Indian Institute of Technology Patna

CS564 - Foundation of Machine Learning

Assignment #2 DBSCAN and Hierarchical Clustering.
Submission Date: 20th April 2024

Baskar Natarajan - 2403res19(IITP001799)
Jyotisman Kar – 2403res35(IITP001751)
SEMESTER-1
MTECH AI & DSC
INDIAN INSTITUTE OF TECHNOLOGY PATNA.

1. Problem Description:

- Cluster the provided dataset “students.csv” of graduated students using DBSCAN and Hierarchical clustering algorithms.
- Choose the most relevant attributes (maximum 10 or minimum 5). Plot the clusters and count the number of points belonging to each cluster.
- Each cluster must be plotted using assorted colors. You can take any attribute to consider any axes.
- For DBSCAN consider the Midpoint = 5 and eps= 0.5
- Find out the Silhouette Score

2. Data Patterns:

Data chosen are,

i. age,
ii. sports,
iii. music,
iv. shopping,
v. NumberOffriends

b. Young individuals interested in sports and music, with moderate shopping habits and a large social network.

c. Older individuals with diverse music preferences, moderate interest in sports, frequent shopping habits, and a small social network.

d. Individuals with specific interests in certain sports or music genres, with varying shopping behaviors and social network sizes.

e. These patterns can provide insights into the characteristics and behaviors of different segments within the dataset, which can be useful for targeted marketing, personalized recommendations, and understanding consumer preferences.

3. DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

a. Business Documentation:

- **Objective:** DBSCAN is used for clustering data points based on their density in a dataset. It is particularly useful for discovering clusters of arbitrary shapes and showing outliers (noise).
- **Use Case:** DBSCAN can be applied in various domains such as customer segmentation in retail, fraud detection in finance, and anomaly detection in network traffic.
- **Benefits:**
  - Finds clusters of varying shapes and sizes.

- Robust to noise and outliers.
- Does not require specifying the number of clusters in advance.

- **Limitations:**
  - Sensitivity to the choice of distance metric and density parameters.
  - Difficulty in clustering data with varying densities.

b. Technical Documentation:

- **Overview:** DBSCAN defines clusters as dense regions separated by sparser regions. It classifies points as core points, border points, or noise based on the density of neighboring points within a specified radius (eps) and the minimum number of points (min_samples).
- **Algorithm Steps:**
  - **Core Point Identification:** For each data point, DBSCAN calculates the number of neighboring points within a specified radius (epsilon).
  - **Density Reachability:** Points with enough neighbors within the epsilon radius are classified as core points.
  - **Cluster Expansion:** Core points are connected to form clusters, and density-reachable points are assigned to the same cluster. Border points, which are reachable from a core point but do not have enough neighbors to be considered core points, are included in the cluster.
  - **Noise Identification:** Points that are not core points or reachable from any core point are considered noise and are not assigned to any cluster.
- **Parameters:**
  - Epsilon (eps): Radius of the neighborhood around each point.
  - Smallest samples (min_samples): Minimum number of points within the epsilon radius to define a core point.
- **Implementation:** DBSCAN involves finding core points, expanding clusters by connecting core points, and marking noise points. It is implemented by calculating pairwise distances between points and iterating through each point to assign it to a cluster or mark it as noise.

4. Hierarchical Clustering

a. Business Documentation:

- **Objective:** Hierarchical clustering organizes data into a tree-like structure (dendrogram) based on the similarity between data points. It is used to find hierarchical relationships and structure in the data.

- **Use Case:** Hierarchical clustering can be applied in market segmentation, biological taxonomy, and document clustering.
- **Benefits:**
  - Reveals hierarchical relationships within the data.
  - No need to specify the number of clusters beforehand.
  - Provides an intuitive visualization of cluster hierarchy.
- **Limitations:**
  - Computationally expensive for large datasets.
  - Interpretation of clusters can be subjective.

b. Technical Documentation:

