TP : Clustering

Realise par :

Ziad Ben Saada

Encadre par :

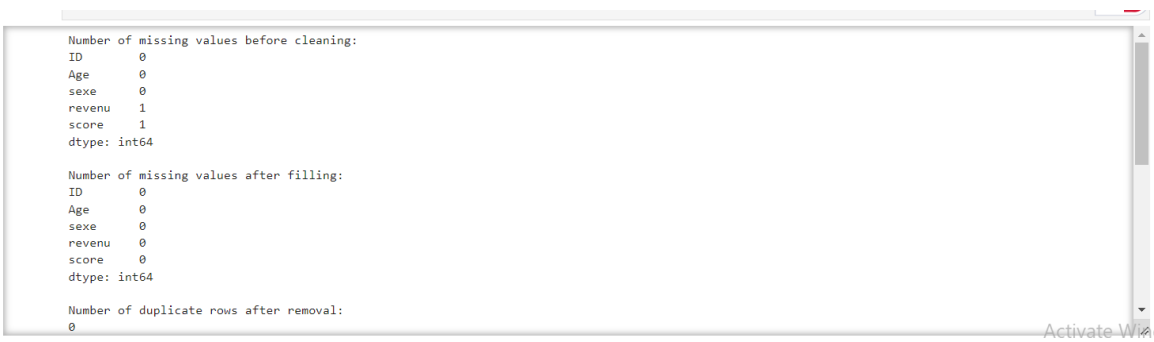
Abdelhadi FENNAN

Introduction:

Customer segmentation is a crucial task in marketing and business analytics, aiming to divide a customer base into groups of individuals with similar characteristics or behaviors. One popular technique for customer segmentation is K-Means clustering, an unsupervised learning algorithm used to partition data into clusters based on similarities. In this TP (Travaux Pratiques), we will explore how to perform customer segmentation using K-Means clustering in Python. The goal is to identify distinct groups of customers based on their age, annual income, and spending score.

1- Data Loading & Cleaning

```
# Load data (replace 'donnees_clients.csv' with your actual file path)
data = pd.read_csv("donnees_clients.csv")

# Check for missing values
print("Number of missing values before cleaning:")
print(data.isnull().sum())

# Calculate mean for 'revenu' and 'score' columns
mean_revenu = data["revenu"].mean()
mean_score = data["score"].mean()

# Fill missing values with mean values
data["revenu"].fillna(mean_revenu, inplace=True)
data["score"].fillna(mean_score, inplace=True)

# Check for missing values after filling
print("\nNumber of missing values after filling:")
print(data.isnull().sum())

# Remove duplicate rows
data.drop_duplicates(inplace=True)

# Check for duplicate rows after removal
print("\nNumber of duplicate rows after removal:")
print(data.duplicated().sum())

# Save the cleaned dataset
data.to_csv('cleaned_donne_clients.csv', index=False)
```



```
Number of missing values before cleaning:
ID        0
Age       0
sexe      0
revenu    1
score     1
dtype: int64

Number of missing values after filling:
ID        0
Age       0
sexe      0
revenu    0
score     0
dtype: int64

Number of duplicate rows after removal:
0
```

2- Data Preprocessing

```
# Drop non-numeric column
data_numeric = data.drop(columns=['sexe'])

# Normalize the data
scaler = StandardScaler()
data_scale = scaler.fit_transform(data_numeric)
```

