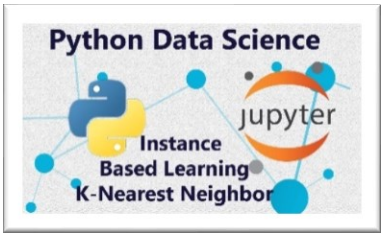
- algorithmique-

ALGORITHME DES k PLUS PROCHES VOISINS



Plan du chapitre

I. INTRODUCTION AU MACHINE LEARNING

Dans ce chapitre, nous allons travailler avec un algorithme d'apprentissage automatique, souvent appelé un algorithme de machine learning. Le principe de ces algorithmes est d'utiliser un grand nombre de données afin "d'apprendre à la machine" à résoudre des problèmes.
Bien que cette idée d'apprentissage automatique date de la fin des années 1950, le machine learning a pris toute son importance avec la montée en puissance du Big Data, offrant des quantités de données à analyser sur d'innombrables sujets. À noter aussi l'importance des stratégies mises en place par les GAFAM (Google, Apple, Facebook, Amazon et Microsoft) afin de récupérer un grand nombre de données concernant leurs clients. Ces données sont très souvent utilisées pour alimenter des algorithmes de machine learning (ce qui permet par exemple à Amazon de proposer à ces clients des "suggestions d'achats" souvent très pertinentes.)

Retour au plan

II. LES DONNÉES D'EDGAR ANDERSON RELATIVES A DIFFÉRENTES ESPÈCES D'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 la largeur et la longueur des sépales, la largeur et la longueur des pétales. Par souci de simplification, nous nous intéresserons uniquement à la largeur et à la longueur des pétales. 50 de ces mesures se trouvent dans le fichier *iris.csv* téléchargeable depuis le groupe de travail sur l'ENT. Ce jeu de données présente aujourd'hui un intérêt essentiellement pédagogique. En effet, il est exclusivement utilisé par des personnes désirant s'initier aux algorithmes de machine learning.

	A	B	C
1	petal_length	petal_width	species
2	1.4	0.2	0
3	1.4	0.2	0
4	1.3	0.2	0
5	1.5	0.2	0

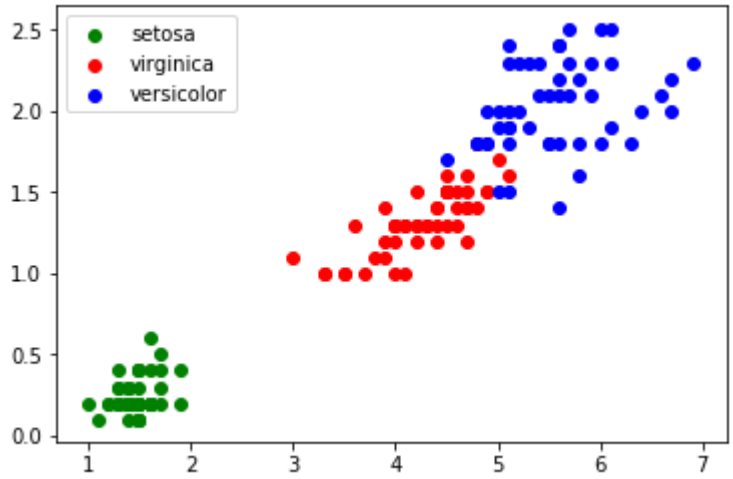
Les valeurs du champ *"species"* ("espèces") sont *0* pour l'espèce *"setosa"*, I pour "virginica" et *2* pour *"versicolor"*.

