Medoid-based shadow value validation and visualization

Weksi Budiaji [a,b,1,*]

[a] Agribusiness Department, Sultan Ageng Tirtayasa University, Serang, Indonesia
[b] Institut of Applied Statistics and Computing, University of Natural Resources and Life Sciences, Vienna, Austria
[1] budiaji@untirta.ac.id
* corresponding author

ARTICLE INFO

Article history
Received January 25, 2019
Revised April 2, 2019
Accepted April 4, 2019
Available online June 1, 2019

Keywords
Cluster validation
Cluster visualization
Internal criteria
Medoid
Shadow value

ABSTRACT

A silhouette index is a well-known measure of an internal criteria validation for the clustering algorithm results. While it is a medoid-based validation index, a centroid-based validation index that is called a centroid-based shadow value (CSV) has been developed. Although both are similar, the CSV has an additional unique property where an image of a 2-dimensional neighborhood graph is possible. A new internal validation index is proposed in this article in order to create a medoid-based validation that has an ability to visualize the results in a 2-dimensional plot. The proposed index behaves similarly to the silhouette index and produces a network visualization, which is comparable to the neighborhood graph of the CSV. The network visualization has a multiplicative parameter (c) to adjust its edges visibility. Due to the medoid-based, in addition, it is more an appropriate visualization technique for any type of data than a neighborhood graph of the CSV.

1. Introduction

Cluster analysis is an unsupervised method to group objects such that homogenous objects are clustered in the same group. As an unsupervised method in which pre-defined class memberships are absent, the partitioning result from a cluster analysis has to be validated via relative, external, or internal criteria validation [1]. The validation criteria differ with respect to the compactness assumption and information provided in the data.

Among the three criteria, the relative criteria do not require the compactness assumption. It is based on a re-sampling scheme via either cross-validation or bootstrap methods. The latter is a sampling with replacement strategy to assess the stability of clusters [2] and select the appropriate number of clusters [3]. The stability is then visualized in a heatmap figure [4] where a block diagonal figure depicts the most stable cluster result.

The external criteria are commonly applied in a benchmarking process with either known classes or the “gold standard” algorithm [5]. When a new clustering algorithm is developed, the routine process to evaluate this algorithm is by applying it in a supervised environment. Then, an evaluation measure compares this new algorithm to the existing/ gold algorithms. Two examples of external criteria applied to compare a new algorithm are the clustering accuracy rate [6] and the cluster purity [5], [7]-[8]. Reference [9], moreover, has addressed many external criteria, e.g. Rand [10], and adjusted Rand [11].

The internal criteria, on the other hand, are applied when a real data set, which is lacking true classes, is analyzed by means of cluster analysis. Reference [12] has cited as many as 19 internal validation indices, e.g. silhouette [13], and gap statistic [14]. A silhouette index is a well-known internal validation index

http://dx.doi.org/10.26555/ijain.v5i2.326

[15], which gains popularity due to its visualization of each cluster. It non-linearly combines the compactness and separation assumptions [16]. A similar approach to silhouette has been developed namely a shadow value [17]-[18], in which the value is calculated based on the first and second closest centroids and can also be visualized as in a silhouette index.

Although the silhouette and shadow values can be visualized, they depict different figures for the same case such that presenting these two simultaneously requires extra attention. Well-separated clusters, for instance, are indicated by high values of the silhouette [13]. On the contrary, they have small indices in the case of shadow values [18]. While the silhouette index as a medoid-based approach is suitable for any type of data, the shadow value, which is based on a centroid-based method, is applicable for numerical data only. The latter can produce a 2-dimensional representation of the clustering results in a neighborhood graph. For any type of data, however, a neighborhood graph is not visible and a medoid-based approach visualization via the silhouette is also absent.

