Université de MOULOUD MAMMERI de Tizi Ouzou
Faculté de Génie électrique et de l'informatique
Département informatique

DATA MINING

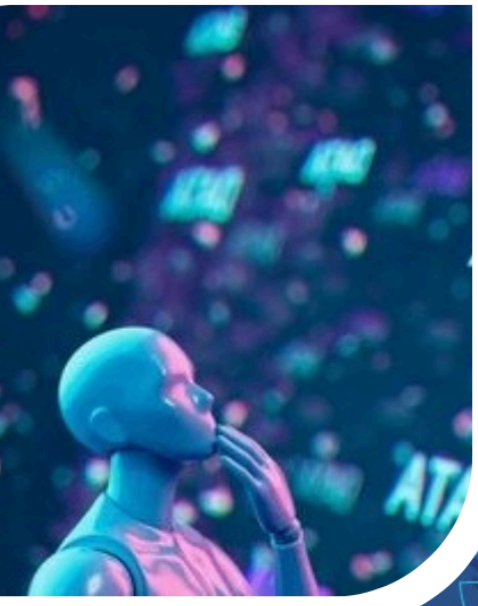
MINI PROJET 1:

Application du Clustering K-Means pour l'Analyse des Loisirs et Intérêts

DECEMBER 2024

Prepared For
- Mme AOUDJIT
- Mme OUDJOUDI

Prepared By
- Anis BENINI



Sommaire

I. Introduction

1. Contexte du projet

Ce projet vise à utiliser l'algorithme de clustering K-Means pour regrouper des personnes selon leurs loisirs et intérêts, en s'appuyant sur un dataset riche mais contenant des valeurs manquantes. L'objectif est d'identifier des groupes naturels de personnes partageant des intérêts similaires.

2. Objectifs principaux

- Charger & prétraiter les données fournies
- Implémenter l'algorithme K-Means
- Visualiser les clusters pour une meilleure interprétation des résultats

3. Technologie et langages utilisés

- Python
- Libraries : Pandas, NumPy, Matplotlib, Seaborn, Scikit-Learn, Scipy, Missingno

II. Méthodologie

1. Importation et exploration du dataset

a) Chargement des données :
Lecture du fichier CSV « **kaggle_Interests_group.csv** » Via Pandas.

b) Description des données :

- Le dataset comporte **6340 lignes** et **219 colonnes**, incluant :
  - **group** (catégorique) : identifie les groupes d'individus.
    - I → 1809
    - P → 1731
    - C → 1725
    - R → 1075
  - **grand_tot_interests** (quantitatif) : total des intérêts d'un individu.
  - **interest1** à **interest217** : variables binaires indiquant la présence d'intérêts spécifiques.
  - Des valeurs manquantes (**NaN**) à traiter.

```
Total Missing Values: 1139339
Percentage of Missing Values: 82.06%

Missing Values per Column:
interest1      5347
interest2      6339
interest3      6305
interest4      6315
interest5      5542
               ...
interest213    6338
interest214    6268
interest215    1397
interest216    2282
interest217    6193
Length: 217, dtype: int64
```

2. Nettoyage & Préparation des données

➢ Gestion des valeurs manquantes et normalisation (partiellement visible) :
- Les valeurs manquantes (**NaN**) ont été remplacées par **0**.

- Before :

	group	grand_tot_interests	interest1	interest2	interest3	interest4	interest5	interest6
0	C	17	NaN	NaN	NaN	NaN	NaN	NaN
1	C	43	1.0	NaN	NaN	NaN	1.0	NaN
2	C	27	NaN	NaN	NaN	NaN	NaN	NaN
3	C	34	NaN	NaN	NaN	NaN	NaN	NaN
4	C	36	NaN	NaN	NaN	NaN	1.0	NaN

- After :

	group	grand_tot_interests	interest1	interest2	interest3	interest4	interest5	interest6
0	0	17	0.0	0.0	0.0	0.0	0.0	0.0
1	0	43	1.0	0.0	0.0	0.0	1.0	0.0
2	0	27	0.0	0.0	0.0	0.0	0.0	0.0
3	0	34	0.0	0.0	0.0	0.0	0.0	0.0
4	0	36	0.0	0.0	0.0	0.0	1.0	0.0