- **Overview:** Hierarchical clustering builds a hierarchy of clusters by iteratively merging or splitting clusters based on a similarity metric. It can be agglomerative (bottom-up) or divisive (top-down).
- **Algorithm Steps:**
  - **Initialization:** Treat each data point as a singleton cluster.
  - **Pairwise Similarity:** Calculate the similarity (distance) between all pairs of data points.
  - **Cluster Merge:** Iteratively merge the two most similar clusters based on a specified linkage criterion until all points belong to a single cluster or until a predefined number of clusters is reached.
  - **Dendrogram Construction:** Construct a dendrogram to visualize the hierarchical relationships between clusters.
- **Parameters:**
  - Linkage method: Figures out how the distance between clusters is measured during merging. Common methods include single linkage, complete linkage, and average linkage.
  - Distance metric: Specifies the distance measure between data points (e.g., Euclidean distance, Manhattan distance).
- **Implementation:** Hierarchical clustering starts by treating each data point as a singleton cluster and iteratively merges or splits clusters based on the chosen linkage method and distance metric. The process continues until all points belong to a single cluster or until a predefined number of clusters is reached.

5. Python Libraries used are,

a. **Pandas** – To Read CSV file and create data frame.

b. **Numpy** – Array Operations

c. **Matplotlib.pyplot** - Plot the graph

d. **sklearn.preprocessing** - ***StandardScaler*** – Standardize the features.

e. **sklearn.metrics** - to calculate *silhouette_score*

f. **scipy.cluster.hierarchy** - Draw *dendrogram*, calculate *linkage*

6. Major Functions are,

a. *find_neighbors*

i. Find the Neighbours based on given epsilon distance.

b. *expand_cluster*

- Expand the clusters based on the neighbors found. If the current cluster value is –1, add it to the outlier list, else find the neighbors again with the epsilon value and minimum samples.

c. *dbscan_clustering*

- Calculate the Density based clustering for the given data.
- Find the Neighbours based on the epsilon distance and minimum sample values given.
- Expand the cluster with identified labels and Neighbours.
- Finally return the calculated labels

d. *hierarchical_clustering*

- **method='ward'**: This parameter specifies the method used to compute the distance between clusters during hierarchical clustering.
- In this case, 'ward' refers to Ward's method, which minimizes the variance when merging clusters and is commonly used for hierarchical clustering.
- Other commonly used methods include **'single'**, **'complete'**, **'average'**, etc., each of which computes the distance between clusters differently
- **linkage_matrix**: This is the output of the linkage function, which is a hierarchical representation of the clustering structure. It contains information about how the clusters are merged at each step of the hierarchical clustering process.
- **dendrogram**(linkage_matrix, p=n_clusters, no_plot=True): This part of the line generates the dendrogram using the provided linkage_matrix computed from hierarchical clustering
  - The **p** parameter specifies the number of clusters to display in the dendrogram.
  - When **no_plot** is set to **True**, it prevents the function from plotting the dendrogram directly, which means it won't display the

dendrogram on the screen but will still compute it in the background.

- ['**ivl**']: After generating the dendrogram, we access the information stored in the dictionary returned by the dendrogram function.
- Above Variable contains the labels of the leaves in the dendrogram, which represent the original data points.
- These labels are the indices of the data points in the original dataset.
- clusters: Finally, we assign the labels of the leaves (data points) in the dendrogram to the variable clusters.
- Calculate the Cluster labels from the above variable received from dendrogram function and return with Linkage matrix.

