**Département Mathématiques et Informatique**

**Rapport du Projet**

**Data Mining , Machines**

**et Deep Learning**

**Filière :**

**« Ingénierie Informatique : Big Data et Cloud Computing »**

**II-BDCC**

**Utiliser la méthode K-means pour un problème de prévision**

Réalisé par : Encadré par :

Tarik Fertahi

Amina Dahmouni M. Zakaria EN-NAIMANI

Zakaria Mansouri

**Année Universitaire : 2020-2021**

**Année Universitaire : 2022-2023**

Introduction

Machine Learning (apprentissage automatique) est un sous-domaine de l’Intelligence Artificielle. C’est un ensemble de méthodes et de techniques ou algorithmes qui permettent aux ordinateurs de construire des modèles et de générer des prédictions sans utiliser d'instructions explicites ou être guidés par un humain.

Types d'apprentissage

Il existe 3 types d'apprentissage sont les suivantes :

**1- Apprentissage supervisé**

L'apprentissage supervisé (supervised learning en anglais) en input des individus labellisés, et l’objectif est de classer un nouvel individu non labellisé. C’est une tâche d'apprentissage automatique consistant à apprendre une fonction de prédiction à partir d'exemples annotés, au contraire de l'apprentissage non supervisé. On distingue les problèmes de régression des problèmes de classement. Ainsi, on considère que les problèmes de prédiction d'une variable quantitative sont des problèmes de régression tandis que les problèmes de prédiction d'une variable qualitative sont des problèmes de classification.

**2- Apprentissage non supervisé**

L'apprentissage non supervisé en input des individus non labellisés, et l’objectif est de faire le regroupement. Désigne la situation d'apprentissage automatique où les données ne sont pas étiquetées (par exemple étiquetées comme « balle » ou « poisson »). Il s'agit donc de découvrir les structures sous-jacentes à ces données non étiquetées. Puisque les données ne sont pas étiquetées, il est impossible à l'algorithme de calculer de façon certaine un score de réussite. Ainsi, les méthodes non supervisées présentent une auto-organisation qui capture les modèles comme des densités de probabilité ou, dans le cas des réseaux de neurones, comme combinaison de préférences de caractéristiques neuronales encodées dans les poids et les activations de la machine.

**3- Apprentissage par renforcement**

L’apprentissage par renforcement ou Reinforcement Learning est une méthode de Machine Learning. Elle consiste à entraîner des modèles d’intelligence artificielle d’une manière bien spécifique. La machine apprend de ses propres erreurs. La machine fait des choix, ensuite récompensée s’il s’agit d’un bon choix, si non pénalisé.

Techniques descriptives du Datamining

Les techniques descriptives visent à mettre en évidence des informations présentes mais cachées par le volume des données. Ces techniques consistent à réduire, résumer, synthétiser les données.

Méthodes descriptives

Il existe 3 méthodes descriptives sont les suivantes :

1- Analyse factorielle

2- Détection d’associations entre les individus

3- Segmentation

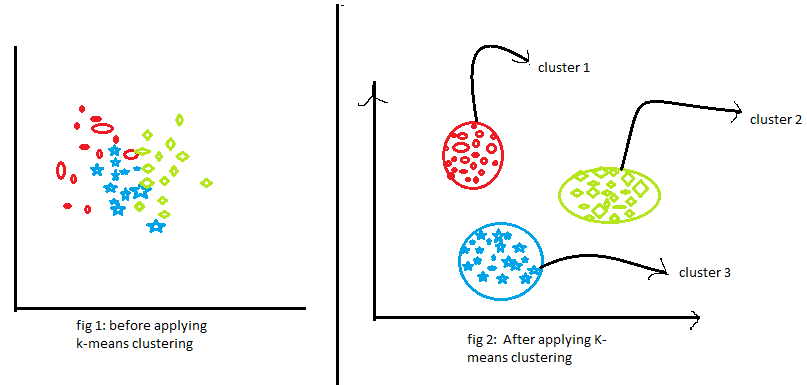
Segmentation

Segmentation c’est une méthode descriptive qui permet de regrouper des objets en groupes, ou classes, ou familles, ou segments, ou clusters, de sorte que :

* 2 objets d’un même groupe se ressemblent le plus possible,
* 2 objets de groupes distincts diffèrent le plus possible.

k-means

k-means est un outil de segmentation qui permet de répartir un ensemble de données en classes homogènes.



