KHULNA UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Department of Computer Science and Engineering

**Course No:** 4239

**Course Name:** Data Mining

**Submitted To:**Animesh Kumar PaulAssistant ProfessorDepartment of Computer Science and Engineering

**Submitted By:**

Kristy Saha

Roll: 1607113

Sec: A

**Introduction:**

In this modern era people depend on machines, they want to do every possible task using machines or technology rather than doing it themselves. For developing this type of system machine learning, deep learning etc methods are proposed. These methods need a lot of data for training the machines. In the real world, different types and amounts of data are found such as numeric, nominal, ordinal, etc. Data mining is used for processing data and find out important information from it. Clustering is one type of data mining that divides a population of data into several groups with similar datapoint.

Here, we are applying 6 clustering algorithm and observing their impact on a specific dataset.

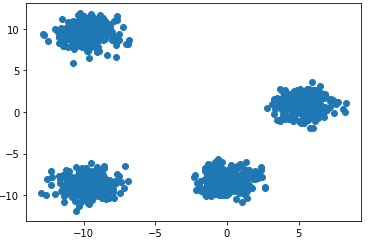
**Dataset:**

For dataset 'sklearn make-blobs' dataset is used. It generates isotropic Gaussian blobs for clustering. It creates multiclass datasets by allocating each class one or more normally-distributed clusters of points. This dataset provides greater control regarding the centers and standard deviations of each cluster and is used to demonstrate clustering [ 1 ].

Here 1200 samples with 2 features are taken from the make-blobs dataset.

Dataset Source: <https://scikitlearn.org/stable/modules/generated/sklearn.datasets.make_blobs.html>

This is the visual representation of the dataset :



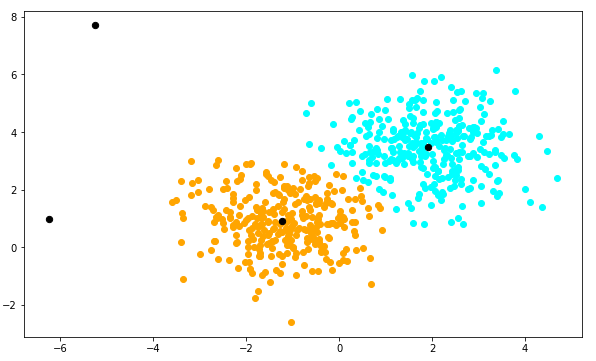
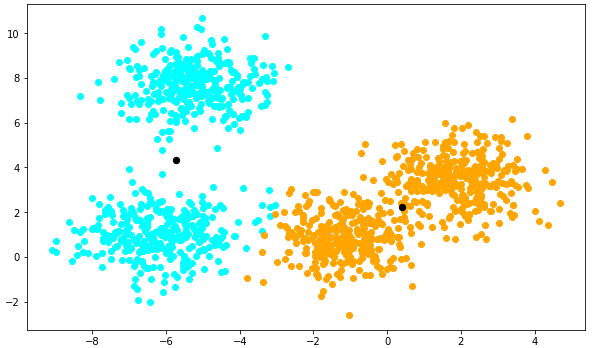
There are many clustering algorithm. Here some clustering algorithms are applied in this dataset.

**K-means:**

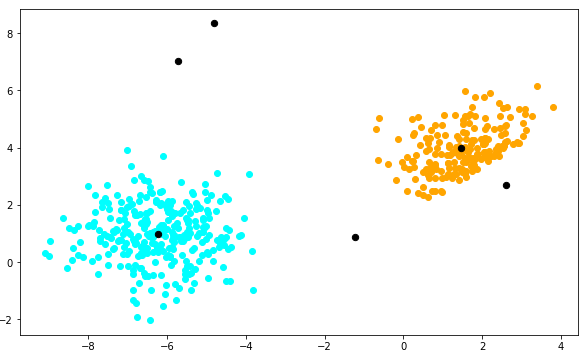
It is implemented using sklearn library [2]. Where 3 parameters are used.

* **n\_clusters: Number of clusters**
* **n\_init :** Number of time the k-means algorithm will run
* **max\_iter :** Maximum number of iterations of the k-means algorithm for a single run

By changing cluster number different results are found.



