



The use of conventional clustering methods combined with SOM to increase the efficiency

Martin Kotyrba¹ · Eva Volna¹ · Robert Jarusek¹ · Pavel Smolka¹

Received: 31 August 2020 / Accepted: 17 June 2021 / Published online: 24 June 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

This article reflects research in the field of artificial intelligence and demonstrates a higher efficiency achievement of conventional clustering methods in combination with unconventional methods. It concerns a new hybrid approach based on the SOM (Self-Organizing Maps) method. We focused on the possibility of combining SOM with other clustering methods—CLARA, CURE a K-means. Method SOM is primarily useful in the first phases of the process, where knowledge of the data is too vague. It is thus followed by the use of a selected clustering algorithm. It then works with preprocessed data. Its performance, compared with its outputs on unprocessed data, is more efficient, which is proved by the performed experimental study on the benchmark data set Fundamental Clustering Problems Suite (FCPS). Part of the experimental verification was also a comparison of the achieved outputs with other approaches using this dataset based on a standard metrics—Rand index.

Keywords Artificial intelligence · Clustering · SOM · Rand index

Abbreviations

α_s	Shrinking factor (CURE), learning parameter (SOM)	<i>DBSCAN</i>	Density-Based Spatial Clustering of Applications with Noise
ε	Radius	<i>DENCLUE</i>	DENSity-based CLUstEring
μ	Learning parameter	<i>EFCM</i>	Extended Fuzzy C-Means
ρ	Surroundings of the winning neuron (SOM)	<i>FCPS</i>	Fundamental Clustering Problems Suite
$D(j)$	Euclidean distance	<i>MinPts</i>	Minimum number of other objects
<i>BIRCH</i>	Balanced Iterative Reducing and Clustering using Hierarchies	<i>MLP</i>	Multilayer perceptron
<i>CURE</i>	Clustering Using REpresentatives	<i>OPTICS</i>	Ordering points to identify the clustering structure
<i>CLARA</i>	Clustering LARge Applications	<i>PAM</i>	Partitioning Around Medoids
<i>CLARANS</i>	Clustering Large Applications based on RANdomized Search	<i>SEEFC</i>	Self-organizing-map based extended fuzzy c-means
<i>CLIQUE</i>	CLustering In QUEst	<i>SOM</i>	Self-Organizing Maps
		<i>STING</i>	STatistical INformation Grid

✉ Eva Volna
eva.volna@osu.cz

Martin Kotyrba
martin.kotyrba@osu.cz

Robert Jarusek
robert.jarusek@osu.cz

Pavel Smolka
pavel.smolka@osu.cz

¹ Department of Informatics and Computers, University of Ostrava, 30. dubna 22, 70103 Ostrava, Czech Republic

1 Clustering algorithms

A clustering analysis is a data analysis tool sorting various objects into clusters in a way that similarity of two objects belonging to one group is maximal whereas similarity with objects outside this cluster is minimal [14]. The difference between clustering and classification is that when classifying, objects are sorted into classes known before whereas when clustering, the classes are not known before, but they

are the result of clustering. In most cases, the basic way of data representation for clustering is the use of $n \times p$ data matrix X . This matrix contains n objects, where matrix rows represent individual objects and p columns represent their properties.

Clustering methods can also be divided into various groups according to various criteria. The most commonly used division is according to the resulting cluster structure, namely *hierarchical* and *non-hierarchical*. The results in the hierarchical methods are clusters sorted in a hierarchical structure. Each cluster is represented by one hierarchical tree. Individual tree nodes represent clusters. Hierarchical methods can also be divided according to the approach of cluster creation—*agglomerative* and *divisive*. The agglomerative approach stems from individual objects which gradually form clusters until all objects are in a single cluster. The divisive approach takes a set as one unit which then forms a hierarchical subsets system by gradual dividing the objects. Apart from this basic division, there are numerous other groups of clustering methods, which can be divided into the following categories based on their clustering approach. Classification of individual methods into these categories is not completely strict, it often varies according to the author, some methods can be classified into more categories, respectively. The most frequent classification is as follows [11]:

- *Partitioning methods* Construct various partitions and then evaluate them by some criterion.
- *Hierarchical methods* Create a hierarchical decomposition of the set of data using some criterion.
- *Density-based methods* They are based on connectivity and density functions.
- *Grid-based methods* They are based on a multiple-level granularity structure.
- *Model-based methods* A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other.
- *Others*.

Table 1 presents an overview of the most frequent clustering algorithms. Methods which we will further deal with are in bold. They have been selected so that they belong to different categories of clustering methods.

1.1 DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a data clustering algorithm proposed in 1996 [10]. It is a typical representative of clustering according to density. Density-based clustering take grounds from the definition of a cluster as an area with a higher spatial density of points compared with other areas. It holds that the surroundings of an object in a cluster must

locate a minimum number of other objects (*MinPts*) in the distance defined by radius ε . The advantages of density-based methods include noise recognition of any shape and robustness to outlying values.

The DBSCAN algorithm works in the following steps:

1. Select a point from the observed set.
2. Find all points within reach of the selected point. If the distance of *MinPts* points from the selected point is smaller than ε , such a point is a *core point* and create a new cluster around it.
3. Find all objects directly reachable from the *core point*. Clusters can be merged.
4. Finish if no other object can be added to any cluster, otherwise continue with step 1.

Both output ε and *MinPts* parameters are crucial for cluster creation as their selection significantly affects the clustering result.

