displays a video player for a BBB session. The video frame shows the BigBlueButton logo and the text "Welcome To BigBlueButton". Below the video, it says "BigBlueButton is an open source web conferencing system designed for online learning". A central callout box contains the text "Ceci est un test d'écho privé. Prononcez quelques mots. Avez-vous entendu de l'audio ?" with "Oui" and "Non" buttons. Around this central box are icons for various features: CHAT, BREAKOUT ROOMS, POLLING, SCREEN SHARING, EMOJIS, and MULTI-USER WHITEBOARD.

For more information visit bigbluebutton.org →

Moodle : essai de la visio avec BBB depuis le web

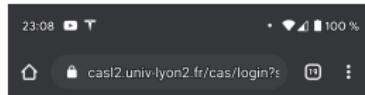
The screenshot shows a Moodle course page titled "Introduction". On the left, there's a sidebar with course navigation and a message from the teacher. The main content area displays a video player for a BBB session. The video frame shows a blue circular logo with a white letter 'b'. Below the video, the text reads "Welcome To BigBlueButton" and "BigBlueButton is an open source web conferencing system designed for online learning". A central modal window titled "Paramètres de la webcam" is open, showing a preview of a person's face and settings for camera and audio. Around this central window, there are icons and descriptions for various BBB features: CHAT, BREAKOUT ROOMS, POLLING, SCREEN SHARING, EMOJIS, and MULTI-USER WHITEBOARD.

For more information visit bigbluebutton.org →

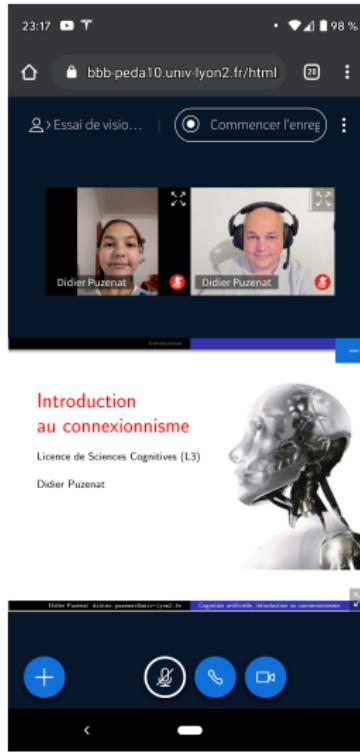
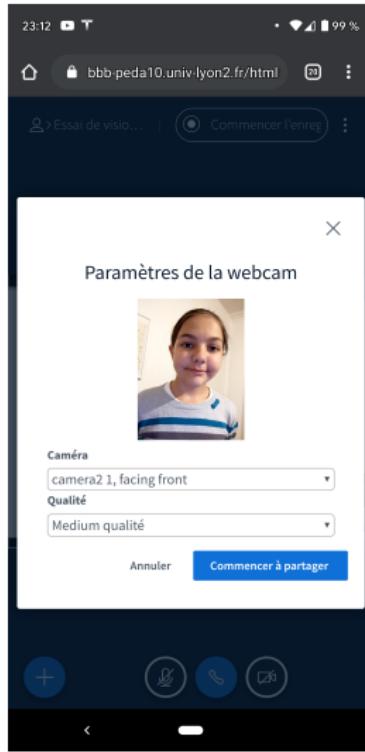
Moodle : essai de la visio avec BBB depuis le web

The screenshot shows a Moodle-based video conference interface. On the left, there's a sidebar with navigation links like 'MESSAGES', 'DÉCOURAGE', 'NOTES', 'ACTIVITÉS', 'LETS DISCUSS 0.1', and 'Didier Puzenat (mod)'. The main area has a video feed of a man wearing headphones. Below the video, a blue bar says 'Introduction'. To the right of the video is a slide with the title 'Introduction au connexionnisme' in large orange font, followed by 'Licence de Sciences Cognitives (L3)' and 'Didier Puzenat'. To the right of the slide is a stylized silver robotic head profile. At the bottom, there's a footer with 'Didier Puzenat didier.puzenat@univ-lyon2.fr' and 'Cognition artificielle, introduction au connexionnisme'. There are also various video control icons at the bottom.

Moodle : essai de la visio avec BBB depuis un smartphone

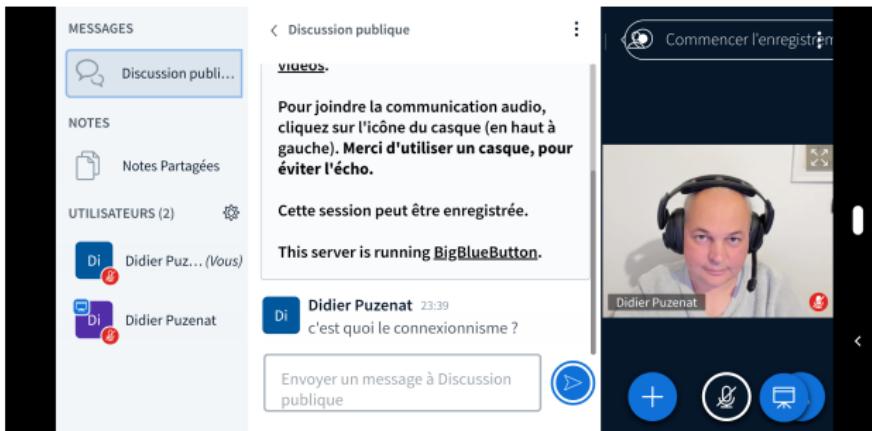


Moodle : essai de la visio avec BBB depuis un smartphone



Moodle : essai de la visio avec BBB depuis un smartphone

Et on peut aussi tenir le smartphone horizontalement...



Remarques :

- ça fonctionne aussi avec une tablette,
- voire avec certaines télévisions sous Android (pas testé).

1 Prologue...

- Organisation des cours et TD
- Continuité pédagogique
- L'IA selon Wikipédia
- L'IA dans la littérature, le cinéma, et la recherche

2 Introduction

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Définition de l'intelligence selon Wikipédia

Intelligence (« naturelle »)

L'intelligence est l'ensemble des facultés mentales permettant de comprendre les choses et les faits, de découvrir les relations entre elles et d'aboutir à la connaissance conceptuelle et rationnelle (par opposition à la sensation et à l'intuition). Elle permet de comprendre et de s'adapter à des situations nouvelles et peut en ce sens être également définie comme la faculté d'adaptation.

L'intelligence peut être également perçue comme la capacité à traiter l'information pour atteindre ses objectifs.

Et la **cognition artificielle** ? l'intelligence artificielle ?

1 Prologue...

- Organisation des cours et TD
- Continuité pédagogique
- L'IA selon Wikipédia
- L'IA dans la littérature, le cinéma, et la recherche

2 Introduction

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la [littérature et du cinéma](#)



Asimov : cerveau positronique, les lois de la robotique (pour le meilleur et le pire), etc.

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la [littérature et du cinéma](#)



Data : encore du positronique, guidé par la recherche d'une humanité (par exemple via les émotions)...

