A- Une branche de l'Intelligence Artificielle

L'intelligence artificielle a vu officiellement le jour en 1956, au Dartmouth College de Hanover, dans l'État du New Hampshire (États-Unis), lors d'une école d'été organisée par quatre chercheurs américains, John McCarthy (1927-2011), Marvin Minsky (1927-2016), Nathaniel Rochester (1919-2001) et Claude Shannon (1916-2001). Depuis, l'expression « intelligence artificielle », inventée par John McCarthy sans doute pour frapper les esprits, est devenue très populaire.

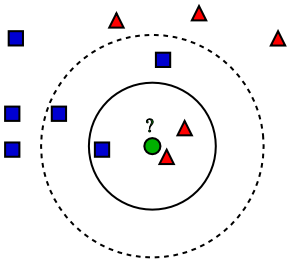
En intelligence artificielle (IA), plus précisément en apprentissage automatique (Machine Learning), la méthode des *k* plus proches voisins est une méthode d'apprentissage supervisé (supervised learning). En abrégé k-NN ou KNN, de l'anglais *k-Nearest Neighbors*.

L'algorithme *k*-NN est parmi les plus simples des algorithmes de machines learning.

B- Les k plus proches voisins

La méthode *k*-NN est basée sur l'apprentissage préalable, ou l'apprentissage faible qui classe des éléments en fonction de caractéristiques (ou variables prédictives).

Ce jeu de données d'apprentissage (training set en anglais) va servir de référence à la classification (étiquetage) de nouveaux éléments non étiquetés.



B.1- La classification

Elle permet de classer les nouveaux éléments en un nombre fini de catégories (ex : un nouvel email est-il spam ou non).

B.2- La régression

Elle fournit la valeur moyenne, des k plus proches voisins (un nombre).

Par exemple, on peut estimer automatiquement le prix d'un appartement en fonction de variables prédictives (telles que sa situation, son exposition, sa surface,...) en faisant la moyenne du prix des 3 appartements ayant les caractéristiques les plus proches.

B.3- Algorithme KNN

Vous pouvez lancer la présentation `Anim1_KNN.odp` pour visualiser cet algorithme.

Données d'entrées :

```
base_apprentissage, k, inconnue
```

Algorithme :

```
Pour chaque donnée dans base_apprentissage
        Calculez la distance entre l'inconnue et la donnée courante.
        Mémoriser cette distance et la donnée associée dans une liste de couple
        (distance, donnée).
Triez la liste distance_Donnée du plus petit au plus grand sur les distances.
Choisissez les k premières entrées de cette liste.
Obtenir les étiquettes des k données de la base d'apprentissage sélectionnées.
Si régression, renvoyer la moyenne des k étiquettes.
Si classification, renvoyer l'étiquette majoritaire parmis ces k étiquettes.
```

B.4- Calcul de la distance.

Il existe plusieurs façons de calculer la distance. Lorsque les variables prédictives sont comparables on utilise la distance euclidienne.

La distance euclidienne entre deux points du plan dont les coordonnées sont $(x_A; y_A)$ et $(x_B; y_B)$ est donnée par la formule (qui découle du théorème de Pythagore) :

$$dist(A,B) = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

Si on a trois caractéristiques numériques pour chaque point (on peut penser à des points de l'espace) de coordonnées $(x_A; y_A; z_A)$ et $(x_B; y_B; z_B)$, on aura :

$$dist(A,B) = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2}$$

On le généralise dans le cas où chaque point est caractérisé par quatre, cinq … et N caractéristiques numériques (variables prédictives).

B.5- Choix de k, le nombre de voisins

Dans le cas de la classification binaire (deux types d'étiquettes différentes), on utilise un k impair pour être sûr d'obtenir une étiquette majoritaire. On ne choisit pas la valeur k=1 car elle trop sujette aux erreurs ou aberrations possibles dans le jeu de données d'apprentissage.

Regarder la seconde présentation : `Anim2_KNN.odp`

Le choix raisonnable de k est donc un art à part entière. La technique la plus répandue (celle qui donne de meilleurs résultats pratiques) consiste à tester plusieurs valeurs de k et à choisir celle qui donne les meilleurs résultats.

B.6- Complexité algorithmique de KNN.

Un algorithme de recherche de voisinage consiste à analyser l'ensemble des `n` valeurs de `A` (`A` étant le jeu de données d'apprentissage) et à regarder si ce point est plus proche ou non qu'un des plus proches voisins déjà trouvé. On obtient alors un temps de calcul linéaire en la taille de `A` : `O(n)` (tant que `k` est très inférieur à `n`).

Cette méthode est appelée : recherche séquentielle ou recherche linéaire.

B.7- Conclusion.

L'algorithme de classification k-NN est sans doute un des plus simples de toutes les techniques d'apprentissage automatique (machines learning).

Il fonctionne bien avec des données prédictives numériques et si le jeu de données d'apprentissage n'est pas trop important.

Le choix du jeu de données d'apprentissage est déterminant dans l'efficacité des prédictions et nécessite une expertise métier.