1.2 K-means

K-means is a non-hierarchical clustering algorithm proposed in 1967 [22]. This algorithm assumes that clustered objects can be understood as points in a certain Euclidean space and the number of clusters k is given in advance. The number of clusters influences a random selection of initial cluster centers—centroids, which are points in the same space as clustered objects. Objects are sorted to a cluster whose centroid they are the closest. In the following step, a new centroid is defined based on mean values of objects belonging to it. The algorithm is repeated until the position of centroids stabilizes.

The K-means algorithm works in the following steps:

1. Choose the number of clusters k , generate or set cluster centroids.
2. Assign each object to a cluster with the smallest Euclidean distance to the centroid. If the selected cluster does not equal to the initial cluster, relocate there the current object and recalculate the modified centroid.
3. Finish if no cluster has been changed, otherwise continue with step 2.

A limitation of the K-means methods is the possibility of working with metric data and the possibility of the existence of more solutions based on the initial object layout.

1.3 CLARA

Method CLARA (Clustering LARge Applications) [17] is based on the use of sampling and method PAM (Partitioning Around Medoids), which is one of numerous modifications of the K-means method. Unlike K-means,

Table 1 Overview of the selected most frequent clustering algorithms [19, 28]

Category	Methods	Cluster shape	Large datasets	Sensitivity to object ordering	Sensitivity to outliers
Partitioning methods	K-means	Convex	Y	Big	Big
	PAM	Convex	N	Medium	Small
	CLARA	Convex	Y	Big	Small
	CLARANS	Convex	Y	Big	Small
Hierarchical methods	BIRCH	Convex	Y	Medium	Small
	CURE	Arbitrary	Y	Medium	Small
	ROCK	Arbitrary	N	Medium	Small
	CHAMELEON	Arbitrary	N	Medium	Small
Density-based methods	DBSCAN	Arbitrary	Y	Medium	Small
	OPTICS	Arbitrary	Y	Small	Small
	DENCLUE	Arbitrary	Y	Medium	Small
Grid-based methods	STING	Arbitrary	Y	Small	Small
	CLIQUE	Convex	N	Small	Medium
Model-based methods	SOM	Arbitrary	Y	Medium	Small

PAM consists in substituting a centroid for a medoid, which is a value dividing a series of ascending order results into the same halves. Another improvement is the possibility of using arbitrary metrics.

The CLARA method is modified in a way to process a larger data amount compared with PAM. When clustering with CLARA, a random sample from the processed data is selected. The sample is then clustered into k clusters. The rest of the objects are then put into those clusters. Sampling is based on a random choice of a small number of objects which are then a subject to the application of the PAM method. Then, objects which are not part of the sample are assigned. This is performed several times and the best result is then selected.

Both input parameters, *Number of samples* and *Size of samples*, are crucial for clustering as their selection influences the clustering result.

1.4 CURE

The essence of the CURE (Clustering Using REpresentatives) algorithm consists in the fact that each cluster has a finite number of representatives. First, a random selection of objects is performed; second, they are divided into fractions. Each fraction is subjected to a hierarchical cluster analysis. Then, outliers are identified and based on the formed auxiliary clusters, a required number of final clusters is created [33]. Algorithm CURE is an effective

algorithm for large datasets.

Algorithm CURE works in the following steps.

1. Draw random sample s .
2. Partition sample to p partitions with size s/p .
3. Partially cluster partitions into s/pq clusters
4. Eliminate outliers:
 - By random sampling.
 - If a cluster grows too slow, eliminate it.
5. Cluster partial clusters.
6. Label data in disk.

Both input parameters, *shrinking factor* $\alpha \in \langle 0,1$ and *Number of repetitions*, are crucial for the creation of clusters as their selection affects the clustering result.

2 Self-organizing maps

Self-Organizing Maps (SOM) belong to a group of self-learning neural networks with a teacher. [18]. It concerns a two-layer network. The number of neurons contained in the input layer determines the dimension of the input data. The second layer, i.e., the output layer, is organized into a

certain topological structure—most commonly into a two-dimension grid with neurons laid into a square or hexagonal structure. A self-learning algorithm stems from the strategy of competitive learning. When learning, we gradually show the network individual training patterns. Having shown a training pattern, the neurons compete, and the winning neuron is that whose Euclidean distance from the shown pattern is the smallest. The weights of the winning neuron are then adjusted to be as close as possible to the pattern. The degree of weight adjustment is given by learning parameter $\mu \in (0,1)$, Other neurons' weights are, in addition, adjusted by their membership to the surroundings of the winning neuron, which is determined by the radius ρ .

3 Literature overview

This literature research stems from the focus of this article and deals with hybrid approaches to clustering using the SOM method.

In [34], the authors proposed a new approach to clustering and visualization of students' cognitive structural models. They used SOM in combination with Ward's clustering to conduct cluster analysis. The authors [16] presented a new algorithm where they use SOM to achieve rough clusters and center of clusters, while they use the K-means method to finish the clustering. Work [32] deals with the classification of objects moving in a video into three classes: pedestrians, bicycles, and vehicles. The cre-

Algorithm SOM works in the following steps.

1. Initialization of the learning parameter α , surroundings ρ and initial values of network weights w_{ij} .
2. Showing a learning pattern $x = (x_1, x_2, \dots, x_n)$.
3. For each output neuron j , calculate Euclidean distance $D(j)$ from the learning pattern (1).

$$D(j) = \sum_i (w_{ij} - x_j)^2 \quad (1)$$

4. Find neuron j^* , whose (j^*) is the smallest.
5. Update the weights of all neurons in the neuron j^* surroundings according to the Kohonen rule (2):

$$w_{ij}(\text{new}) = w_{ij}(\text{old}) + \alpha [x_i - w_{ij}(\text{old})] \quad (2)$$

6. Jump to Step 2 until all learning patterns are shown.
7. Update the learning parameter α and radius ρ .
8. A terminating condition if not fulfilled, repeat from Step 2.

