Bangabandhu Sheikh Mujibur Rahman Digital University

Faculty: Cyber Physical Systems

Department: IoT AND Robotics Engineering (IRE)

CourseTitle: DATA Science

Course Code: IOT 4313

Assignment: 02

Submitted To
Nurjahan Nipa
Lecturer
Department of IRE, BDU.

Submitted By
Sadat Hasnat Sabbir
ID. 1901002
Department of IRE
Session: 2019-20

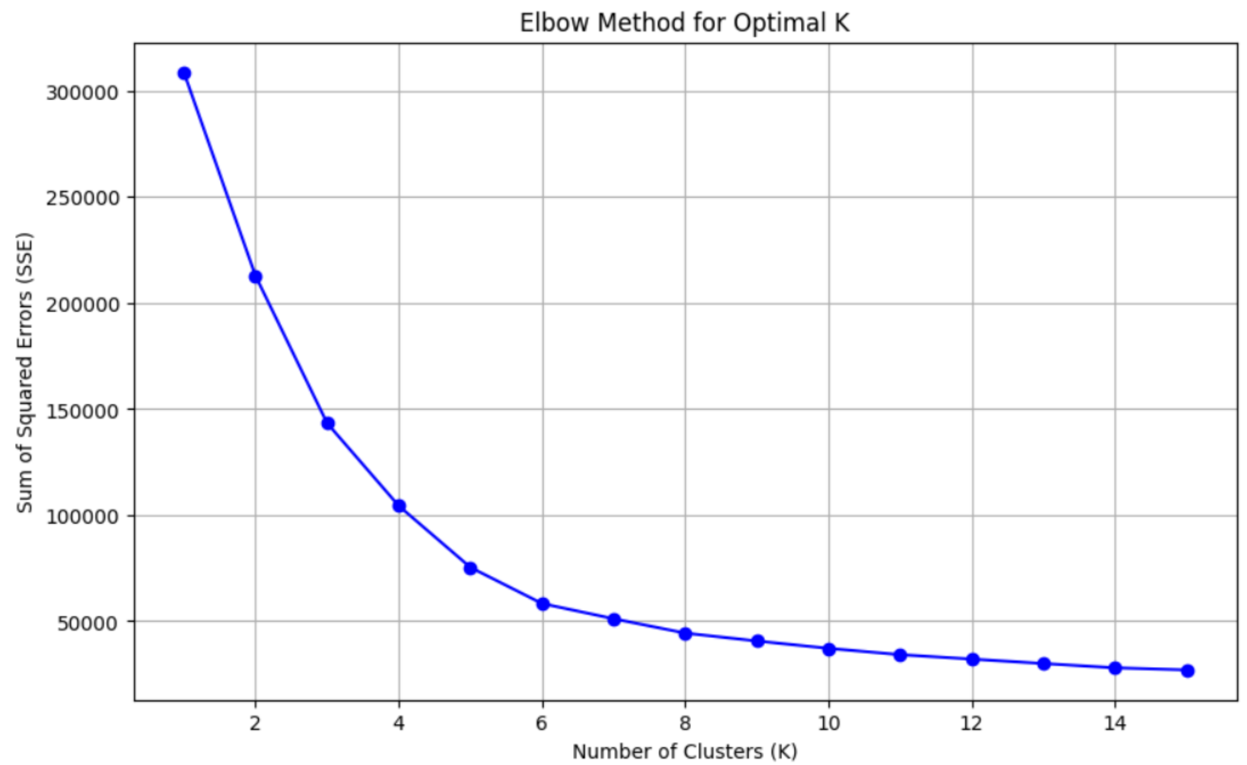
Date of Submission : 10th October 2023

K Means Clustering

K-means clustering is a popular unsupervised machine learning algorithm used for data clustering and segmentation. It partitions a dataset into K clusters, where each data point belongs to the cluster with the nearest mean (centroid). The algorithm iteratively refines cluster assignments by minimizing the sum of squared distances between data points and their respective cluster centroids. K-means is simple, efficient, and widely used in various applications, such as image compression, customer segmentation, and data analysis. However, it requires specifying the number of clusters (K) beforehand and is sensitive to the initial placement of centroids.

