Improvement of the Validity Index for Determination of an Appropriate Data Partitioning

Artur Starczewski[1](✉) and Adam Krzyżak[2,3]

[1] Institute of Computational Intelligence, Częstochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Częstochowa, Poland
artur.starczewski@iisi.pcz.pl

[2] Department of Computer Science and Software Engineering,
Concordia University, Montreal, Canada
krzyzak@cs.concordia.ca

[3] Department of Electrical Engineering, Westpomeranian University of Technology,
70-313 Szczecin, Poland

Abstract. In this paper a detail analysis of an improvement of the *Silhouette* validity index is presented. This proposed approach is based on using an additional component which improves clusters validity assessment and provides better results during a clustering process, especially when the naturally existing groups in a data set are located in very different distances. The performance of the modified index is demonstrated for several data sets, where the *Complete–linkage* method has been applied as the underlying clustering technique. The results prove superiority of the new approach as compared to other methods.

Keywords: Clustering · Cluster validity index · Complete–linkage clustering technique

1 Introduction

Clustering aims at grouping data into homogeneous subsets (called clusters), inside which elements are similar to each other and dissimilar to elements of other clusters. The purpose of clustering is to discover natural existing structures in a data set. These techniques are widely used in various fields such as pattern recognition, image processing, data exploration, etc. It should be noted that due to a large variety of data sets different clustering algorithms and their configurations are formed. Generally, among clustering methods two major categories are distinguished: partitioning and hierarchical clustering. Partitioning clustering relocates elements of a data set between clusters iteratively until a given clustering criterium is obtained. For example, the well-known algorithms of

A. Krzyżak—Carried out this research at WUT during his sabbatical leave from Concordia University.

L. Rutkowski et al. (Eds.): ICAISC 2017, Part II, LNAI 10246, pp. 159–170, 2017.
DOI: 10.1007/978-3-319-59060-8_16

this type include *K-means* and its variations [5,24] or *Expectation Maximization* (EM) [19]. On the other hand, hierarchical clustering is based on the agglomerative or divisive approach. The method known as the agglomerative hierarchical clustering starts from many clusters, which are then merged into larger ones until only one cluster has been formed. However, the divisive clustering methods start from a single cluster, which includes all elements of a data set, and then it is split into smaller clusters. For instance, well-known agglomerative hierarchical clustering methods include the *Single-linkage*, *Complete-linkage* or *Average-linkage* [16,20,25]. Nowadays, a large number of new clustering algorithms appears, e.g., [13,14]. But, for a wide variety of data sets a single clustering algorithm producing optimal data partitions does not exist. Moreover, the same clustering algorithm can also create different partition schemes of data depending on the choice of input parameters. Thus, the question asking how to find the best fit of a partition scheme to a data set is still very relevant.

The process of evaluation of partitioned data is a very difficult task and it is known as cluster validation. In this evaluation process, an estimation of the occurrence of the right clusters is very frequently realized by validity indices. In the literature on the subject, cluster validation techniques are often classified into three groups–external, internal and relative validation [16,31]. The external validation techniques are based on previous knowledge about data. On the other hand, the internal methods use only the intrinsic properties of the data set. The relative techniques compare partition schemes of a data set, which are created by changing values of input parameters of a clustering algorithm. The key parameter for many clustering methods is the number of clusters and this is most frequently changed. Next, the partitions are compared, i.e. depending on the approach used, the maximum or the minimum value of a validity index is used to determine the best fit of a partition scheme to the data set. So far, a number of authors have proposed different validity indices or modifications of existing indices, e.g., [1,11,12,15,17,18,32,36]. In the literature new interesting solutions for cluster evaluation are constantly suggested. For example, a stability index based on variation on some information measures over partitions generated by a clustering model is in [23], a new measure of distances between clusters is proposed in [30]. Papers [33,37] present indices which use the *knee-point* detection. It should be noted that cluster validity indices such as, e.g., the *Dunn* [10], *Davies-Bouldin (DB)* [8], *PBM* [21] or *Silhouette (SIL)* [26] indices are very frequently used to evaluate the efficacy of the new proposed validity approaches in detecting the right data partitioning. It is important to note that clustering algorithms in conjunction with cluster validity indices can be used during a process of designing various neural networks [2–4] and neuro-fuzzy structures [6,7,9,27–29].

