Clustering Algorithm - Predicting Household Electricity Consumption in the Face of El Niño and La Niña Events: A Fuzzy Logic-Based Model

Myllee Sarleth Mosquera Rivas
Department of Applied Sciences and Engineering
EAFIT University
Medellín, Colombia
msmosquerr@eafit.edu.co

***Resumen*—The following article explores five clustering algorithms: Mountain, Subtractive, Kmean, Fuzzy – Kmeans, and Density-based spatial clustering of applications with noise (DBSCAN) , with the purpose of grouping data and extracting features in both high and low dimensions. Additionally, it provides a concise analysis to determine the optimal parameters for each clustering algorithm, using measures of intra and extra cluster validation. This analysis aims to improve the effectiveness and accuracy of the results obtained by the algorithms, which can be crucial in various applications.**

-A. Mesh

Intra - Cluster Validation:

For intra-cluster validation, the Dunn index was implemented in Python, and the functions from the Sklearn library were used to calculate the Davies-Bouldin and Silhouette indices. Below is a brief description of these indices:

- Dunn Index: The Dunn index measures the compactness of clusters and the separation between them. It is defined as the ratio of the minimum inter-cluster distance to the maximum intra-cluster distance. A higher Dunn index indicates better clustering, with smaller intra-cluster distances and larger inter-cluster distances.
- Davies-Bouldin Index: The Davies-Bouldin index evaluates the clustering quality based on the average similarity between each cluster and its most similar cluster, relative to the average dissimilarity between clusters. A lower Davies-Bouldin index indicates better clustering, with clusters that are well-separated and compact.
- Silhouette score: Measures how similar an object is to its own cluster compared to other clusters. It ranges from -1 to 1, where a score close to 1 indicates that the object is well-matched to its own cluster and poorly matched to neighboring clusters, and a score close to -1 indicates that the object is poorly matched to its own cluster and well-matched to neighboring clusters. A higher Silhouette score indicates better clustering.

Extra - Cluster Validation

For Extra - Cluster validation the Rand index is was used, This index is a commonly used measure to assess the similarity between two partitions or clusterings of a dataset. It is used in cluster validation to compare the similarity between the clustering obtained by a clustering algorithm and a reference or considered true clustering.

The Rand Index measures the fraction of pairs of samples that agree in both their classification in the "trueçlustering and the clustering obtained by the clustering algorithm. It can take values between 0 and 1, where 0 indicates that the clusterings do not match and 1 indicates a perfect match.

Normalization

$$x_{norm} = \frac{x - min(x)}{max(x) - min(x)} \tag{1}$$

Mountain

The Mountain method is a clustering algorithm that relies on identifying and grouping dense regions in a multidimensional data space. To achieve this, it employs an iterative search strategy to find potential clustering centroids, which are centroids in the space surrounding the data set where the closest points are concentrated. After identifying and slightly refining these centroids, they become the clusters that form the final result. This method is based on the idea of finding "peaks.ºr mountains in the density of the space and then assigning each point close to these peaks to the corresponding cluster. The Mountain algorithm is useful for data of any length and dimensionality.

In (1) you can find the pseudocode that served as the basis for the implementation of this algorithm. As mentioned previously, the main premise for cluster selection lies in the densities with respect to the data points, followed by the analysis of densities in relation to the centroids being discovered during the process. The functions in question are:

$$m(v) = \sum_{i=1}^{N} exp\left(-\frac{||v - x_i||^2}{2\sigma^2}\right) \tag{2}$$

and

$$m_{new} = m(v) - m(c_i) * exp\left(-\frac{||v - x_i||^2}{2\sigma^2}\right) \quad (3)$$

Where v is a ponit in the mesh, x_i is a point in the data, c_i is a center selected, α and β are parameters that determine the height and smoothness of the resulting mountain function. Using the Iris dataset, analyses were conducted to determine the optimal values for the parameters α and β for the Mountain algorithm. This involved seeking to establish a relationship between these parameters, and below, the analyzed relationships and the results obtained for each are presented.

In the following image, the relationship between the parameters is compared, where $\beta = \alpha$, by evaluating the external validation indices Davies-Bouldin, Dunn, and Silhouette. It is observed that the variation in the Dunn index is minimal and that, as the parameter values increase, the change is not significant. As for the Silhouette index, greater variability is achieved compared to Dunn. As mentioned earlier, it is expected that both Dunn and Silhouette maximize their value to consider these parameters as optimal, while Davies-Bouldin should be minimized.

Selecting the appropriate parameters is challenging due to the low variability in the Dunn and Silhouette indices. However, prioritizing the results of the Davies-Bouldin index, it is observed that its minimum values are found at two key points: $\alpha = \beta = 0.4$ and $\alpha = \beta = 0.7$. Although the other indices may indicate otherwise, the low variability does not allow for a deeper analysis. These parameters suggest a total of 2 clusters and 5 clusters for the Iris dataset, respectively.