In this paper, we propose a new formula of a medoid-based validation technique that imitates the silhouette and centroid-based shadow value characters. This technique produces also a similar figure to that of the silhouette such that a side-by-side presentation of these two figures is consistently interpretable. With the new medoid-based validation technique, in addition, a 2-dimensional graph of visualization for any type of data is also possible surpassing a neighborhood graph of the shadow value.

2. The Proposed Method

2.1. Shadow Value for Medoid-based Clustering

The silhouette value can be calculated via

$$si\ (x) = \frac{(b_x - a_x)}{\max(a_x\ ,b_x)}, \tag{1}$$

where a_x and b_x are the average distance of object *x* to all objects within the cluster and to all objects within the nearest cluster, respectively [13]. The value of (1) has then a minimum -1 and maximum 1 where the best-separated clusters have a value equal to 1. It is also applicable for any distance. The shadow value, on the other hand, is attained by

$$sh\ (x) = \frac{2\ d(x,\ \ c(x))}{d(x,\ \ c(x)) + d(x,\ \ c`(x))}, \tag{2}$$

where *d*(*x*, c(*x*)) is the distance between object *x* to the first closest centroid and *d*(*x*, c`(*x*)) is the distance between object *x* to the second closest centroid [18]. The poorly separated clusters are indicated by a shadow value of 1 in (2) meaning that the first and second closest centroids are equidistant from *x*. Due to the centroid calculation to determine the center of the cluster, the centroid-based shadow value (2) is only valid for numerical distances.

The centroid-based shadow value (CSV) has 0 as a minimum value, which is achieved when the object is very close to the centroid. When an object has twice the distance to the second closest centroid compared to the first closest centroid, the shadow value is 0.67, which is considered as a high shadow value. Fig. 1, moreover, illustrates well-separated clusters via the silhouette and shadow values where high peaks occur in the silhouette and low peaks appear in the shadow plot. Due to the contradictory image between silhouette and shadow plots, it requires careful consideration when interpreting a side-by-side of these plots.

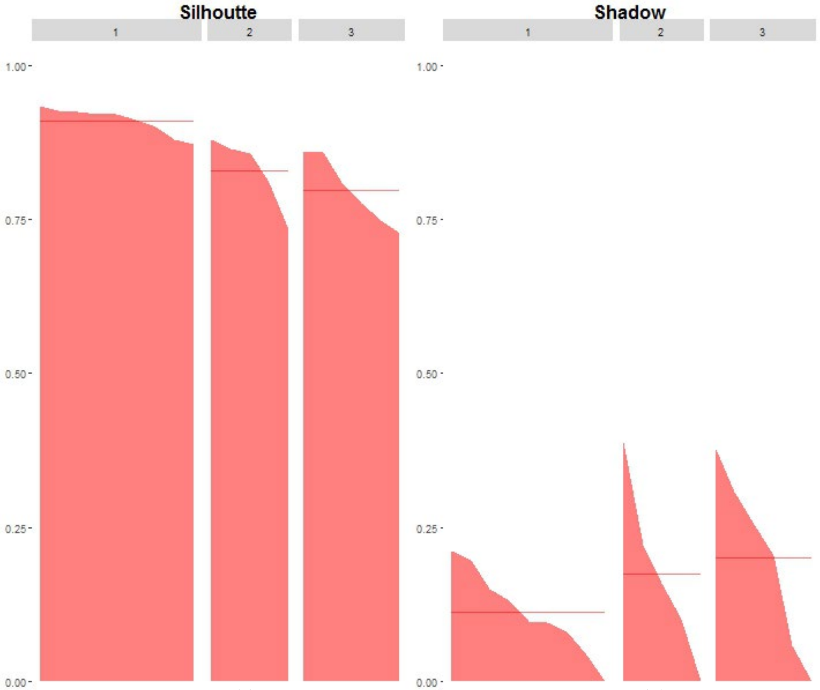
A new formula is developed to calculate a new shadow value in a medoid-based clustering technique. To adapt the silhouette and CSV characters, these following constraints are applied:

1. The lower and upper bounds of the value are 0 and 1.
2. The worst separated cluster is 0, while the best is 1.
3. The value of 0 is valid for an equidistant between the first and second closest medoids.
4. The value of 1 is achieved when the object is the medoid object.

