Spectral Clustering

Giorgia delle Grazie s300879

29 agosto 2023

1 Introduction

Spectral clustering constitutes an advanced clustering method that draws upon concepts stemming from graph theory and linear algebra in order to group data based on their intrinsic relationships, even in scenarios where such relationships manifest in a non-linear or intricate manner.

The approach of spectral clustering is predicated upon the notion of representing data and their interactions through a matrix of similarity (or dissimilarity), subsequently harnessing the eigenvalues and eigenvectors associated with this matrix to extract valuable insights for the clustering process.

2 Dataset description

The datasets of N points used in this report are Cricle,that contains two columns corresponding to x-values and y-values of the points, and Spiral, that contains three columns, the third one contains the index of the correct cluster.

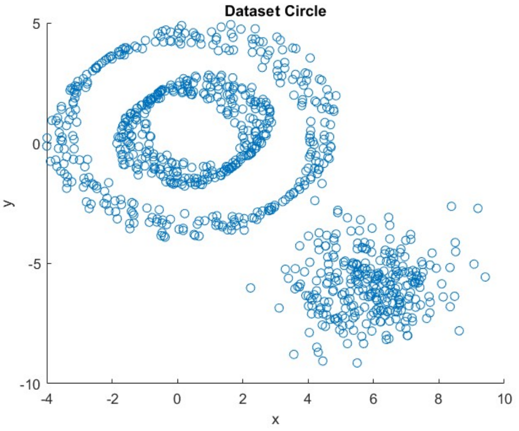

Figura 1: N points of Circle dataset

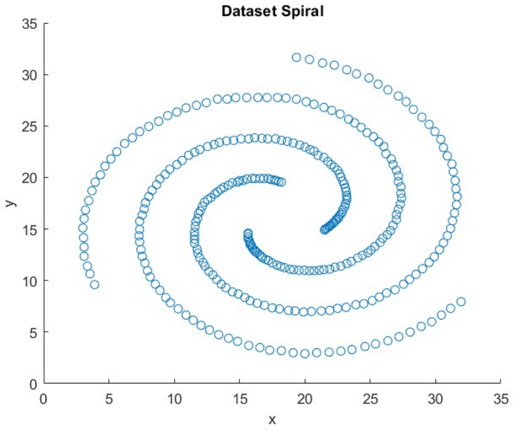

Figura 2: N points of Spiral dataset

3 K-nearest neighborhood similarity graph

Given a set of data points X point it is defined *similarity function* for two points:

$$s_{ij} = exp\left(-\frac{||\mathbf{X_i} - \mathbf{X_j}||^2}{2\sigma^2}\right) \tag{1}$$

The *k-nearest neighborhood similarity graph* $G = (V, E)$, where $V = v_1, ..., v_n$ denotes a non-empty set of vertices and E denotes the set of edges, a set of pair of vertices.
In the context of the similarity graph, each vertex $v_i \in V$ corresponds to a data point X_i. An edge connects vertices v_i and v_j if the similarity s_{ij} between the corresponding data points X_i and X_j is

positive or exceeds a specific threshold, indicating a substantial similarity required for a connection. Importantly, this threshold assures a meaningful linkage between data points.
It is assumed that $s_{ij} = s_{ji}$, and the edge linking v_i and v_j carries a weight proportional to s_{ij}. This arrangement leads to an observation that the resulting similarity graph is undirected.

