Expert Systems With Applications 150 (2020) 113294

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Weighted consensus clustering and its application to Big data

Rasim M. Alguliyev, Ramiz M. Aliguliyev, Lyudmila V. Sukhostat*

Institute of Information Technology, Azerbaijan National Academy of Sciences, 9A, B. Vahabzade Street, AZ1141, Baku, Azerbaijan

ARTICLE INFO

Article history:
Received 28 November 2018
Revised 9 August 2019
Accepted 5 February 2020
Available online 14 February 2020

Keywords:
Weighted consensus clustering
Big data
Utility function
Purity-based utility function
Co-association matrix

ABSTRACT

The aim of this study is the development of a weighted consensus clustering that assigns weights to single clustering methods using the purity utility function. In the case of Big data that does not contain labels, the utility function based on the Davies-Bouldin index is proposed in this paper. The Banknote authentication, Phishing, Diabetic, Magic04, Credit card clients, Covertype, Phone accelerometer, and NSL-KDD datasets are used to assess the efficiency of the proposed consensus approach. The proposed approach is evaluated using the Euclidean, Minkowski, squared Euclidean, cosine, and Chebychev distance metrics. It is compared with single clustering algorithms (DBSCAN, OPTICS, CLARANS, k-means, and shared nearby neighbor clustering). The experimental results show the effectiveness of the proposed approach to the Big data clustering in comparison to single clustering methods. The proposed weighted consensus clustering using the squared Euclidean distance metric achieves the highest accuracy, which is a very promising result for Big data clustering. It can be applied to expert systems to help experts make group decisions based on several alternatives. The paper also provides directions for future research on consensus clustering in this area.

1. Introduction

Today, expert systems are widely used in various areas, such as business, medicine, process management, space technology, etc. To extract knowledge from existing growing information, the need to develop approaches based on machine learning is relevant. Practice shows that clustering is used for research and evaluation of clustered knowledge. Thus, clustered expert systems are developed to explain the data when diagnosing and making final decisions.

Now there is a tendency to use ensemble methods in cluster analysis when it is difficult to apply a certain algorithm to data clustering (Ghosh & Acharya, 2011). One of the clustering problems is a high computational cost, which makes it difficult to get good results on the complex Big data.

A consensus approach is widely used to increase the accuracy and stability of clustering results (Berikov & Pestunov, 2017; Franek & Jiang, 2014; Ghosh & Acharya, 2011; Jia, Liu & Jiao, 2011; Kashef & Kamel, 2010; Nguyen & Caruana, 2007; Wu et al., 2017). The approach is to find an agreed solution due to the possibility of sharing the methods of cluster analysis. It is possible to construct the most suitable clustering scheme for a particular domain by applying a consensus approach to a different set of algorithms according to their advantages and distinctive features. In developing the final decision, different points of view are considered that not only do not contradict but, on the contrary, compensate for the shortcomings of each method.

Consensus clustering helps generate reliable partitions, handle noise, and outliers (Nguyen & Caruana, 2007). It acts as a promising solution for data clustering. A lot of work was devoted to consensus clustering based on k-means, but the studies are still preliminary and fragmentary.

At the same time, consensus clustering searches for a common partition, which is consistent with the existing single clustering methods. In this case, the problem is the development of the utility function and its efficient optimization. To this end, the paper proposes a utility function based on purity.

Consensus clustering is a promising solution for finding clusters in high-dimensional data because of its reliability and versatility. The main contribution of our work is as follows:

(1) In this paper, a weighted consensus clustering for the efficient combination of single partitions for Big data application is proposed. The results of the study show that the weighted consensus clustering based on purity utility function (PWCC) is efficient, clear, and reliable.
(2) The proposed approach can be applied to any expert system. It is designed for any number of alternatives and can be implemented in any architecture. Weighted consensus clustering will allow experts to use it when making group decisions.

* Corresponding author.
E-mail address: lsuhostat@hotmail.com (L.V. Sukhostat).

https://doi.org/10.1016/j.eswa.2020.113294
0957-4174/

(3) The experimental results on real datasets of various size using the Euclidean, Minkowski, squared Euclidean, cosine, and Chebychev distance metrics show that weighted consensus clustering is highly efficient, comparable and superior to state-of-the-art approaches according to clustering quality.

The rest of the paper is organized as follows. Section 2 gives a literature review of existing works on consensus clustering. In Section 3, clustering evaluation metrics are presented. Single clustering methods are considered in Section 4. Section 5 describes the proposed approach. Section 6 compares the proposed approach with single clustering methods to illustrate the benefits of the proposed implementation. Section 7 provides a statistical significance test. Then a discussion of the results is given, followed by conclusion and future work.

2. Related work

Data clustering requires the development of a general algorithm and selection of the best decision criteria (Cabrerizo et al., 2015; Pérez, Mata, Chiclana, Kou & Herrera-Viedma, 2016). However, due to the large variety of existing methods, it is difficult to choose the most effective one. Researchers have developed many clustering approaches that take into account the advantages and disadvantages of various methods to construct one final clustering (Table 1). Nguyen and Caruana (2007) presented three Expectation-Maximization (EM) like algorithms (Iterative Voting Consensus, Iterative Probabilistic Voting Consensus, and Iterative Pairwise Consensus) to solve the problem of consensus clustering. The proposed algorithms are variations of the k-means algorithm using different distance measures applied to the vector of base-level clustering.

Fuzzy consensus clustering (FCC) was studied by Wu et al. (2017). The objective function of FCC using the proposed fuzzified contingency matrix was defined. Then a family of FCC Utility functions termed as FCCU that can transform FCC to a weighted piecewise fuzzy c-means clustering (piFCM) problem was obtained. Experiments on various real-world datasets demonstrated the excellent performance of FCC, even with a majority of poor basic partitions.

A homogeneous clustering ensemble based on the Particle Swarm Clustering algorithm (PSC) was proposed by De Oliveira, Szabo and de Castro (2017). It can be considered as a two-step procedure: first, several base partitions are obtained from the data, afterward; a final partition is computed by a consensus function that takes the base partitions as input. The proposed clustering ensemble uses PSC both to generate the base partitions and in the consensus function, and supports both disjoint and overlapping partitions.

A graph-based algorithm for combining multiple clusterings based on cliques (maximally complete subgraphs) was described (Mimaroglu & Yagci, 2012). The algorithm finds a substantial subset of all the cliques quickly for speedup of clustering in a large graph.

An ensemble clustering approach based on ensemble-driven cluster uncertainty estimation and local weighting strategy was proposed (Huang, Wang & Lai, 2018). The uncertainty of clusters was estimated by considering the cluster labels in the entire ensemble based on an entropic criterion, and a new ensemble-driven cluster validity index (ECI) was proposed. A local weighting scheme was presented to extend the conventional co-association matrix into the locally weighted co-association matrix via the ECI measure. Also, two novel consensus functions locally weighted evidence accumulation (LWEA) and locally weighted graph partitioning (LWGP) consensus functions were proposed. Although the above approaches are valid solutions, they do not consider a large amount of data clustering.

The contribution of work (Hidri, Zoghlami & Ayed, 2018) concerns parallel algorithms and distributed clustering used to analyze Big data. The aim is to cluster data in a compact format, which represents an informative version of the whole data. Comprehensive experiments on both numerical and categorical datasets were conducted to study the impact of the use of consensus-based sampling tendency with clustering in a large scale environment.

An ensemble clustering approach based on sparse graph representation and probability trajectory analysis was proposed (Huang, Lai & Wang, 2016). A dense similarity measure was further derived from the K-elite neighbor graph using probability trajectories. On its basis, two consensus functions (probability trajectory accumulation (PTA) and probability trajectory based graph partitioning (PTGP)) were proposed.

A spectral ensemble clustering (SEC) algorithm was proposed in Liu, Wu, Liu, Tao and Fu (2017). The time and space complexities of SEC were decreased by identifying the equivalent relationship between SEC and weighted K-means. The intrinsic consensus objective function of SEC, which bridges the co-association matrix based methods with the methods with explicit global objective functions, was shown. The robustness, generalizability, and convergence properties of SEC were investigated to show its superiority in theory. It was extended to handle incomplete basic partitions.

The volume of information that experts need to process is growing rapidly with the development of technology. It is necessary to develop new intellectual approaches based on machine learning to help experts make consensus decisions from a possible set of alternatives. So, Zheng, Li, Hong and Li (2013) proposed PENETRATE (PErsonalized NEws recommendaTion framework using ensemble hieRArchical clusTEring) to recommend news articles within each user's group. Probabilistic Latent Semantic Indexing (Hofmann, 1999) and Latent Dirichlet Allocation (Blei, Ng & Jordan, 2003) models were used to create a user's profile.

Another example of a consensus clustering application is medical expert systems for diagnosing various diseases (Lock & Dunson, 2013). An integrative statistical model for the simultaneous estimation of both the consensus clustering and the source-specific clusterings was proposed by Lock and Dunson (2013). The approach showed flexible and computationally scalable results in clustering multisource biomedical data.

The paper (Alhusain & Hafez, 2017) proposes a random forest (RF) cluster ensemble (RfcluE), a cluster ensemble approach to discover the underlying structure of genetic data based on RFs. However, the main concern underlying the RF algorithm is that, for each run, a different proximity matrix is generated due to its random nature, therefore producing a different clustering result each time. In the paper, this problem was solved.

Summarizing the analysis of the state of research on the application of consensus clustering to the Big data analysis, we can draw the following conclusions. Works aimed at improving the quality of clustering often require large computational resources. Also, consensus clustering is a fairly popular research area due to the continuous growth of data volumes. This confirms the importance of our research.

This paper proposes a new method based on weighted consensus using the purity utility function for Big data clustering. To evaluate the proposed approach, in addition to k-means, various other clustering algorithms are considered (DBSCAN (Ester, Kriegel, Sander & Xu, 1996), OPTICS (Ankerst, Breunig, Kriegel & Sander, 1999), CLARANS (Ng & Han, 1994) and shared nearby neighbor clustering (SNNC) (Shaneck, Kim & Kumar, 2009). Experiments on medium and large datasets show that the proposed approach ensures high clustering efficiency.

Table 1
Summary of methods based on consensus clustering.

