COURS DE *DATA SCIENCE*

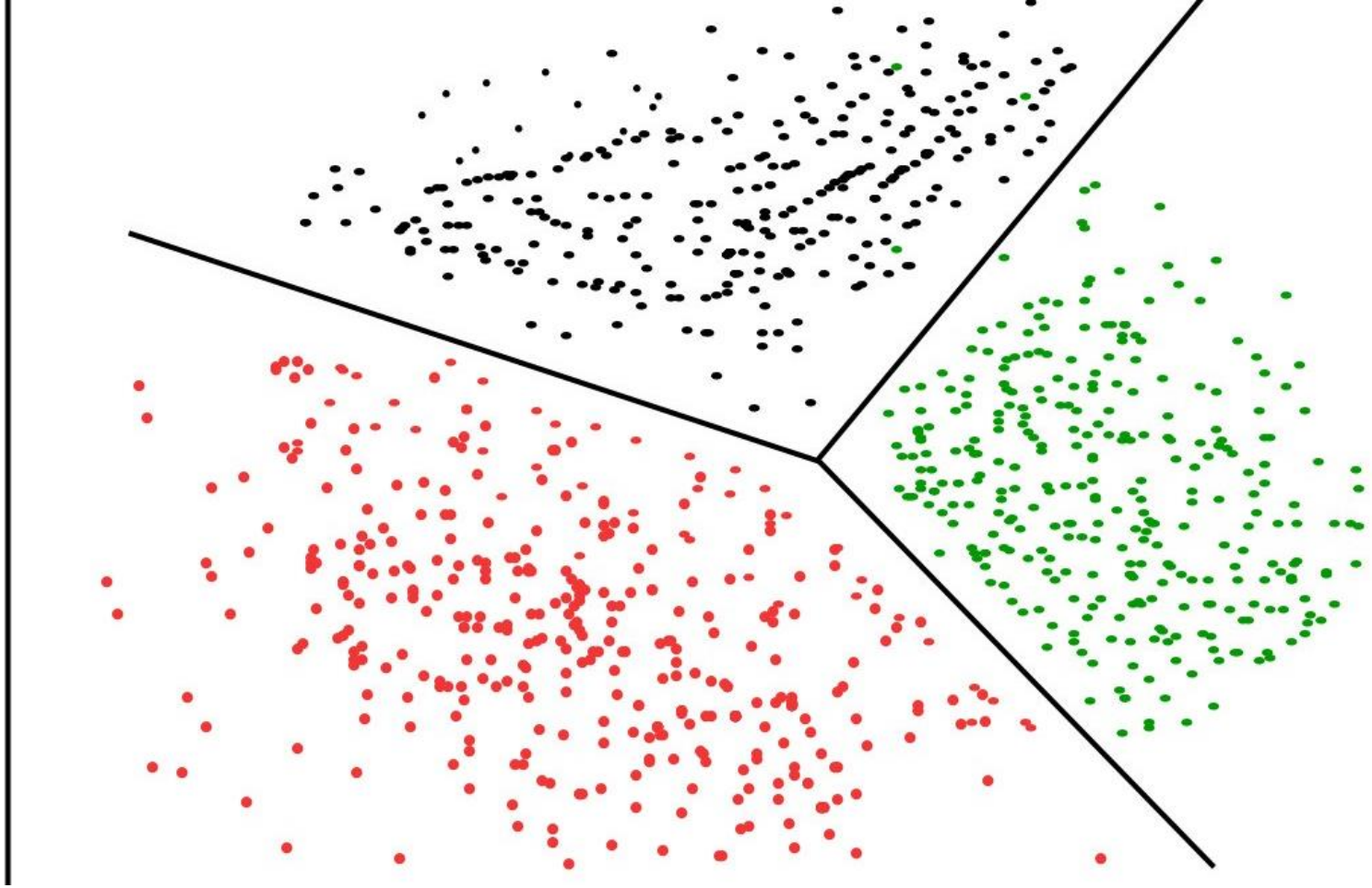
4è Partie: Apprentissage non supervisé

Dr. Pegdwendé Nicolas Sawadogo

sawadogonicholas44@gmail.com

PROGRAMME DE LA SÉANCE 4

1. Modèles de clustering
2. Techniques de réduction de dimensions
3. Apprentissage non supervisé sous Python



MODÈLES DE CLUSTERING | SECTION 1

NOTION D'APPRENTISSAGE NON SUPERVISÉ

- On parle d'apprentissage non supervisé en absence de variable cible
  - L'objectif est simplement de regrouper les individus en fonction de leurs caractéristiques
  - On cherche à créer des groupes, des clusters
  - La machine n'a pas d'échantillon d'entraînement, d'où l'appellation « non supervisé »