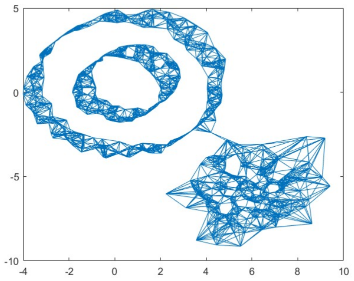

Figura 3: K-nearest neighborhood similarity graph k = 10 for Circle

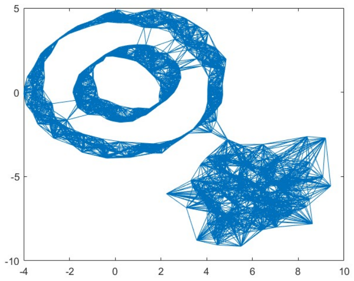

Figura 4: K-nearest neighborhood similarity graph k = 20 for Circle

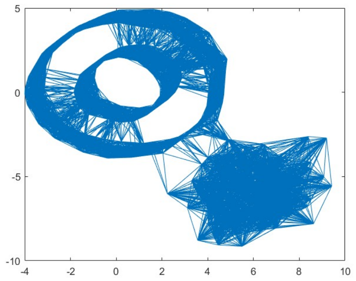

Figura 5: K-nearest neighborhood similarity graph k = 40 for Circle

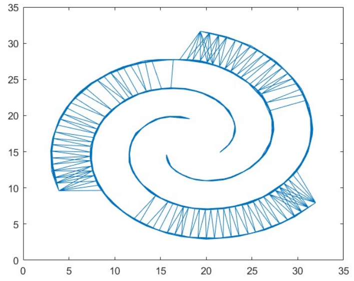

Figura 6: K-nearest neighborhood similarity graph k = 10 for Spiral

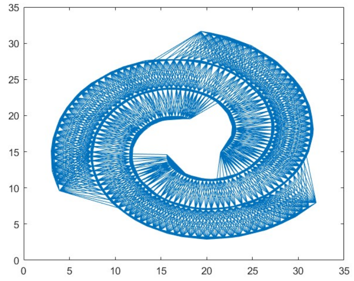

Figura 7: K-nearest neighborhood similarity graph k = 20 for Spiral

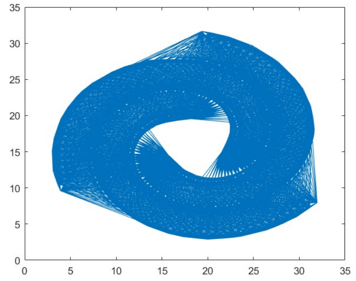

Figura 8: K-nearest neighborhood similarity graph k = 40 for Spiral

The weighted adjacency matrix, denoted as W_{ij}, takes on the value of s_{ij} when $i \neq j$, while it is set to $W_{ij} = 0$ when $i = j$. This matrix has been constructed using the *similarity function* (1), and its patterns for both datasets are presented below, with the utilization of diverse values of the parameter $k = 10,\ 20,\ 40$, while maintaining a constant value of $\sigma = 1$.

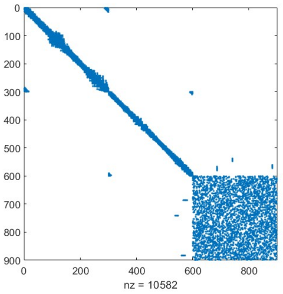

Figura 9: Pattern of W for Circle with k = 10

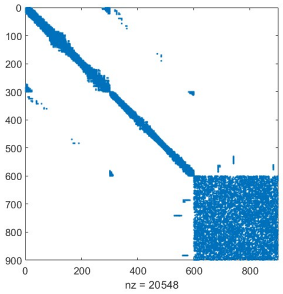

Figura 10: Pattern of W for Circle with k = 20

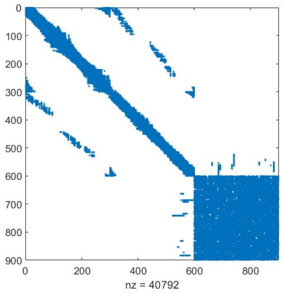

Figura 11: Pattern of W for Circle with k = 40

As shown in the previously figure, until element 600, the arrangement of neighboring points follows a diagonal pattern. This configuration could suggest that the first 600 points are tightly connected, giving rise to particularly strong proximity relationships. Indeed, these points constitute a dataset with a circular spatial arrangement.
However, after element 600, the pattern of neighboring points appears to change, and the points start to include neighbors from the entire area, no longer following a diagonal arrangement but a square one. This indicates a shift in the distribution of the data or in their neighboring relationships.
Increasing k the counts of neighbors along the diagonal experience an increment.