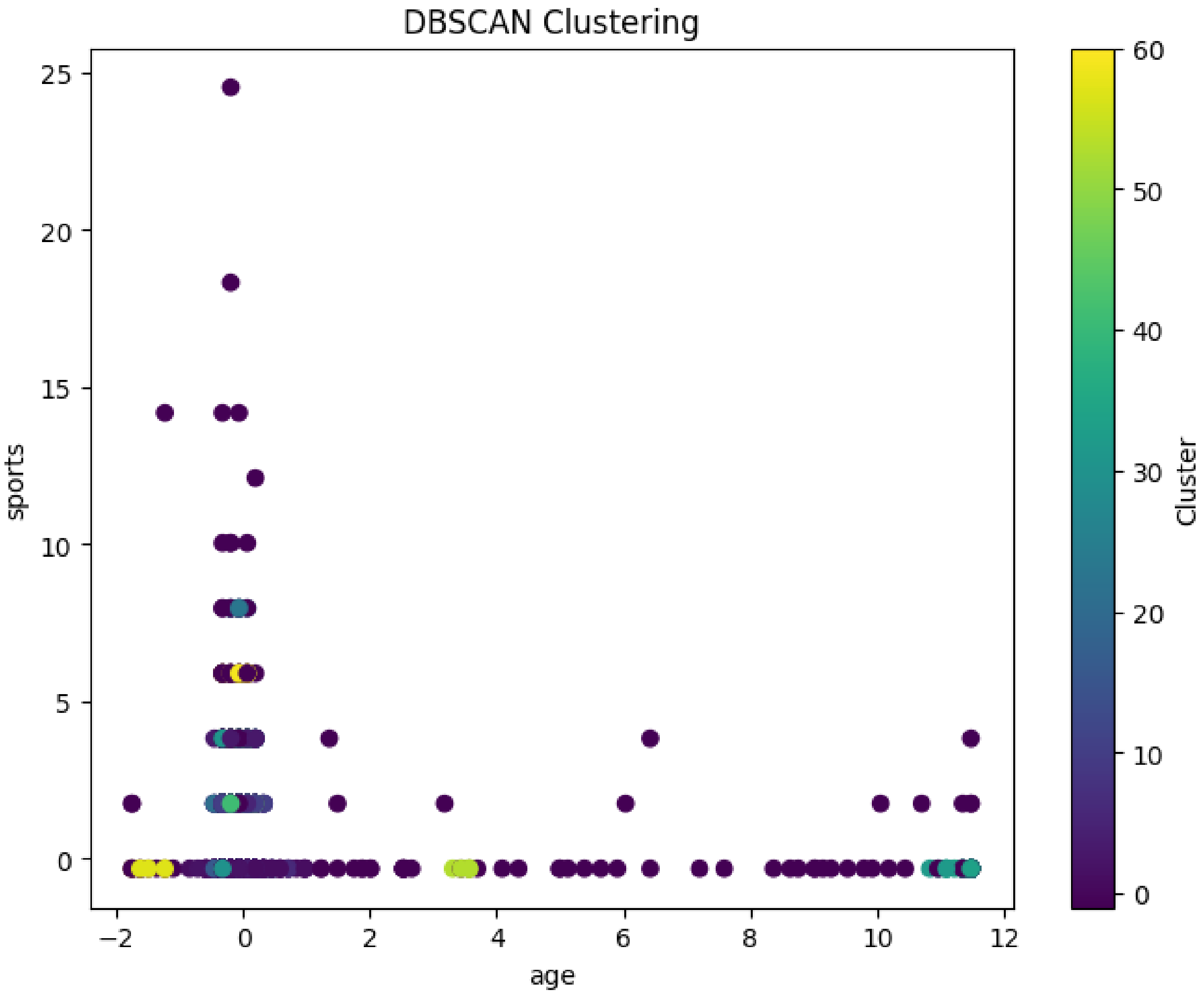
e. Important Steps are,

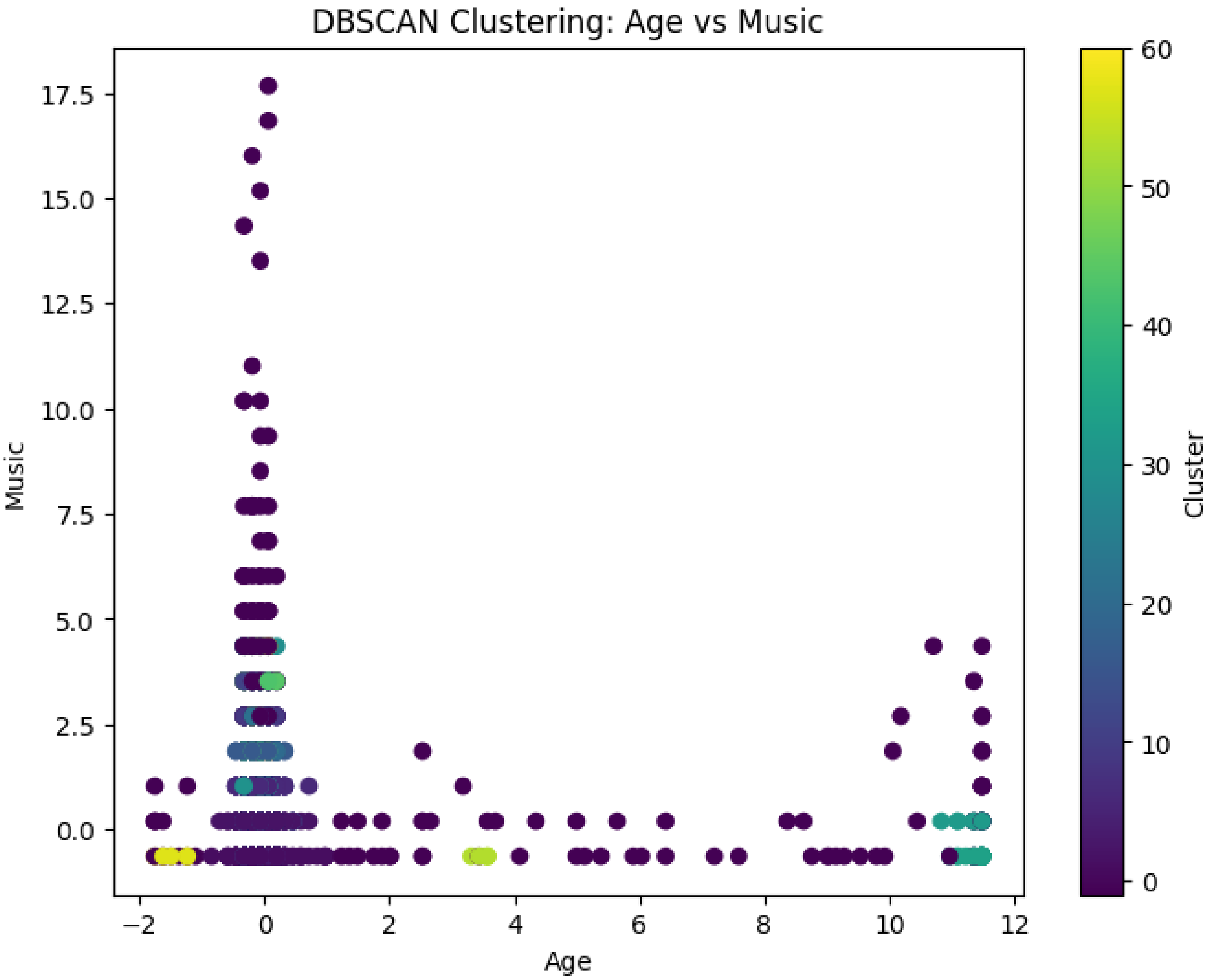
i. Read CSV File(students.csv).
ii. Select the Columns which we want to do clustering.
   1. age,
   2. sports,
   3. music,
   4. shopping,
   5. NumberOffriends
ii. Drop Null Rows from the data frame.
iii. Convert age column in to numeric, if error make it null
iv. Re
v. move the null rows of age column.
vi. Standardize the selected features available in data frame.
vii. Do DBSCAN Clustering.
viii. Calculate the sum of datapoints available in each cluster and print.
ix. Calculate the silhouette_score for the identified DBSCAN clusters and given data.
x. Do the Hierarchical Clustering.
xi. Calculate the silhouette_score for the identified Hierarchical clusters and given date.
xii. Plot DBSCAN Graph
xiii. Plot Hierarchical Graph
xiv. Plot Dendrogram Graph.