2.1 Ways of representing SOM results

- *U-Matrix* is created from a learned self-organizing map and shows the Euclidean distance of a neuron from its directly neighboring neurons using coloring in a 2D map or as a 3D map. Using U-Matrix, we are able to divide a self-organizing map into areas corresponding to related input data and to set their boundaries clearly.
- *Activation map* depicts the frequency of neuron activation on the output map. It is usually in grayscale while the most frequently activated neuron is depicted in black and the least activated in white.

ated feature vectors are sent to SOM, whose output are three cluster centers initializing the cluster centers of K-means. Method K-means then terminates the clustering. The authors in [31] presented an adaptive approach which uses the Kohonen Self-Organizing Map, extended with online K-means clustering, for classification of real-time input sensor data for further processing. Article [9] presented a new clustering algorithm SOM + +, which combines methods K-means and SOM. Method K-means is used here to initialize the weight values on neuron synapses in the SOM method. The authors stated that the proposed approach SOM + + brought better results than SOM in all criteria. The authors in [1] proposed and experimentally verified a new hybrid algorithm for image segmentation based on SOM and Extended Fuzzy C-Means (EFCM) named self-organizing-map based extended fuzzy c-means

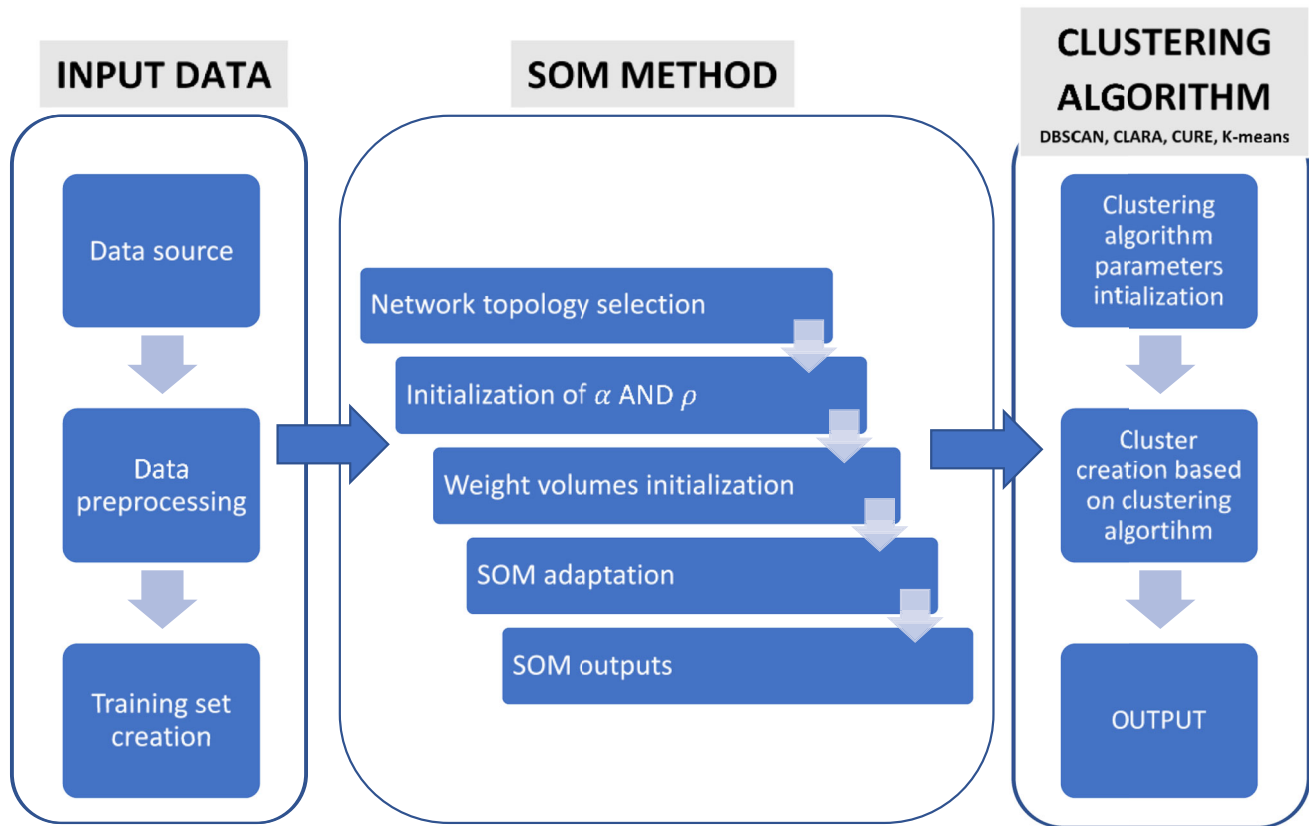


Fig. 1 Proposed methodology

(SEEFEC). The authors in [12] proposed a new hybrid approach based on a combination of self-organizing map and hierarchical clustering, while they discovered genes displaying expression regulation during characteristic stages of *M. truncatula* flower and pod development. Work [6] presented a new hybrid method of SOM and DBSCAN called SOM-DBSCAN for image segmentation. Work [27] dealt with a breast-cancer data analysis for survivability studies and prediction. The authors proposed a hybrid approach based on SOM, DBSCAN, and MLP (multilayer perceptron). SOM grouped patients with similar characteristics. DBSCAN then created 9 clusters, where the patients have a different survivability time. Finally, the survivability prediction accuracy for each group was improved by using MLP.