References	Proposed approach	Main contribution	Method limitations	Experimental datasets	Big data clustering
De Oliveira et al. (2017)	A homogeneous clustering ensemble based on PSC	• A consensus function based on the Particle Swarm Clustering algorithm. • An alignment-free efficient representation for both disjoint and overlapping partitions.	• Sensitivity of initialization issues • High computational costs	Iris, Wine, Soybean, Inosphere, Heart (47–351 instances) datasets	–
Hidri et al. (2018)	Divide-and-conquer strategies using the consensus tendency combined with sampling to handle distributed storage, analysis and clustering of massive data in large-scale environment	• Speeds up the calculation based on a consensus tendency • Increases scalability based on MapReduce model combined with data sampling.	Time and space complexity	KDDCup1999, Forest Covertype, 2 × KDD, 4 × KDD large datasets	+
Alhusain and Hafez (2017)	A cluster ensemble approach (RFcluE) for determining the underlying structure of genetic data based on RFs	Combining multiple clusterings, generated based on RFs, produces high quality and robust clustering results in comparison to a single run of RF clustering	No evaluation in terms of time complexity	Human genotype large datasets	+
Huang et al. (2016)	An ensemble clustering approach based on sparse graph representation and probability trajectory analysis	• Uses only a small number of probably reliable links rather than all graph links regardless of their reliability • Incorporates global information to construct more accurate local links by exploiting the random walk trajectories.	Computational cost increases with the number of base clusterings	Multiple Features, Image Segmentation, MNIST, Optical Digit Recognition, Landsat Satellite, Pen Digits, USPS, Forest Covertype, KDD99-10P, KDD99 datasets	+
Wu et al. (2017)	Systematic framework of FCC based on a utility function	• A family of utility functions for FCC was gained • FCC trnsformation to a weighted piecewise FCM, which gains high efficiency via iterative process. • Both vertical and horizontal segmentation schemes for big data clustering, which is further parallelized on the Spark platform.	The method does not minimize the dimensionality after data grouping.	Wine, Dermatology, Libras, Breast_w, Satimage, Pendigits, tr12 CLUTO (178-10992 instances) datasets	–
Liu et al. (2017)	Spectral Ensemble Clustering (SEC) for Big data	• Using spectral clustering of a co-association matrix decreases the time and space complexities of SEC. • SEC was extended to adapt to incomplete basic partitions, which enables a row-segmentation scheme suitable for big data clustering.	High computational costs	Iris, Wine, MNIST, Dermatology, Libras, Breast_w, Satimage, Pendigits, cacmcisi CLUTO, classic CLUTO, cranmed CLUTO, hitech CLUTO, k1b CLUTO, la12 CLUTO, mm CLUTO, re1 CLUTO, reviews CLUTO, sports CLUTO, tr11 CLUTO, tr12 CLUTO, tr41 CLUTO, tr45 CLUTO, letter LIBSVM datasets	+
Huang et al. (2018)	An ensemble clustering approach based on cluster uncertainty estimation and local weighting strategy	• Estimation the clusters uncertainty using an entropic criterion, which requires no access to the original data features. • Cluster validity index to evaluate and weight the clusters in the ensemble to evaluate the reliability at the cluster-level. • Two novel consensus functions to construct the final clusterings.	Sensitivity of initialization issues	Caltech20, Forest Covertype, Image Segmentation, ISOLET, Letter Recognition, Landsat Satellite, Multiple Features, MNIST, Optical Digit Recognition, Pendigits, Semeion, Steel Plates Faults, Texture, Vehicle Silhouettes, USPS datasets	–
Nguyen and Caruana (2007)	EM-like consensus clustering algorithms which utilize a feature map constructed from the set of base clusterings	• Variations of k-means using different distance measures applied to the vector of base clusterings. • The algorithms generate multiple consensus clusterings with different restarts, and the best consensus clustering can be selected.	High computational costs	Australia, Bergmark, Forest Covertype, Letters datasets	–
Lock and Dunson (2013)	Bayesian consensus clustering	• Computationally scalable • Robust to the unique features of each data.	High computational costs	Multisource biomedical datasets	+

(*continued on next page*)

Table 1 (*continued*)

References	Proposed approach	Main contribution	Method limitations	Experimental datasets	Big data clustering
Zheng et al. (2013)	PENETRATE framework based on consensus hierarchical clustering method for news recommendation	• Integrates multiple group-oriented news hierarchies to capture the general reading preference of individual users • Automatic determination of the number of clusters • Discovery of arbitrary clustering shapes	• The accuracy is not satisfactory. • Limited news clustering by membership to a single cluster	Websites news dataset	+
Mimaroglu and Yagci (2012)	Cliques for combining multiple clusterings (CLICOM) method	Method produces good quality final clusterings due to its similarity measure based on object co-associations.	• Depends on the structure of the graph • Sensitivity of initialization issues	IMAGESEG, Iris, Glasside and synthetic (1000–40000 features) datasets	–

3. Evaluation metrics

External and internal indices are considered in this paper to evaluate the clustering results. The external index refers to the comparison of a clustering solution with a real clustering. It is important when evaluating the performance of the clustering algorithm on large datasets. The internal index evaluates clusters against their structural properties. Internal cluster validation is aimed at measuring the quality of clustering under real conditions when there is no knowledge of real clustering.

3.1. External validity metrics

Taking into account the compactness and the separation factors, the metrics for clustering methods evaluation indicate the correctness of the separation into clusters.

Assume that the dataset D is divided into classes $C^+ = (C_1^+, \ldots, C_{k^+}^+)$ (true clustering). And using the clustering procedure to this dataset, clusters $C = (C_1, \ldots, C_k)$ were obtained.

A comparison of the clustering methods solutions is based on counting the number of coincidences C^+ and C. Based on the results, a decision is made: an abnormal/normal behavior. The most well-known metrics for estimating the distance of clustering methods based on pairs of data points are purity (Boutin & Hascoet, 2004; Rubinov, Soukhorukova & Ugon, 2006), the Mirkin metric (Mirkin, 1996), partition coefficient (Bezdek & Pal, 1998), the variation of information (Patrikainen & Meila, 2006), F-measure (Rosenberg & Hirschberg, 2007).

Purity. The purity of the entire collection of clusters is estimated as a weighted sum of the purities of individual clusters (Aliguliyev, 2009; Boutin & Hascoet, 2004; Rubinov et al., 2006):

$$purity(C) = \frac{1}{n}\sum_{p=1}^{k} \max_{p^+=1,\ldots,k^+} \left|C_p \cap C_{p^+}^+\right| \tag{1}$$

where k^+ is the initial number of classes, and k is the number of clusters to be found.

Mirkin metric. This metric is defined as follows (Mirkin, 1996):

$$M(C, C^+) = \frac{1}{n^2}\left(\sum_{p=1}^{k}|C_p|^2 + \sum_{p^+=1}^{k^+}\left|C_{p^+}^+\right|^2 - 2\sum_{p=1}^{k}\sum_{p^+=1}^{k^+}\left|C_p \cap C_{p^+}^+\right|^2\right) \tag{2}$$

This metric is equal to 0 for identical clusterings and is positive otherwise (Mirkin, 1996).

F-measure is calculated based on value F for clusters C_p and C_{p+}^+:

$$F\left(C_p, C_{p^+}^+\right) = \frac{2\frac{\left|C_p \cap C_{p^+}^+\right|}{|C_p|}\frac{\left|C_p \cap C_{p^+}^+\right|}{\left|C_{p^+}^+\right|}}{\frac{\left|C_p \cap C_{p^+}^+\right|}{|C_p|} + \frac{\left|C_p \cap C_{p^+}^+\right|}{\left|C_{p^+}^+\right|}} \tag{3}$$

F-measure of the entire dataset is defined as the sum of the F-measures of individual clusters, weighted by the cluster size, i.e.

$$F(C) = \sum_{p=1}^{k}\frac{|C_p|}{n}\cdot\max_{C_{p^+}^+ \in C^+} F\left(C_p, C_{p^+}^+\right) \tag{4}$$

The higher the *F*-measure, the better the clustering solution is.

Partition coefficient (PC) is calculated according to the following equation:

$$PC\left(C, C^+\right) = \frac{1}{kk^+}\sum_{p=1}^{k}\sum_{p^+=1}^{k^+}\left(\frac{\left|C_p \cap C_{p^+}^+\right|}{|C_p|}\right)^2 \tag{5}$$

The higher the value $PC(C, C^+)$, the better the clustering solution is.

Variation of information (VI). This measure evaluates the amount of information that is obtained and lost when moving from clustering C to another clustering C^+. According to Aliguliyev (2009); Patrikainen and Meila (2006)

$$VI\left(C, C^+\right) = \frac{1}{n\log n}\sum_{p=1}^{k}\sum_{p^+=1}^{k^+}\left|C_p \cap C_{p^+}^+\right|\log\left(\frac{|C_p|\left|C_{p^+}^+\right|}{\left|C_p \cap C_{p^+}^+\right|^2}\right) \tag{6}$$

The less *VI*, the better the clustering solution is.

3.2. Internal validity metrics

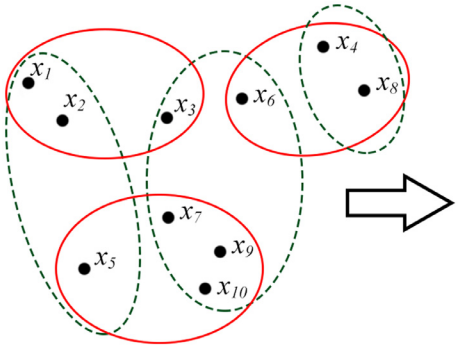
The Davies-Bouldin index and the Calinski-Harabasz index were considered as internal indices for the evaluation of the unlabelled datasets in this paper.

Davies-Bouldin index (DB). This index is based on a ratio of within-cluster and between-cluster distances (Davies & Bouldin, 1979). DB is calculated as follows:

$$DB = \frac{1}{k}\sum_{i=1}^{k} R_i \tag{7}$$

$$R_i = \max_{i\neq j}\left(\frac{\delta(C_i) + \delta\left(C_j\right)}{dist\left(C_i, C_j\right)}\right) \tag{8}$$

where k is the number of clusters, δ is the variance within the cluster, *dist* is the distance between the i^{th} and j^{th} clusters. The target value is the minimum of the index.



	$C_1^{\pi_1}$	$C_2^{\pi_1}$	$C_3^{\pi_1}$	Σ
C_1^{π}	2	1	0	3
C_2^{π}	1	3	0	4
C_3^{π}	0	1	2	3
Σ	3	5	2	10

Fig. 1. Illustration of a co-association matrix for three clusters.

Calinski–Harabasz index (CH) (Calinski & Harabasz, 1974) is characterized by the following function:

$$CH = \frac{B(k)/(k-1)}{W(k)/(n-k)} \tag{9}$$

$$B(k) = \sum_{i=1}^{k} n_i \, dist^2(O_i, O) \tag{10}$$

$$W(k) = \sum_{i=1}^{k} \sum_{x \in C_i} dist^2(x, O_i) \tag{11}$$

where k is the number of clusters, n is the number of objects in the considered dataset D, C_i is the i th cluster, n_i is the number of objects in C_i, O is the center of the dataset D, O_i is the center of C_i, $W(k)$ is the sum of the within-cluster dispersions for all the clusters, and $B(k)$ is the weighted sum of the squared distances between the C_i and dataset D. The most likely number of clusters is the value of k, at which the CH index reaches its maximum value (Calinski & Harabasz, 1974).