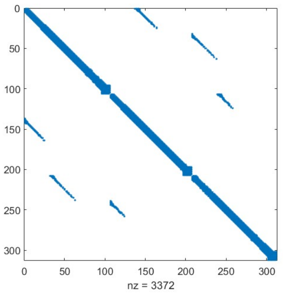

Figura 12: Pattern of W for Spiral with k = 10

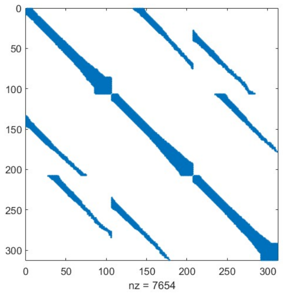

Figura 13: Pattern of W for Spiral with k = 20

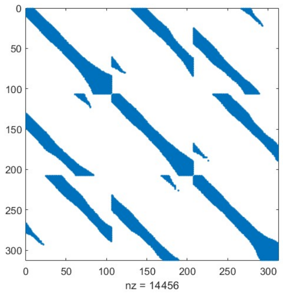

Figura 14: Pattern of W for Spiral with k = 40

Consider the Spiral dataset, where data points are organized into three spiral arrangements. When calculating k-nearest neighbors, each point's neighborhood is extended to include points from the other two spiral structures. As a result, this creates a diagonal pattern within the data.
As k increases, the number of neighbors belonging to the other spirals grows for each individual point. This leads to a denser sparsity pattern in the adjacency matrix.

4 Construct the degree matrix and the Laplacian matrix

The *Laplacian matrix* is define as:

$$L = D - W \tag{2}$$

where **D** represents the degree matrix, it is a diagonal matrix where the diagonal elements correspond to the degrees, which are defined as:

$$d_i = \sum_{j=1}^{N} w_{ij} \tag{3}$$

and W is the adjacency matrix.

5 Number of connected components and suitable number of clusters

The count of connected components within the graph corresponds to the multiplicity k of eigenvalue 0 within the Laplacian matrix. Through the observation of the least eigenvalues of matrix L, it becomes possible identify an appropriate quantity of clusters.
For the Circle dataset:

- k=10, the number of connected components is 2
- k=20, the number of connected components is 1
- k=40, the number of connected components is 1

For the Spiral dataset:

- k=10, the number of connected components is 1
- k=20, the number of connected components is 1
- k=40, the number of connected components is 1

As the value of k is elevated in both scenarios, a harmonious connected component takes shape within the Circle dataset. This occurrence stems from the increased interactions among neighboring points, harmonizing not only within the two upper-left circles but also between this structure and the scatter of points in the lower-right region.

For the purpose of determining an optimal cluster count, attention is directed towards the following graph:

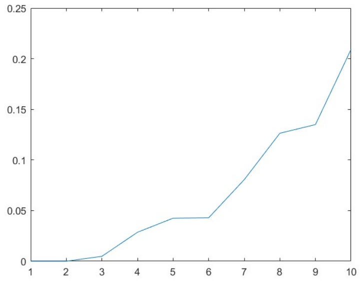

Figura 15: Eigenvalues of the Laplacian
for Circle, k = 10
number of cluster = 3

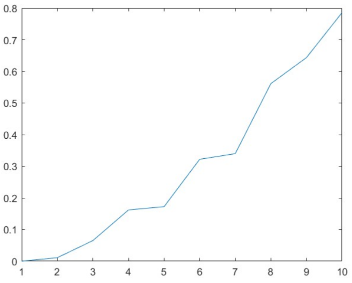

Figura 16: Eigenvalues of the Laplacian
for Circle, k = 20
number of cluster = 2

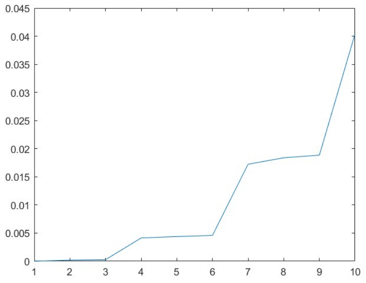

Figura 17: Eigenvalues of the Laplacian
for Circle, k = 40
number of cluster = 2

In the dataset Circle, under the condition where k=10, the minimum eigenvalue is 0, as anticipated. The second smallest eigenvalue is not precisely zero, yet it remains below 10^{-6}. The third eigenvalue continues to exhibit relatively diminutive magnitude, while beginning from the fourth eigenvalue, the magnitude becomes notably substantial. This indicates that an appropriate cluster count is 3. The same procedure for selecting the optimal number of clusters will also be applied for the subsequent cases: for $k = 20$, the first two smallest eigenvalue are not exactly zero, but rather sufficiently small; meanwhile, starting from the third eigenvalue,its value begins to increase.
A more or less similar behavior is observed with $k = 40$.
The number of clusters is selected using the same procedure also for the Spiral dataset.