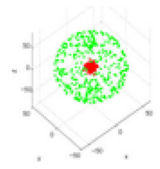
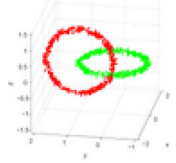
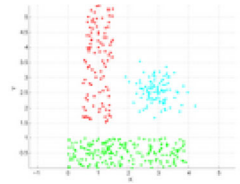
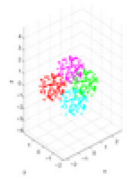
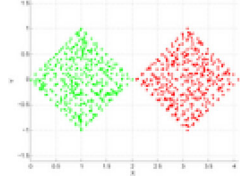
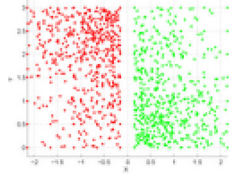
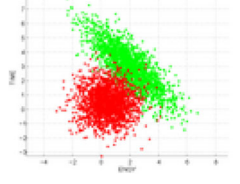
Most works in the area of clustering algorithms deal with the experimental comparison of individual clustering methods on various datasets. Works where the authors developed hybrid algorithms taking grounds in a combination of various clustering methods are rare. We have not found any works that would present a hybrid combination of individual clustering algorithms with SOM as well as have been evaluated on benchmark datasets.

Therefore, it is this issue that is dealt with in this article. We focused on the possibility of combining the SOM method with other clustering methods—CLARA, CURE, and K-means, which we consider the main contribution of this article.

4 SOM-based clustering methodology

This work presents our proposed methodology based on SOM for analyzing the number of clusters in datasets. Figure 1 depicts how it works. The input data is preprocessed (normalized) in order to create a training set for SOM. Next, we will propose a suitable network topology (dimensions of the output layer in 2D), initialize the learning parameter α with the surroundings of the winning neuron ρ , and randomly generate the values of the network weights w_{ij} . SOM adaptation follows. The results of the learning process are neuron coordinates and their distances to the neighboring neurons. Prior to entering the selected clustering algorithm, we eliminate neurons that do not have any assigned object. According to the clustering algorithm, we initialize its parameters and perform the calculation itself, i.e., creation of clusters according to the selected

Table 2 Used selected test problems from the FCPS dataset [29]

Name	Problem	Description	Image
Atom	Different variances and linear not separable	Size 800 Dimensions 3 Classes 2	
Chainlink	Linear not separable	Size 1000 Dimensions 3 Classes 2	
Lsun	Different variances and inter cluster distances	Size 400 Dimensions 2 Classes 3	
Tetra	Almost touching clusters	Size 400 Dimensions 3 Classes 4	
TwoDiamonds	Cluster borders defined by density	Size 800 Dimensions 2 Classes 2	
Wingnut	Density vs. distance	Size 1016 Dimensions 2 Classes 2	
EngyTime	Gaussian mixture	Size 4096 Dimensions 2 Classes 2	

algorithm. It then works with the preprocessed data and its performance is more effective compared with its outputs on unprocessed data, which is proved by the carried out experimental study.

Method SOM is primarily useful in the first phases of the process when the knowledge of the data is too vague. An advantage of SOM, as a tool to reduce and cluster data, consists in its ability to structure data topologically based on mutual connections and their projection in a 2D map.

5 Experiments

The comparison of individual clustering methods used the benchmark dataset Fundamental Clustering Problems Suite (FCPS) [29]. The FCPS can be downloaded from the following website: <https://www.uni-marburg.de/fb12/arbeitsgruppen/datenbionik/data>. This repository contains a set of benchmark problems that test the limits of clustering algorithms. A description of the used dataset is given in Table 2, which provides the following parameters for each

Table 3 Optimal parameters setting and Rand index

	DBSCAN		CLARA		CURE	
	Parameters	Rand index	Parameters	Rand index	Parameters	Rand index
Atom	ϵ : 15 MinPts: 3	1	Number of samples: 15 size of samples: 200	0.54	α : 0.1 Number of repeat.: 20	0.66
Chainlink	ϵ : 0.12 MinPts: 5	1	Number of samples: 3 Size of samples: 30	0.64	α : 1.45 number of repeat.: 20	0.65
Lsun	ϵ : 0.55 MinPts: 5	1	Number of samples: 5 size of samples: 100	0.9	α : 0.3 number of repeat.: 3	1
Tetra	ϵ : 0.5 MinPts: 10	0.95	Number of samples: 15 size of samples: 150	1	α : 0.5 number of repeat.: 10	1
TwoDiamonds	ϵ : 0.16 MinPts: 9	1	Number of samples: 5 size of samples: 75	1	α : 0.4 number of repeat.: 4	1
Wingnut	ϵ : 2.5 MinPts: 5	1	Number of samples: 40 size of samples: 700	0.91	α : 0.75 number of repeat.: 6	0.97
EngyTime	ϵ : 0.29 MinPts: 38	0.72	Number of samples: 45 size of samples: 1250	0.91	α : 0.15 number of repeat.: 20	0.77

tested set: dimension, number of clusters, number of objects, and clustering problem they represent.

5.1 Metrics

We used the metrics of the Rand index, which expresses the level of similarity between two data clusters and which is often used to compare the performance of various clustering algorithms Table 3.

