**TP5 : CUSTERING (K-means)**

Donn**é**es : Mall\_Customers.csv

**# Importer les librairies**

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

**# Importer le dataset**

dataset = pd.read\_csv('Mall\_Customers.csv')

X = dataset.iloc[:, [3, 4]].values

**# Utiliser la mÃ©thode elbow pour trouver le nombre optimal de clusters**

from sklearn.cluster import KMeans

wcss = []

for i in range(1, 11):

kmeans = KMeans(n\_clusters = i, init = 'k-means++', random\_state = 0)

kmeans.fit(X)

wcss.append(kmeans.inertia\_)

plt.plot(range(1, 11), wcss)

plt.title('La mÃ©thode Elbow')

plt.xlabel('Nombre de clusters')

plt.ylabel('WCSS')

plt.show()

**# Construction du modÃ¨le**

from sklearn.cluster import KMeans

kmeans = KMeans(n\_clusters = 5, init = 'k-means++', random\_state = 0)

y\_kmeans = kmeans.fit\_predict(X)

**# Visualiser les rÃ©sultats**

plt.scatter(X[y\_kmeans == 1, 0], X[y\_kmeans == 1, 1], c = 'red', label = 'Cluster 1')

plt.scatter(X[y\_kmeans == 2, 0], X[y\_kmeans == 2, 1], c = 'blue', label = 'Cluster 2')

plt.scatter(X[y\_kmeans == 3, 0], X[y\_kmeans == 3, 1], c = 'green', label = 'Cluster 3')

plt.scatter(X[y\_kmeans == 4, 0], X[y\_kmeans == 4, 1], c = 'cyan', label = 'Cluster 4')

plt.scatter(X[y\_kmeans == 0, 0], X[y\_kmeans == 0, 1], c = 'magenta', label = 'Cluster 5')

plt.title('Clusters de clients')

plt.xlabel('Salaire annuel')

plt.ylabel('Spending Score')

plt.legend()

**# Interpreter les resultats obtenus**