



A multi-objective clustering approach based on different clustering measures combinations

Beatriz Flámia Azevedo^{1,2,3} · Ana Maria A. C. Rocha³ · Ana I. Pereira^{1,2,3}

Received: 30 July 2023 / Revised: 14 July 2024 / Accepted: 4 November 2024
© The Author(s) 2024

Abstract

Clustering methods aim to categorize the elements of a dataset into groups according to the similarities and dissimilarities of the elements. This paper proposes the Multi-objective Clustering Algorithm (MCA), which combines clustering methods with the Nondominated Sorting Genetic Algorithm II. In this way, the proposed algorithm can automatically define the optimal number of clusters and partition the elements based on clustering measures. For this, 6 intra-clustering and 7 inter-clustering measures are explored, combining them 2-to-2, to define the most appropriate pair of measures to be used in a bi-objective approach. Out of the 42 possible combinations, 6 of them were considered the most appropriate, since they showed an explicitly conflicting behavior among the measures. The results of these 6 Pareto fronts were combined into two Pareto fronts, according to the measure of intra-clustering that the combination has in common. The elements of these Pareto fronts were analyzed in terms of dominance, so the nondominated ones were kept, generating a hybrid Pareto front composed of solutions provided by different combinations of measures. The presented approach was validated on three benchmark datasets and also on a real dataset. The results were satisfactory since the proposed algorithm could estimate the optimal number of clusters and suitable dataset partitions. The obtained results were compared with the classical k -means and DBSCAN algorithms, and also two hybrid approaches, the Clustering Differential Evolution, and the Game-Based k -means algorithms. The MCA results demonstrated that they are competitive, mainly for the advancement of providing a set of optimum solutions for the decision-maker.

This work has been supported by FCT Fundação para a Ciência e Tecnologia within the R&D Units Project Scope UIDB/00319/2020, UIDB/05757/2020 (<https://doi.org/10.54499/UIDB/05757/2020>), UIDP/05757/2020 (<https://doi.org/10.54499/UIDP/05757/2020>) and Erasmus Plus KA2 within the project 2021-1-PT01-KA220-HED-000023288. Beatriz Flámia Azevedo is supported by FCT Grant Reference SFRH/BD/07427/2021.

✉ Beatriz Flámia Azevedo
beatrizflamia@ipb.pt

- ¹ Research Centre in Digitalization and Intelligent Robotics (CeDRI), Instituto Politecnico de Bragança, 5300-252 Bragança, Portugal
- ² Laboratório Associado para a Sustentabilidade e Tecnologia em Regiões de Montanha Instituto Politecnico de Bragança, 5300-252 Bragança, Portugal
- ³ ALGORITMI Research Centre/LASI, University of Minho, Campus de Gualtar, 4710-057 Braga, Portugal

Keywords Automatic clustering · Multi-objective optimization · Clustering measures · Machine learning

Mathematics Subject Classification 78M50 · 65K05 · 91C20

1 Introduction

Clustering is an unsupervised data partitioning method that aims to divide the dataset according to characteristics intrinsic to each element, satisfying some criteria such that elements of the same cluster are more similar than those in different ones (Aggarwal and Reddy 2013). Among the unsupervised methods, clustering techniques are the most popular (Azevedo et al. 2024b). Due to its versatility, the clustering procedure is very useful in engineering, health sciences, humanities, economics, education, and other areas (Azevedo et al. 2023; Bi et al. 2020; Liu and Liu 2024; Shi et al. 2025; Tambunan et al. 2020), for that reason several clustering techniques have been proposed over the years.

Usually, in the literature, two clustering problems are commonly reported. The first problem refers to premature convergence at local optimal points, and the second is around the dependence of initial parametrization, especially regarding the number clusters (Azevedo et al. 2024b; Eesa and Orman 2020; Morimoto et al. 2021). In many cases, the estimation of the number of clusters is difficult to predict due to the lack of domain knowledge of the problem, clusters differentiation in terms of shape, size, and density, and when clusters are overlapping in nature (Dutta et al. 2019; Zhao et al. 2024).

Determining the most suitable number of cluster partitions can be considered an optimization problem. Therefore, several studies propose to use nature-inspired metaheuristics to find a solution that maximizes the separation between different clusters and maximizes the cohesion between data elements in the same cluster (Azevedo et al. 2024b; Qaddoura et al. 2021). In turn, Ikotun and Ezugwu (2022) improved the k -means clustering algorithm by using the Symbiotic Organisms Search Algorithm as a global search metaheuristic for generating the optimal initial cluster centroids. Behera et al. (2022) presented a novel approach to define the number of optimal clusters using two hybrid Firefly Particle Swarm Optimization algorithms. Initially, the approach focused on searching for the optimal number of clusters and gradually moved towards global optimal cluster centers. Wang et al. (2024) used the k -means clustering algorithm combined with a self-adapting Genetic Algorithm and Particle Swarm Optimization to identify the optimal solution for a vehicle routing problem. Wadhwa et al. (2023) presented a modified Density-based spatial clustering of applications with noise (DBSCAN) clustering-based scheme where density-based clusters are formed using the DBSCAN, in which the algorithm parameters were estimated by the Bat Algorithm. All approaches were tested on several benchmark datasets, and real-life problems, and the authors considered several statistical tests to justify the effectiveness of the suggested approaches.

A good clustering algorithm should maintain high similarity within the cluster and higher dissimilarities in distinct clusters. Most current clustering methods have also been proposed to integrate different distance measures to obtain the optimal clustering division. The idea is to maximize the distance measure between distinct clusters and, at the same time, minimize a similarity distance measure between the points in the same cluster. However, the weights for several distance measures are challenging to set (Liu et al. 2018). So, a multi-objective optimization algorithm is a suitable strategy for this problem. However, selecting an appropriate measure of the objective function is a nontrivial matter, and the outcome of clustering

can significantly depend on this choice. Many works explore multi-objective algorithms using different objective functions, extracting patterns, and providing multiple partitions as solutions, as can be seen in Morimoto et al. (2021).

Dutta et al. (2019) proposed a Multi-objective Genetic Algorithm for automatic clustering, which takes advantage of the local search ability of k -means with the global search ability of the Genetic Algorithm to find the optimal k . The objective was to minimize the intra-cluster distance and maximize the inter-cluster distance. Kaur and Kumar (2022) presented a multi-objective clustering algorithm based on a vibrating particle system, considering as an objective function the intra-cluster variance and the connectedness; besides the vibrating particle system was used for optimizing the objectives to obtain good clustering results.

Binu Jose and Das (2022) presented a multi-objective approach for clustering to establish the relationship between inter-cluster and intra-cluster distances. Three objective functions were considered to simultaneously minimize the sum of the distance between the elements and their centroids, maximize the sum of the distance between the centroids, and minimize the sum of the distance between elements of the same cluster. Nevertheless, this approach needs the prior specification of the optimal number of centroids, and their final position is obtained randomly from the optimal centroid position that generates the minimum sum of the distance between the elements and their centroids. The algorithms were tested in different benchmark datasets and compared with single objective clustering algorithms, demonstrating superior performance.

The approach proposed in this work explores bio-inspired strategies and clustering techniques to achieve a robust clustering algorithm, named Multi-objective Clustering Algorithm (MCA), combined with the Nondominated Sorting Genetic Algorithm II (Kok et al. 2011), to define the optimal number of cluster sets and the partitioning of the elements, minimizing an intra-clustering measure and maximizing an inter-clustering one. To this end, 42 combinations between 6 intra-clustering measures and 7 inter-clustering measures were analyzed, and the most prominent ones were selected to be used in the MCA as a bi-objective function. As it is been considered a multi-objective approach, the results of each combination consist of a set of Pareto fronts. The elements of these Pareto fronts were analyzed in terms of dominance, generating a hybrid Pareto front composed of elements provided by different combinations of measures.

The main contributions of this work are the use of a multi-objective strategy and the combination of different solutions in the definition of the optimal number of cluster sets and the partitioning of their elements. Single objective algorithms minimize a single measure at a time, which has limitations in exploring specific geometries, dimensions, amount of data, or any other reason, which can indicate, in the eyes of the decision-maker, a solution not suitable for the partitioning of the dataset. By providing a set of optimal solutions, the decision-maker has the variability and flexibility to choose the most appropriate solution according to his/her knowledge or preferences. Moreover, the hybrid Pareto front allows the consideration of different measures that enrich the diversity and robustness of the solution set.

This paper is organized as follows. After the introduction, the clustering measures explored in the paper are described in Sect. 2; these measures are divided into two categories: intra-clustering measures and inter-clustering measures. After that, Sect. 3 defines the multi-objective concepts and the algorithm proposed, which uses the clustering measures to define the set of the optimum solution automatically. The results and discussions are presented in Sect. 4 and a comparison between the MCA results with the k -means, DBSCAN, Clustering Differential Evolution, and the Game-Based k -means algorithms are presented in Sect. 5. Finally, the main conclusions of this research and future steps are presented in Sect. 6.

2 Clustering measures

To partition the dataset into different groups, it is necessary to establish some measures for computing the distances between each element. The choice of distance measures is fundamental for the algorithm's performance, as it strongly influences the clustering results. A multi-objective clustering algorithm considers different clustering measures to automatically define the optimal number of clusters by minimizing the intra-cluster distance and maximizing the inter-cluster distance. Based on the measures criterion, the objective is to group data elements close to each other in the feature space, reflecting their similarity. Many well-known methods are explored in the literature, such as single linkage, complete linkage, and average linkage, among others (Institute; Sokal and Michener 1958; Sorensen 1948). The following sections present the intra- and inter-clustering measures, respectively. Before presenting these measures, consider the notation:

- X is the dataset, in which $X = \{x_1, x_2, \dots, x_m\}$ where x_i is an element of the dataset;
- m is the number of elements x that the set X is composed;
- C defines the set of centroids of the form $C = \{c_1, c_2, \dots, c_k\}$, where c_j defines the centroid j ;
- k is the number of centroids in which X is partitioned;
- C_j defines the cluster j , in which $C_j = \{x_1^j, x_2^j, \dots, x_i^j\}$;
- x_i^j represents an element i that belongs to cluster j ;
- $\#C_j$ is the number of elements of cluster C_j ;
- $D(\cdot, \cdot)$ represents the Euclidean distance between two elements;
- s^* is a vector of solution, in which $s^* = \{C_1^*, C_2^*, \dots, C_k^*\}$, where C refers to a cluster set.

2.1 Intra-clustering measures

Intra-cluster measures refer to the distance among elements of a given cluster. There are many ways to compute the intra-clustering measure. The ones considered in this work are presented below.

The sum of the distances between the elements x_i^j belonging to C_j to their centroids c_j is denoted by Sxc_j , as presented in Eq. (1),

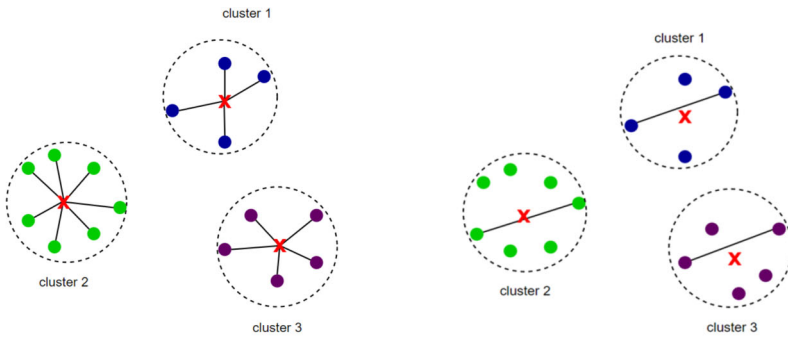
$$Sxc_j = \sum_{i=1}^{\#C_j} D(x_i^j, c_j) \quad \text{for } j = 1, \dots, k, \quad (1)$$

where $\#C_j$ represents the number of elements in the cluster C_j .

Thus, Sxc measure corresponds to the sum of Sxc_j , for each cluster C_j , as presented in Eq. (2),

$$Sxc = \sum_{j=1}^k Sxc_j. \quad (2)$$

Figure 1a illustrates this measure, in which the points (blue, green, and magenta) describe a particular set of clusters C_j , the red cross represents the centroids of the clusters, and the black lines represent the distance considered. Thereby, Sxc corresponds to the sum of the length of each black line.



(a) Distance between elements until its cen- (b) Furthest neighbor distance inside the cluster-
troid ter

Fig. 1 Intra-cluster distance (Color figure online)

The mean of Sxc is represented by Mxc , and calculated as defined by Eq. (3),

$$Mxc = \frac{Sxc}{k}. \tag{3}$$

The measure $SMxc$ represents the mean of the distance in each cluster from the element to its centroid, in terms of the number of elements belonging to each cluster set $\#C_j$, as defined in Eq. (4),

$$SMxc = \sum_{j=1}^k \frac{Sxc_j}{\#C_j}. \tag{4}$$

The measure $MSMxc$ represents the mean of the distance in each cluster from the element to its centroid, in terms of the number of clusters, which is described in Eq. (5),

$$MSMxc = \frac{SMxc}{k}. \tag{5}$$

Another inter-clustering measure considered is the sum of the distance of the furthest neighbor within the cluster (FNc). It evaluates the sum of the furthest neighbor distance of each cluster C_j , where x_i^j and x_l^j belong to the same cluster j . Figure 1b illustrates this measure, as well as described in Eq. (6).

$$FNc = \sum_{j=1}^k \max\{D(x_i^j, x_l^j)\} \text{ for } i = 1, \dots, \#C_j, l = 1, \dots, \#C_j, i \neq l \tag{6}$$

Thus, the mean of the furthest neighbor distance ($MFNc$) evaluates the mean of the furthest neighbor distance in terms of the number of clusters k , as presented in Eq. (7).

$$MFNc = \frac{FNc}{k} \tag{7}$$

2.2 Inter-clustering measures

The inter-cluster measures define the distance between elements that belong to different clusters. The inter-cluster measures considered are presented below.

