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

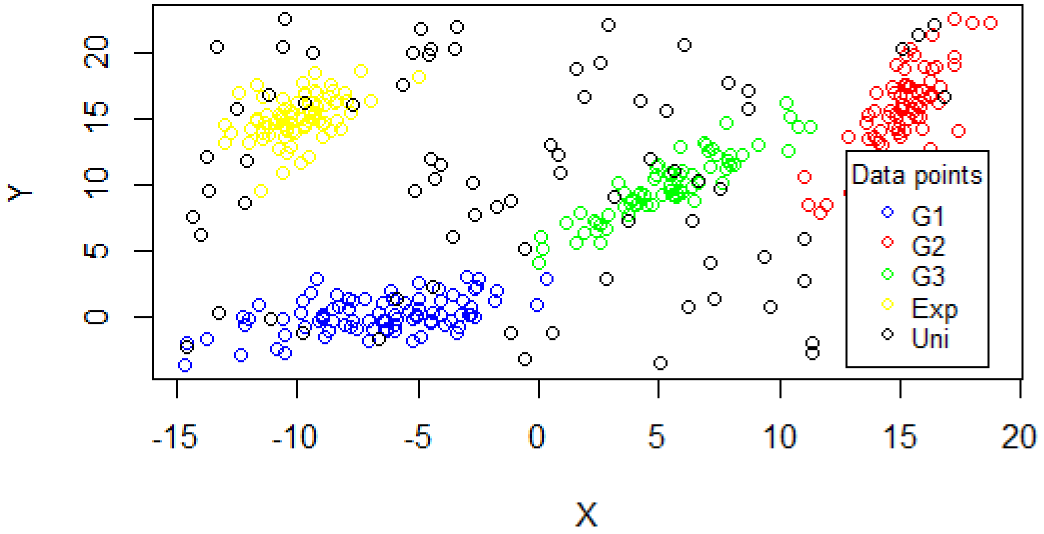
The dataset was generated using three variations of Gaussian distribution, one exponential distribution and one uniform distribution.

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 **413 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

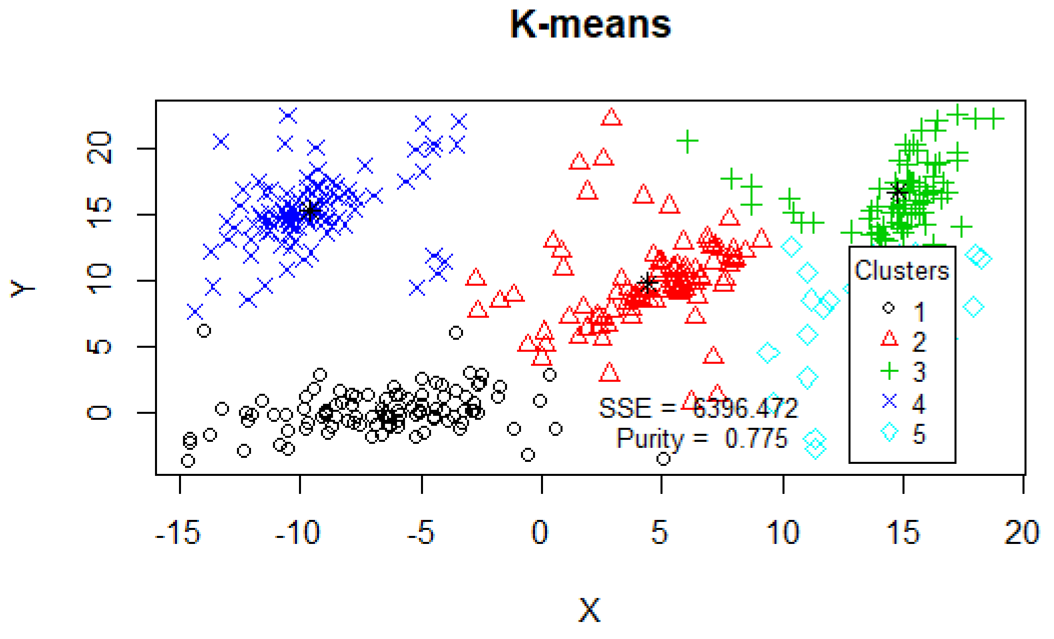
K-Means partitions the dataset into k clusters, such that each data point belongs to the cluster with the nearest mean, which serves as the representative of the cluster.

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.