Although, based on prior knowledge of this database, it would commonly be expected to obtain 3 clusters, the algorithm would yield 3 clusters when $\alpha = \beta = 0.5$. However, it is at this point where Davies obtains its maximum value, suggesting that a classification of 3 clusters is not suitable for our algorithm. Therefore, we will consider that $\alpha = \beta = 0.4$ is the best option, as it is closer to the expected result.

When evaluating the Rand Index, as mentioned earlier, a value close to 1 indicates a good clustering of data or a high agreement with the actual clustering. In the case of alpha = beta, we observe that the maximum values are obtained at $\alpha = \beta = 0.7$ and 0.9, resulting in a total of 5 and 9 clusters respectively. This does not contrast with our previous conclusion of selecting $\alpha = \beta = 0.4$, where the index reaches its minimum value.

Considering our prior knowledge of the database, a total of 9 clusters deviates considerably from the theoretical classification. On the other hand, 5 clusters seem to be a more reasonable option, although still slightly deviating from reality. However, it is important to note that the index value at this point is above 0.75, indicating an excellent agreement between the clusterings.Therefore, taking into account primarily the results of the Davies and Rand indices, we can conclude that the best option for this case is $\alpha = \beta = 0.7$.

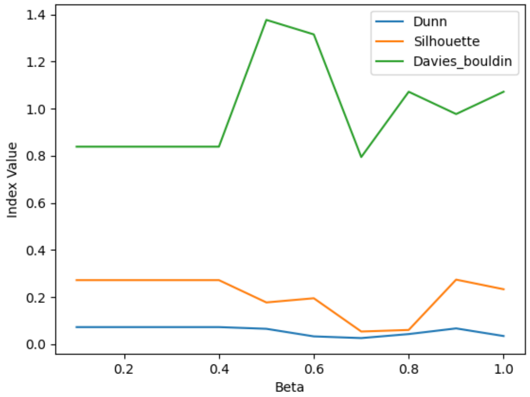

Figura 1: Intra cluster Validation with $\alpha = \beta$

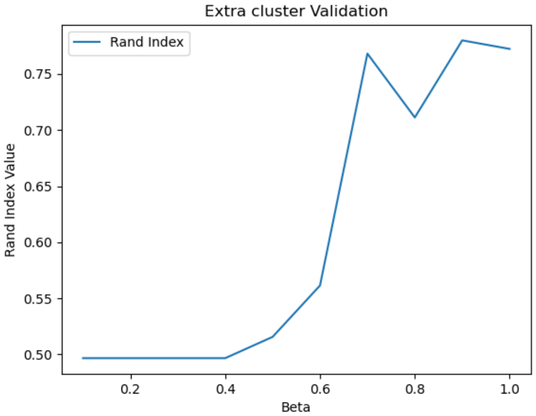

Figura 2: Extracluster Validation with $\alpha = \beta$

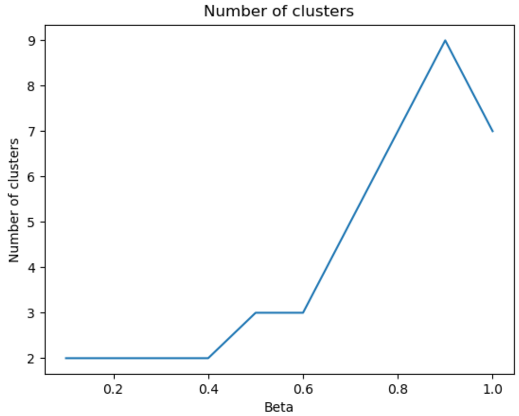

Figura 3: Number of clusters with $\alpha = \beta$

Now, considering the relationship $\beta = 1.5\alpha$, we can see that Davies-Bouldin obtains its minimum value, and Silhouette its maximum value at $\alpha = 1$, indicating that $\alpha = 1$, and $\beta = 1.5$, generate optimal clusters for the grouping of this data. According to the graph, this grouping is represented by 2 clusters.

Upon analyzing the Rand Index, it can be observed that the best data clustering is achieved when alpha falls between 0.5 and 0.7, with values higher than 0.7 also yielding satisfactory clustering, albeit at a slightly lower level. As previously mentioned with the intra-cluster validation indices, the optimal solution is attained at alpha = 1 and beta = 1.5. However, at this juncture, the Rand Index value hovers around 0.70. According to this index, the best results are obtained at alpha = 0.6 and 0.7.

Nevertheless, at alpha = 0.7, the Davies index reaches its maximum value instead of its minimum, whereas at alpha = 0.6, the Davies index displays one of its lowest values. Additionally, the number of clusters associated with this latter point deviates significantly from reality, prompting its dismissal.