The measure Scc represents the sum of the distance between centroids (Sokal and Michener 1958), is illustrated in Fig. 2a, and defined in Eq. (8),

$$Scc = \sum_{\substack{t,j=1, \\ t \neq j}}^k D(c_j, c_t). \tag{8}$$

Thus, the mean of the distance between centroids (Mcc) is based on Scc , and it is presented in Eq. (9).

$$Mcc = \frac{Scc}{k} \tag{9}$$

The sum of the furthest neighbor distance between elements of different clusters C_j ($FNcc$) also known as complete linkage (Sorensen 1948), is illustrated in Fig. 2b, and described in Eq. (10),

$$FNcc = \sum_{j=1}^k \sum_{t>j}^k \max \{D(x_i^j, x_l^t)\} \text{ for } i = 1, \dots, \#C_j, l = 1, \dots, \#C_t, i \neq j. \tag{10}$$

The mean of the sum of the furthest neighbor distances between elements of k different clusters ($MFNcc$) is described in Eq. (11),

$$MFNcc = \frac{FNcc}{k}. \tag{11}$$

Another measure considered is the sum of the nearest neighbor distance between elements of different clusters ($NNcc$), known as single linkage (Sokal and Michener 1958), is illustrated in Fig. 2c and defined in Eq. (12),

$$NNcc = \sum_{j=1}^k \sum_{t>j}^k \min \{D(x_i^j, x_l^t)\} \text{ for } i = 1, \dots, \#C_j, l = 1, \dots, \#C_t, i \neq l. \tag{12}$$

The mean of the nearest neighbor distance between elements of different clusters ($MNNcc$), is defined in Eq. (13).

$$MNNcc = \frac{NNcc}{k} \tag{13}$$

Finally, the sum of the distance between an element i , which belongs to cluster j , to all other elements of the dataset that belong to a cluster t (Sxx), is illustrated in Fig. 2d, and expressed in Eq. (14),

$$Sxx = \frac{1}{2} \sum_{j=1}^k \sum_{t=1}^k \sum_{i=1}^{\#C_j} \sum_{l=1}^{\#C_t} D(x_i^j, x_l^t) \text{ for } t \neq j, l \neq i. \tag{14}$$

3 Multi-objective approach

In this section, the main concepts of the multi-objective approach are presented, as well as the developed algorithm, the Multi-objective Clustering Algorithm (MCA) that uses the Nondominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al. 2002).

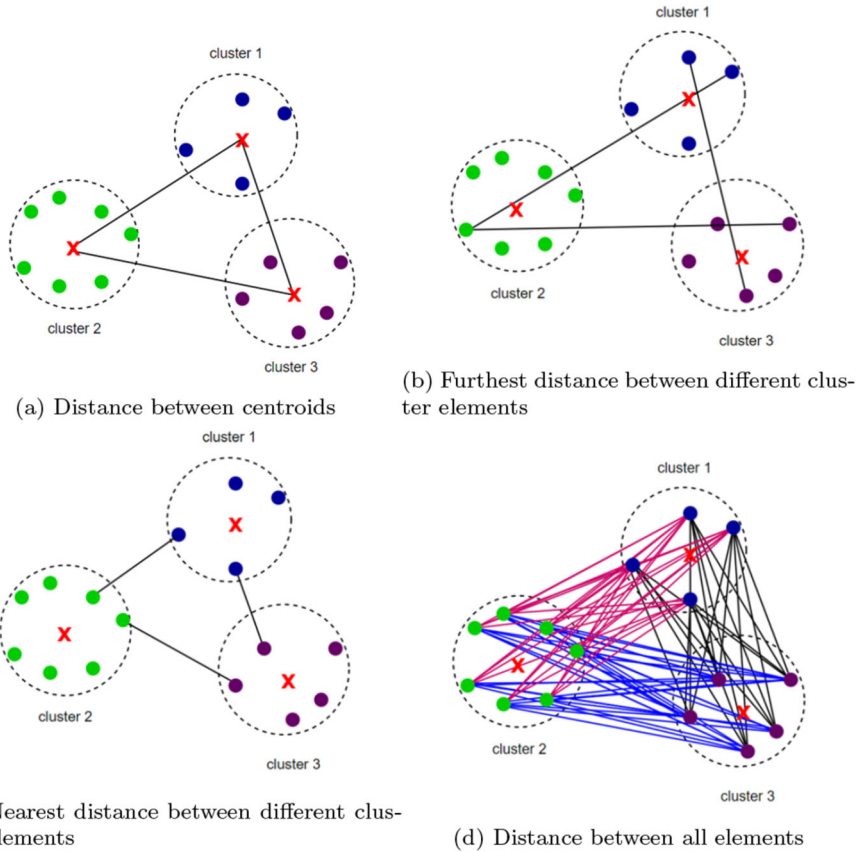


Fig. 2 Inter-clustering distances (Color figure online)

3.1 Multi-objective concepts

Multi-objective optimization is an area of multiple-criteria decision-making concerning mathematical optimization problems that involve several objective functions to be minimized or maximized simultaneously. These objectives are conflicting, meaning there is a trade-off between them (Deb 2001).

In the single-objective optimization problem, there is only one objective function to be optimized, and its superiority is determined by comparing the objective function values. Meanwhile, in the multi-objective optimization problem, the problem to be optimized involves multiple conflicting objectives (h objectives), i.e., a vector of objective functions. In this case, the quality of a solution is determined by the nondominated criterion (Deb 2001).

An unconstrained multi-objective optimization problem is defined in the form of Eq. (15),

$$\min_{x \in \mathbb{R}} F = \{f_1(x), f_2(x), \dots, f_h(x)\} \tag{15}$$

in which the solution x is a vector of d decision variables, $x = (x_1, x_2, \dots, x_d)$.

The optimal solutions according to a multi-objective approach will be specified based on a mathematical concept of partial ordering (Deb 2011). Thereby, multi-objective optimization

algorithms use the concept of dominated and nondominated solutions, where the nondominated solutions will constitute the Pareto front, representing the optimal set of solutions for a multi-objective optimization problem. Specifically, a decision variable vector $x' \in S$ is called a dominated solution if there exist $\bar{x} \in S$ such that $f_i(\bar{x}) \leq f_i(x')$ for all $i = 1, \dots, h$. Then, the vectors of the objective function are treated as optimal if none of their solutions can be improved without deteriorating at least one of the other objectives. Thereby, any solution that is nondominated by any other set member is known as a nondominated solution (Deb 2011).

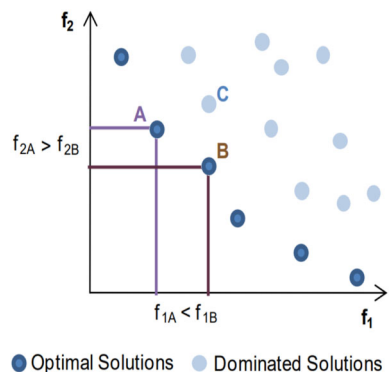
Figure 3 illustrates this concept. The dark blue circles represent the nondominated solutions, constituting the Pareto front, and the light blue circles represent the dominated solutions, that were dominated by the solution represented by the dark blue circles. A nondominated solution cannot be improved in any objective without sacrificing performance in another. Note that solution A is equally optimal as solution B , since A first objective function value (f_{1A}) is smaller than B second objective function value (f_{1B}), while A second objective function value (f_{2A}) is higher than B second objective function value (f_{2B}). Meanwhile, solution C is dominated by solutions A and B .

3.2 Nondominated Sorting Genetic Algorithm

The Nondominated Sorting Genetic Algorithm II (NSGA-II), is a popular bio-inspired algorithm for solving multi-objective optimization problems. It was developed by Deb et al. (2002) and is based on evolutionary procedures of the Genetic Algorithm. Evolutionary algorithms explore a large solution space efficiently, making them well-suited for finding global optima in complex, multimodal landscapes (Yang and Gen 2010). Unlike deterministic algorithms, evolutionary algorithms, also known as metaheuristics, do not require derivative information of the objective function; this makes them suitable for optimization problems where analytical derivatives are unavailable, difficult to compute, or unreliable (Azevedo et al. 2024b). Besides, evolutionary algorithms used to be robust in handling noisy, non-linear, and non-convex objective functions. The NSGA-II is an appropriate algorithm for the work since it has strong exploration capabilities, fast convergence, and computational efficiency.

The NSGA-II uses a fast nondomination sorting procedure to categorize solutions according to the level of nondomination and a crowding distance operator to preserve diversity in the evolutionary procedure. Moreover, elitism is achieved by controlling the elite members of the population as the algorithm progresses to maintain the diversity of the population until it converges to a Pareto-optimal front (Deb et al. 2002).

Fig. 3 Pareto front in two-objective space. Adapted from Villa and Labayrade (2011) (Color figure online)



Basically, the NSGA-II algorithm starts by randomly initializing the population, in which each objective f_i ($i \in \{1, \dots, h\}$) is identified. The population is composed of N individuals represented by a candidate solution x (Kok et al. 2011). Afterward, each element x is faced with genetic operations which entail a simulated binary crossover and a polynomial mutation. The NSGA-II algorithm is applied iteratively until a specified stopping criterion is met (Kok et al. 2011). More details about the NSGA-II can be found in Deb et al. (2002), Kok et al. (2011), and the algorithm code is available on Matlab[®], specifically by `gamultiobj` function (MATLAB 2019).

3.3 Multi-objective clustering algorithm

The algorithm developed in this work, named Multi-objective Clustering Algorithm (MCA), evaluates intra- and inter-clustering measures to define the optimal number of centroids and their optimal position. This is achieved by simultaneously minimizing intra-clustering distances and maximizing inter-clustering distances by combining several pairs of intra- and inter-clustering measures.

A bi-objective programming problem can be defined in order to minimize an intra-clustering measure and maximize an inter-clustering measure as follows:

$$\min F = \{f_a, -g_b\}$$

where f_a , for $a = 1, \dots, 6$, is the intra-clustering measure and g_b , for $b = 1, \dots, 7$, is the inter-clustering measure, defined in Sect. 2.

The MCA can be defined in 8 stages, as presented below. To better explain the MCA, consider a given dataset $X = \{x_1, x_2, \dots, x_m\}$ composed of m elements, where $x_i \in \mathbb{R}^d$ (d is the number of variables of the dataset), the idea is to partition X into k optimal groups (clusters). Thus, the output of the MCA is a vector of the solution $s^* = \{C_1^*, C_2^*, \dots, C_k^*\}$, in which C refers to a cluster set.

Stage 1—Input data the algorithm starts with the input of the dataset X , and the definition of the values k_{min} and k_{max} , which represent the minimum, and maximum number of centroids that could be assigned. The MCA automatically defines the optimal number of cluster partitions in a range of possible partitions. This value can be given by the user or considered by default $k_{min} = 2$ and $k_{max} = \lceil \sqrt{m} \rceil$ (Pal and Bezdek 1995).

Stage 2—Measures selection a pair of measures is selected, one intra-measure and another inter-measure, in which f_a , is an intra-clustering measure, among $f_1 = Sxc$, $f_2 = Mxc$, $f_3 = SMxc$, $f_4 = MSMxc$, $f_5 = FNc$, and $f_6 = MFNc$, and g_b , is the inter-clustering measure among $g_1 = Scc$, $g_2 = Mcc$, $g_3 = FNcc$, $g_4 = MFNcc$, $g_5 = NNcc$, $g_6 = MNNcc$, and $g_7 = Sxx$.

Stage 3—Centroids calculation after the measures selection, the process needs to calculate the centroids. The Centroids Calculation (CC) iterative procedure randomly generates k_{max} ordered pairs, which are the possible candidates for k centroids. Each candidate ordered pair is associated with a random value ω belonging to $[0, 1]$. If $\omega > \gamma$, for a fix γ , the candidate pair will be considered as a centroid. If the dimension of C is smaller than k_{min} , the centroid candidates with the largest ω are added to C until $\#C = k_{min}$ (Heris 2015). The next step is to calculate the Euclidean distance D between all the elements of X up to each centroid j . The closest elements of each centroid c_j define a cluster set C_j . To avoid small cluster sets, a minimum number of elements per cluster is defined, as ζ , (Memarsadeghi et al. 2007). Thereby, the centroids c_j that have less than ζ associated elements are automatically removed from the set of centroids, and the elements become part of other remaining centroid, which

is the closest one in terms of Euclidean distance of the elements considered. As default, it is considered $\zeta = \lfloor \sqrt{m} \rfloor$, with $\zeta \in \mathbb{N}$ (Dutta et al. 2019). The set C has all remaining centroids. To improve the algorithm’s performance, the coordinates of each centroid j assume the coordinates of their barycenter cluster c_j , composed of its x_i^j elements.

For a better understanding of the centroids calculation, the Algorithm 1 is presented.

Algorithm 1 Centroids Calculation

Input: $X, \gamma, \zeta, k_{min}, k_{max}$

Generate (c_j, ω_j) , for $j = 1, \dots, k_{max}$. Guarantee that all c_j are different values.

Set $C = \{\emptyset\}$

for $j = 1, \dots, k_{max}$ **do**

if $\omega_j > \gamma$ **then**

$C = C \cup \{c_j\}$

end if

end for

if $\#C < k_{min}$ **then**

$C = C \cup \{c_i\}$, in which c_i is the candidate with the highest ω_i

end if

$k = \#C$

Evaluate $D(x_i, c_j)$, for all $c_j \in C, i \in X$

Associate each x_i to the nearest c_j

for $j = 1 : k$ **do**

if $\#C_j < \zeta$ **then**

c_j is eliminated, and associate element with the nearest cluster

 Update k

end if

end for

Update c_j based on its cluster barycenter coordinates.