In this paper, an improvement of the *Silhouette* index is described. For this purpose the new versions of this cluster validity index called the $SILA$ and $SILAv1$ have been presented. The first version of the index, i.e. $SILA$ is described in paper [34]. The next version is called $SILAv1$ and it uses an exponent defined by (9). A detailed explanation of the modifications involving the use of the component is presented in Sect. 2. In order to present effectiveness of the validity indices several experiments were performed for various data sets.

This paper is organized as follows: Sect. 2 presents a detailed description of the *Silhouette* index and an explanation of the proposed modifications of this index. Section 3 illustrates experimental results on data sets. Finally, Sect. 4 presents conclusions.

2 Improvement of the Silhouette Index

First, in this section the *Silhouette* index is described in more detail. Next, a modification of the index and an explanation of this change are presented.

2.1 The Detail Description of the Silhouette Index

Let us denote K-partition scheme of a data set X by $C = \{C_1, C, ..., C_K\}$, where C_k indicates k_{th} cluster, $k = 1, .., K$. Cluster compactness is measured based on a mean of within-cluster distances. The average distance $a(\mathbf{x})$ between element $\mathbf{x}$ and the other elements $\mathbf{x}_k$ belonging to the same cluster is defined as:

$$a(\mathbf{x}) = \frac{1}{n_k - 1} \sum_{\mathbf{x}_k \in C_k} d\left(\mathbf{x}, \mathbf{x}_k\right) \tag{1}$$

where n_k is the number of elements in C_k and $d\left(\mathbf{x}, \mathbf{x}_k\right)$ is a function of the distance between $\mathbf{x}$ and $\mathbf{x}_k$.

Furthermore, the mean of distances of $\mathbf{x}$ to the other elements $\mathbf{x}_l$ belonging to cluster C_l, where $l = 1, ..., K$ and $l \neq k$, can be written as:

$$\delta(\mathbf{x}, \mathbf{x}_l) = \frac{1}{n_l} \sum_{\mathbf{x}_l \in C_l} d\left(\mathbf{x}, \mathbf{x}_l\right) \tag{2}$$

where n_l is the number of elements in C_l. Thus, the smallest distance $\delta(\mathbf{x}, \mathbf{x}_l)$ can be defined as:

$$b(\mathbf{x}) = \min_{\substack{l,k=1 \\ l \neq k}}^{K} \delta(\mathbf{x}, \mathbf{x}_l) \tag{3}$$

The so-called *silhouette width* of element $\mathbf{x}$ can be expressed as follows:

$$S(\mathbf{x}) = \frac{b(\mathbf{x}) - a(\mathbf{x})}{max\left(a(\mathbf{x}), b(\mathbf{x})\right)} \tag{4}$$

Finally, the *Silhouette* (SIL) index is defined as:

$$SIL = \frac{1}{\mathrm{n}} \sum_{\mathbf{x} \in X} \frac{b(\mathbf{x}) - a(\mathbf{x})}{\max(a(\mathbf{x}), b(\mathbf{x}))} \tag{5}$$

where n is the number of elements in the data set X.

The value of the index is from the range -1 to 1 and a maximum value (close to 1) indicates the right partition scheme. Unfortunately, the index can detect incorrect data partition if differences between cluster distances are large [34].

2.2 Modification of the Silhouette Index

As mentioned above the modification of the *Silhouette* index was proposed in paper [34] and it involves using an additional component $A(\mathbf{x})$, which corrects values of the index. Thus, the new index, called the $SILA$ index, in that paper was defined as follows:

$$SILA = \frac{1}{n}\left(\sum_{\mathbf{x}\in X}\left(S(\mathbf{x})\cdot A(\mathbf{x})\right)\right) \tag{6}$$

where the $S(x)$ is the *silhouette width* (Eq. (4)), whereas the additional component $A(\mathbf{x})$ was expressed as:

$$A(\mathbf{x}) = \frac{1}{(1+a(\mathbf{x}))} \tag{7}$$

or it can be written as follows:

$$A(\mathbf{x}) = \frac{1}{(1+a(\mathbf{x}))^q} \tag{8}$$

where the exponent $q = 1$.