Considering that theoretically, 3 clusters are expected for the Iris dataset, it is noted that at alpha = 0.8, 3 clusters are achieved. At this juncture, the Rand Index value is approximately 0.72. Although the Davies index does not attain its minimum value at this point, it exhibits a relatively low value compared to some other options. On the other hand, the Silhouette index reaches one of its highest values. Thus, although this option may not be the best according to the indices, it is the one that comes closest to reality and yields good results when computing the indices.

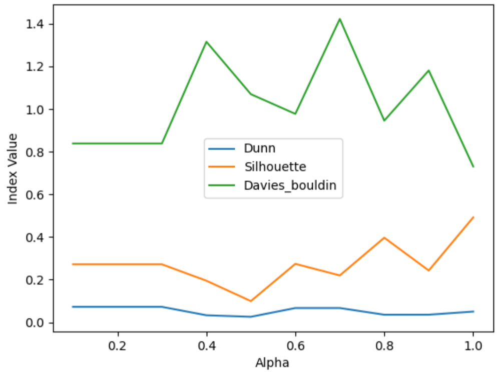

Figura 4: Intra cluster Validation with $\beta = 1.5\alpha$

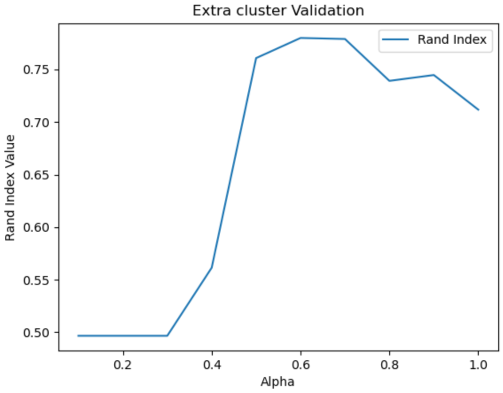

Figura 5: Extra cluster Validation with $\beta = 1.5\alpha$

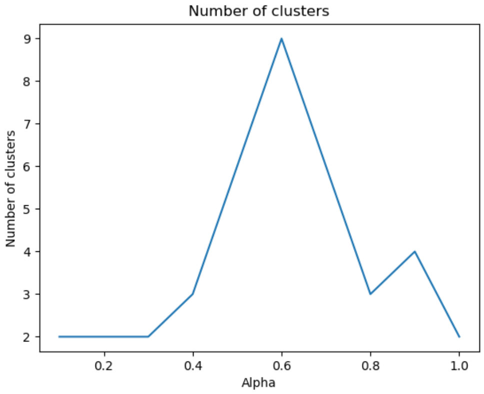

Figura 6: Number of cluster Validation with $\beta = 1.5\alpha$

Other possibilities were explored; however, the results lead to very similar conclusions to the previous ones, so they will not be mentioned here.

Subtractive

Similar to the Mountain algorithm, the subtractive algorithm is a clustering method based on the gradual elimination of less dense points in a dataset to identify the initial centroids of the clusters, which correspond to those with higher densities. The Subtractive algorithm offers an efficient way to identify initial centroids and can be adapted to different types of data and distributions.

Similar to the Mountain method, this method also features 2 density functions, for which it is of interest to find the optimal parameters for cluster construction. These density functions in question are:

$$D_i = \sum_{i=1}^{N} exp\left(-\frac{||x_i - x_j||^2}{(r_a/2)^2}\right) \tag{4}$$

and

$$D_i = D_i - D_{c1} * exp\left(-\frac{||v - x_i||^2}{2\sigma^2}\right) \tag{5}$$

When implementing the same strategy as before to find the parameters of the Subtractive algorithm, we observe that, regardless of the parameters used, the algorithm tends to converge towards a single solution. Therefore, in this case, the conclusions obtained are consistent regardless of the parameters employed. To validate this observation, additional tests were conducted with different simulated datasets, leading to the same conclusions. It can be inferred that this method rapidly converges towards a single solution. However, it is worth noting that this single solution may not be the most optimal.

In this case our database has 8 inputs and 1 outputs, so we are in the space R^n, with $n = 9$, and $l = 3$, so the mesh will have a total of $N = 262144$ vertices, and m = 2.

DBSCAN

The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm is a density-based clustering method that groups data points based on local density. It works by defining two parameters: ϵ, which specifies the maximum distance between two points for one to be considered a neighbor of the other, and minPts, which specifies the minimum number of points within epsilon to form a cluster.Here is a brief pseudocode of the implemented algorithm:

Algorithm 1 DBSCAN Algorithm

```
Input: Dataset $D$, $\varepsilon$, $minPts$
Output: Clusters $C$
$visited \leftarrow \emptyset \quad C \leftarrow \emptyset$
for $each\ point\ P\ in\ D$ do
    if $P\ is\ not\ visited$ then
        mark $P$ as visited  $neighbors \leftarrow$ regionQuery($P, \varepsilon$)
        if $|neighbors| \geq minPts$ then
            $C_{new} \leftarrow$ expandCluster($P, neighbors, \varepsilon, minPts$)
            add $C_{new}$ to $C$
        end
    end
end
return $C$
```

Learning 1: High Dimensions

Fuzzy K-means

To perform learning in high dimensions, the UMAP algorithm was used to expand the database dimension to n = 15, And a total of 5000 data points. Under this dimension, the Fuzzy K-means model, with $m = 1$, yields 3 clusters as a result. However, it is observed that the majority of the points are covered by only 2 clusters, while one cluster contains very few data points. This is clearly due to initially selecting 3 clusters randomly to be placed at midpoints, resulting in only 2 distinctive clusters.