Output: The k , the centroids c_j , and $s^* = \{C_1^*, C_2^*, \dots, C_k^*\}$.

Stage 4—Optimization method to identify the Pareto front associated with that bi-objective function of the problem, it is necessary to use a multi-objective algorithm. In this case, the MCA uses the NSGA-II (Deb et al. 2002), as defined in Sect. 3.2. This iterative evolutionary process revisits Stage 3 as many times as necessary, refining and advancing the population algorithm. Stage 4 can be replaced with another multi-objective optimization process, such as Multi-objective Particle Swarm Optimization (Coello-Coello and Lechuga 2002), Multi-objective Grey Wolf Optimizer (Mirjalili et al. 2016), or Multi-objective Genetic Algorithm (Dutta et al. 2019), among others. Here, the NSGA-II was chosen due to its popularity and its high exploration ability.

Stage 5—Stopping criterion the Stages 2, 3, and 4 are repeated until all combinations of pairs of measures have been evaluated.

Stage 6—Normalization process considering the different measures with varying orders of magnitude, normalizing the values is essential to facilitate a comprehensive comparison and ensure a fair and meaningful analysis of the results. Then, each Pareto front generated is normalized through the Min-Max scaling method (Müller and Guido 2016). That is, each solution s of the Pareto front is individually normalized between $[0, 1]$, using Eq. (16),

$$f_i^* = \frac{f_i(s) - f_i^{min}}{f_i^{max} - f_i^{min}}, \tag{16}$$

where $f_i(s)$ represents the components of the bi-objective function F at the solution s , f_i^{min} and f_i^{max} are respectively the smallest and the highest solution value, of the function f_i that

belong to the Pareto front. Thus, $f_i^* = (f_i(s^*))$ is a normalized Pareto front solution of each Pareto front considered a given set of measures.

Stage 7—Nondominated procedure evaluation: in this stage, all normalized Pareto front are evaluated regarding nondominated criterion, and the nondominated solutions are selected to compose a hybrid Pareto front (HPF).

Stage 8—Hybrid Pareto front: the HPF is the set of nondominated solutions, considering all the normalized solutions of the Pareto fronts obtained for each pair of measures.

The pseudocode of the MCA algorithm is presented in Algorithm 2.

Algorithm 2 Multi-objective Clustering Algorithm

Input: X
for a given intra-measure **do**
 for $b = 1, \dots$, number of inter-clustering measures **do** ▷ for each inter-measure
 $C \leftarrow CC(X, \gamma, \zeta, k_{min}, k_{max})$, ▷ Centroids Calculation
 $PF^{*,a,b} \leftarrow NSGA-II(X, f_a, -g_b)$ ▷ Pareto front
 end for
end for
 $NPF^{*,b} \leftarrow \text{normalize } PF^{*,b}$ ▷ normalize Pareto front
Output: HPF \leftarrow nondominated solution ($NPF^{*,b}, b = 1, \dots$, number of inter-clustering measures) ▷
 hybrid Pareto front

4 Results and discussion

To validate the proposed approach, 4 datasets are considered, as described in Table 1, according to the number of elements, features, Number of clusters, and references.

The dataset 1, named My data (Heris 2015), is used to illustrate the conflicts between distance measures and to test the behavior of the algorithm under different pair of measures. The most prominent measures found were used to test the algorithm in the other datasets.

After that, the MCA algorithm is tested in two higher dimensional datasets, the dataset 2, named Thyroid (Fränti and Sieranoja 2018) and the dataset 3 named Breast (Fränti and Sieranoja 2018). Finally, the dataset 4, named MathE, is used to validate the approach on a completed unknown problem (Azevedo et al. 2024a), regarding to the number of centroids. Note that, in the case of MathE data, the number of clusters is completely unknown, since the decision is directly influenced by the decision-maker’s knowledge as will be discussed in Sect. 4.5.

The results presented in this paper were obtained using an Intel(R) i5(R) CPU @ 1.60 GHz with 8 GB of RAM using Matlab 2019a[®] software (MATLAB 2019).

Table 1 Datasets description

Dataset	Elements	Features	No. clusters	References
1—My data	300	2	3	Heris (2015)
2—Thyroid	215	5	2	Fränti and Sieranoja (2018)
3—Breast	699	9	2	Fränti and Sieranoja (2018)
4—MathE	291	2	–	Azevedo et al. (2024a)

4.1 Conflict analysis

In order to analyze the most prominent pair of measures to be considered as objective function, a simulation was performed to study the behavior of all 2-to-2 combinations, between the 6 intra-clustering measures and the 7 inter-clustering measures, presented in Sect. 2, considered the dataset 1. In this case, there are 42 possible combinations. Thus, to identify if the objectives presented in Sect. 2 are conflicting, 3000 random elements of each pair of functions were evaluated. Figure 4 shows the obtained values, considering the 2-to-2 combinations.

As it is looked for a combination that minimizes the intra-clustering distance and simultaneously maximizes the inter-clustering measure, it is necessary to select the pair of measures that has a behavior closest to the behavior of Fig. 3, since the second objective function is considered negative, as defined in Sect. 3.3.

Based on this, as objective functions, it is possible to select the most appropriate measures for use in the MCA. Thereby, the 2 intra-clustering measures, SM_{xc} , and FN_c , and the 6 inter-clustering measures, Sc_c , Mcc , $FNcc$, $NNcc$, $MFNcc$, and $MNNcc$, were chosen because they presented an explicitly conflicting behavior among the measures. The mentioned combinations are highlighted in Fig. 4, which results in 12 combinations of pairs of measures.

Once the combinations were selected, the MCA was performed in four different datasets, to validate the behavior observed in the previous simulation. Thus, the following parameters is considered $k_{min} = 2$, $k_{max} = \lceil \sqrt{m} \rceil$, $\gamma = 0.4$, and it was considered 10 runs for each measure combination. Regarding the parameters of NSGA-II, a population size equal to 100 was considered, and a maximum generation equal to 400. For gamultiobj , the algorithm stops when the geometric mean of the relative variation of the spread value over 100 generations

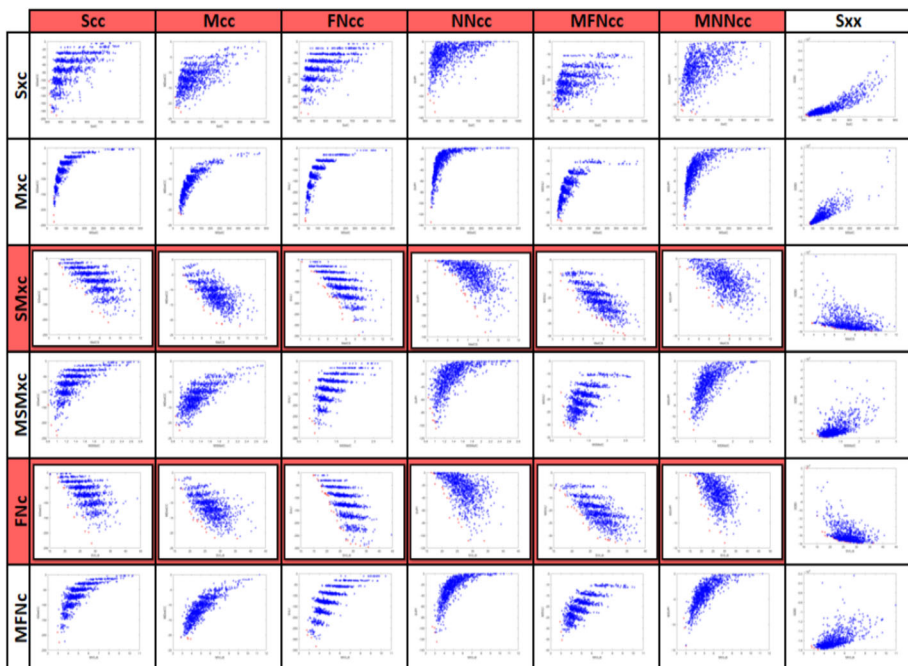


Fig. 4 Clustering measures behavior (Color figure online)

is less than 10^{-4} , and the final spread is less than the average spread over the past 100 generations, as defined in documentation of `gamultiobj` function (MATLAB 2019). The results of the four datasets is presented below.

4.2 Results for dataset 1—My data

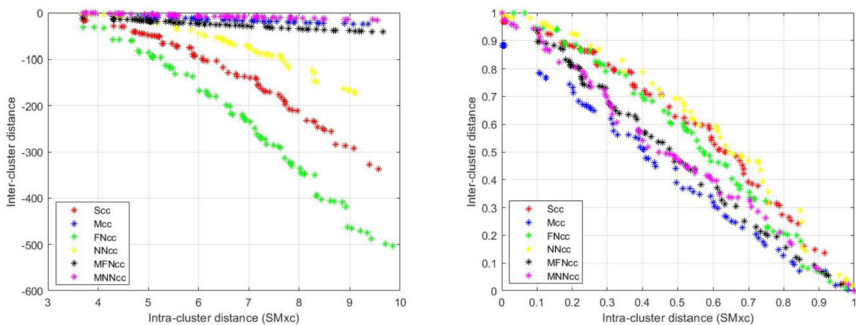
The following two sections present the results for $SMxc$ and FNc measures, respectively, combined with the Mcc , $FNcc$, $NNcc$, $MFNcc$, and, $MNNcc$ inter-clustering measures. For the dataset 1 is it considered $k_{max} = 17 = \lfloor \sqrt{300} \rfloor$, as 300 is the number of elements of the dataset. The Pareto fronts were analyzed before and after the normalization to validate the MCA process.

4.2.1 Results of the intra-clustering measure $SMxc$ with inter-clustering measures

The $SMxc$ represents the mean distances between the elements belonging to C_j to their centroids c_j in terms of the number of elements that belong to each cluster set $\#C_j$. Each $SMxc$ combination was executed 10 times by the MCA. The solutions generated by all executions were compared, and the nondominated solutions from all runs generated the Pareto front. Thereby, as six 2-to-2 combinations are being considered, the result consists of 6 different Pareto fronts and they are described in Fig. 5. The red Pareto front is the result of the combination $SMxc-Scc$, the blue one is $SMxc-Mcc$, the green one is $SMxc-FNcc$, the yellow one is $SMxc-NNcc$, the black one is $SMxc-MFNcc$ and the magenta one is the combination provided by the $SMxc-MNNcc$, as objective function 1 and 2, respectively.

The objective functions that define the Pareto fronts involve measures based on sums and means, which result in 6 Pareto fronts of different ranges. Thus, to have a fair comparison between the Pareto fronts, they were normalized by their range (MATLAB 2019), as better defined in Sect. 3.3.

The normalization results are presented in Fig. 5b. In this way, it is possible to verify that the values generated by the combinations Mcc , $MFNcc$, and $MNNcc$ in the second objective function (inter-clustering measure) dominate the combinations Scc , $FNcc$, and $NNcc$ measures. Moreover, the hypervolume (Bringmann and Friedrich 2013) of each Pareto Front was calculated. Thereby, the Pareto front with the highest hypervolume is considered



(a) Pareto fronts of $SMxc$ combinations con- (b) Normalized Pareto fronts of $SMxc$ combinations considering dataset 1

Fig. 5 Pareto fronts for each combination involving the $SMxc$ considering dataset 1 (Color figure online)

optimal, and this measure was applied to evaluate the competing Pareto fronts (Ahmadi 2023; Bringmann and Friedrich 2013). The values of the results are described in Table 2. From this table, the Pareto fronts provided by the SM_{xc} , as the first objective function, combined (individually) with the measures M_{cc} (blue), MFN_{cc} (black), and, MNN_{cc} (magenta), as the second objective function, showed the highest values, so the Pareto fronts provided by the measures M_{cc} , MFN_{cc} , and MNN_{cc} are more appropriated.

Table 2 Hypervolume values for the SM_{xc} Pareto front combinations considering Dataset 1

Measures	Sc_c	M_{cc}	FN_{cc}	NN_{cc}	MFN_{cc}	MNN_{cc}
Hypervolume	0.369	0.473	0.365	0.375	0.477	0.417

The solutions of the Pareto fronts present several possible clustering distributions among the elements of the dataset in terms of distance measures and the number of partitions in the dataset. Table 3 presents the number of solutions that belong to each combination between SM_{xc} , and the means M_{cc} , MFN_{cc} , and MNN_{cc} , according to the number of k , optimal number of clusters.

Table 3 Pareto front solutions generated by SM_{xc} measure combined with the 3 inter-clustering measures considering dataset 1

Measures/k	2	3	4	5	6	7	8	9	10	11	12	13	Total
$SM_{xc}-M_{cc}$	2	25	5	11	8	8	7	7	5	6	7	0	91
$SM_{xc}-MFN_{cc}$	2	0	11	19	9	7	5	6	5	6	8	0	78
$SM_{xc}-MNN_{cc}$	0	28	8	7	8	10	12	8	6	4	2	4	97

As the dataset 1 is composed of 300 elements, the k can vary between 2 and 17. However, the maximum number of clusters indicated by the algorithm for each Pareto front was 12 in the Pareto fronts $SM_{xc}-M_{cc}$ and $SM_{xc}-MFN_{cc}$ and 14 in the Pareto front $SM_{xc}-MNN_{cc}$. Table 3 shows 91 solutions in the $SM_{xc}-M_{cc}$ Pareto front, 78 solutions in the Pareto front $SM_{xc}-MFN_{cc}$, and 97 in the $SM_{xc}-MNN_{cc}$, which results in a total of 266 solutions. This table shows that $k = 3$ is the most frequent solution, which is the optimal number of k indicated by the dataset documentation.