Nous allons dans un 1er temps réaliser une représentation graphique des données contenues dans le fichier à l'aide du script python suivant :

```python
import pandas    # module permettant de traiter des fichiers csv
import matplotlib.pyplot as plt

#traitement fichier  CSV
iris=pandas.read_csv("iris.csv")   #lecture du fichier iris.csv
x=iris.loc[:,"petal_length"]       # extraction du champs "petal_length"
y=iris.loc[:,"petal_width"]        # extraction du champs "petal_width"
espece=iris.loc[:,"species"]       # extraction du champs "species"

# tracé des points correspondants à l'esèce 0 ("setosa")
plt.scatter(x[espece == 0], y[espece == 0], color='green', label='setosa')
# tracé des points correspondants aux 2 autres espèces
plt.scatter(x[espece == 1], y[espece == 1], color='red', label='virginica')
plt.scatter(x[espece == 2], y[espece == 2], color='blue', label='versicolor')
plt.legend()  # affichage de la légende
plt.show()    # visualisation du graphique
```

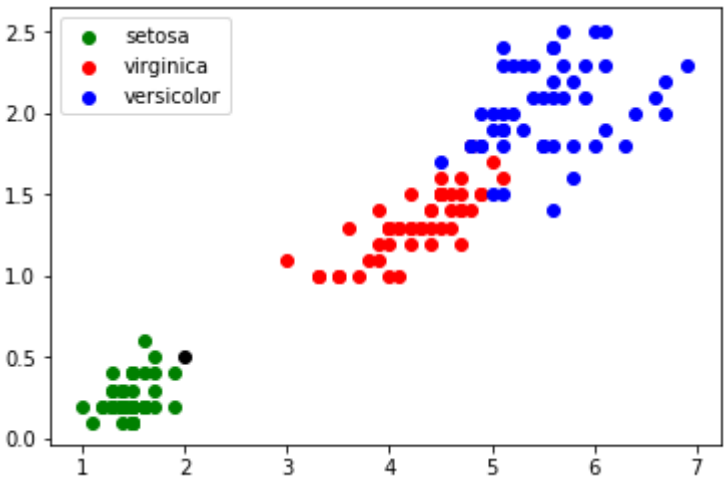


On constate que ces points sont regroupés par espèces d'iris.

Retour au plan

III. COMMENT APPRENDRE A RECONNAITRE UNE ESPÈCE D'IRIS ?

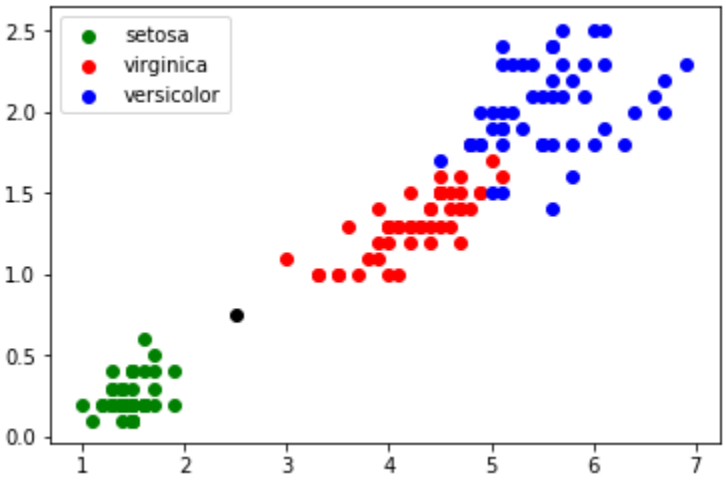
Considérons maintenant une iris dont les pétales mesurent 0,5 cm de large et 2 cm de long. Comment déterminer l'espèce à laquelle cette iris appartient ? Afin de répondre à cette problématique, nous allons placer cette nouvelle donnée sur le graphique :




```python
plt.scatter (2.0, 0.5, color="black")
```

Dans ce cas, on voit clairement qu'il y a de fortes chances pour que l'iris soit de l'espèce *"setosa"*.

Cependant, il existe des cas où il est beaucoup plus difficile de répondre, par exemple pour une iris dont les pétales mesurent 0,75 cm de large et 2,5 cm de long.