Cluster=2 cluster=4



Cluster=6

Fig1: Kmeans for cluster number 2, 4 and 6

|  |  |  |  |
| --- | --- | --- | --- |
| **Number of Clusters** | **ARI** | **NMI** | **Silhouette** |
| 2 | 0.478 | 0.616 | 0.501 |
| 4 | 0.954 | 0.939 | 0.623 |
| 6 | 0.779 | 0.834 | 0.445 |

**Advantage:** Here, are some experimental advantage of k-means

* It computes cluster very fast for a large number of datapoint
* Easy to implement and has many available library functions
* Scales large dataset

**Disadvantage:** Here, are some experimental disadvantage of k-means

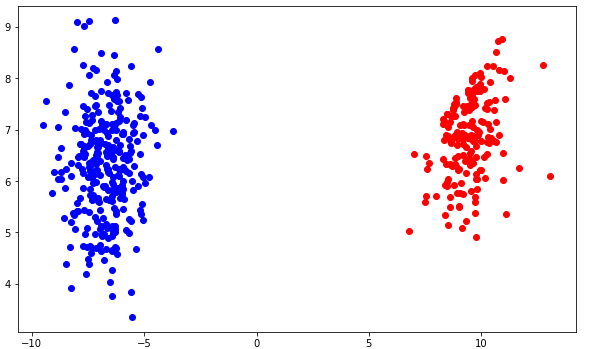
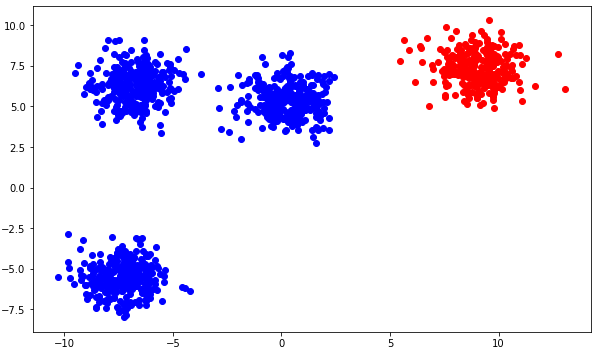
* Value of the cluster number needed to be given before clustering. For different cluster number different kind of result is generated. so it is hard to predict which cluster number is perfect and optimum for clustering
* It did not work well for global cluster
* It Did not work well for high dimensional data and cluster in original data

**Brich:**

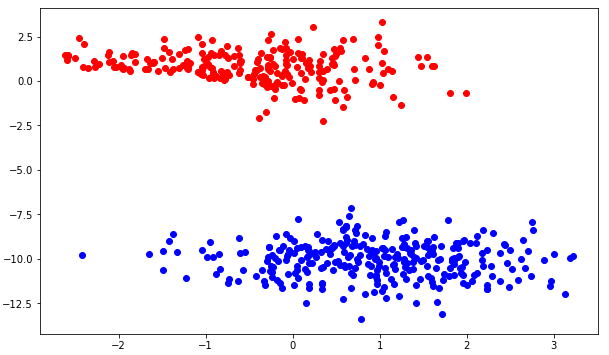
It is implemented using sklearn library [3]. Where 3 parameters are used.

* **N\_clusters : Number of clusters**
* **Threshold :** the maximum number of data points a sub-cluster in the leaf node of the CF tree can hold
* **Branching\_factor:** the maximum number of CF sub-clusters in each node

For different cluster number BRICH shows different result.

****

Cluster=2 cluster=4

****

Cluster=6

Fig1: BRICH for cluster number 2,4 and 6

|  |  |  |  |
| --- | --- | --- | --- |
| **Number of Clusters** | **ARI** | **NMI** | **Silhouette** |
| 2 | 0.333 | 0.577 | 0.545 |
| 4 | 0.689 | 0.782 | 0.425 |
| 6 | 0.822 | 0.890 | 0.547 |

**Advantage:** Here, are some experimental advantage of BRICH

* It computes cluster very fast for a large number of datapoint
* It has many available library functions
* Scales large dataset