4. Methodology

4.1. DBSCAN

The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) method is based on the concepts of internal and boundary points, density reachability, D-connectivity, the threshold ε, and the minimum number of points in a cluster ($MinPts$) (Ester et al., 1996). The ε-neighbors of the point $p \in D$ are understood as the set of points the distance to which does not exceed ε

$$N_\varepsilon(p) = \{q \in D \,|\, dist\,(p, q) \le \varepsilon\} \tag{12}$$

$MinPts$ is chosen experimentally so that $|N_\varepsilon(q)| \ge MinPts$. $MinPts$ adjusts the "noise" threshold. This clustering algorithm is computationally simple and resistant to disturbances.

4.2. OPTICS

Sensitivity to the choice of parameters of the DBSCAN algorithm has generated a number of its modifications. One of them is OPTICS (Ordering Points to Identify the Clustering Structure) (Ankerst et al., 1999), which allows us to order the initial set and simplify the process of clustering.

A reachability diagram is constructed in this method due to which it becomes possible, with a fixed $MinPts$ value, to process not only the specified ε value but also all $\varepsilon^* < \varepsilon$. To order the set D for each of its elements, two parameters are calculated - core distance and reachability distance. OPTICS allows solving clustering problems in conditions where clusters have not only different shapes but also different data density distribution in each class.

Table 2
Co-association matrix.

π \ π_i	$C_1^{(i)}$	$C_2^{(i)}$	$\cdots$	$C_{K_i}^{(i)}$	Σ
C_1	$n_{11}^{(i)}$	$n_{12}^{(i)}$	$\cdots$	$n_{1K_i}^{(i)}$	n_{1+}
C_2	$n_{21}^{(i)}$	$n_{22}^{(i)}$	$\cdots$	$n_{2K_i}^{(i)}$	n_{2+}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
C_K	$n_{K1}^{(i)}$	$n_{K2}^{(i)}$	$\cdots$	$n_{KK_i}^{(i)}$	n_{K+}
Σ	$n_{+1}^{(i)}$	$n_{+2}^{(i)}$	$\cdots$	$n_{+K_i}^{(i)}$	n

4.3. CLARANS

CLARANS (Ng & Han, 1994) was proposed for Clustering Large Applications with Randomized Search and combines the advantages of the PAM and CLARA (Leonard & Rousseeuw, 1990) algorithms. The main idea of PAM is to select one point from each cluster as a medoid. In contrast to k-means, it includes the medoid instead of the mean, which makes the algorithm more efficient.

CLARA is a better solution for handling Big data compared to PAM. But its disadvantage is that it considers data samples, rather than complete datasets. In this regard, to improve the efficiency of clustering, the CLARANS algorithm was proposed. It uses a sampling technique to reduce the search space (Aboubi, Drias & Kamel, 2016). This approach is performed dynamically at each iteration. At the same time, the clustering process is a search by a graph which node is a set of k medoids. The cost function is assigned to each node (Berry & Browne, 2006):

$$Cost\,(X, M)\frac{\sum_{i=1}^{n} dist\,(x_i, rep(M, x_i))}{n} \tag{13}$$

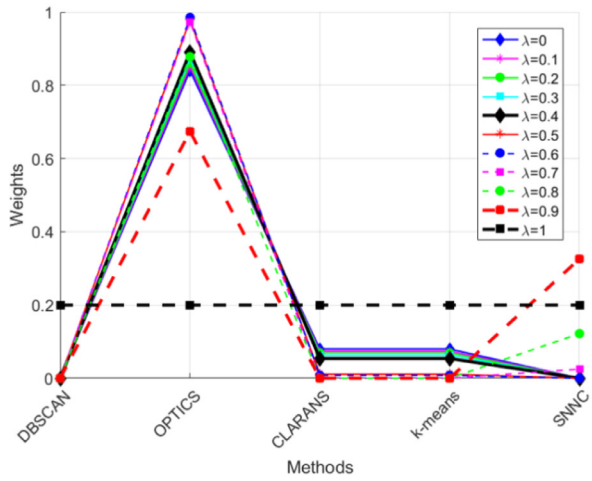
where M is the set of medoids, $rep(M, x_i)$ returns the medoid in M closest to x_i.

CLARANS, like PAM, moves from one node to one of its neighbors until it finds a minimum cost solution. CLARA is effective in reducing the search space. And the CLARANS algorithm does this dynamically.

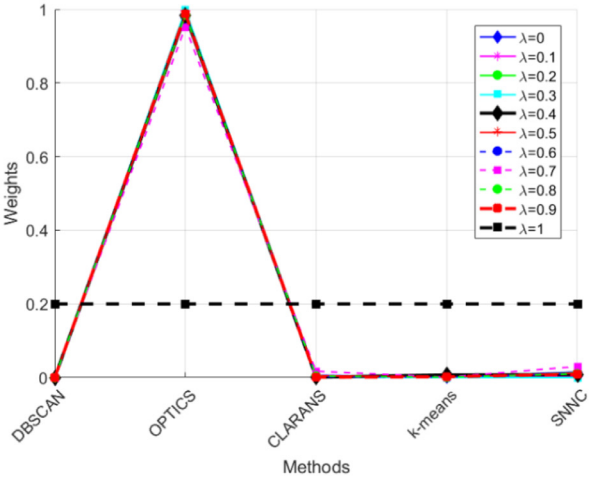
4.4. k-means

The k-means algorithm builds k clusters arranged in such a way as to minimize the standard deviation of the object's samples from the cluster's centers (Kanungo et al., 2002). At the same time, the initial arrangement of clusters greatly affects the algorithm.

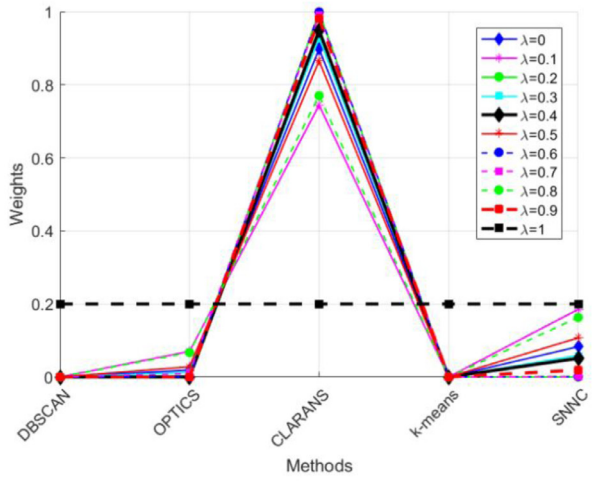
The objective function of the algorithm is the mean square distance (the Euclidean metric) between the object samples and the



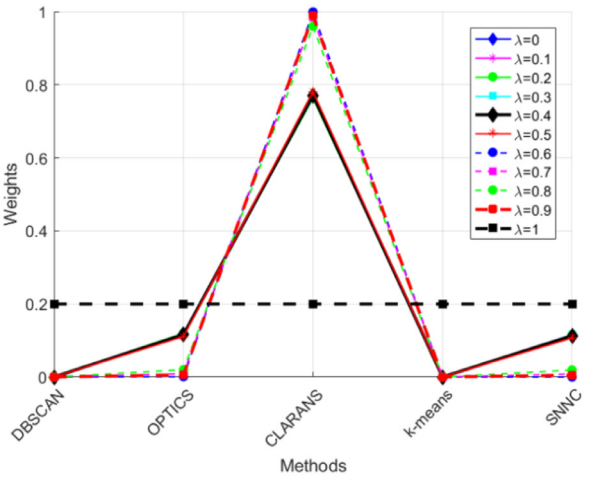
a) *Diabetic dataset*



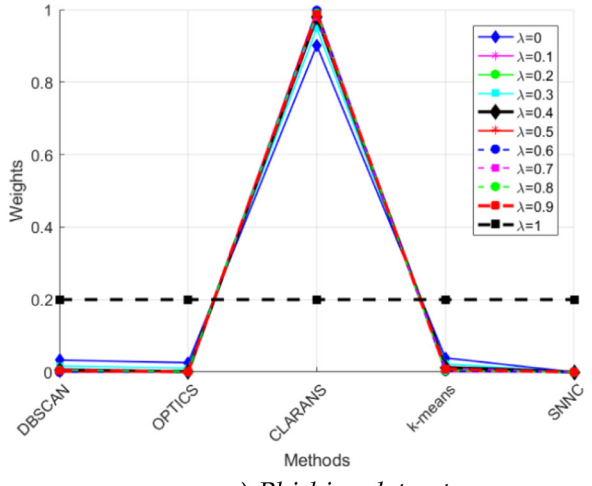
b) *Magic04 dataset*



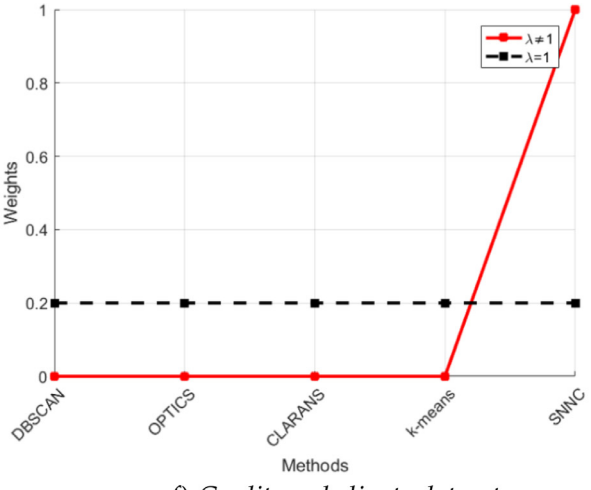
c) *NSL-KDD_All dataset*



d) *Banknote authentication dataset*

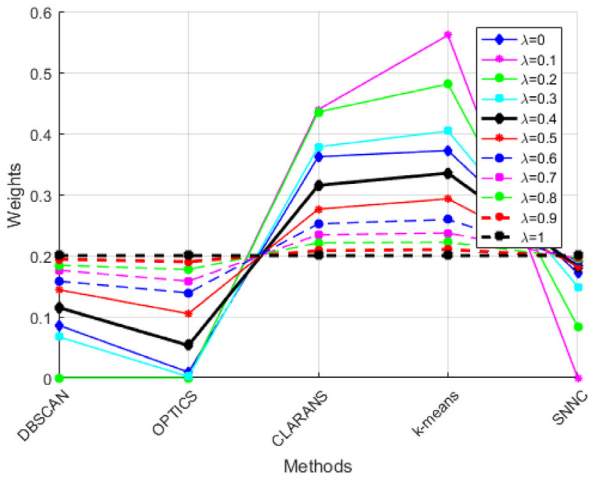


e) *Phishing dataset*

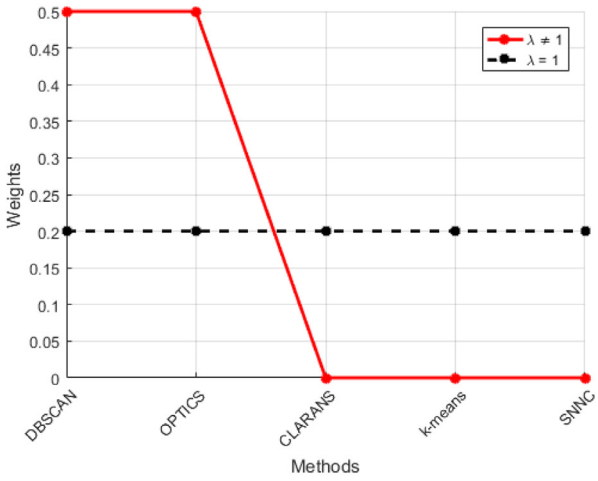


f) *Credit card clients dataset*

Fig. 2. The dependence of weights of single clustering methods on the parameter λ.



g) *Phone accelerometer dataset*



h) *Covertype dataset*

Fig. 2. Continued

centers of their clusters:

$$f(X,C)=\sum_{j=1}^{k}\sum_{i=1}^{n}\left\|x_i-O_j\right\|^2 \tag{14}$$

where O_j is the center of the cluster C_j, calculated by Eq. (15)

$$O_j=\frac{1}{|C_j|}\sum_{i=1}^{n}x_i \tag{15}$$

4.5. SNNC

The SNNC (Shared Nearest Neighbor Clustering) algorithm was proposed by Jarvis and Patrick, where a link is created between a pair of points p and q if and only if p and q have each other in their closest k-nearest neighbor (Ertoz, Steinbach & Kumar, 2002; Jarvis & Patrick, 1973). This algorithm is an extension of the DBSCAN.