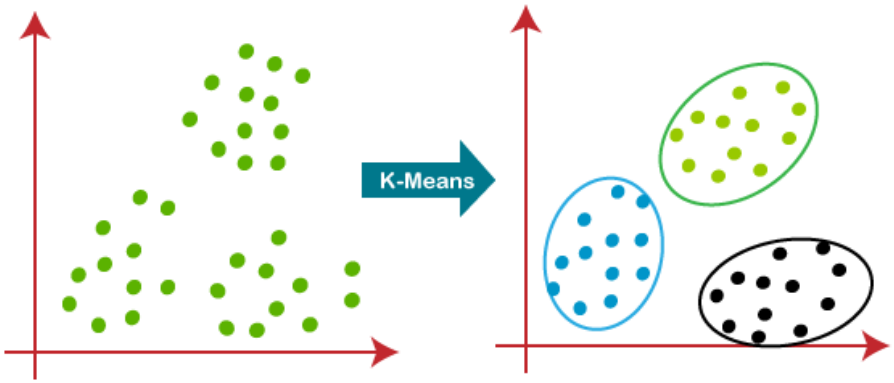
- Exemples
  - K-means (K-moyennes)
  - Classification Ascendante Hiérarchique (CAH)
  - Density-Based Clustering of Applications with Noise (DBSCAN)