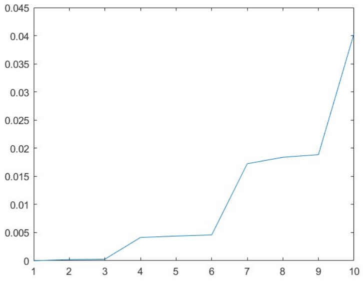

Figura 18: Eigenvalues of
the Laplacian
for Spiral, k = 10
number of cluster = 3

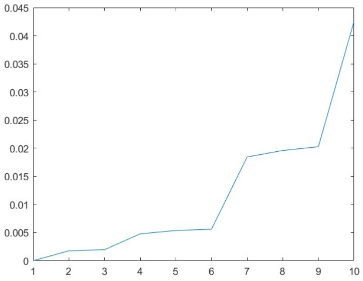

Figura 19: Eigenvalues of
the Laplacian
for Spiral, k = 20
number of cluster = 3

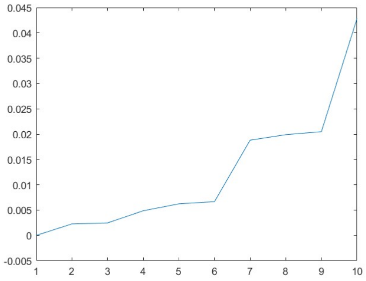

Figura 20: Eigenvalues of
the Laplacian
for Spiral, k = 40
number of cluster = 6

6 Define the matrix U and K-Means clustering

After computing the M eigenvectors $u_1, ..., u_M \in \mathbf{R}^N$ that correspond to the M smallest eigenvalues of the Laplacian matrix, the matrix $U \in \mathbf{R}^{NxM}$ is define using these vectors as columns. This process is mapping each initial point in $\mathbf{R}^N$ to a point in $\mathbf{R}^M$; in this way, the new positions of the mapped points are essentially easier to group together, in which we can apply k-means to the points y_i using the number of clusters M found previously.

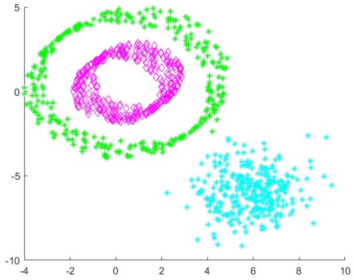

Figura 21: Spectral clustering
for Circle data, k = 10

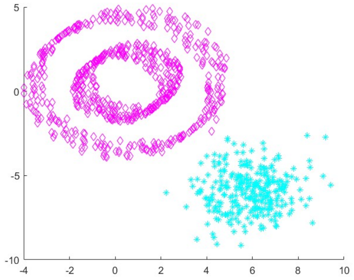

Figura 22: Spectral clustering
for Circle data k = 20

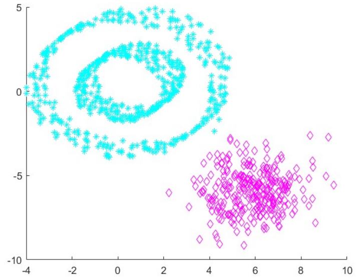

Figura 23: Spectral clustering
for Circle data k = 40

In this case, it is readily apparent that when $k = 10$, the partitioning of data into clusters exhibits a distinct clarity: the two circles belong to two separate clusters, while the remaining data points, situated in the bottom-right region, constitute the third cluster.
As the value of k increases, the two circles become part of the same cluster, while the bottom region continues to belong to a distinct cluster.