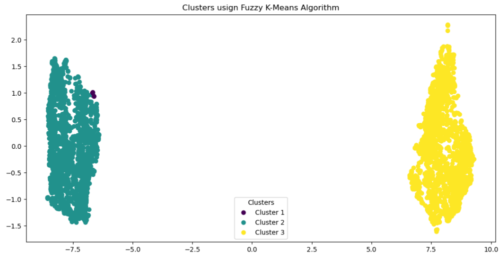

Figura 7: Cluster with Fuzzy Kmeans, m=1, and Random clusters = 3

Performing the Intra-cluster validation, with a Silhouette Score value of 0.51074, indicates that the clusters are quite well separated, and the points within each cluster are relatively close to each other compared to points in other clusters. The Davies Bouldin Score index, with a value of 1.38312, suggests that the separation of the clusters is good; however, some clusters could be better separated (such as the case of the cluster mentioned with very few points). Finally, the Calinski index with a value of 63086.627 indicates a good concentration of points in each of the clusters.

K means

When considering an initial number of clusters of 2, the algorithm yields a very good separation of the clusters and a good grouping of the data within each of them. However, when we assign 3 random clusters, we indeed obtain the 3 clusters, but the separation of the clusters is not very good, so this would not be the best grouping.

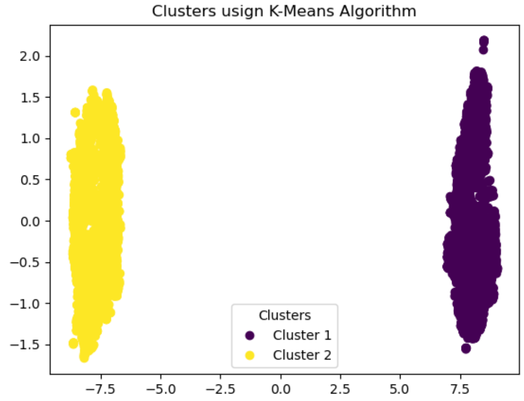

Figura 8: Cluster with Kmeans, and Random clusters = 2

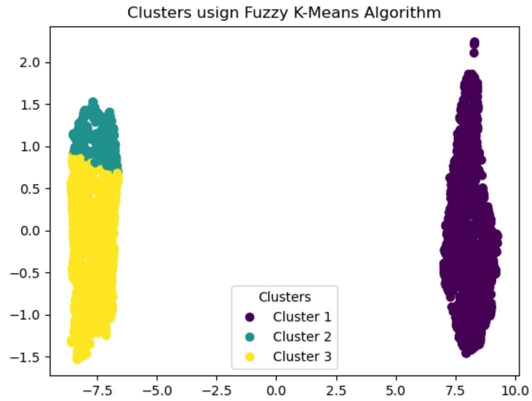

Figura 9: Cluster with Kmeans, and Random clusters = 3

Mountain

Realizing the implementation in high dimensions with the Mountain algorithm, using $\alpha = \beta = 0.7$ (this according to the results with toy data), the algorithm yields a total of 10 clusters. However, upon searching for the points belonging to each cluster, it is found that these points are grouped into only 5 clusters, indicating that some clusters are not very significant. Regarding the intra-cluster validation indices, we have a Silhouette score of 0.46, which suggests that the clustering is acceptable but not ideal. On the other hand, the Davies index with a value of 1.525 indicates that this clustering does not present a good separation between the clusters. Lastly, the Dunn index with a value of 0.01664 indicates a very poor separation between clusters, which is evident in the graph as many clusters can be observed clustered together on the left side.

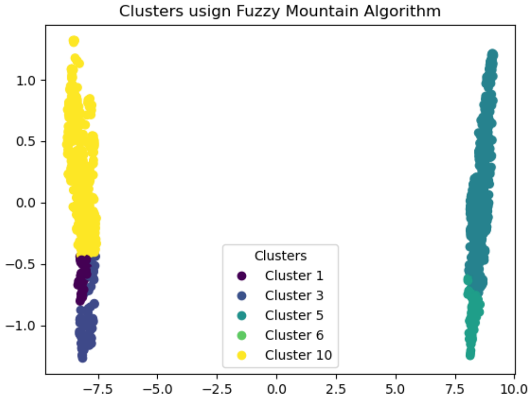

Figura 10: Cluster with Mountain,$\alpha = \beta = 0.7$

Now, taking into account the analysis with the toy data, clusters were created using the Mountain method with $\alpha = 0.8$ and $\beta = 1.5\alpha$. Only one cluster is obtained, whereas a minimum of 2 clusters would be expected. Therefore, it is evident that this is a poor grouping of the data.