Problématique

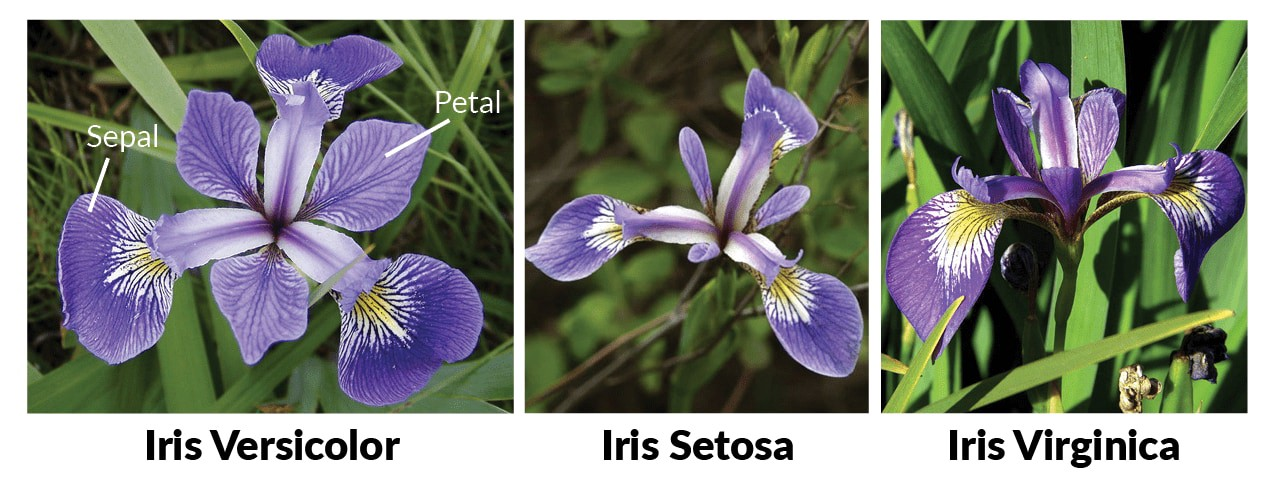
Utiliser l’algorithme Kmeans pour prédire le type d’une Iris flower.

**Le jeu de données :**

Le jeu de données Iris a été initialement publié à l’ [UCI Machine Learning Repository: Iris Data Set.](https://archive.ics.uci.edu/ml/datasets/Iris) Ce *Data Set* de 1936 est souvent utilisé pour tester des algorithmes d’apprentissage automatique et des visualisations.

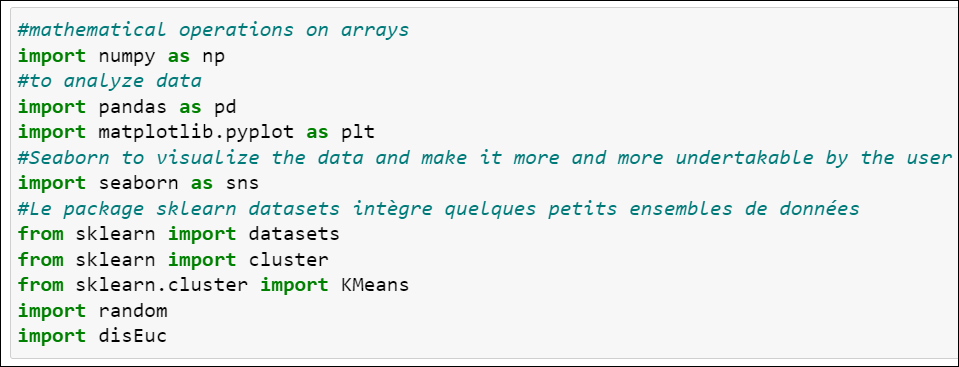
Le jeu de données *Iris* contient trois variantes de la fleur *Iris.* Il contient 150 instances (ligne du jeu de données). Chaque instance est composée de quatre attributs pour décrire une fleur d’Iris. Le jeu de données est étiqueté par le type de fleur. Ainsi pour quatre attributs décrivant une fleur d’Iris, on saura de quelle variante il s’agit.