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la [littérature et du cinéma](#)



L'agent Smith : une IA purement logicielle, a priori sans émotion et avec un complexe de supériorité (puis une IA dépassée qui fait de la résistance)

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la littérature et du cinéma



TRON le simple programme, le MCP un programme qui a très « mal tourné » et Quorra une « IA » non programmée qui émerge de la complexité de son environnement

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la littérature et du cinéma



Jane (l'IA amie de Ender) : émerge également de la complexité, reconnue par Ender comme étant une nouvelle forme de vie
« consciente »

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la littérature et du cinéma



Caprica (l'ado, pas celle avec la robe rouge ;-): les premiers pas du transhumanisme

L'IA dans la littérature et au cinéma (et dans les séries)

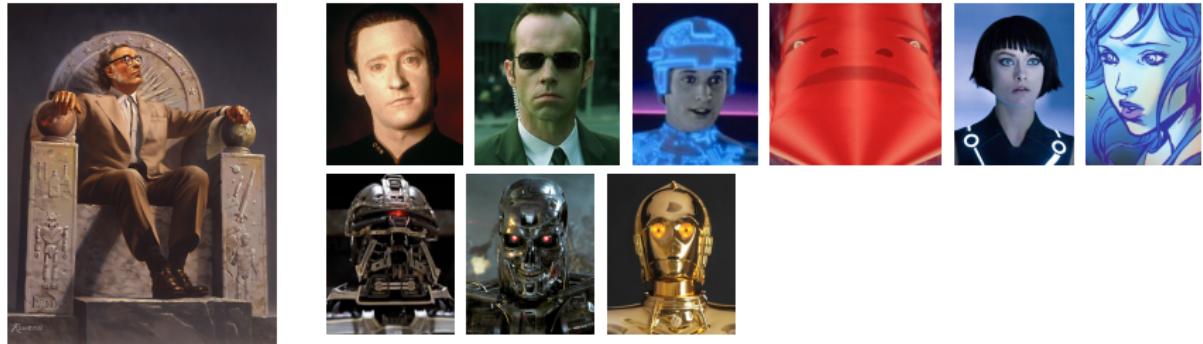
Quelques exemples d'IA dans la [littérature et du cinéma](#)



Terminator : le fantasme de l'IA qui prend le pouvoir (skynet) avec une armée de robots plus ou moins évolués (dont un T2 doué d'une certaine capacité d'apprentissage)

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la [littérature et du cinéma](#)



Z-6PO (aka C-3PO) : ne comprend rien aux humains mais efficace dans son domaine d'expertise (traducteur)

L'IA dans la littérature et au cinéma (et dans les séries)

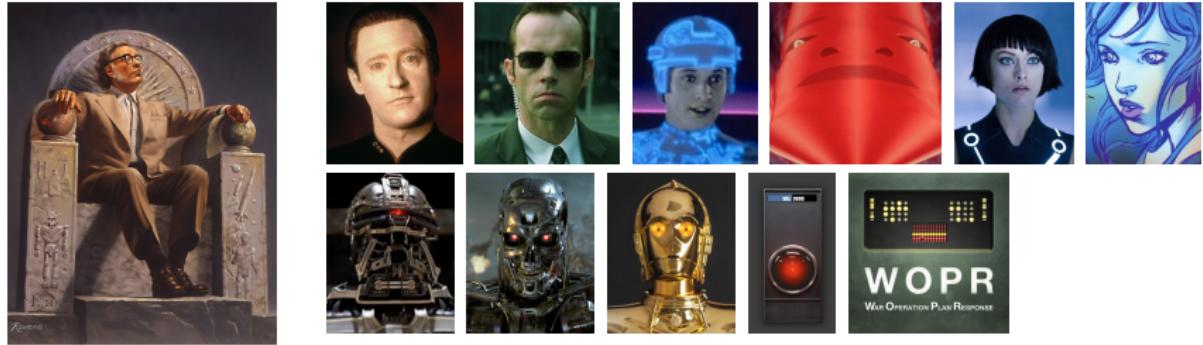
Quelques exemples d'IA dans la littérature et du cinéma



Hall 9000 : pour moi... le danger de l'anthropomorphisme (des humains) dans la relation IA-humain

L'IA dans la littérature et au cinéma (et dans les séries)

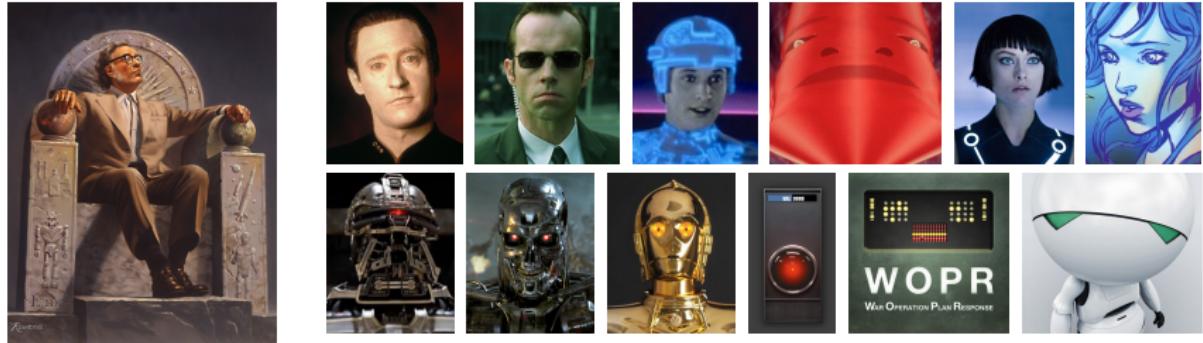
Quelques exemples d'IA dans la littérature et du cinéma



Le WOPR : il apprend très bien, pour lui le nombre de morts n'est qu'une variable, mais heureusement il préfère les échecs à la guerre thermonucléaire globale

L'IA dans la littérature et au cinéma (et dans les séries)

Quelques exemples d'IA dans la littérature et du cinéma



Marvin : l'IA fidèle mais dépressive car beaucoup trop intelligente pour être heureuse

L'IA dans la littérature et au cinéma (et dans les séries)

Réaliste pour les années 2020 (2020-2030) :

- Hall 9000 : prévu pour tenir compagnie, assistant numérique, joue aux échecs, identifie les personnes, lit sur les lèvres, prend des décisions en conséquence.
- Z-6PO et D2R2 (voire le T2) : comprend son environnement physique mais pas humain, capacités moteurs limitées.
- WOPR (et Skynet ?) : de l'aide à la décision à la... décision.

L'IA dans la littérature et au cinéma (et dans les séries)

Réaliste pour les années 2020 (2020-2030) :

- Hall 9000 : prévu pour tenir compagnie, assistant numérique, joue aux échecs, identifie les personnes, lit sur les lèvres, prend des décisions en conséquence.
- Z-6PO et D2R2 (voire le T2) : comprend son environnement physique mais pas humain, capacités moteurs limitées.
- WOPR (et Skynet ?) : de l'aide à la décision à la... décision.

Les autres :

pour le moment encore du domaine de la SF
→ système à base de règles sans explosion combinatoire ?
cerveau positronique ? vie émergente ? singularité ?
scan du cerveau humain ? et circuit suffisant pour un tel scan ?

1 Prologue...

2 **Introduction**

- Intelligence Artificielle symbolique et numérique
- Le neurone formel et le perceptron
- Apprentissage et Généralisation
- Topologie des réseaux de neurones
- Les modèles à base de prototypes

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Le Connexionnisme est au centre de recherches pluridisciplinaires

Domaines s'intéressant aux réseaux de neurones artificiels :

- **Biologie** : neuropsychologie et fonctionnement du cerveau, réseaux de neurones . . .