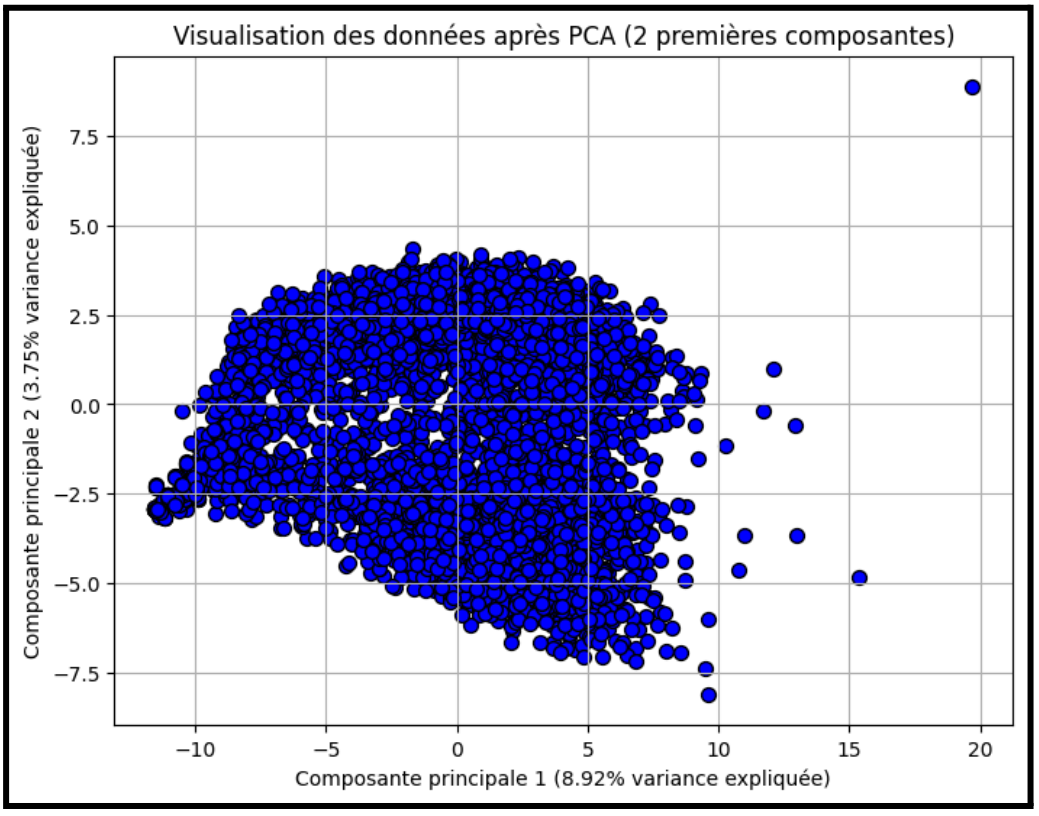
➢ **Standardisation** : C'est une méthode de transformation des données qui consiste à ajuster les valeurs de chaque caractéristique (ou variable) pour qu'elles aient une moyenne de **0** et un écart-type de **1**.
Cela est utile pour centrer et normaliser les caractéristiques avant d'appliquer des algorithmes d'apprentissage automatique.

```
array([[-1.27325087, -1.29142213, -0.4309427 , ..., -1.88017358,
        -1.33305999, -0.15406647],
       [-1.27325087,  0.36161424,  2.3204941 , ...,  0.53089044,
         0.74912783, -0.15406647],
       [-1.27325087, -0.65563891, -0.4309427 , ...,  0.53089044,
         0.74912783, -0.15406647],
       ...,
       [-0.32310839,  0.67950585, -0.4309427 , ...,  0.53089044,
         0.74912783, -0.15406647],
       [-0.32310839,  0.04372263, -0.4309427 , ...,  0.53089044,
         0.74912783, -0.15406647],
       [-0.32310839,  1.82391565,  2.3204941 , ...,  0.53089044,
         0.74912783, -0.15406647]])
```