Definition [23]: Given a set of n elements $S = \{o_1, \dots, o_n\}$ and two partitions of S to compare, $Y = \{Y_1, \dots, Y_r\}$, a partition of S into r subsets, and $Z = \{Z_1, \dots, Z_s\}$, a partition of S into s subsets, define the following:

- a , the number of element pairs in S which belong to the same subset in Y and to the same subset in Z
- b , the number of element pairs in S which belong to different subsets in Y and to different subsets in Z
- c , the number of element pairs in S which belong to the same subset in Y and to different subsets in Z
- d , the number of element pairs in S which belong to different subsets in Y and to the same subset in Z

The Rand index, R is defined as (3):

$$R = \frac{a + b}{a + b + c + d} \tag{1}$$

The Rand index can be perceived as a percentage ratio of right decisions made by the algorithm [23]. The following formula is used to compute it (4):

$$R = \frac{TP + TN}{TP + FP + FN + TH}, \tag{2}$$

where TP represents the number of true positives, TN represents the number of true negatives, FP represents the number of false positives, and FN represents the number of false negatives.

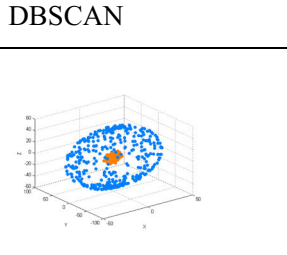
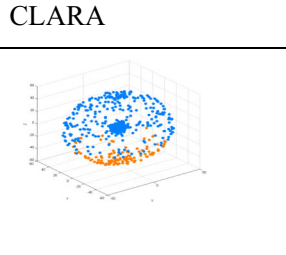
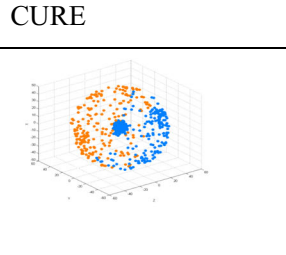
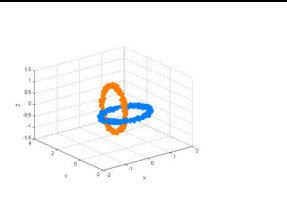
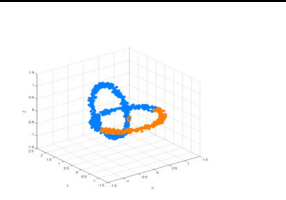
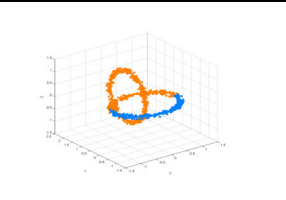
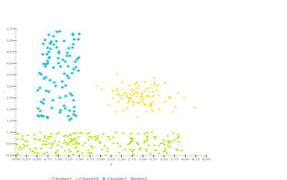
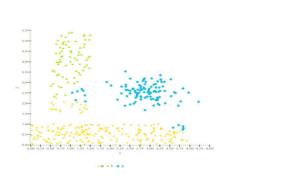
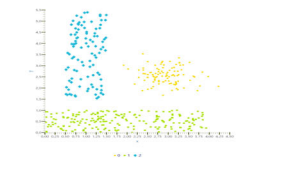
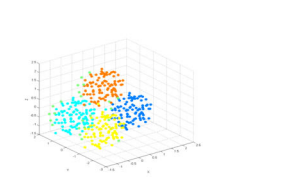
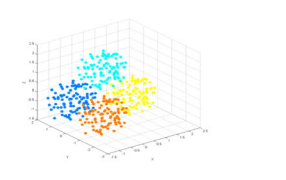
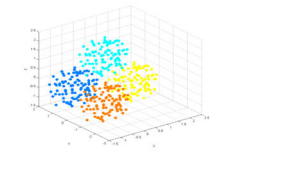
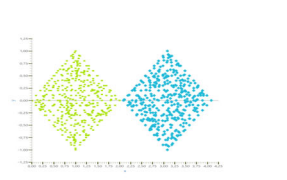
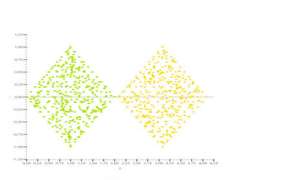
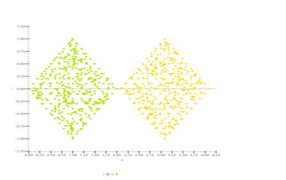
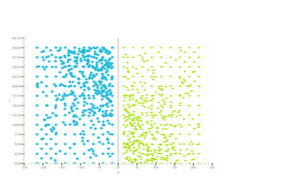
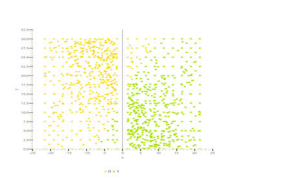
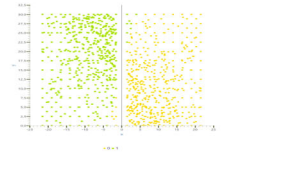
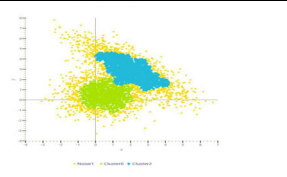
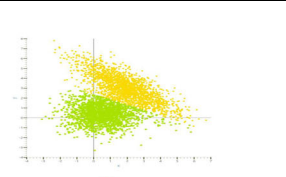
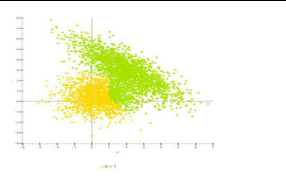
5.2 Experimental validation

Each method was tested on selected data repeatedly. It namely concerned 100 runs for methods that can give a different result based on the initial object division. Methods that are not sensitive to object arrangement were run several times as well for the purposes of finding suitable parameters setting. For each method, we provided the successfulness rate of correct placing of objects into the clusters with respect to pattern classes using the Rand index, Eq. (2).