Note that the value of exponent $q = 1$ can be insufficient for large difference of distances between clusters. Hence, the new modification of the index includes the component $A(\mathbf{x})$ in which the q exponent is defined as below:

$$q = 2 + \frac{K^2}{n} \tag{9}$$

where n is the number of elements in a data set. Thus, the new version of the index so-called $SILAv1$ can be presented in the following way:

$$SILAv1 = \frac{1}{n}\left(\sum_{\mathbf{x}\in X}\left(\frac{b(\mathbf{x})-a(\mathbf{x})}{\max\left(a(\mathbf{x}),b(\mathbf{x})\right)}\cdot\frac{1}{(1+a(x))^q}\right)\right) \tag{10}$$

where the q is expressed by (9).

This approach can ensure a better performance of this index than that previous version called the $SILA$ and its efficiency was proved based on the experiments carried out on different data sets. In the next section a detailed explanation of the modifications involving the use of the additional component is presented.

2.3 Remarks

As mentioned above, the *Silhouette* index takes values between -1 and 1. Appropriate data partitioning is identified by a maximum value of the index, which can be close to 1. Notice that the definition of the *silhouette width* can be also expressed as follows [26]:

$$S(\mathbf{x}) = \begin{cases} 1-\frac{a(\mathbf{x})}{b(\mathbf{x})} & if \quad b(\mathbf{x}) > a(\mathbf{x}) \\ 0 & if \quad b(\mathbf{x}) == a(\mathbf{x}) \\ \frac{b(\mathbf{x})}{a(\mathbf{x})}-1 & if \quad b(\mathbf{x}) < a(\mathbf{x}) \end{cases} \tag{11}$$

Here, it is clear that when $b(\mathbf{x})$ is much greater than $a(\mathbf{x})$, the ratio of $a(\mathbf{x})$ to $b(\mathbf{x})$ is very small, and $\mathrm{S}(\mathbf{x})$ is close to 1. But in the modified version of the index, the $SILAv1$ (or $SILA$), the additional component $A(\mathbf{x})$ makes it possible to correct the value of the *silhouette width*. In $A(\mathbf{x})$ a measure of cluster compactness $a(\mathbf{x})$ is used and plays a very important role. For instance, when a clustering algorithm greatly increases sizes of clusters, the factor $a(\mathbf{x})$ also increases and the ratio of $1/(1+a(x))^q$ decreases significantly. It decreases the value of the index and thus, the large differences of distances between clusters do not affect the final result so much. This modified *silhouette width* can be expressed as follows:

$$S_m(\mathbf{x}) = \begin{cases} \left(1 - \frac{a(\mathbf{x})}{b(\mathbf{x})}\right) \cdot \frac{1}{(1+a(\mathbf{x}))^q} & if \quad b(\mathbf{x}) > a(\mathbf{x}) \\ 0 & if \quad b(\mathbf{x}) == a(\mathbf{x}) \\ \left(\frac{b(\mathbf{x})}{a(\mathbf{x})} - 1\right) \cdot \frac{1}{(1+a(\mathbf{x}))^q} & if \quad b(\mathbf{x}) < a(\mathbf{x}) \end{cases} \tag{12}$$