The basic idea of SNNC is based on determining the core points around which clusters of various sizes and shapes are built, without worrying about determining their number (Malchiodi, Bassis & Valerio, 2008). Counting the number of points shared between two points p and q in their k-nearest neighbor list based on the distance metric allows us to determine the similarity between them. The greater the number of shared points, the higher similarity between p and q.

5. Proposed approach

Let us denote the following notations: $X=\{x_1,x_2,\ldots,x_n\}$ are the points in the dataset, where n is the total number of data points in the dataset. $x_i=\{x_{i1},x_{i2},\ldots,x_{im}\}\in R^m$ is the point in the dataset, where m is the dimension of data points. The partition of X into K crisp clusters is represented as a collection of K subsets of objects in $C=\{C_k|k=1,\ldots,K\}$ with $C_k\cap C_{k'}=\emptyset\ \forall k\neq k'$ and $\bigcup_{k=1}^{K}C_k=X$ or as a vector of labels $\pi=(L_\pi(x_1),L_\pi(x_2),\ldots,L_\pi(x_n))^T$, for any i $x_i\xrightarrow{L_\pi}\{1,2,\ldots,K\}$.

Existing consensus clustering methods can be classified, as a rule, into two categories, that is, methods with global objective functions and without them (Vega-Pons & Ruiz-Shulcloper, 2011). In this paper, we mainly focus on the previous methods, which are usually formulated as combinatorial optimization problems as follows. Given the r basic partitions from X to $\Pi=\{\pi_1,\pi_2,\ldots,\pi_r\}$ (a basic partition is the result produced by a single clustering algorithm (for example, DBSCAN, OPTICS, etc.). And there are K_i-clusters in π_i, for $1\le i\le r$. The goal is to find a consensus partition π by solving the following optimization task:

$$\pi^*=\underset{\pi}{\operatorname{argmax}}\ (w_iU(\pi,\pi_i)) \tag{16}$$

where π^* is a consensus function, U is a utility function, which measures the similarity between π and any π_i, π_i is the basic partition, and $w_i\in[0,1]$, $\sum_{i=1}^{r}w_i=1$.

In other words, we expect to find the optimal partition, which is the most consistent with the basic partition. Different utility functions measure the similarity of two partitions in different aspects, providing different objective functions for consensus clustering. The proposed method uses a utility function based on purity to aggregate all basic partitions into a consensual one, which makes decisions by general agreement:

$$\text{maximize } f=(1-\lambda)\cdot\sum_{i=1}^{r}w_iU(\pi,\pi_i)+\lambda\cdot\|w\|^2 \tag{17}$$

subject to

$$\sum_{i=1}^{r}w_i=1,\ w_i\ge 0,\ \forall i \tag{18}$$

Where $0\le\lambda\le 1$ is the regularization parameter, which specifies the trade-off between the maximization of the weighted utility function and the smoothness enforced by w. In our experiments, λ will be determined experimentally. We use the Euclidean distance, the Minkowski distance for $p=3$ and $p=4$, squared Euclidean distance, cosine distance, and Chebychev distance.

The first term in Eq. (17) is used to minimize the weighted distance between individual partitions and π. The second term in Eq. (18) is a regularization term for ensuring the smoothness of the weights.

The goal of the method is to combine these basic partitions into PWCC method, which ensures that the consensus clustering algorithm will be highly efficient and reliable. As formulated in Eq. (16), the utility function is defined on two partitions π and π_i to measure their similarity at the partition level. To calculate the purity utility function, we can use the following association table (Table 2).

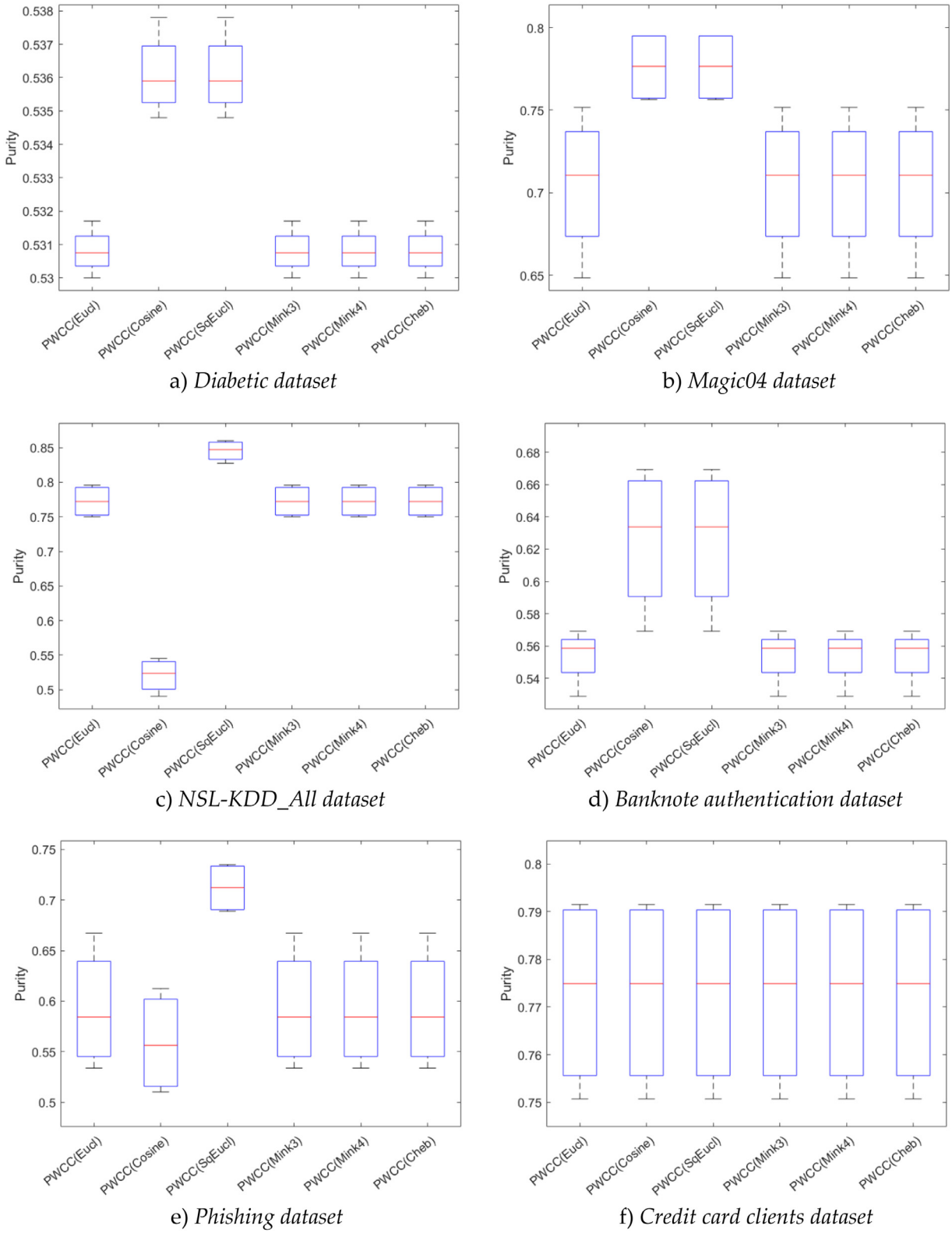
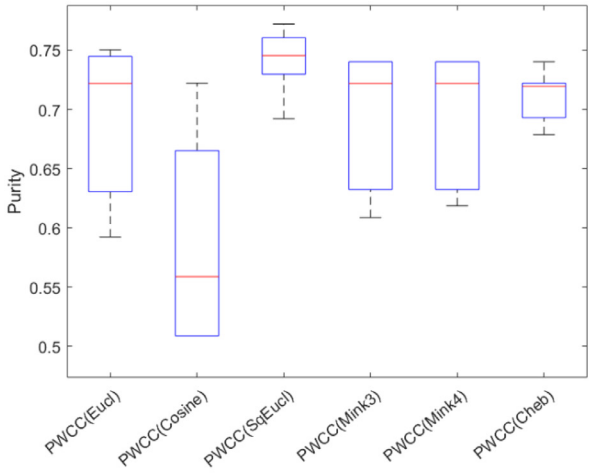


Fig. 3. Boxplot diagram of the performance evaluation for all considered datasets.

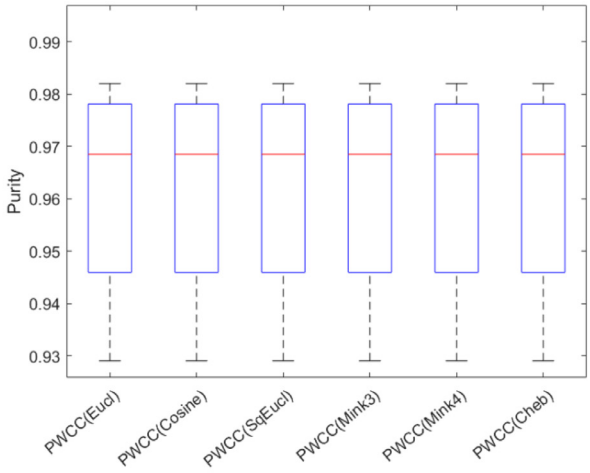
Here π contains K clusters and π_i - K_i clusters, $n_{kj}^{(i)}$ denotes the number of points contained by both cluster $C_j^{(i)}$ in π_i and cluster C_k in π, $n_{k+} = \sum_{j=1}^{K_i} n_{kj}^{(i)}$ is the number of points in C_k, $n_{+j}^{(i)} = \sum_{k=1}^{K} n_{kj}^{(i)}$ is the number of points in $C_j^{(i)}$, n is the total number of points.

Fig. 1 shows an example of a co-association matrix for a data set of 10 points. It illustrates one basic partition π_1, highlighted in green, and true clustering, marked in red.