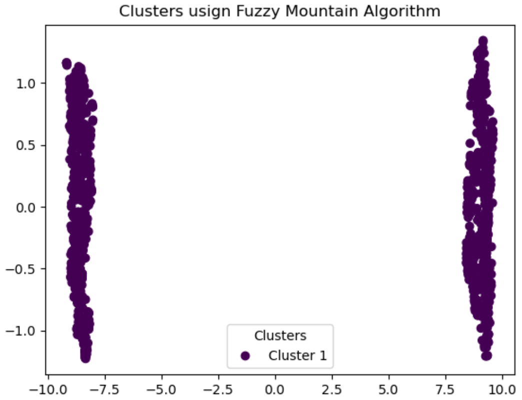

Figura 11: Cluster with Mountain,$\alpha = \beta = 0.8$

Subtractive

With a value of $r_a = 0.3$ and $r_b = 1.5r_a$, a total of 3 clusters is obtained. A Silhouette value of 0.123 indicates that the clustering is not good, as there are points that could be better assigned to other clusters. The Davies index with a value of 1.523 indicates that although the separation of the clusters is acceptable, it is not optimal. This could be mainly due to the clusters on the left side of the graph. On the other hand, the Dunn index indicates that the separation of the clusters is very poor. Thus, based on the results of the indices, we can conclude that this is not the best clustering of the data.

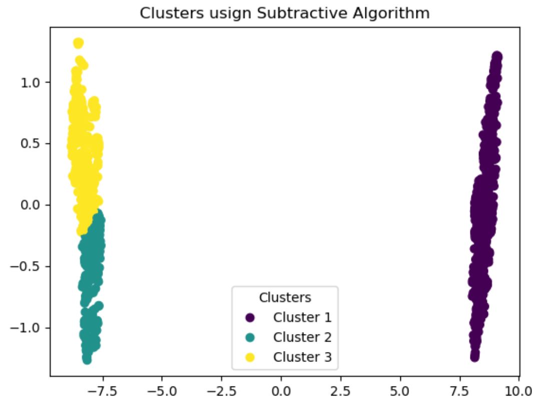

Figura 12: Cluster with Subtractive,$r_a = 0.3, r_b, 1.5r_b$

DBSCAN

With a maximum distance $\epsilon = 2.5$ and a minimum number of points equal to 100, 5 clusters are obtained. However, when grouping the data, a total of 4 clusters are reached. Regarding the indices, they indicate that this clustering is poor as well as the separation of the clusters, since a value of 0.05 is reached for Silhouette, and 0.84 for Davies.

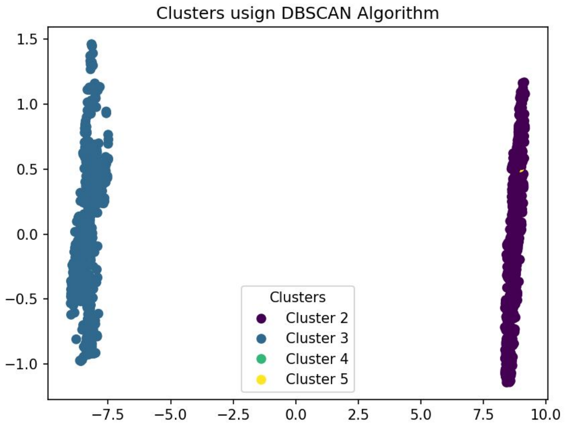

Figura 13: Cluster with DBSCAN, $\epsilon = 2.5$, Min Points = 100

Increasing the maximum distance to $\epsilon = 3$, and increasing the minimum number of points to 200, the expected number of clusters is reached, which is 2. According to the Silhouette index with a value of 0.90, the data clustering is very good, and there is a good separation between clusters. This conclusion is supported by the Davies indices with a value of 0.143, and a Calinski value of 47639.253.