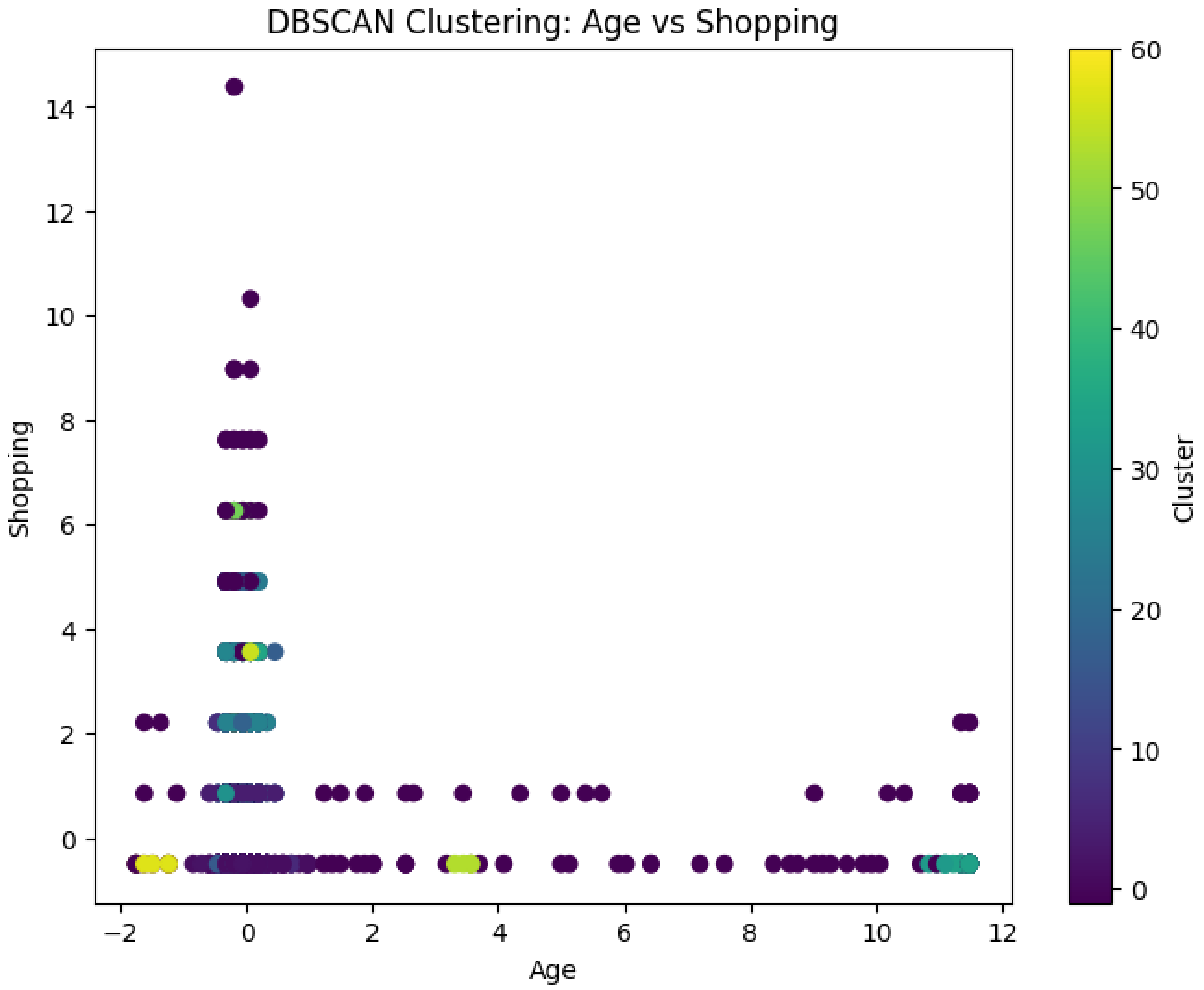
7. Output:

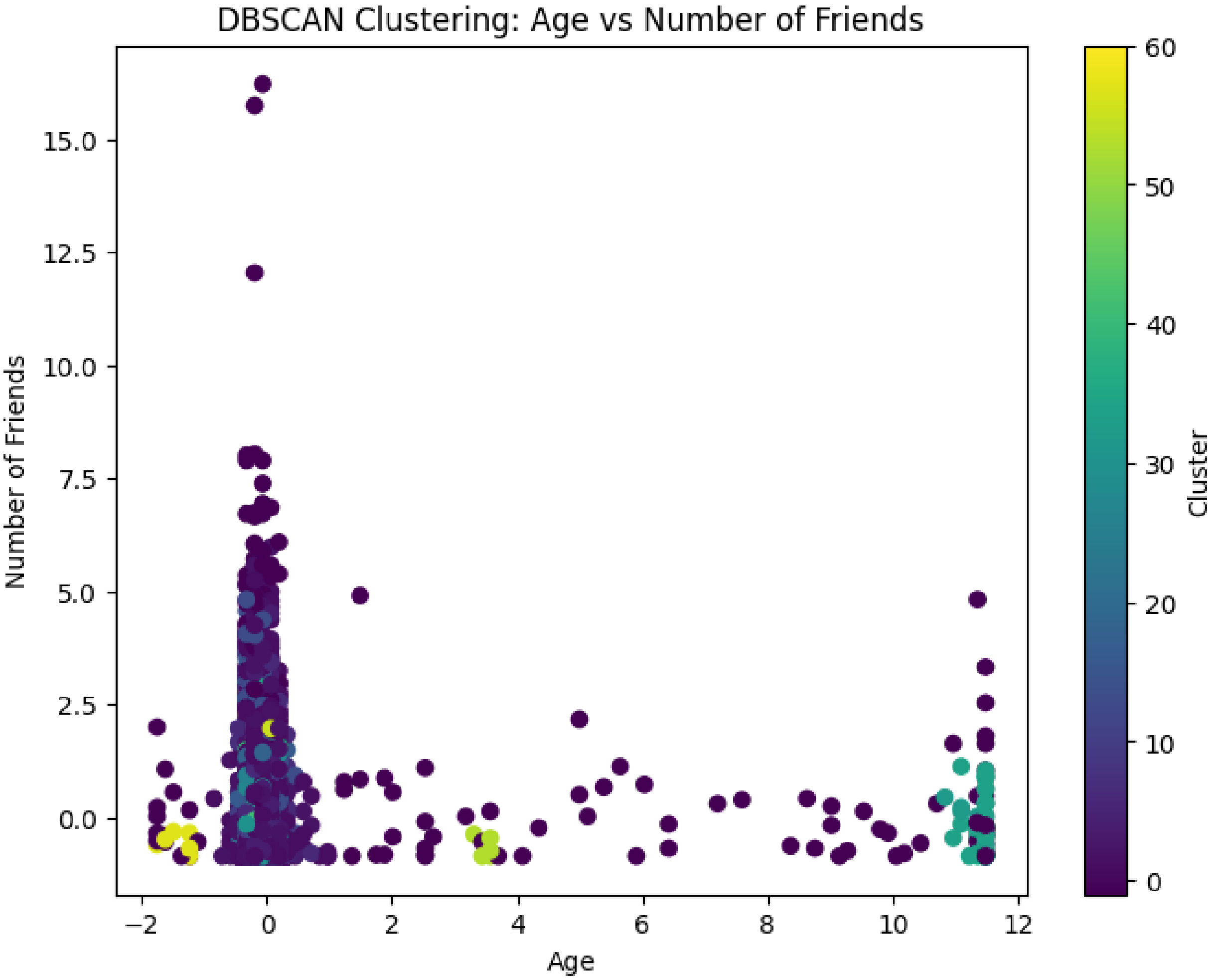
```
Total Data Points in Each Cluster (DBSCAN):
Cluster 1: 4672 data points
Cluster 2: 2073 data points
Cluster 3: 61 data points
Cluster 4: 897 data points
Cluster 5: 90 data points
Cluster 6: 724 data points
Cluster 7: 56 data points
Cluster 8: 251 data points
Cluster 9: 79 data points
Cluster 10: 397 data points
Cluster 11: 32 data points
Cluster 12: 7 data points
Cluster 13: 764 data points
Cluster 14: 26 data points
Cluster 15: 11 data points
Cluster 16: 235 data points
Cluster 17: 50 data points
Cluster 18: 158 data points
Cluster 19: 26 data points
Cluster 20: 74 data points
Cluster 21: 83 data points
Cluster 22: 29 data points
Cluster 23: 7 data points
Cluster 24: 11 data points
...
Cluster 59: 5 data points
Cluster 60: 5 data points
```

Silhouette Score for DBSCAN Clustering: 0.26098543905394356
Silhouette Score for Hierarchical Clustering: 0.3201211080787094

