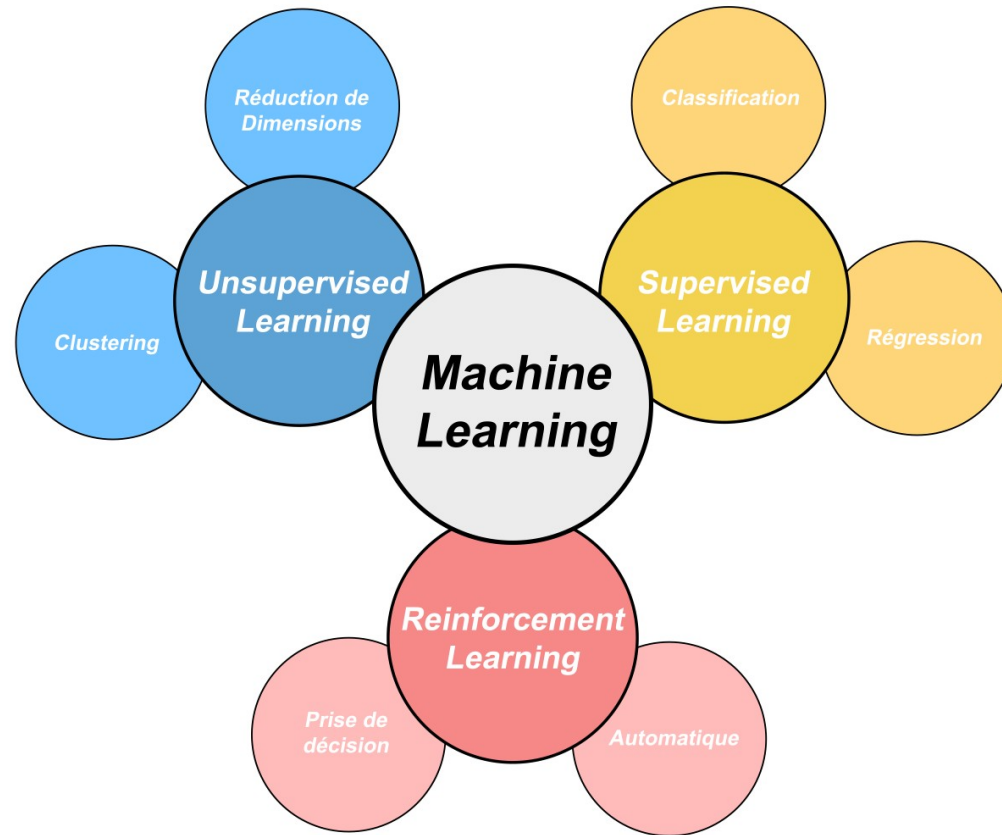


# **Machine Learning (ML) appliqué à la Reconnaissance d'Activités Humaines (RAH)**

Nombreux domaines qui utilisent la RAH : jeux vidéos, systèmes informatiques assistant l'Homme dans ses tâches (ex: médecine), équipement sportif (ex: chaussures intelligentes, podomètre ...)

# Généralités



## 3 grands types de ML :

- Apprentissage par renforcement :  
Permet d'adapter le comportement de la machine en fonction des récompenses données (ou non) par l'environnement, ex : *Machine qui joue aux échecs*

- Apprentissage non supervisé :  
Permet de trouver des modèles récurrents dans de grands ensembles de données, ex : *techniques de clustering, comme les HMM*

- Apprentissage supervisé :  
Permet de faire des classifications ou des régressions à partir d'un ensemble de données

# Apprentissage supervisé

- Régression :

On déduit d'un ensemble de classes et de variables la prochaine variable d'une classe.

- Classification :

On déduit d'un ensemble de classes et de variables la classe d'une variable encore non classée.

**→ La RAH repose sur cette partie du ML puisque l'on cherche à classer un ensemble de variables pour représenter les comportements. On utilise généralement les données d'un accéléromètre ou plus généralement d'un gyroscope comme variables.**

# Techniques courantes de classification en apprentissage supervisé

- **SVM** (Support Vector Machine)
  - > **OVR** (One-VS-Rest)
  - > **OVO** (One-VS-One)
- **KNN** (K-Nearest Neighbors)
- **Decision Tree**
- **Naive Bayes**

# SVM

## Classification binaire sur 2 principes :

- Kernel approximation  
Dans le cadre de certaines hypothèses, possible de prendre un ensemble de données dans un espace de dimension  $N$  et de l'augmenter avec une fonction discriminatoire linéaire qui permet de séparer linéairement les variables avec un hyperplan dans l'espace de dimension  $N+1$ .
- Marge maximale  
On optimise l'hyperplan séparant les variables en maximisant la distance entre le plan et les points les plus proches.

## Élargissement à une classification d'arité $>2$ :

- OVR :  
On applique la SVM en séparant à chaque itération une nouvelle classe de variables du reste des variables, jusqu'à séparation totale.
- OVO :  
On applique la SVM en séparant toutes les classes 2 à 2 jusqu'à séparation totale des classes.

→ La méthode est fondée sur des bases mathématiques solides, donc il y a une logique à cette méthode. De plus, la phase de learning de cette méthode est rapide.

## • KNN

À partir d'un ensemble de  $N$  variables appartenant à  $N$  classes, on choisit pour chaque nouvelle variable sa classe en fonction de la classe de ses  $K$  plus proches voisins (par rapport à une fonction distance à définir).

On regarde parmi les  $K$  variables quelle classe apparaît le plus souvent, que l'on peut pondérer par la distance entre la variable et le voisin.

→ *Méthode simple mais qui dépend beaucoup du choix de  $K$  et dont les calculs peuvent devenir très coûteux pour un gros dataset*

## • Decision Tree

À partir d'un ensemble de  $N$  points appartenant à  $N$  classes, on cherche un arbre binaire où chaque nœud est un test et chaque feuille une classe afin de déterminer pour chaque nouvelle variable sa classe en lui faisant parcourir l'arbre.

Il existe plusieurs séquences de tests à choisir selon leur capacité à séparer correctement les variables.

→ *Méthode efficace sur de gros datasets, assez simple à mettre en place, mais risque de sur-apprentissage pour un ensemble trop compliqué qui casse les bénéfices de cette méthode.*

## • Naive Bayes

Les naïves de Bayes sont un ensemble d'algorithmes qui repose sur l'indépendance des composantes de chaque variable en terme de probabilité. À chaque classe ces algorithmes associent une probabilité liée à une composante.

Pour chaque variable, on choisit la classe qui associe la plus forte probabilité à ses composantes (à une pondération près, si nécessaire)

→ *Méthode efficace dans le cadre de l'indépendance des coordonnées de variables, ce qui n'est pas souvent le cas en RAH (ex : balancement de bras)*