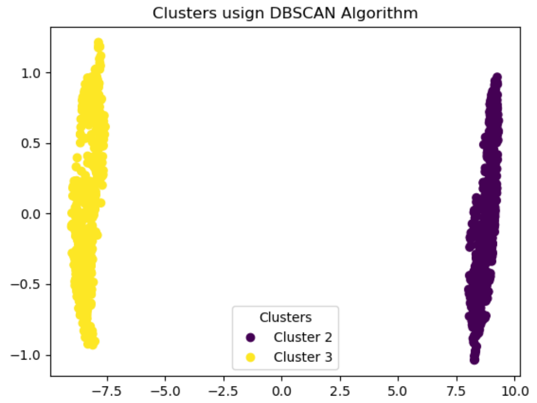

Figura 14: Cluster with DBSCAN, $\epsilon = 3$, Min Points = 200

Learning 2: Original Dimensions

Fuzzy Kmeans

Conserving the original dimension (n=9), using the Fuzzy Kmeans algorithm with a parameter $m = 1.5$, and a total of 2 initially randomly selected clusters, we obtain a total of 2 clusters as expected. It is too evident that this is the optimal clustering. This result is supported by the Silhouette score indices with a value of 0.901, indicating that the clustering is quite good as it is very close to its maximum value of 1. Likewise, with a value of 0.143, the Davies index indicates that there is a very good separation between the clusters. Lastly, the Calinski index with a value of 47639.253 indicates that there is very good intra-cluster cohesion, and the points within each cluster are closely grouped together.

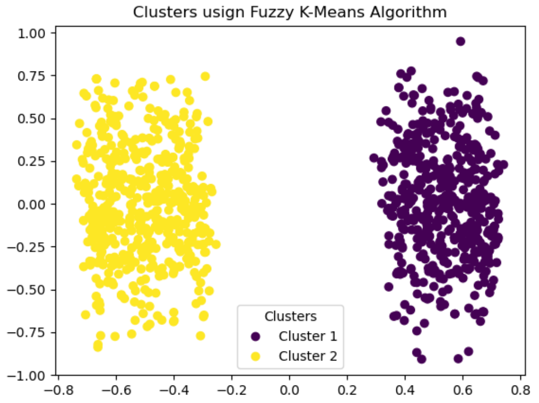

Figura 15: Cluster with Fuzzy Kmeans, $m = 1.5$, and 2 Random Clusters

K means

It can be observed that, in the case of the K-means algorithm, the exact same data grouping is achieved as with the Fuzzy K-means algorithm. It is evident that, when learning in high dimensions, it becomes much more complex to visualize and analyze the space compared to working in the original space. Although the original space may be high-dimensional, it somewhat facilitates the identification of possible groupings.

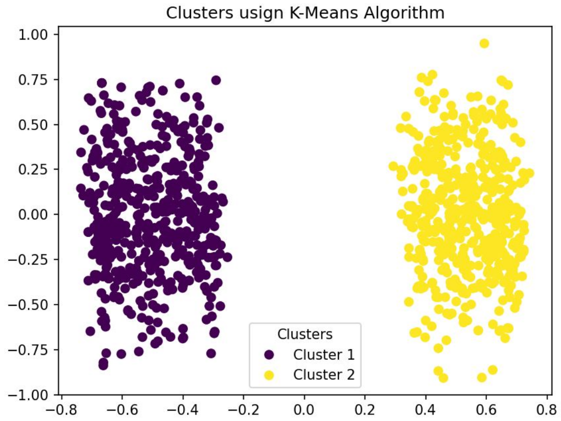

Figura 16: Cluster with Kmeans, (2 Random Clusters)

Subtractive

In the original space, with $r_a = 0.3$ and $r_b = 1.5r_a$, we end up with a total of 2 clusters, however, there is a poor separation between the clusters, particularly noticeable in the clustering on the left side. This conclusion is supported by the intracluster validation indices, as evidenced by a Silhouette index value of 0.117, a Davies index value of 2.08, and a Calinski value of 139.627, all indicating that this clustering is too poor due to the lack of good separation between the clusters.

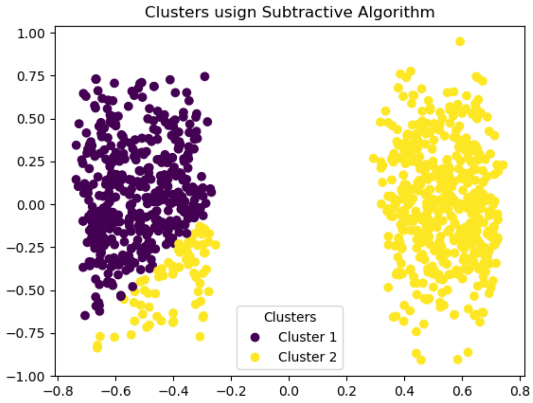

Figura 17: Cluster with Subtractive, $r_a = 0.3, r_b = 1.5r_a$

We can see that with the subtractive algorithm, we end up with 2 or more clusters, which according to the indices is negative. For instance, for $r_a = 0.8$, the Silhouette index obtains a low value of 0.142, Davies a value of 2.047, and Calinski presents a value of 157.423. All of them indicate that the data clustering is not the most suitable as the cohesion and separation between clusters are too poor.

