

TP - Joueurs de rugby

Objectifs :

- ▷ Appliquer l'algorithme k-voisins

1 Travail à faire

On a relevé la taille et le poids des différents joueurs de rugby du *Top 14* ainsi que leur poste sur le terrain au cours de la saison 2023-2024. Vous allez travailler sur un tableau de données des joueurs de rugby du Top 14 issues de la page de la ligue nationale de rugby : <https://top14.lnr.fr/joueurs>

Votre travail consiste à attribuer un poste sur le terrain à un joueur à partir de son poids et de sa taille en utilisant l'**algorithme des k plus proches voisins** :

— Demi mêlée, arrière, 1ère ligne, 2ème ligne, ..., etc.



2 Exercices

Exercice 1 : Extraction des données

1. Analyser le fichier **JoueurTop14.csv** avec *Notepad++* par exemple.
2. Ecrire une fonction `liste_joueurs` de paramètre un fichier `csv` et qui renvoie une liste de dictionnaires contenant les données 'Poste', 'Taille' (en cm) et 'Poids' (en kg) du fichier.

Exercice 2 : Liste des postes

Ecrire une fonction `postes` qui a pour paramètre la liste des joueurs et qui renvoie un dictionnaire dont les clés sont les postes et les valeurs 0, plus tard cela correspondra au nombre de postes.

Exercice 3 : Liste des joueurs

Ecrire une fonction `liste_joueurs_par_rapport` qui a les paramètres `taille`, `poids`, `liste` des joueurs qui renvoie une liste de dictionnaires ayant deux clés le poste et la distance par rapport aux 'taille' et 'poids' saisis.

Exercice 4 : KNN

Ecrire une fonction `KNN` qui a les paramètres `k` le nombre de voisins, `taille`, `poids`, `liste_des_joueurs` qui renvoie le dictionnaire des postes mis à jour.

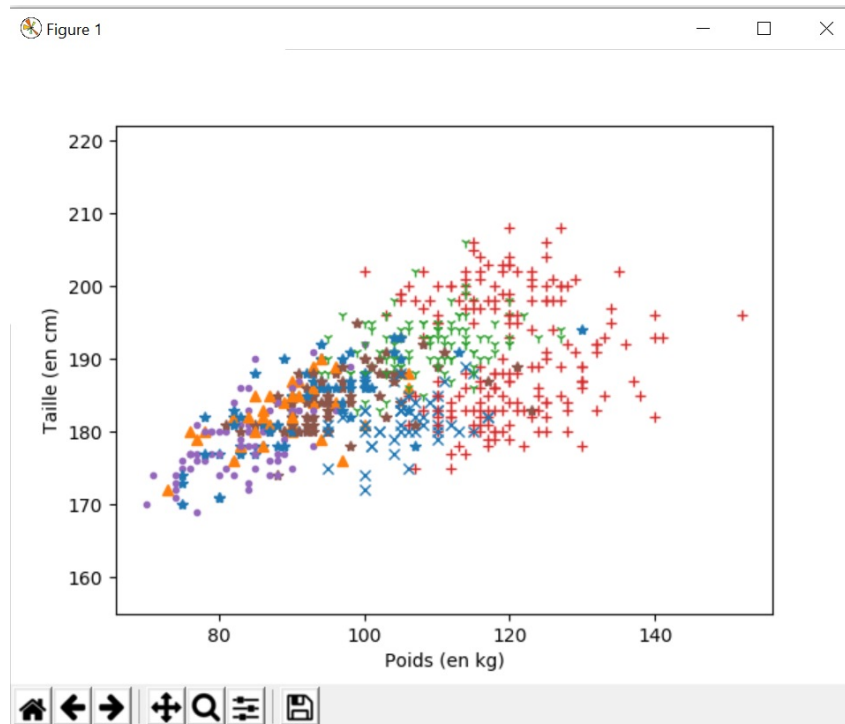
Exercice 5 : Prédiction

Ecrire une fonction `prediction` qui a les paramètres `taille`, `poids`, `k` le nombre de voisins, `liste_des_joueurs`, qui renvoie une prédiction sur le 'Poste' le plus adapté.

3 Le graphique

Voici le nuage de points de cette série réalisée avec le module `matplotlib`.

Il faut être dans un repère orthonormé pour utiliser la distance euclidienne.



Voici le code Python permettant d'afficher le nuage de points :

```

1 import matplotlib.pyplot as plt
2
3 def representation(data=joueurs):
4
5     for i in range(len(data)):
6         lacouleur="tab:"
7         if(data[i]['Poste']=="Talonneur"):
8             lacouleur+="blue"
9             lemarker="x"
10            label="Avant"
11        elif(data[i]['Poste']=="Pilier"):
12            lacouleur+="red"
13            lemarker="+"
14            label="Melee"
15        elif(data[i]['Poste']=="2eme ligne"):
16            lacouleur+="red"
17            lemarker="+"
18            label="2eme ligne"
19        elif(data[i]['Poste']=="3eme ligne"):
20            lacouleur+="green"
21            lemarker="1"
22            label="3eme ligne"
23        elif(data[i]['Poste']=="Melee"):
24            lacouleur+="purple"
25            lemarker="."
26            label="Melee"
27        elif(data[i]['Poste']=="Ouverture"):
28            lacouleur+="purple"
29            lemarker="."
30            label="Ouverture"

```

```
31 elif (data[i]['Poste']=="Centre"):  
32     lacouleur+="brown"  
33     lemarker="*"  
34     label="Centre"  
35 elif (data[i]['Poste']=="Ailier"):  
36     lacouleur+="blue"  
37     lemarker="*"  
38     label="Ailier"  
39 else :  
40     lacouleur+="orange"  
41     lemarker="^"  
42     label="Arriere"  
43  
44 plt.plot(int(data[i]['Poids']), int(data[i]['Taille']),  
45          color=lacouleur,marker=lemarker,label=label)  
46 plt.xlabel("Poids (en kg)")  
47 plt.ylabel("Taille (en cm)")  
48 plt.axis('equal') #pour avoir un repere orthonorme  
49 plt.show()  
50 representation()
```