

# Programmation d'une Intelligence Artificielle : arbres ou apprentissage automatique

Paul Mangold

L3 MIASHS

22 Janvier 2021

## Fonctionnement du Cours

### Déroulement du cours :

- 12 Cours/TP de 2h ;
- a priori en visio pour encore un moment...

### Déroulement des TP :

- un peu de temps pour les questions sur le cours d'avant (5min) ;
- petit cours avec slides (20-30min max) ;
- TP/TD sur machine :
  - souvent implémentation d'algorithmes et expérimentation ;
  - parfois un peu de théorie sur papier (pas de panique !).

Langage et outils :

- Python ;
- principalement des notebooks.

Librairies :

- `numpy`: calcul matriciel...
- `pandas`: dataframes, csv...
- `networkx`: dessiner des graphes !
- `sklearn`: machine learning.

### 1ère partie : Graphes.

- graphes, arbres ;
- problèmes sur les graphes ;
- application des graphes à l'IA ;
- algèbre linéaire sur les graphes.

### 2ème partie : Machine Learning

- ML supervisé à base d'arbres ;
- apprentissage semi-supervisé ;
- apprentissage non-supervisé ;
- chaînes de Markov et PageRank.

De manière indicative (ça sera adapté un peu) :

- TP : certains TPs seront ramassés et notés ( $\sim 20\%$  de la note);
- DM : quelques DM facultatifs pour approfondir (bonus) ;
- Projet : un mini-projet et un projet de machine learning ( $\sim 30\%$  de la note) ;
- Exam : une interro à mi-semester et un exam final ( $\sim 10\%$  et  $40\%$  de la note).

## Petite histoire de l'intelligence artificielle

# Petite histoire de l'intelligence artificielle

## Le terme "Intelligence Artificielle"

- 1950 – Turing pose la question : “Can Machine Think?”  
→ test de Turing.
- 1956 – conférence de Dartmouth :  
“chaque aspect de l'apprentissage ou toute autre caractéristique de l'intelligence peut être si précisément décrit qu'une machine peut être conçue pour le simuler.”  
→ création du terme “intelligence artificielle”.



- 1943 – McCullough et Pitts : modèle de neurone  
→ inspiration biologique : reçoit des stimuli (0/1) en entrée et s'active ou non.
- 1951 – Minsky : première machine à réseau neuronal.
- 1958 – Rosenblatt : perceptron : idée d'apprendre les poids.
- 1960 – Widrow-Hoff : descente de gradient pour apprendre les poids.

Des belles avancées :

- 1950-1960 – premiers programmes pour jouer aux échecs et aux dames.  
→ l'IA pour les jeux sert d'étalon pour l'IA.
- 1955 – Newell et Simon : “Théoricien Logique” qui peut démontrer des théorèmes.

À ce moment là, on pense pouvoir faire des machines intelligences dans les 10 prochaines années...

- 1965 – Simon : “d’ici vingt ans les machines pourront faire tout ce que l’humaine peut faire.”
- 1970 – Minsky : “dans trois à huit ans nous aurons une machine avec l’intelligence générale d’un être humain ordinaire.”

- 1969 – Minsky et Papert : le perceptron ne permet pas de faire un XOR.
- 1971 – Cook : “The Complexity of Theorem-Proving Procedures”  
→ il existe des problèmes très difficiles (SAT est NP-complet), théorie de la complexité.
- Année 1980 – paradoxe de Moravec :  
→ raisonnement (difficile pour l'humain) se fait pas trop mal avec une machine ;  
→ reconnaissance de visages, déplacements (faciles pour l'humain) sont difficiles avec des machines.

Idées de réseaux de neurones :

- Année 1980 : une tâche complexe est le résultat de la composition de tâches simples.
- 1986–1987 – Rumelhart-Lecun : rétropropagation du gradient.

Progression de la puissance de calcul :

- 1965 – Loi de Moore : la puissance des ordinateurs double tous les deux ans.
- 1997 – Deep Blue bat Kasparov aux échecs.

→ Années 2000 – machines puissantes permettent des réseaux de neurones plus grand, avec des méthodes inspirées par les statistiques, les probabilités, l'optimisation...

On peut réécrire le contenu de ce cours :

1ère partie : IA “à l'ancienne” basée sur l'algorithmique, et les graphes.

- calcul d'itinéraires ;
- IA pour des jeux (morpions, sudoku, dames, échecs...) ;
- ...

2ème partie : IA “moderne” basée sur l'apprentissage statistique.

- apprendre à classer des objets ;
- prédire des résultats ;
- rassembler des objets similaires ;
- classer des pages web ;
- ...