Considering the results of the three Pareto fronts, there are 266 possible solutions for the decision-maker, being a hard task to choose only one, especially in a situation where there is little prior information about the dataset. Furthermore, if one has chosen to consider the solutions of only one Pareto front, a set of solutions that may be crucial to the decision-maker can be discarded. Analyzing the results of Fig. 5, none of the three Pareto fronts based on means measures completely dominates the others two Pareto front based on means. Thus, it was decided not to select one of the three Pareto fronts but rather to compare their solutions in terms of dominance (last step of the MCA). Taking this into account, a hybrid Pareto front is proposed, composed of the nondominated solutions concerning all possible solutions of the three Pareto fronts.

Table 4 presents the nondominated solutions that appear in the hybrid Pareto front after the dominance assessment. Thereby, the hybrid Pareto front has 117 solutions in total, with

Table 4 Hybrid Pareto front solutions generated by the SM_{xc} combined with the 3 inter-clustering measure considering dataset 1

Measures/k	2	3	4	5	6	7	8	9	10	11	12	Total
$SM_{xc}-Mcc$	2	25	5	11	8	8	7	7	5	5	5	88
$SM_{xc}-MFNcc$	2	0	0	0	0	0	0	0	0	0	7	9
$SM_{xc}-MNNcc$	0	20	0	0	0	0	0	0	0	0	0	20

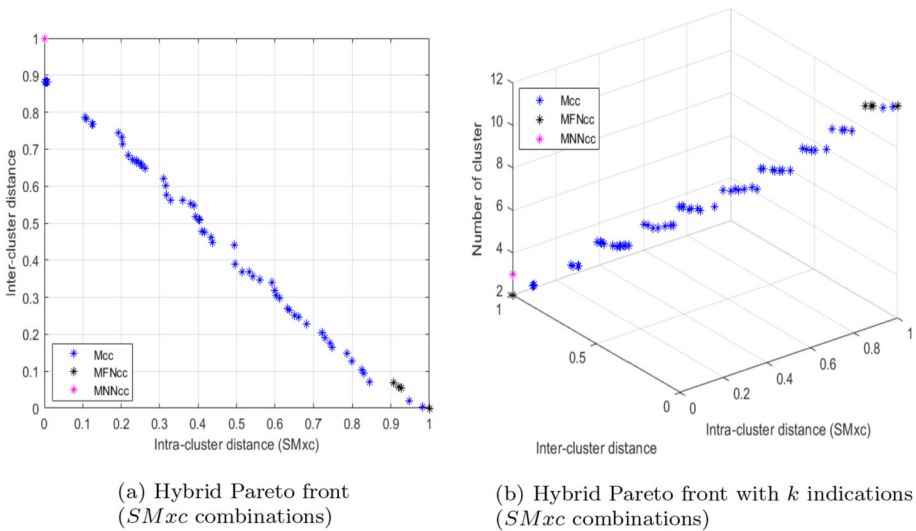


Fig. 6 Hybrid Pareto front of dataset 1 for SM_{xc} combinations considering dataset 1 (Color figure online)

88 coming from $SM_{xc}-Mcc$, 9 from $SM_{xc}-MFNcc$ and 20 from $SM_{xc}-MNNcc$. In this case, there was a reduction of 67% of the solutions, which means a filter of the most appropriate solutions for the problem. It is important to highlight that even with this refinement, there is still a prevalence of solutions with $k = 3$ in the hybrid Pareto front.

Figure 6a presents the results of the hybrid Pareto front, in which colors differentiate the solutions of each pair of measure combination, so the $SM_{xc}-Mcc$ solutions are in blue, the $SM_{xc}-MFNcc$ solutions are in black, and the $SM_{xc}-MNNcc$ are represented in magenta. The number of partitions k indicated by each solution can vary between 2 and 12, as indicated by the z -axis of Fig. 6b. Note that, even if the solutions have the same partition value k , the elements' distribution and each centroid's position can be different, generating different values of intra- and inter-clustering measures. Thus, multiple solutions for the same k may indicate completely different solutions in terms of clustering measures.

Figures 7b–d illustrate three examples of cluster partitioning found in the hybrid Pareto front, as marked in Fig. 7a. In these cases, the elements of each cluster are differentiated by colors. The solution 1 (s1) and solution 3 (s3) are obtained through the extreme points of the hybrid Pareto front, whereas solution 2 (s2) represents a central solution. Solution 1, illustrated in Fig. 7b, proposes a partitioning of the dataset into 3 clusters, and is obtained from the combination of measures $SM_{xc}-MNNcc$; this solution has the lowest value of the intra-clustering measure and also the lowest value of the inter-clustering measure; as solution 1 is an extreme point of the Pareto front, it gives the optimal value of the objective function 1

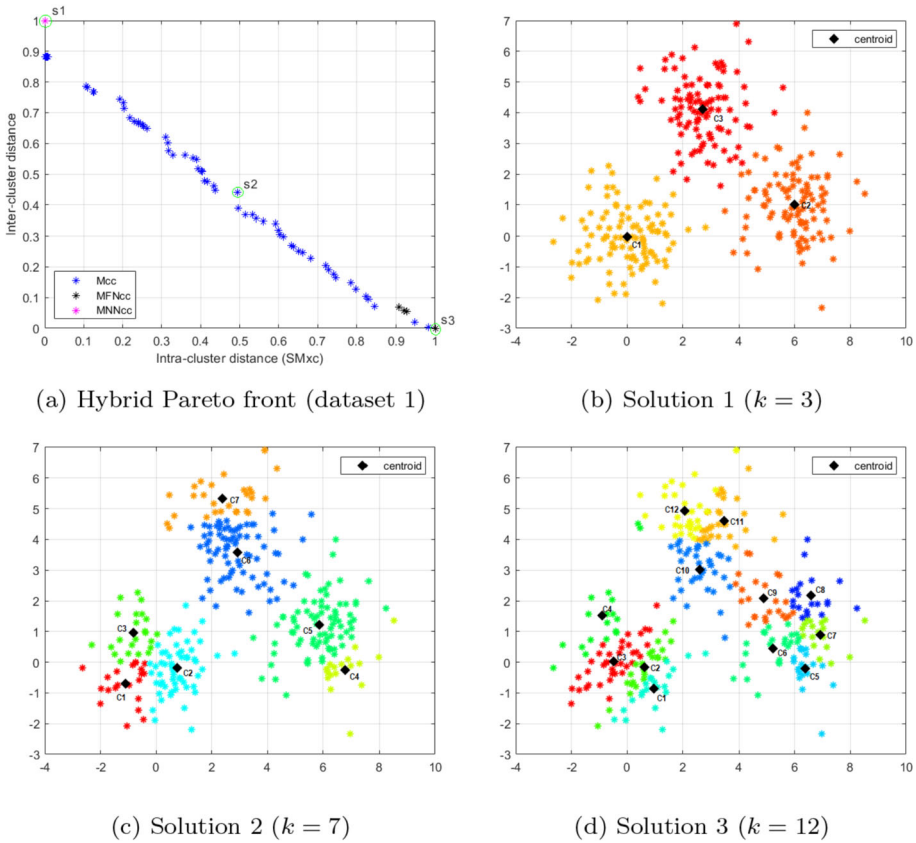
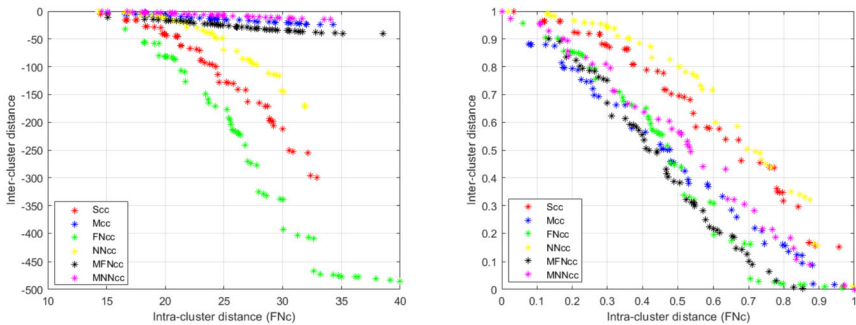


Fig. 7 Hybrid Pareto front solutions ($SMxc$ combinations—dataset 1) (Color figure online)

representing the intra-clustering measure, and penalizes the objective function 2 of the inter-clustering measure. On the other hand, solution 2, provided from $SMxc-Mcc$ measures, and illustrated in Fig. 7c, suggests a division into 7 clusters. This is a tradeoff solution in the hybrid Pareto front, which means that the values of the objective function 1 and 2 are more balanced than in the other solutions. Finally, solution 3, illustrated in Fig. 7d, obtained from the combination $SMxc-MFNcc$, indicates a division into 12 clusters. This solution is also generated by an extreme of the Pareto front, but in this case the opposite extreme of solution 1. Thus, solution 3 has the worst value of intra-clustering distance and at the same time the best value of inter-clustering distance. According to Heris (2015), the recommended solution for dataset 1 is $k = 3$, so the solution 1 is the closest to the recommendation (Heris 2015), since the association of the elements and the distance measures can be different even with the same k value, which represents different solutions. On the other hand, the solutions 2 and 3 are more partitioned than solution 1, which for the decision-maker can be interesting when he wants to divide the dataset into smaller groups, ensuring greater similarity between the elements.

4.2.2 Results of the intra-clustering measure FNc with inter-clustering measures

This section presents the results of the Further Neighbor distance (FNc) combined with the distance measures Sc , Mcc , $FNcc$, $NNcc$, $MFNcc$, and $MNNcc$. Again, each 2-to-2 combination generates a Pareto front of different ranges, resulting in 6 different Pareto fronts, as presented in Fig. 8a. Each Pareto front was normalized in its range, and the results are illustrated in Fig. 8b. Thus, the red Pareto front is the result of the $FNc-Sc$ combination, the blue one represents the $FNc-Mcc$ combination, the green is the $FNc-FNcc$ combination, the yellow one is the $FNc>NNcc$ combination, the black one the $FNc-MFNcc$ combination, and the magenta one is the $FNc-MNNcc$ combination.



(a) Pareto fronts of the FNc combinations (b) Normalized Pareto front for the FNc considering dataset 1

Fig. 8 Pareto fronts for each combination involving the FNc considering dataset 1 (Color figure online)

The hypervolume of each Pareto front was evaluated, which is presented in Table 5. As can be seen, the Pareto fronts obtained with the measures Mcc , $MNNcc$, and $MNNcc$ are the ones that obtained the highest values in the hypervolume. These solutions were chosen to be analyzed regarding dominance, and they generated the hybrid Pareto front for the FNc combinations.

Table 6 describes the Pareto front solutions generated by the FNc measure combined with the Mcc , $MFNcc$, and $MNNcc$ in terms of the number of clusters k . As can be seen, there are 69 solutions in the Pareto front of the $FNc-Mcc$ measures, 61 in the $FNc-MFNcc$ Pareto front, and 60 in the combination of the $FNc-MNNcc$ measures.

These three Pareto fronts have 190 possible solutions that were confronted in terms of dominance, and the results are presented in Table 7. In this case, this reminds 83 solutions at the hybrid Pareto front, 34 from the $FNc-Mcc$ combination, 43 from the $FNc-MFNcc$ combination, and 6 from the $FNc-MNNcc$, which represent a refinement of 56% in the number of solutions.

Table 5 Hypervolume values for the FNc combinations

Measures	Sc	Mcc	$FNcc$	$NNcc$	$MFNcc$	$MNNcc$
Hypervolume	0.427	0.486	0.428	0.416	0.504	0.452

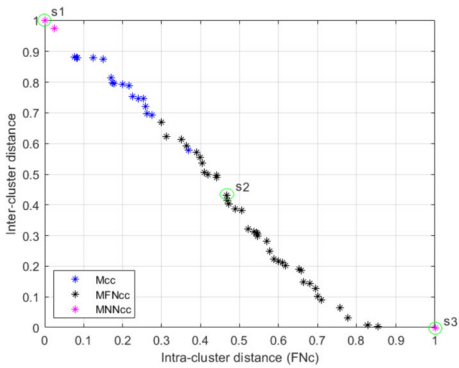
Table 6 Pareto front solutions generated by the FNc measure combined with the 3 inter-clustering measures

Measures/k	2	3	4	5	6	7	8	9	10	11	12	Total
$FNc-Mcc$	2	14	8	9	3	10	7	3	5	5	3	69
$FNc-MFNcc$	2	0	3	11	3	9	6	6	7	7	7	61
$FNc-MNNcc$	4	9	5	7	4	8	4	6	5	4	4	60

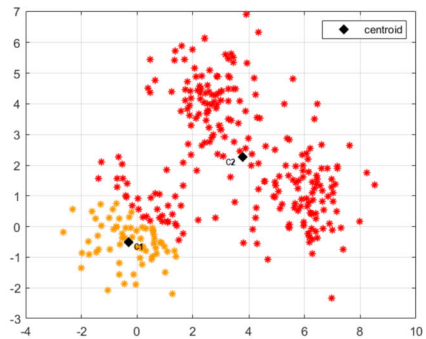
Table 7 Hybrid Pareto front solutions generated by the FNc measure combined with the 3 inter-clustering measures considering dataset 1

Measures/k	2	3	4	5	6	7	8	9	10	11	12	Total
$FNc-Mcc$	2	14	8	7	2	0	0	0	0	0	1	34
$FNc-MFNcc$	2	0	0	0	3	7	6	6	7	7	5	43
$FNc-MNNcc$	4	0	0	0	0	0	0	0	0	0	2	6

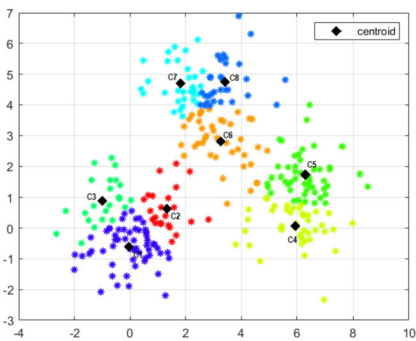
Figure 9a presents the hybrid Pareto front, composed by the solution of the three mean measures, Mcc (blue), $MFNcc$ (black), and $MNNcc$ (magenta). As with $SM\chi c$, three solutions were chosen to illustrate the possibilities for partitioning in clusters. Solution 1 (s1),



