**Report Clustering Methods**

**Homework 1**

**NYCU**

This report is present a comprehensive analysis of clustering algorithms applied to a variety of open-source datasets. The objective is to evaluate the performance and applicability of different clustering techniques on datasets that exhibit a range of complexities and structural characteristics. This evaluation is conducted using nine distinct datasets: Complex9, Curves2, Diamond9, Disk-5000n, Gausians1, Hypercube, Impossible, Disk-4000n, and Mopsi-Joensuu, and Sizes5 sourced from the Clustering Benchmark database.

The clustering methods that evaluate selected for this study include Mean Shift, K-means, Agglomerative, DBSCAN, HDBSCAN and OPTICS. These methods were chosen based on methods that we have studied, distinct methodological approaches and their potential for revealing intricate patterns in complex datasets. Each method will be thoroughly tested across all datasets, with a focus on tuning hyper-parameters to optimize performance on clustering pattern.

In the experiment, the dataset was displayed on a 2D graph as shown in Figure 1, arranged in the following order: compex9, curves2, diamond9, disk5000n, gaussians1, hypercube, impossible, disk-4000n, mopsi-joensuu, and sized5. Each dataset exhibits its own unique pattern and presents specific extraction challenges. Consequently, a method that is suitable for one dataset might not necessarily be effective for another. Additionally, the performance of clustering depends on the precise tuning of each hyper-parameter.

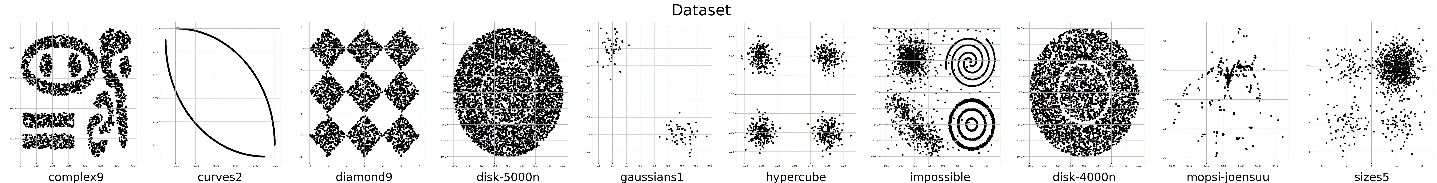


Figure 1. Dataset Collections

The reason of choosing MeanShift Clustering is because this method can discover blop in a smooth density of sample. Because it is centroid based algorithm, which work by updating candidate for centroid to be mean of points within a given region. Then this method will filter the candidates in post-processing stage to estimate near-duplicates from final set of centroid. The hyper parameter of this algorithm is based on bandwith that can automatically base on sized region.

Then, the K-means clustering is chosen, because it has simplicity, speed, and fixed number of clusters. This algorithm that work by trying to separate samples in n groups of equal variance, minimizing a criterion known as the inertia. This algorithm has hyper-parameter of the number of clusters that should be specified.

Any other clustering algorithm that interested to discuss is Agglomerative Clustering. It is hierarchical clustering using a bottom to up approach. Each observation starts in its own clusters, and clusters are successively merge together. The hyper-parameter that will mainly be discussed on this report is n\_clusters and linkages. The n\_clusters is total number of clusters that will find. The linkage determines which distance to use between sets of features.

Different with the previous algorithm, the DBSCAN Clustering views clusters as areas of high density separated by areas of low density. Due to this rather generic view, BDSCAN can find any type of shape of cluster. This algorithm has two parameters there are min\_samples and epsilon. Higher main\_samples or lower epsilon is mean higher density necessary to form a cluster.

The OPTICS Clustering has similarities with DBSCAN. The key of different of them are that OPTICS build reachability graph, which assigns each sample both reachability and cluster ordering. This algorithm has hyper parameter that was used on this report is min\_samples, xi and min\_clusters. The min\_samples is the number of samples in a neighborhood for a point to be considered as a core point. Then the xi will determine the minimum steepness on the reachability plot that constitutes a cluster boundary. And the samples are minimum number of samples in each cluster.

The last algorithm that will discuss on this report is HDBSCAN. This algorithm is interesting to discuss because it is an extension of DBSCAN and OPTICS. This algorithm alleviates this assumption and explores all possible density scales by building an alternative representation of the clustering problem. This hyper parameter of this algorithm that will discuss is min\_cluster\_size, min\_sample, max\_cluster\_size and leaf\_sized. The min\_cluster\_size is minimum number of samples in a group. The min\_sample is number of samples in a neighborhood for a point to be considered as a core point. The max\_cluster\_size is the maximum limit to the size of clusters returned. Then, leaf\_sized is leaf size for trees responsible for fast nearest neighbour queries.

The hyper parameter that was used on the experiment described on Table 1-6. These hyper parameter is best value that was found on the clustering algorithm based on dataset.