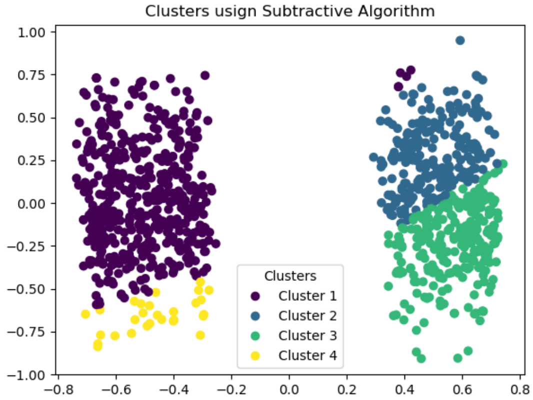

Figura 18: Cluster with Subtractive, $r_a = 0.8, r_b = 1.5r_a$

-B. DBSCAN

Learning with the DBSCAN algorithm, with $\epsilon = 3$ and Min Points = 200, results in a total of 1 cluster, and with $\epsilon = 1.5$ and Min Points = 100, a total of 4 clusters are obtained. In this scenario, a total of 3 significant clusters can be visualized; however, the intra-cluster validation indices indicate that this is a very poor clustering of data. The Silhouette and Calinski indices reach very low values of 0.051 and 53.688, respectively, and Davies reaches a value of 2.582, which greatly exceeds the ideal. Therefore, this clustering is not the most suitable.

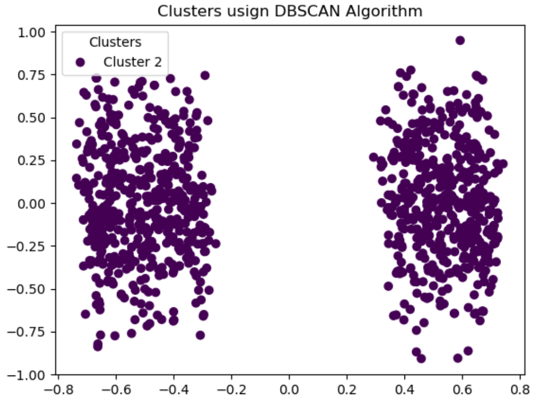

Figura 19: Cluster with DBSCAN, $\epsilon = 3$, Min Points = 200

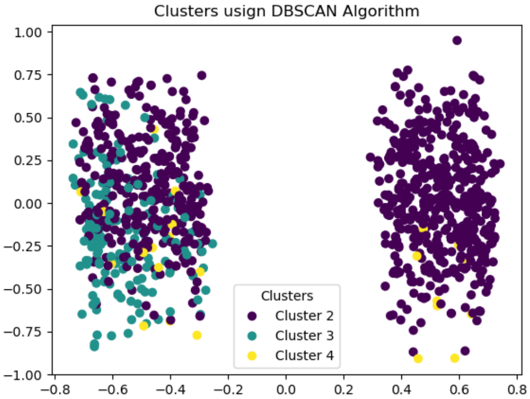

Figura 20: Cluster with DBSCAN, $\epsilon = 1.5$, Min$_{Points} = 100$

I. Learning 3: Low Dimensions

Fuzzy Kmeans

Now, to learn in low-dimensional space, we reduce the space to 3 dimensions using PCA. With this, we randomly select 3 initial clusters and set a parameter epsilon = 1.5. This leads us to a total of 3 central clusters. However, the validation indices with values of 0.170 for Silhouette, 1.818 for Davies, and 188.010 for Calinski, indicate that this is not the best clustering, as we can observe that there are mainly 2 clusters that are too close together.

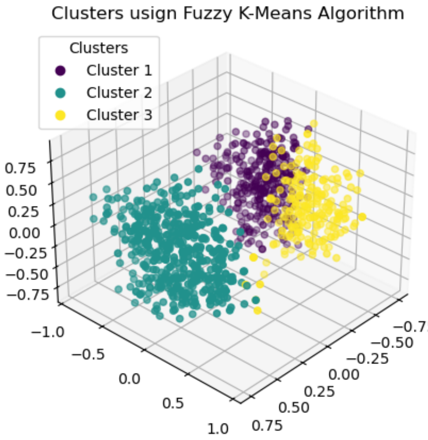

Figura 21: Cluster with Fuzzy Kmeans, $m = 1.5$

Kmeans

With K-means, we see that the result is identical to the one found with Fuzzy K-means, so it is also not considered a good clustering.