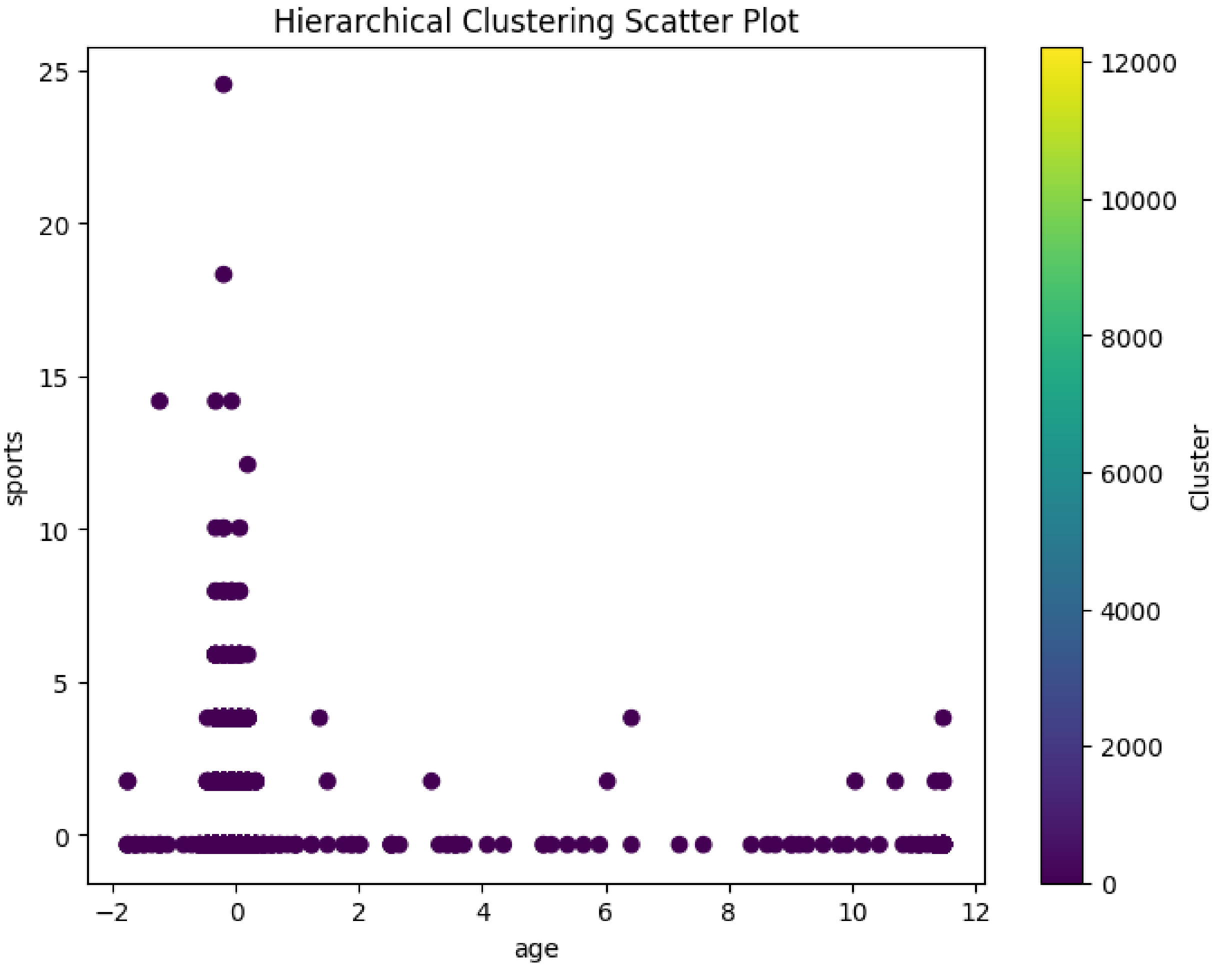

DBSCAN Clustering
sports
25
20
15
10
5
0
age
−2
0
2
4
6
8
10
12
Cluster
60
50
40
30
20
10
0