MODÈLES DE CLUSTERING

K-Means ou K-moyennes

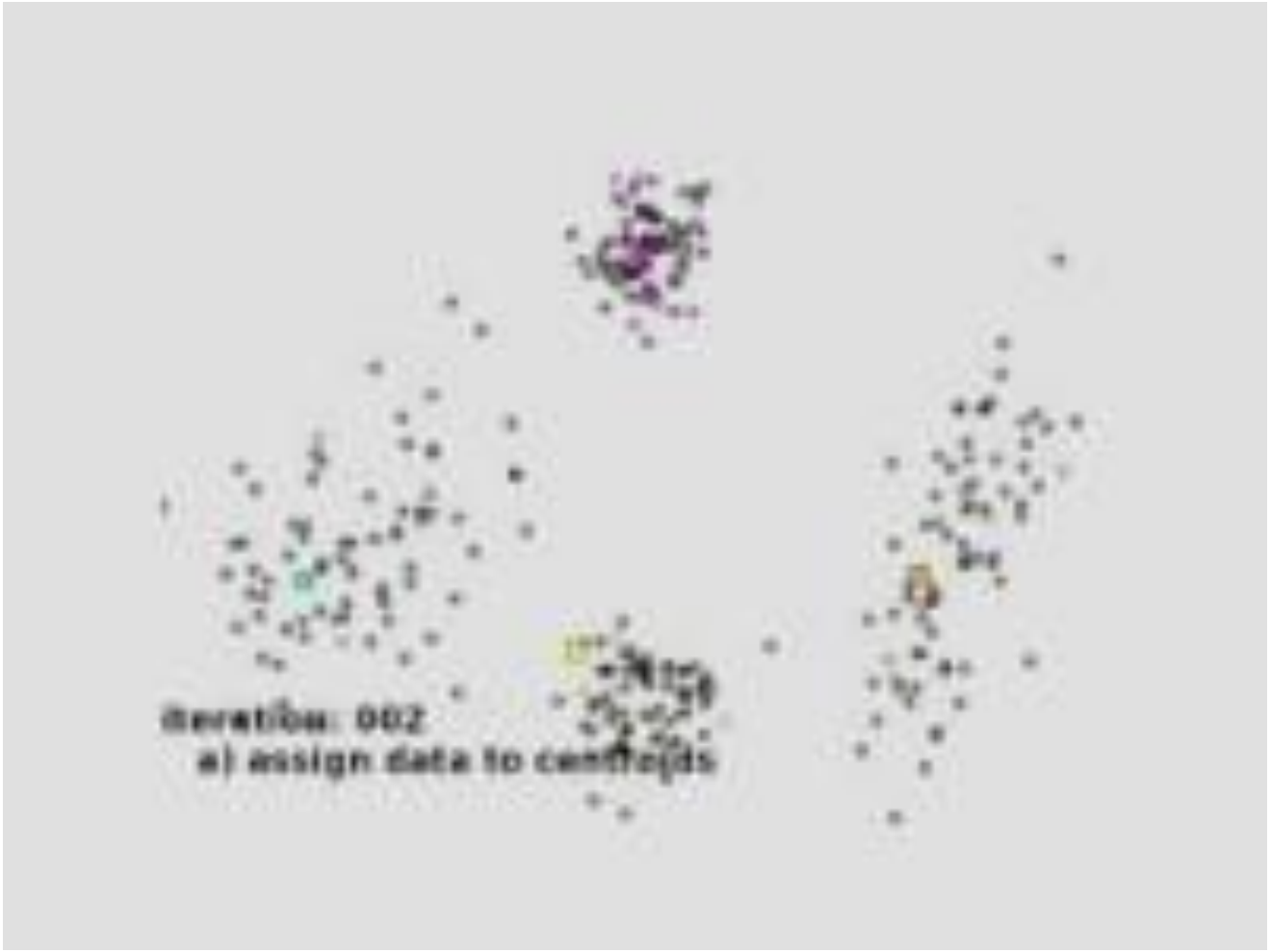
- Principe

1. Tirage au hasard de K centres de classes parmi les individus
2. Calculer les distances entre chaque individu et les K centres de classes
3. Associer chaque individu au centre de classe le plus proche
4. Recalculer les centres de classes en faisant la moyenne
5. Répéter à partir de l'étape 2, jusqu'à ce que ça converge



MODÈLES DE CLUSTERING

K-Means ou K-moyennes : démo



MODÈLES DE CLUSTERING

K-Means ou K-moyennes : Choix du K

- Par expérience
  - On se base sur une connaissance ou une vision métier
  - On sait par exemple qu'il y'a deux grandes catégories de supporters: les abonnés, et les « occasionnels »
- Par la visualisation
  - En visualisant la distribution des données, on aperçoit les clusters potentiels
- Méthode du coude
  - Rechercher l'apparition d'un coude dans l'évolution des inerties intra-classes

MODÈLES DE CLUSTERING

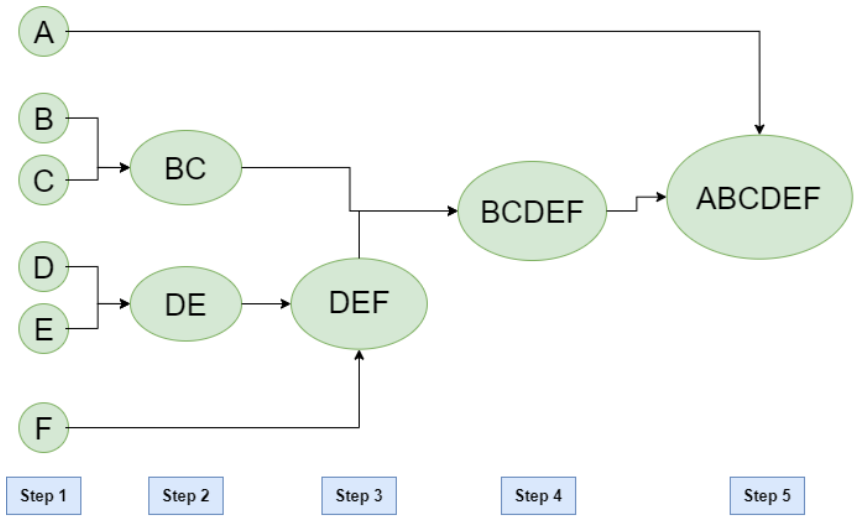
K-Means ou K-moyennes

- Avantages
  - Modèle simple à conceptualiser
  - L'algorithme converge
- Inconvénients
  - Il faut choisir un nombre de classes
  - Les résultats varient parfois selon l'initialisation
- Exercice
  - Réaliser une classification automatique en deux classes avec les données suivantes:
  - X1 = 0; X2 = 2; X3 = 6; X4 = 11