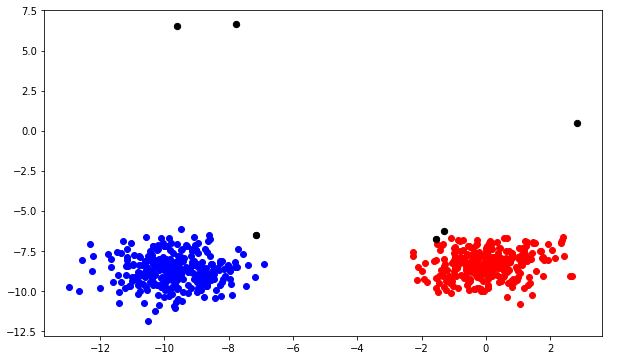
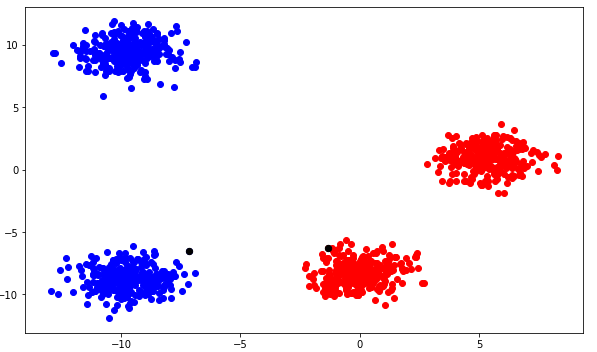
**Disadvantage:** Here, are some experimental disadvantage of BRICH

* It can only process numeric attribute so, string type attribute had to be dropped or converted to numeric value

**K-medoid:**

It has one parameter: Cluster number

For different cluster number different result is found

****

Cluster=2 Cluster=6

fig: k-medoid for cluster number 4 and 6

|  |  |  |  |
| --- | --- | --- | --- |
| **Number of Clusters** | **ARI** | **NMI** | **Silhouette** |
| 2 | 0.499 | 0.667 | 0.494 |
| 6 | 0.619 | 0.766 | 0.215 |

**Advantages:** Here, are some experimental advantage of K-medoid

* It selected the most centered member belonging to a cluster.
* There are no outliers in our dataset, but if there were any outlier K-mediod algorithm could have ignored it while making clusters.
* It’s simple and efficient to implement.

**Disadvantages:** Here, are some experimental disadvantage of K-medoid

* that they are not suitable for clustering non-spherical groups of objects
* It obtained different results for different runs on the dataset because the finitial medoids are chosen randomly

**AGNES(AGglomerative NESting):**

AGNES has parameters like cluster number, distance metric and linkage criteria.

**Advantage:** Here, are some experimental advantage of AGNES

* Easy to understand the algorithm
* Easy for implementation
* As it counts all the datapoint as a single cluster initially, no datapoint get missing during this clustering algorithm

**Disadvantage:** Here, are some experimental disadvantage of AGNES

* It takes huge computational time for many samples or data
* It rarely provides the best solution for clustering as it follows leaf to root method
* Need to change the parameters arbitrarily like distance metric and linkage criteria, this puts a huge impact in final outcome

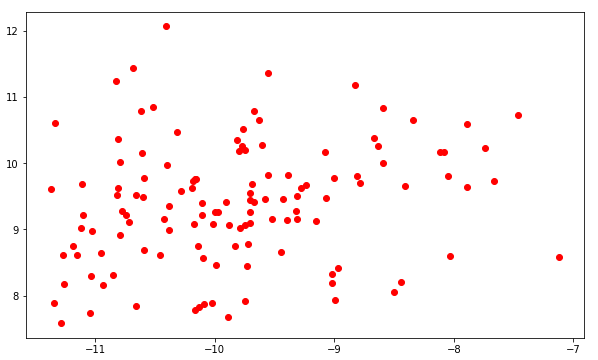
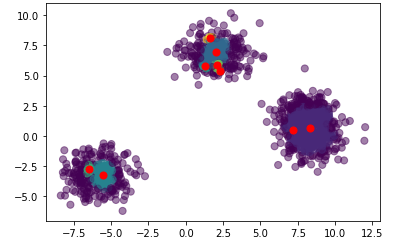
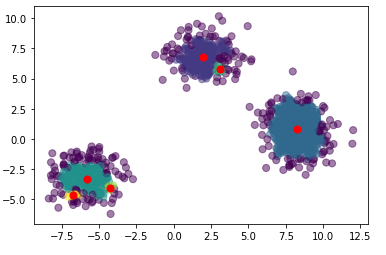
**