3. Réduction de la dimensionnalité

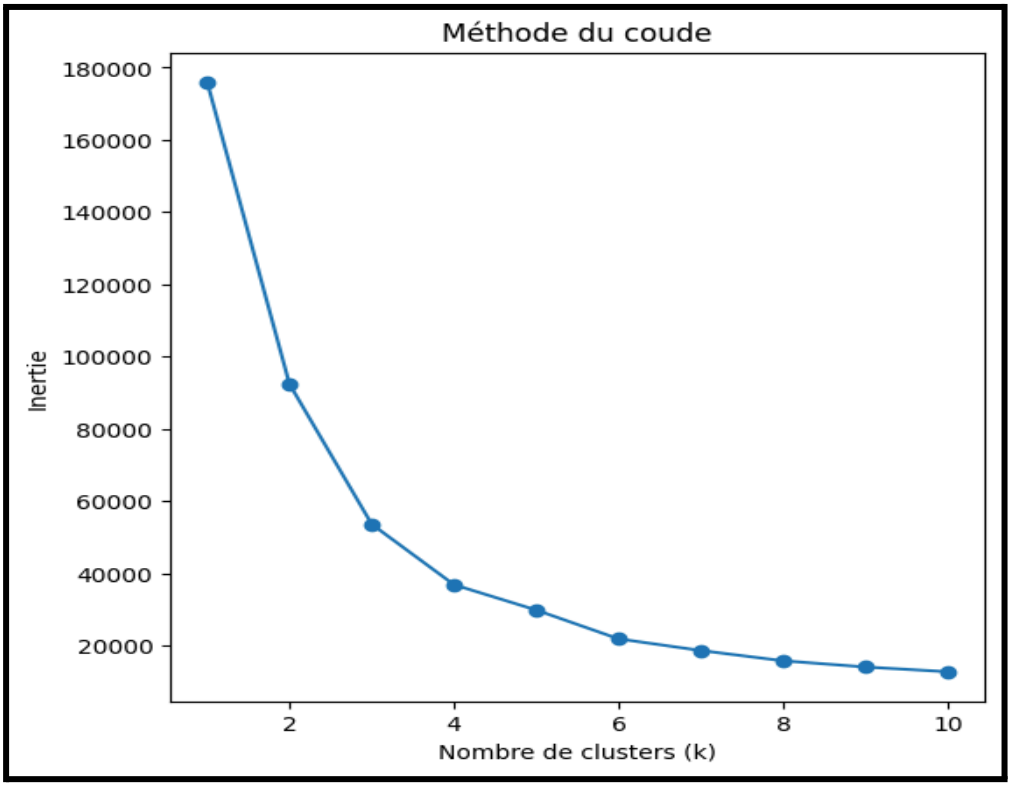
La réduction de dimensionnalité permet de simplifier un dataset en diminuant le nombre de variables (ou dimensions) tout en conservant l'essentiel de l'information.
Cela facilite la visualisation, réduit le bruit, améliore les performances des algorithmes (comme K-means) et atténue les effets de la malédiction de la dimensionnalité.

- **Techniques employées :**
  - L'Analyse en Composantes Principales (PCA) a été utilisée pour réduire les dimensions tout en conservant **95 %** de la variance des données.
- **Pourquoi on a utilisé PCA ?**
  - La PCA permet d'atténuer la malédiction de la dimensionnalité et facilite la visualisation des clusters.
  - Ajouter une visualisation de la variance expliquée par chaque composante principale (à l'aide d'un screen plot) serait utile pour évaluer la pertinence de cette réduction.
- Résultats :
  - Variance expliquée cumulée : **0.9508420997241283**
  - Nombre de composantes nécessaires pour expliquer **95%** de la variance : **183**