Let us look at the first situation. When $b(\mathbf{x})$ is greater than $a(\mathbf{x})$, the ratio of $a(\mathbf{x})$ to $b(\mathbf{x})$ is less than 1 and the value of $S_m(\mathbf{x})$ is positive (see Eq. (12)). Notice that when the number of clusters K decreases from K_{max} to a correct number of clusters c^*, then the clusters newly created by a clustering algorithm become larger and the value of $a(x)$ increases. However, the value of $a(\mathbf{x})$ is not very great and the factor $A(\mathbf{x})$ does not decrease so much. Thus, the value of $S_m(\mathbf{x})$ increases and it is only slightly reduced by $A(\mathbf{x})$. Generally, for compact clusters subdivided into smaller ones, when they are merged in larger clusters, the changes of their compactness and separability are not very significant. On the other hand, when the number of clusters K is equal to the right number c^*, the separability of clusters increases abruptly due to relatively large distances between clusters and now $b(\mathbf{x})$ is much larger than $a(\mathbf{x})$. Hence, when $K = c^*$, the component $S_m(\mathbf{x})$ increases considerably. Notice that $A(\mathbf{x})$ does not change significantly, since the changes of clusters compactness are still small and so $a(\mathbf{x})$ does not increase so much. Thus, the value of $S_m(\mathbf{x})$ is not considerably reduced by $A(\mathbf{x})$. In turn, when the number of clusters $K < c^*$, then cluster sizes can be really large and now the factor $a(\mathbf{x})$ strongly increases. Consequently, $A(\mathbf{x})$ decreases significantly and reduces the value of the index. It overcomes problems with too great differences of distances between clusters, and allows for indication of the appropriate data partitioning by the validity index.

The other situation takes place when $a(\mathbf{x})$ and $b(\mathbf{x})$ are equal. This means that it is not clear to which clusters the element should belong. In this case, the $SILAv1$ index (or $SILA$ and *Silhouette* indices) equals 0 (see Eqs. (11) and (12)). The last situation occurs when the factor $a(\mathbf{x})$ is larger than $b(\mathbf{x})$. In this case, the values of $S_m(\mathbf{x})$ (or $S(\mathbf{x})$) are negative. Thus, $\mathbf{x}$ should be assigned to another cluster. Notice that when $b(\mathbf{x})$ is equal to 0, then $S_m(\mathbf{x}) = -1/(1+a(x))^q$.

As mentioned above, the $SILA$ index uses $q = 1$. However, such value q can cause that the $A(\mathbf{x})$ is too small to appropriately correct the *silhouette width*. However, when q is too large, the influence of $A(\mathbf{x})$ can be very strong and then the value of the index greatly decreases. Hence, the issue of the choice of the exponent q for $A(\mathbf{x})$ is a very significant problem. The new version of the index called $SILAv1$ contains a formula of the change of the exponent q depending on the number of clusters and it is expressed by (9). It should be noted that in this definition the important role is played by the ratio K^2/n, which makes that for large K the value of q is greater than 2 (close to 3) and for small K it is close to 2. This approach causes that the index does not obtain too large values for high K (q is close 3). It is very important because this index is strongly decreased by the component $A(\mathbf{x})$ with q close to 2 for small K, when values of $a(\mathbf{x})$ are large. Thus, component $A(\mathbf{x})$ has now a suitable influence on the index and it makes it possible to overcome the drawback of the *Silhouette* index, where large differences of distances between clusters can provide incorrect results. It should be observed that the new index can take values between 1 and $-1/(1+a(x))^q$.

In the next section the results of the experimental studies are presented to confirm effectiveness of this approach.

3 Experimental Results

In this section several experiments were carried out to verify effectiveness of the new index in detecting correct clusters. The experiments have been conducted on different data sets using hierarchical clustering. It should be noted that proper clustering of data is not possible without the knowledge of the right number of clusters occurring in the given data set. Thus, this parameter is a very important for most of the clustering algorithms but it is not usually known in advance. Cluster validity indices are often used to determine this parameter.

The experiments relate to determining the number of clusters in data sets when the *Complete-linkage* hierarchical clustering is applied as the underlying clustering method. In each step this algorithm combines the two clusters with the smallest maximum pairwise distance. Furthermore, three validity indices, i.e. the *Silhouette* (SIL), *SILA* and $SILAv1$ are used to indicate the right number of clusters. Note that the best range of the number of clusters for data clustering analysis should be varied from $K_{max} = \sqrt{n}$ to $K_{min} = 2$ [22]. However, for the hierarchical clustering the number varies from $K_{max} = n$ to $K_{min} = 2$. To show the efficacy of the proposed approaches the values of validity indices are also presented on the plots, where the number of clusters was from the range $K_{max} = \sqrt{n}$ to $K_{min} = 2$. Moreover, it is assumed that the values of the validity indices are equal to 0 for $K = 1$.