Task 01: Given a Mall_Customer dataset with 5 attributes. Utilize K-means clustering algorithm to identify the appropriate number of clusters. You may use any language and libraries to implement K-mean clustering algorithm. Your K-mean clustering algorithm should look for appropriate values of K at least in the range of 0 to 15 and show their corresponding sumof-squared errors (SSE)

a) Import all Necessary Libraries and essential files.
b) Load the dataset (Mall_Customer.csv)
c) Preprocess the data if necessary (e.g., encode 'Gender' to numerical values)
d) Choose a range of K values to explore (from 1 to 15)
e) List to store sum-of-squared errors for each K
f) Perform K-means clustering for each K value and compute the SSE
g) Plot the Elbow Method graph to visualize SSE for different K values
h) Determine the optimal K value based on the Elbow Method (manually)
i) Perform K-means clustering with the optimal K value
j) Now 'data' contains a new column 'Cluster' indicating the cluster assignments for each customer.

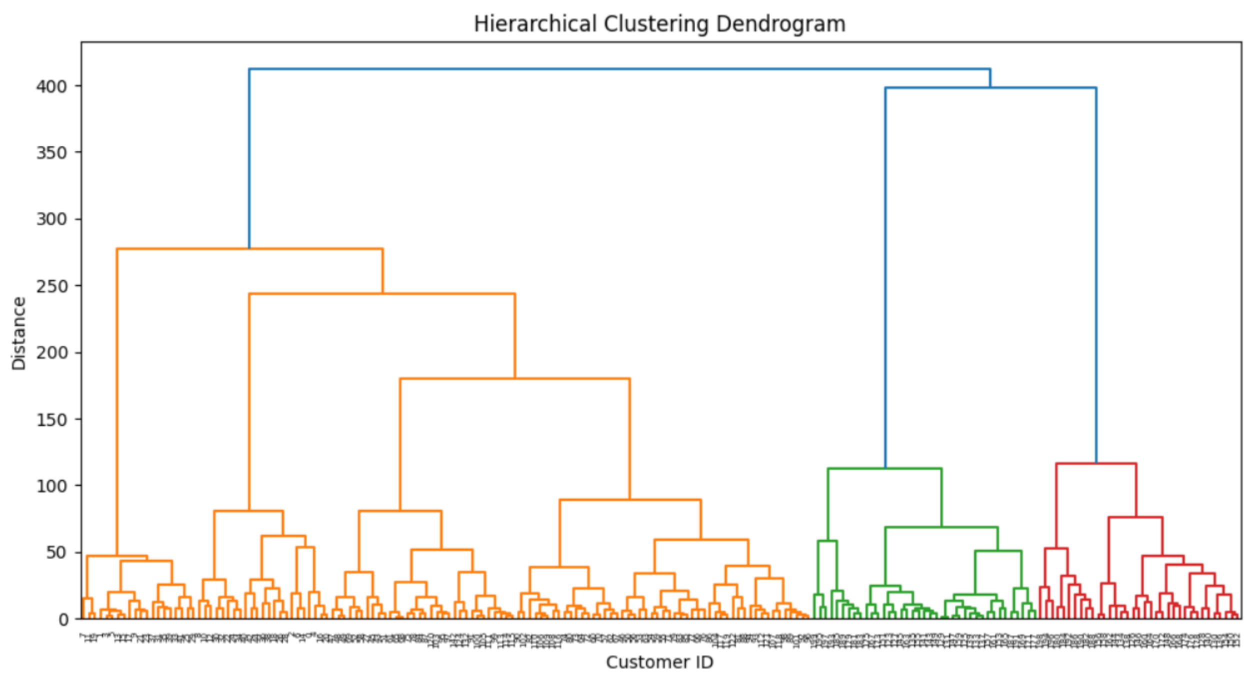


Hierarchical clustering

Hierarchical clustering is a data analysis technique that organizes data into nested clusters, forming a tree-like structure called a dendrogram. It doesn't require specifying the number of clusters beforehand and is valuable for exploring data relationships. It can be agglomerative (merging data points into clusters) or divisive (splitting a single cluster into smaller ones).

Task 02: In this part, you will apply hierarchical clustering algorithm (agglomerative or divisive) to the provided mall dataset.

a) Import all Necessary Libraries and essential files.
b) Load the dataset (Mall_Customer.csv)
c) Preprocess the data if necessary (e.g., encode 'Gender' to numerical values)
d) Choose the linkage method and distance metric ('complete', 'average', 'ward', etc.) and ('euclidean', 'manhattan', 'cosine', etc.)
e) Perform hierarchical clustering
f) Plot the dendrogram to visualize hierarchical clustering
g) Decide the number of clusters based on dendrogram visualization
h) Now 'cluster_labels' contains cluster assignments for each customer.

