



Travaux pratiques

Afin de travailler sur un exemple, nous allons utiliser un jeu de données relativement connu dans le monde du machine learning : le jeu de données "iris".

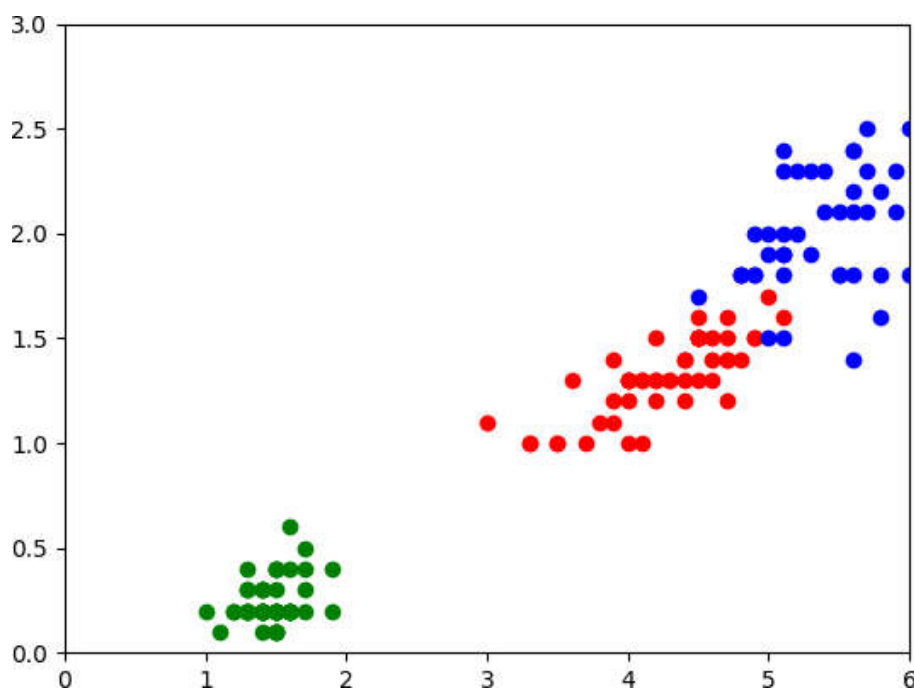
En 1936, Edgar Anderson a collecté des données sur 3 espèces d'iris : "iris Setosa", "iris Virginica" et "iris Versicolor".

Pour chaque iris étudié, Anderson a mesuré (en cm) :

- La largeur des sépales
- La longueur des sépales
- La largeur des pétales
- La longueur des pétales
- L'espèce ("iris Setosa", "iris Virginica" ou "iris Versicolor")

Par souci de simplification, nous nous intéresserons uniquement à la largeur, la longueur des pétales et l'espèce de l'iris.

On trouvera 150 de ces mesures dans le **fichier *iris.csv*** où l'espèce de l'iris est un chiffre en 0 et 2 avec 0 pour "iris Setosa", 1 pour "iris Virginica" et 2 pour "iris Versicolor".



On fournit également le fichier **TP_fleur.py** , qui permet de lire le fichier csv et de représenter dans un repère chaque mesure par un point, où la longueur d'un pétale correspond à son abscisse, la largeur du pétale à son ordonnée et l'espèce de l'iris à sa couleur (vert pour "iris Setosa", rouge pour "iris Virginica" et bleu pour "iris Versicolor").

- 1). Lire et s'appropriier le code python.
- 2) Rajouter « à la bonne ligne » dans le code le point de coordonnées **(2.0; 0.5)** et de couleur noire.
- 3) Déterminer visuellement la classe du point que vous venez de rajouter avec l'algorithme des 3-ppv.
- 4) Reprendre les deux questions précédentes avec le point de coordonnées **(5.1; 1.7)**.

A partir du fichier « kplusproche.py »:

- 5) Écrire la fonction « **donne_coords** » qui prend en paramètres deux iris et renvoie une paire correspondant aux coordonnées (écrire des assertions pour bien faire comprendre la fonction aux élèves)
- 6) Écrire la fonction « **distance_iris** » qui prend en paramètres deux iris et renvoie la distance euclidienne associée à leurs coordonnées ;
- 7) Écriture de la fonction « **k_plus_proche_1** » qui prend en paramètres une liste d'iris, un iris et un entier k et renvoie la liste des k plus proches voisins de l'iris. On utilisera ici la fonction **sort** avec un lambda (à discuter)
- 8) Tester votre code avec quelques points au hasard.