I – Apprentissage machine

1. Un peu de vocabulaire

Le Machine Learning ou apprentissage automatique est une sous-catégorie de l'intelligence artificielle. C'est la science qui consiste à faire en sorte que les machines apprennent et agissent de manière identique aux humains, en apprenant de manière autonome à partir des interactions du monde réel et des ensembles de données d'enseignement que nous leur fournissons. Ces données peuvent être des chiffres, des mots, des images, etc.

La démarche de machine learning se déroule en deux étapes :

1. Une phase d'**apprentissage** : on fournit des données à la machine sur lesquelles on applique des algorithmes pour "apprendre" de l'information.
2. Une phase de **prédiction** : la machine peut alors faire des prédictions sur des jeux de données nouvelles et inconnues.

Au sein de l'apprentissage machine, on peut distinguer :

- l'**apprentissage supervisé** : les données d'entraînement contiennent des **échantillons de données associés à des catégories**.
  Les catégories peuvent être
  - discrètes (ex : blanc/noir, [0, 1, 2], ...) ; associées à des algorithmes de **classification**.
  - continues (ex : nombre réel entre 0 et 1) ; associées à des algorithmes de **régression**.
- l'**apprentissage non supervisé** : les données d'entraînement contiennent des échantillons de données brutes, sans étiquetage particulier. Les algorithmes associés permettent alors par exemple d'**identifier des ensembles (clustering)** au sein des données.

2. Démarche générale

Pendant la phase d'apprentissage, il faut pouvoir **collecter de grandes quantités de données** et les soumettre à l'algorithme pour analyse.

Ces données doivent être séparées en un **jeu d'entraînement** et un **jeu de test** pour valider le modèle obtenu.

Il est assez fréquent de devoir appliquer un **prétraitement** aux données pour les mettre en forme pour que l'algorithme mis en œuvre soit plus efficace.

Après avoir entraîné la machine, **on évalue son efficacité sur le jeu de données tests**.

Les algorithmes mis en jeu nécessitent généralement un **paramétrage** (définition de quelques valeurs spécifiques). Pendant la phase d'apprentissage, on peut chercher à ajuster ces paramètres pour obtenir les meilleurs scores de prédiction sur le jeu de données tests.

Ensuite, il n'y a plus qu'à appliquer l'algorithme ajusté sur des données inconnues pour obtenir des **prédictions** nouvelles.

II – Exemple d'algorithme d'apprentissage supervisé de classification

1. Principe de l'algorithme des k plus proches voisins

Nous allons voir un exemple classique, assez simple, d'algorithme d'apprentissage machine.

L'**algorithme des k plus proches voisins** (ou **knn** pour k nearest neighbours) est une **méthode d'apprentissage supervisé**. Il permet de faire une **classification** de données.

On peut dire que cet algorithme repose sur le dicton populaire « qui se ressemble, s'assemble ! ».

Prenons un exemple permettant de bien sentir l'idée de cet algorithme.
On remarque que dans la cour de récréation en Primaire, les jeunes enfants ont tendance à se regrouper entre filles ou entre garçons, mais assez peu à se mélanger.
En considérant ceci, on peut essayer de prédire le sexe d'un enfant-test (de genre inconnu) simplement en observant quel est le genre majoritaire (fille ou garçon) de ses plus proches voisins.
Voir l'animation *Geogebra* en annexe (`https://youtu.be/JPPYlLzch6A`).

De façon plus générale, **l'algorithme doit prédire la classe d'un élément en fonction de la classe majoritaire de ses k plus proches voisins**.

Remarque 1 : le nombre de classes n'est pas limité à deux comme dans l'exemple fille/garçon.

Remarque 2 : dans la pratique, la notion de proximité (proches voisins) n'est pas forcément liée à une distance physique (combien de mètres ?) entre l'élément test et les autres éléments des données de travail, mais plutôt à un « qualificatif de similarité » entre les éléments (quelques exemples possibles : couleur des yeux, âge, production de pétrole, PIB, etc.)

2. Descriptif des données

Dans le but d'un apprentissage pour classification, les données contiennent deux types d'informations :

- des **caractéristiques numériques destinées à la comparaison** de deux éléments ;
- un **critère destiné à la classification** d'un élément.

Les caractéristiques des éléments doivent permettre de définir une « distance » séparant les éléments.

Enfin, l'algorithme repose sur un **entier k précisant le nombre de voisins à considérer** pour établir la classe de l'élément non étiqueté (de classe inconnue).

3. Fonction distance

Nous limiterons notre étude à 2 caractéristiques pour chaque élément, ce qui permettra de représenter les éléments dans un repère orthonormé du plan (abscisse et ordonnée sont les 2 caractéristiques).
Nous utiliserons la distance euclidienne entre ces points du plan pour définir « la proximité » entre deux éléments.

Formule de la distance :
Dans un repère orthonormé du plan, on définit les points $A(x_A ; y_A)$ et $B(x_B ; y_B)$.
La longueur AB vaut :

$$AB = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

4. Description de l'algorithme

Un ensemble de n données est fourni, chacune de ces données contient deux données numériques $(x_i ; y_i)$ et une donnée supplémentaire correspondant à une classe c_i .
Un nouvel élément N étant donné avec ses valeurs numériques $(x_N ; y_N)$, on cherche à la classe c_N inconnue à laquelle il pourrait appartenir.

Voici les différentes étapes permettant d'apporter une réponse :

- Parcourir l'ensemble des n données de travail et calculer les distances entre $(x_i; y_i)$ et $(x_N; y_N)$ pour chaque élément i ;
- Classer ces résultats par distance croissante.
- Extraire de la liste les k plus proches éléments (correspondant aux plus petites distances précédentes).
- Dans ces k éléments, compter le nombre d'occurrences de chaque classe présente.
- Attribuer au nouvel élément N la classe la plus fréquente. C'est tout !