Le Connexionnisme est au centre de recherches pluridisciplinaires

Domaines s'intéressant aux réseaux de neurones artificiels :

- **Biologie** : neuropsychologie et fonctionnement du cerveau, réseaux de neurones...
- **Statistiques** : alternative aux méthodes classiques

Le Connexionnisme est au centre de recherches pluridisciplinaires

Domaines s'intéressant aux réseaux de neurones artificiels :

- **Biologie** : neuropsychologie et fonctionnement du cerveau, réseaux de neurones...
- **Statistiques** : alternative aux méthodes classiques
- **Informatique fondamentale** : mémoire associative, théorie des graphes, réseau d'automates...

Le Connexionnisme est au centre de recherches pluridisciplinaires

Domaines s'intéressant aux réseaux de neurones artificiels :

- **Biologie** : neuropsychologie et fonctionnement du cerveau, réseaux de neurones...
- **Statistiques** : alternative aux méthodes classiques
- **Informatique fondamentale** : mémoire associative, théorie des graphes, réseau d'automates...
- **Informatique appliquée** : reconnaissance de formes, robotique, traitement de la parole... parallélisme massif, circuit VLSI, tolérance aux pannes...

Le Connexionnisme est au centre de recherches pluridisciplinaires

Domaines s'intéressant aux réseaux de neurones artificiels :

- **Biologie** : neuropsychologie et fonctionnement du cerveau, réseaux de neurones...
- **Statistiques** : alternative aux méthodes classiques
- **Informatique fondamentale** : mémoire associative, théorie des graphes, réseau d'automates...
- **Informatique appliquée** : reconnaissance de formes, robotique, traitement de la parole... parallélisme massif, circuit VLSI, tolérance aux pannes...
- **Sciences Cognitives** : modélisation (mémoire, émotions, raisonnement...), validation d'hypothèses...

Les réseaux de neurones

- **Définition en biologie** Cellules vivantes mettant en œuvre une très grande quantité de neurones connectés par des synapses.
= siège de la mémoire, de l'intelligence, de la conscience... .

Les réseaux de neurones

- **Définition en biologie** Cellules vivantes mettant en œuvre une très grande quantité de neurones connectés par des synapses.
= siège de la mémoire, de l'intelligence, de la conscience... .
- **Définition en informatique** Ensemble d'automates élémentaires interconnectés qui s'influencent par l'intermédiaire de ces connexions.
⇒ peut être simulé sur un ordinateur !
 - application : e.g. science de l'ingénieur
 - recherche : e.g. test d'hypothèse biologiques

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
logique séquentiel localisé programmation	analogique parallèle distribué adaptation

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique séquentiel localisé programmation	<i>analogique</i> → <i>données brutes</i> parallèle distribué adaptation

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique séquentiel localisé programmation	<i>analogique</i> → <i>données brutes</i> <i>parallèle</i> → <i>très rapide</i> <i>distribué</i> <i>adaptation</i>

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique séquentiel localisé programmation	<i>analogique</i> → <i>données brutes</i> <i>parallèle</i> → <i>très rapide</i> <i>distribué</i> → <i>tolérance aux pannes</i> <i>adaptation</i>

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique séquentiel localisé programmation	<i>analogique</i> → <i>données brutes</i> <i>parallèle</i> → <i>très rapide</i> <i>distribué</i> → <i>tolérance aux pannes</i> <i>adaptation</i> → <i>pas de développement</i>

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- l'IA numérique = les réseaux de neurones artificiels
 - l'IA symbolique = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<p><i>données structurées ← logique maîtrisé ! ← séquentiel localisé programmation</i></p>	<p><i>analogique → données brutes parallèle → très rapide distribué → tolérance aux pannes adaptation → pas de développement</i></p>

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique <i>maîtrisé !</i> ← séquentiel <i>rassurant !</i> ← localisé programmation	<i>analogique</i> → <i>données brutes</i> <i>parallèle</i> → <i>très rapide</i> <i>distribué</i> → <i>tolérance aux pannes</i> <i>adaptation</i> → <i>pas de développement</i>

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique <i>maîtrisé !</i> ← séquentiel <i>rassurant !</i> ← localisé <i>connu !</i> ← programmation	<i>analogique</i> → <i>données brutes</i> <i>parallèle</i> → <i>très rapide</i> <i>distribué</i> → <i>tolérance aux pannes</i> <i>adaptation</i> → <i>pas de développement</i>

Comparaison IA symbolique / numérique

L'Intelligence Artificiel comprend :

- **l'IA numérique** = les réseaux de neurones artificiels
- **l'IA symbolique** = « systèmes experts », « logique floue »

Intelligence Artificielle « IA symbolique »	Connexionnisme « IA numérique »
<i>données structurées</i> ← logique <i>maîtrisé !</i> ← séquentiel <i>rassurant !</i> ← localisé <i>connu !</i> ← programmation	<i>analogique</i> → <i>données brutes</i> <i>parallèle</i> → <i>très rapide</i> <i>distribué</i> → <i>tolérance aux pannes</i> <i>adaptation</i> → <i>pas de développement</i>

Conclusion : l'IA numérique a eu du mal à percer !

Choisir entre IA symbolique et numérique

- **IA symbolique** à utiliser pour des problèmes de « haut niveau », lorsque l'on possède déjà :
 - des variables clairement identifiées
 - une très bonne expertise sur le domaine (les règles)
- **IA numérique** à utiliser pour les données brutes, éventuellement avec du bruit, peu structurées...

Choisir entre IA symbolique et numérique

- **IA symbolique** à utiliser pour des problèmes de « haut niveau », lorsque l'on possède déjà :
 - des variables clairement identifiées
 - une très bonne expertise sur le domaine (les règles)

Par exemple :

- aide à la décision, diagnostique...
- modélisation de processus cognitifs de haut niveau

- **IA numérique** à utiliser pour les données brutes, éventuellement avec du bruit, peu structurées...

Par exemple :

- classification, prévision...
- modélisation de pré-traitements intelligents

Les fondements biologiques du connexionnisme

- **XVII** Cerveau = rôle central comme organe de commande
- **XIX** début de la neuropsychologie
 - = rapport entre « la matière » et la fonction
 - = les fonctions cérébrales sont localisées et cartographiées

Les fondements biologiques du connexionnisme

- **XVII** Cerveau = rôle central comme organe de commande
- **XIX** début de la neuropsychologie
 - = rapport entre « la matière » et la fonction
 - = les fonctions cérébrales sont localisées et cartographiées

Ordres de grandeur comparés des réseaux de neurones :

- système nerveux central : 10^{11} neurones
 - un processeur récent (RTX 3080 GPU) : 10^{10} transistors)
- 1 mm³ de néo-cortex :
 - 60 000 neurones