Let $m_{jk}^{(i)} = (\max n_{j1}^{(i)}, \max n_{j2}^{(i)}, \ldots, \max n_{jK_i}^{(i)})$ and $m_k^{(i)} = n_{+k}^{(i)}$, then we can define the utility function to measure the similarity between the two partitions presented in Table 2. In this paper, the



g) *Phone accelerometer dataset*



h) *Covertype dataset*

Fig. 3. Continued

Table 3
Summary of the datasets.

Dataset	Number of instances	C_1^-	C_1^+	Number of attributes
Diabetic	1151	611	540	19
Phishing	11,055	4898	6157	30
NSL-KDD_All	148,517	71,463	77,054	41
Banknote authentication	1372	762	610	4
Magic04	19,020	12,332	6688	10
Credit card clients	30,000	23,364	6636	23
Phone Accelerometer	13,062,475	6240,983	6821,492	6
Covertype	581,012	20,510	56,0502	54

purity utility function has the following form:

$$U(\pi, \pi_i) = \sum_{k=1}^{K_i} \frac{n_{+k}^{(i)}}{n} \max_j n_{jk}^{(i)} = \sum_{k=1}^{K_i} \frac{m_k}{n} m_{jk}^{(i)} \tag{19}$$

Suppose that there are r basic partitions. The task is to find a weighted consensus π with a set of $\{w_1, w_2, \ldots, w_r\}$ weights assigned to each method.

We initialize $w_i = \frac{1}{r}$, $\forall i$. The weights w_i are obtained in Matlab 2016a using Optimization Toolbox to solve the optimization problem Eq. (17)–(18).

6. Experimental results

The number of experiments was conducted to evaluate the productivity of the proposed approach. The experiments were carried out using Windows® 10–64 bits operating system platform with core i7 processor 2.5 GHz, 8.0GB RAM. The proposed approach was evaluated on R 3.4.1.

Eight datasets from the UCI repository (Eggermont, Kok & Kosters, 2004; Lichman, 2013), including Banknote authentication, Phishing, Diabetic, Magic04, Credit card clients, Phone accelerometer, Covertype, and the NSL-KDD dataset (Aggarwal & Sharma, 2015) were used as input data. For the experiments, the training and test sets in the last data set were combined (148,517 samples) into the NSL-KDD_All dataset. These datasets have a small, medium, and large size. The characteristics of the datasets are presented in Table 3.

All datasets contain two classes: C_1^+ and C_2^+. Samples in class C_1^+ are considered as anomalies. The values in the datasets were standardized during the preprocessing.

The experiments were focused on comparing the clustering results of the proposed approach with different distance metrics (Euclidean distance (Eucl), cosine distance (Cos), squared Euclidean (SqEucl) distance, Minkowski distance when $p = 3$ (Mink3), Minkowski distance when $p = 4$ (Mink4) and Chebychev (Cheb) distance) with single clustering methods (DBSCAN (Ester et al., 1996), OPTICS (Ankerst et al., 1999), CLARANS (Ng & Han, 1994), k-means and SNNC (Shaneck et al., 2009)).

The influence of the regularization parameter λ on the performance of clustering algorithms on different datasets was considered. We used the values $\lambda = 0.1$, $\lambda = 0.2$, $\lambda = 0.3$, $\lambda = 0.4$, $\lambda = 0.5$, $\lambda = 0.6$, $\lambda = 0.7$, $\lambda = 0.8$, $\lambda = 0.9$, and $\lambda = 1.0$. The initial values of the weights w_i for all basic partitions are taken to be equal to $w_i = \frac{1}{5} = 0.2$, $i = 1, \ldots, 5$.

Fig. 2 shows the dependence of single clustering methods weights on the parameter λ for all datasets. In Fig. 2(a), you can see the influence of the parameter λ on the weights of five clustering methods for the Diabetic dataset. The weights of the SNNC method at $\lambda = 0.7$, $\lambda = 0.8$, and $\lambda = 0.9$ significantly differ from $\lambda = 0.0 \div 0.6$ and are equal to 0.025, 0.122, and 0.326, respectively. The best result for this method was observed at $\lambda = 0.9$, and for the whole dataset - for the OPTICS method at $\lambda = 0.6$.

According to Fig. 2(b), the OPTICS method receives weights of ~0.98 for the Magic04 dataset when $\lambda = 0.0 \div 0.9$. For this dataset, DBSCAN, CLARANS, k-means, and SNNC showed the lowest weights.

Fig. 2(c) shows that the weights of the DBSCAN, OPTICS, and k-means methods are smaller than the weights of the CLARANS and SNNC methods for the NSL-KDD_All dataset. In this case, the best result for $\lambda = 0.2$, $\lambda = 0.6$, $\lambda = 0.7$, and $\lambda = 0.9$ showed the CLARANS method. And the DBSCAN and k-means methods were ineffective.

For the Banknote authentication dataset (Fig. 2(d)), the values of all five methods practically coincide in two cases: 1) at $\lambda = 0$,

Table 4
Comparison of weights of single clustering methods for $\lambda = 0.6$.

Dataset	DBSCAN	OPTICS	CLARANS	k-means	SNNC
Diabetic	0.200	0.204	0.197	0.197	0.202
Phishing	0.178	0.182	0.258	0.174	0.208
NSL-KDD_All	0.202	0.197	0.225	0.201	0.175
Banknote authentication	0.199	0.185	0.217	0.213	0.186
Magic04	0.221	0.239	0.181	0.184	0.175
Credit card clients	0	0	0	0	1
Phone Accelerometer	0.161	0.125	0.263	0.262	0.189
Covertype	0.500	0.500	0	0	0
a) *Euclidean distance metric*					
Diabetic	1	0	0	0	0
Phishing	0	0	0	0	1
NSL-KDD_All	0	0	0	1	0
Banknote authentication	0	0	1	0	0
Magic04	0	1	0	0	0
Credit card clients	0	0	0	0	1
Phone Accelerometer	0.200	0.203	0.198	0.197	0.202
Covertype	0.500	0.500	0	0	0
b) *Cosine distance metric*					
Diabetic	0	0.984	0.008	0.008	0
Phishing	0	0	0.998	0.002	0
NSL-KDD_All	0	0	0.999	0	0.001
Banknote authentication	0	0.001	0.999	0	0
Magic04	0	0.986	0.002	0.001	0.011
Credit card clients	0	0	0	0	1
Phone Accelerometer	0	0	0.195	0.761	0.045
Covertype	0.500	0.500	0	0	0
c) *Squared Euclidean distance metric*					
Diabetic	0.199	0.205	0.197	0.197	0.202
Phishing	0.180	0.187	0.250	0.179	0.204
NSL-KDD_All	0.201	0.199	0.222	0.200	0.178
Banknote authentication	0.200	0.188	0.212	0.211	0.189
Magic04	0.217	0.227	0.19	0.189	0.177
Credit card clients	0	0	0	0	1
Phone Accelerometer	0.160	0.134	0.253	0.263	0.190
Covertype	0.500	0.500	0	0	0
d) *Minkowski distance metric ($p = 3$)*					
Diabetic	0.199	0.203	0.198	0.198	0.202
Phishing	0.180	0.190	0.247	0.182	0.201
NSL-KDD_All	0.200	0.200	0.220	0.200	0.180
Banknote authentication	0.201	0.190	0.208	0.211	0.190
Magic04	0.215	0.220	0.195	0.191	0.179
Credit card clients	0	0	0	0	1
Phone Accelerometer	0.167	0.137	0.248	0.255	0.193
Covertype	0.500	0.500	0	0	0
e) *Minkowski distance metric ($p = 4$)*					
Diabetic	0.194	0.206	0.198	0.198	0.204
Phishing	0.181	0.204	0.236	0.193	0.186
NSL-KDD_All	0.185	0.224	0.214	0.184	0.193
Banknote authentication	0.205	0.197	0.194	0.211	0.193
Magic04	0.212	0.211	0.202	0.195	0.180
Credit card clients	0	0	0	0	1
Phone Accelerometer	0.201	0.164	0.213	0.214	0.208
Covertype	0.500	0.500	0	0	0
f) *Chebychev distance metric*					

$\lambda = 0.1$, $\lambda = 0.2$, $\lambda = 0.3$, $\lambda = 0.4$, and $\lambda = 0.5$, and 2) at $0.6 \leq \lambda \leq 0.9$. The CLARANS method showed the best result, and its weight was 0.999 at $\lambda = 0.6$.

For the Phishing dataset (Fig. 2(e)), the weights of the five clustering methods for different values of λ are fairly close, except at $\lambda = 0$ and $\lambda = 0.2$. In this case, the CLARANS method has the greatest weight. For the Credit card clients dataset (Fig. 2(f)), the best result for all $\lambda \neq 1$ showed the SNNC algorithm, and all other methods turned out to be ineffective.

According to Fig. 2(g), the largest weights are achieved by the CLARANS and k-means methods with values of $\lambda > 1$. In this case, the largest weight for the Phone accelerometer dataset is obtained for k-means with $\lambda = 0.1$.

Considering the Covertype dataset (Fig. 2(h)), the DBSCAN and OPTICS methods showed the best results. And the CLARANS, k-means, and SNNC methods were ineffective.

Thus, summarizing the above, the CLARANS method showed the best results for NSL-KDD_All, Banknote authentication and

Table 5
Comparison of the performance of the proposed approach with single clustering methods for the Diabetic dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.5308	0.4999	0.5517	0.1932	0.2509
OPTICS	**0.5361**	**0.4974**	0.5358	0.1955	**0.2524**
CLARANS	0.5308	0.4982	**0.5816**	**0.1884**	0.2511
k-means	0.5308	0.4982	**0.5816**	**0.1884**	0.2511
SNNC	0.5352	0.4975	0.5348	0.1957	0.2521
PWCC(Eucl)	0.5308	0.4982	**0.5816**	**0.1884**	0.2511
PWCC(Cosine)	**0.5361**	**0.4974**	0.5358	0.1955	**0.2524**
PWCC(SqEucl)	**0.5361**	**0.4974**	0.5358	0.1955	**0.2524**
PWCC(Mink3)	0.5308	0.4982	**0.5816**	**0.1884**	0.2511
PWCC(Mink4)	0.5308	0.4982	**0.5816**	**0.1884**	0.2511
PWCC(Cheb)	0.5308	0.4982	**0.5816**	**0.1884**	0.2511

Table 6
Comparison of the performance of the proposed approach with single clustering methods for the Banknote authentication dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.5554	0.4995	0.6006	**0.1615**	0.2567
OPTICS	0.5554	0.4946	0.5750	0.1850	0.2536
CLARANS	**0.6239**	0.4694	0.6209	0.1825	**0.2675**
k-means	0.6122	0.4748	**0.6219**	0.1780	0.2615
SNNC	0.5554	0.4978	0.5671	0.1863	0.2529
PWCC(Eucl)	0.5554	0.4991	0.5199	0.1907	0.2533
PWCC(Cosine)	**0.6239**	**0.4693**	0.6209	0.1825	**0.2675**
PWCC(SqEucl)	**0.6239**	**0.4693**	0.6209	0.1825	**0.2675**
PWCC(Mink3)	0.5554	0.4991	0.5199	0.1907	0.2533
PWCC(Mink4)	0.5554	0.4991	0.5199	0.1907	0.2533
PWCC(Cheb)	0.5554	0.4991	0.5199	0.1907	0.2533