In all the experiments the Weka machine learning toolkit [35] has been used, where the Euclidean distance and the min-max data normalization have been also applied.

3.1 Data Sets

Eight generated artificial data sets are used in the experiments. These data consist of various cluster structure, densities and dimensions. For instance, the first four of them called *Data* 1, *Data* 2, *Data* 3 and *Data* 4 are 2- dimensional with

3, 5, 8 and 15 clusters, respectively. The scatter plot of these data is presented in Figs. 1 and 2. Additionally, Table 1 shows a detailed description of all these artificial data. As it can be observed on the plots the distances between clusters are very different and some clusters are quite close. Generally, clusters are located in groups and some of clusters are very close and others quite far. Moreover, the sizes of the clusters are different and they contain various number of elements. Hence, many clusters validity indices can provide incorrect partitioning schemes.

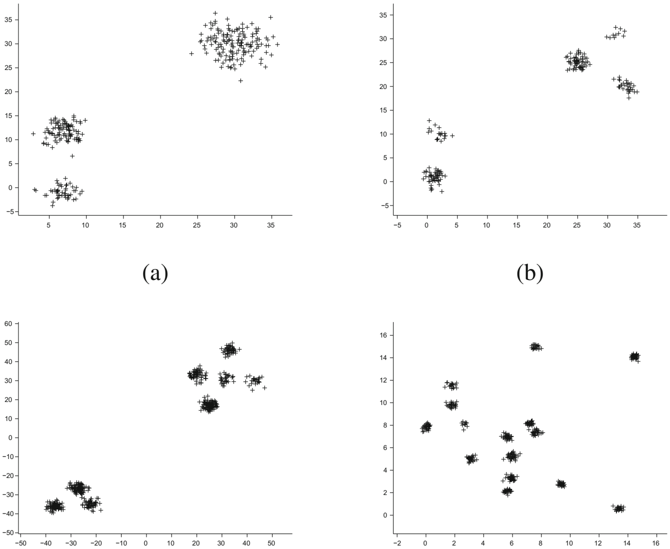
Table 1. Detailed description of the artificial data sets

Data sets	No. of elements	Features	Classes	No. of elements per class
Data 1	300	2	3	50,100,150
Data 2	170	2	5	10,20,30,50,60
Data 3	495	2	8	25,30,50,50,60,80,100,100
Data 4	429	2	15	31,39,38,18,29,30,32,27,10,39,22,27,39,20,28
Data 5	550	3	4	100,100,150,200
Data 6	820	3	6	100,100,100,150, 170,200
Data 7	800	3	7	70,80,100,100,100, 150,200
Data 8	460	3	9	30,30,40,40,50,50, 50,70,100

Experiments. The hierarchical *Complete-linkage* method as the underlying clustering method was used for partitioning of these data. In Figs. 3 and 4 a comparison of the variations of the *Silhouette*, *SILA* and $SILAv1$ indices with respect to the number of clusters are presented. As mentioned above, on the plots the maximal value of the number of clusters K_{max} is equal to $\sqrt{n}$ and values of the validity indices are equal 0 for $K = 1$. It can be seen that the $SILAv1$ index provides the correct number of clusters for all the data sets. However, the previous index *SILA* indicates incorrect partition schemes for two sets, i.e., *Data* 3 and *Data* 6. On the contrary, the *Silhouette* index incorrectly selects all partitioning schemes and this index mainly provides high distinct peaks when the number of clusters $K = 2$. This means that when the clustering method combines clusters into larger ones and differences of distances between them are large, influence of the separability measure is significant and consequently, this index provides incorrect results. On the other hand, despite the fact that the differences of distances between clusters are large, the $SILAv1$ index provides the correct partitioning for all these data. Notice that the component $A(\mathbf{x})$ (in the $SILAv1$ or *SILA* indices) poorly reduces values of this index when the number of clusters $K > c^*$, because then they are not so large and have a compact structure.

