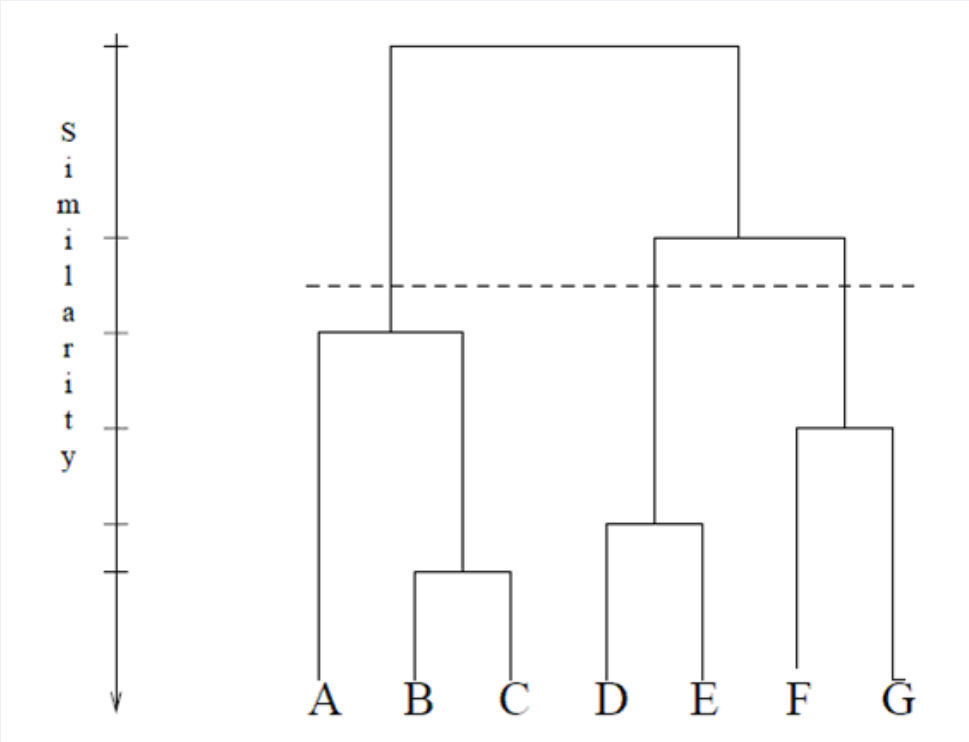
Linkage clustering

Mahdi Akbari Zarkesh
9612762638

Hierarchical clustering:

In hierarchical clustering, we assign each object (data point) to a separate cluster. Then compute the distance (similarity) between each of the clusters and join the two most similar clusters. Let's understand further by solving an example.

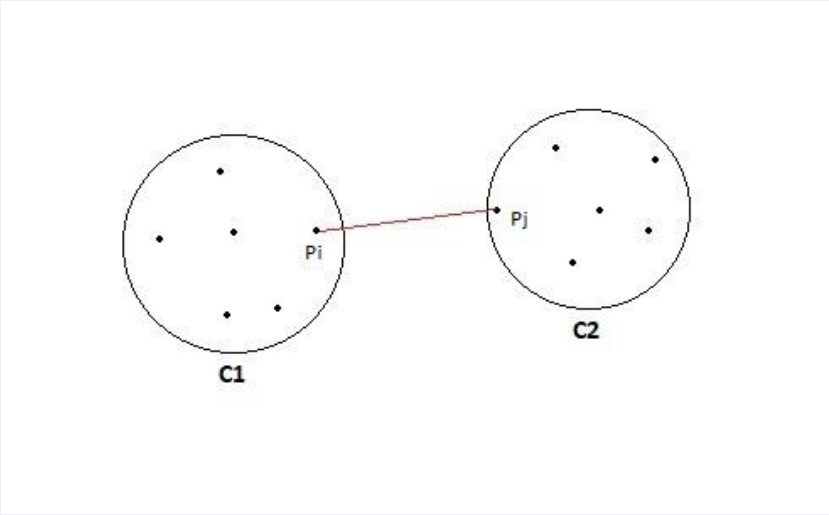


"HOW DO WE CALCULATE THE SIMILARITY BETWEEN TWO CLUSTERS???"

Calculating the similarity between two clusters is important to merge or divide the clusters. There are certain approaches which are used to calculate the similarity between two clusters:

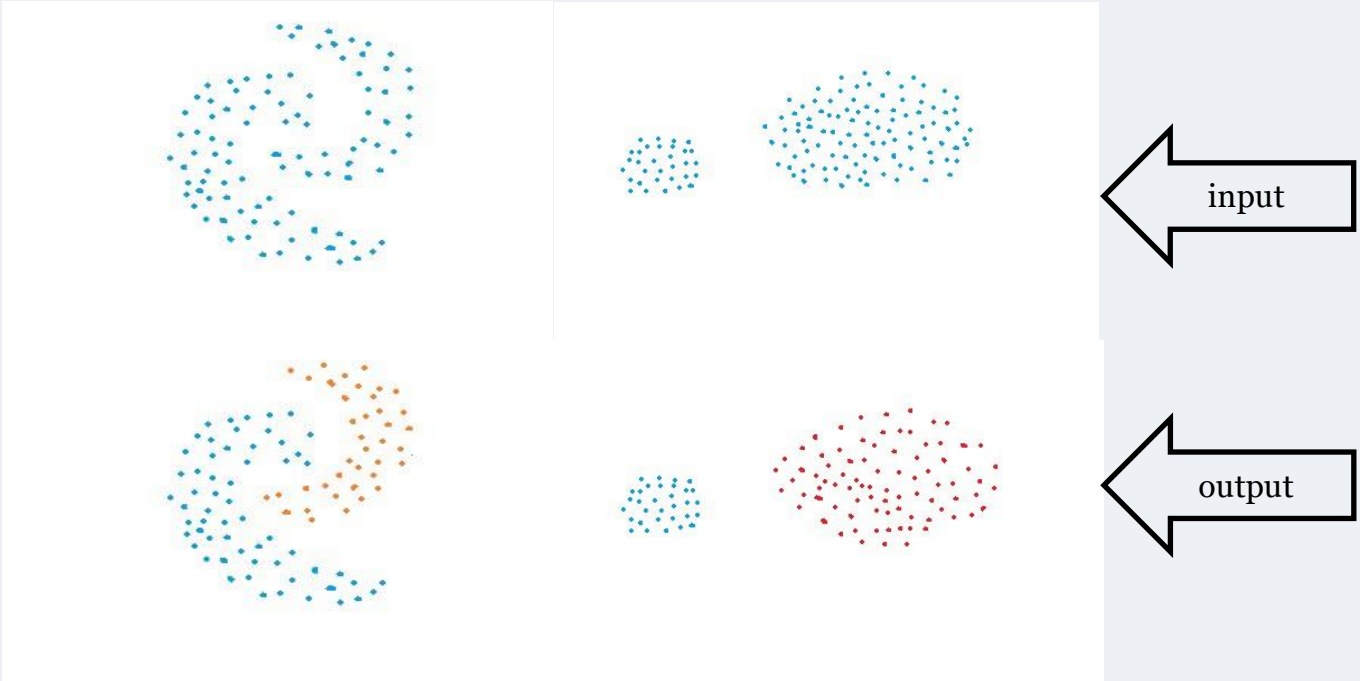
- MIN
- MAX
- Group Average
- Distance Between Centroids
- Ward's Method
- ...

MIN: Also known as single linkage algorithm can be defined as the similarity of two clusters C1 and C2 is equal to the minimum of the similarity between points Pi and Pj such that Pi belongs to C1 and Pj belongs to C2.
In simple words, pick the two closest points such that one point lies in cluster one and the other point lies in cluster 2 and take their similarity and declare it as the similarity between two clusters.



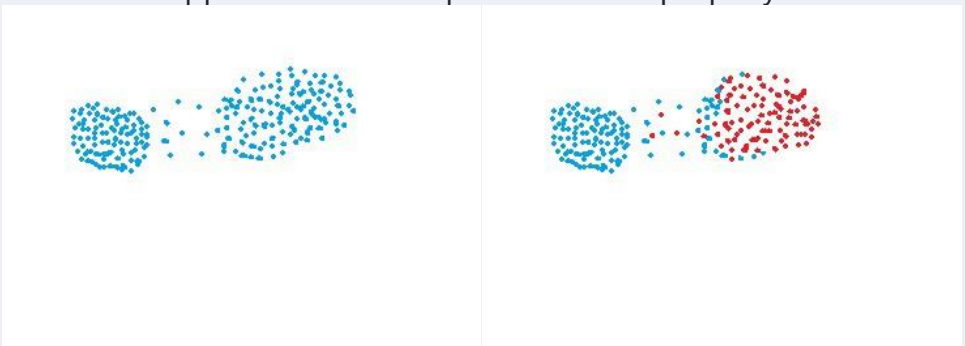
Pros of MIN:

- This approach can separate non-elliptical shapes as long as the gap between two clusters is not small.