Finalement, le jeu de données ne contient pas de [valeurs manquantes](https://mrmint.fr/donnees-manquantes-data-science). Ce qui nous dispense de les traiter.



# **Implémentation de K-Means:**

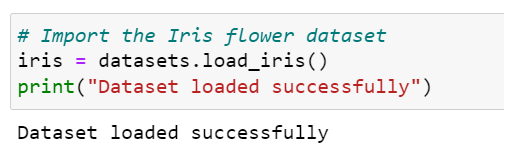
## **Import des librairies nécessaires**



## **Chargement des données**

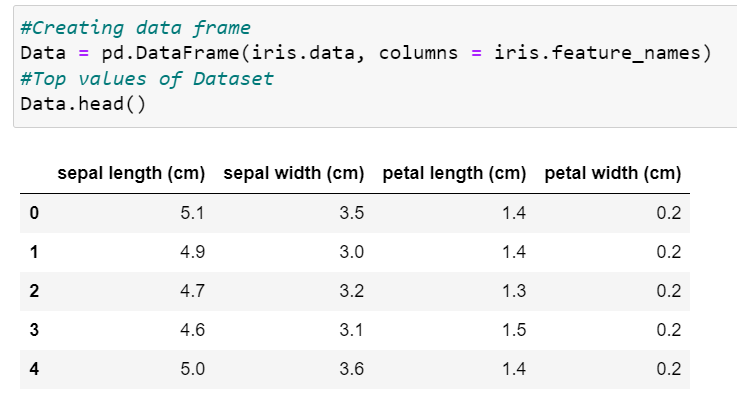
La librairie Sickit Learn offre des méthodes utilitaires pour charger des jeux de données populaires comme celui d’Iris. Ces méthodes se retrouvent dans la classe **datasets.**

Pour charger notre jeu de données Iris, on utilise la méthode **load\_iris()**.

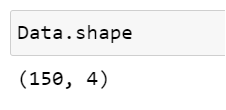


## **Utilisation de la librairie Pandas**

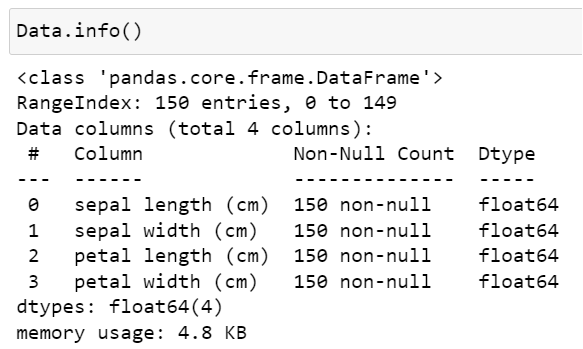
Pandas est une librairie assez utilisée quand on fait du Machine Learning avec Python. Grâce à sa classe **DataFrame**, on peut manipuler aisément les données en format tabulaire.



Informations générales sur notre dataset :



Nous avons 150 échantillons et 4 caractéristiques.



Le jeu de données ne contient pas de [valeurs manquantes](https://mrmint.fr/donnees-manquantes-data-science).

## **Construction du modèle K-means**

Choisissez le nombre de clusters k :

Nous avons utilisé la méthode Elbow pour déterminer le nombre optimal de clusters pour le clustering k-means.

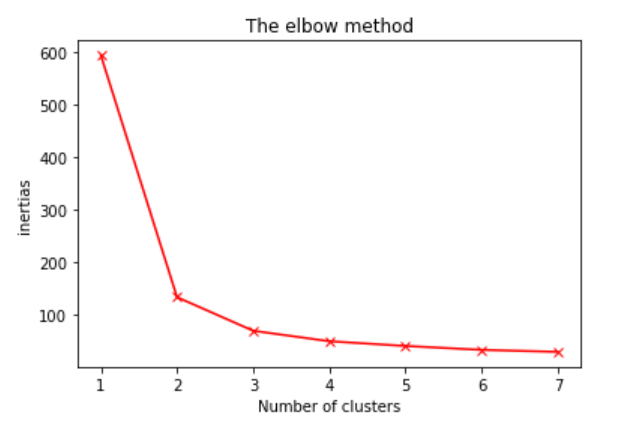
C’est une étape fondamentale pour tout algorithme non supervisé qui consiste à déterminer le nombre optimal de clusters dans lesquels les données peuvent être regroupées.