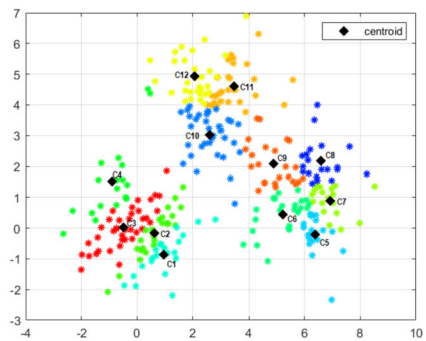
(a) Hybrid Pareto front in dataset 1



(b) Solution 1 ($k = 2$)



(c) Solution 2 ($k = 8$)



(d) Solution 3 ($k = 12$)

Fig. 9 Hybrid Pareto front solutions (FNc combinations - dataset 1) (Color figure online)

was provided by the combination $FNc-MNNcc$, and is the solution with the lowest number of cluster divisions, it is 2, as can be seen in Fig. 9b. As it is located in an extreme region of the Pareto front, it gives priority to one objective function over another, so solution 1 has the lowest intra-clustering measure, and has the lowest value of the inter-clustering measure than any other solution. On the other hand, solution 3 has the opposite behavior to solution 1 since it is located at the opposite extreme of the Pareto front, so it has the highest intra-clustering measure and the highest inter-clustering measure than any other solution. Solution 3 was also generated by the combination of $FNc-MNNcc$, and indicates a division of the dataset into 12 clusters, as seen in Fig. 9d. Finally, solution 2, is a tradeoff solution on the Pareto front, so the interest of both objective functions is balanced. Solution 2 indicates a division of the dataset in 8 clusters, provided by the combination of $FNc-MFNcc$ measures, as can be seen in Fig. 9c. As in the previous section, the hybrid Pareto front offers several possibilities to the decision-maker, who can use their knowledge of the data to choose the most appropriate solution according to their interests.

4.2.3 MCA sensitivity analysis

The sensitivity analysis evaluated the impact of the γ value in the algorithm. For this, it was considered four different values for γ and 10 runs for each pair of measure combinations.

Tables 8 and 9 presents the results for Mydata dataset considering different values for the γ parameter, the pair of measure combination, the minimum (k_{min}) and the maximum (k_{max}) number of clusters found by the MCA, the average time per execution (in seconds), and the number of solutions in each the Pareto front, for the combinations $SMxc$ and FNc with the mean measures, respectively.

As observed, four different γ values were considered, and the results are similar across all measure combinations presented in both tables in terms of the minimum and maximum number of centroids, execution time, and final number of solutions in each Pareto Front. Therefore, based on multiple executions of the MCA for the same pair of measures-in this case, 10 executions-the final result is not impacted by the γ parameter. Consequently, following the recommendations in reference (Heris 2015), $\gamma = 0.4$ will be used in this work.

Table 8 SMxc Sensitivity analysis for Mydata

γ	Measures	k_{min}	k_{max}	Average time (s)	N. solution in PF
0.2	SMxc-Mcc	2	12	80	95
	SMxc-MFNcc	2	12	100	72
	SMxc-MNNcc	2	11	80	93
0.4	SMxc-Mcc	2	12	70	91
	SMxc-MFNcc	2	12	80	78
	SMxc-MNNcc	3	13	80	97
0.6	SMxc-Mcc	2	12	90	90
	SMxc-MFNcc	2	11	60	89
	SMxc-MNNcc	2	12	80	90
0.8	SMxc-Mcc	2	10	90	98
	SMxc-MFNcc	2	10	90	97
	SMxc-MNNcc	2	10	70	98

Table 9 FNC sensitivity analysis for Mydata

γ	Measures	k_{min}	k_{max}	Average time (s)	N. Solution in PF
0.2	SMxc–Mcc	2	11	80	95
	SMxc–MFNcc	2	10	90	86
	SMxc–MNNcc	2	11	80	78
0.4	SMxc–Mcc	2	12	60	69
	SMxc–MFNcc	2	12	70	61
	SMxc–MNNcc	2	12	60	60
0.6	SMxc–Mcc	2	11	80	94
	SMxc–MFNcc	2	10	60	87
	SMxc–MNNcc	2	12	70	90
0.8	SMxc–Mcc	3	10	70	68
	SMxc–MFNcc	2	10	60	70
	SMxc–MNNcc	3	10	80	67

4.3 Results for dataset 2-Thyroid

To perform the MCA in a high-dimensional dataset the Thyroid dataset is used, which is composed of 215 elements and 5 features, and the results are presented in the next two sections. As the dataset 2 is composed of 215 elements, thus the parameters $k_{min} = 2$ and $k_{max} = 15$ are considered, as well as $\gamma = 0.4$, and 10 runs for each measure combination.

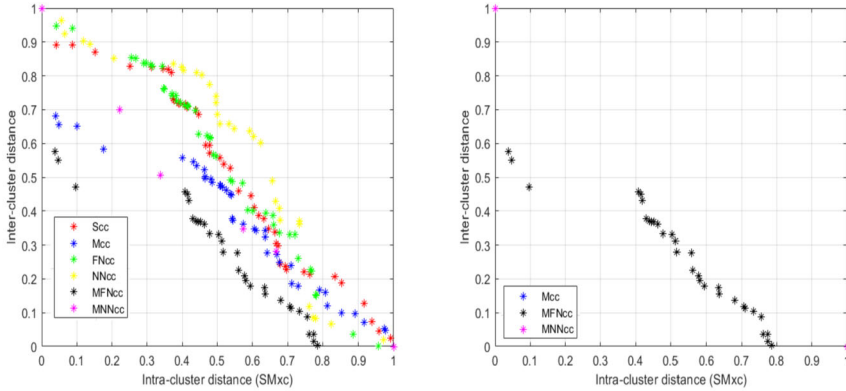
4.3.1 Results of the intra-clustering measure *SMxc* with inter-clustering measures

Figure 10a presents the normalized Pareto fronts for the *SMxc* measure combined with the *Mcc*, *FNcc*, *NNcc*, *MFNcc*, and, *MNNcc* inter-clustering measures. As it is possible to see, as the previous dataset, the solution of the normalized Pareto front provided by the means (represented by blue, black, and magenta colors) tends to dominate the solutions provided by the sum measures (represented by the red, green and yellow colors). To prove this observation the hypervolume of each Pareto front is evaluated and presented in Table 10. Therefore, the Pareto fronts provided by the measures *Mcc*, *MFNcc*, and *MNNcc* are the ones with the highest hypervolume value and could be considered the most appropriate to be used in this approach.

Figure 10b presents the HPF obtained. This HPF is composed of 176 solutions combined with the inter-clustering measures *Mcc*, *MFNcc*, and *MNNcc*.

Table 11 describes the solution that composed the HPF of Fig. 10b, the 176 solutions in terms of number of optimal clusters. Note that, there is a predominant presence of $k = 2$ solutions, which also corresponds to the literature indication of optimum solution.

As this dataset is one example of high-dimensional, it is not possible to illustrate the dataset distribution among the clusters.



(a) Normalized Pareto fronts for the $SMxc$ (b) Hybrid Pareto front of $SMxc$ and mean measures, for dataset 2

Fig. 10 Pareto fronts and hybrid Pareto front of $SMxc$ and means measures, for dataset 2 (Color figure online)

Table 10 Hypervolume values for the $SMxc$ combinations considering dataset 2

Measures	Scc	Mcc	$FNcc$	$NNcc$	$MFNcc$	$MNNcc$
Hypervolume	0.472	0.599	0.466	0.547	0.660	0.790

Table 11 Hybrid Pareto front solutions generated by the $SMxc$ measure combined with the 3 inter-clustering measures considering dataset 2

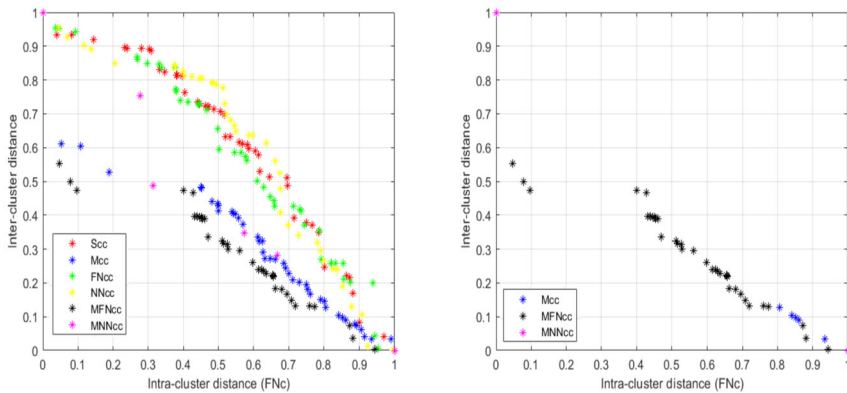
Measures/k	2	3	4	5	6	7	Total
$SMxc-Mcc$	30	0	0	4	0	8	42
$SMxc-MFNcc$	42	5	5	12	7	6	77
$SMxc-MNNcc$	57	0	0	0	0	0	57

4.3.2 Results of the intra-clustering measure FNc with inter-clustering measures

Figure 11a presents the normalized Pareto fronts for the intra-clustering measure FNc combined with the Mcc , $FNcc$, $NNcc$, $MFNcc$, and, $MNNcc$ inter-clustering measures. Again, the solution of the normalized Pareto fronts provided by the means (represented by blue, black, and magenta colors) tends to dominate the solutions provided by the sums measures (represented by the red, green, and yellow colors). Also, the Pareto fronts provided by the measures Mcc , $MFNcc$, and $MNNcc$ are the ones with the highest hypervolume value as presented in Table 12.

Table 12 Hypervolume values for the FNc combinations considering dataset 2

Measures	Scc	Mcc	$FNcc$	$NNcc$	$MFNcc$	$MNNcc$
Hypervolume	0.422	0.643	0.475	0.552	0.657	0.799



(a) Normalized Pareto fronts for FNC consid- (b) Hybrid Pareto front of FNC and mean ering dataset 2

Fig. 11 Pareto fronts and hybrid Pareto front of FNC and means measures, for dataset 2 (Color figure online)

Table 13 Hybrid Pareto front solutions generated by the FNC measure combined with the 3 inter-clustering measures considering dataset 2

Measures/k	2	3	4	5	6	7	Total
$FNC-Mcc$	38	5	0	2	0	6	51
$FNC-MFNcc$	63	1	0	1	3	0	68
$FNC-MNNcc$	12	0	0	0	0	0	12

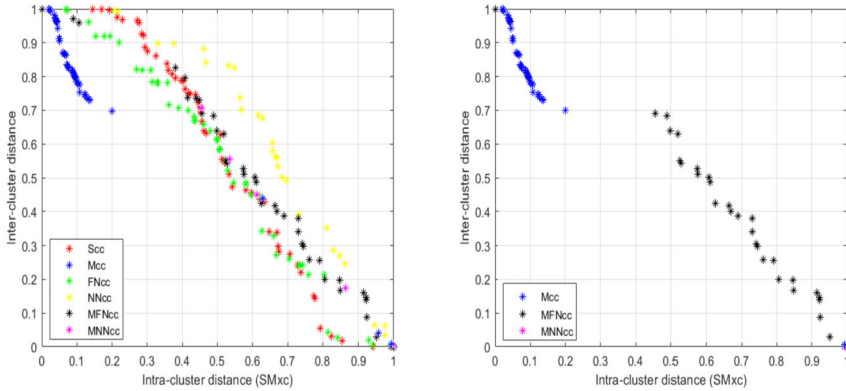
Figure 11b presents the HPF obtained. In this case, the HPF is composed of 131 solutions, provided by the intra-clustering measure FNC combined with the inter-clustering measures Mcc , $MFNcc$, and $MNNcc$, and described in Table 13.

4.4 Results for dataset 3–Breast dataset

The dataset 3 named Breast dataset is also evaluated by the MCA. This dataset is composed of 9 features, and provides another high-dimension dataset for the methodology evaluation. The dataset 3 is composed of 699 elements, resulting in the flowing MCA parameters: $k_{min} = 2$ and $k_{max} = 26$. It is also considered $\gamma = 0.4$ and 10 runs for each measure combination.

4.4.1 Results of the intra-clustering measure $SMxc$ with inter-clustering measures

The normalized Pareto fronts obtained with the $SMxc$ measure combined with the 6 inter-clustering measures are presented in Fig. 12a and the hypervolume of each Pareto front obtained is described in Table 14. Although the measure Mcc presents the lowest value compared to the others, the set of three Pareto fronts generated by the means stands out about the Pareto front generated by the sums measures.



(a) Normalized Pareto fronts for SM_{xc} con- (b) Hybrid Pareto front of SM_{xc} and mean sidering dataset 3 sidering dataset 3

Fig. 12 Hybrid Pareto front of SM_{xc} and means measures, for dataset 3 (Color figure online)

Table 14 Hypervolume values for the SM_{xc} combinations considering dataset 3

Measures	S_{cc}	M_{cc}	FN_{cc}	NN_{cc}	MFN_{cc}	MNN_{cc}
Hypervolume	0.916	0.900	0.914	0.916	0.938	0.971

Table 15 Pareto front solutions generated by SM_{xc} measure combined with the 3 inter-clustering measures considering dataset 3

Measures/k	2	3	4	5	6	7	8	9	Total
$SM_{xc}-M_{cc}$	22	13	7	4	8	9	4	5	72
$SM_{xc}-MFN_{cc}$	11	0	8	11	11	9	0	0	50
$SM_{xc}-MNN_{cc}$	53	0	2	0	0	0	0	0	55

Figure 12b presents the hybrid Pareto front considering only the nondominated solution of the three Pareto front generated by the means measures.