Les fondements biologiques du connexionnisme

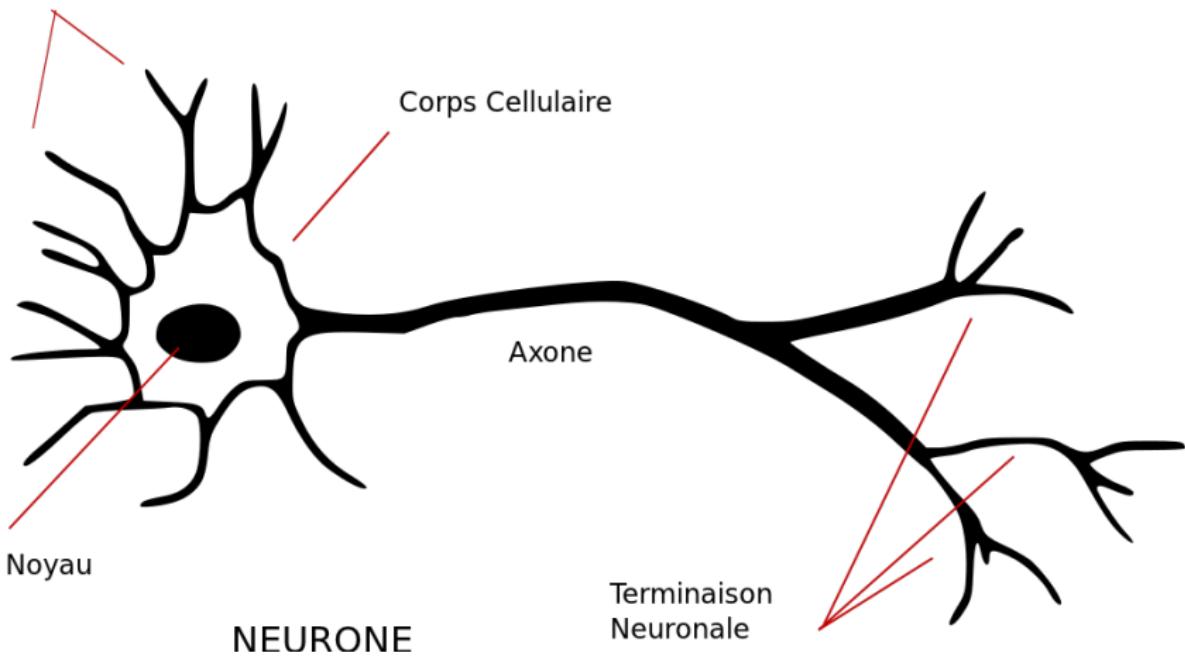
- **XVII** Cerveau = rôle central comme organe de commande
- **XIX** début de la neuropsychologie
 - = rapport entre « la matière » et la fonction
 - = les fonctions cérébrales sont localisées et cartographiées

Ordres de grandeur comparés des réseaux de neurones :

- système nerveux central : 10^{11} neurones
 - un processeur récent (RTX 3080 GPU) : 10^{10} transistors)
- 1 mm³ de néo-cortex :
 - 60 000 neurones
 - $3 \cdot 10^9$ synapses !
 - 1,5 km de dendrites !
 - 3 km d'axone !

Les fondements biologiques du connexionnisme

Dendrites



Rappel du fonctionnement d'un neurone biologique

Le rôle du neurone biologique est de :

- ① recevoir l'influx des autres neurones, via des dendrites
- ② « intégrer » les influx reçus, dans le corps cellulaire
- ③ engendrer éventuellement un influx nerveux
- ④ transmettre l'influx à un autre neurone, via l'axone
(l'axone fait de 1 mm à 1 m !)

Attention : l'influx, le « potentiel d'action », est codé en fréquence
⇒ tous les potentiels d'action ont la même amplitude

1 Prologue...

2 Introduction

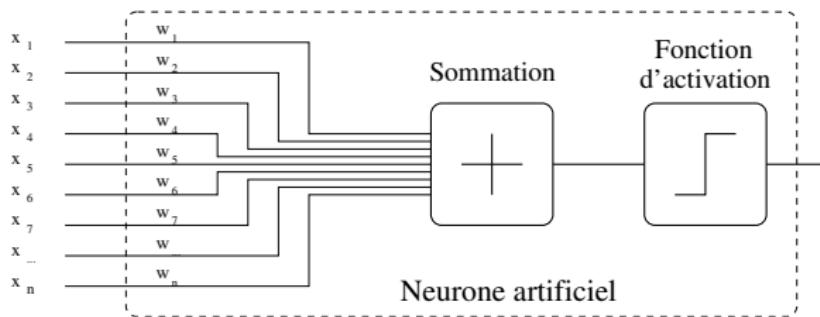
- Intelligence Artificielle symbolique et numérique
- Le neurone formel et le perceptron
- Apprentissage et Généralisation
- Topologie des réseaux de neurones
- Les modèles à base de prototypes

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Le neurone formel de Mc Culloch et Pitts (1943)



- vecteur d'entrée $X = (x_1, x_2, \dots, x_n)$
- vecteur poids $W = (w_1, w_2, \dots, w_n)$
- la sortie dépend de la fonction d'activation

Remarque : on est passé à un codage en amplitude !

Exemple de « l'automate booléen à seuil »

Description de l'automate booléen à seuil :

- vecteur d'entrée booléen
- fonction d'activation de Heaviside (notée \mathcal{H})
⇒ sortie aussi booléenne
- poids et le seuil réels

Fonctionnement :

- ① calcul de la somme pondérée des entrées : $\sum_{i=1}^n w_i x_i$
- ② calcul de la sortie de la cellule :

$$s = \mathcal{H}\left(\sum_{i=1}^n w_i x_i - \theta\right) = \begin{cases} 1 & \text{si } \sum_{i=1}^n w_i x_i > \theta \\ 0 & \text{si } \sum_{i=1}^n w_i x_i \leq \theta \end{cases}$$

Présentation du perceptron de Rosenblatt (1958)

Description du perceptron :

- entrées réelles
- fonction d'activation de Heaviside (notée \mathcal{H})
⇒ sortie encore booléenne
- poids et le seuil réels

Principe : un perceptron, avec 1 neurone, divise l'espace d'entrée \mathbb{R}^n en deux (et seulement deux) sous-espaces séparés par un hyperplan d'équation :

$$w_1x_1 + \dots + x_iw_i + \dots + x_nw_n - \theta = 0$$

⇒ on va pouvoir faire un « choix » ⇒ de la classification

Interprétation géométrique du perceptron

Si on se limite à \mathbb{R}^2 : le perceptron divise l'espace d'entrée en deux demi-plans par une droite d'équation :

$$w_1x_1 + x_2w_2 - \theta = 0$$

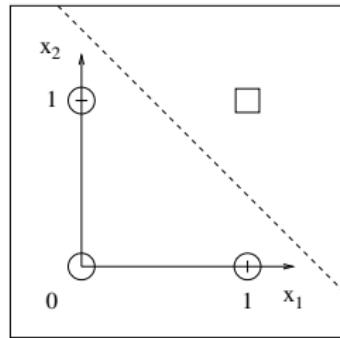
- ⇒ selon l'intensité de l'influx en entrée de la cellule, on considère un demi-plan ou l'autre
- ⇒ le perceptron est capable d'effectuer une discrimination de l'espace d'entrée en deux classes **linéairement séparables**.