Nous démontrons maintenant la méthode donnée en utilisant la technique de clustering K-Means en utilisant la bibliothèque Sklearn de python.

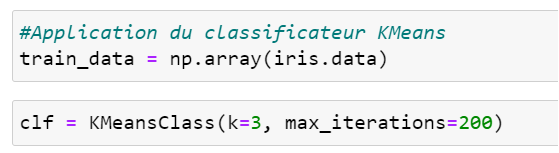
Pour trouver Inertie : C’est la somme des distances au carré des échantillons à leur centre de grappe le plus proche.

Nous itérons les valeurs de k de 1 à 8 et calculons l’inertie pour chaque valeur de k dans la plage donnée.

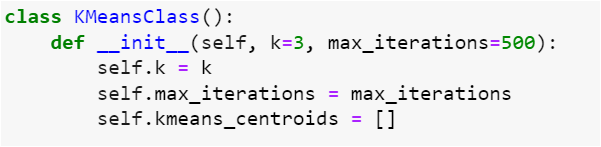




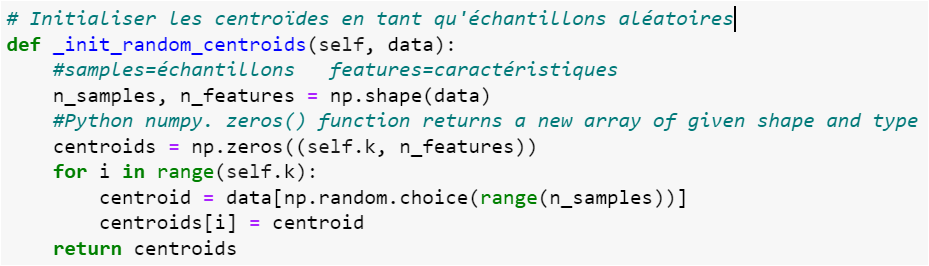
Pour déterminer le nombre optimal de clusters, nous devons sélectionner la valeur de k au « Elbow», c’est-à-dire le point après lequel l’inertie commence à diminuer de manière linéaire. Ainsi, pour Iris dataset, nous concluons que le nombre optimal de clusters pour les données est 3.



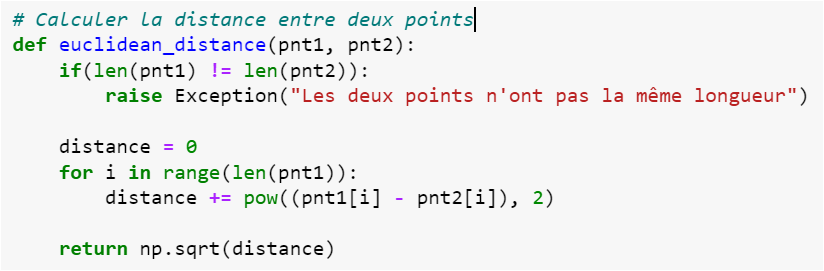
Définition de la classe KMeansClass avec le constructeur **\_\_init\_\_()** qui sera appelée lors de la création de l’objet pour l’initialisation:



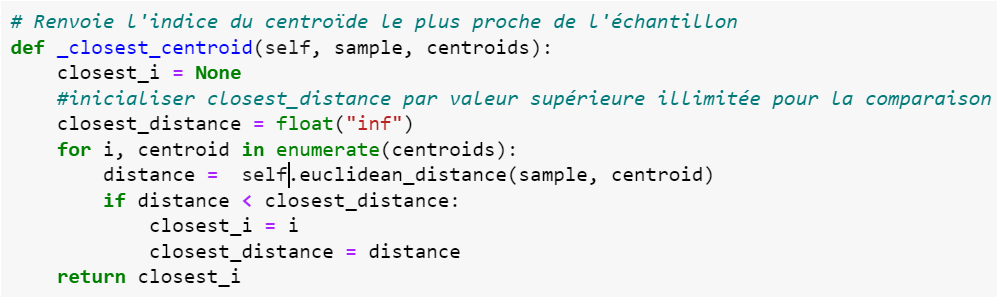
Initialiser les centroïdes aléatoirement :