4. Choix du nombre de clusters

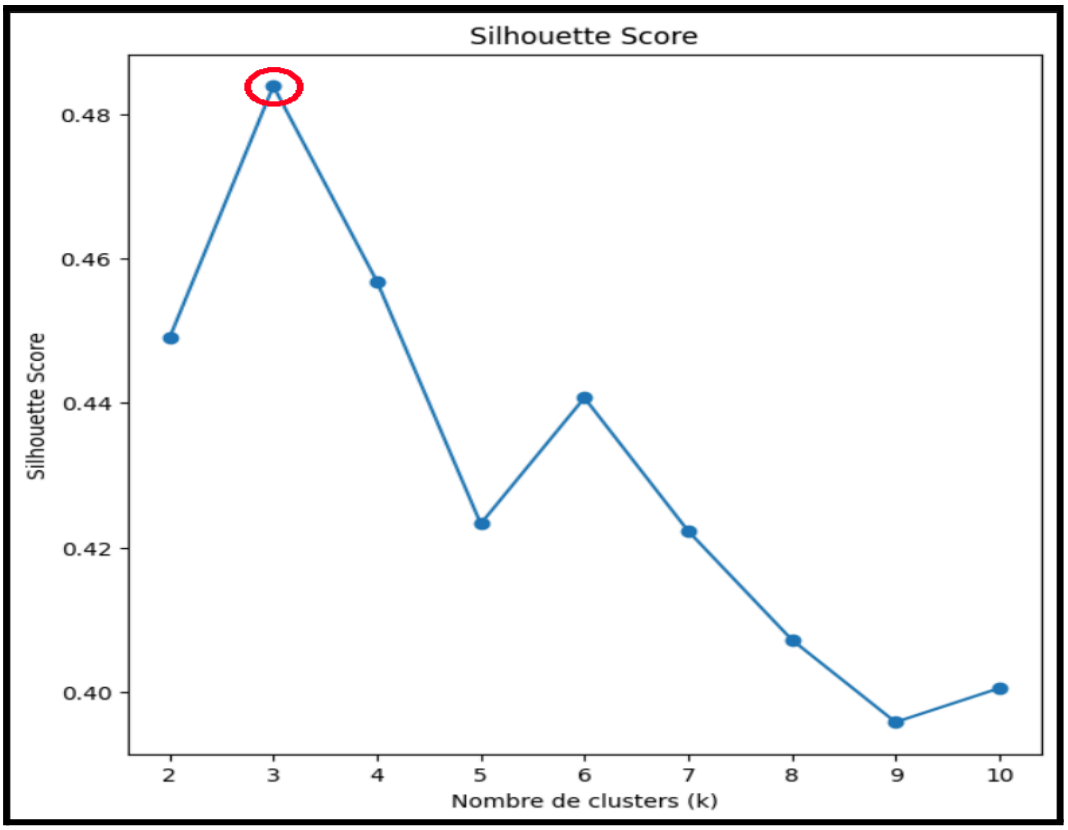
- **Méthodologie :**
  - La méthode du coude a été appliquée en calculant l'inertie pour un nombre de clusters variant de 1 à 10.
  - La méthode Silhouette Score : Ce score mesure à quel point les points d'un cluster sont similaires entre eux tout en étant bien séparés des autres clusters, Test de **k** dans la plage [2, 10].
- **Visualisation :**
  - Méthode du coude :



-

➔ Le choix final du nombre de clusters (évalué à 3) semble cohérent avec la courbe. Une analyse complémentaire avec des méthodes comme le coefficient de silhouette pourrait renforcer cette décision

- La méthode Silhouette Score :



➔ Le Silhouette Score varie en fonction de **K**, et le point où le score est le plus élevé est considéré comme le nombre optimal de clusters

---

5. Application de K-Means

Après la conversion des données en un format compatible avec l'algorithme K-Means, on va réaliser une implémentation de K-Means :

- Steps :
  - ❖ Utilisation de l'algorithme de clustering de **scikit-learn**.
  - ❖ Définition du nombre de clusters (**HyperParamètres K**).

- **Résultats :**
  - K-Means a été appliqué avec **3** clusters, et les labels ont été ajoutés au dataset.