# Good Old-Fashioned Artificial Intelligence (GOFAI)

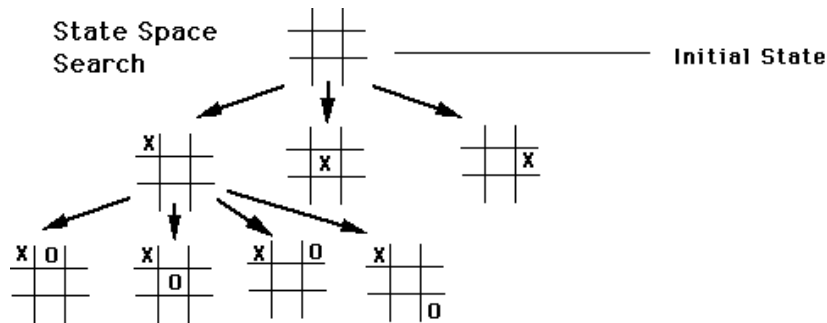
Ce sont des problèmes d'algorithmique :

- prouver un théorème ;
- jouer au morpion, aux échecs, aux dames ;
- trouver son chemin dans un labyrinthe ;
- résoudre un Sudoku ;
- trouver un itinéraire optimal pour la factorielle ;
- intelligence artificielle dans un jeu vidéo...

# Good Old-Fashioned Artificial Intelligence (GOFAI)

## Représentation d'un problème : des arbres

Coups possibles au Morpion :



Chaque flèche représente un coup joué par un joueur.

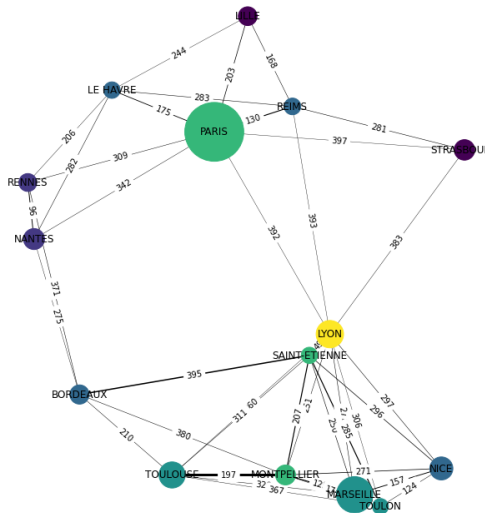
→ on associe un score à chaque flèche puis on choisit la meilleure.



# Good Old-Fashioned Artificial Intelligence (GOFAI)

## Représentation d'un problème : des graphes

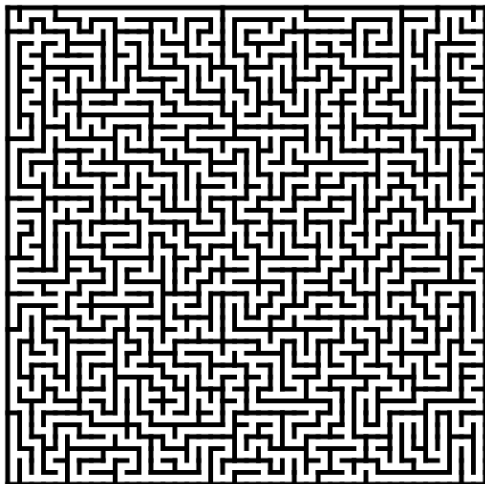
Trouver un itinéraire (parcours de graphe) :



# Good Old-Fashioned Artificial Intelligence (GOFAI)

## Représentation d'un problème : des graphes

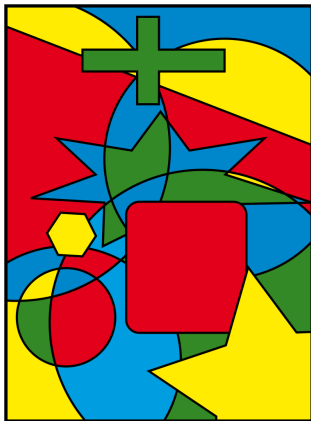
Plus compliqué... ou bien ?



# Good Old-Fashioned Artificial Intelligence (GOFAI)

## Représentation d'un problème : des graphes

Coloriage de graphes (éviter que deux couleurs se touchent).