Table 7
Comparison of the performance of the proposed approach with single clustering methods for the NSL-KDD_All dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.5134	0.4987	0.6716	0.0587	0.3753
OPTICS	0.6400	0.4608	0.5489	0.0868	0.3148
CLARANS	**0.8489**	**0.2565**	**0.8533**	**0.0563**	**0.4111**
k-means	0.5188	0.4993	0.6832	0.0582	0.3752
SNNC	0.5470	0.4956	0.6018	0.0924	0.2553
PWCC(Eucl)	0.7552	0.3697	0.7734	0.0690	0.3897
PWCC(Cosine)	0.5188	0.4993	0.6832	0.0582	0.3752
PWCC(SqEucl)	**0.8489**	**0.2565**	**0.8533**	**0.0563**	**0.4111**
PWCC(Mink3)	0.7552	0.3697	0.7734	0.0690	0.3897
PWCC(Mink4)	0.7552	0.3697	0.7734	0.0690	0.3897
PWCC(Cheb)	0.7552	0.3697	0.7734	0.0690	0.3897

Table 8
Comparison of the performance of the proposed approach with single clustering methods for the Phishing dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.5569	0.4987	0.5312	0.1468	0.2549
OPTICS	0.5569	0.4987	0.5309	0.1468	0.2549
CLARANS	**0.7166**	**0.4062**	**0.7283**	0.1189	**0.3053**
k-means	0.5569	0.4957	0.6550	0.1229	0.2524
SNNC	0.5569	0.4988	0.6739	0.1016	0.2745
PWCC(Eucl)	0.5905	0.4836	0.7174	**0.0948**	0.2927
PWCC(Cosine)	0.5569	0.4988	0.6739	0.1016	0.2745
PWCC(SqEucl)	**0.7166**	**0.4062**	**0.7283**	0.1189	**0.3053**
PWCC(Mink3)	0.5905	0.4836	0.7174	**0.0948**	0.2927
PWCC(Mink4)	0.5905	0.4836	0.7174	**0.0948**	0.2927
PWCC(Cheb)	0.5905	0.4836	0.7174	**0.0948**	0.2927

Table 9
Comparison of the performance of the proposed approach with single clustering methods for the Magic04 dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.7531	0.3719	0.7675	0.1028	0.3093
OPTICS	**0.7740**	**0.3498**	**0.7963**	0.0906	**0.3414**
CLARANS	0.6484	0.4577	0.6439	0.1272	0.2769
k-means	0.6484	0.4833	0.5842	0.1329	0.2741
SNNC	0.6569	0.4507	0.7724	**0.0837**	0.2671
PWCC(Eucl)	0.7053	0.4157	0.7694	0.0912	0.3058
PWCC(Cosine)	**0.7740**	**0.3498**	**0.7963**	0.0906	**0.3414**
PWCC(SqEucl)	**0.7740**	**0.3498**	**0.7963**	0.0906	**0.3414**
PWCC(Mink3)	0.7053	0.4157	0.7694	0.0912	0.3058
PWCC(Mink4)	0.7053	0.4157	0.7694	0.0912	0.3058
PWCC(Cheb)	0.7053	0.4157	0.7694	0.0912	0.3058

Table 10
Comparison of the performance of the proposed approach with single clustering methods for the Credit card clients dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.7788	0.4889	0.5410	0.1162	0.3261
OPTICS	0.7788	0.4891	0.5405	0.1162	0.3261
CLARANS	0.7788	0.4890	0.5407	0.1162	0.3261
k-means	0.7788	0.4890	0.5407	0.1162	0.3261
SNNC	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**
PWCC(Eucl)	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**
PWCC(Cosine)	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**
PWCC(SqEucl)	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**
PWCC(Mink3)	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**
PWCC(Mink4)	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**
PWCC(Cheb)	0.7788	**0.4135**	**0.7893**	**0.0783**	**0.3722**

Table 11
Comparison of the performance of the proposed approach with single clustering methods for the Phone Accelerometer dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.5136	0.4999	0.6611	**0.0600**	0.2583
OPTICS	0.5089	0.4998	0.5348	0.0904	0.2501
CLARANS	0.7393	0.3858	**0.7499**	0.0660	0.3435
k-means	**0.7403**	**0.3845**	0.7257	0.0619	**0.3613**
SNNC	0.6325	0.4649	0.6257	0.0856	0.2687
PWCC(Eucl)	0.6932	0.4191	0.6829	0.0732	0.3168
PWCC(Cosine)	0.5872	0.4693	0.5939	0.0856	0.2671
PWCC(SqEucl)	**0.7403**	**0.3845**	0.7257	0.0620	**0.3613**
PWCC(Mink3)	0.6891	0.4206	0.6826	0.0741	0.3121
PWCC(Mink4)	0.6932	0.4206	0.6826	0.0741	0.3121
PWCC(Cheb)	0.7109	0.4043	0.7050	0.0704	0.3263

Table 12
Comparison of the performance of the proposed approach with single clustering methods for the Covertype dataset.

Method	Purity	Mirkin	F-measure	VI	PC
DBSCAN	0.9647	**0.0681**	**0.9820**	**0.0115**	0.2330
OPTICS	0.9647	**0.0681**	**0.9820**	**0.0115**	0.2330
CLARANS	0.9647	0.4610	0.4894	0.0605	**0.4632**
k-means	0.9647	0.4315	0.4489	0.0580	0.4625
SNNC	0.9647	0.1230	0.9320	0.2380	0.4303
PWCC(Eucl)	0.9647	0.1972	0.8449	0.0298	0.3968
PWCC(Cosine)	0.9647	0.0722	0.9780	0.0127	0.3021
PWCC(SqEucl)	0.9647	0.0788	**0.9820**	**0.0115**	0.4058
PWCC(Mink3)	0.9647	0.2405	0.7851	0.0342	0.4323
PWCC(Mink4)	0.9647	0.1530	0.8871	0.0237	0.4194
PWCC(Cheb)	0.9647	0.2206	0.8139	0.0316	0.3940

Phishing datasets, OPTICS for Diabetic and Magic04 datasets, and SNNC for Credit card clients dataset. And the result reached 100%. $w_i = 0.2$ ($i = 1, \ldots, 5$) is the optimal solution for $\lambda = 1$, which was confirmed from experiments.

Thus, the best results for the Diabetic, Phishing, NSL-KDD_All, Banknote authentication, Magic04, Credit card clients, and Covertype datasets were obtained at $\lambda = 0.6$, and for the Phone Accelerometer dataset at $\lambda = 0.1$. A set of weights obtained with the proposed approach for all datasets is shown in Table 4. It can be seen that the values of the weights are comparable to the results of the proposed approach.

Table 13
The resultant rank of the clustering methods.

Method	The number of times the method is in the sth rank s =							Resultant rank
	1	2	3	4	5	6	7	
PWCC (SqEucl)	24	2	4	0	0	0	0	28.5714
PWCC (Cosine)	15	4	7	1	2	1	0	25.1429
CLARANS	14	4	7	1	4	0	0	24.7143
PWCC (Eucl)	8	8	9	1	1	3	0	23.1429
PWCC (Mink3)	8	8	9	1	1	3	0	23.1429
PWCC (Mink4)	8	8	9	1	1	3	0	23.1429
PWCC (Cheb)	8	8	9	1	1	3	0	23.1429
OPTICS	8	3	7	7	3	2	0	21.4286
DBSCAN	2	9	7	7	3	1	1	20.4286
k-means	4	6	8	4	4	4	0	20.0000
SNNC	6	5	4	6	5	4	0	19.8571

Table 14
P values produced by Wilcoxon's rank sum test by comparing PWCC(SqEucl) with other methods.

Dataset	PWCC(Eucl)	PWCC(Cosine)	PWCC(Mink3)	PWCC(Mink4)	PWCC(Cheb)
Diabetic	0.0286	1	0.0286	0.0286	0.0286
Phishing	0.0286	1	0.0286	0.0286	0.0286
NSL-KDD_All	0.0286	0.0286	0.0286	0.0286	0.0286
Banknote authentication	0.0571	1	0.0571	0.0571	0.0571
Magic04	0.0286	0.0286	0.0286	0.0286	0.0286
Credit card clients	1	1	1	1	1
Phone Accelerometer	0.0261	1.5114e-04	0.0076	0.0076	0.0024
Covertype	1	1	1	1	1

The maximum weights were assigned to the DBSCAN and OPTICS methods when applying the Euclidean distance metric to the proposed approach for the Covertype dataset, and for the Credit card clients' dataset, the SNNC method (Table 4 (a)) received the largest weight.

The application of the cosine distance metric allowed determining precisely the best approach for the Banknote authentication, Magic04, Covertype, and Credit card clients' datasets, according to Table 4(b) and Fig. 2. Table 4(c) showed almost 100% determination of the best approaches on all datasets for the proposed approach with the squared Euclidean distance metric. According to Table 4(d) and (e), the CLARANS method received the largest weight for the Phishing, Banknote authentication, Phone Accelerometer, and NSL-KDD_All datasets when using the Mink3 and Mink4 distance metrics. According to Chebychev distance metric (Table 4(f)), the best result for the Phishing dataset showed the CLARANS method, and for the Diabetic dataset – the OPTICS method, and for the Phone Accelerometer dataset – k-means and CLARANS.

The results of the proposed approach based on the metrics purity, Mirkin, PC, VI, and F-measure are presented in Tables 5-12. The best results in the tables were marked in bold.

The comparison of the proposed approach with single clustering methods (Table 5) for Diabetic dataset showed that the first one when considering the cosine distance metric, showed the best results for the largest number of metrics (Purity, Mirkin, and PC), which coincided with the result of OPTICS method.

And for squared Euclidean distance, the best result was achieved only for two metrics: Mirkin and PC. The highest result according to the F-measure and VI metrics showed CLARANS and k-means, which coincided with the indicators of the proposed approach for Mink3, Mink4 and Chebychev distance metrics.

Based on the results of the experiments, the proposed consensus approach with two distance metrics (cosine and squared Euclidean distance metrics) showed the best indicators according to Purity (0.6239), Mirkin (0.4693), and PC (0.2675) for the Banknote authentication dataset (Table 6), but gave a worse result than k-means and DBSCAN methods using the F-measure and VI metrics, respectively.

From Table 7, it can be concluded that the proposed approach with the squared Euclidean distance gave the best result according to all five metrics and coincided with the result of the CLARANS method for NSL-KDD_All.

Analysing Table 8 for the Phishing dataset, it can be concluded that the proposed consensus approach with squared Euclidean distance gave the best result for the Purity (0.7166), Mirkin (0.4062), F-measure (0.7283) and PC (0.3053). Despite this indicator, the Euclidean, Mink3, Mink4, and Chebychev distances, the results of the basic partitions surpassed by VI and amounted to 0.0948.