Table 4 provides an overview of the results for individual methods with optimal parameter setting, with the best-achieved result in bold for a given tested set. Optimal parameters setting for each used method is shown in Table 3. The most successful on this dataset was method DBSCAN, which achieved a 100% rating in five sets. Method CURE achieved 100% successfulness in three datasets, while it was better than DBSCAN in one case. Method CLARA achieved 100% successfulness in two datasets, while it was more successful than DBSCAN in one of them. Methods CLARA and CURE were not able to cope with linear not separable clusters. Unlike DBSCAN, both methods were able to create better clusters for set *Engytime*, method CLARA primarily.

Table 5 states an overview of result for the SOM method. For each tested set, a U-matrix and an activation

Table 4 Experimental outcomes (noise is depicted in green)

	DBSCAN	CLARA	CURE
Atom			
Chainlink			
Lsun			
Tetra			
TwoDiamonds			
Wingnut			
EngyTime			

map for the output neuron layer is depicted. For the purposes of better visibility, the created clusters were highlighted using a black line. There is a clear ability of the SOM method to determine the number of clusters. The clusters were formed by neurons with the smallest distance

to their topological neighbors (in the U-matrix depicted in a blue hue). In the activation map, there is a number of assigned objects for each neuron. The parameters of the SOM method for individually tested sets are stated in Table 6. For individual sets, the size of the output layer and

Table 5 Setting the parameters of the SOM method

	Network size	It	α_{start}
Atom	10 × 10	40,000	0.05
Chainlink	15 × 15	550,000	0.1
Lsun	12 × 12	50,000	0.1
Tetra	15 × 15	45,000	0.1
TwoDiamonds	10 × 10	45,000	0.1
Wingnut	10 × 10	55,000	0.1
EngyTime	10 × 10	55,000	0.1

the value of the learning parameter (α_{start}) were determined experimentally based on the complexity of the solved problem.

6 Experimental outcomes and the contribution of the proposed methodology

The outcomes of the SOM method were subsequently processed by other clustering methods. The outcomes of the SOM methods are neurons coordinates and their distances to the neighboring neurons. In order to achieve better results in further clustering, it was necessary to remove neurons that do not have any assigned object. This resulted in better cluster separation. The parameters of the testing methods are provided in Table 7. The K-means method only requires setting the number of clusters. This parameter is set by the user according to the given data set. The outcomes of the experimental study are presented in Table 8, which states the averages values of the Rand index from 100 measurement runs.

Method DBSCAN can be considered the most successful out of the compared methods (it achieved Rand index 1 in most cases). That is why we did not combine it with the SOM method. However, this method is very sensitive to the initialization of parameters. One of the possible ways how to find the initial setting of the parameter ε is to use the SOM method; specifically for the U-matrix, where the parameter ε has the value ranging from the smallest to the biggest distance to the neighboring neurons.

All of the used methods combined with the SOM method achieved better results than when used separately. The methods were able to solve problems which would otherwise be unsolvable for them separately, or they would

not solve them correctly. The combination of SOM and other clustering methods led to the most significant improvement in the set called Atom, which no method was able to solve. This set was solved at 100% successfulness rate by a combination of SOM + CLARA, SOM + K-means; 99% by SOM + CURE, respectively. However, no combination with SOM led to a solution for the set called Chainlink. It was caused by the fact that the clusters are linear not separable, which also remained in the output SOM layer. Concerning the set Engytime, the successfulness rate was comparable with that achieved by using the methods separately.

7 Comparison with other academic works

In this chapter, we focused on published academic works in the area of clustering algorithms, where the proposed approached were experimentally verified on the FCPS datasets [29] and the outcomes were evaluated according to the Rand index, Eq. (2). The chart in Fig. 2 depicts the average values of the Rand index achieved by methods published by other authors and by the methods proposed in this work. The red color shows values achieved by our improved hybrid clustering algorithms, i.e., SOM + K-means, SOM + CURE, and SOM + CLARA. The chart shows that the proposed approached overcame, in most cases, the experimental outcomes published in other academic works [2–5, 7, 8, 13, 15, 20, 21, 24–26, 30].

8 Conclusions and future work

Most works from the area of clustering algorithms deal with an experimental comparison of individual clustering methods on various datasets. Works where the authors also proposed a hybrid algorithm stemming from a various combination of clustering algorithms are rare. This article deals with this issue.

The proposed methodology increases the efficiency of clustering methods, which was proved by the performed experimental study. This is the main contribution of our work. We focused on the possibility of combining method SOM (Self-Organizing Maps) with other clustering methods—CLARA, CURE, and K-means. Method SOM is primarily useful in the first phases of the process when knowledge of the data is too vague. The use of the selected clustering algorithm follows. The algorithm then works with preprocessed data. Its performance, compared with

Table 6 U—matrix and activation map of the outputs of the SOM method

Atom		
Chainlink		
Lsun		
Tetra		
TwoDiamonds		
Wingnut		
EngyTime		

Table 7 Setting the parameters of the methods

	DBSCAN	CLARA	CURE	K-means
Atom	ϵ : 15	Number of samples: 10	α : 0.5	Number of clusters: 2
	MinPts: 3	Size of samples: 75	Number of repeat.: 5	
Chainlink	ϵ : 0.12	Number of samples: 10	α : 0.8	Number of clusters: 2
	MinPts: 5	Size of samples: 200	Number of repeat.: 5	
Lsun	ϵ : 0.55	Number of samples: 5	α : 0.9	Number of clusters: 3
	MinPts: 5	Size of samples: 75	Number of repeat.: 5	
Tetra	ϵ : 0.5	Number of samples: 10	α : 0.5	Number of clusters: 4
	MinPts: 10	Size of samples: 150	Number of repeat.: 5	
TwoDiamonds	ϵ : 0.16	Number of samples: 10	α : 0.3	Number of clusters: 2
	MinPts: 9	Size of samples: 75	Number of repeat.: 5	
Wingnut	ϵ : 2.5	Number of samples: 10	α : 0.5	Number of clusters: 2
	MinPts: 5	Size of samples: 200	Number of repeat.: 5	
EngyTime	ϵ : 0.29	Number of samples: 10	α : 0.8	Number of clusters: 2
	MinPts: 38	Size of samples: 300	Number of repeat.: 5	