Simulation des fonctions ET , OU , et XOR

x_1	x_2	ET	OU	OU exclusif
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

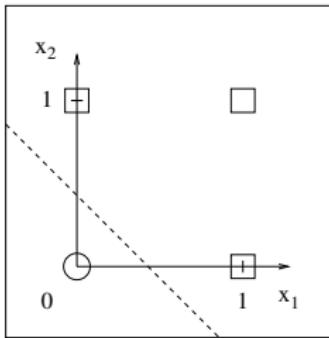
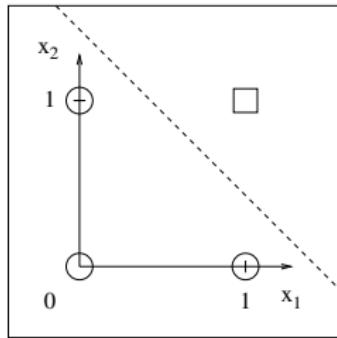
Simulation des fonctions *ET*, *OU*, et *XOR*

x_1	x_2	<i>ET</i>	<i>OU</i>	<i>OU exclusif</i>
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0



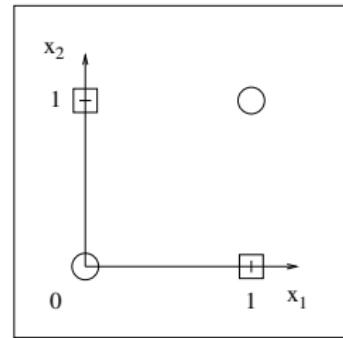
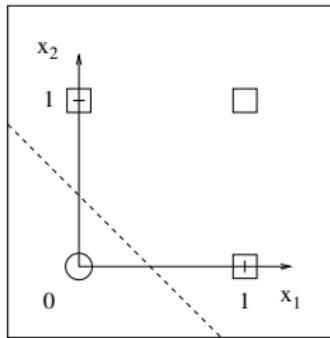
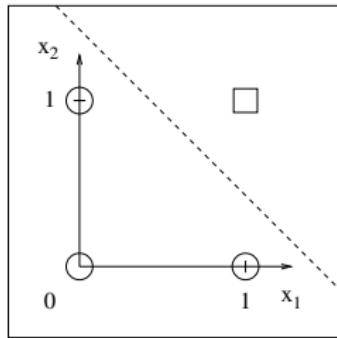
Simulation des fonctions *ET*, *OU*, et *XOR*

x_1	x_2	<i>ET</i>	<i>OU</i>	<i>OU exclusif</i>
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0



Simulation des fonctions *ET*, *OU*, et *XOR*

x_1	x_2	<i>ET</i>	<i>OU</i>	<i>OU exclusif</i>
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0



Le problème du « ou exclusif » (XOR)

⇒ Impossible de discriminer les valeurs de vérité du *OU exclusif*!

La limitation est significative car le pourcentage de fonctions booléennes linéairement séparables diminue très rapidement si on augmente la dimension de l'espace d'entrée (n) :

n	séparable	total
2	14	16
3	104	256
4	1882	65536
5	94572	$4,3 \cdot 10^9$
6	5028134	$1,8 \cdot 10^{19}$

Théorème de McCulloch et Pitts

Remarque : on peut simuler un NON logique

$$\begin{cases} 0 \times w_1 + 0 - \theta > 0 \\ 1 \times w_1 + 0 - \theta \leq 0 \end{cases} \quad \begin{cases} \theta < 0 \\ w_1 \leq \theta \end{cases} \quad \begin{cases} \theta = -0,5 \\ w_1 = -1 \end{cases}$$

Théorème de McCulloch et Pitts

Remarque : on peut simuler un NON logique

$$\begin{cases} 0 \times w_1 + 0 - \theta > 0 \\ 1 \times w_1 + 0 - \theta \leq 0 \end{cases} \quad \begin{cases} \theta < 0 \\ w_1 \leq \theta \end{cases} \quad \begin{cases} \theta = -0,5 \\ w_1 = -1 \end{cases}$$

⇒ on a les 3 opérateurs de la logique booléenne

⇒ on peut synthétiser toutes les fonctions booléennes

Théorème de McCulloch et Pitts

Remarque : on peut simuler un NON logique

$$\begin{cases} 0 \times w_1 + 0 - \theta > 0 \\ 1 \times w_1 + 0 - \theta \leq 0 \end{cases} \quad \begin{cases} \theta < 0 \\ w_1 \leq \theta \end{cases} \quad \begin{cases} \theta = -0,5 \\ w_1 = -1 \end{cases}$$

⇒ on a les 3 opérateurs de la logique booléenne

⇒ on peut synthétiser toutes les fonctions booléennes

Théorème : Toute fonction booléenne de n variables peut être synthétisée par un **réseau** d'automates à seuil possédant n entrées

Théorème de McCulloch et Pitts

Remarque : on peut simuler un NON logique

$$\begin{cases} 0 \times w_1 + 0 - \theta > 0 \\ 1 \times w_1 + 0 - \theta \leq 0 \end{cases} \quad \begin{cases} \theta < 0 \\ w_1 \leq \theta \end{cases} \quad \begin{cases} \theta = -0,5 \\ w_1 = -1 \end{cases}$$

⇒ on a les 3 opérateurs de la logique booléenne

⇒ on peut synthétiser toutes les fonctions booléennes

Théorème : Toute fonction booléenne de n variables peut être synthétisée par un **réseau** d'automates à seuil possédant n entrées
... mais encore faut-il savoir le construire !

1 Prologue...

2 **Introduction**

- Intelligence Artificielle symbolique et numérique
- Le neurone formel et le perceptron
- Apprentissage et Généralisation
- Topologie des réseaux de neurones
- Les modèles à base de prototypes

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Apprentissage et Généralisation

Objectif : trouver les w_i et θ automatiquement
en présentant une base d'exemples au réseau

Apprentissage et Généralisation

Objectif : trouver les w_i et θ automatiquement
en présentant une base d'exemples au réseau

Il faut :

- un algorithme **d'apprentissage**
- une base d'exemples
= couples (vecteur d'entrée X , sortie désirée pour X)

Exemple : algorithme d'apprentissage du perceptron

Algorithme :

- ① On présente un exemple = on calcule la sortie du réseau
- ② Si la sortie ne correspond pas à la sortie désirée
= si on est pas du bon côté de l'hyperplan
⇒ on modifie les poids et le seuil
- ③ On boucle...

Remarques : l'algorithme converge toujours
... si c'est linéairement séparable !

La généralisation

Définition : la généralisation est la **réponse du réseau sur des exemples qui n'étaient pas dans la base d'apprentissage**

Objectifs :

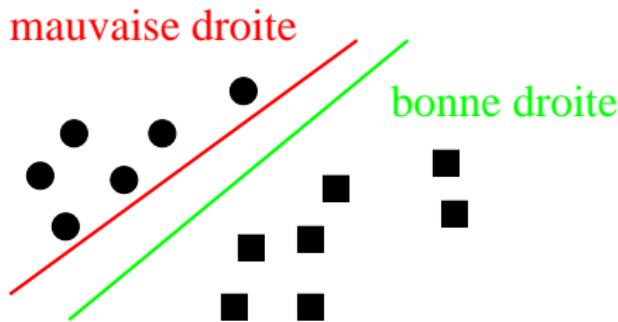
- connaître la réponse pour des exemples « inconnus » couples (vecteur d'entrée X , ?)
- évaluer la qualité de l'apprentissage couples (vecteur d'entrée X , sortie connue)

Performances en généralisation

Les performances en généralisation dépendent :

- de la qualité de l'algorithme d'apprentissage
- de l'adéquation du réseau (topologie, taille, dynamique...)
- de la qualité des exemples utilisés pour l'apprentissage,
il faut notamment une bonne
distribution des exemples

Exemple d'un mauvais
apprentissage typique de
l'algorithme classique du
perceptron



Le danger du sur-apprentissage

Dans une base d'apprentissage « réelle », on a souvent :

- un **bruit** important
- quelques **exemples non représentatifs**
- éventuellement des **erreurs** d'étiquetage...