For the Magic04 dataset (Table 9), the proposed approach with cosine and squared Euclidean distances showed the best results for four metrics (Purity, Mirkin, F-measure, and PC), but gave way, according to VI, to the SNNC method.

Comparing the results in Table 10, Purity gave the same results for all methods and amounted to 0.7788 for the Credit card clients' dataset. The proposed approach showed an excellent result for all distance metrics.

Research showed that the proposed method reveals more anomalies in clusters than each single clustering method.

For the Phone Accelerometer dataset (Table 11), the proposed approach outperformed such single clustering methods as DBSCAN and OPTICS according to Purity, Mirkin, F-measure, and PC, but slightly conceded according to VI.

According to Table 12, the proposed approach using the squared Euclidean distance metric gave the best result for a Covertype dataset according to Purity, F-measure, and VI. It slightly conceded DBSCAN and OPTICS, according to Mirkin. PC metric showed that a lower score was obtained for CLARANS.

According to the ranks obtained for the Purity, Mirkin, F-measure, VI, and PC metrics, the resultant rank was calculated (Davies & Bouldin, 1979) for all clustering methods (single clustering methods and the proposed consensus approach) (Table 13).

From Table 13, the proposed approach using the squared Euclidean metric showed the best result for all data sets according

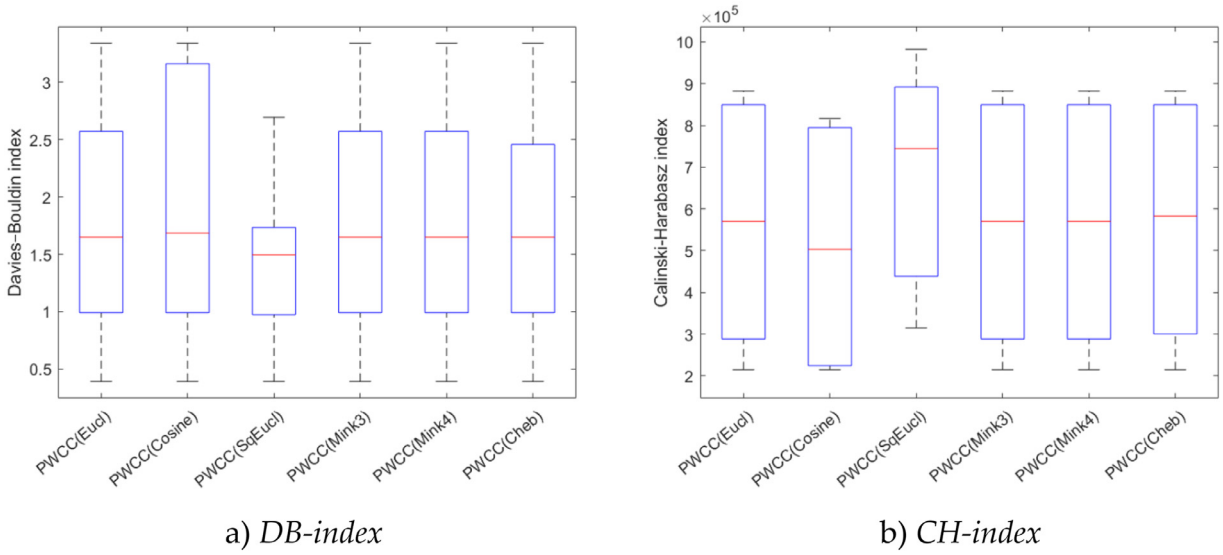


Fig. 4. Boxplot diagram of the performance evaluation for the Phone Accelerometer dataset.

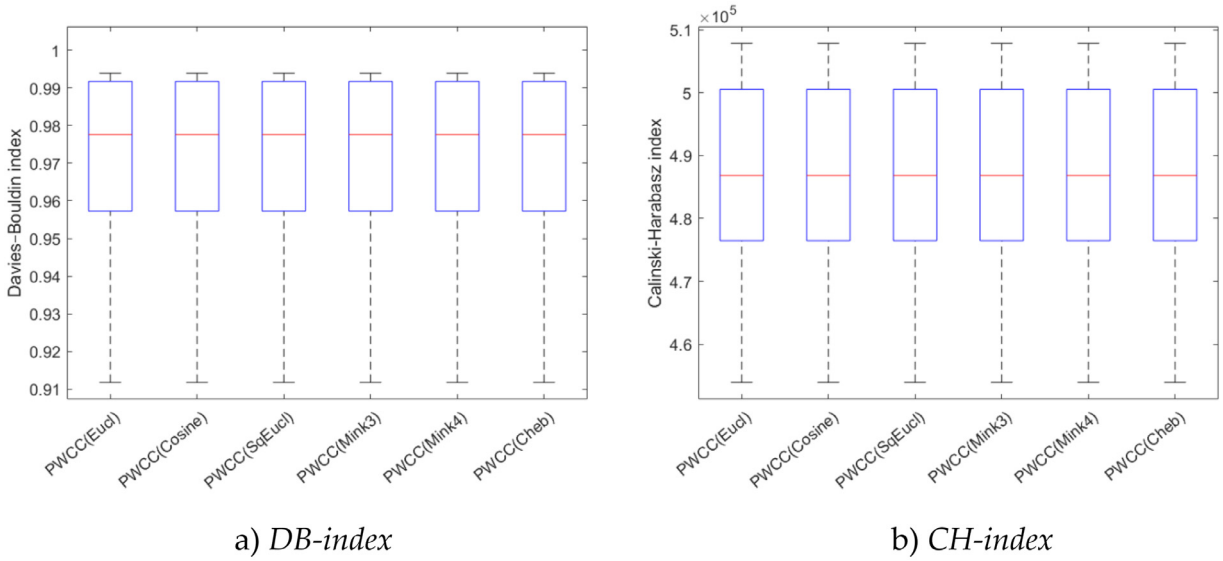


Fig. 5. Boxplot diagram of the performance evaluation for the Covertype dataset.

to three metrics (Purity, Mirkin, and PC), and the worst result was shown by the SNNC method according to the Purity.

The proposed approach with the cosine metric has the second rank according to Purity, Mirkin, and PC when considering all data sets in general, even though it received unsatisfactory results for NSL-KDD_All and Phishing datasets.

From Tables 5-13 we can conclude:

(1) The selection of the distance metric significantly affects the quality of consensus clustering.
(2) The best results among the clustering methods showed the proposed purity-based consensus clustering approach**.**
(3) The method using Euclidean, Minkowsky ($p = 3$ and $p = 4$), and Chebychev distance metrics have the same ranks (Table 13) equal to 23.1429.
(4) The proposed approach based on squared Euclidean and cosine metrics outperforms single clustering methods.

7. Statistical significance test

7.1. Statistical analysis of proposed approach

To evaluate the statistical significance of the summarization results, a statistical significance test, known as the Wilcoxon's rank-sum test for independent samples (Hollander, Wolfe & Chicken, 2015), was conducted at the 5% significance level. Five groups, corresponding to the five methods (PWCC(Eucl), PWCC(Cos), PWCC(Mink3), PWCC(Mink4) and PWCC(Cheb)), have been created for each dataset. Two groups are compared at a time one corresponding to PWCC(SqEucl) method and the other five methods. Each group consists of the Purity values for the datasets produced by ten consecutive runs of the corresponding method.

To establish that this goodness is statistically significant, Table 14 reports the P values produced by Wilcoxon's rank-sum test for comparison of two groups at a time. As a null hypothesis, it is assumed that there are no significant differences between

the median values of two groups, whereas the alternative hypothesis is that there is a significant difference in the median values of the two groups (Alguliev, Aliguliyev & Mehdiyev, 2011). It is clear from the table that P values are much less than 0.05 (5% significance level) for the large Phone accelerometer dataset.

It indicates that the best median values obtained by PWCC(SqEucl) are statistically significant and did not occur by chance.

The boxplot diagrams are given below for a visual comparison of the performances of the considered methods (Fig. 3). It can be seen that the maximum accuracy and the highest minimum accuracy are observed for PWCC(SqEucl). And for the Diabetic, Magic04, and Banknote authentication datasets, the PWCC(SqEucl) method also showed high results.

7.2. Statistical analysis of unlabeled Big data clustering

In this section, we demonstrate the ability to apply weighted consensus clustering to unlabeled big data analysis. To this end, the purity-based utility function was replaced with a DB index based one. In this case, the task is to minimize the objective function (Eq. (17)) to find the optimal consensus partition.

The superiority of the proposed method using the squared Euclidean metric can be seen from Figs. 4 and 5, which presents the results of the proposed approach evaluation based on the DB index and the CH index.

The proposed model also surpassed the proposed method when using other distance metrics for more than 50% of the results for the Phone Accelerometer dataset (Fig. 4). The worst performance, according to DB and CH indices (Fig. 4), is observed for the proposed method with the cosine metric.

According to Fig. 5, the proposed approach showed similar results for all considered distance metrics. The above analysis allows us to conclude that the proposed method provides high accuracy for big data clustering.

8. Conclusions and future work

The emergence of the Big data area has led to the widespread development of supervised ensemble methods. However, the increasing attention of researchers is now focused on the development of unsupervised methods due to the need to analyze large amounts of data without predetermined class labels.

This paper reviewed several studies on the development of consensus clustering methods. It is one of the state-of-the-art areas for knowledge discovery issues addressing Big data.

In this paper, a weighted consensus clustering for efficient integration of single clustering methods was proposed. The purpose of this paper was to show that weighting improves the solution for clustering large datasets. The comparison was made using eight datasets containing anomalous values. The results obtained by the DBSCAN, OPTICS, CLARANS, k-means and SNNC algorithms are used when forming the final solution. The quality of the clustering result was estimated using five metrics (Purity, Mirkin, PC, VI, and F-measure).

Based on the experimental results, the following conclusions can be drawn:

- The experimental results showed that the proposed algorithm more accurately detects anomalies compared to single clustering methods. PWCC using Euclidean, Minkowski ($p = 3$ and $p = 4$), squared Euclidean, cosine, and Chebychev distance metrics was compared to single clustering methods. The best partition was determined when applying the proposed algorithm with the squared Euclidean distance metric to the considered datasets. Based on the experimental results, it can be concluded that the PWCC compensates for the shortcomings of each considered method and increases the efficiency of clustering.
- The proposed approach is designed for any number of alternatives and can be implemented in any architecture. It can be applied by experts when making a group decision.
- Despite this, the proposed weighted consensus depends on prior knowledge of the number of clusters and the initialization of cluster centers.