4 Conclusions

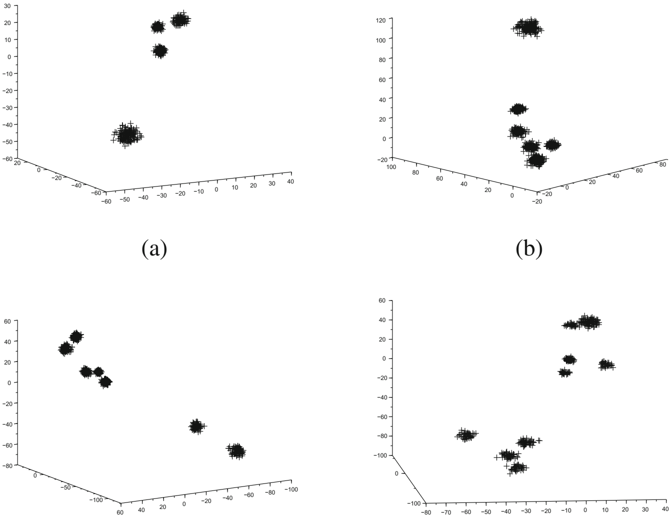
As mentioned above, the *Silhouette* index can indicate an incorrect partitioning scheme when there are large differences of distances between clusters in a



(a) (b)

(c) (d)

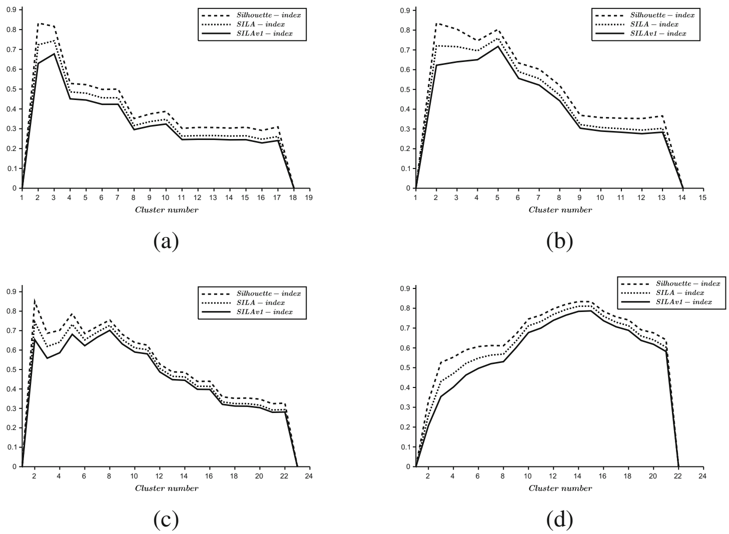
Fig. 1. 2-dimensional artificial data sets: (a) *Data* 1, (b) *Data* 2, (c) *Data* 3, and (d) *Data* 4



(a) (b)

(c) (d)

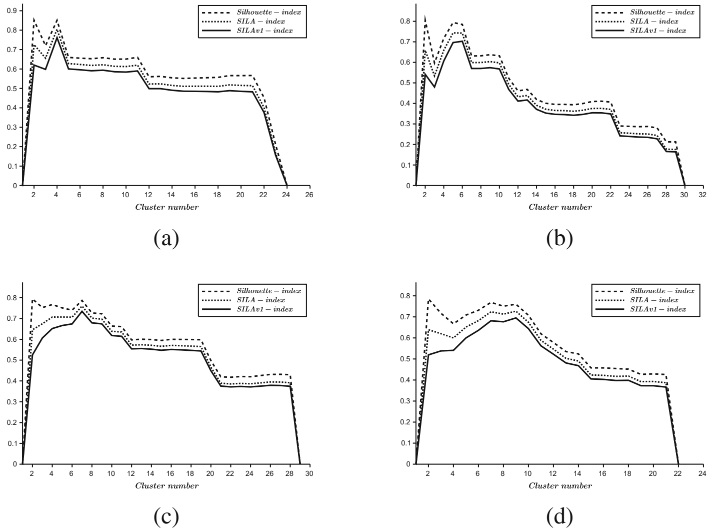
Fig. 2. 3-dimensional artificial data sets: (a) *Data* 5, (b) *Data* 6, (c) *Data* 7, and (d) *Data* 8



(a) (b)

(c) (d)

Fig. 3. Variations of the *Silhouette*, *SILA* and *SILAv*1 indices with respect to the number of clusters for 2-dimensional data sets: (a) *Data* 1, (b) *Data* 2, (c) *Data* 3, and (d) *Data* 4 partitioned by the *Complete-linkage* method.