⇒ pour certains problèmes,
il est **inutile de chercher à apprendre tous les exemples**

⇒ arrêt de l'apprentissage une fois un certain pourcentage
de la base d'exemples apprise (« taux d'apprentissage »)

⇒ évite un « sur-apprentissage »

= *en apprenant trop « par cœur » on généralise moins bien*

The credit assignment problem

On a vu que l'on peut synthétiser toute fonction booléenne en utilisant plusieurs couches de perceptron

mais on n'a pas la « sortie désirée » des couches non terminales
⇒ comment apprendre ?

The credit assignment problem

On a vu que l'on peut synthétiser toute fonction booléenne en utilisant plusieurs couches de perceptron

mais on n'a pas la « sortie désirée » des couches non terminales
⇒ comment apprendre ?
= « *credit assignment problem* »

Problème mis en évidence par Minsky et Papert en 1969 et resté ouvert 20 ans !

The credit assignment problem

On a vu que l'on peut synthétiser toute fonction booléenne en utilisant plusieurs couches de perceptron

mais on n'a pas la « sortie désirée » des couches non terminales
⇒ comment apprendre ?
= « *credit assignment problem* »

Problème mis en évidence par Minsky et Papert en 1969 et resté ouvert 20 ans !

Solution : algorithme de la rétro-propagation du gradient
.... on va voir plus tard

Résumé chronologique des débuts du connexionnisme

- **Années 40** : Van Neumann, Mc Culloch, Turing...
Fondement d'une science des systèmes capables
d'auto-organisation (la cybernétique)

Résumé chronologique des débuts du connexionnisme

- **Années 40** : Van Neumann, Mc Culloch, Turing. . .
Fondement d'une science des systèmes capables
d'auto-organisation (la cybernétique)
- **Années 60** : Rosenblatt reprend les automates de Mc Culloch
et Pitts pour faire de la reconnaissance de formes

Résumé chronologique des débuts du connexionnisme

- **Années 40** : Van Neumann, Mc Culloch, Turing. . .
Fondement d'une science des systèmes capables d'auto-organisation (la cybernétique)
- **Années 60** : Rosenblatt reprend les automates de Mc Culloch et Pitts pour faire de la reconnaissance de formes
- **Années 70** : Minsky et Papert mettent en évidence les limites du perceptron ⇒ années de l'IA symbolique (gros travaux)
On veut simuler directement le raisonnement grâce à de la logique / des inférences (systèmes experts, logique floue)
Problème : explosion combinatoire. . .

Résumé chronologique des débuts du connexionnisme

- **Années 40** : Van Neumann, Mc Culloch, Turing. . .
Fondement d'une science des systèmes capables
d'auto-organisation (la cybernétique)
- **Années 60** : Rosenblatt reprend les automates de Mc Culloch
et Pitts pour faire de la reconnaissance de formes
- **Années 70** : Minsky et Papert mettent en évidence les limites
du perceptron ⇒ années de l'IA symbolique (gros travaux)
On veut simuler directement le raisonnement grâce à de la
logique / des inférences (systèmes experts, logique floue)
Problème : explosion combinatoire. . .
- **1986** : algorithme de la rétro-propagation du gradient !

1 Prologue...

2 **Introduction**

- Intelligence Artificielle symbolique et numérique
- Le neurone formel et le perceptron
- Apprentissage et Généralisation
- Topologie des réseaux de neurones
- Les modèles à base de prototypes

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Les trois grandes familles

La capacité du réseau à résoudre le problème va dépendre de :

Les trois grandes familles

La capacité du réseau à résoudre le problème va dépendre de :

- l'algorithme d'apprentissage...
- des exemples à apprendre
- de l'architecture du réseau

Les trois grandes familles

La capacité du réseau à résoudre le problème va dépendre de :

- l'algorithme d'apprentissage...
- des exemples à apprendre
- de **l'architecture du réseau** :
 - **topologie** du réseau
 - **dynamique** du réseau (propagation de l'influx)
 - **taille** du réseau

Les trois grandes familles

La capacité du réseau à résoudre le problème va dépendre de :

- l'algorithme d'apprentissage...
- des exemples à apprendre
- de **l'architecture du réseau** :
 - **topologie** du réseau
 - **dynamique** du réseau (propagation de l'influx)
 - **taille** du réseau

Il existe 3 grandes familles de réseaux :

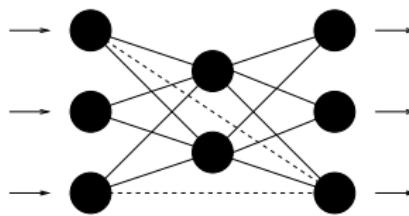
- les réseaux à couches
- les réseaux à connexions latérales
- les réseaux récurrents

Les réseaux à couches

Définition : réseau à couches = « graphe acyclique de cellules »

On peut définir une ou plusieurs :

- cellules d'entrée
- cellules de sortie



Remarques :

- les connexions sont orientées (entrée → sortie)
- le nombre de couches = nombre de cellules entre les cellules d'entrée et les cellules de sortie
- de loin les réseaux les plus utilisés !

Propagation de l'influx dans un réseau à couches

La propagation dans un réseau à couches est dite

« *feed forward* » = « alimenter et faire suivre »

Propagation de l'influx dans un réseau à couches

La propagation dans un réseau à couches est dite

« *feed forward* » = « alimenter et faire suivre »

Dynamique du réseau :

- ① l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= les cellules de la première couche (entrée) sont « forcées »

Propagation de l'influx dans un réseau à couches

La propagation dans un réseau à couches est dite

« feed forward » = « alimenter et faire suivre »

Dynamique du réseau :

- ① l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= les cellules de la première couche (entrée) sont « forcées »
- ② **l'influx se propage de couche en couche** (entrée → sortie)
= on active les couches les unes après les autres

Propagation de l'influx dans un réseau à couches

La propagation dans un réseau à couches est dite

« feed forward » = « alimenter et faire suivre »

Dynamique du réseau :

- ① l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= les cellules de la première couche (entrée) sont « forcées »
- ② **l'influx se propage de couche en couche** (entrée → sortie)
= on active les couches les unes après les autres
- ③ la réponse du réseau peut être lue sur la dernière couche

Propagation de l'influx dans un réseau à couches

La propagation dans un réseau à couches est dite

« feed forward » = « alimenter et faire suivre »

Dynamique du réseau :

- ① l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= les cellules de la première couche (entrée) sont « forcées »
- ② **l'influx se propage de couche en couche** (entrée → sortie)
= on active les couches les unes après les autres
- ③ la réponse du réseau peut être lue sur la dernière couche

Remarque : le temps de réponse est constant et connu

⇒ **temps réel** possible

Les réseaux à connexions latérales

Réseau à connexions latérales =

réseau à couches + **connexions entre cellules d'une même couche**

Les réseaux à connexions latérales

Réseau à connexions latérales =
réseau à couches + **connexions entre cellules d'une même couche**

Dynamique, pour chaque couche de l'entrée vers la sortie :