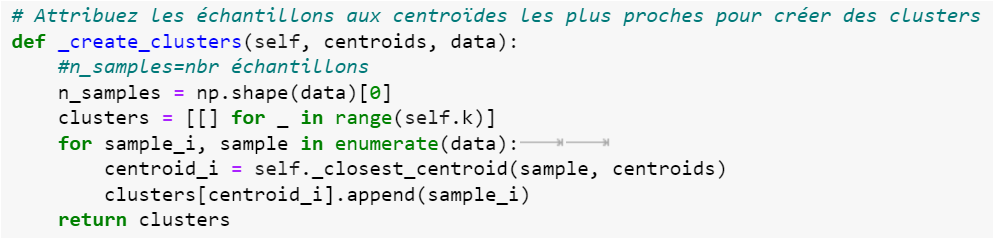
Calculer la distance euclidien entre deux points :



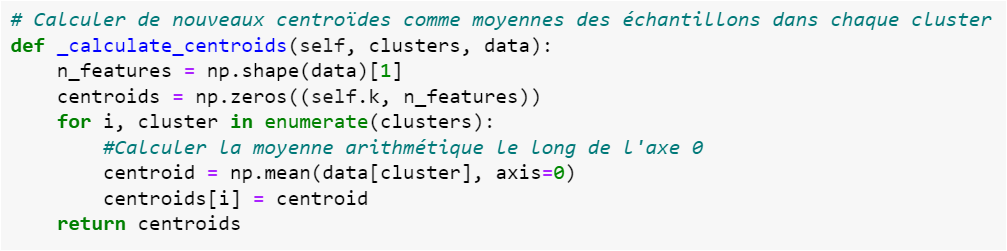
Envoyer l'indice du centroïde le plus proche de l’échantillon :



Affecter les échantillons aux centroïdes les plus proches pour créer des clusters :



Calculer de nouveaux centroïdes comme moyennes des échantillons dans chaque cluster :