(a) (b)

(c) (d)

Fig. 4. Variations of the *Silhouette*, *SILA* and *SILAv*1 indices with respect to the number of clusters for 3-dimensional data sets: (a) *Data* 5, (b) *Data* 6, (c) *Data* 7, and (d) *Data* 8 partitioned by the *Complete-linkage* method.

data set. Consequently, to improve the index performance and to overcome the drawback, a change of the index has been proposed. It is based on the use of the additional component, which contains a measure of cluster compactness. The value of this measure increases when a cluster size increases considerably. Hence, the additional component decreases and it reduces the high values of the index caused by large differences between clusters. As the underlying clustering algorithms the *Complete-linkage* was selected to investigate the behaviour of the proposed validity indices. The conducted tests have proven the advantages of the proposed $SILA$ and $SILAv1$ indices compared to the *Silhouette* index. In these experiments, several data sets were used and the number of clusters varied within a wide range. All the presented results confirm high efficiency of the $SILAv1$ index.

References

1. Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Prez, J.M., Perona, I.: An extensive comparative study of cluster validity indices. Pattern Recogn. **46**, 243–256 (2013)
2. Bilski, J., Smoląg, J.: Parallel architectures for learning the RTRN and Elman dynamic neural networks. IEEE Trans. Parallel Distrib. Syst. **26**(9), 2561–2570 (2015)
3. Bilski, J., Wilamowski, B.M.: Parallel learning of feedforward neural networks without error backpropagation. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2016. LNCS, vol. 9692, pp. 57–69. Springer, Cham (2016). doi:10.1007/978-3-319-39378-0_6
4. Bilski, J., Kowalczyk, B., Żurada, J.M.: Application of the givens rotations in the neural network learning algorithm. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2016. LNCS, vol. 9692, pp. 46–56. Springer, Cham (2016). doi:10.1007/978-3-319-39378-0_5
5. Bradley, P., Fayyad, U.: Refining initial points for K-Means clustering. In: Proceedings of the Fifteenth International Conference on Knowledge Discovery and Data Mining, pp. 9–15. AAAI Press, New York (1998)
6. Cpałka, K., Rebrova, O., Nowicki, R., Rutkowski, L.: On design of flexible neuro-fuzzy systems for nonlinear modelling. Int. J. Gen. Syst. **42**(6), 706–720 (2013)
7. Cpałka, K., Rutkowski, L.: Flexible Takagi-Sugeno fuzzy systems. In: Proceedings of the 2005 IEEE International Joint Conference on Neural Networks IJCNN (2005)
8. Davies, D.L., Bouldin, D.W.: A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell. **1**(2), 224–227 (1979)
9. Duch, W., Korbicz, J., Rutkowski, L., Tadeusiewicz, R. (eds.): Biocybernetics and Biomedical Engineering 2000. Neural Networks, vol. 6. Akademicka Oficyna Wydawnicza EXIT, Warsaw (2000)
10. Dunn, J.C.: Well separated clusters and optimal fuzzy partitions. J. Cybernetica **4**, 95–104 (1974)
11. Fränti, P., Rezaei, M., Zhao, Q.: Centroid index: cluster level similarity measure. Pattern Recogn. **47**(9), 3034–3045 (2014)
12. Fred, L.N., Leitao, M.N.: A new cluster isolation criterion based on dissimilarity increments. IEEE Trans. Pattern Anal. Mach. Intell. **25**(8), 944–958 (2003)
13. Gabryel, M.: A bag-of-features algorithm for applications using a NoSQL database. Inf. Softw. Technol. **639**, 332–343 (2016)