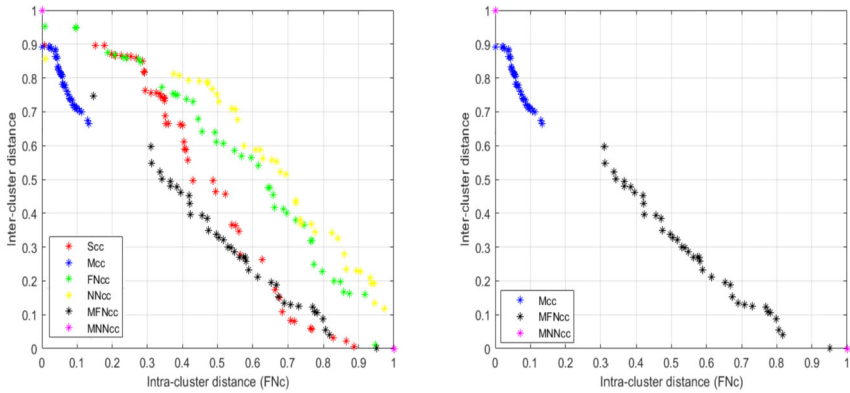
Analyzing the results of Table 15, the largest number of solutions with $k = 2$ is observed, since this is the optimal solution indicated in the literature.

4.4.2 Results of the intra-clustering measure FNc with inter-clustering measures

The normalized Pareto fronts obtained with the FNc measure combined with the 6 inter-clustering measures are presented in Fig. 13a and the hypervolume of each Pareto front obtained is described in Table 16. Again the measure Mcc presents the lowest value about the other, but the set of three Pareto fronts generated by the means stands out in relation to the Pareto front generated by the sums measures.

Figure 13b presents the hybrid Pareto front considering only the nondominated solutions of the three Pareto front generated by the means measures.

An analysis of the results in Table 17 shows that there is a greater number of solutions with $k = 2$, as occurs in the other dataset, indicating that the literature optimal solution is predominant in the HPF.



(a) Normalized Pareto fronts for FNC consid-(b) Hybrid Pareto front of FNC and mean ering dataset 3 measures, considering dataset 3

Fig. 13 Hybrid Pareto front of FNC and means measures, for dataset 3 (Color figure online)

Table 16 Hypervolume values for the FNC combinations considering dataset 3

Measures	Scc	Mcc	$FNCc$	$NNcc$	$MFNCc$	$MNNcc$
Hypervolume	0.923	0.905	0.918	0.922	0.933	0.971

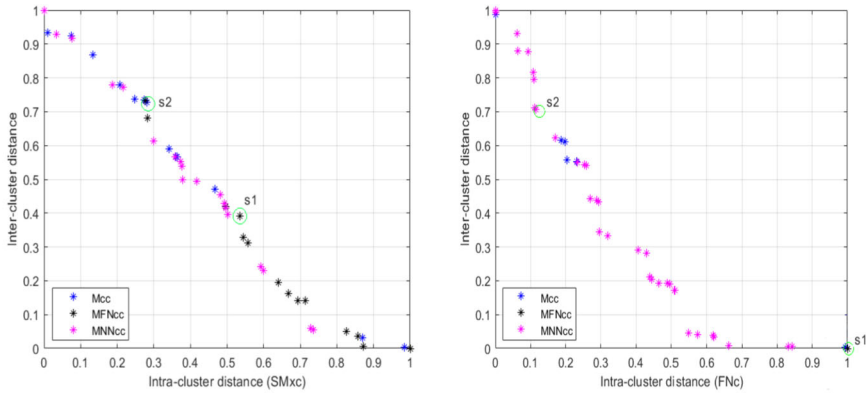
Table 17 Pareto front solutions generated by FNC measure combined with the 3 inter-clustering measures considering dataset 3

Measures/k	2	3	4	5	6	7	8	9	Total
$FNC-Mcc$	6	21	2	1	6	1	0	0	72
$FNC-MFNCc$	21	12	8	4	6	2	0	0	50
$FNC-MNNcc$	27	2	0	0	3	0	0	0	55

4.5 Results for dataset 4

To test the approach on a real case study, the methodology previously presented was applied on the dataset 4. Dataset 4 is composed of 291 elements and 2 variables, and it is provided by the MathE project (Azevedo et al. 2022; Flámia Azevedo et al. 2022). The MathE project aims to provide students all over the world with an online platform to help them learn college mathematics and also support students who want to deepen their knowledge on a multitude of mathematical topics at their own pace. More details about the MathE project are described in Azevedo et al. (2022, 2024a) and Flámia Azevedo et al. (2022), and can also be found on the platform Website (mathe.ipb.pt).

One of the particularities of the MathE platform is the *Student’s Assessment* section, which is composed of multiple-choice questions for the students to train and practice their skills. The answers given by each student over the 3 years the platform has been online define the dataset 4. Thus, each element of the dataset refers to one particular student using the Student’s Assessment section. Thereby, the first instance represents the rate of correct answers (x -axis) provided by the previous student’s test, and the second instance represents the number of



(a) Hybrid Pareto front of $SMxc$ and mean(b) Pareto front of FNc and means measures, for dataset 2 in dataset 2

Fig. 14 Hybrid Pareto front of dataset 4 (Color figure online)

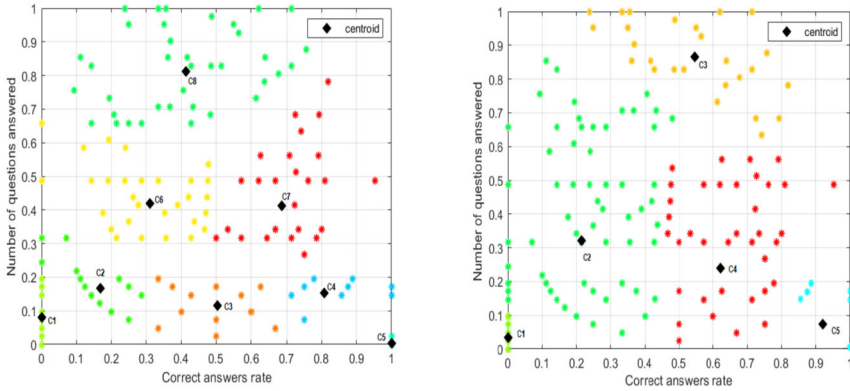
questions answered by this student (y-axis), while MathE user. To support the analysis of the results, the y-axis which initially ranges from 1 to 42 (number of questions answered) has been normalized by interval; it is between 0 and 1.

Preliminary studies using a single objective approach involving cluster categorization and MathE students' data, did not show satisfactory results in terms of the number of clusters and also in the division of the elements (Azevedo et al. 2022; Flámia Azevedo et al. 2022), i.e. the extracted patterns did not provide the necessary information to be used by the project. This occurs since the single objective procedure only provides a single solution, which, although feasible, is irrelevant to the decision-maker. For this reason, the dataset 4 is an excellent example to be analyzed with the MCA, since the choice of the optimal solution strongly depends on the sensitivity of the decision-maker and his/her prior information about the dataset.

Thus, the combination of intra- and inter-clustering measures previously defined are appropriate to identify the optimal partitioning of the dataset 4. Thereby, the intra-clustering measure $SMxc$ was combined 2-by-2 with the inter-clustering measures involving mean measures (Mcc , $MFNcc$, and $MNNcc$) since they were the ones that obtained the better results in the previous sections. After that, the results of each Pareto front obtained were analyzed in terms of dominance, resulting in a hybrid Pareto front that includes the results of the three mean measures, as shown in Fig. 14a. The same process was repeated for the intra-clustering measure FNc , and the results of the hybrid Pareto front are illustrated in Fig. 14b.

Knowing the profile of the students enrolled in the MathE platform, it is known that there is a diversity of students with different backgrounds (country, age, course and university year attending, and level of difficulty in Mathematics, among others). Therefore, a division into a few groups is not a significant result for the project, given the diversity of the public, especially in terms of performance in Mathematics subjects, as already explored in previous works.

From the Pareto front of the $SMxc$ measure (Fig. 14a), it is possible to select between 2 and 12 clustering divisions. Whereas in the Pareto front involving the FNc measure (Fig. 14b), the k varies from 2 to 13.



(a) Clustering partition of the chosen optimum solution at Figure 14a - Solution 1 ($k = 8$) (b) Clustering partition of the chosen optimum solution at Figure 14a - Solution 2 ($k = 5$)

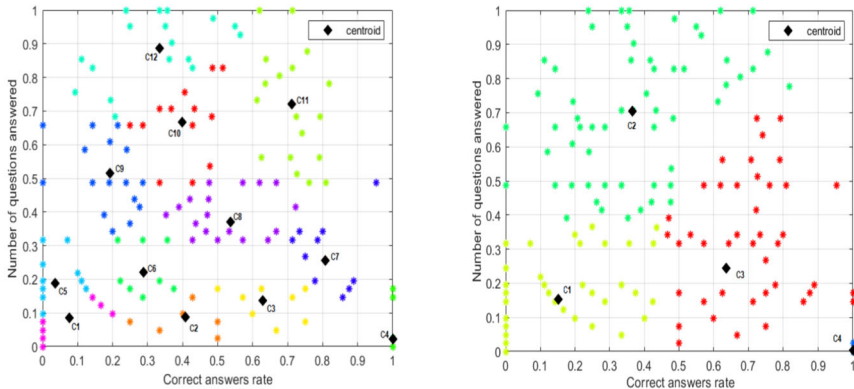
Fig. 15 Solution selected from Fig. 14a (Color figure online)

Considering this information, two solutions from each hybrid Pareto front were chosen to be better analyzed. Thus, Fig. 15 presents the result of the solutions chosen and highlighted in Fig. 14a, being Fig. 15a denoted as Solution 1 (s1), and Fig. 15b denoted as Solution 2 (s2).

The solution 1 illustrated in Fig. 15a divides the dataset into 8 clusters, and some interesting conclusions can be drawn from this division. Considering that the minimum value of y-axis is 1 and the maximum is 42, the y-axis values were normalized between 0 and 1. Analyzing the results, in terms of the number of questions answered, clusters 1 to 5 are composed of students who answer few questions, while clusters 6 and 7 comprise students who answer a larger number of questions than the aforementioned groups. Finally, cluster 8 is made up of students who answer the highest number of questions on the platform. In terms of performance (rate of correct answers), considering clusters 1 to 5, the performance of the students gradually increases for cluster 1 up to cluster 5, where in cluster 1 the students have a success rate equal to 0, (all answers given were incorrect), while in cluster 5 they had 1 (all answers given were correct). At this point, it is important to point out that dataset 4 is composed of multiple equal entries (student with equal number of questions answered and equal performance), which are superimposed in the figure. For this reason, cluster 5, although it seems to be composed of 1 student, actually includes 17 students, all with 1 answered question and 1 correct answer.

In clusters 6 and 7, the students use the platform more than the previous groups. In this case, the students in cluster 6 performed below 0.5, and those in cluster 7 were above 0.5. Finally, in cluster 8, the students answered the largest number of questions, and their performance varies between 0.1 and 0.8, so little can be said about overall performance of this group given the variability of solutions in terms of performance.

On the other hand, Fig. 15b shows the solution 2 results, indicating a solution composed of 5 clusters. In this case, each cluster is composed of much more students than the clusters presented in the solution of Fig. 15a. Thus, cluster 1 have a small number of students, that answered few questions, and all of their answers were incorrect. In cluster 2 there are a large number of students, some of than with many answers computed, other with few answers. All of these students had less than 0.5 of hit rate in their answers. On the other hand, in cluster 4, most of the students had more than 0.5 of hit rate in their answers. Cluster 3 is composed



(a) Clustering partition of the optimal solution chosen in Figure 14b ($k = 12$) Solution 1
 (b) Clustering partition of the chosen optimum solution at Figure 14b ($k = 4$) Solution 2

Fig. 16 Solution selected from Fig. 14b (Color figure online)

of the students who answered more questions, although some students of this cluster had a poor performance, most of them answered more than 0.5 of hit rate in their answers. Finally, in cluster 5, there is a group of cluster that had the best performance, but their answered few questions in relation to the other students.

In turn, Fig. 16 illustrates two solutions taken from the Pareto front of the FNc measure combination (Fig. 14b). Figure 16a presents the solution 1 (s1) composed by 12 clusters. Similar to the results of Fig. 15a in cluster 1 up to cluster 4, there are the students who answered few questions; while in cluster 5 up to cluster 7, there are the students who answered a few more questions than the previous groups; and in clusters 8 until cluster 12 there are the students who answered more questions. Thus, cluster 11 is composed of students who answered a large number of questions and have a performance higher than 0.6. In this case, the students answered correctly more than half of the questions they answered. In clusters 10 and 12, there are the students who answered many questions but got correct answers in less than half of the questions (in most cases). In terms of performance, the students in cluster 8 presented an average performance, between 0.35 and 0.75, with many answers. Finally, the performance of the students in clusters 1, 2, 5, and 6 is surpassed by the students in the clusters 3, 4, and 7.

On the other hand, Fig. 16b presents the solution 2 (s2), composed by 4 clusters, selected from Fig. 14b. In this solution it not possible to detailed the students characteristics, as done in the results of Fig. 16a, since the number of cluster is three times smaller and each cluster is composed of much more students. Thus, cluster 1 has the students, that answered few questions, and all of them had less than 0.5 in hit rate. Cluster 2 is composed of the students who answered more questions, although most of them had a poor performance, some students had more than 0.5 in their hit rate. In cluster 3, most of the students had more than 0.5 in their hit rate. While, in cluster 4, there is a group of students who had the best performance, but their answered few questions in relation to the others students.