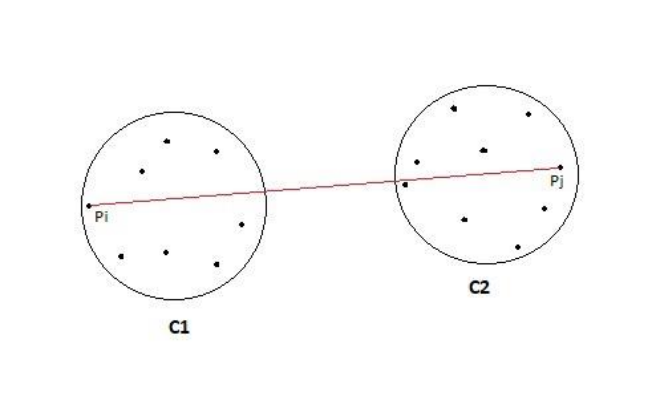
Cons of MIN:

- MIN approach cannot separate clusters properly if there is noise between clusters.



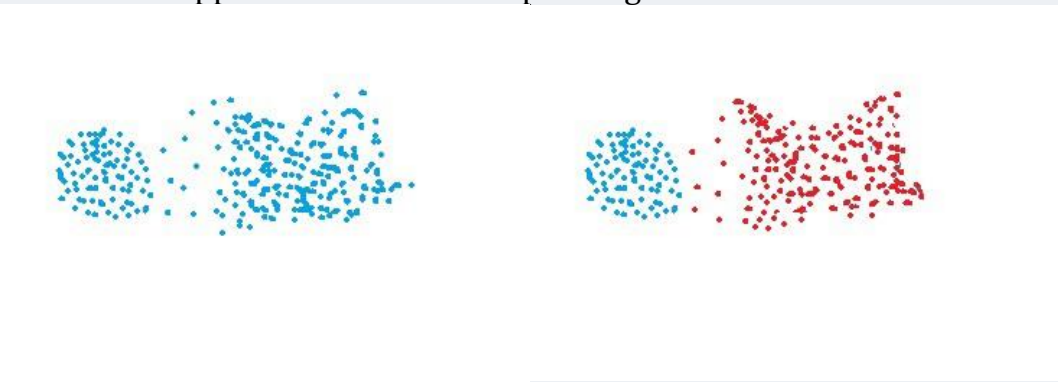
MAX: Also known as the complete linkage algorithm, this is exactly opposite to the MIN approach. The similarity of two clusters C1 and C2 is equal to the maximum of the similarity between points Pi and Pj such that Pi belongs to C1 and Pj belongs to C2.

Mathematically this can be written as,
$Sim(C1,C2) = Max\ Sim(Pi,Pj)$ such that $Pi \in C1$ & $Pj \in C2$
In simple words, pick the two farthest points such that one point lies in cluster one and the other point lies in cluster 2 and take their similarity and declare it as the similarity between two clusters.



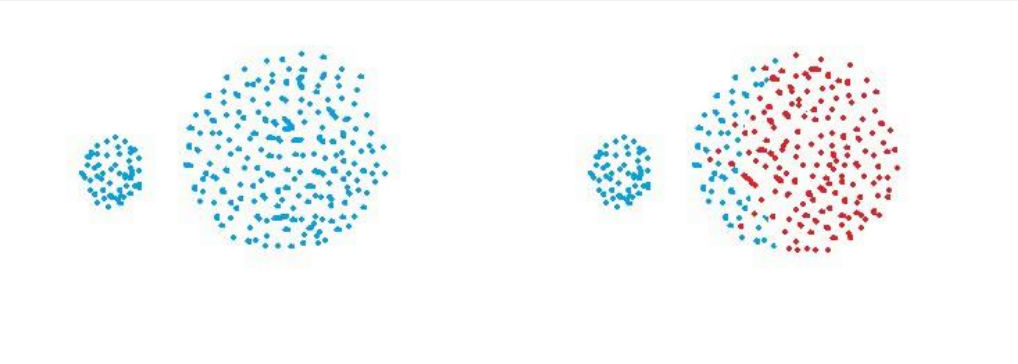
Pros of MAX:

- MAX approach does well in separating clusters if there is noise between clusters.

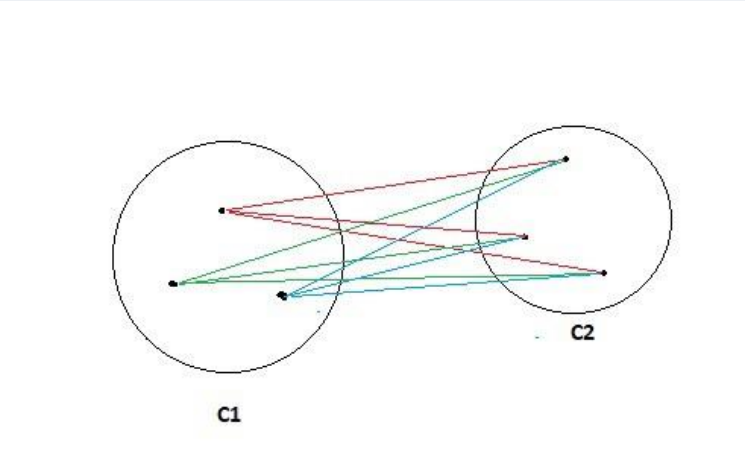


Cons of Max:

- Max approach is biased towards globular clusters.
- Max approach tends to break large clusters.



Group Average: Take all the pairs of points and compute their similarities and calculate the average of the similarities.
Mathematically this can be written as,
$sim(C1,C2) = \sum sim(Pi, Pj)/|C1|*|C2|$
where, $Pi \in C1$ & $Pj \in C2$

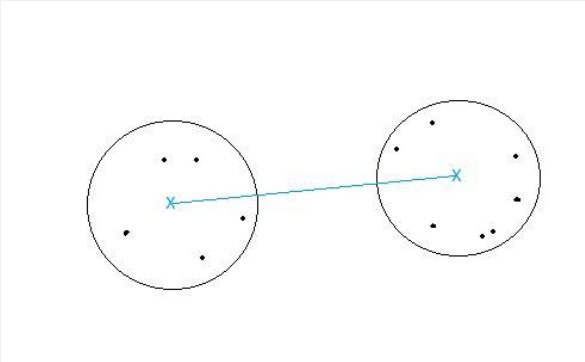


Pros of Group Average:

- The group Average approach does well in separating clusters if there is noise between clusters.

Cons of Group Average:

- The group Average approach is biased towards globular clusters.
- Distance between centroids: Compute the centroids of two clusters C1 & C2 and take the similarity between the two centroids as the similarity between two clusters. This is a less popular technique in the real world.



Ward's Method: This approach of calculating the similarity between two clusters is exactly the same as Group Average except that Ward's method calculates the sum of the square of the distances Pi and PJ.
Mathematically this can be written as,
$sim(C1,C2) = \sum (dist(Pi, Pj))^2/|C1|*|C2|$

Pros of Ward's method:

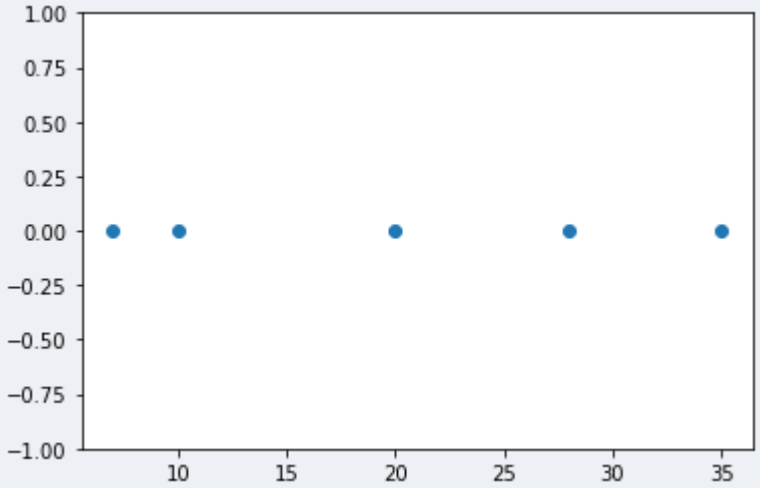
- Ward's method approach also does well in separating clusters if there is noise between clusters.

Cons of Ward's method:

- Ward's method approach is also biased towards globular clusters.

Example:

Objective : For the one dimensional data set {7,10,20,28,35}, perform hierarchical clustering and plot the dendogram to visualize it.
Solution : First, let's the visualize the data.