With these constraints, the new shadow values in a medoid-based clustering are then simplified into

$$msv\,(x) = \frac{d(x,\ m`(x)) - d(x,\ m(x))}{d(x,\ m`(x))}, \tag{3}$$

where $d(x,\ m(x))$ is the distance between object x to the first closest medoid and $d(x,\ m`(x))$ is the distance between object x to the second closest medoid. Due to a medoid being the cluster center, any distance method is applicable in the medoid-based shadow value (MSV).



(a) (b)

Fig. 1. Silhouette (a) and shadow (b) values of well-separated clusters.

Fig. 2 illustrates the MSV of well-separated clusters where it depicts a similar figure to the silhouette plot (Fig. 1(a)), i.e. all bars have high peaks. Table 1, in addition, compares the index of the CSV vs MSV in a specified distance of the second closest centroid/medoid. An object that has an equidistant between the first and second closest centroid/medoid, has CSV equal to 1 compared to 0 in the MSV.

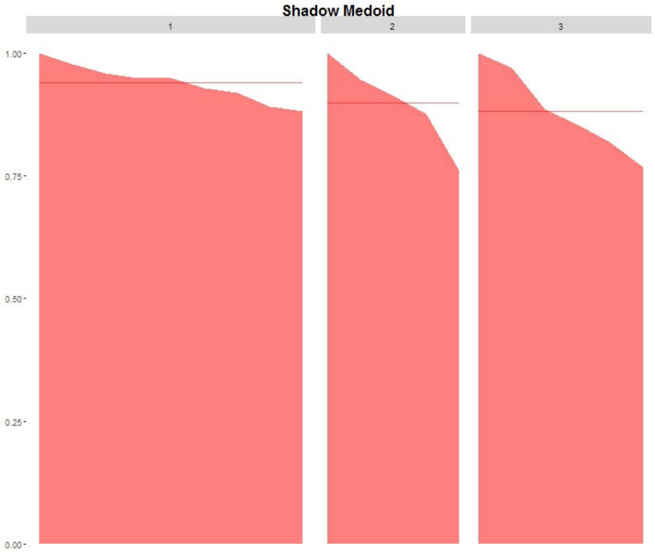


Fig. 2. Medoid-based shadow value (MSV) of well-separated clusters.

Table 1. Centroid-based shadow (CSV) and medoid-based shadow (MSV) values comparison

Indices	Distance to the second closest centroid/ medoid									
	1x	2x	3x	4x	5x	6x	7x	8x	9x	10x
CSV	1.00	0.67	0.50	0.40	0.33	0.29	0.25	0.22	0.20	0.18
MSV	0.00	0.50	0.67	0.75	0.80	0.83	0.86	0.88	0.89	0.90

2.2. Visualization

The CSV gains an advantage over the silhouette because the former can be visualized in two-dimensional space of a network graph topology, called a neighborhood graph. The graph has *k* nodes, where *k* is the number of clusters and is an undirected graph with an average of shadow values of the closest clusters as its edges/ lines [18]. The cluster similarity is measured by the average shadow value within a cluster and the closest cluster. Fig. 3 illustrates a neighborhood graph of well-separated clusters where all clusters have thin edges. A thick edge in a neighborhood graph, on the other hand, denotes a high shadow value indicating poor-separated clusters.

The representation of either thin or thick edges in a neighborhood graph is naturally attractive where a thick edge implies poorly separated clusters (close to each other). The thickness of the edge characteristic in the CSV is retained to develop a new technique of visualization in the MSV. The MSV visualization, however, gains an advantage due to its suitability for any type of data, i.e. numerical, binary, categorical, and mixed variables. In the MSV visualization, there are two types of graphs, namely the medoids and all-object visualizations.