Table . Mean Shift hyper parameter

|  |  |  |
| --- | --- | --- |
| No | Dataset | hyper parameter |
| bandwidth |
| 1. | Complex9 | 152.857 |
| 2. | Curves2 | 0.03242 |
| 3. | Diamond9 | 1.7941 |
| 4. | Disk-5000n | 5.2169 |
| 5. | Gausians1 | 0.0788 |
| 6. | Hypercube | 1.0193 |
| 7. | Impossible | 4.3579 |
| 8. | Disk-4000n | 5.2083 |
| 9. | Mopsi-Joensuu | 0.2527 |
| 10. | Sizes5 | 4.0281 |

Table . K-mean hyper parameter

|  |  |  |
| --- | --- | --- |
| No | Dataset | hyper parameter |
| k |
| 1. | Complex9 | 10 |
| 2. | Curves2 | 10 |
| 3. | Diamond9 | 10 |
| 4. | Disk-5000n | 10 |
| 5. | Gausians1 | 10 |
| 6. | Hypercube | 10 |
| 7. | Impossible | 10 |
| 8. | Disk-4000n | 10 |
| 9. | Mopsi-Joensuu | 10 |
| 10. | Sizes5 | 10 |

Table . Agglomerative hyper parameter

|  |  |  |  |
| --- | --- | --- | --- |
| No | Dataset | hyper parameter | |
| n\_clusters | linkages |
| 1. | Complex9 | 10 | ward |
| 2. | Curves2 | 10 | ward |
| 3. | Diamond9 | 10 | ward |
| 4. | Disk-5000n | 10 | ward |
| 5. | Gausians1 | 10 | ward |
| 6. | Hypercube | 10 | ward |
| 7. | Impossible | 10 | ward |
| 8. | Disk-4000n | 10 | ward |
| 9. | Mopsi-Joensuu | 10 | ward |
| 10. | Sizes5 | 10 | ward |

Table . DBSCAN hyper parameter

|  |  |  |  |
| --- | --- | --- | --- |
| No | Dataset | hyper parameter | |
| epsilon | min\_samples |
| 1. | Complex9 | 16 | 10 |
| 2. | Curves2 | 0.0016 | 10 |
| 3. | Diamond9 | 0.16 | 10 |
| 4. | Disk-5000n | 1.6 | 2 |
| 5. | Gausians1 | 1.6 | 2 |
| 6. | Hypercube | 0.16 | 10 |
| 7. | Impossible | 1 | 8 |
| 8. | Disk-4000n | 1.6 | 2 |
| 9. | Mopsi-Joensuu | 0.2 | 10 |
| 10. | Sizes5 | 0.8 | 8 |

Table . HDBSCAN hyper parameter

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Dataset | hyper parameter | | |
| min\_samples | x\_i | min\_clusters |
| 1. | Complex9 | 16 | 10 |  |
| 2. | Curves2 | 0.0016 | 10 |  |
| 3. | Diamond9 | 0.16 | 10 |  |
| 4. | Disk-5000n | 1.6 | 2 |  |
| 5. | Gausians1 | 1.6 | 2 |  |
| 6. | Hypercube | 0.16 | 10 |  |
| 7. | Impossible | 1 | 8 |  |
| 8. | Disk-4000n | 1.6 | 2 |  |
| 9. | Mopsi-Joensuu | 0.2 | 10 |  |
| 10. | Sizes5 | 0.8 | 8 |  |

Table . OPTICS hyper parameter

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Dataset | hyper parameter | | | |
| min\_cluster\_size | min\_samples | max\_cluster\_size | leaf\_sized |
| 1. | Complex9 | 16 | 10 |  |  |
| 2. | Curves2 | 0.0016 | 10 |  |  |
| 3. | Diamond9 | 0.16 | 10 |  |  |
| 4. | Disk-5000n | 1.6 | 2 |  |  |
| 5. | Gausians1 | 1.6 | 2 |  |  |
| 6. | Hypercube | 0.16 | 10 |  |  |
| No | Dataset | hyper parameter | No | Dataset | hyper parameter |
| 7. | Impossible | 1 | 8 |  |  |
| 8. | Disk-4000n | 1.6 | 2 |  |  |
| 9. | Mopsi-Joensuu | 0.2 | 10 |  |  |
| 10. | Sizes5 | 0.8 | 8 |  |  |

After the testing all methods and tuning the hyper-parameter, It’s got the results of the clustering that is shown on Figure 2. On this results, it shows that each result has their own special pattern and different from each other. So this shows that each method will be useful according to the clustering problem to be solved.