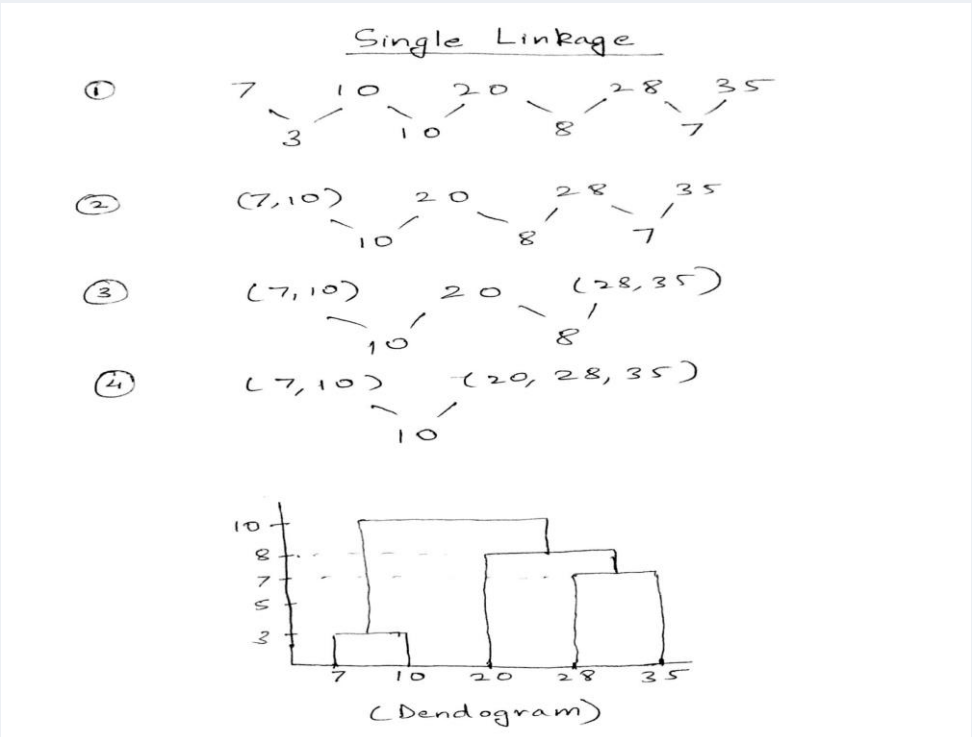


Observing the plot above, we can intuitively conclude that:

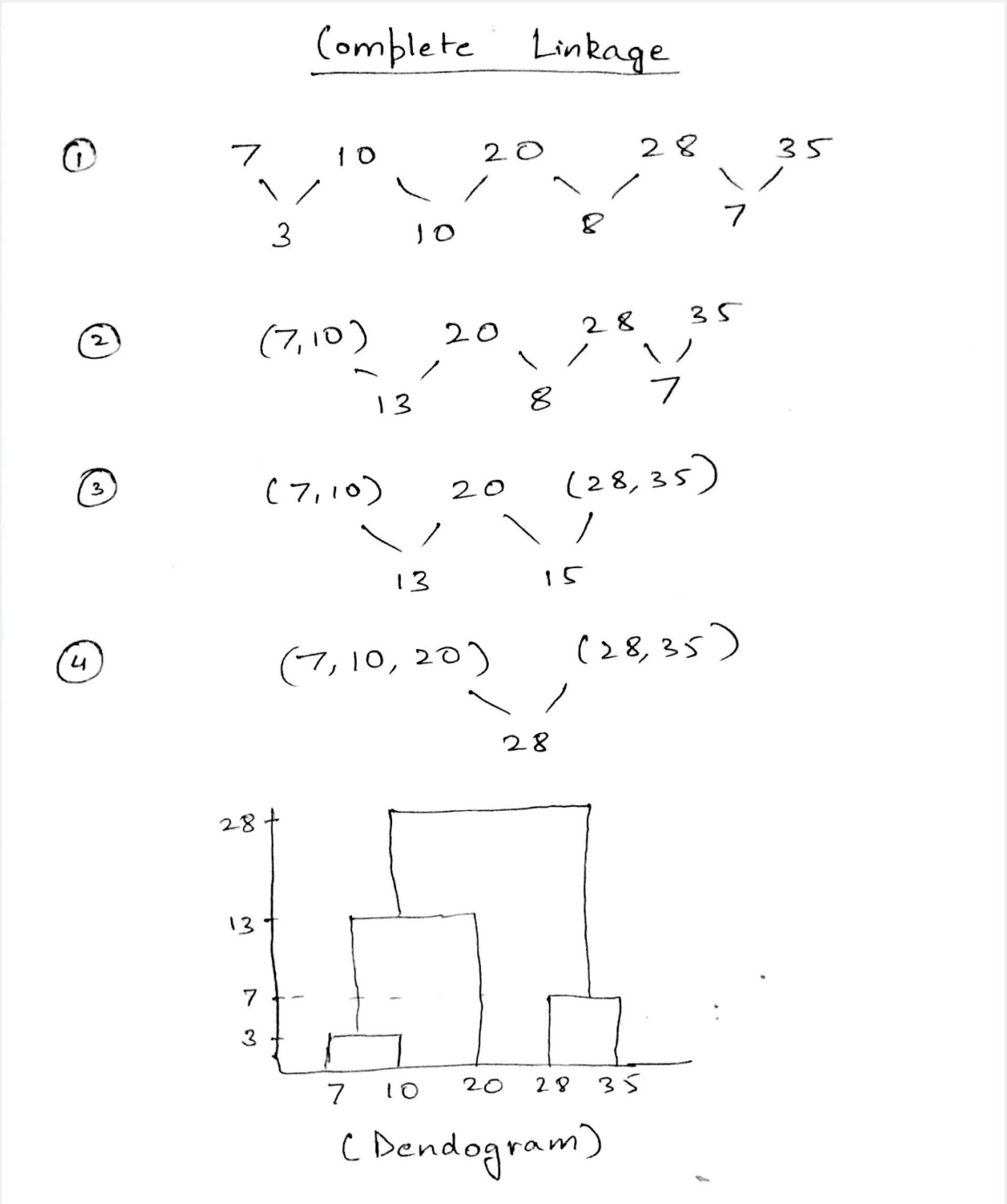
1. The first two points (7 and 10) are close to each other and should be in the same cluster
2. Also, the last two points (28 and 35) are close to each other and should be in the same cluster
3. Cluster of the center point (20) is not easy to conclude

agglomerative hierarchical clustering :

1) **Single Linkage** : In single link hierarchical clustering, we merge in each step the two clusters, whose two closest members have the smallest distance.



2) **Complete Linkage** : In complete link hierarchical clustering, we merge in the members of the clusters in each step, which provide the smallest maximum pairwise distance.