However, many unresolved problems remain in this area, which should stimulate researchers to develop new and improve existing consensus clustering approaches. In the future, we will focus on the following issues:

- Existing methods based on consensus clustering consider a fixed number of basic partitions that play an important role in making the final decision. A small number of basic partitions can make it difficult for an expert to make the correct final decision. Interactively selecting the optimal number of basic partitions will improve clustering accuracy and reduce computational costs. The relevance of the use of deep learning methods will allow for solving this issue.
- Parallelization of the proposed algorithm will speed up the clustering process, performing it on multiple threads. The desired clustering can be obtained for different pairs of clusters.
- In this paper, the number of clusters for datasets was known in advance. However, to determine the optimal number of clusters in large datasets, it is necessary to develop a new approach.
- Existing methods for initialization of cluster centers have high computational complexity. The development of a new method that determines initial centers and improves clustering accuracy is an interesting area of research.
- The interpretability of clustering results is important for extracting knowledge from the information obtained to assist the expert. The development of approaches in this direction is a promising research direction and can be useful in many applications of expert systems.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Credit authorship contribution statement

Rasim M. Alguliyev: Conceptualization, Investigation, Methodology, Validation, Writing - review & editing. **Ramiz M. Aliguliyev:** Conceptualization, Investigation, Methodology, Validation, Writing - review & editing, Visualization. **Lyudmila V. Sukhostat:** Conceptualization, Investigation, Methodology, Validation, Writing - review & editing, Visualization.

Acknowledgement

This work was supported by the Science Development Foundation under the President of the Republic of Azerbaijan – Grant № EIF-KETPL-2-2015-1(25)-56/05/1.

References

Aboubi, Y., Drias, H., & Kamel, N. (2016). BAT-CLARA: BAT-inspired algorithm for clustering large applications. *IFAC-PapersOnLine, 49*(12), 243–248. doi:10.1016/j.ifacol.2016.07.607.

Aggarwal, P., & Sharma, S. K. (2015). Analysis of KDD dataset attributes-class wise for intrusion detection. *Procedia Computer Science, 57*, 842–851. doi:10.1016/j.procs.2015.07.490.

Alguliev, R. M., Aliguliyev, R. M., & Mehdiyev, C. A. (2011). pSum-SaDE: A modified p-median problem and self-adaptive differential evolution algorithm for text summarization. Applied Computational Intelligence and Soft Computing, *2011*, 1–13. doi:10.1155/2011/351498.

Alhusain, L., & Hafez, A. M. (2017). Cluster ensemble based on random forests for genetic data. *BioData Mining, 10*(37), 1–25. doi:10.1186/s13040-017-0156-2.

Aliguliyev, R. M. (2009). Performance evaluation of density-based clustering methods. *Information Sciences, 179*(20), 3583–3602. doi:10.1016/j.ins.2009.06.012.

Ankerst, M., Breunig, M. M., Kriegel, H.-P., & Sander, J. (1999). OPTICS: Ordering points to identify the clustering structure. ACM SIGMOD Record, *28*(2), 49–60. doi:10.1145/304181.304187.

Berikov, V., & Pestunov, I. (2017). Ensemble clustering based on weighted coassociation matrices: Error bound and convergence properties. Pattern Recognition, *63*, 427–436. doi:10.1016/j.patcog.2016.10.017.

Berry, M. W., & Browne, M. (2006). *Lecture notes in data mining*. Singapore: World Scientific. doi:10.1142/6103.

Bezdek, J. C., & Pal, N. R. (1998). Some new indexes of cluster validity. *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics, 28*(3), 301–315. doi:10.1109/3477.678624.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Boutin, F., & Hascoet, M. (2004). Cluster validity indices for graph partitioning. In *Proceedings of the 8th international conference on information visualization* (pp. 376–381). IEEE. doi:10.1109/IV.2004.1320171.

Cabrerizo, F. J., Chiclana, F., Al-Hmouz, R., Morfeq, A., Balamash, A. S., & Herrera-Viedma, E. (2015). Fuzzy decision making and consensus: Challenges. Journal of Intelligent & Fuzzy Systems, *29*(3), 1109–1118. doi:10.3233/IFS-151719.

Calinski, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics, 3*(1), 1–27. doi:10.1080/03610927408827101.

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1*, (2), 224–227. doi:10.1109/TPAMI.1979.4766909.

De Oliveira, J. V., Szabo, A., & de Castro, L. N. (2017). Particle swarm clustering in clustering ensembles: Exploiting pruning and alignment free consensus. Applied Soft Computing, *55*, 141–153. doi:10.1016/j.asoc.2017.01.035.

Eggermont, J., Kok, J. N., & Kosters, W. A. (2004). Genetic programming for data classification: Partitioning the search space. In Proceedings of the 2004 ACM symposium on applied computing (pp. 1001–1005). ACM. doi:10.1145/967900.968104.

Ertoz, L., Steinbach, M., & Kumar, V. (2002). A new shared nearest neighbor clustering algorithm and its applications. In *Workshop on clustering high dimensional data and its applications at 2nd SIAM international conference on data mining* (pp. 105–115). Philadelphia: Society for Industrial and Applied Mathematics. doi:10.1007/11540007_60.

Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd international conference on knowledge discovery and data mining* (pp. 226–231). AAAI Press.

Franek, L., & Jiang, X. (2014). Ensemble clustering by means of clustering embedding in vector spaces. *Pattern Recognition, 47*(2), 833–842. doi:10.1016/j.patcog.2013.08.019.

Ghosh, J., & Acharya, A. (2011). Cluster ensembles. In *Wiley interdisciplinary reviews: Data mining and knowledge discovery: 1* (pp. 305–315). doi:10.1002/widm.32.

Hidri, M. S., Zoghlami, M. A., & Ayed, R. B. (2018). Speeding up the large-scale consensus fuzzy clustering for handling big data. *Fuzzy Sets and Systems, 348*, 50–74. doi:10.1016/j.fss.2017.11.003.

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the 22nd annual international SIGIR conference on research and development in information retrieval* (pp. 50–57). ACM. doi:10.1145/312624.312649.

Hollander, M., Wolfe, D. A., & Chicken, E. (2015). *Nonparametric statistical methods* (3rd ed.). New Jersey: John Wiley & Sons. doi:10.1002/9781119196037.

Huang, D., Lai, J. H., & Wang, C. D. (2016). Robust ensemble clustering using probability trajectories. *IEEE Transactions on Knowledge and Data Engineering, 28*(5), 1312–1326. doi:10.1109/TKDE.2015.2503753.

Huang, D., Wang, C.-. D., & Lai, J.-. H. (2018). Locally weighted ensemble clustering. *IEEE Transactions on Cybernetics, 48*(5), 1460–1473. doi:10.1109/TCYB.2017.2702343.

Jarvis, R. A., & Patrick, E. A. (1973). Clustering using a similarity measure based on shared nearest neighbors. *IEEE Transactions on Computers, C-22*(11), 1025–1034. doi:10.1109/T-C.1973.223640.

Jia, J., Liu, B., & Jiao, L. (2011). Soft spectral clustering ensemble applied to image segmentation. *Frontiers of Computer Science in China, 5*(1), 66–78. doi:10.1007/s11704-010-0161-9.

Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence, *24*(7), 881–892. doi:10.1109/TPAMI.2002.1017616.

Kashef, R., & Kamel, M. S. (2010). Cooperative clustering. *Pattern Recognition, 43*(6), 2315–2329 https://doi.org/0.1016/j.patcog.2009.12.018.

Leonard, K., & Rousseeuw, P. J. (1990). Finding groups in data: An introduction to cluster analysis. New York: John While & Sons. doi:10.1002/9780470316801.

Lichman, M. (2013). *UCI machine learning repository*. University of California http://archive.ics.uci.edu/ml/ Accessed 7 September 2018.

Liu, H., Wu, J., Liu, T., Tao, D., & Fu, Y. (2017). Spectral ensemble clustering via weighted k-means: Theoretical and practical evidence. IEEE Transactions on Knowledge and Data Engineering, *29*(5), 1129–1143. doi:10.1109/TKDE.2017.2650229.

Lock, E. F., & Dunson, D. B. (2013). Bayesian consensus clustering. *Bioinformatics, 29*(20), 2610–2616. doi:10.1093/bioinformatics/btt425.

Malchiodi, D., Bassis, S., & Valerio, L. (2008). Discovering regression data quality through clustering methods. In *New directions in neural networks - 18th Italian workshop on neural networks* (pp. 76–85). Amsterdam: IOS Press. doi:10.3233/978-1-58603-984-4-76.

Mimaroglu, S., & Yagci, M. (2012). CLICOM: Cliques for combining multiple clusterings. Expert Systems with Applications, *39*, 1889–1901. http://doi.org/10.1016/j.eswa.2011.08.059.

Mirkin, B. (1996). *Mathematical classification and clustering*. Boston: Kluwer Academic Press. doi:10.1007/978-1-4613-0457-9.

Ng, R. T., & Han, J. (1994). Efficient and effective clustering methods for spatial data mining. In *Proceedings of the 20th VLDB Conference* (pp. 144–155). San Francisco: Morgan Kaufmann Publishers Inc.

Nguyen, N., & Caruana, R. (2007). Consensus clusterings. In *Proceedings of the 7th IEEE international conference on data mining* (pp. 607–612). IEEE. 10.1109/ICDM.2007.73.

Patrikainen, A., & Meila, M. (2006). Comparing subspace clusterings. *IEEE Transactions on Knowledge and Data Engineering, 18*(7), 902–916. doi:10.1109/TKDE.2006.106.

Pérez, L. G., Mata, F., Chiclana, F., Kou, G., & Herrera-Viedma, E. (2016). Modelling influence in group decision making. *Soft Computing, 20*(4), 1653–1665. doi:10.1007/s00500-015-2002-0.

Rosenberg, A., & Hirschberg, J. (2007). V-Measure: A conditional entropy-based external cluster evaluation measure. In Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (pp. 410–420). Association for Computational Linguistics. doi:10.7916/D80V8N84.

Rubinov, A. M., Soukhorukova, N. V., & Ugon, J. (2006). Classes and clusters in data analysis. *European Journal of Operational Research, 173*(3), 849–865. doi:10.1016/j.ejor.2005.04.047.

Shaneck, M., Kim, Y., & Kumar, V. (2009). Privacy preserving nearest neighbor search. In S. Y. Philip, & J. P. T. Jeffrey (Eds.), *Machine learning in cyber trust* (pp. 247–276). New York: Springer-Verlag. doi:10.1007/978-0-387-88735-7_10.

Vega-Pons, S., & Ruiz-Shulcloper, J. (2011). A survey of clustering ensemble algorithms. *International Journal of Pattern Recognition and Artificial Intelligence, 25*(3), 337–372. doi:10.1142/S0218001411008683.

Wu, J., Wu, Z., Cao, J., Liu, H., Chen, G., & Zhang, Y. (2017). Fuzzy consensus clustering with applications on big data. *IEEE Transactions on Fuzzy Systems, 25*(6), 1430–1445. doi:10.1109/TFUZZ.2017.2742463.

Zheng, L., Li, L., Hong, W., & Li, T. (2013). PENETRATE: Personalized news recommendation using ensemble hierarchical clustering. Expert Systems with Applications, *40*(6), 2127–2136. doi:10.1016/j.eswa.2012.10.029.