HIERARCHICAL CLUSTERING

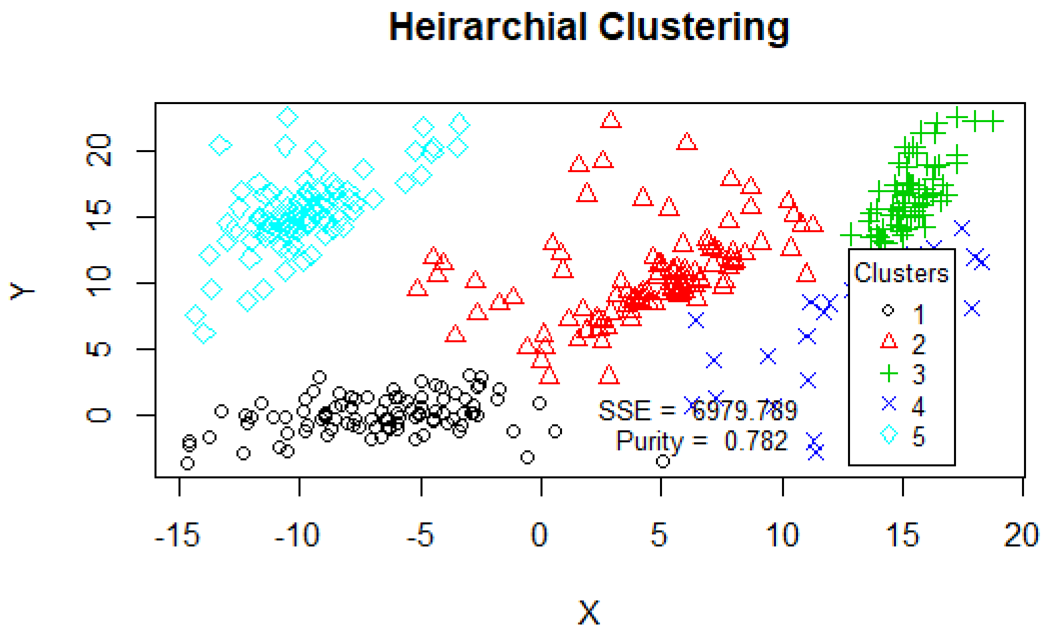
Hierarchical clustering clusters the data based on a hierarchy. Agglomerative methods were used for clustering the chosen dataset. Specifically, the D. Ward method was used.

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 –



DBScan

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

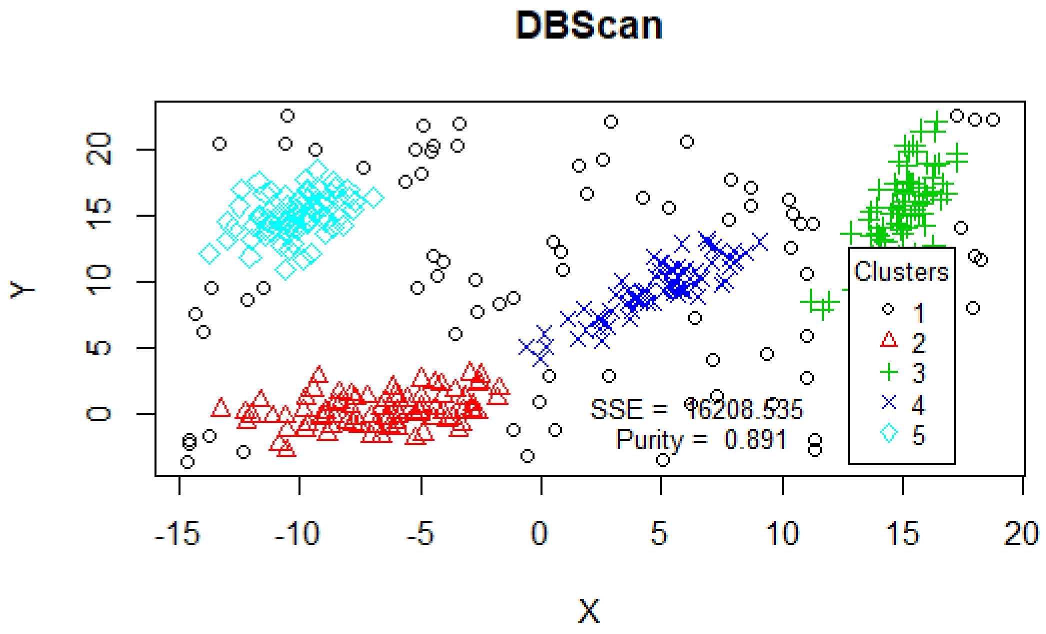
It takes as an input ε and minimum number of points required for a region to be considered as dense. Based on the ε-neighborhood of points and the number of points such a neighborhood contains, clusters are created.

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.