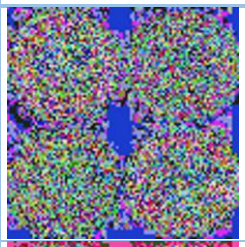
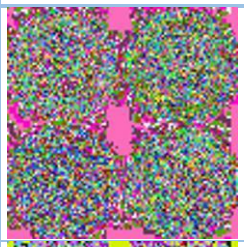
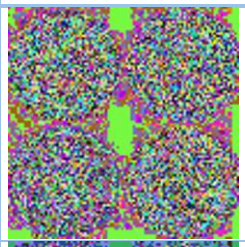
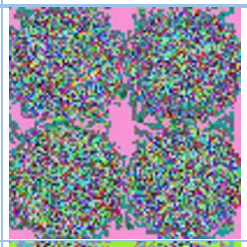
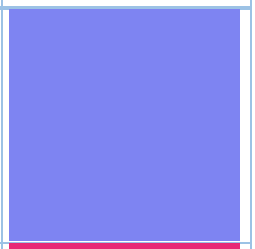
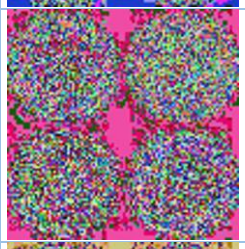
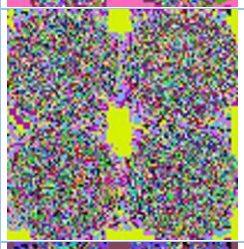
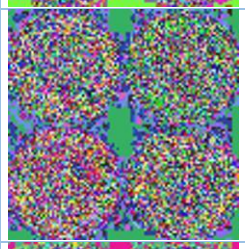
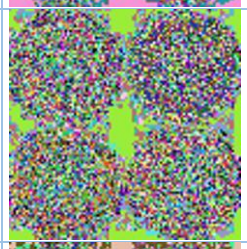
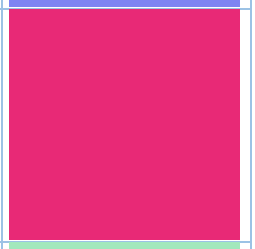
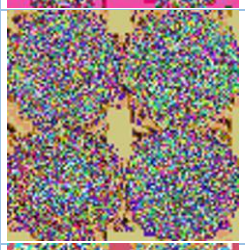
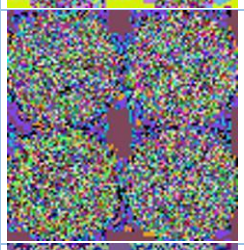
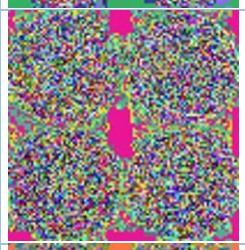
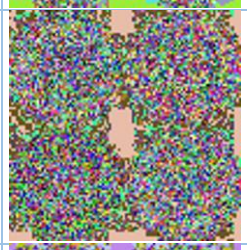
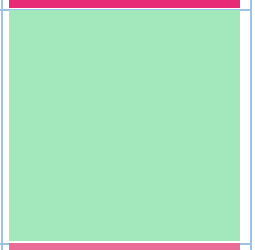
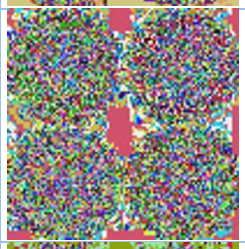
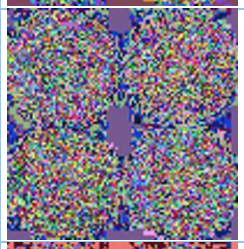
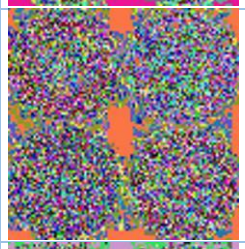
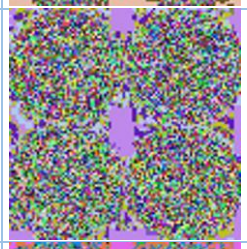
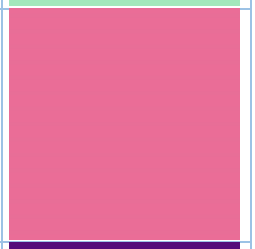
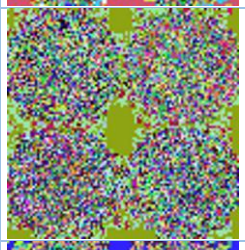
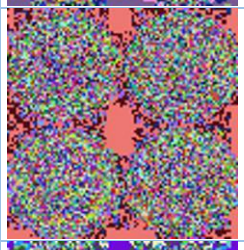
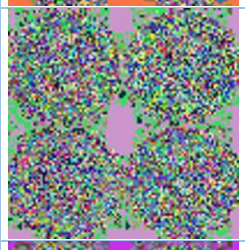
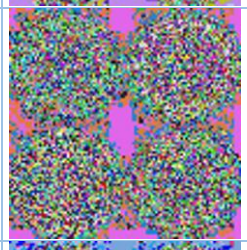
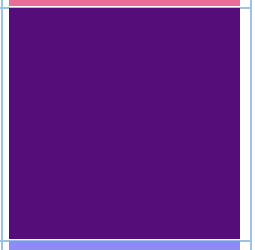
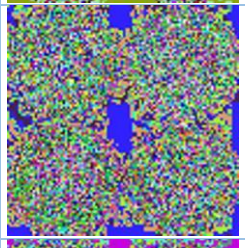
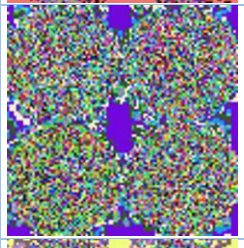
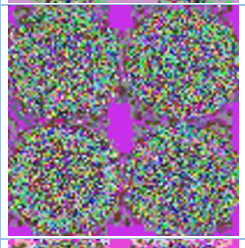
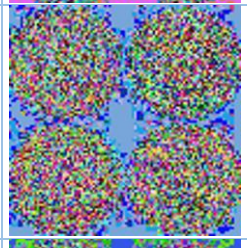
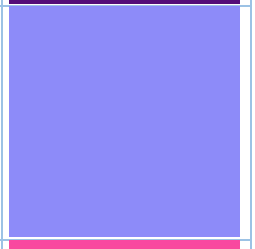
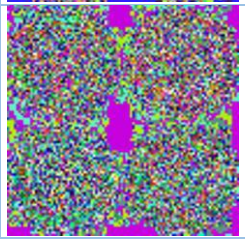
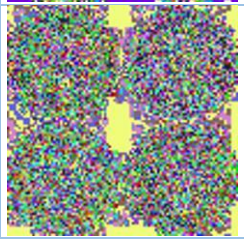
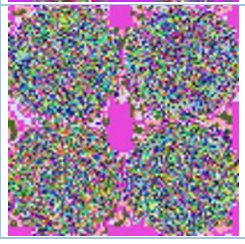
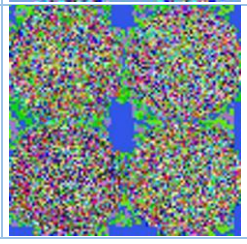
A dendrogram is a diagram that represents a tree. This chart display is often used in a variety of contexts:
• In hierarchical clustering, the order in which the clusters produced by the relevant analyzes are arranged.
• In computational biology, the clustering of genes or specimens sometimes shows thermal maps in the margins.
• In phylogeny, it shows evolutionary relationships between different biological species. In this case, the dendrogram is also called the phylogenetic tree.

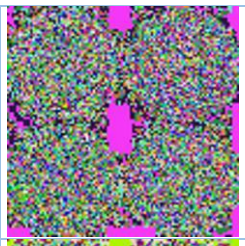
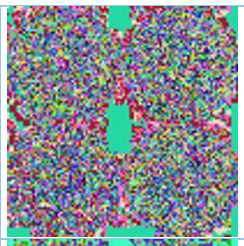
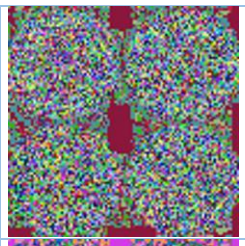
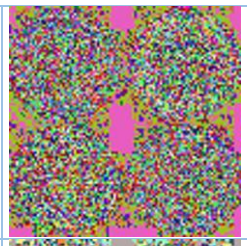
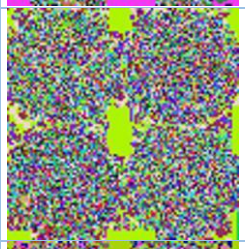
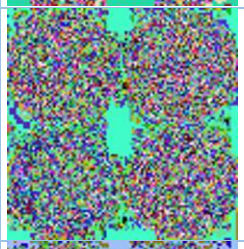
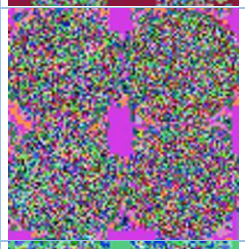
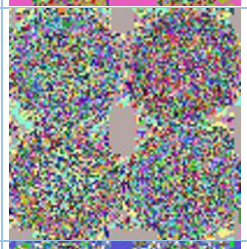
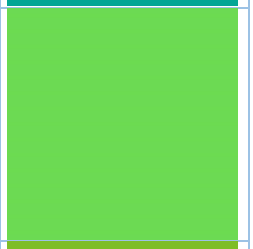
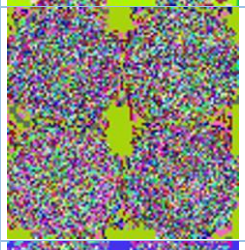
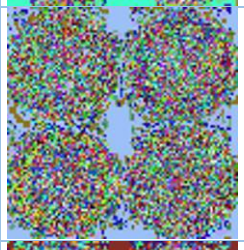
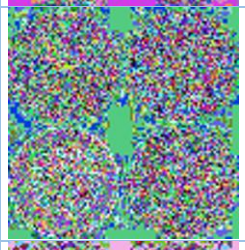
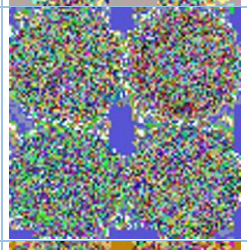
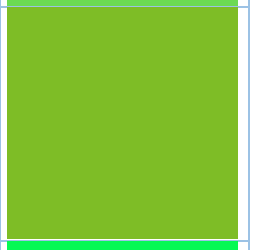
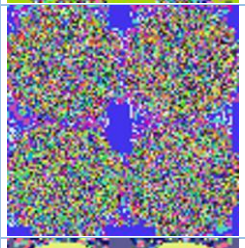
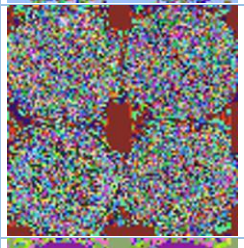
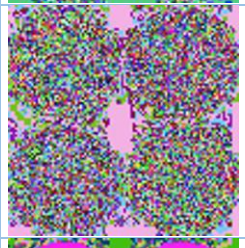
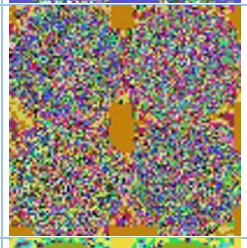
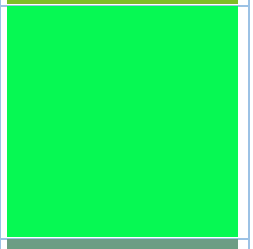
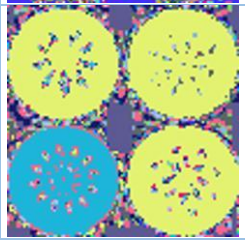
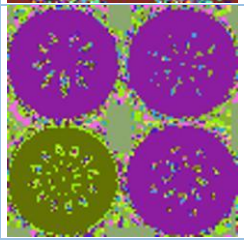
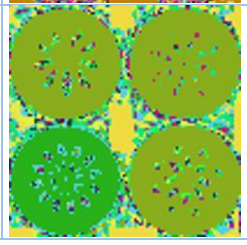
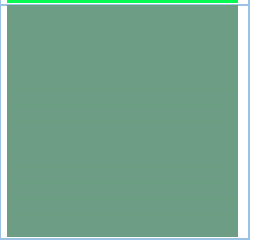
There are many ways to calculate the distance between two pairs of data, the most appropriate of which can be chosen according to the type of data.

