L'algorithme des k plus proches voisins

Algorithmes d'apprentissage

Jusqu'à présent, tous les algorithmes que nous avons pu étudier, ou écrire, donnaient une réponse à un problème bien défini. Cette réponse étaient généralement exacte, ou approchée dans le cas des algorithmes gloutons. Les **algorithmes d'apprentissage** ont pour but de "deviner" une réponse en se basant sur un grand nombre d'exemples. Ils sont utilisés dans les conditions suivantes :

- Le problème est trop complexe pour que la réponse puisse être calculée de façon directe.
- Définir ce qu'est la meilleure solution n'est pas évident. Par exemple, comment déterminer la meilleure traduction d'une phrase dans une autre langue ?
- Les informations permettant de déterminer la bonne réponse sont partielles. On peut connaître, par exemple, les images de quelques nombres par une fonction mais pas sa définition explicite.

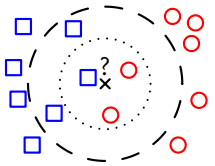
Pour résoudre ce genre de problème avec un algorithme d'apprentissage, il faut disposer d'un grand nombre d'exemples. Ces exemples sont généralement déterminés à l'avance, comme des banques d'images avec, pour chacune, une indication sur les objets ou animaux qu'elle contient. Ils peuvent également être générés, par exemple, en simulant le déroulement d'un jeu afin de déterminer une stratégie permettant de gagner. L'apprentissage peut alors se faire par raffinement successifs afin de rapprocher le plus possible les réponses obtenues des réponses attendues, comme c'est le cas avec les réseaux de neurones. Ou alors, l'algorithme va chercher les exemples les plus semblables aux paramètres donnés en entrée afin de déterminer la réponse la plus probable.

Ils sont comment tes voisins ?

L'**algorithme des k plus proches voisins**, ou k-NN (pour k nearest neighbors), est un algorithme de classification. À chaque élément donné, il faut déterminer à quelle classe il appartient, parmi celles déjà prédéfinies.
Pour trouver la bonne réponse, on regarde la classe la plus représentée parmi les k exemples les plus proches. Il faut donc avoir une notion de distance permettant de déterminer quels sont les exemples les plus proches. Pour l'instant, nous utiliserons juste la **distance euclidienne** dans le plan, c'est-à-dire la longueur du segment reliant les deux points considérés.

Exercice 1 : À l'aide de la figure ci-contre, déterminez si l'élément représenté par la croix est un carré ou un cercle en considérant :

1) le plus proche voisin
2) les 3 plus proches voisins
3) les 5 plus proches voisins



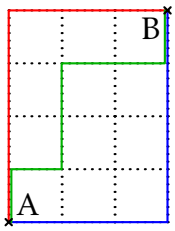
Distances

En mathématiques, une **distance** est une fonction d qui prend deux éléments et renvoie un réel positif et qui vérifie les propriétés suivantes pour n'importes quels éléments a, b et c :

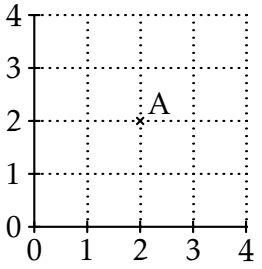
- Séparation : $d(a,b) = 0 \iff a = b$
- Symétrie : $d(a,b) = d(b,a)$
- Inégalité triangulaire : $d(a,c) \leq d(a,b) + d(b,c)$

La distance euclidienne n'est pas la seule distance que l'on peut définir. On peut, par exemple, considérer la **distance de Manhattan** entre deux points qui correspond à la somme des écarts sur les différentes coordonnées.

- **Distances dans le plan entre** $A(x_A,y_A)$ **et** $B(x_B,y_B)$ **:**
  - Distance euclidienne : $\sqrt{(x_B-x_A)^2+(y_B-y_A)^2}$.
  - Distance de Manhattan : $|x_B-x_A|+|y_B-y_A|$. Dans la figure ci-contre, les 3 chemins ont la même longueur, qui est la distance de Manhattan entre les deux points.



- **Distances dans l'espace entre** $A(x_A,y_A,z_A)$ **et** $B(x_B,y_B,z_B)$ **:**
  - Distance euclidienne : $\sqrt{(x_B-x_A)^2+(y_B-y_A)^2+(z_B-z_A)^2}$.
  - Distance de Manhattan : $|x_B-x_A|+|y_B-y_A|+|z_B-z_A|$.

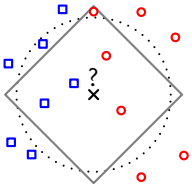


Exercice 2 : Soit un point A du plan ci-contre.

1) Tracer tous les points à une distance euclidienne de 2 de A.
2) Tracer tous les points à une distance de Manhattan de 2 de A.

Exercice 3 : Déterminez la catégorie à laquelle appartient l'élément représenté par la croix en regardant les 3 plus proches voisins :

1) en utilisant la distance euclidienne.
2) en utilisant la distance de Manhattan.



Les distances ne sont pas uniquement utilisées dans le plan ou l'espace. Par exemple, la **distance de Hamming** entre deux chaînes de caractères de même longueur correspond au nombre de symboles qui sont différents entre les deux chaînes. Si on note $d_H(m_1,m_2)$ la distance de Hamming entre les mots m_1 et m_2, alors $d_H(\texttt{vélo},\texttt{véto}) = 1$ et $d_H(\texttt{01234},\texttt{12345}) = 5$.

Exercice 4 : Déterminer la distance de Hamming entre les chaînes suivantes.

1) `chapeau` et `chameau` 2) `camion` et `chaton` 3) `0001` et `1000`

Pour des chaînes de longueur différente, ou non, on peut également utiliser la **distance de Levenshtein** qui mesure le nombre d'édition (modification, ajout, retrait) de symboles qu'il faut faire pour passer d'une chaîne à l'autre. Par exemple on peut passer de `niche` à `chiens` en enlevant le `n` et le `i` puis en rajoutant le `i`, le `n` et le `s`. La distance est donc de 5.

Exercice 5 : Déterminer la distance de Levenshtein entre les chaînes suivantes.

1) `verre` et `verte` 2) `honda` et `hyundai` 3) `01234` et `12345`

Pour aller plus loin

Comme nous l'avons vu, le choix de la distance utilisée ou la valeur de k peut modifier le résultat obtenu. Pour la distance, c'est en général le type de données qui va guider le choix. Par contre, pour la valeur de k, il est possible de mettre de côté une partie des exemples et de tester pour différentes valeurs de k, laquelle donne le plus souvent la réponse attendue. C'est une autre technique d'apprentissage.

Il existe des variantes de l'algorithme des k plus proches voisins. On peut pondérer les valeurs des voisins en fonction de leur distance à l'élément étudié. Ainsi le voisin le plus proche peut avoir beaucoup plus de poids que les deux suivants réunis s'ils sont plus éloignés.

Cet algorithme n'est pas utilisé que pour faire de la **classification** mais aussi de la **régression**, c'est à dire approximer l'image de l'élément donné par une fonction en faisant la moyenne des images des k plus proches voisins.