- ① activation en fonction de la couche précédente,
via les connexions inter-couches exclusivement

Les réseaux à connexions latérales

Réseau à connexions latérales =

réseau à couches + **connexions entre cellules d'une même couche**

Dynamique, pour chaque couche de l'entrée vers la sortie :

- ① activation en fonction de la couche précédente,
via les connexions inter-couches exclusivement
- ② activation en fonction du nouvel état de la couche courante,
via les connexions intra-couche exclusivement

Les réseaux à connexions latérales

Réseau à connexions latérales =
réseau à couches + **connexions entre cellules d'une même couche**

Dynamique, pour chaque couche de l'entrée vers la sortie :

- ① activation en fonction de la couche précédente,
via les connexions inter-couches exclusivement
- ② activation en fonction du nouvel état de la couche courante,
via les connexions intra-couche exclusivement

Remarques :

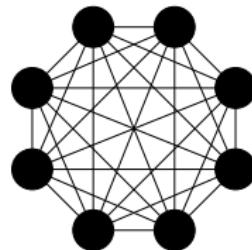
- généralement peu de couches
- généralement pas de connexions latérales sur chaque couche

Exemple : les cartes de Kohonen (on verra plus tard...)

Les réseaux récurrents

Définition : réseau récurrents =

« **graphe avec une ou plusieurs boucles** »



Particularités :

- partiellement ou totalement interconnecté
- connexions orientées, éventuellement dans les 2 sens

Remarques :

- L'état d'activation de l'ensemble des cellules à un instant t forme une « **configuration d'états** »
- Il existe des configurations d'états dites « **stables** »

Propagation de l'influx dans un réseau récurrent

« *Feedback neural networks* » = « alimenté en retour »

Propagation de l'influx dans un réseau récurrent

« *Feedback neural networks* » = « alimenté en retour »

Dynamique du réseau :

- ➊ l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= certaines (voire toutes) les cellules sont « forcées »

Propagation de l'influx dans un réseau récurrent

« Feedback neural networks » = « alimenté en retour »

Dynamique du réseau :

- ① l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= certaines (voire toutes) les cellules sont « forcées »
- ② pour chaque cellule :
l'influx se propage d'une cellule à l'autre
= on active toutes les cellules
⇒ forme une nouvelle « configuration d'états »

Propagation de l'influx dans un réseau récurrent

« *Feedback neural networks* » = « alimenté en retour »

Dynamique du réseau :

- ① l'exemple à apprendre (en apprentissage) ou la forme à reconnaître (en généralisation) est présentée au réseau
= certaines (voire toutes) les cellules sont « forcées »
- ② pour chaque cellule :
l'influx se propage d'une cellule à l'autre
= on active toutes les cellules
⇒ forme une nouvelle « configuration d'états »
- ③ si une configuration stable est atteinte, la réponse du réseau peut être lue sur certaines (voire toutes) les cellules

Ordre d'activation des cellules

En « *feedforward* » l'ordre d'activation n'a pas d'importance : il n'a pas d'influence sur le comportement / la sortie du réseau

Ordre d'activation des cellules

En « *feedforward* » l'ordre d'activation n'a pas d'importance : il n'a pas d'influence sur le comportement / la sortie du réseau **mais** ce n'est pas le cas avec une dynamique « *feedback* » !

Important : dans un réseau récurrent l'ordre d'activation des cellules influe sur l'évolution des configurations d'états du réseau, et donc sur l'éventuelle réponse du réseau

Ordre d'activation des cellules

En « *feedforward* » l'ordre d'activation n'a pas d'importance : il n'a pas d'influence sur le comportement / la sortie du réseau **mais** ce n'est pas le cas avec une dynamique « *feedback* » !

Important : dans un réseau récurrent l'ordre d'activation des cellules influe sur l'évolution des configurations d'états du réseau, et donc sur l'éventuelle réponse du réseau

L'activation peut être :

- « *synchrone* » : toutes les cellules s'activent « simultanément » sur la base de l'état d'activation précédent
- « *asynchrone* » : les cellules s'activent les unes après les autres, dans un ordre donné (fixé) ou aléatoire, sur la base de l'état d'activation courant

Lecture de la réponse (la sortie) d'un réseau récurrent

Dans un réseau à couches : la réponse est lue sur la dernière couche
après un temps constant et connu

Lecture de la réponse (la sortie) d'un réseau récurrent

Dans un réseau à couches : la réponse est lue sur la dernière couche
après un temps constant et connu

Dans un réseau récurrent :

- le réseau peut converger vers une configuration d'états stable
= propager de nouveau l'influx ne change pas la configuration
⇒ **la réponse du réseau est la configuration stable trouvée !**
- le réseau peut converger vers un cycle, voire ne pas converger
⇒ pas de réponse

Lecture de la réponse (la sortie) d'un réseau récurrent

Dans un réseau à couches : la réponse est lue sur la dernière couche
après un temps constant et connu

Dans un réseau récurrent :

- le réseau peut converger vers une configuration d'états stable
= propager de nouveau l'influx ne change pas la configuration
⇒ **la réponse du réseau est la configuration stable trouvée !**
- le réseau peut converger vers un cycle, voire ne pas converger
⇒ pas de réponse

Remarque : le temps de réponse est variable (stabilisation)
⇒ temps réel délicat à mettre en œuvre

Principe de l'apprentissage d'un réseau récurrent

Principe : **on utilise les états stables du réseau récurrent pour mémoriser les exemples à apprendre**

Principe de l'apprentissage d'un réseau récurrent

Principe : **on utilise les états stables du réseau récurrent pour mémoriser les exemples** à apprendre

⇒ on modifie les poids / les seuils
afin de converger sur une forme apprise à partir d'une forme :

- incomplète
- complète mais bruitée
- incomplète et bruitée

Principe de l'apprentissage d'un réseau récurrent

Principe : **on utilise les états stables du réseau récurrent pour mémoriser les exemples** à apprendre

⇒ on modifie les poids / les seuils
afin de converger sur une forme apprise à partir d'une forme :

- incomplète
- complète mais bruitée
- incomplète et bruitée

Exemple de réseau récurrent : les réseaux de Hopfield
(on verra plus tard, très bientôt ;-)

1 Prologue...

2 **Introduction**

- Intelligence Artificielle symbolique et numérique
- Le neurone formel et le perceptron
- Apprentissage et Généralisation
- Topologie des réseaux de neurones
- Les modèles à base de prototypes

3 Les modèles classiques du connexionnisme

4 Réseaux neuronaux convolutifs et apprentissage profond

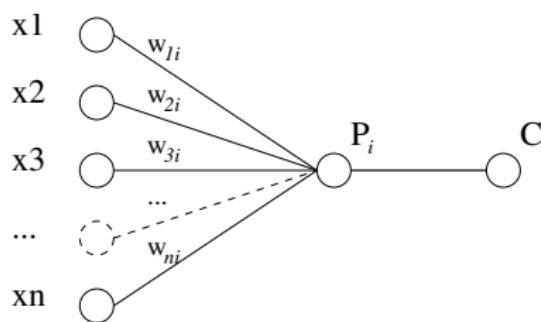
5 Algorithmes génétiques

Les modèles à base de prototypes

Prototype = cellule dont les connexions entrantes, pondérées par des réels, stockent un **exemple représentatif d'une classe**