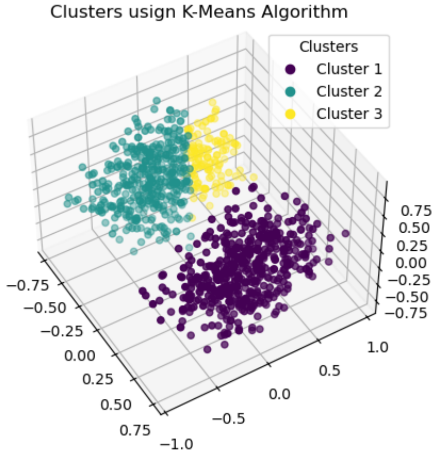

Figura 22: Cluster with Kmeans

Mountain

With a parameter $r_a = 0.3$, we reach a total of 2 clusters; however, it is evident that one of the clusters has very few data points, so it could even be considered not a cluster. Taking this into account, and with values in the indices of 0.142 for Silhouette, 0.979 for Davies, and 14.207 for Calinski, the result is too poor, so this clustering is evidently not appropriate.

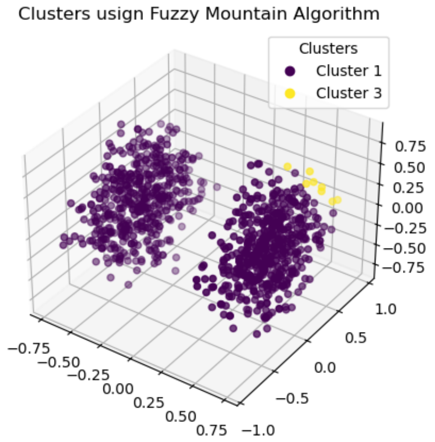

Figura 23: Cluster with Mountain, $r_a = 0.3, r_b = 1.5r_a$

Subtractive

With a value of $r_a = 0.8$, a total of 3 clusters is achieved; however, it is notable that there is not a good separation between the clusters. This is supported by the indices Silhouette = 0.265, Davies = 1.020, Calinski = 355, so we can conclude that this data clustering is not appropriate.

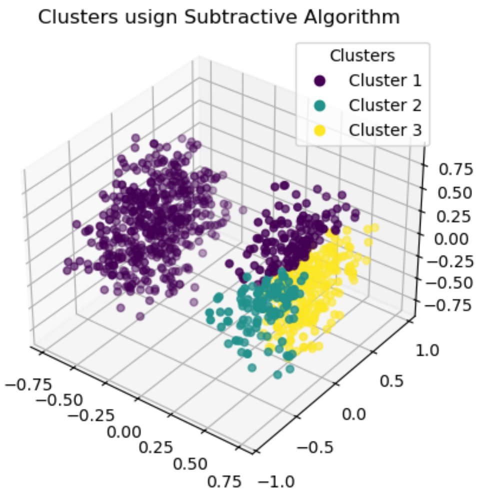

Figura 24: Cluster with Subtractive, $r_a = 0.8$

Similarly, with a parameter $r_a = 0.3$, and validation intra-cluster index values too close, the conclusion is reached that these 2 clusters do not provide a good data clustering.

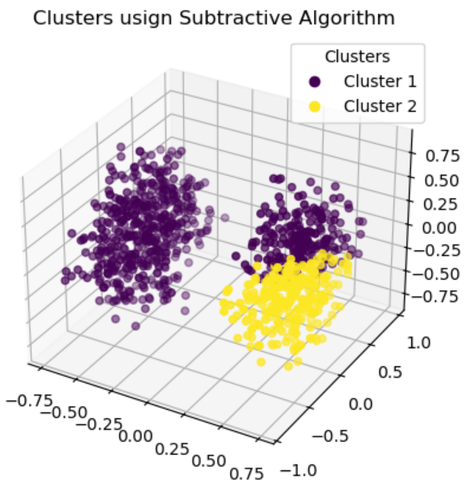

Figura 25: Cluster with Subtractive, $r_a = 0.3$

DBSCAN

Learning with the DBSCAN algorithm, with a maximum distance $\epsilon = 1.5$, and a minimum of points = 200, a total of 3 clusters are obtained. However, when clustering the data, only 2 central clusters are recognized. Nevertheless, with Silhouette indices having a value of 0.16, Davies of 0.81, and Calinski of 8.43, it is concluded that this clustering is not appropriate.

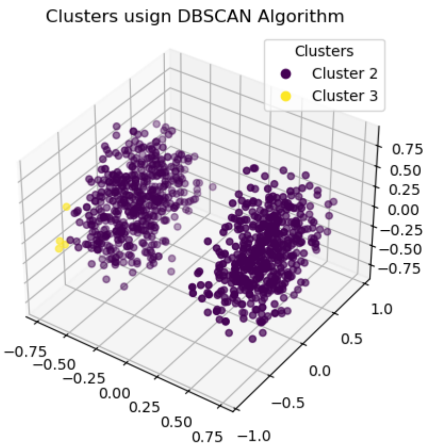

Figura 26: Cluster with DBSCAN, $\epsilon = 1.5$, Min Points = 200

REFERENCIAS

K. Hammouda and F. Karray, “A comparative study of data clustering techniques,” *University of Waterloo, Ontario, Canada*, vol. 1, 2000.