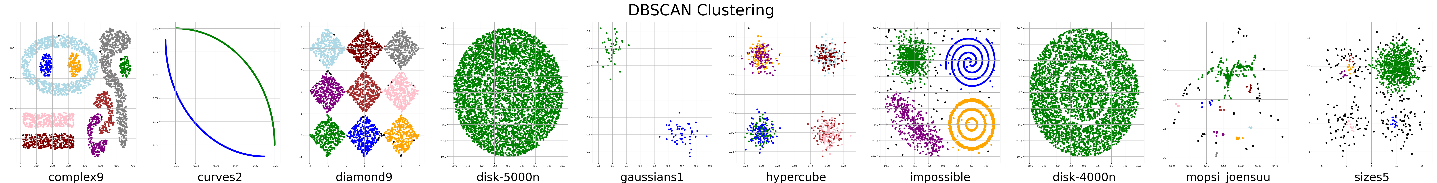
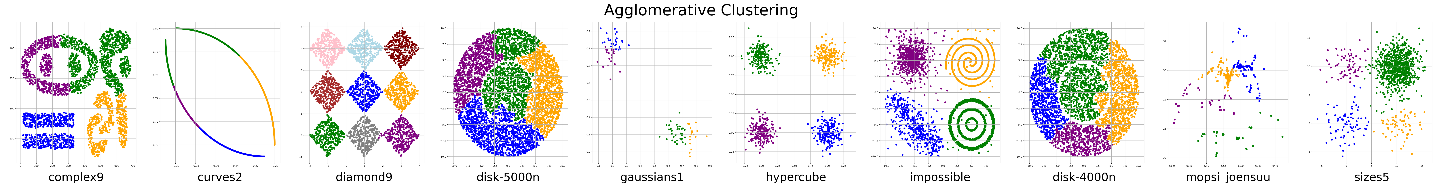
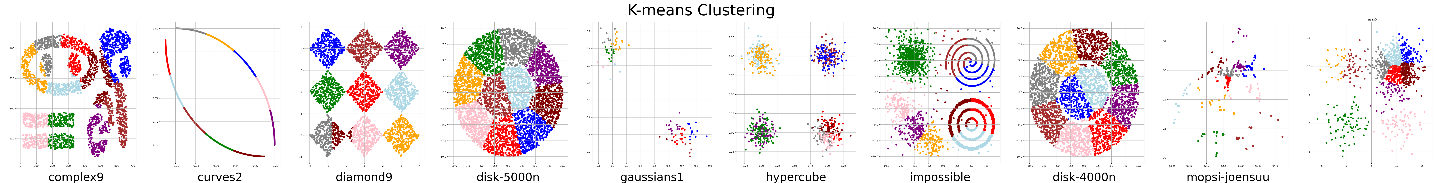
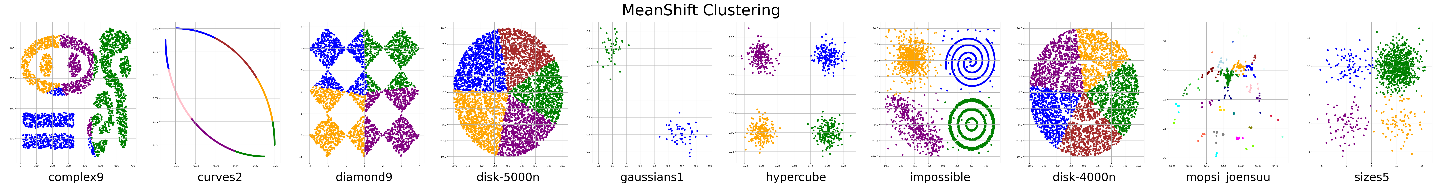
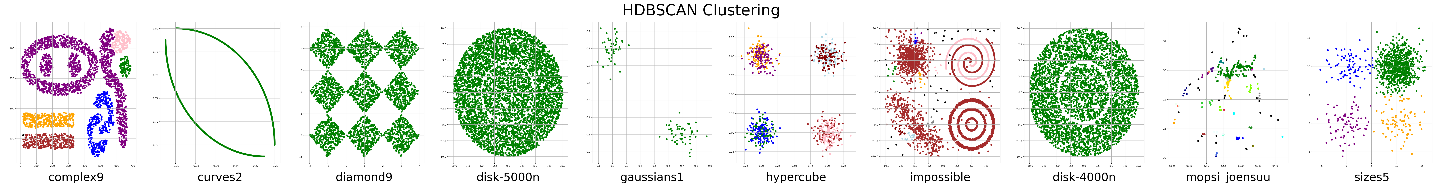
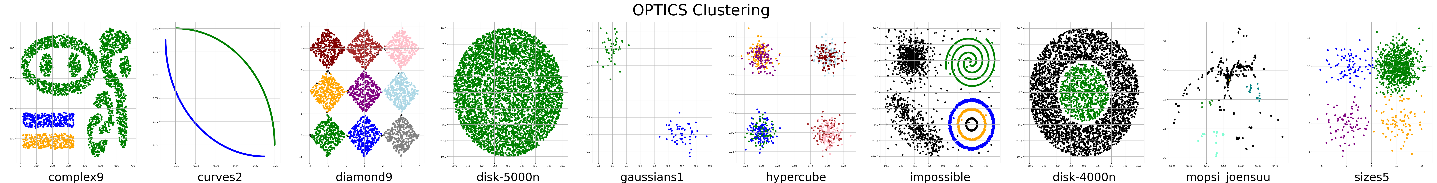
  

Figure 2 Results of Clustering Methods

DBSCAN Clustering

HDBSCAN Clustering

Agglomerative Clustering

K-means Clustering

OPTICS Clustering