MODÈLES DE CLUSTERING

Classification Ascendante Hiérarchique

- Principe
  1. Au début, chaque individu constitue une classe
  2. Calculer les distances entre les classes deux à deux
  3. Agréger les deux classes les plus proches entre elles
  4. Répéter à partir de l'étape 2, jusqu'à ce qu'il ne reste plus qu'une classe



MODÈLES DE CLUSTERING

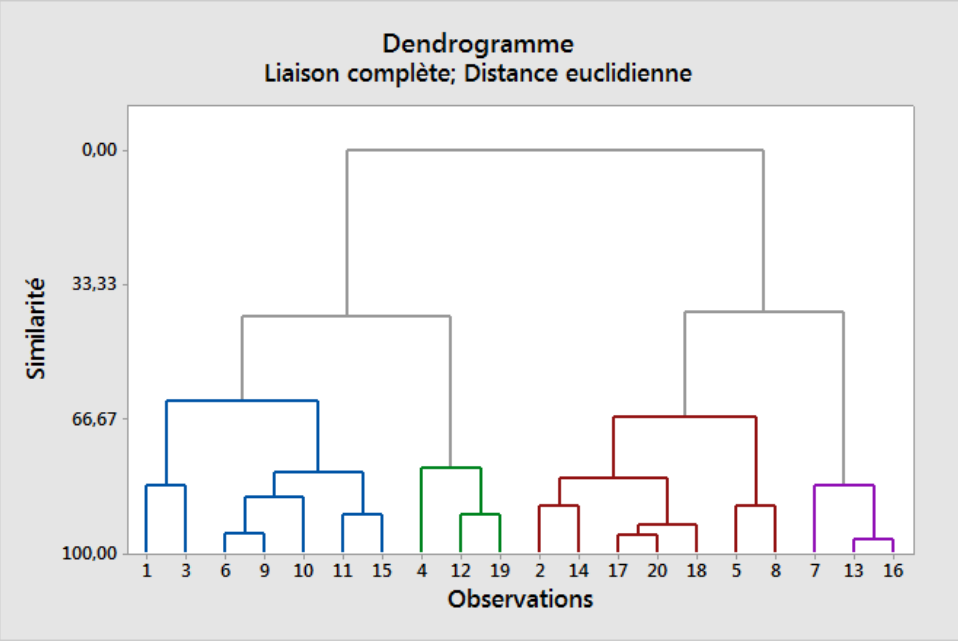
Classification Ascendante Hiérarchique

- Comment calculer la distance entre 2 classes ?
  - Saut minimum (*single linkage*): distance minimale
  - Saut maximum (*complete linkage*): distance maximale
  - Distance moyenne
  - Critère de ward: minimisation de l'inertie intra-classes
- Plusieurs types de distances
  - Distance euclidienne classique
  - Distance de Mahalanobis
  - Distance de Minkowski

MODÈLES DE CLUSTERING

Classification Ascendante Hiérarchique

- Choix du nombre de classes
  - Par visualisation ou expérience métier comme pour le K-Means
  - Un outil de visualisation en plus: le dendrogramme
  - Cela représente la hiérarchisation des classes

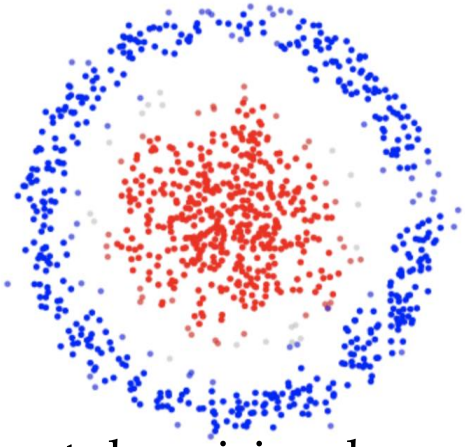


MODÈLES DE CLUSTERING

Classification Ascendante Hiérarchique

- Exercice
  - Réaliser une CAH sur les données suivantes à l'aide des critères du saut minimum et du saut maximum
  - X1 = 0; X2 = 2; X3 = 6; X4 = 11
  - Représenter les dendrogrammes
  - Comparer avec le KMeans