Overall, the conclusions that can be drawn allow for a wide variety of details, which would be difficult to achieve if a solution with few clusters. Furthermore, this was only possible because the decision-maker had the power to choose and use prior knowledge about the data.

Table 18 DBSCAN number of centroids results

Dataset	$\epsilon = 1$ $minpoints = 5$	$\epsilon = 1$ $minpoints = 10$	$\epsilon = 0.5$ $minpoints = 5$	$\epsilon = 0.5$ $minpoints = 10$
1—My data	1 (5)	2 (10)	2 (27)	3 (58)
2—Thyroid	−1	−1	−1	−1
3—Breast	3 (12)	2 (23)	3 (11)	2 (34)
4—MathE	−1	−1	−1	−1

5 Algorithms comparison results

To compare the MCA results with the classical clustering algorithm of the literature, the four datasets considered were performed with two classical clustering algorithms, the k -means (Arthur and Vassilvitskii 2007) and DBSCAN algorithm (Ester et al. 1996), and also two hybrid approaches, the Clustering based on Differential Evolution Algorithm (CDE) (Heris 2015; Storn and Price 1997), and a novel approach named Game-based k -means (GBK-means) algorithm (Jahangoshai Rezaee et al. 2021).

The k -means, is one of the most well-known and simple clustering algorithms. It consists of trying to separate samples into groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squares (Arthur and Vassilvitskii 2007). The k -means is not an automatic clustering algorithm, thus, it depends on the initial estimation of the parameter k , which represents the number of clusters division. The four datasets were performed by the k -means with 10 executions, considering the k as indicated by the literature, it is $k = 3$ for dataset 1, $k = 2$ for dataset 2, $k = 2$ for dataset 3, and for the dataset 4 several values are possible, thus, it was considered $k = 4$ and $k = 8$, since this values were also presented previously in the MCA results. The k -means could work with all datasets returning satisfactory results.

In turn, the Density-based spatial clustering of applications with noise (DBSCAN), is a popular clustering algorithm used for clustering spatial data points based on their density distribution. DBSCAN is particularly effective in identifying clusters of arbitrary shapes and handling noise in the data. DBSCAN operates by grouping together points that are closely packed, forming high-density regions. It defines clusters as continuous regions of high density separated by regions of low density (Ester et al. 1996). This algorithm is based on a threshold for a neighborhood search radius ϵ and a minimum number of neighbors $minpoints$ required to identify a core point and define the clustering division (Ester et al. 1996).

Table 18 summarizes the results achieved with each algorithm considering different values of ϵ and $minpoints$. Note that the value -1 indicates that the DBSCAN considered all elements as outliers, it is the algorithm was not able to define the clusters. The number inside the parentheses indicates the number of elements considered outliers for each parametrization considered.

The datasets were also performed with two hybrid approaches. The Clustering based on Differential Evolution Algorithm (Heris 2015; Storn and Price 1997) is a bio-inspired metaheuristic approach that automatically defines the optimum number of cluster partitioning. The algorithm uses the Davies–Bouldin index (DB) (Davies and Bouldin 1979) as a clustering measure to define the number of clusters. Whereas the Game-based k -means (GBK-means) algorithm (Jahangoshai Rezaee et al. 2021) deals with the problem of clustering from a novel perspective, which considers the bargaining game approach in the k -means algorithm for

clustering data. It enables competition between cluster centers to attract the largest number of similar entities to its cluster. In the approach, the centroids change their positions in such a way that they have minimum distances with the maximum possible entities compared to the other centroids. Thereby objective function combines the clustering distance and bargaining game measures and uses a PSO algorithm to find the optimal solution. But, the GBK-means require a priori amount of cluster information. More details about the approach can be seen at Jahangoshai Rezaee et al. (2021).

Figure 17a presents the four algorithms results for the dataset 1. The DBSCAN solution provided by $\epsilon = 0.5$ and $minpoints = 10$ is presented in Fig. 17b, but 58 were considered outliers and were removed from the dataset. This last solution was chosen since it is the closest solution to the one suggested by the literature (Heris 2015). Whereas Fig. 17c, d present the hybrid algorithm results, the CDE and the GBK-means, respectively.

Comparing the results of Fig. 17, the k -means and CDE presented similar results, which is identical to the solution found by the MCA, presented in Fig. 7b, that correspond to the literature solution. Although the DBSCAN and GBK-means algorithms could divide the dataset into 3 clusters, they do not present satisfactory results in the dataset 1, compared to the other algorithms.

About dataset 4, Fig. 18 presents the k -means (Fig. 18a, b), CDE (Fig. 18c, d), and the GBK-means (Fig. 18e, f) algorithms results considering 4 and 8 clusters.

The results k -means and CDE results are similar to the ones presented by the MCA and discussed in previous sections. The GBK-means presented other alternative results that could be considered by the decision-maker, but the MCA, k -means, and CDE results outstanding the GBK-means results in this dataset.

About the dataset 2, the k -means, CDE and GBK-means could be able to propose a clustering division, while in the DBSCAN all elements were considered outliers. And, regarding dataset 3, all algorithms considered could identify the clustering divisions.

6 Conclusions and future work

The advantage of using multi-objective strategies in the clustering task is to combine multiple objectives simultaneously. The MCA approach makes substantial advancements by employing a multi-objective strategy and combining diverse solutions to determine the optimal number of cluster sets and their element partitioning.

Unlike single-objective algorithms that minimize one measure at a time, the proposed approach overcomes limitations through a multi-objective approach, offering a set of optimal solutions. This characteristic empowers decision-makers to choose the most fitting solution according to their expertise or preferences.

In this way, this paper explored several clustering measures, namely intra- and inter-clustering measures, to develop a clustering algorithm that automatically defines the optimal number of cluster partitions and consequently classifies the elements of the dataset according to their similarities and dissimilarities. For this, 42 combinations of 6 intra-clustering and 7 inter-clustering measures were analyzed, with the aim of defining the most appropriate pair of measures to be used in a multi-objective approach. Thereby, 2 intra-clustering measures ($SMxc$ and FNc) and 6 inter-clustering measures (Sc , Mcc , $FNcc$, $NNcc$, $MFNcc$, and $MNNcc$) were selected. After that, each generated Pareto front was analyzed in terms of dominance and by the hypervolume method, defining that the 3 mean-based measures (Mcc , $MFNcc$, and $MNNcc$) are most appropriate to be used as inter-clustering measures. The

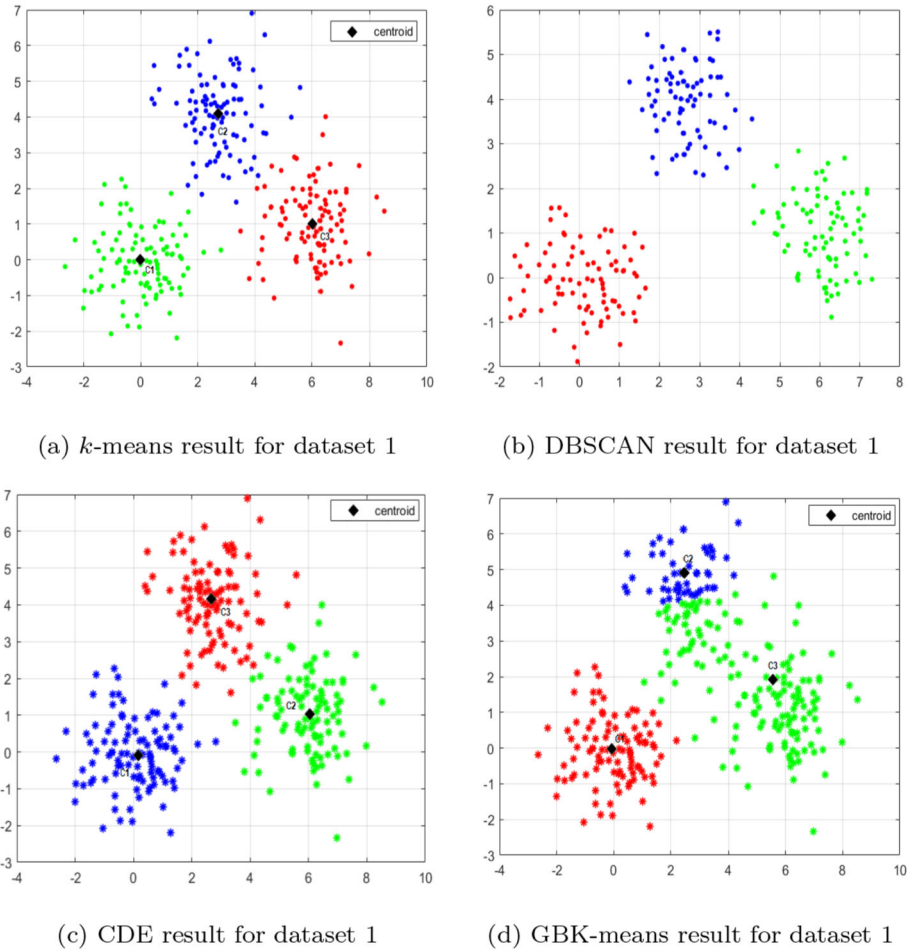


Fig. 17 Dataset 1 results for k -means, DBSCAN, CDE and GBK-means algorithms (Color figure online)

solutions of three Pareto fronts were confronted in terms of dominance to create a hybrid Pareto front composed of the nondominated solutions, considering different pairs of measure combinations. This methodology was tested on a benchmark dataset and also on a dataset from a real case study that describe the student’s performance in the MathE project.

From the range and variability of each Pareto front generated, it is possible to perceive the impact of combining different measures to solve a problem. In Tables 4 and 7, for example regarding dataset 1, the combination $SMxc - MFNcc$ and $FNc - MFNcc$ have no solutions with k equal to 3. Whereas the combination $SMxc - MNNcc$ has solutions with $k = 4$, but no solution with $k = 2$. In this way, if only one Pareto front was considered, the solution would be restricted to the optimum provided by one combination of pairs of measures and could be inappropriate for the decision-maker. So, combining the solutions of the Pareto fronts provided by different measures enriches the final solution.

A clustering algorithm should be able to define the optimal number of partitions and also the optimal distribution of the elements. The fact that the MCA indicates multiple solutions

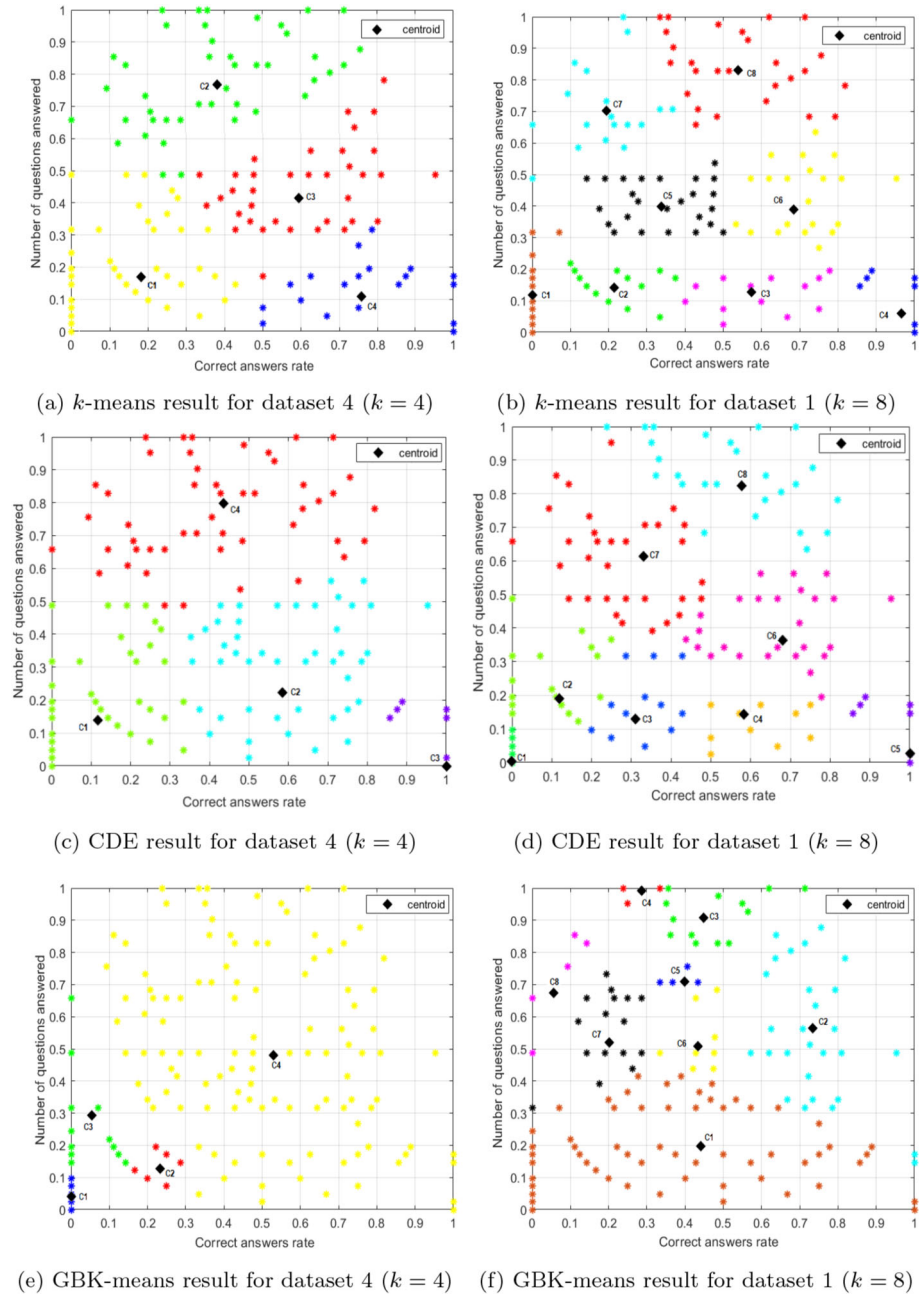


Fig. 18 Dataset 4 results for k -means, CDE, and GBK-means algorithms (Color figure online)

with equal k values does not mean that these solutions are all equal since the elements can be distributed over different clusters. Often the value of the intra- and inter-clustering measures is not relevant to the decision-maker, so the origin of the solution, i.e. the measures that were used to generate it, is not a decisive factor in the choice of the solution, being indifferent whether the solution was originated by 'a' or 'b' measures, so there is no penalty in presenting the results through a hybrid Pareto front. These results justify the proposed method of producing a hybrid Pareto front by considering solutions from different Pareto fronts. If the Pareto fronts are considered individually, the results may be different, so putting them together provides a robust result in terms of variability for the decision-maker.