	Cluster	group	grand_tot_interests	interest1	interest2	interest3	interest4	interest5	interest6	interest7	...	interest208	interest209
0	1	0	17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
1	0	0	43	1.0	0.0	0.0	0.0	1.0	0.0	0.0	...	0.0	0.0
2	2	0	27	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
3	2	0	34	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
4	2	0	36	0.0	0.0	0.0	0.0	1.0	0.0	0.0	...	0.0	0.0

5 rows × 220 columns

6. Visualisation des clusters

A) La distribution des tailles des clusters

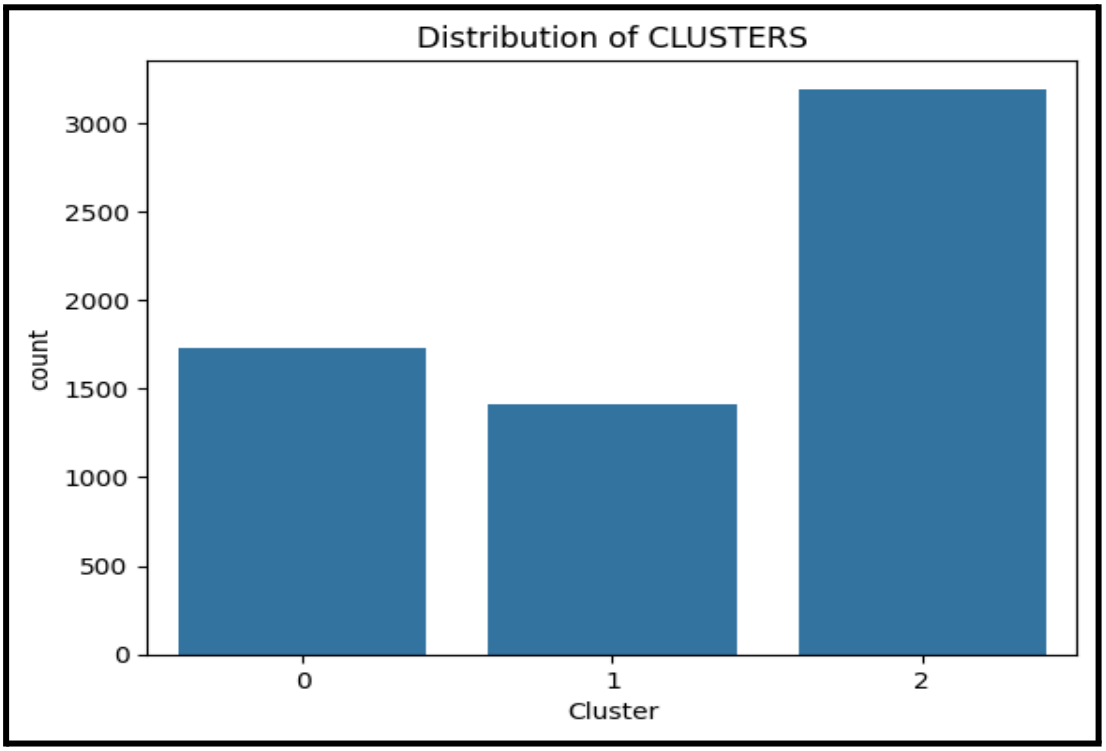
Objectif :

Comprendre la répartition des données entre les clusters générés par l'algorithme K-Means.
Cette visualisation permet d'évaluer si certains clusters contiennent significativement plus de données que d'autres, ce qui pourrait indiquer un déséquilibre ou une segmentation efficace.

Méthodologie :

- Utilisation de **countplo**t de la bibliothèque **Seaborn** pour représenter le nombre d'échantillons par cluster.
- Chaque barre correspond au nombre d'observations dans un cluster spécifique, défini par la variable **Cluster** dans le dataset traité.