Table 8 Rand index—summary of the outcomes combined with the SOM method

	DBSCAN	CLARA	SOM + CLARA	CURE	SOM + CURE	K-means	SOM + K-means
Atom	1	0.54	1	0.66	0.99	0.59	1
Chainlink	1	0.64	0.72	0.65	0.71	0.55	0.66
Lsun	1	0.9	1	1	1	0.72	0.99
Tetra	0.95	1	1	1	1	1	0.99
TwoDiamonds	1	1	1	1	1	1	1
Wingnut	1	0.91	1	0.97	1	0.72	1
EngyTime	0.72	0.91	0.93	0.77	0.93	0.91	0.85

the outcomes on unprocessed data, is more effective, which is proved by the performed experimental study on the benchmark dataset Fundamental Clustering Problems Suite (FCPS). Part of the experimental verification was also a comparison of the achieved results with other approaches using this dataset based on standard metrics—the Rand index. The chart in Fig. 2 shows that in most cases, our proposed approach overcame the outcomes published in other academic works.

With respect to our future work, we would like to focus on another improvement of the efficiency of clustering algorithms, primarily combined with algorithms from the area of soft computing. Regarding SOM, we would like to propose an algorithm which could estimate suitable sizes of the output layer as well as define its suitable dimension (1D, 2D, or 3D ...).

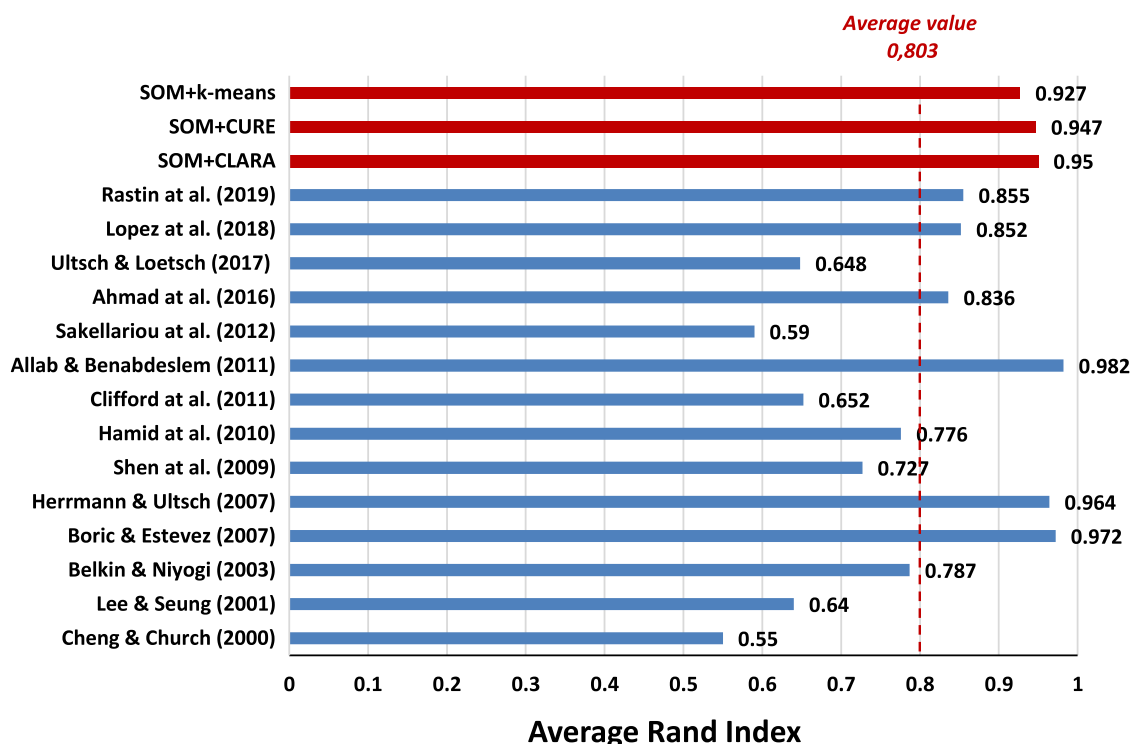


Fig. 2 Average Rand index values. The red color shows values achieved by the proposed method, i.e., SOM + K-means, SOM + CURE, and SOM + CLARA (color figure online)

Funding This work was supported by TACR, project no. TL02000313 and also by University of Ostrava grant SGS05/PrF/2020.