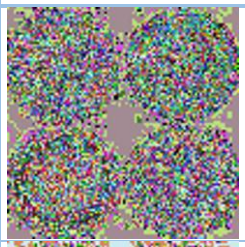
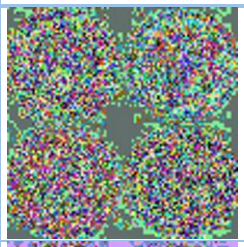
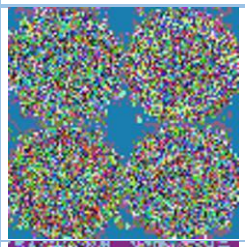
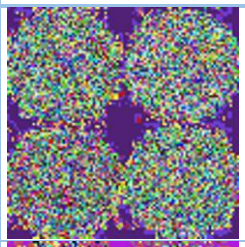
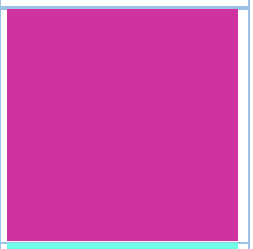
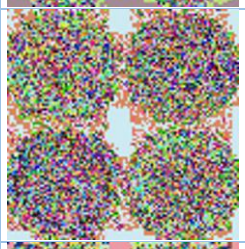
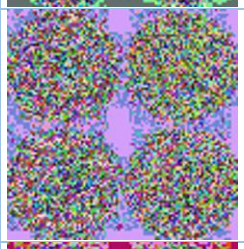
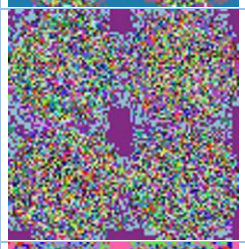
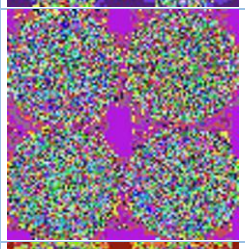
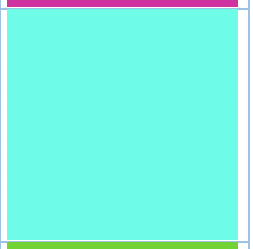
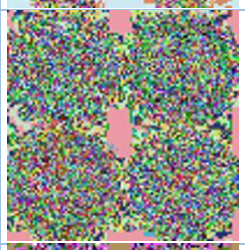
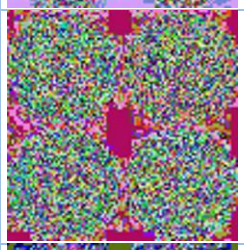
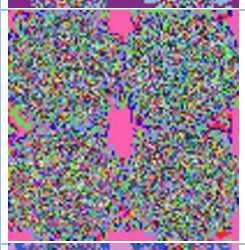
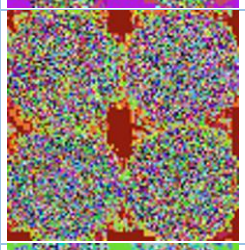
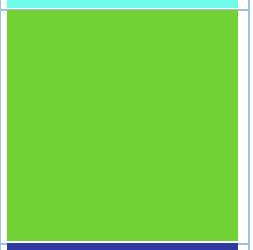
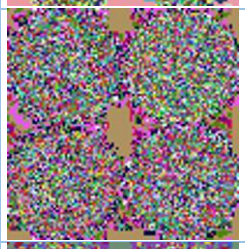
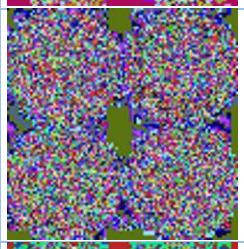
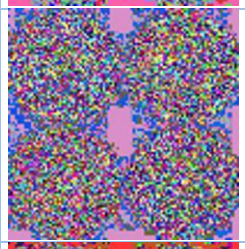
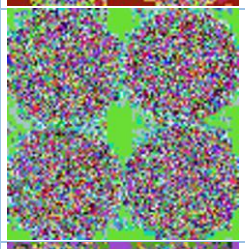
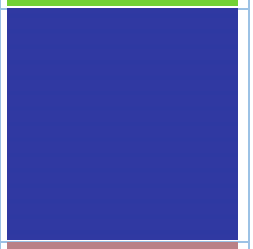
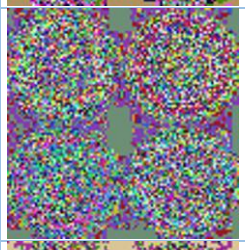
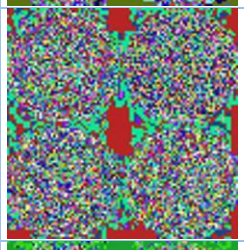
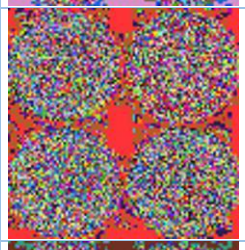
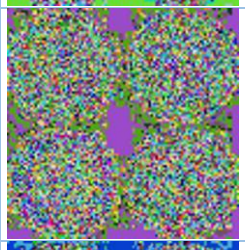
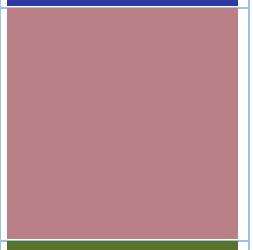
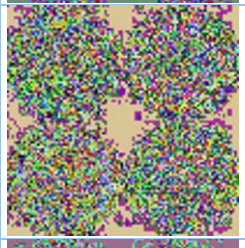
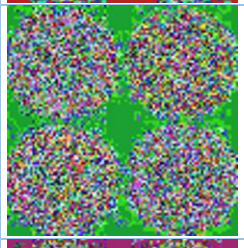
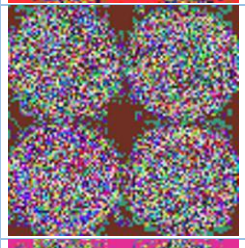
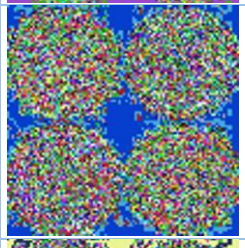
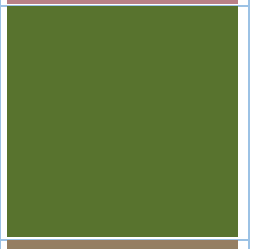
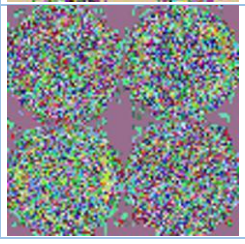
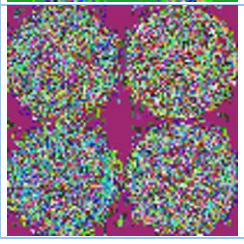
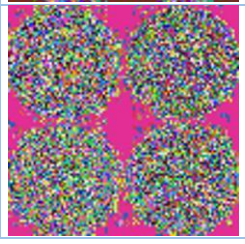
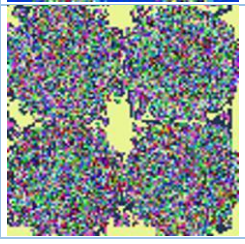
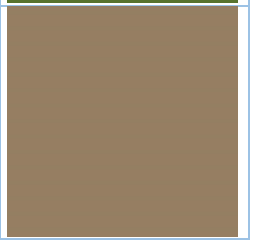
Value	Description
`'euclidean'`	Euclidean distance (default).
`'squaredeuclidean'`	Squared Euclidean distance. (This option is provided for efficiency only. It does not satisfy the triangle inequality.)
`'seuclidean'`	Standardized Euclidean distance. Each coordinate difference between obse dividing by the corresponding element of the standard deviation, `S = nans` Use `DistParameter` to specify another value for `S`.
`'mahalanobis'`	Mahalanobis distance using the sample covariance of `X`, `C = nancov(X)`. to specify another value for `C`, where the matrix `C` is symmetric and positive
`'cityblock'`	City block distance.
`'minkowski'`	Minkowski distance. The default exponent is 2. Use `DistParameter` to specify a different exponent `P`, where `P` is a positiv of the exponent.
`'chebychev'`	Chebychev distance (maximum coordinate difference).
`'cosine'`	One minus the cosine of the included angle between points (treated as vect
`'correlation'`	One minus the sample correlation between points (treated as sequences of
`'hamming'`	Hamming distance, which is the percentage of coordinates that differ.
`'jaccard'`	One minus the Jaccard coefficient, which is the percentage of nonzero coor
`'spearman'`	One minus the sample Spearman's rank correlation between observations (treated as sequences of values).