La fonction d'activation évalue la ressemblance entre :

- un exemple en entrée
 $X = (x_1 x_2 \dots x_n)$
- le prototype
 $P_i = (w_{1i} w_{2i} \dots w_{ni})$



Le choix de la mesure est lié à la nature de l'espace d'entrée

Hypersphères contre hyperplans

Un prototype s'active si l'entrée lui ressemble

= si l'exemple est dans la zone d'influence définie par ses poids

⇒ le prototype sépare l'espace en deux sous espaces :

- à l'intérieur de l'hypersphère
- à l'extérieur de l'hypersphère

Hypersphères contre hyperplans

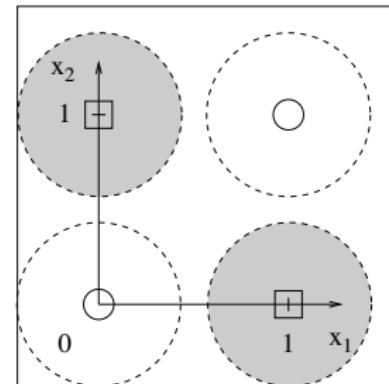
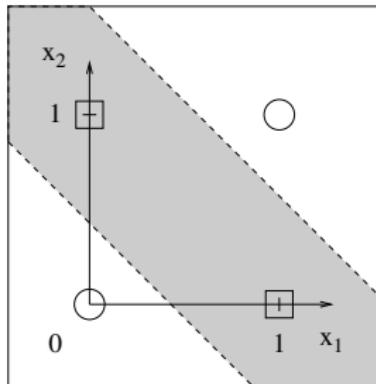
Un prototype s'active si l'entrée lui ressemble

= si l'exemple est dans la zone d'influence définie par ses poids

⇒ le prototype sépare l'espace en deux sous espaces :

- à l'intérieur de l'hypersphère
- à l'extérieur de l'hypersphère

Exemple sur le
problème du
XOR :



Prototypes et localité

Dans un réseau à base de prototypes,
la « connaissance » stockée est :

- localisée
- cohérente avec l'entrée

Prototypes et localité

Dans un réseau à base de prototypes,
la « connaissance » stockée est :

- localisée → facile à retrouver
- cohérente avec l'entrée → facile à « visualiser »

Prototypes et localité

Dans un réseau à base de prototypes,
la « connaissance » stockée est :

- localisée → facile à retrouver
- cohérente avec l'entrée → facile à « visualiser »

⇒ on peut analyser la connaissance du réseau
= pouvoir explicatif (limité)

Apprentissage pour un réseau à base de prototypes

Pour apprendre, on peut :

- **changer les poids**
- **changer les seuils**
- **ajouter** des prototypes, on parle de « réseau incrémental »
- **supprimer** des prototypes, on parle « d'élagage »

Apprentissage pour un réseau à base de prototypes

Pour apprendre, on peut :

- **changer les poids** = déplacer le centre de l'hypersphère
- **changer les seuils** = changer le rayon de l'hypersphère
- **ajouter** des prototypes, on parle de « réseau incrémental »
- **supprimer** des prototypes, on parle « d'élagage »

Apprentissage pour un réseau à base de prototypes

Pour apprendre, on peut :

- **changer les poids** = déplacer le centre de l'hypersphère
- **changer les seuils** = changer le rayon de l'hypersphère
- **ajouter** des prototypes, on parle de « réseau incrémental »
- **supprimer** des prototypes, on parle « d'élagage »

Les principaux modèles à base de prototypes :

- les cartes de Kohonen (1982-84)
- les algorithmes LVQ de Kohonen (1988-90)
- le néocognitron de Fukushima (1980)
- le modèle ART de Carpenter et Grossberg (1987-88)

1 Prologue...

2 Introduction

3 Les modèles classiques du connexionnisme

- Réseaux de Hopfield
- Les réseaux LVQ et les cartes de Kohonen
- Machine à vecteurs de support
- Le Perceptron multi-couches et la rétro-propagation du gradient
- Le Perceptron multi-couches, de la théorie à la pratique !

4 Réseaux neuronaux convolutifs et apprentissage profond

5 Algorithmes génétiques

Le modèle de réseau récurrent de Hopfield (1982 / 1984)

Le modèle de Hopfield :

- réseau récurrent
- totalement interconnecté
- connexions symétriques : $w_{ij} = w_{ji}$
- pas « d'auto-connexions » : $w_{ii} = 0$
- activation asynchrone
 - on ne change qu'une cellule à la fois
 - a permis à Hopfield de prouver la convergence
- les points peuvent être appris ou calculés

Inspiration : Hebb (biologiste) → « **il faut renforcer les poids des cellules activées simultanément** »

Calcul des poids d'un réseau Hopfield

Apprentissage selon la **règle de Hebb** $\Rightarrow w_{ij} = w_{ij} + \mu y_i y_j$ avec :

- μ : paramètre déterminant l'intensité de l'apprentissage
- y_i : sortie de la cellule $i = \ll \text{état} \gg$ de la cellule i

Calcul des poids :

- pour P exemples binaires $(-1, 1)$ X^1, X^2, \dots, X^P :

$$\begin{cases} w_{ij} = \sum_{p=1}^P x_i^p x_j^p \text{ pour } i \neq j \\ w_{ii} = 0 \text{ pour } i = j \end{cases}$$

- pour P exemples binaires plus classique, dans $(0, 1)$:

$$\begin{cases} w_{ij} = \sum_{p=1}^P [2x_i^p - 1][2x_j^p - 1] \text{ pour } i \neq j \\ w_{ii} = 0 \text{ pour } i = j \end{cases}$$

Apprentissage des poids d'un réseau Hopfield

On obtient les mêmes poids que par le calcul,
avec l'**algorithme d'apprentissage** suivant :

- ① démarrer avec des connexions nulles
- ② présenter un exemple X^P : $y_i = x_i^P$ pour $i = 1 \rightarrow n$
- ③ modifier les connexions :

$$\begin{cases} +1 & \text{si les cellules sont actives ou inactives simultanément} \\ -1 & \text{sinon} \end{cases}$$

- ④ boucler en 2,
jusqu'à ce que les P exemples aient été présentés

Généralisation dans un réseau Hopfield

Algorithme de généralisation :

- ① on présente l'exemple X : $y_i = x_i$ pour $i = 1 \rightarrow n$
- ② tant que le réseau n'a pas convergé,
pour chaque cellule (dans un ordre aléatoire) :

on active la cellule : $y'_i = \sum_{j=1}^n y_j w_{ji}$

on détermine le nouvel état de la cellule :

$$\begin{cases} 1 & \text{si } y' > \theta_i \\ y_i & \text{si } y' = \theta_i \\ 0 & \text{si } y' < \theta_i \end{cases}$$

Remarques :

- on a convergé si les états (y_i) ne changent plus
- on prend généralement tous les $\theta_i = 0$

Borne sur la capacité de stockage d'un réseau Hopfield

Quel est le **nombre de formes que l'on peut stocker** et reconnaître avec une « fréquence raisonnable » avec un réseau de Hopfield de n cellules (avec l'algorithme d'apprentissage vu) ?

Selon des études expérimentales de Hopfield :

$$P_{max} = 0,15n$$

Borne théorique en 1987 :

$$P_{max} \equiv \frac{n}{2 \log_2 n}$$