MODÈLES DE CLUSTERING

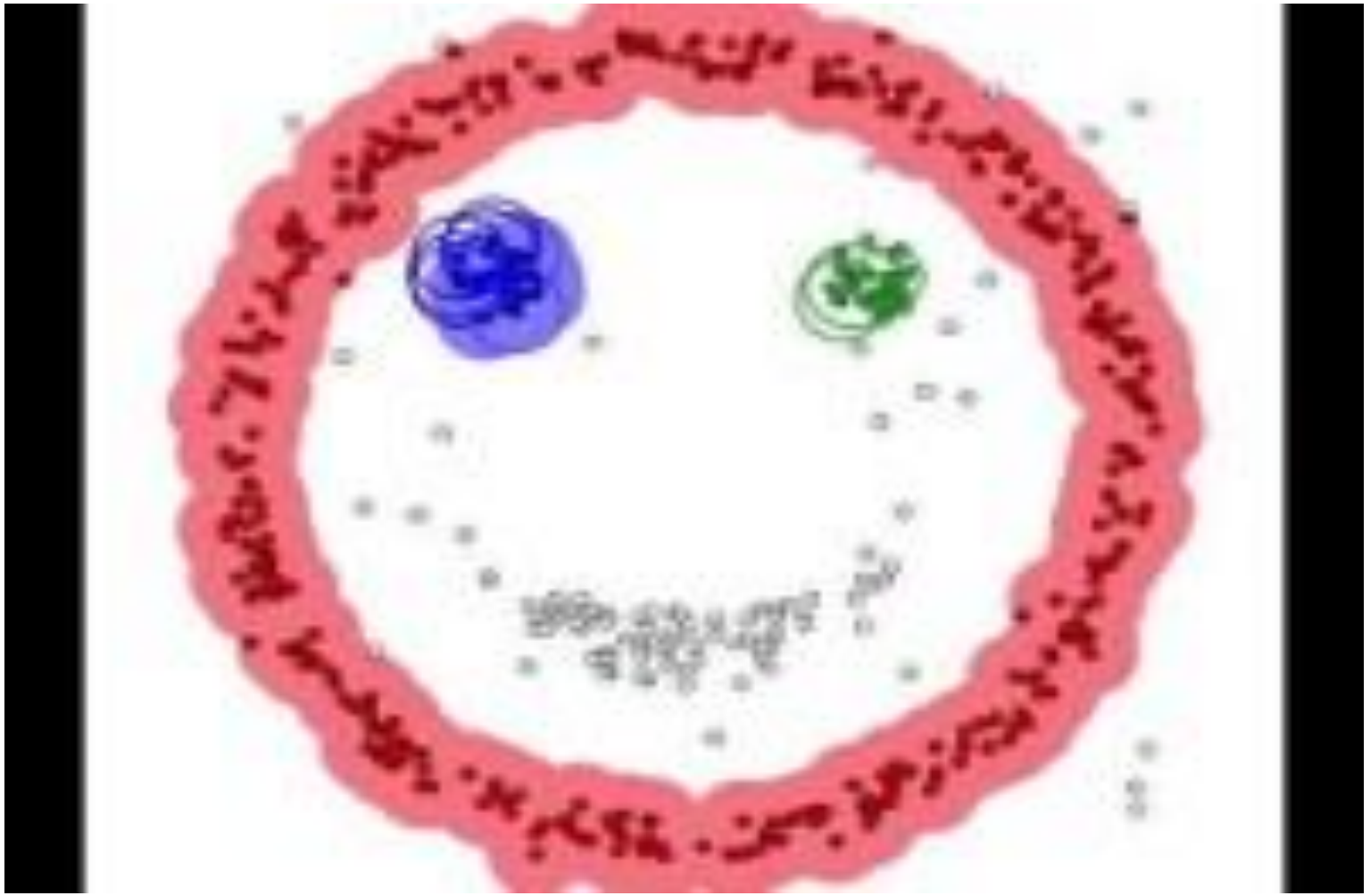


DBSCAN: Principe

- Concepts
  - Point central: un individu ayant suffisamment de voisins dans un rayon donné
  - Point frontière: n'est pas un point central mais apparaît dans le voisinage d'un point central
  - Point bruit/aberrant: ni central, ni frontière
- Paramètres
  - Rayon de voisinage
  - Nombre minimal de voisins pour être un point dense
- Algorithme
  1. Partir d'un premier point dense et lui associer tous ses voisins
  2. Pour tout point dense classifié, associer à la même classe tous ses voisins, et ainsi de suite
  3. Répéter à partir de l'étape 1, jusqu'à épuiser les points denses

MODÈLES DE CLUSTERING

DBSCAN: Démo



MODÈLES DE CLUSTERING

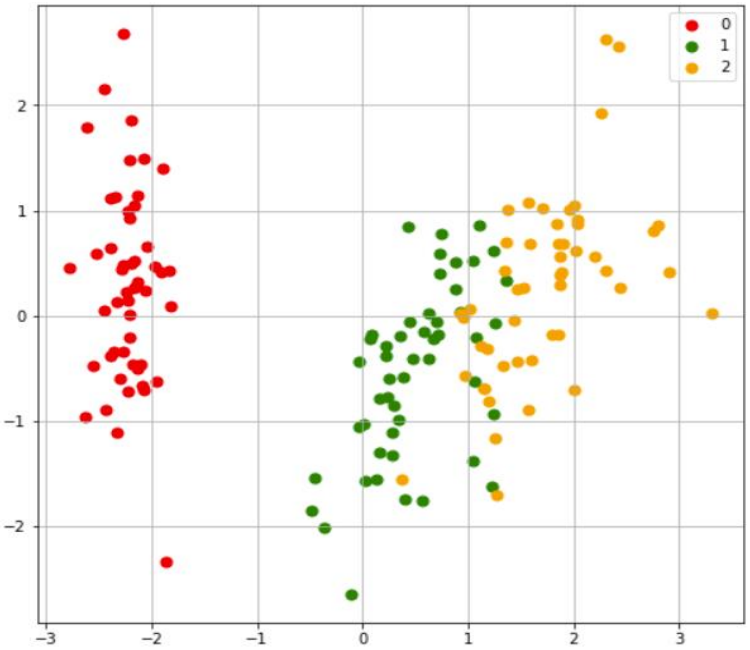
DBSCAN

- Avantages
  - Algorithme simple à conceptualiser
  - Capable de faire fi du bruit
  - Reconnaît des clusters imbriquées
- Inconvénients
  - Incapable de gérer des clusters de densité variables
  - Impossible de définir un nombre de clusters

EVALUATION D'UN MODÈLE DE CLUSTERING

Evaluation visuelle

- Réaliser une ACP
  - Projeter les individus sur le plan factoriel
  - Afficher les données en 2D, en définissant des couleurs en fonction des clusters
  - Comparer les colorations à la distribution des données



EVALUATION D'UN MODÈLE DE CLUSTERING

Indice de Rand

- Permet de comparer deux partitions
  - Fait abstraction des différences entre les noms des clusters
  - Comparer un clustering à la vérité terrain, si connue
- Formule