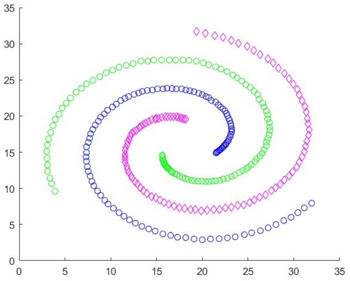

Figura 24: Spectral clustering for Spiral data, k = 10

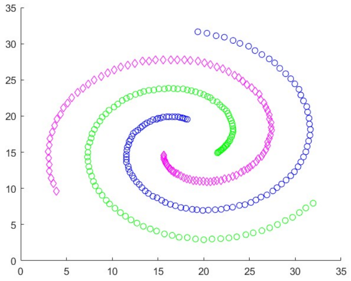

Figura 25: Spectral clustering for Spiral data k = 20

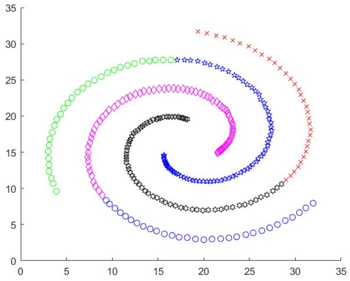

Figura 26: Spectral clustering for Spiral data k = 40

Regarding the Spiral dataset, for $k = 10$ and $k = 20$, the clusters uphold the division in accordance with the three spiral structures. However, with an increment in the value of $k = 40$, divisions emerge within the spirals themselves due to heightened interconnections among the distinct spirals.

7 Compare with other cluster methods

In this section, the obtained results will be compared with other cluster methods, particularly K-means and DBSCAN.

K-Means is a clustering method in which given a dataset X consisting of n points $x_1, x_2, ..., x_n$ in a d-dimensional space, its goal is to partition these points into k clusters, where k is a predetermined number of clusters. Each cluster is represented by a centroid, which is the mean point of the data within the cluster.
K-Means works better with clusters that have shapes similar to spheres. Clusters with irregular or elongated shapes might not fit well with K-Means.

DBSCAN is a clustering method in which the algorithm creates a circle of radius ϵ for each point of the dataset and classifies the data into core points, border points, and noise points.
A point is classified as a core point if it has more than a specified minimum number of points within the circle's radius, as a border point if it has fewer than the specified minimum number of points but is in the neighborhood of a core point, and as a noise point otherwise.
Unlike K-Means, this approach allows for the identification of potential outliers, and it performs well in situations with clusters of different sizes, densities, and irregular shapes.

The following figure illustrates the outcomes of various clustering methods applied on the two different datasets.

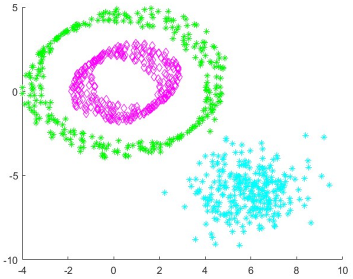

Figura 27: Spectral clustering for Circle data, k = 10

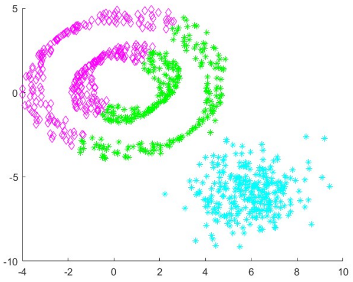

Figura 28: Kmeans clustering for Circle data k = 10

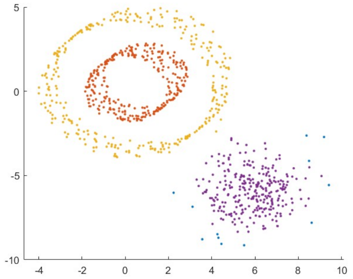

Figura 29: Dbscan clustering for Spiral data k = 10

As mentioned earlier, K-Means encounters challenges when dealing with clusters of non-spherical shapes, which results in the incorrect partitioning of points within overlapping circles, as shown in the second figure.
Using DBSCAN, we obtain a result quite similar to that achieved through spectral clustering. Additionally, please note the the presence of an additional cluster that includes points classified as noise.