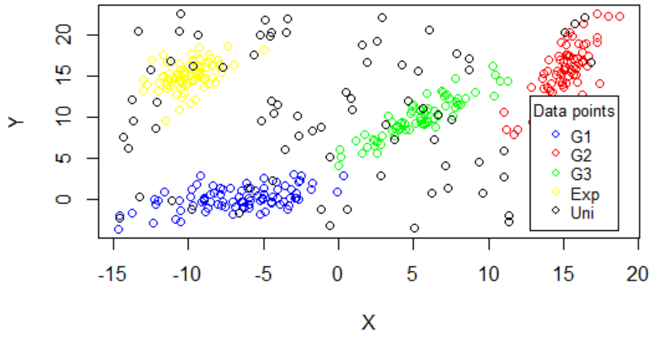
R functions used

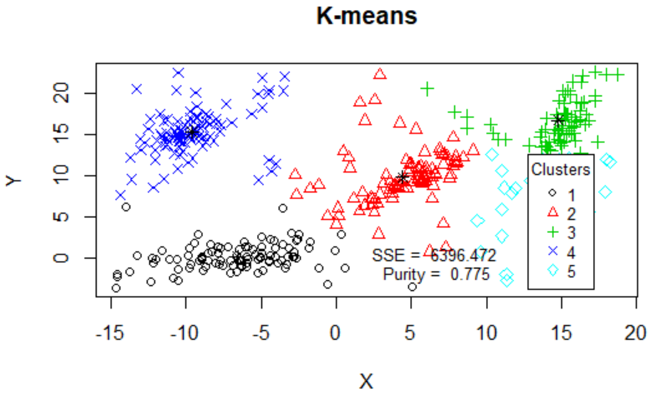
S. No.	Function name	Package	Use
1.	dbscan	dbscan	to create clusters using DBScan.

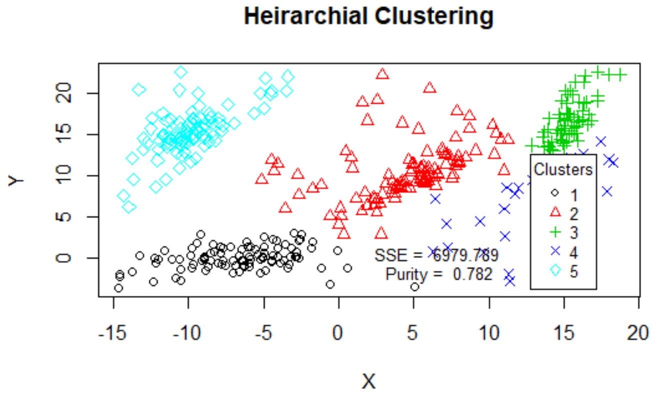
When applied on the chosen dataset, DBScan presented with the following clusters –

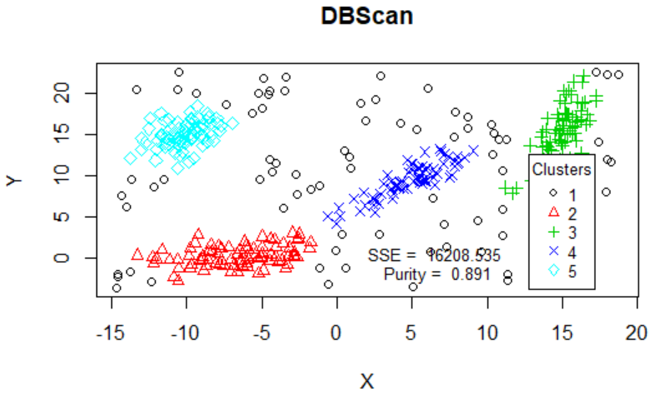


OBSERVATIONS









S. No.	Algorithm	SSE	Purity
1.	K-means	6396.472	0.776
2.	Hierarchical	6879.769	0.782
3.	DBScan	16208.535	0.891

We observe that DBScan gives the highest purity (approx. 90%) for the chosen data set. But, its SSE (squared sum of errors) is maximum.

K-means and Hierarchical based clustering, give almost the same purity (approx. 80%) for this dataset. Their SSEs are also approximately equal.

Why DBScan gives the best results?

About the dataset

The dataset we chose had 5 data distributions, one out of which was uniform. The uniform data was spread uniformly across the entire range, as can be observed from the first plot.

Both K-means and hierarchical clustering divide the data into clusters based on some kind of distance heuristic. Both algorithms provided with similar clusters (as can be observed from the plots), where hierarchical clustering turned out to be a little better (observe purity and SSE).

In the case of this dataset, DBScan provided the best results, (purity of approx. 90%). One of the reasons for DBScan's success is the fact that it clusters data based on density, rather than distance. The uniform distribution in our data, could be separated into a cluster because of its density.
We observed the SSE* of DBScan to be very high, which is probably the result of calculating distances of points from the center of the uniformly distributed cluster. Since, the cluster is spread throughout the range (observe plot), distance of all points from the center of the cluster must be very high.

*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/