REPORT

MCS-203: DATA MINING

SURBHI MITTAL

M.Sc. Computer Science

Roll No. 43

PROBLEM STATEMENT

1. Generate 5 random numbers between [50, 100]. These will be the cluster sizes.

2. Generate bi-variate data using three variations of Gaussian distribution, one uniform distribution and one exponential distribution. Use the cluster sizes as in step1.

3. Assign class labels 1-5 to the points and, merge all the data points. Plot these where the five clusters are clearly separated. If they are not, adjust the parameters of the distributions to get the clusters separated. Each label should be one color.

4. Apply k-means, hierarchical clustering and DB Scan on the dataset separately, and plot clusters for each clustering scheme (each cluster in different color).
The number of cluster for DB Scan may be different from 5.

5. For each scheme, find Purity and SSE and print on the plots.

6. Make sure that the legend is there on the graph. (Each cluster should have different symbol).

DATASET GENERATION

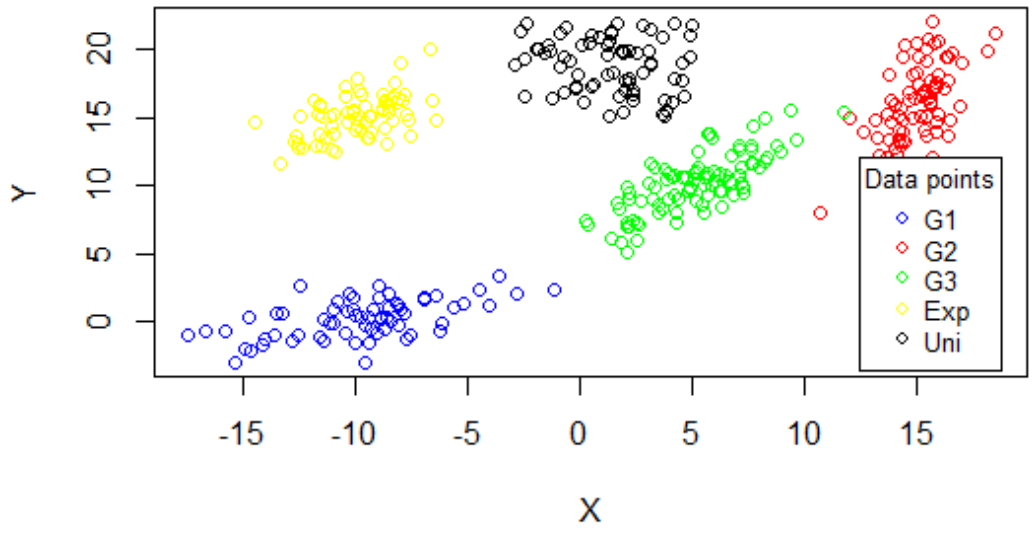
The dataset was generated using three variations of Gaussian distribution, one exponential distribution and one uniform distribution.

R functions used

S. No.	Function name	Package	Use
1.	sample	base	to generate random numbers.
2.	mvnorm	MASS	to generate bivariate normal data.
3.	rmvpe	LaplacesDemon	to generate bivariate exponential data.
4.	runif	stats	to generate uniformly distributed data.

For Gaussian and exponential distribution, covariance and mean for data was provided, whereas, for uniformly distributed data, the range was specified.