14. Gabryel, M., Grycuk, R., Korytkowski, M., Holotyak, T.: Image indexing and retrieval using GSOM algorithm. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2015. LNCS, vol. 9119, pp. 706–714. Springer, Cham (2015). doi:10.1007/978-3-319-19324-3_63
15. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: Clustering validity checking methods: part II. ACM SIGMOD Rec. **31**(3), 19–27 (2002)
16. Jain, A., Dubes, R.: Algorithms for Clustering Data. Prentice-Hall, Englewood Cliffs (1988)
17. Kim, M., Ramakrishna, R.S.: New indices for cluster validity assessment. Pattern Recogn. Lett. **26**(15), 2353–2363 (2005)
18. Lago-Fernández, L.F., Corbacho, F.: Normality-based validation for crisp clustering. Pattern Recogn. **43**(3), 782–795 (2010)
19. Meng, X., van Dyk, D.: The EM algorithm - an old folk-song sung to a fast new tune. J. R. Stat. Soc. Ser. B (Methodol.) **59**(3), 511–567 (1997)
20. Murtagh, F.: A survey of recent advances in hierarchical clustering algorithms. Comput. J. **26**(4), 354–359 (1983)
21. Pakhira, M.K., Bandyopadhyay, S., Maulik, U.: Validity index for crisp and fuzzy clusters. Pattern Recogn. **37**(3), 487–501 (2004)
22. Pal, N.R., Bezdek, J.C.: On cluster validity for the fuzzy c-means model. IEEE Trans. Fuzzy Syst. **3**(3), 370–379 (1995)
23. Pascual, D., Pla, F., Sánchez, J.S.: Cluster validation using information stability measures. Pattern Recogn. Lett. **31**(6), 454–461 (2010)
24. Pelleg, D., Moore, A.W.: X-means: extending k-means with efficient estimation of the number of clusters. In: Proceedings of the Seventeenth International Conference on Machine Learning, pp. 727–734 (2000)
25. Rohlf, F.: Single-link clustering algorithms. In: Krishnaiah, P.R., Kanal, L.N. (eds.) Handbook of Statistics, vol. 2, pp. 267–284. Amsterdam, North-Holland (1982)
26. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**, 53–65 (1987)
27. Rutkowski, L., Cpałka, K.: Compromise approach to neuro-fuzzy systems. In: Sincak, P., Vascak, J., Kvasnicka, V., Pospichal, J. (eds.) Intelligent Technologies - Theory and Applications. New Trends in Intelligent Technologies. Frontiers in Artificial Intelligence and Applications, pp. 85–90. IOS Press, Amsterdam (2002)
28. Rutkowski, L., Przybył, A., Cpałka, K., Er, M.J.: Online speed profile generation for industrial machine tool based on neuro-fuzzy approach. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2010. LNCS, vol. 6114, pp. 645–650. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13232-2_79
29. Rutkowski, L., Cpałka, K.: A neuro-fuzzy controller with a compromise fuzzy reasoning. Control Cybern. **31**(2), 297–308 (2002)
30. Saha, S., Bandyopadhyay, S.: Some connectivity based cluster validity indices. Appl. Soft Comput. **12**(5), 1555–1565 (2012)
31. Sameh, A.S., Asoke, K.N.: Development of assessment criteria for clustering algorithms. Pattern Anal. Appl. **12**(1), 79–98 (2009)
32. Shieh, H.-L.: Robust validity index for a modified subtractive clustering algorithm. Appl. Soft Comput. **22**, 47–59 (2014)
33. Starczewski, A.: A new validity index for crisp clusters. Pattern Anal. Appl. 1–14 (2015). doi:10.1007/s10044-015-0525-8

34. Starczewski, A., Krzyżak, A.: A modification of the silhouette index for the improvement of cluster validity assessment. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2016. LNCS, vol. 9693, pp. 114–124. Springer, Cham (2016). doi:10.1007/978-3-319-39384-1_10
35. Weka 3: Data Mining Software in Java. University of Waikato, New Zealand. http://www.cs.waikato.ac.nz/ml/weka/
36. Wu, K.L., Yang, M.S., Hsieh, J.N.: Robust cluster validity indexes. Pattern Recogn. **42**, 2541–2550 (2009)
37. Zhao, Q., Fränti, P.: WB-index: a sum-of-squares based index for cluster validity. Data Knowl. Eng. **92**, 77–89 (2014)