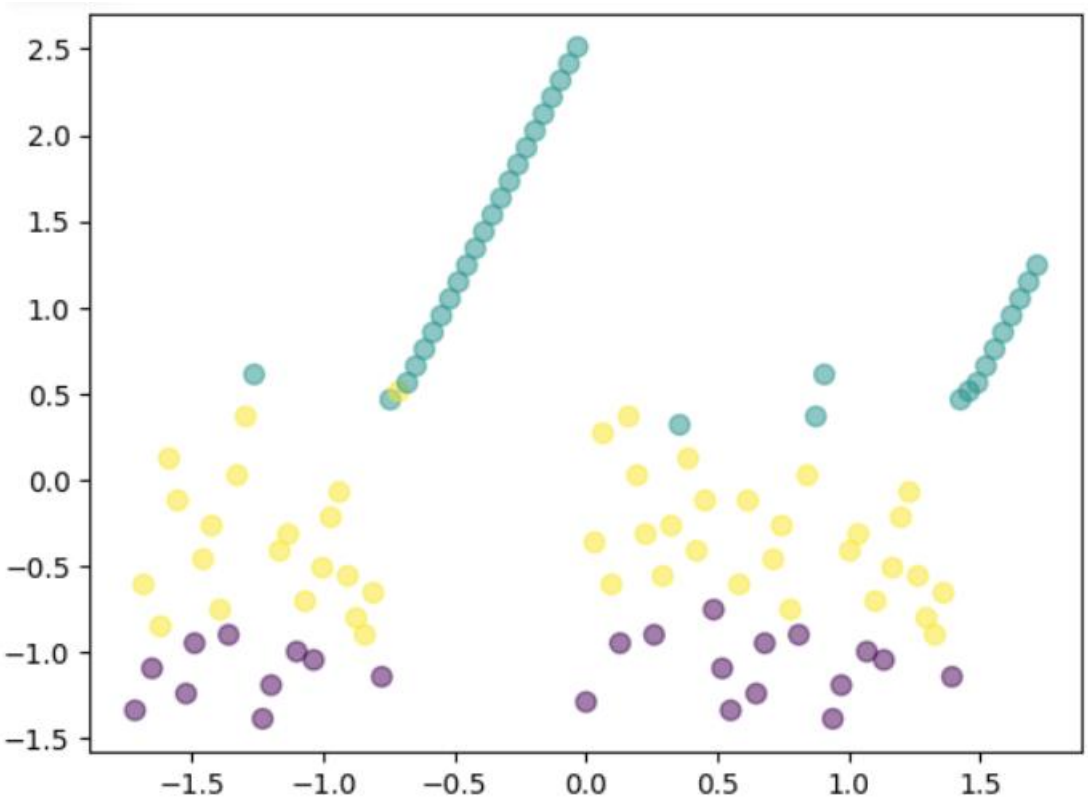
3- K-Means Clustering

```
# Define the number of clusters
n_clusters = 3

# Run k-means algorithm
kmeans = KMeans(n_clusters=n_clusters)
labels = kmeans.fit(data_scale).labels_

# Display the results
print("Labels des clusters:", labels)
print("Centroides des clusters:", kmeans.cluster_centers_)
```

```
Labels des clusters: [0 2 0 2 2 2 0 0 2 2 2 0 2 2 1 0 0 2 2 0 2 0 2 2 2 2 2 2 2 0 1 2 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 2 2 2 0 2 2 2 0 2 2 1 2 2 2 0 0 0 2 2 0
 0 2 2 2 0 2 1 1 0 0 2 2 0 2 0 2 2 2 2 2 2 2 0 1 1 1 1 1 1 1 1 1 1]
Centroides des clusters: [[-0.14569288 -1.10316498 -1.20293338 -1.56489014]
 [ 0.2320294   1.19029358  1.21344295  0.94238788]
 [-0.10332828 -0.34839594 -0.31518181  0.07726383]]
```

4- Visualization and Interpretation

```
# Visualiser les clusters
import matplotlib.pyplot as plt
plt.scatter(data_scale[:, 0], data_scale[:, 1], c=labels, s=50, alpha=0.5)
plt.show()
```



Conclusion

Customer segmentation using K-Means clustering offers valuable insights into understanding the diverse behaviors and preferences within a customer base. In this TP, we successfully applied K-Means clustering to segment customers based on their age, annual income, and spending score. By preprocessing the data, handling missing values, removing duplicates, and normalizing the features, we ensured the robustness and accuracy of the clustering process.

Through the implementation of K-Means clustering, we identified distinct customer segments, each exhibiting unique characteristics and purchasing patterns. Visualizing the clusters allowed us to gain a comprehensive understanding of the distribution of customers across different segments.

The insights derived from customer segmentation can inform various business strategies, including targeted marketing campaigns, personalized product recommendations, and tailored customer experiences. By understanding the needs and preferences of different customer segments, businesses can optimize resource allocation, enhance customer satisfaction, and drive growth and profitability.

In conclusion, customer segmentation using K-Means clustering serves as a powerful tool for businesses to gain actionable insights from their data, ultimately leading to more effective decision-making and improved business outcomes.

Lien GitHub: **https://github.com/ziadbensaada/Clustering-Using-K-means**