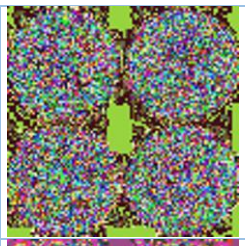
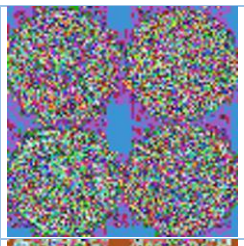
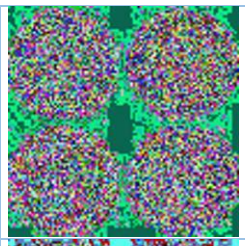
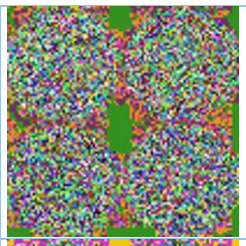
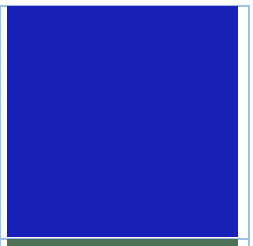
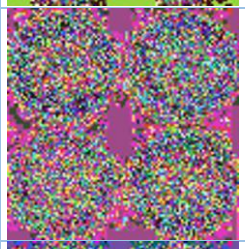
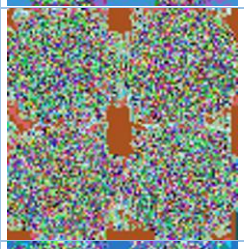
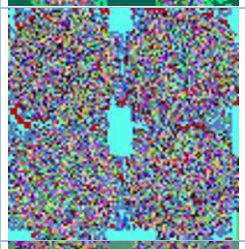
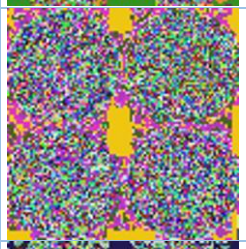
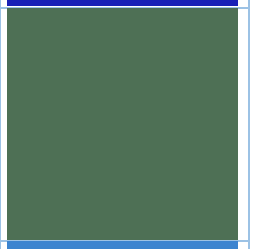
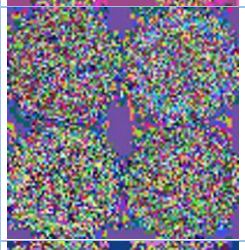
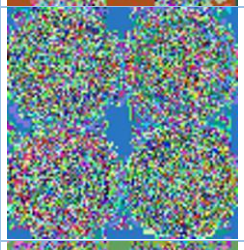
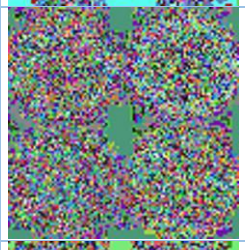
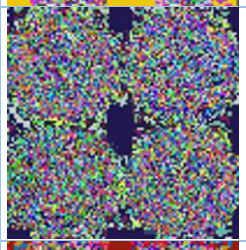
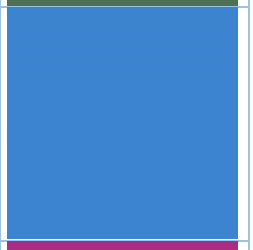
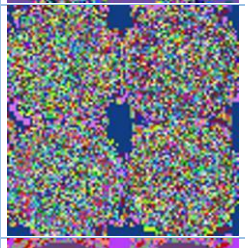
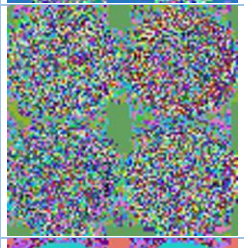
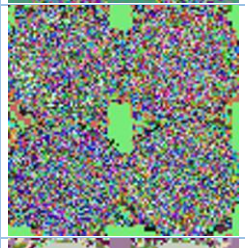
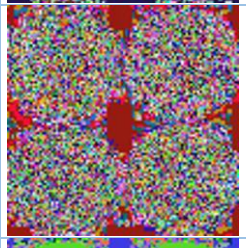
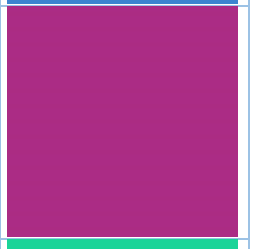
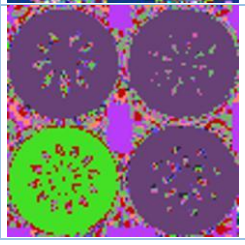
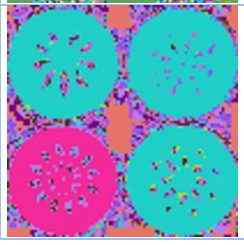
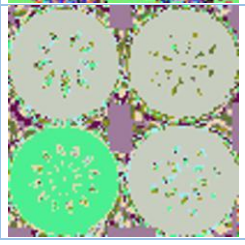
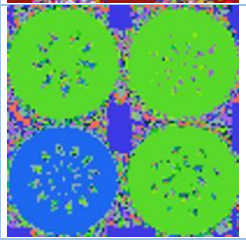
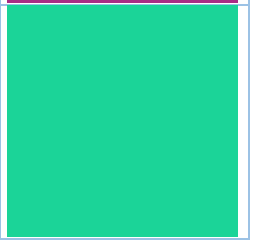
Our work

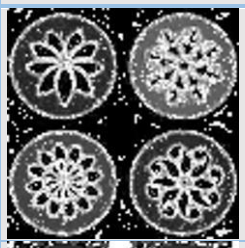
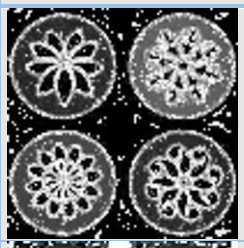
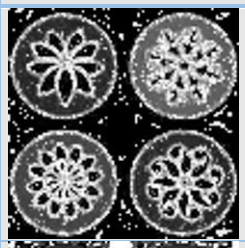
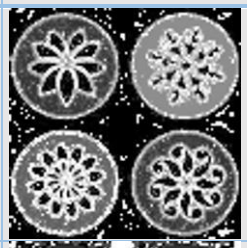
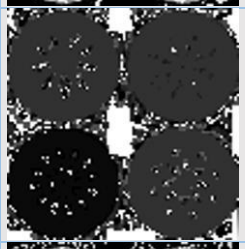
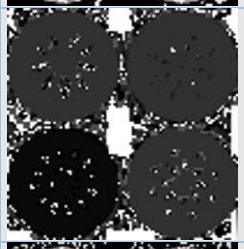
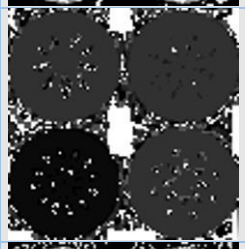
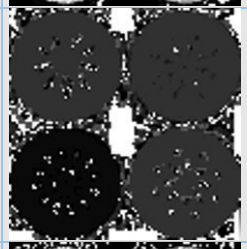
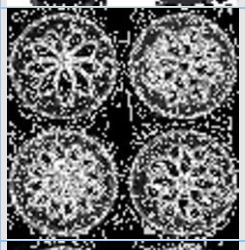
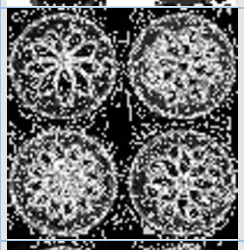
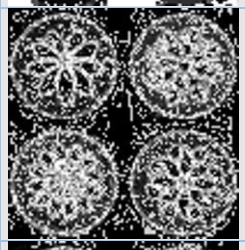
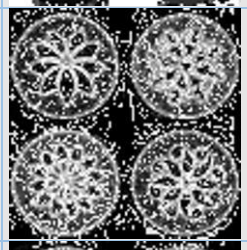
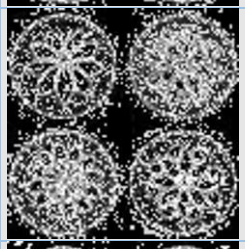
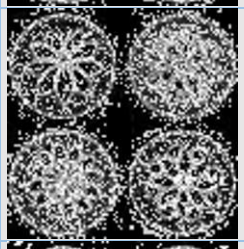
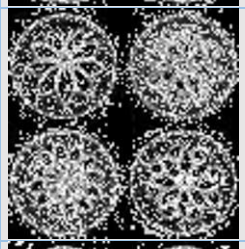
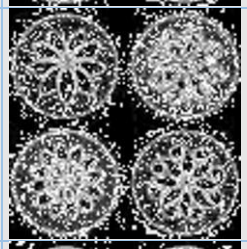
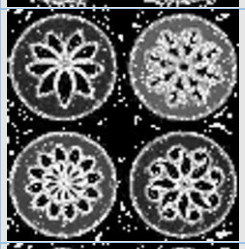
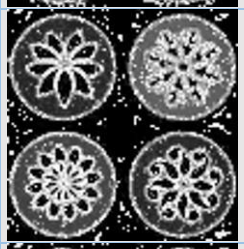
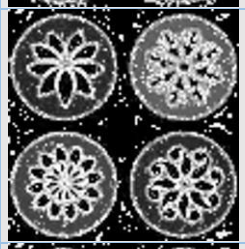
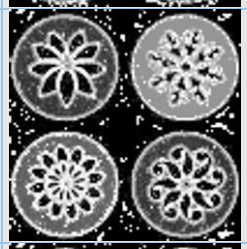
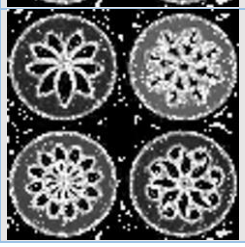
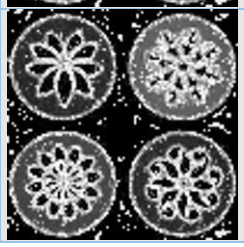
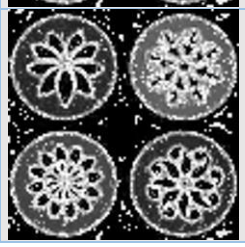
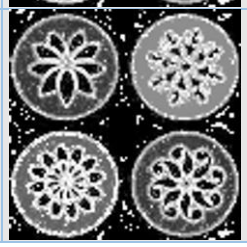
We use this picture for data