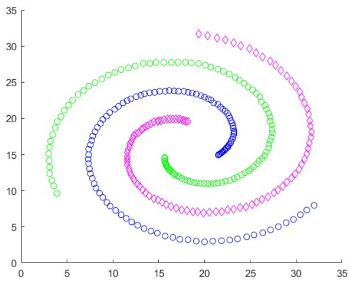

Figura 30: Spectral clustering for Circle data, k = 10

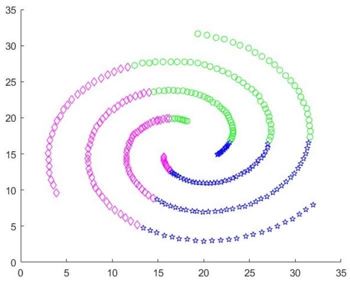

Figura 31: Kmeans clustering for Circle data k = 10

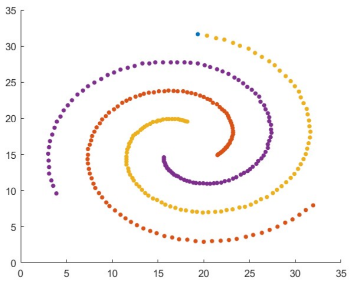

Figura 32: Dbscan clustering for Spiral data k = 10

Similarly to the case of the Circle dataset, in the Spiral dataset as well, the K-Means method proves to be ineffective due to the intricate data structure. This inefficiency is attributed to the K-Means algorithm's tendency to form spherical partitions, which do not align well with the non-globular nature of the data.
Conversely, regarding the DBSCAN approach, we obtain results similar to those achieved through spectral clustering.
Furthermore, there is a persistence of points identified as noise, in reduced quantities compared to the Circle dataset.

8 Spectral clustering on 3D dataset

Creating a 3D dataset within this section, the spectral clustering algorithm will be tested on the following data:

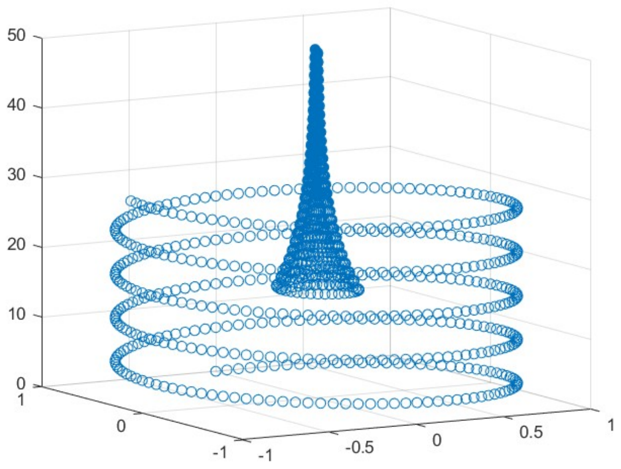

Figura 33: 3D dataset

With $k = 10$ in this scenario, two connected components emerge, representing both the spiral shape and the inverted cone. As the value of k is increased for both cases $k = 20, 1, k = 40$, a single connected component is obtained. This is due to the increase in interactions among neighboring points, as highlighted in the three graphs below.

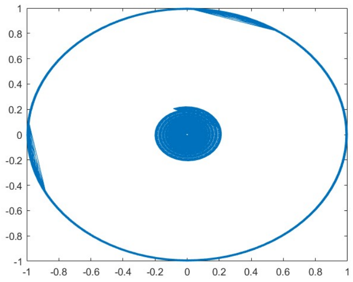

Figura 34: K-nearest neighborhood similarity graph k = 10

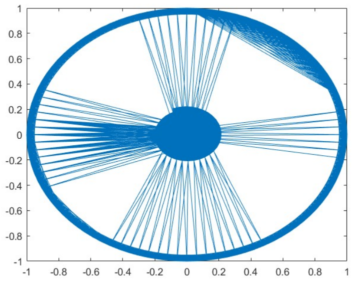

Figura 35: K-nearest neighborhood similarity graph k = 20

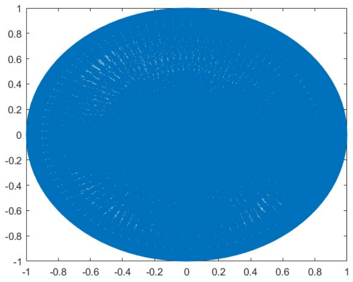

Figura 36: K-nearest neighborhood similarity graph k = 40

The same definition of adjacency is used in this case but with a different formulation of Laplacian matrix:

$$L_{sym} := D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I -^{-\frac{1}{2}} W D^{-\frac{1}{2}} \tag{4}$$