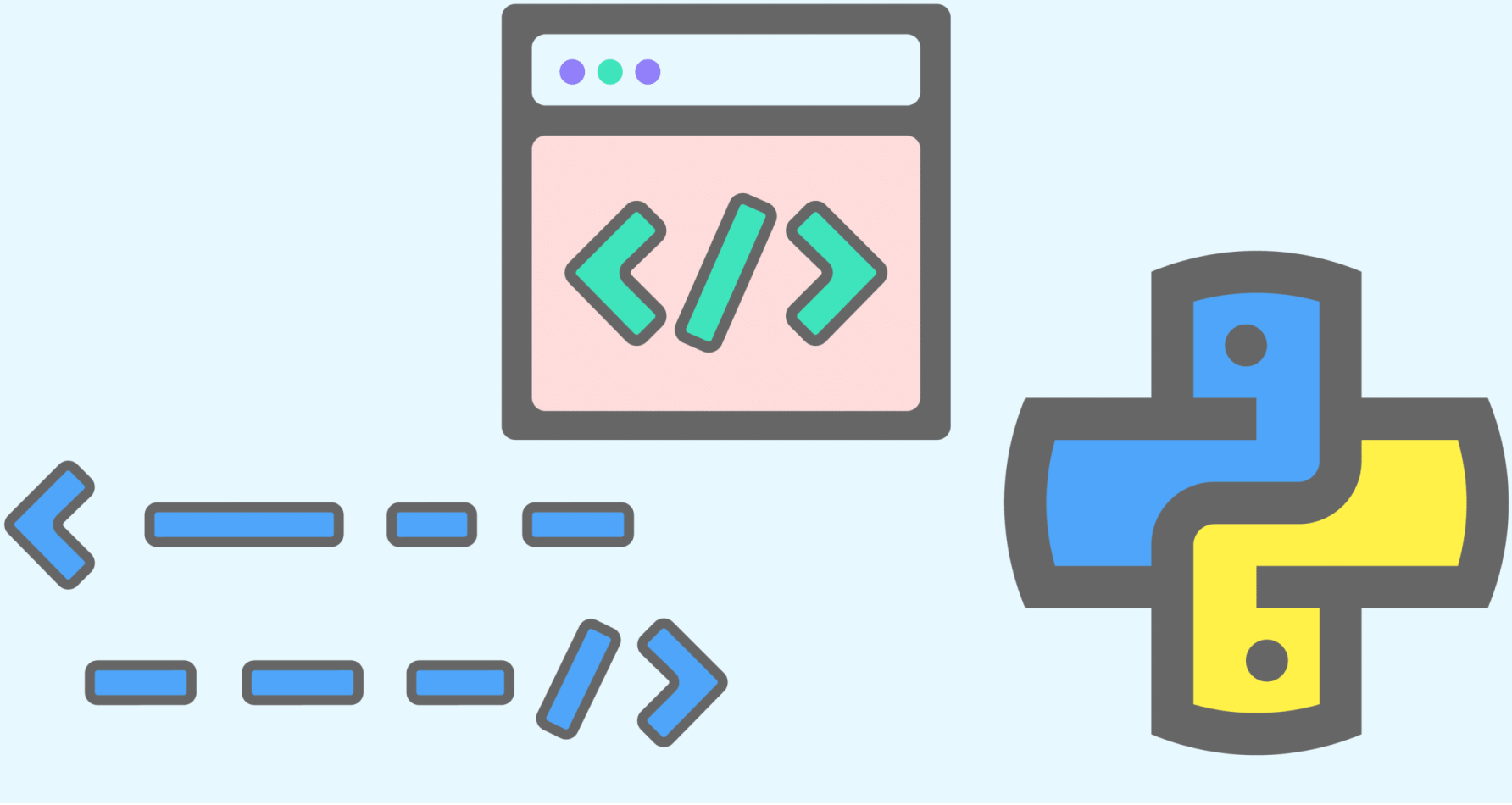
$$\mathbf{Z}_1 = (Z_{11}, \ldots, Z_{1n}) \text{ et } \mathbf{Z}_2 = (Z_{21}, \ldots, Z_{2n})$$

$$R = \frac{a+d}{a+b+c+d} = \frac{a+d}{\binom{2}{n}} \in [0,1]$$

  - a: nombre de paires dans une même classe dans Z1 et dans Z2
  - b: nombre de paires dans une même classe dans Z1 mais séparées dans Z2
  - c: nombre de paires séparées dans Z1 mais dans une même classe dans Z2
  - d: nombre de paires séparées dans Z1 et séparées dans Z2

EVALUATION D'UN MODÈLE DE CLUSTERING

- Exercice
  - Deux méthodes de clustering ont conduit aux partitions suivantes
  - Z1 = {1, 1, 2, 2, 2}; Z2 = {1, 2, 2, 1, 2}
  - Calculer l'indice de Rand
- Correction
  - a = 1; b=3 ; $C_5^2 = \frac{5!}{(2!(5-2)!)} = \frac{120}{12} = 10$
  - R = $\frac{3+1}{10} = \frac{4}{10} = 0.40$