Declarations

Conflict of interest There have been no involvements that might raise the question of bias in the work reported or in the Conclusions, implications, or opinions stated. The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

- Aghajari E, Chandrashekhar GD (2017) Self-organizing map based extended fuzzy C-means (SEEFC) algorithm for image segmentation. *Appl Soft Comput* 54:347–363
- Ahmad T, Desai N, Wilson F, Schulte P, Dunning A, Jacoby D, O'Connor C (2016) Clinical implications of cluster analysis-based classification of acute decompensated heart failure and correlation with bedside hemodynamic profiles. *PLoS one* 11(2):0145881
- Allab K, Benabdeslem K (2011) Constraint selection for semi-supervised topological clustering. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 28–43). Springer, Berlin, Heidelberg.
- Belkin M, Niyogi P (2003) Using manifold structure for partially labeled classification. In *Advances in neural information processing systems* (pp. 953–960).
- Boric N, Estevez PA (2007) Genetic programming-based clustering using an information theoretic fitness measure. In *2007 IEEE Congress on Evolutionary Computation* (pp. 31–38). IEEE.
- Chen Q, Yuen KKF, Guan C (2017) Towards a hybrid approach of self-organizing map and density-based spatial clustering of applications with noise for image segmentation. In *2017 10th International Conference on Developments in eSystems Engineering (DeSE)* (pp. 238–241). IEEE.
- Cheng Y, Church GM (2000) Biclustering of expression data. In *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology* (Vol. 8, No. 2000, pp. 93–103).
- Clifford H, Wessely F, Pendurthi S, Emes RD (2011) Comparison of clustering methods for investigation of genome-wide methylation array data. *Front Genet* 2:88. <https://doi.org/10.3389/fgene.2011.00088>
- Dogan Y, Birant D, Kut A (2013) SOM++: integration of self-organizing map and k-means++ algorithms. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition* (pp. 246–259). Springer, Berlin, Heidelberg.
- Ester M, Krieger HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. *Knowledge Discov Data Min* 96(34):226–231
- Everitt BS, Landau S, Leese M, Stahl D (2011) *Cluster analysis*. Wiley
- Firnhaber C, Pühler A, Küster H (2005) EST sequencing and time course microarray hybridizations identify more than 700 *Medicago truncatula* genes with developmental expression regulation in flowers and pods. *Planta* 222(2):269–283
- Hamid JS, Meaney C, Crowcroft NS, Granerod J, Beyene J (2010) Cluster analysis for identifying sub-groups and selecting

- potential discriminatory variables in human encephalitis. *BMC Infect Dis* 10(1):364
14. Hennig C, Meila M, Murtagh F, Rocci R (Eds.) (2015) *Handbook of cluster analysis*. CRC Press.
 15. Herrmann L, Ultsch A (2007) Label propagation for semi-supervised learning in self-organizing maps. In *International Workshop on Self-Organizing Maps: Proceedings (2007)*.
 16. Huai-bin W, Hong-liang Y, Zhi-Jian XU, Zheng Y (2010) A clustering algorithm use SOM and k-means in intrusion detection. In *2010 International Conference on E-Business and E-Government* (pp. 1281–1284). IEEE.
 17. Kaufman L, Rousseeuw PJ (1987) Clustering by means of Medoids. In: Dodge Y (ed) *Statistical data analysis based on the L_1 norm and related methods*. North-Holland, Amsterdam, pp 405–416
 18. Kohonen T (1982) Self-organized formation of topologically correct feature maps. *Biol Cybern* 43(1):59–69
 19. Kotyrba M, Volná E, Komínková Oplatková Z (2014) Comparison of modern clustering algorithms for twodimensional data. In *Proceedings-28th European Conference on Modelling and Simulation, ECMS 2014*. European Council for Modelling and Simulation.
 20. Lee DD, Seung HS (2001) Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems* (pp. 556–562).
 21. Lopez C, Tucker S, Salameh T, Tucker C (2018) An unsupervised machine learning method for discovering patient clusters based on genetic signatures. *J Biomed Inform* 85:30–39
 22. MacQueen J (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, No. 14, pp. 281–297).
 23. Rand WM (1971) Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc* 66(336):846–850
 24. Rastin P, Cabanes G, Verde R, Bennani Y, Couronne T (2019) Generative histogram-based model using unsupervised learning. In *International Conference on Neural Information Processing* (pp. 634–646). Springer, Cham.
 25. Shen R, Olshen AB, Ladanyi M (2009) Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics* 25(22):2906–2912
 26. Sakellariou A, Sanoudou D, Spyrou G (2012) Combining multiple hypothesis testing and affinity propagation clustering leads to accurate, robust and sample size independent classification on gene expression data. *BMC Bioinform* 13(1):270
 27. Shukla N, Hagenbuchner M, Win KT, Yang J (2018) Breast cancer data analysis for survivability studies and prediction. *Comput Methods Program Biomed* 155:199–208
 28. Šefar S (2017) *Comparative study of clustering methods* (in Czech). Diploma Thesis. University of Ostrava.
 29. Ultsch A (2005) Clustering with SOM: U*C. In *Proceedings of the Workshop on Self-Organizing Maps (WSOM '05)*, Paris, France, (pp. 75–82).
 30. Ultsch A, Loetsch J (2017) Machine-learned cluster identification in high-dimensional data. *J Biomed Inform* 66:95–104
 31. Van Laerhoven K (2001) Combining the self-organizing map and k-means clustering for on-line classification of sensor data. In *International Conference on Artificial Neural Networks* (pp. 464–469). Springer, Berlin, Heidelberg.
 32. Wu J, Xia J, Chen J, Cui Z (2011) Moving object classification method based on SOM and K-means. *JCP* 6(8):1654–1661
 33. Xu D, Tian Y (2015) A comprehensive survey of clustering algorithms. *Ann Data Sci* 2(2):165–193
 34. Yorek N, Ugulu I, Aydin H (2016) Using self-organizing neural network map combined with ward's clustering algorithm for visualization of students' cognitive structural models about aliveness concept. *Comput Intell Neurosci*, 2016.

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