Fig: AGNES clustering

|  |  |  |  |
| --- | --- | --- | --- |
| **Number of Clusters** | **ARI** | **NMI** | **Silhouette** |
| 6 | 0.819 | 0.889 | 0.598 |

**DBSCAN(Density Based Spatial Clustering of Application with Noise):**

DBSCAN takes 2 parameter: epsilon, minimum points

****

(Epsilon=0.3, Min Point=5) (Epsilon=0.3, Min Point=10)

Fig: DBSCAN clustering

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameters** | **ARI** | **NMI** | **Silhouette** |
| Epsilon=0.3,  Min Point=5 | 0.499 | 0.667 | 0.494 |
| Epsilon=0.3,  Min Point=10 | 0.619 | 0.766 | 0.215 |

**Advantages:** Here, are some experimental advantage of DBSCAN

* This algorithm doesn’t require the number of clusters, confusion of choosing the value of number of clusters is freed in this algorithm
* It can make cluster of arbitrary shapes
* Though this dataset doesn’t have any noise DBSCAN can handle noise in dataset better.

**Disadvantages:** Here, are some experimental disadvantage of DBSCAN

* It cannot handle high dimensional data
* Complex hyper-parameter handling

**CHAMELEON**

**Advantage:** Here, are some experimental advantage of CHAMELEON

* It can handle large dataset
* It can recognize arbitrary shaped clusters

**Disadvantage:** Here, are some experimental disadvantage of CHAMELEON

* Algorithm is pretty complex
* A lot of parameters need to handle

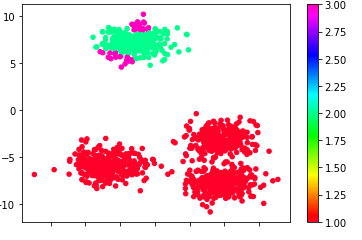
****

fig: CHAMELEON clustering

**Comparative Analysis:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Result** | **K-Means** | **BRICH** | **KMedoid** | **AGNES** | **DBSCAN** | **CHAMELEON** |
| **ARI** | 0.779 | 0.822 | 0.619 | 0.819 | 0.619 | 0.0005 |
| **NMI** | 0.834 | 0.890 | 0.766 | 0.889 | 0.766 | -0.005 |
| **Silhouette** | 0.445 | 0.547 | 0.215 | 0.598 | 0.215 | 0.0035 |

In this table, experimental result for cluster = 6 using different algorithm is found. Here

ARI: Represents correlation between clusters. Value of ARI can be between -1 to 1. Similar clustering have a positive ARI, 1.0 is the perfect match score

NMI: Represents a normalization of the Mutual Information to scale the results between 0 (no mutual information) and 1 (perfect correlation)

Silhoutte: Represents separation distance between the resulting clusters.

From this table, it is seen AGNES has greater ARI, NMI and Silhouette score than others. But from the visual representation, it is seen AGNES created only one cluster and due to large computational time only 500 data samples are ran into AGNES whereas other algorithm was fed with 1200 datapoints.

So, if we consider without AGNES as rest of them fed with same data samples, it is seen BRICH performs better than others as it works well for large dataset. K-means and K-mediod doesn’t work well for arbitrary shaped clustering that’s why they didn’t give expected result .Reason for not finding better result in DBSCAN than K-means,k-medoids and BRICH because of arbitrary parameter- epsilon, min-points.

**Colab Notebook Link:**

[**https://colab.research.google.com/drive/1RrCqMHrLtNpohTOmJrEIavUzYyi-VC6k?usp=sharing**](https://colab.research.google.com/drive/1RrCqMHrLtNpohTOmJrEIavUzYyi-VC6k?usp=sharing)

**Reference:**

1. <https://scikit-learn.org/stable/datasets/sample_generators.html#sample-generators>
2. https://scikitlearn.org/stable/modules/generated/sklearn.cluster.KMeans.html
3. <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.Birch.html>
4. <https://github.com/Moonpuck/chameleon_cluster>
5. <https://esigma6.wordpress.com/2018/11/03/2-3-9-clustering-performance-evaluation/>