The dataset used for analysis has **381 observations.** The plot of the graph is as follows-



Where, G1: Gaussian distribution 1
G2: Gaussian distribution 2
G3: Gaussian distribution 3

Exp: Exponential distribution
Uni: Uniform distribution

CLUSTERING ALGORITHMS

K-MEANS

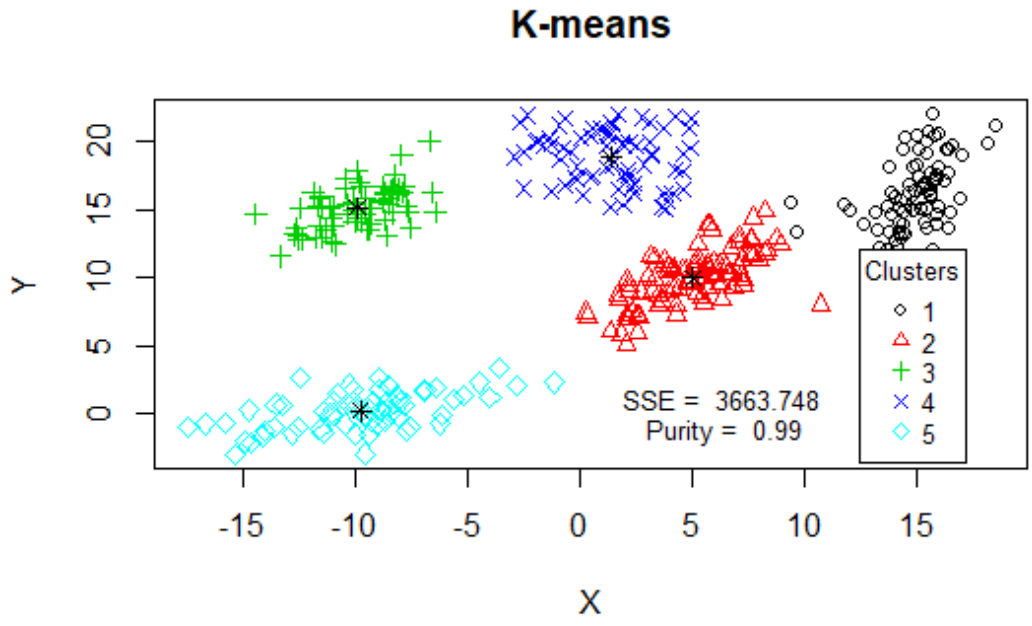
K-Means partitions the dataset into k clusters, such that each data point belongs to the cluster with the nearest mean, which serves as the representative of the cluster.

For our data, we selected, k=5, i.e. 5 clusters.

R functions used

S. No.	Function name	Package	Use
1.	k-means	stats	to apply k-means algorithm on data.

When applied on the chosen dataset, the k-means algorithm presented with the following clusters –



*The centers of the clusters have been marked with asterisks.

K-means gives high purity.

HIERARCHICAL CLUSTERING

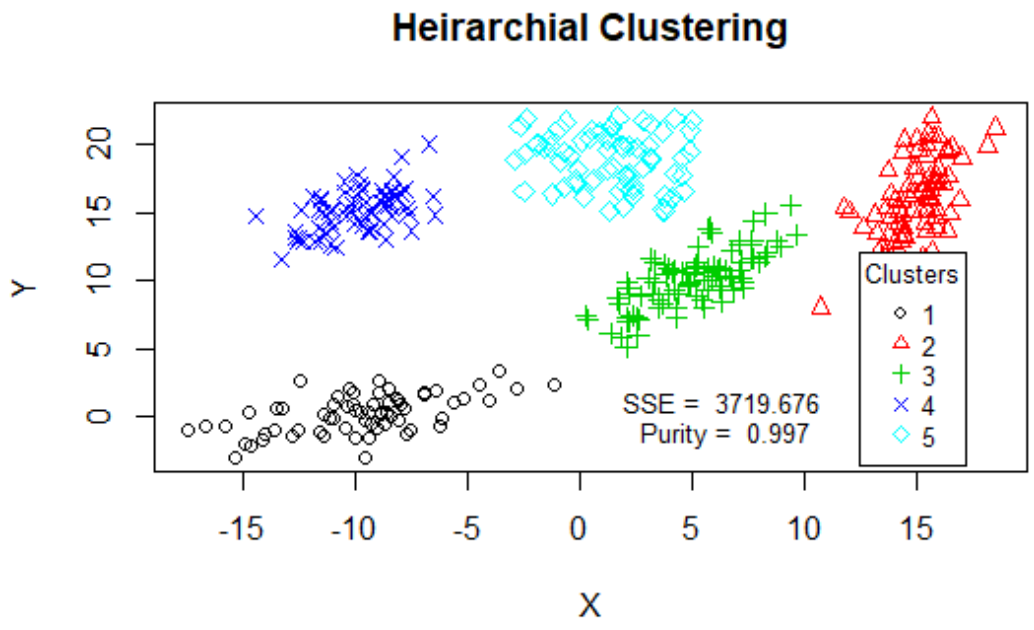
Hierarchical clustering clusters the data based on a hierarchy. Agglomerative methods were used for clustering the chosen dataset. Specifically, the D. Ward method was used.