CLUSTERING ET ACP SOUS PYTHON

SECTION 2

ANALYSE EN COMPOSANTES PRINCIPALES

- Chargement et instanciation du modèle

```
from sklearn.decomposition import PCA
acp = PCA(n_components=2)
X_projected = acp.fit_transform(X)
```

- Sorties

```
acp.explained_variance_ #var. expliquée
acp.explained_variance_ratio_ #% de var. expliquée
```

K-MEANS

- Chargement et instanciation du modèle

```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=4)
kmeans.fit(X)
```

- Sorties

```
kmeans.labels_ #affichage des clusters
kmeans.cluster_centers_ #coord. des centres de classes
kmeans.inertia_ #inertie totale
```

CLASSIFICATION ASCENDANTE HIERARCHIQUE

- Chargement et instanciation du modèle
  - Création de la matrice des liens

```
from scipy.cluster.hierarchy import dendrogram, linkage
Z = linkage(X) #Matrice des liens
```

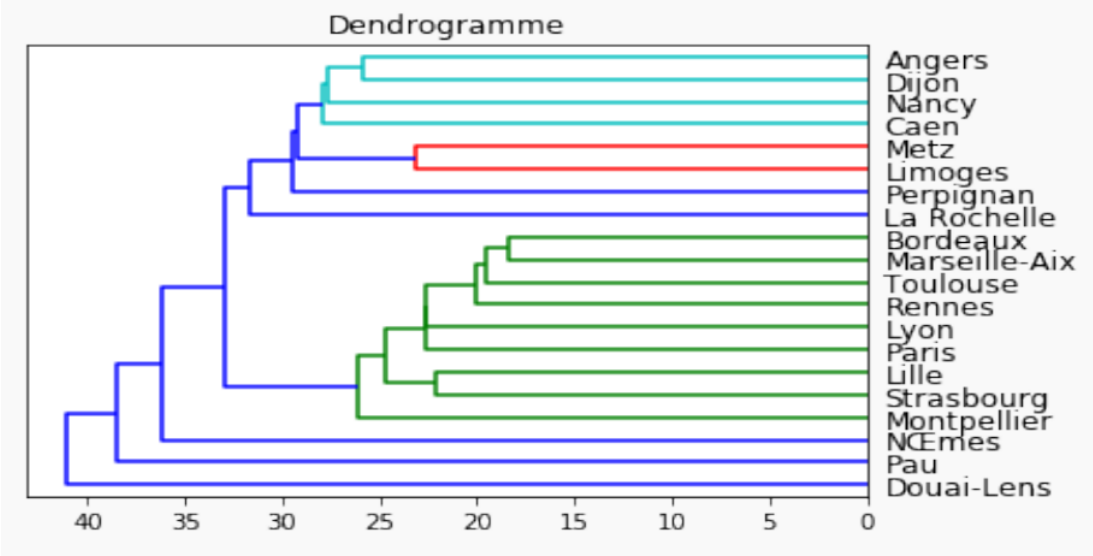
  - Constitution des groupes

```
groupes_cah = fcluster(Z,t=28,criterion='distance')
```

CLASSIFICATION ASCENDANTE HIERARCHIQUE

- Affichage du dendrogramme

```
dendrogram(Z,labels=df.Villes[selection],
orientation='left',color_threshold=28) #découpage à 28
```



EVALUATION DE CLUSTERING ET VISUALISATION

- Indice de Rand

```
from sklearn.metrics.cluster import adjusted_rand_score

adjusted_rand_score(cluster_kmeans, iris.target) #0.73
```

- Graphiques dans Python

```
import matplotlib.pyplot as plt
plt.plot(x, y, ...) // plt.scatter(x, y, ...)
plt.show() // plt.savefig('image1.png')
```

QUELQUES RESSOURCES

- https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.adjusted_rand_score
- https://openclassrooms.com/fr/courses/4452741-decouvrez-les-librairies-python-pour-la-data-science/4740942-maitrisez-les-possibilites-offertes-par-matplotlib