# Good Old-Fashioned Artificial Intelligence (GOFAI)

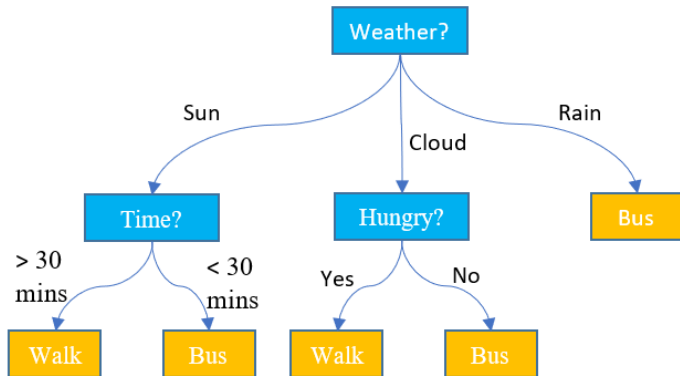
Représentation d'un problème : des graphes

Coloriage de graphes : résolution de Sudoku ?

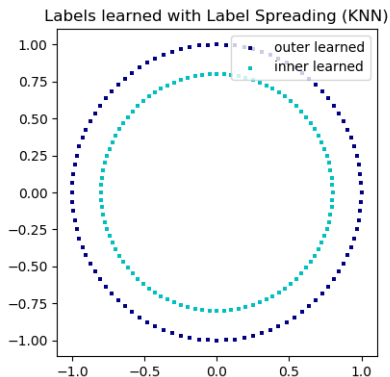
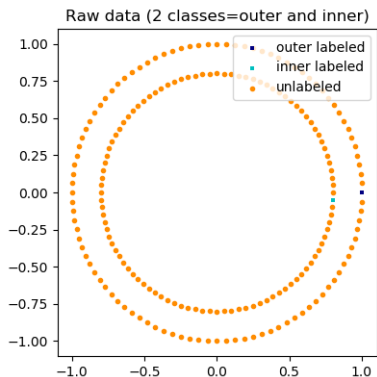
|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 5 | 3 |   |   | 7 |   |   |   |   |
| 6 |   |   | 1 | 9 | 5 |   |   |   |
|   | 9 | 8 |   |   |   |   | 6 |   |
| 8 |   |   |   | 6 |   |   |   | 3 |
| 4 |   |   | 8 |   | 3 |   |   | 1 |
| 7 |   |   |   | 2 |   |   |   | 6 |
|   | 6 |   |   |   |   | 2 | 8 |   |
|   |   |   | 4 | 1 | 9 |   |   | 5 |
|   |   |   |   | 8 |   |   | 7 | 9 |

# Intelligence Artificielle Moderne : Machine Learning

Répondre à une question oui/non :

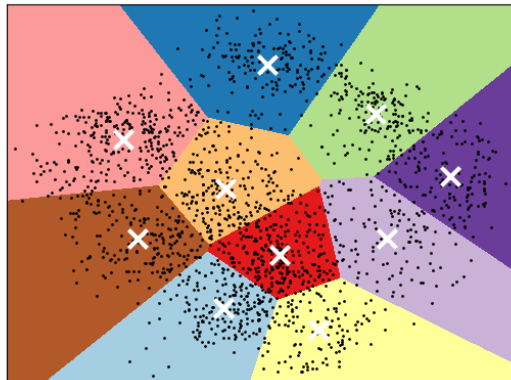


Que faire si on a peu de données avec labels ?



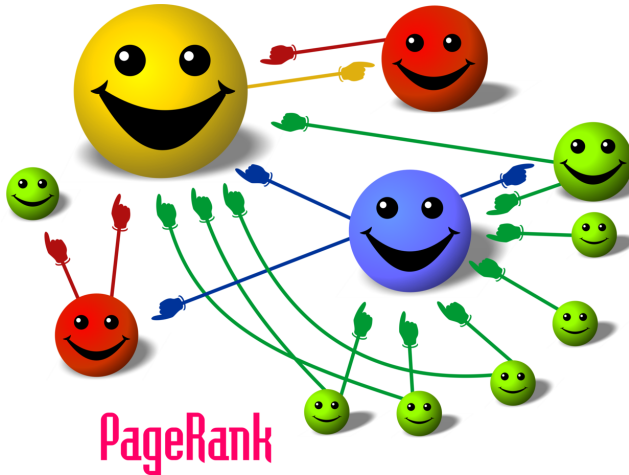
Détecter des similarités ? Trouver des communautés dans un graphe ?

K-means clustering on the digits dataset (PCA-reduced data)  
Centroids are marked with white cross





Donner un score à une page Web ?



TP !

Pour le premier TP, on va regarder

- ce que c'est qu'un graphe ;
- un peu de vocabulaire sur les graphes ;
- comment afficher des graphes en Python ;
- quelques chemins dans les graphes.