For our data, we selected k=5, i.e. 5 clusters.

R functions used

S. No.	Function name	Package	Use
1.	hclust	stats	to create a dendogram tree.
2.	cutree	stats	to divide the dendogram into k clusters.

When applied on the chosen dataset, hierarchical clustering presented with the following clusters –



Hierarchical clustering gives high purity as well.

DBScan

DBScan is a density based clustering algorithm and is known for its ability to find non-linear clusters.

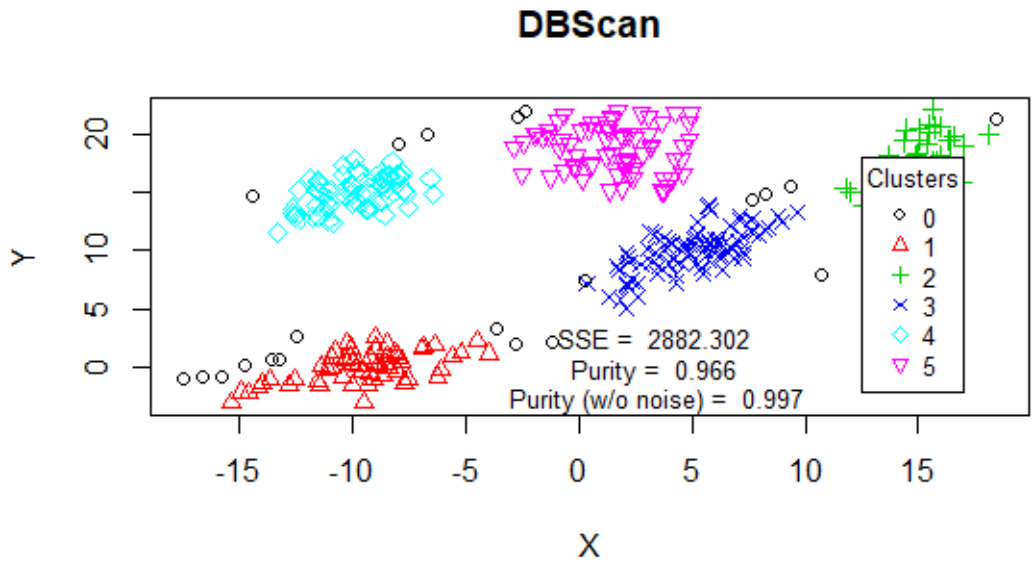
It takes as an input ε and minimum number of points required for a region to be considered as dense. Based on the ε-neighborhood of points and the number of points such a neighborhood contains, clusters are created.

R functions used

S. No.	Function name	Package	Use
1.	dbscan	dbscan	to create clusters using DBScan.

For our data, we chose ε to be 1.5 and minimum number of points to be five. We couldn't specify the number of desired clusters.