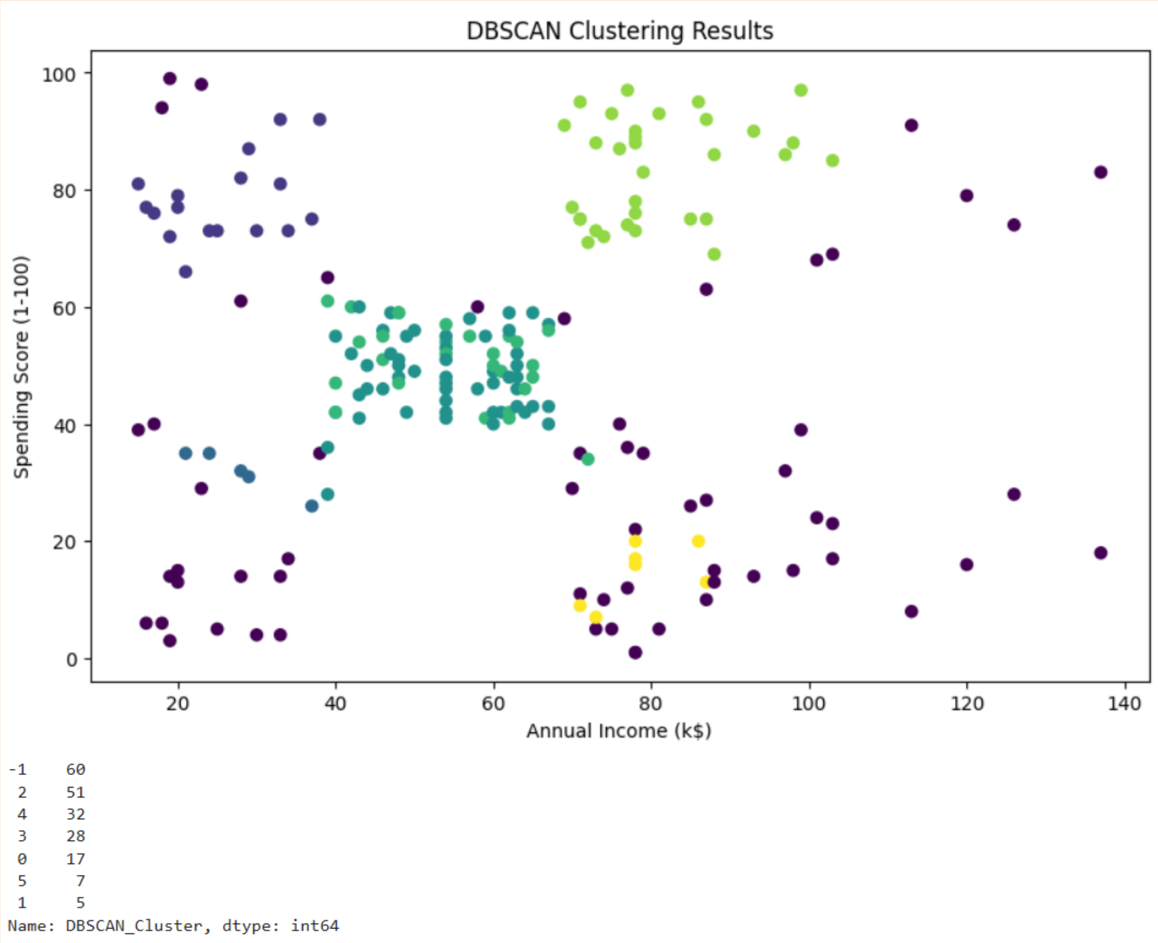


Density-based clustering

Density-based clustering is a data clustering technique that identifies clusters based on regions of high data point density. It's particularly useful for detecting clusters with irregular shapes and varying densities. The most well-known density-based clustering algorithm is DBSCAN (Density-Based Spatial Clustering of Applications with Noise).

Task 03: In this part, you will apply density-based clustering algorithm to the provided dataset.

a) Import all Necessary Libraries and essential files.
b) Load the dataset (Mall_Customer.csv)
c) Preprocess the data if necessary (e.g., encode 'Gender' to numerical values)
d) Select features for clustering (e.g., Age, Annual Income, Spending Score)
e) Standardize the data (important for DBSCAN)
f) Perform DBSCAN clustering
    - Provide Radius for density estimation
    - Provide Minimum number of samples in a neighborhood to form a cluster
g) Add cluster labels to the original dataset
h) Visualize the clusters (scatter plot)
i) Finally, Display basic cluster statistics.



```
-1    60
 2    51
 4    32
 3    28
 0    17
 5     7
 1     5
Name: DBSCAN_Cluster, dtype: int64
```

Github Link: **https://github.com/HasnatSabbir/Data-Science-Assignment-02.git**