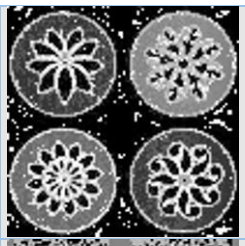
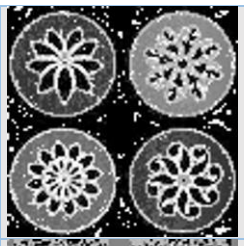
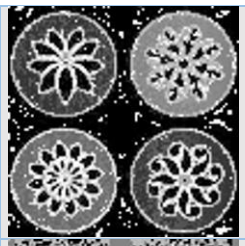
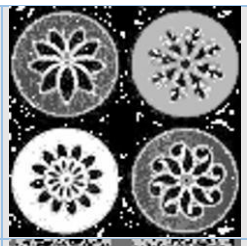
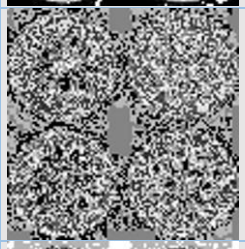
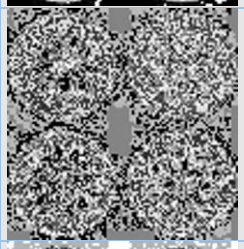
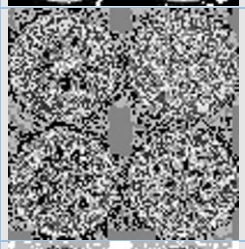
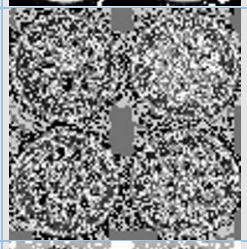
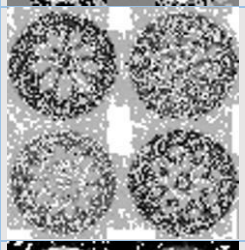
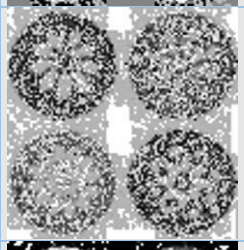
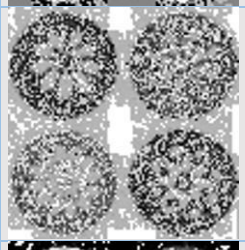
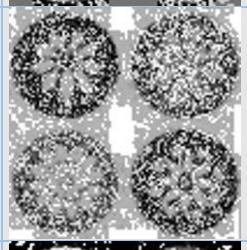
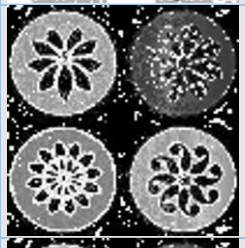
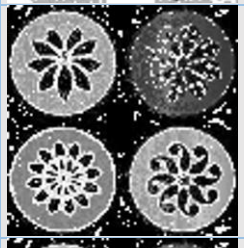
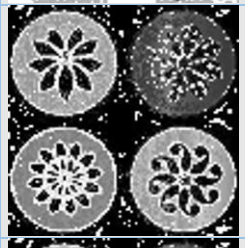
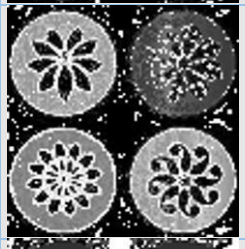
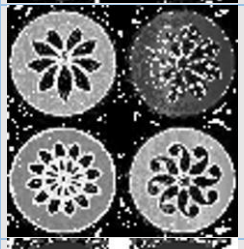
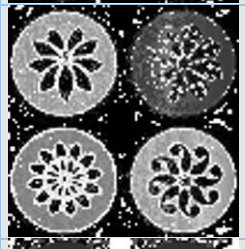
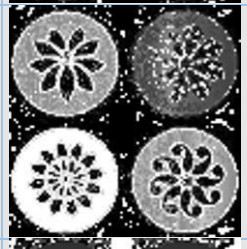
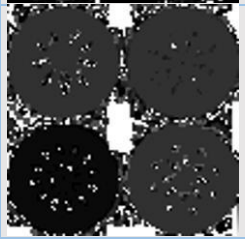
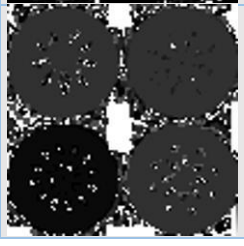
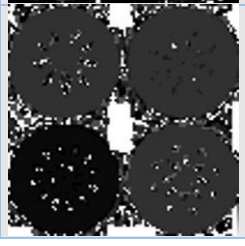
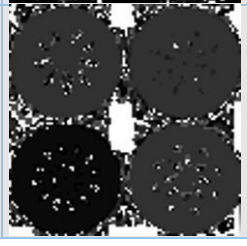
complete	*cutoff = 0.1*	*cutoff = 0.25*	*cutoff = 0.5*	*cutoff = 1*	*cutoff = 5*
euclidean					
squaredeuclidean					
Seuclidean					
Mahalanobis					
Cityblock					
Minkowski					
chebychev					

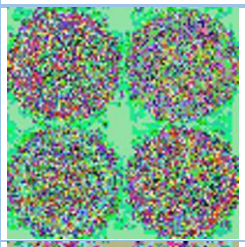
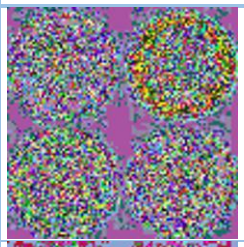
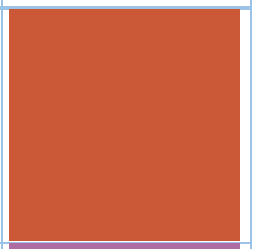
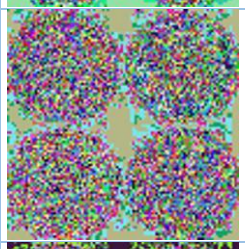
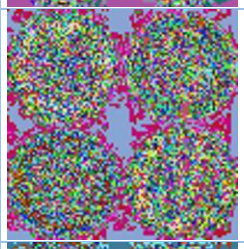
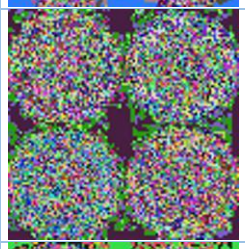
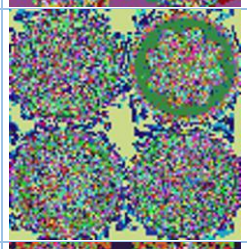
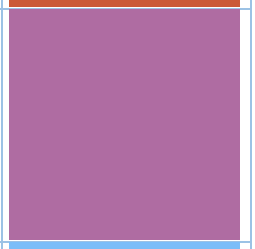
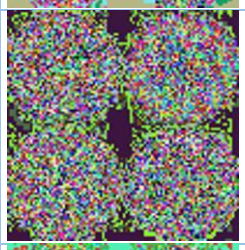
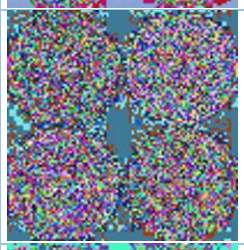
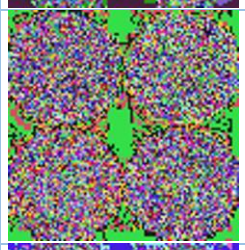
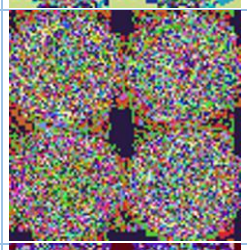
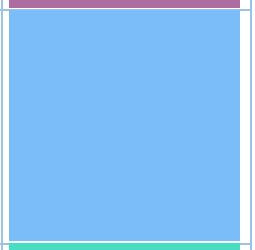
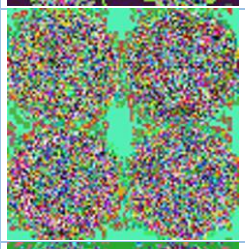
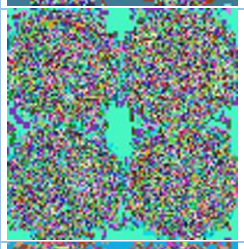
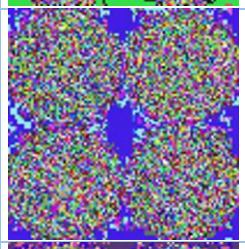
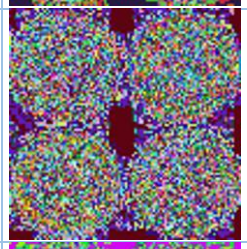
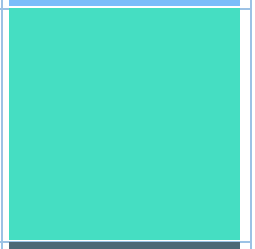
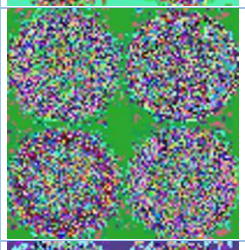
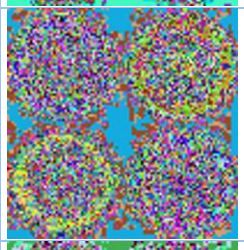
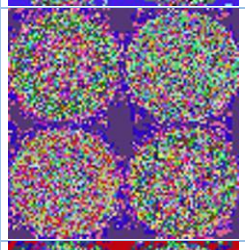
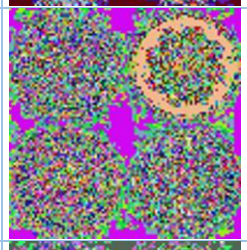
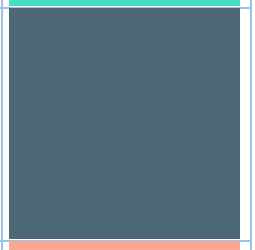
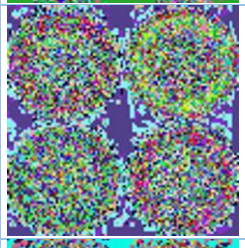
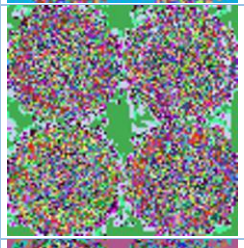
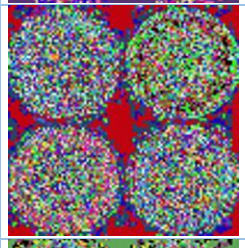
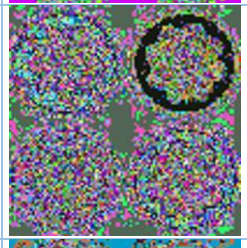
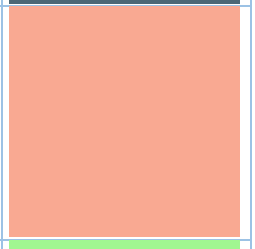
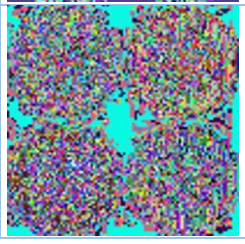
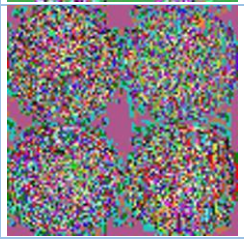
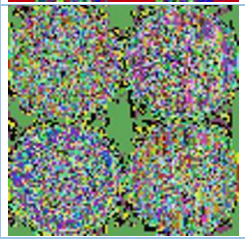
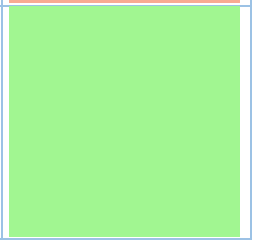
cosine					
correlation					
hamming					
Jaccard					
spearman					