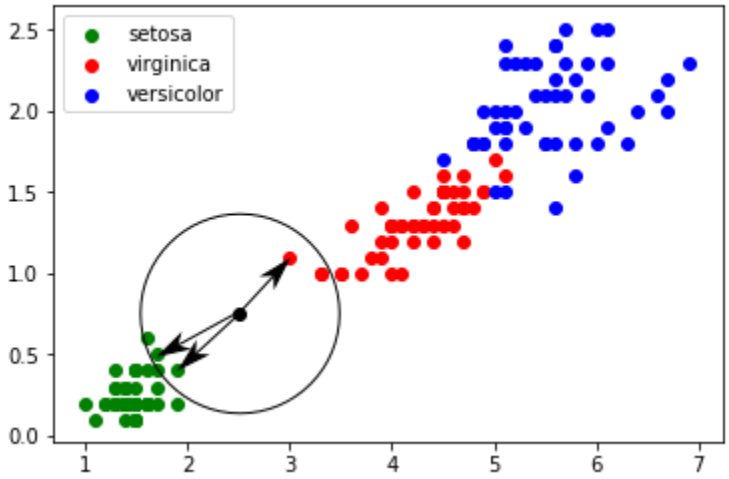
Afin de déterminer si l'iris appartient à l'espèce "sétosa" ou à l'espèce "virginica", nous allons devoir utiliser l'algorithme des k plus proches voisins.

Retour au plan

IV. ALGORITHME DES k PLUS PROCHES VOISINS

Quel est le principe de cet algorithme ?

- On calcule la distance entre le point correspondant à l'iris inconnu et chaque point issu du jeu de données "iris" ;
- On sélectionne les k distances les plus petites (les k plus proches voisins)
- Parmi les k plus proches voisins, on détermine quelle est l'espèce majoritaire, et on attribue à notre iris inconnu cette espèce majoritaire.



Prenons k = 3 :
Les 3 plus proches voisins sont deux *"setosa"* et un *"virginica"*.

D'après l'algorithme des "k plus proches voisins", notre iris inconnue appartient à l'espèce *"setosa"*.

La bibliothèque Python Scikit Learn propose un grand nombre d'algorithmes lié au machine learning (c'est sans aucun doute la bibliothèque la plus utilisée en achine learning). Parmi tous ces algorithmes, Scikit Learn propose l'algorithme des k plus proches voisins.
Importer la bibliothèque :

```
3 from sklearn.neighbors import KNeighborsClassifier
```

Compléter le programme précédent (la fonction *plt.show()* devra être placée en fin de script) :

```
#valeurs
k=3                 # Indiquer le nombre de plus proches voisins
longueur = 2.5      # données concernant l'iris inconnu
largeur = 0.75      # '                                '


#algo knn
d=list(zip(x,y))   #list(zip(x,y)) permet de passer des 2 listes x=[1.4, 1.4, 1.3 ...]  et y=[0.2, 0.2, 0.2...]
# a une liste de tuples d=[(1.4, 0.2), (1.4, 0.2), (1.3, 0.2)...]
model = KNeighborsClassifier(n_neighbors=k)   # cette fonction prend en argumùent le nombre k de plus proches voisins
model.fit(d,espece) # la méthode fit prend en arguments le tableau de tuples d et le tableau des espèces
prediction= model.predict([[longueur,largeur]]) # la méthode predict retourne une liste contenant  1 seul élément
                                                # : le n°  de l'espèce
#Affichage résultats
txt="Résultat : "
if prediction[0]==0:
  txt=txt+"setosa"
if prediction[0]==1:
  txt=txt+"virginica"
if prediction[0]==2:
  txt=txt+"versicolor"
plt.text(4,0.5, txt, fontsize=12)
plt.show()
```

Exécuter le script afin de déterminer l'espèce de l'iris inconnue.

Le résultat dépend-t-il du nombre de voisins ?

Pour répondre à cette question, relancer le script en prenant toujours la même iris inconnue mais en faisant varier le nombre de voisins. Que constatez-vous ?

Retour au plan