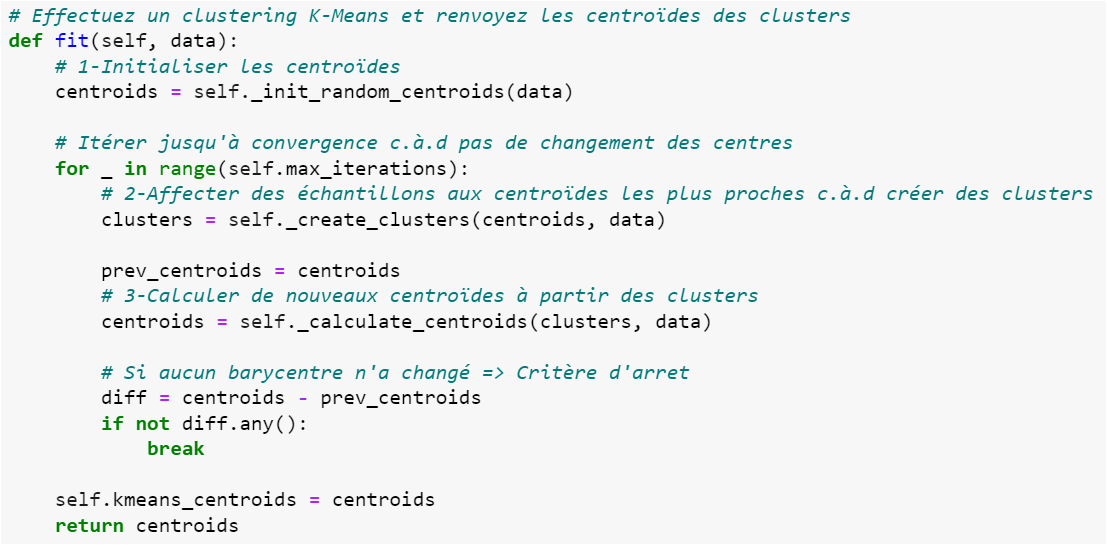


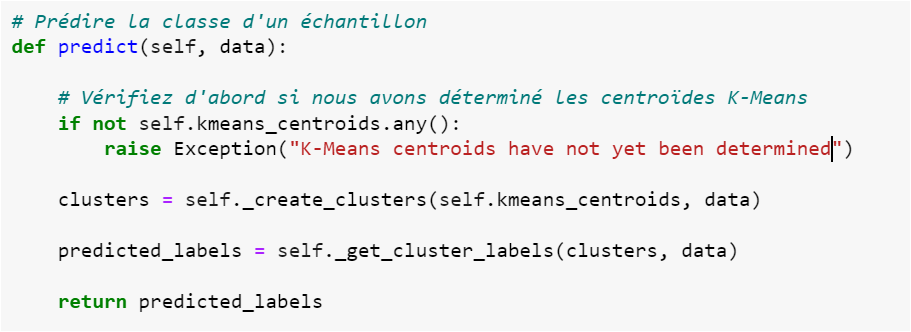
Effectuez un clustering K-Means et renvoyez les centroïdes des clusters :

Nous pouvons maintenant répéter les étapes d'attribution de points aux clusters et de mise à jour itérative des centres de cluster et observer la progression de l’algorithme.

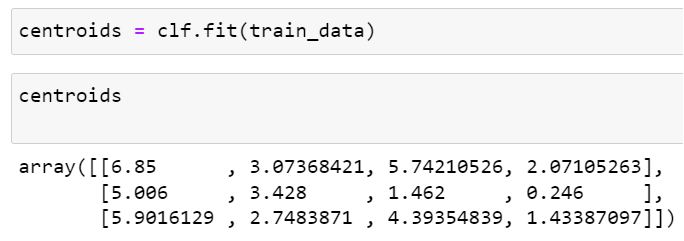
Après chaque itération, tester si un point modifie son appartenance au cluster.

Arrêtez l'algorithme si la convergence a été atteinte, c'est-à-dire que les clusters ne changent pas après l'étape de réallocation.

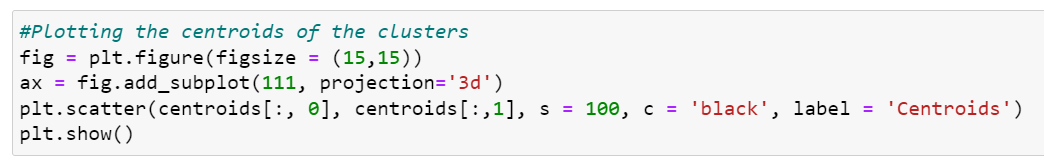


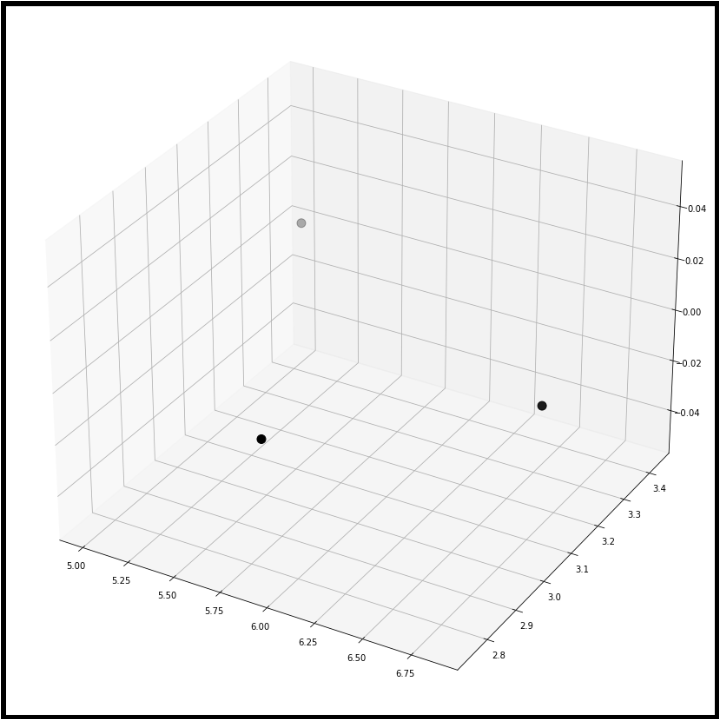


Trouver les centres des 3 clusters :



Affichons les centroïdes des clusters :





Classer les points de données dans les 3 clusters :



Visualiser la répartition des fleurs par clusters :