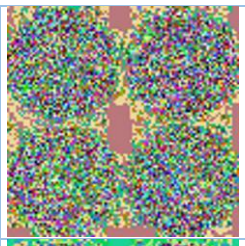
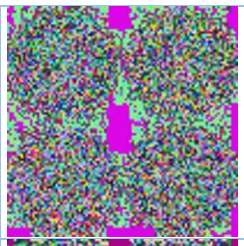
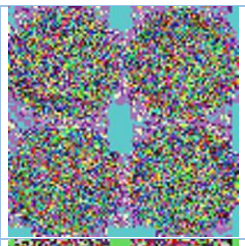
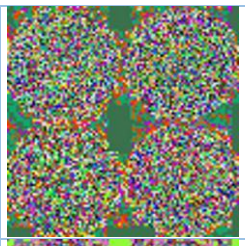
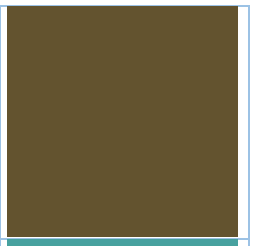
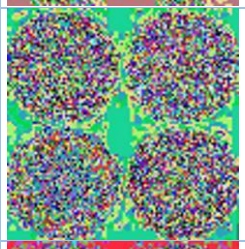
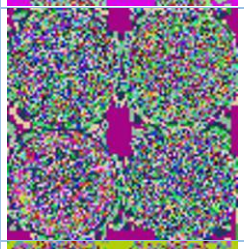
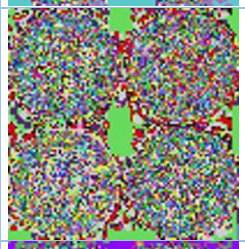
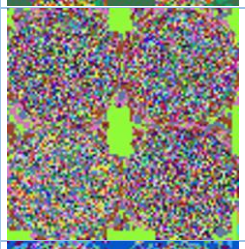
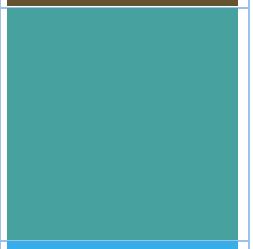
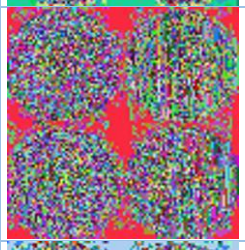
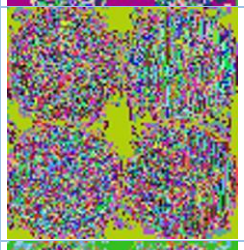
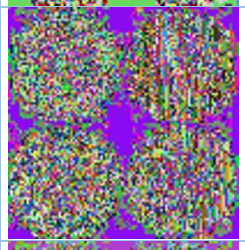
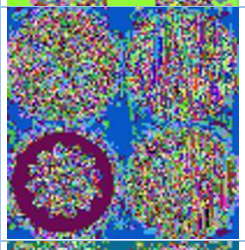
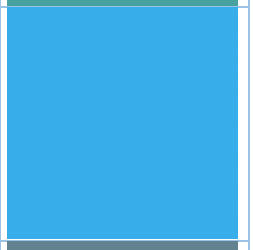
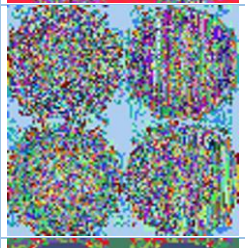
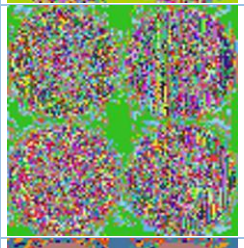
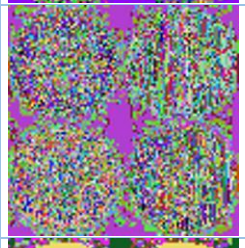
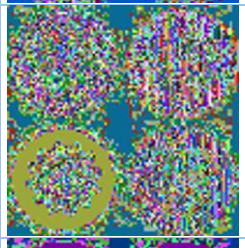
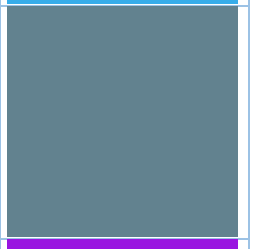
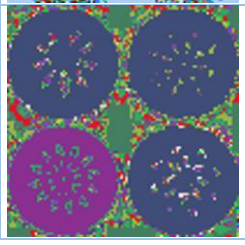
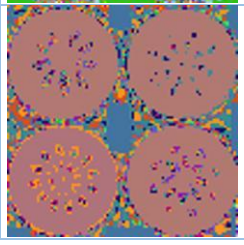
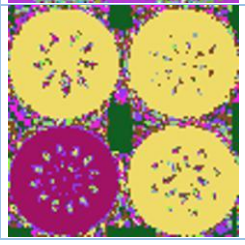
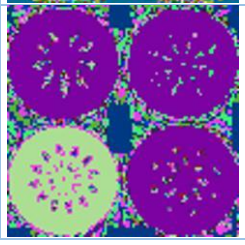
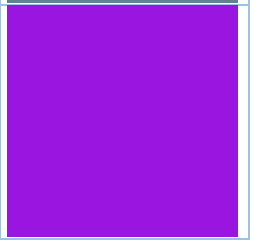
average	*cutoff = 0.1*	*cutoff = 0.25*	*cutoff = 0.5*	*cutoff = 1*	*cutoff = 5*
euclidean					
squaredeuclidean					
Seuclidean					
Mahalanobis					
Cityblock					
Minkowski					
chebychev					

cosine					
correlation					
hamming					
Jaccard					
spearman					

Single B-W	*cutoff = 0.1*	*cutoff = 0.25*	*cutoff = 0.5*	*cutoff = 1*	*cutoff = 5*
euclidean					
squaredeuclidean					
Seuclidean					
Mahalanobis					
Cityblock					
Minkowski					

chebychev					
cosine					
correlation					
hamming					
Jaccard					
spearman					

single	*cutoff = 0.1*	*cutoff = 0.25*	*cutoff = 0.5*	*cutoff = 1*	*cutoff = 5*
euclidean					
squaredeuclidean					
Seuclidean					
Mahalanobis					
Cityblock					
Minkowski					
chebychev					

cosine					
correlation					
hamming					
Jaccard					
spearman					

The result

What is remarkable is that as the number of cutoff increases, the number of clusters decreases significantly.
Also, as can be seen, this type of clustering is very sensitive to the edge. That is, where there are many changes, the algorithm finds many clusters because at these points the color difference between the pixels and, consequently, their distance changes with great intensity.

Thank you :)