1 Problématique

En 1936, le biologiste *Ronald Fisher* a rassemblé les mesures de trois espèces d'iris. Ces informations permettent de distinguer nettement les différences entre ces variétés. Ce travail de collecte et de classification fournit une quantité importante de données mais ne présente aucune analyse. Cependant, il est possible d'utiliser ces données pour pouvoir classifier un iris inconnu.

Iris setosa

Iris versicolor

Iris virginica

Comment prédire une information nouvelle à partir de données brutes ?

2 Utiliser les données

2.1 Présentation graphique des informations

La lecture des données dans un tableau n'est pas très parlante. Une représentation graphique des informations apporte une compréhension plus éclairante.

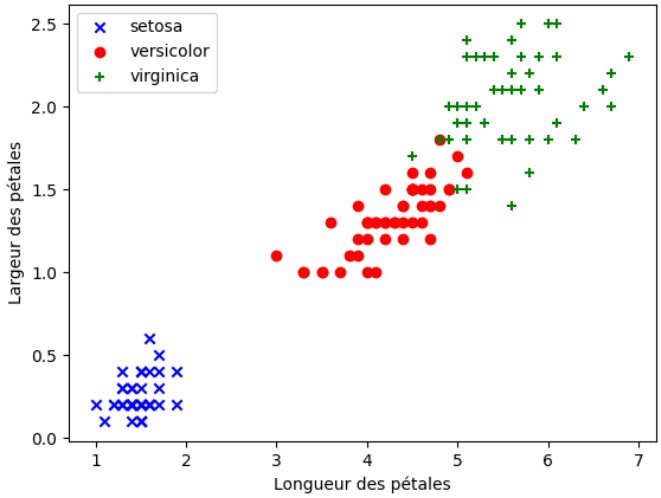

FIGURE 1 – Variétés d'iris en fonction de leurs mesures

Il apparaît que les mesures d'un iris peuvent permettre de déterminer leur variété.

2.2 Prédire la variété

La position d'un iris sur la représentation graphique (figure 1) est caractéristique de sa variété.

Activité 1 :

1. Déterminer la variété des iris suivants :

longueur	1	6	5.1	2.5
largeur	0.5	2.5	1.55	0.85

2. Proposer une méthode pour effectuer un choix dans les cas ambigus.

Christophe Viroulaud

3 Algorithme kNN

3.1 Présentation

Pour déterminer la variété d'un iris inconnu, une stratégie consiste à regarder celle d'un nombre k de voisins. On attribue alors à la fleur inconnue, la variété la plus présente parmi ses k voisins. C'est la méthode des **k plus proches voisins** (**k** **N**earest **N**eighbors).

Complément

L'algorithme *kNN* est une méthode d'apprentissage *supervisé* : l'algorithme reçoit un ensemble de données déjà étiquetées sur lequel il va pouvoir s'entraîner et définir un modèle de prédiction.

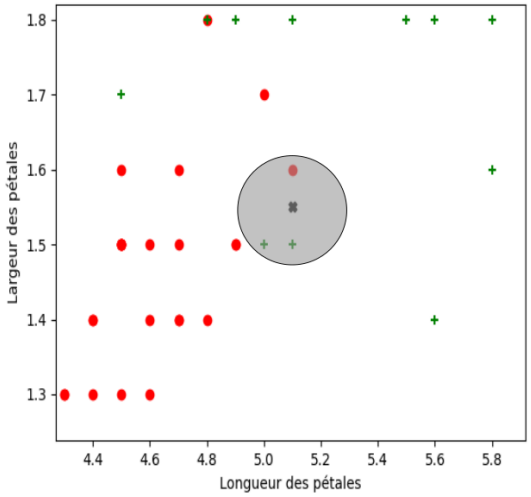

FIGURE 2 – Détermination de l'iris (5.05, 1.5) pour $k = 3$

3.2 Construction de l'algorithme

Une fois la valeur k choisie, pour déterminer les plus proches voisins on calcule la distance entre le point *cible* et les autres. Il existe plusieurs manières de calculer une distance. Le plus naturel ici est de prendre la distance *à vol d'oiseau* ou plus formellement la **distance euclidienne**.

$$d = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

Activité 2 : Écrire *en langage naturel*, l'algorithme kNN.

3.3 Implémentation

Les données sont stockées dans un fichier *csv*. Il faut donc d'abord charger ces informations correctement dans le programme avant de pouvoir les utiliser.

Activité 3 :

1. Télécharger le dossier compressé *iris.zip* sur le site https://cviroulaud.github.io

Christophe Viroulaud

2. Ouvrir le fichier *data-iris.csv* avec un tableur pour observer les données.
3. Ouvrir le fichier *iris-eleve.py*.
4. Compléter la fonction *charger_donnees* en utilisant les informations du fichier *csv*.
5. Compléter la fonction *distance* qui calcule le carré de la distance euclidienne entre deux points du plan.
6. Compléter la fonction *calculer_distances*.
7. Compléter enfin la fonction *trouver_variete*. Le dictionnaire *compteur_voisins* compte le nombre d'apparitions de chaque variété parmi les k voisins.
8. Tester la fonction avec $k = 3$ puis $k = 7$, puis pour les autres iris de l'activité 1.
9. Pour les plus avancés : Modifier le code pour tester un ensemble de 10 iris inconnus. De plus chaque iris déterminé sera ajouté au dictionnaire *varietes* afin d'augmenter l'apprentissage de l'algorithme.

Christophe Viroulaud