According to Heris (2015), considering a single objective strategy, the optimal solution for dataset 1 is $k = 3$, i.e. 3 clusters. In Table 4, which presented the solution of the hybrid Pareto front, $k = 3$ is the most frequent solution for dataset 1, demonstrating the effectiveness of the proposed method on a benchmark problem.

Datasets 2 and 3 are some examples of high-dimensional datasets and the MCA could provide satisfactory solutions, as well as a range of possibilities in terms of the number of clusters and dataset division, whereas the other algorithms were not able to work properly with the dataset. Furthermore, the DBSCAN requires a previous analysis of the parameters ϵ and *minpoints* to achieve the optimum solution, and the k -means and GBK-means require the indication of the number of centroids.

In the case of dataset 4, the distribution of the data is more complex, since there are multiple elements overlapped, and the elements are not as well separated as the first dataset (Dutta et al. 2019). Thus, for dataset 4, which describes a real problem, the multi-objective strategy is much more effective than the single one, since in the multi-objective it is possible to compare and choose among a set of optimal solutions, the one that meets the patterns that the decision-maker wants to extract from the dataset.

Moreover, for the dataset 4, the knowledge of the decision-maker is of great value in defining the best solution. Then, the proposed method is an asset in situations where the single objective approach is not enough, as occurs in Azevedo et al. (2022), Flámia Azevedo et al. (2022) for dataset 4. Although the single clustering algorithm presented a feasible solution, this one is not good enough to extract relevant information for the MathE project.

The MCA results were compared with well-known methods such as k -means and DBSCAN and two hybrid approaches, the CDE and the GBK-means. It was possible to conclude that the k -means, DBSCAN, and GBK-means have some limitations, showing the advantage of the MCA. Although MCA is more computationally demanding, it offers a set of highly variable optimal solutions in terms of the number of clusters and the division of elements between them. This occurs because MCA is able to explore the solution search space more broadly, combining several measures in its methodology.

Although k -means and CDE algorithms work well on diverse datasets, they only provide a single solution. And k -means requires the decision-maker to determine a fixed number of clusters. If the decision-maker needs a new solution with a different number of clusters, or if the solution obtained does not meet his requirements, it will be necessary to run the algorithm again until a suitable solution is found. Whereas the MCA provides all these options at once.

In relation to DBSCAN, which is an algorithm that works well in complex geometries, it tends to have difficulty in high-dimensional sets, and in addition, DBSCAN is very dependent on the ϵ and the *minpoints* parameters, requiring running the code with different parameters until finding the most suitable solution. In the sets presented in this work, DBSCAN failed several times to identify clusters and ended up categorizing the data as outliers.

In the future, to improve the MCA performance, it is expected to explore split and merge strategies considering the measures studied in this paper. The pair of measures that pro-

more larger partitions could especially be very useful for proposing initial splits in clustering algorithms. Furthermore, it is expected to develop a strategy based on clustering validated indices to assist the decision-maker in the choice of the optimal solution, especially in situations where the prior information about the data is scarce.

Acknowledgements I would like to extend my heartfelt gratitude to the peer reviewers for their kind appreciation of my scholarly and practical work.

Author contribution AB The author of the manuscript confirms that he is the founder of the conceptual idea contribution.

Funding Open access funding provided by FCTIFCCN (b-on).

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aggarwal CC, Reddy CK (2013) Data clustering algorithms and applications. CRC Press, Taylor & Francis Group, Boca Raton
- Ahmadi B (2023) C-index, spacing, and hypervolume. https://www.mathworks.com/matlabcentral/fileexchange/125980-c-index-spacing-and-hypervolume?s_tid=prof-contriblnk
- Arthur D, Vassilvitskii S (2007) K-means++: the advantages of careful seeding. In: Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms, SODA '07. Society for Industrial and Applied Mathematics, pp 1027–1035. <https://doi.org/10.1145/1283383.1283494>
- Azevedo BF, Romanenko SF, de Fatima Pacheco M, Fernandes FP, Pereira AI (2022) Data analysis techniques applied to the mathe database. In: Pereira AI, Košir A, Fernandes FP, Pacheco MF, Teixeira JP, Lopes RP (eds) Optimization, learning algorithms and applications. Lecture Notes in Computer Science, vol 13621. Springer, Cham, pp 623–639. https://doi.org/10.1007/978-3-031-23236-7_43
- Azevedo BF, Montañño-Vega R, Varela M, Pereira A (2023) Bio-inspired multi-objective algorithms applied on production scheduling problems. *Int J Ind Eng Comput* 14(2):415–436. <https://doi.org/10.5267/j.ijec.2022.12.001>
- Azevedo BF, Pacheco MF, Fernandes FP, Pereira AI (2024) Dataset of mathematics learning and assessment of higher education students using the MathE platform. *Data in Brief* 53:110236. <https://doi.org/10.1016/j.dib.2024.110236>
- Azevedo BF, Rocha AMAC, Pereira AI (2024) Hybrid approaches to optimization and machine learning methods: a systematic literature review. *J Mach Learn*. <https://doi.org/10.1007/s10994-023-06467-x>
- Behera M, Sarangi A, Mishra D, Mallick PK, Shafi J, Srinivasu PN, Ijaz MF (2022) Automatic data clustering by hybrid enhanced firefly and particle swarm optimization algorithms. *Mathematics*. <https://doi.org/10.3390/math10193532>
- Bi X, Hu X, Wu H, Wang Y (2020) Multimodal data analysis of Alzheimer's disease based on clustering evolutionary random forest. *IEEE J Biomed Health Inf* 24(10):2973–2983. <https://doi.org/10.1109/JBHI.2020.2973324>

- Binu Jose A, Das P (2022) A multi-objective approach for inter-cluster and intra-cluster distance analysis for numeric data. In: Kumar R, Ahn CW, Sharma TK, Verma OP, Agarwal A (eds) *Soft computing: theories and applications*. Springer Nature Singapore, Singapore, pp 319–332
- Bringmann K, Friedrich T (2013) Approximation quality of the hypervolume indicator. *Artif Intell* 195:265–290. <https://doi.org/10.1016/j.artint.2012.09.005>
- Coello-Coello CA, Lechuga MS (2002) Mopso: a proposal for multiple objective particle swarm optimization. In: *Proceedings of the 2002 congress on evolutionary computation*. CEC'02 (Cat. No.02TH8600), vol 2, pp 1051–1056. <https://doi.org/10.1109/CEC.2002.1004388>
- Davies DL, Bouldin DW (1979) A cluster separation measure. *IEEE Trans Pattern Anal Mach Intell* 1(2):224–227. <https://doi.org/10.1109/TPAMI.1979.4766909>
- Deb K (2001) *Multi-objective optimization using evolutionary algorithms*. Wiley, New York
- Deb K (2011) *Multi-objective optimization using evolutionary algorithms: An introduction*. In: Wang L, Ng AHC, Deb K (eds) *Multi-objective evolutionary optimisation for product design and Manufacturing*, 1st edn. Springer, London
- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197. <https://doi.org/10.1109/4235.996017>
- Dutta D, Sil J, Dutta P (2019) Automatic clustering by multi-objective genetic algorithm with numeric and categorical features. *Expert Syst Appl* 137:357–379. <https://doi.org/10.1016/j.eswa.2019.06.056>
- Eesa AS, Orman Z (2020) A new clustering method based on the bio-inspired cuttlefish optimization algorithm. *Expert Syst*. <https://doi.org/10.1111/exsy.12478>
- Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the second international conference on knowledge discovery in databases and data mining*. AAAI Press, Portland, pp 226–231
- Flamia Azevedo B, Rocha AMAC, Fernandes FP, Pacheco MF, Pereira AI (2022) Evaluating student behaviour on the mathe platform-clustering algorithms approaches. In: Simos DE, Rasskazova VA, Archetti F, Kotsireas IS, Pardalos PM (eds) *Learning and intelligent optimization*. Lecture Notes in Computer Science, vol 13621. Springer, Cham, pp 319–333. https://doi.org/10.1007/978-3-031-24866-5_24
- Fränti P, Sieranoja S (2018) K-means properties on six clustering benchmark datasets. <http://cs.uef.fi/sipu/datasets/>
- Heris MK (2015) Evolutionary data clustering in matlab. <https://yarpiz.com/64/yplm101-evolutionary-clustering>
- Ikotun AM, Ezugwu AE (2022) Boosting k-means clustering with symbiotic organisms search for automatic clustering problems. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0272861>
- SAS-Institute. The cluster procedure: Clustering methods. https://documentation.sas.com/doc/en/pgmsascdc/9.4_3.4/statug/statug_cluster_overview.htm
- Jahangoshai Rezaee M, Eshkevari M, Saberi M, Hussain O (2021) Gbk-means clustering algorithm: an improvement to the k-means algorithm based on the bargaining game. *Knowled-Based Syst* 213:106672. <https://doi.org/10.1016/j.knosys.2020.106672>
- Kaur A, Kumar Y (2022) A multi-objective vibrating particle system algorithm for data clustering. *Pattern Anal Appl* 25(1):209–239. <https://doi.org/10.1007/s10044-021-01052-1>
- Kok J, González FC, Kelson N, Périaux J (2011) An FPGA-based approach to multi-objective evolutionary algorithm for multi-disciplinary design optimisation
- Liu X, Liu Q (2024) Optimized diagnosis of local anomalies in charge and discharge of solar cell capacitors. *Energy Inf*. <https://doi.org/10.1186/s42162-024-00329-z>
- Liu C, Liu J, Peng D, Wu C (2018) A general multiobjective clustering approach based on multiple distance measures. *IEEE Access* 6:41706–41719. <https://doi.org/10.1109/ACCESS.2018.2860791>
- MATLAB (2019) The mathworks inc. <https://www.mathworks.com/products/matlab.html>
- Memarsadeghi N, Mount D, Netanyahu N, Moigne J (2007) A fast implementation of the isodata clustering algorithm. *Int J Comput. Geom Appl* 17:71–103. <https://doi.org/10.1142/S0218195907002252>
- Mirjalili S, Saremi S, Mirjalili SM, Coelho LS (2016) Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. *Expert Syst Appl* 47:106–119. <https://doi.org/10.1016/j.eswa.2015.10.039>
- Morimoto CY, Pozo ATR, de Souto MCP (2021) A survey of evolutionary multi-objective clustering approaches
- Müller AC, Guido S (2016) *Introduction to machine learning with Python: a guide for data scientists*. O'Reilly Media, Sebastopol
- Pal N, Bezdek J (1995) On cluster validity for the fuzzy c-means model. *IEEE Trans Fuzzy Syst* 3(3):370–379. <https://doi.org/10.1109/91.413225>

- Qaddoura R, Faris H, Aljarah I (2021) An efficient evolutionary algorithm with a nearest neighbor search technique for clustering analysis. *J Ambient Intell Human Comput* 12:8387–8412. <https://doi.org/10.1007/s12652-020-02570-2>
- Shi X, Yue C, Quan M, Li Y, Nashwan Sam H (2025) A semi-supervised ensemble clustering algorithm for discovering relationships between different diseases by extracting cell-to-cell biological communications. *J Cancer Res Clin Oncol*. <https://doi.org/10.1007/s00432-023-05559-4>
- Sokal RR, Michener CD (1958) A statistical method for evaluating systematic relationships. *Univ Kans Sci Bull* 38:1409–1438
- Sorensen TA (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol Skar* 5:1–34
- Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11(4):341–359. <https://doi.org/10.1023/A:1008202821328>
- Tambunan HB, Barus DH, Hartono J, Alam AS, Nugraha DA, Usman HHH (2020) Electrical peak load clustering analysis using k-means algorithm and silhouette coefficient. In: 2020 International Conference on Technology and Policy in Energy and Electric Power (ICT-PEP), pp 258–262. <https://doi.org/10.1109/ICT-PEP50916.2020.9249773>
- Villa C, Labayrade R (2011) Energy efficiency vs subjective comfort: a multiobjective optimisation method under uncertainty. <https://api.semanticscholar.org/CorpusID:102329612>
- Wadhwa A, Garg S, Thakur, MK (2023) Automatic detection of DBSCAN parameters using BAT algorithm, pp 530–536. <https://doi.org/10.1145/3607947.3608058>
- Wang Y, Luo S, Fan J, Zhen L (2024) The multidepot vehicle routing problem with intelligent recycling prices and transportation resource sharing. *Transp Res Part E Logist Transp Rev* 185:103503. <https://doi.org/10.1016/j.tre.2024.103503>
- Yang XS, Gen M (2010) Introduction to evolutionary algorithms. Springer, Berlin. <https://doi.org/10.1007/978-1-84996-129-5>
- Zhao F, Tang Z, Xiao Z, Liu H, Fan J, Li L (2024) Ensemble cart surrogate-assisted automatic multi-objective rough fuzzy clustering algorithm for unsupervised image segmentation. *Eng Appl Artif Intell*. <https://doi.org/10.1016/j.engappai.2024.108104>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.