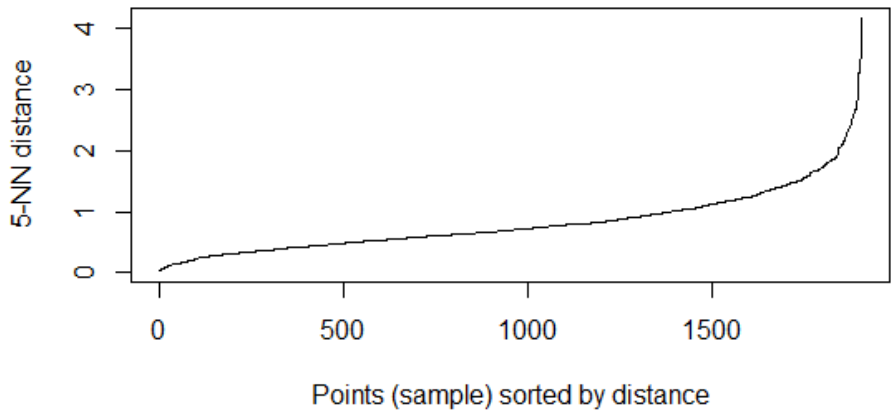
When applied on the chosen dataset, DBScan presented with the following clusters –



The noise points are plotted in black.

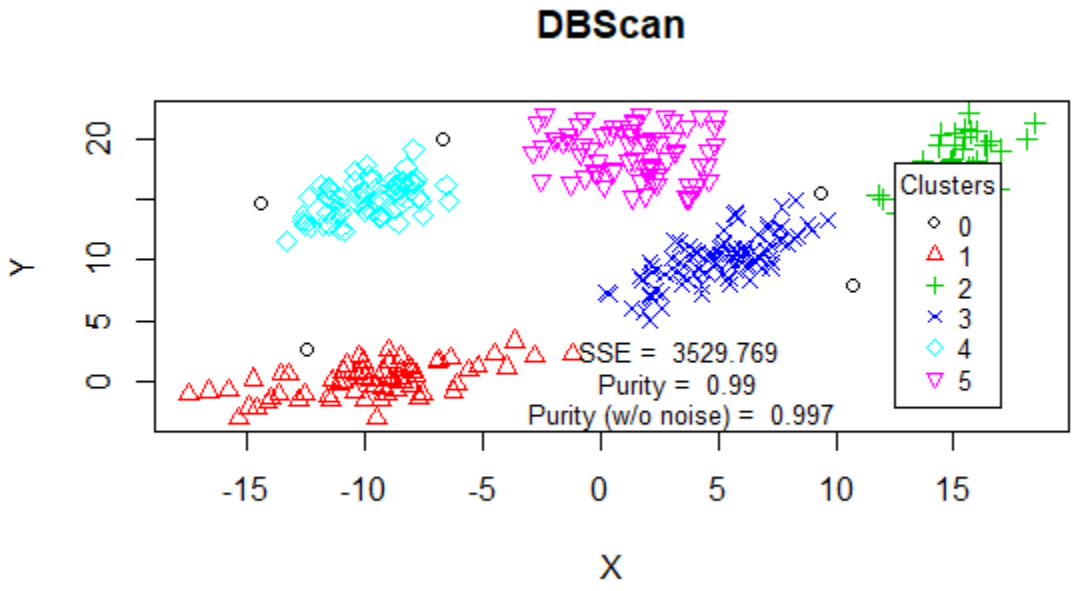
DBScan gives a good purity, as well.

On checking the KNN plot for k=5, we get the following graph,



At the knee of the KNN plot, we get the optimal value of ε for DBScan.

