TD14 KNN. Do you speak French ?

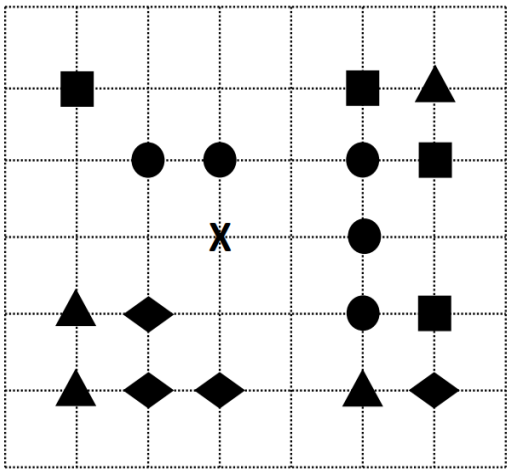
Introduction

L'algorithme des k plus proches voisins intervient dans de nombreux domaines de l'apprentissage automatique : c'est l'un des algorithmes utilisés dans le domaine de l'intelligence artificielle.

Il est utilisé par Amazon pour prévoir quels produits sont susceptibles d'intéresser un client.

Dans ce premier exemple, on a représenté sur un quadrillage les éléments de quatre classes (chaque classe est représentée par un carré, un triangle, un losange ou un disque) ainsi qu'un nouvel élément X.

En appliquant l'algorithme des k plus proches voisins pour la distance usuelle dans le plan, avec $k = 5$, à quelle classe est affecté le nouvel élément X ?



Le but de ce TD est de déterminer si un texte est écrit en français ou en anglais en appliquant l'algorithme des k plus proches voisins.

1. Classification

Pour distinguer le français de l'anglais, nous allons utiliser deux critères : les fréquences d'utilisation des lettres « u » et « h ».

Vous disposez de 20 textes :

- 10 en français (de `F0.txt` à `F9.txt`)
- 10 en anglais (de `A0.txt` à `A9.txt`)

Ces 20 textes sont à télécharger depuis Moodle ainsi que le fichier Python knn.py à compléter.

1.1. Compléter le fichier frequence.py pour déterminer la fréquence (en %) d'utilisation de la lettre « u » et de celle de la lettre en « h » dans chacun des 20 textes.

On optimisera le code pour traiter les 20 textes en une seule exécution du programme.

Remarque : la fonction `texte.lower()` permet de convertir un caractère ou en texte en minuscules.

1.2. Regrouper tous les résultats dans une liste de tuples ainsi constituée :

$$[(f_u, f_h, L), (f_u, f_h, L), \dots]$$

où f_u représente le pourcentage de « u », f_h celui de « h » et L la langue (« F » ou « A »).

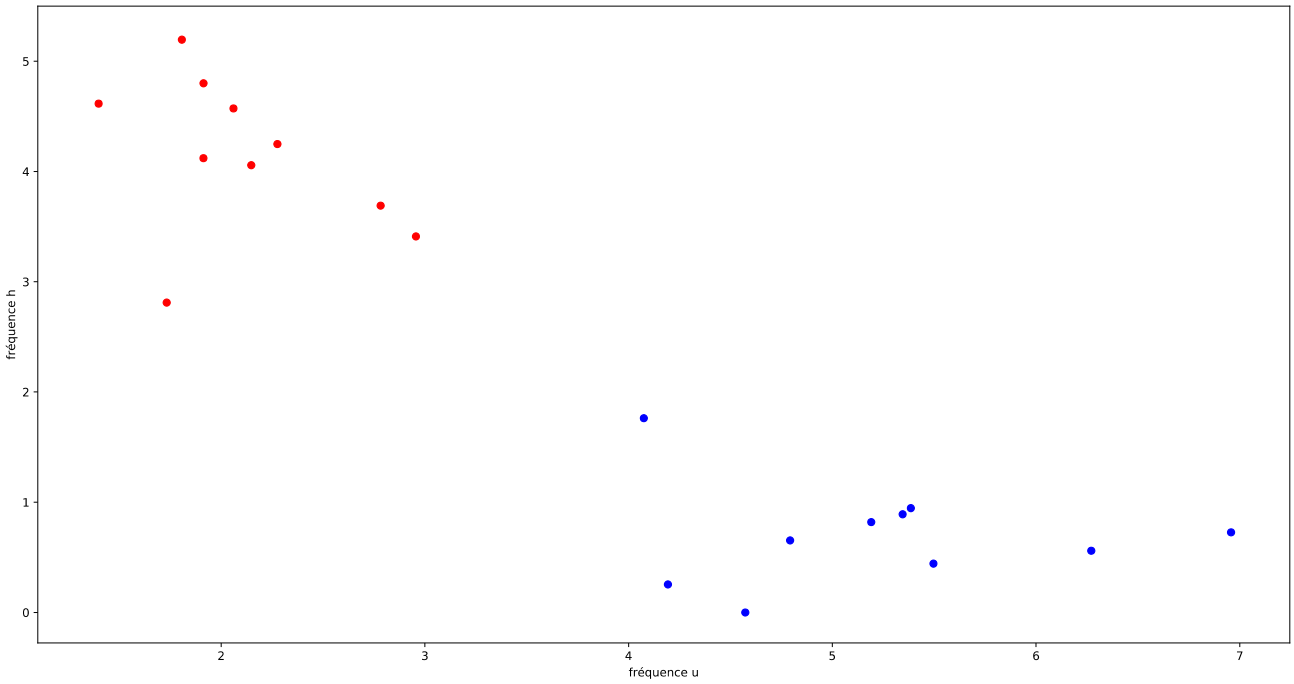
1.3. Représenter toutes les valeurs calculées sur un graphe avec les conventions suivantes :

- en abscisse : la fréquence des « u »
- en ordonnées : la fréquence des « h »
- les textes en français seront représentés en bleu
- les textes en anglais seront représentés en rouge.

Rappels :

- l'instruction `scatter(x, y, c='green')` de la bibliothèque pylab permet d'afficher sur un graphe le point de coordonnées (x, y) en vert.
- Le graphe est affiché après exécution de la fonction `show(block = False)`
- `block = False` permet de continuer l'exécution après affichage.

Graphe attendu :



2. Anglais ou Français

Nous allons maintenant utiliser les données collectées pour déterminer si un texte et en français.

2.1. Demander à l'utilisateur de saisir ou coller un texte.

2.2. Déterminer sur ce texte a fréquence des « u » et des « h ».

2.3. Créer une liste `LT` de tuples ainsi constituée :

$$[(d_0, L), (d_1, L), \dots, (d_{19}, L)]$$

où d_i représente la distance sur le graphe entre le point qui correspondrait au texte saisi et le texte *i* de référence et L la langue de ce texte (« F » ou « A »).

2.4. Classer cette liste par ordre de distance croissante.

2.5. Relever la langue majoritaire dans les trois plus proches voisins (k = 3) pour déterminer si la saisie est en anglais ou en français.

2.6. Afficher la réponse.

Exemples.

```
Saisir un texte : My tailor is rich
Ce texte est en anglais
Saisir un texte : On n'est jamais mieux servi que par soi-même.
Ce texte est en français
```