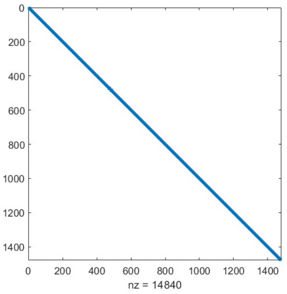

Figura 37: Pattern of W, k=10

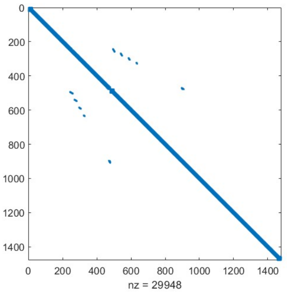

Figura 38: Pattern of W, k=20

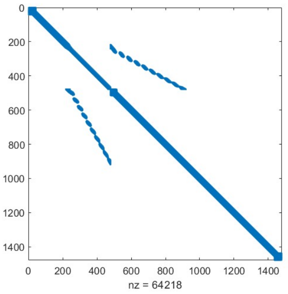

Figura 39: Pattern of W, k=40

The image consists of a structure taking the form of a spiral and an inverted cone. It is worth mentioning that the sparse matrix demonstrates a pattern characterized by two consecutive diagonals—up to element 476, the first diagonal is followed by the second. This pattern indicates the presence of strong proximity relationships. Moreover, as the parameter k increases, there is an increase in the count of elements that differ from zero within the matrix, that do not belong to the diagonal.
Subsequently, for select an appropriate number of clusters, a comparison is made between the magnitudes of the smallest eigenvalues.
An analysis is conducted on the various plots obtained.

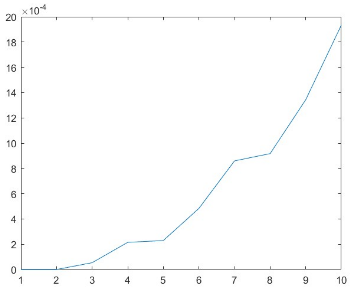

Figura 40: Eigenvalues of the Laplacian
k = 10
numbero of cluster = 2

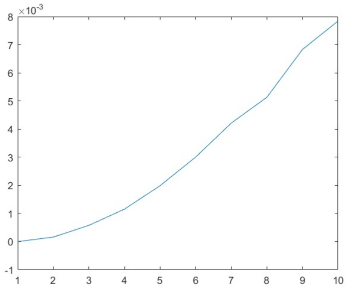

Figura 41: Eigenvalues of the Laplacian
k = 20
numbero of cluster = 3

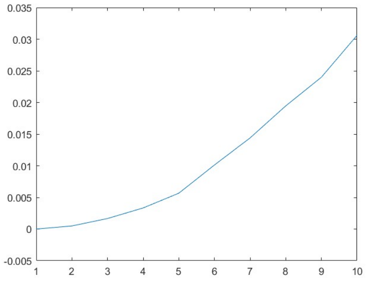

Figura 42: Eigenvalues of the Laplacian
k = 40
numbero of cluster = 4

Upon selecting an appropriate number of clusters, the k-means algorithm is executed, and the ensuing graphical results are showcased below. Furthermore, there is an observed correlation between the increment of "k" and the augmentation of cluster count, attributed to the expanding interconnections between the spiral and the inverted cone. With $k = 10$, two distinct clusters are formed, corresponding to the two structures in the figure: the spiral belongs to one cluster, while the inverted cone belongs to another.

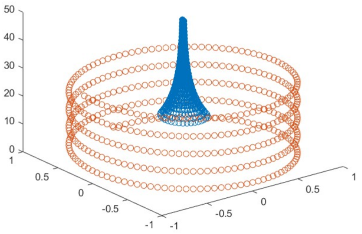

Figura 43: Spectral clustering
k = 10

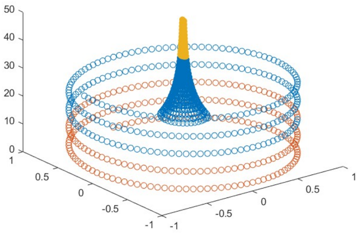

Figura 44: Spectral clustering
k = 20

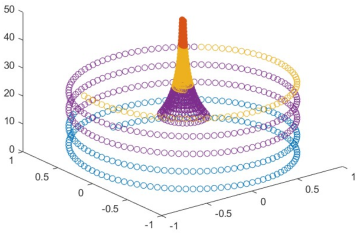

Figura 45: Spectral clustering
k = 40

For $k = 20$, three different clusters emerge: the first encompasses only the lower part the spiral, the second encompasses the final part of the spiral and the central part of the cone, and the third cluster encompasses the tip of the cone.

Finally, with $k = 40$, four clusters are obtained: the first cluster represents the lower part of the spiral, the second represents the central part of the spiral and the final part of the cone, the third cluster encompasses the last part of the spiral and the central part of the cone, and lastly, the fourth cluster represents the tip of the cone.