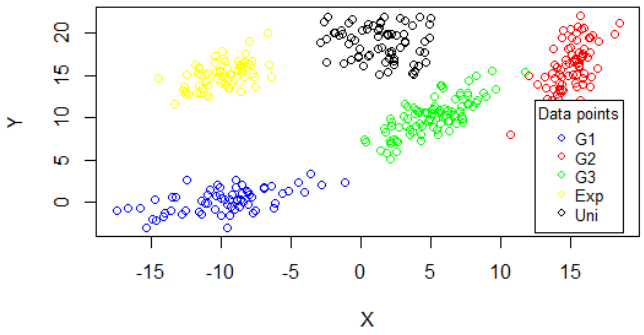
The optimal value of ε seems to be around 2. On giving ε =2, we get the following plot –

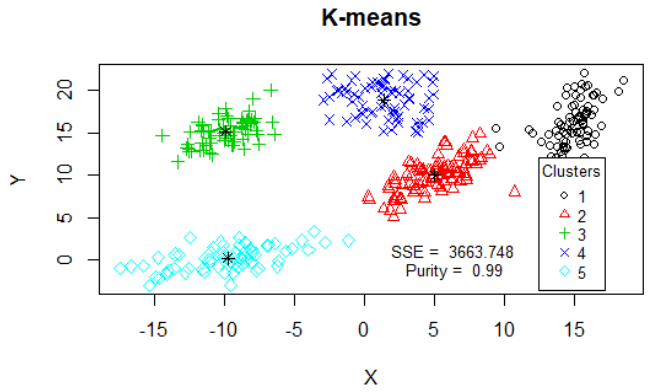


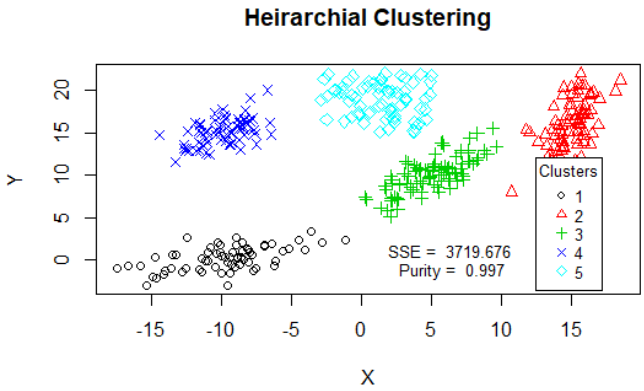
On plotting, the number of noise points reduces, but there is no significant increase in purity.

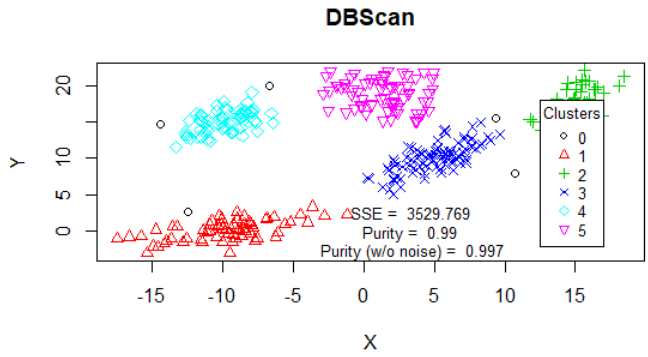
DBScan gives high purity.

OBSERVATIONS









S. No.	Algorithm	SSE	Purity
1.	K-means	3663.748	0.99
2.	Hierarchical	3719.676	0.997
3.	DBScan	3529.769	0.997

We observe all algorithms give high purity. Hierarchical and DBScan give the highest purity of 99.7%.

All clustering schemes give nearly the same SSE but DBScan gives the least SSE (noise points have been ignored).

Why we get these results?

About the dataset

The dataset we chose had 5 data distributions. All distributions were nearly clearly separated.

All clustering schemes gave good results because the data distributions in the dataset were clearly separated. DBScan gave the best results, because all distributions had similar density and the noise points were disregarded from the calculation of SSE.

K-means and hierarchical clustering give good results as well.

*SSE has been calculated by-

1. Finding means (centroids) of all clusters.
2. Calculating error from the point to centroid (of the cluster to which it belongs).
3. Squaring and adding.

BIBLIOGRAPHY

- www.google.com
- www.rdocumentation.org
- https://stat.ethz.ch/R-manual/
- https://stackoverflow.com/