

# Sujet 6 : Une intelligence artificielle peut-elle reconnaître un graphe ?

## Problématique

Les mathématiciens classent les graphes selon leurs propriétés : en ligne, en étoile, en cycle, complet, ou encore arborescent. Mais si on montrait ces graphes à une machine, serait-elle capable de les reconnaître seule, en apprenant à partir d'exemples ?

L'idée est de construire un petit modèle d'**intelligence artificielle (IA)** qui apprend à distinguer différents types de graphes. Par exemple : reconnaître si un graphe est un arbre, un cycle ou un graphe complet. L'apprentissage se fera à partir de données simples (nombre de sommets, nombre d'arêtes, degré moyen, connexité...).

**Question centrale :** Peut-on apprendre à une machine à reconnaître la nature d'un graphe à partir de ses caractéristiques mathématiques ?

## Questions de recherche

### 1. Décrire les familles de graphes.

Quels sont les critères qui différencient un graphe en ligne, en étoile, un cycle ou un graphe complet ? Peut-on les caractériser par des mesures numériques (nombre d'arêtes, degré moyen, connexité, symétrie) ?

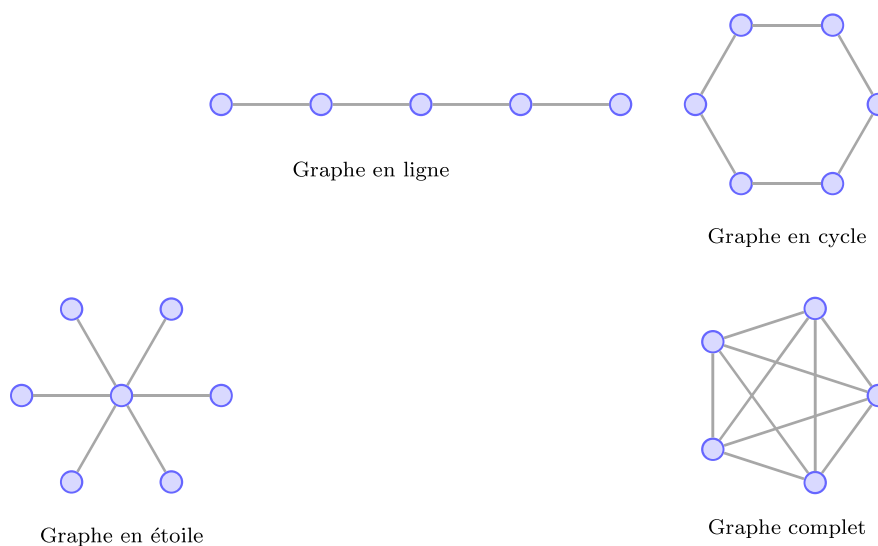


Figure 6: Exemples de familles de graphes que l'IA peut apprendre à reconnaître : ligne, cycle, étoile et complet.

### 2. Construire un jeu de données.

Peut-on créer une liste d'exemples de graphes avec leurs caractéristiques ? (Par exemple, chaque ligne contient : *nombre de sommets, nombre d'arêtes, degré maximal, type du graphe.*)

### 3. Choisir un modèle d'apprentissage.

Quelle méthode simple utiliser pour apprendre à classer ces graphes ? (Exemples possibles : k-plus proches voisins, régression logistique, perceptron simple, arbre de décision.)

#### 4. **Tester et évaluer.**

Si on présente un nouveau graphe inconnu, l'IA peut-elle prédire correctement sa catégorie ? Comment mesurer ses réussites et ses erreurs ?

#### 5. **Analyser les limites et extensions.**

Le modèle apprend-il vraiment ou se contente-t-il d'imiter les exemples ? Peut-on améliorer la reconnaissance en ajoutant d'autres caractéristiques (diamètre, densité, degré moyen pondéré, etc.) ?

### **Notions mathématiques mobilisées**

- Théorie des graphes : sommets, arêtes, degré, connexité, arbres, cycles, graphes complets ;
- Statistiques et apprentissage supervisé : classification, distance, taux d'erreur ;
- Algèbre et géométrie : représentation des données dans un plan ou un espace à plusieurs dimensions ;
- Programmation Python : bibliothèque `networkx` pour générer les graphes, `scikit-learn` pour l'apprentissage.

### **Références et pistes**

- [NetworkX — Python Graph Library](#) ;
- [Scikit-learn — Machine Learning in Python](#) ;
- Vidéo : *3Blue1Brown* — *What is a Neural Network?* (YouTube).