Résultat :



B) La disposition spatiale des clusters

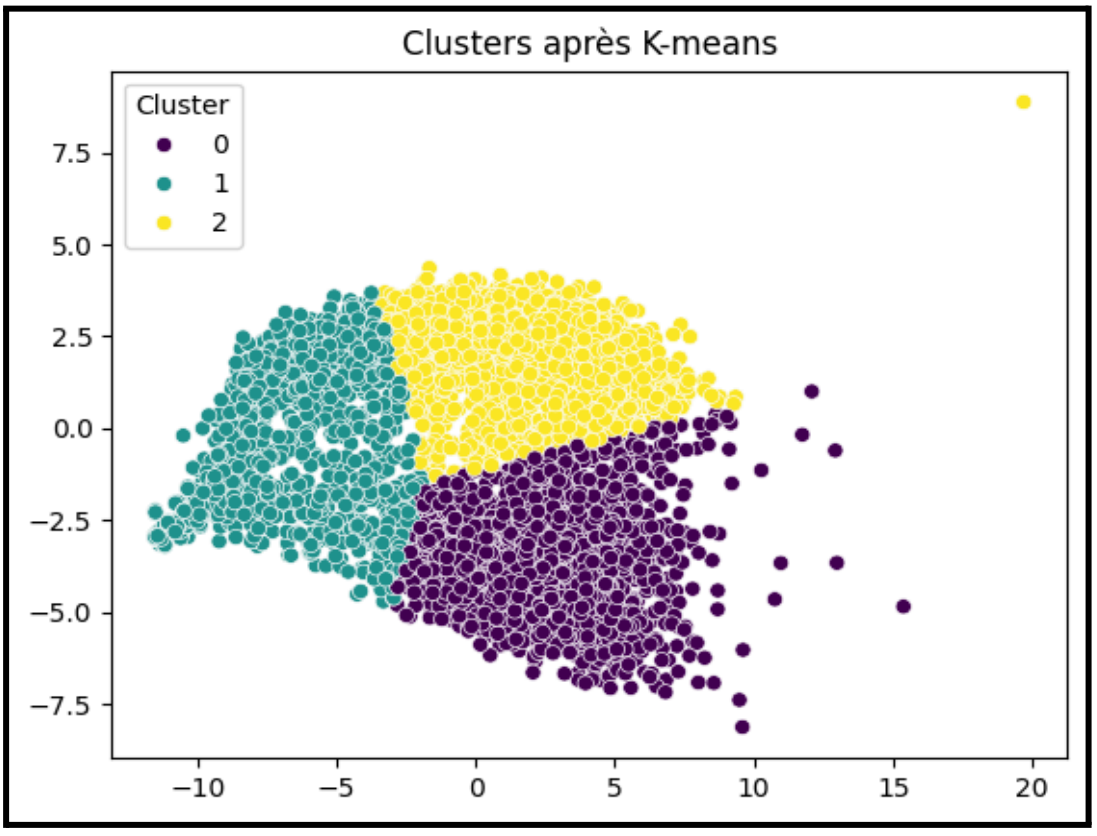
Objectif :

Représenter graphiquement les clusters obtenus après l'application de l'algorithme K-Means pour vérifier leur séparation visuelle.
Cette visualisation en deux dimensions permet de mieux comprendre la cohésion des clusters et leur séparation.

Méthodologie :

- Utilisation de scatterplot (**Seaborn**) pour tracer les données réduites à deux dimensions (via PCA ou une autre méthode).
- Les points sont colorés selon leur appartenance à un cluster, grâce à l'argument hue.
- La palette "**viridis**" offre une distinction claire entre les clusters.

Résultat :



---

7. Discussion

- **Points forts :**
  - Pipeline clair et structuration des étapes.
  - Intégration des techniques standards (PCA, méthode du coude).
- **Limitations :**
  - L'imputation par 0 pourrait biaiser les résultats.
  - Une analyse qualitative des clusters (par exemple, en identifiant des tendances spécifiques au sein de chaque cluster) est absente.

III. Conclusion

Ce projet montre une mise en œuvre réussie de **K-Means** pour regrouper des individus selon leurs intérêts en clusters selon leurs similarités. Il a permis de révéler des structures intéressantes dans les données et s'est montré rapide et efficace. Cependant, sa sensibilité au choix du nombre de clusters (**K**) et des points initiaux souligne l'importance d'une analyse préalable et de l'évaluation des résultats à l'aide de métriques adaptées, comme le coefficient de silhouette.