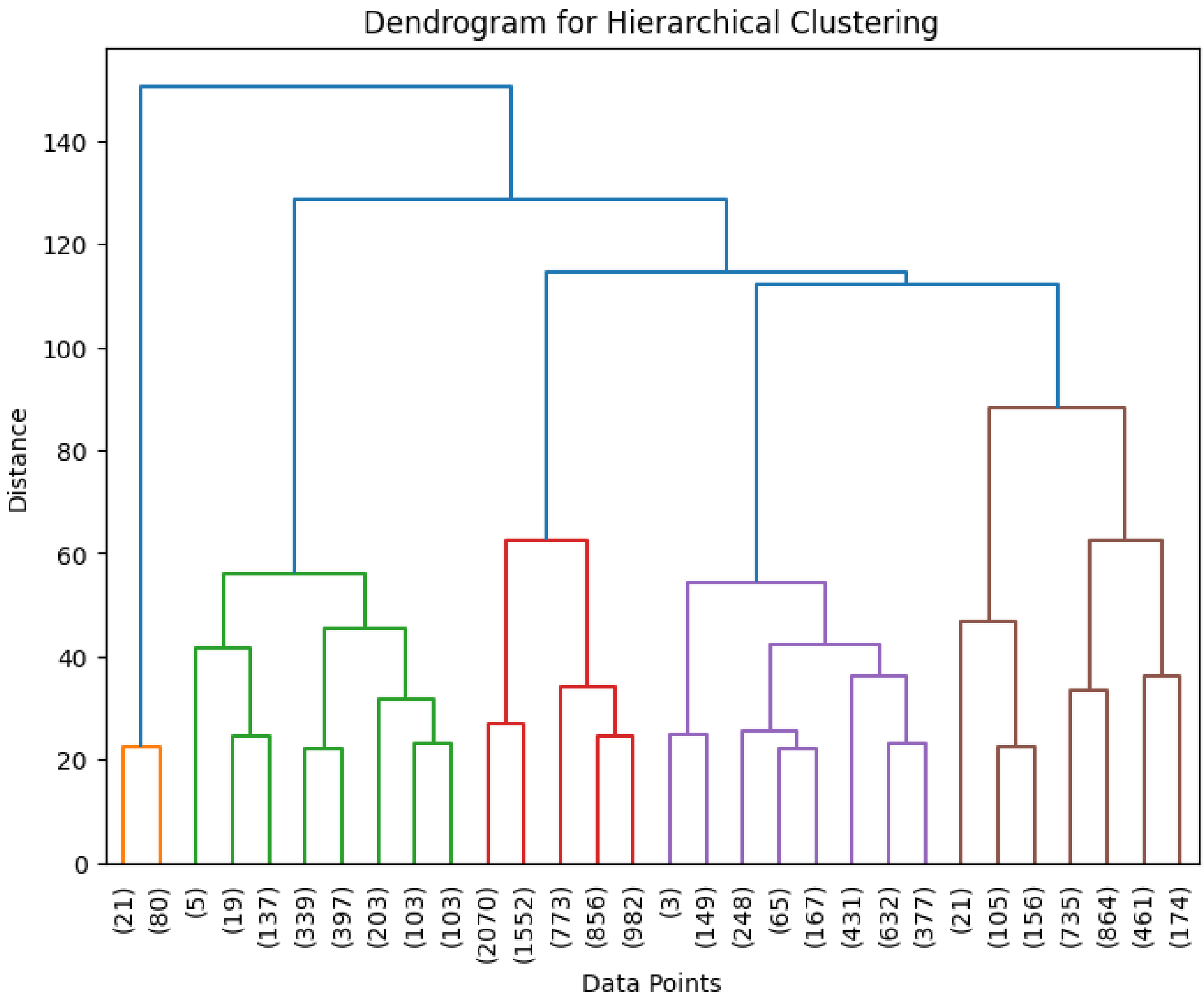

DBSCAN Clustering: Age vs Music
Music
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−2
0
2
4
6
8
10
12
Age
Cluster
60
50
40
30
20
10
0


DBSCAN Clustering: Age vs Shopping
Shopping
14
12
10
8
6
4
2
0
−2
0
2
4
6
8
10
12
Age
Cluster
60
50
40
30
20
10
0



DBSCAN Clustering: Age vs Number of Friends
Number of Friends
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−2
0
2
4
6
8
10
12
Age
Cluster
60
50
40
30
20
10
0


Hierarchical Clustering Scatter Plot
sports
age
25
20
15
10
5
0
−2
0
2
4
6
8
10
12
Cluster
12000
10000
8000
6000
4000
2000
0


Dendrogram for Hierarchical Clustering
Distance
0
20
40
60
80
100
120
140
(21)
(80)
(5)
(19)
(137)
(339)
(397)
(203)
(103)
(103)
(2070)
(1552)
(773)
(856)
(982)
(3)
(149)
(248)
(65)
(167)
(431)
(632)
(377)
(21)
(105)
(156)
(735)
(864)
(461)
(174)
Data Points