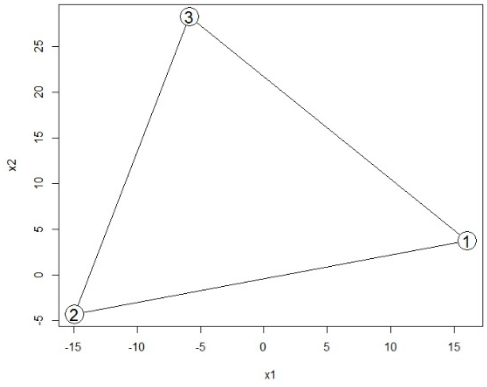


Fig. 3. Neighborhood graph of well-separated clusters

To create a medoids visualization, a matrix of *k* x *k* dimension, **M**, is introduced, where *k* is the number of clusters. The matrix **M** is then defined as

$$\mathbf{M} = \begin{bmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & \ddots & \vdots \\ a_{k1} & \cdots & a_{kk} \end{bmatrix} \begin{cases} a_{ij} = 0, & i = j \\ a_{ij} = \frac{dm_{ij} - \max(\bar{d}m_i, \bar{d}m_j)}{dm_{ij}}, & i \neq j \end{cases}, \tag{4}$$

where dm_{ij} is the distance between the medoids of the cluster *i* and *j*, i.e. the between-cluster distance, and $\bar{d}m_i$ is the average distance between the medoid of the cluster *i* to all objects in the cluster *i*, i.e. the average distance of the within cluster distance. Hence, the matrix **M** in (4) is an MSV among clusters. Applying (4), the off-diagonal elements of the matrix **M** are within [0, 1] where a value close to 0 indicates a low separation index, i.e. the cluster *i* and *j* are poorly separated, conversely, a value close to 1 indicates a high separation index.

The separation indices in the matrix **M** can be directly visualized in a network graph, which consists of nodes and edges. The aesthetics of the network graph consist of three parts; the number of nodes is equal to *k*, the off-diagonal elements of the matrix **M** represent edges, and the diagonal elements of the matrix **M** guarantee that there is no self-loop in each node. To set the thickness of the edges, we apply a transformation to the off-diagonal elements in order to create a network graph. They are transformed into 1—a_{ij} such that a high separation index has a low value of the edge thickness, while a low separation index has a high value of the edge thickness. Thus, they produce either a thin or thick edge that corresponds to a high or low separation index, respectively, in a network graph.

As the nodes and edges are properly defined, they are laid in a 2-dimensional space via a graph layout algorithm. Reference [19] has surveyed many graph layouts, e.g. Kamada-Kawai [20] and Fruchterman-Reingold [21]. The *x* and *y* axes are then meaningless whereas it is more relevant when the data have non-numerical variables than a neighborhood graph. Fig. 4 shows a network graph of the MSV by plotting directly the matrix **M**. The graph is similar to the neighborhood graph, which shows well-separated clusters indicated by thin edges.

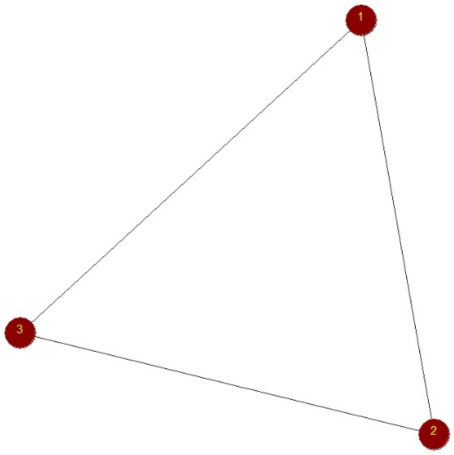


Fig. 4. Medoids network graph of well-separated clusters (Kamada-Kawai layout)

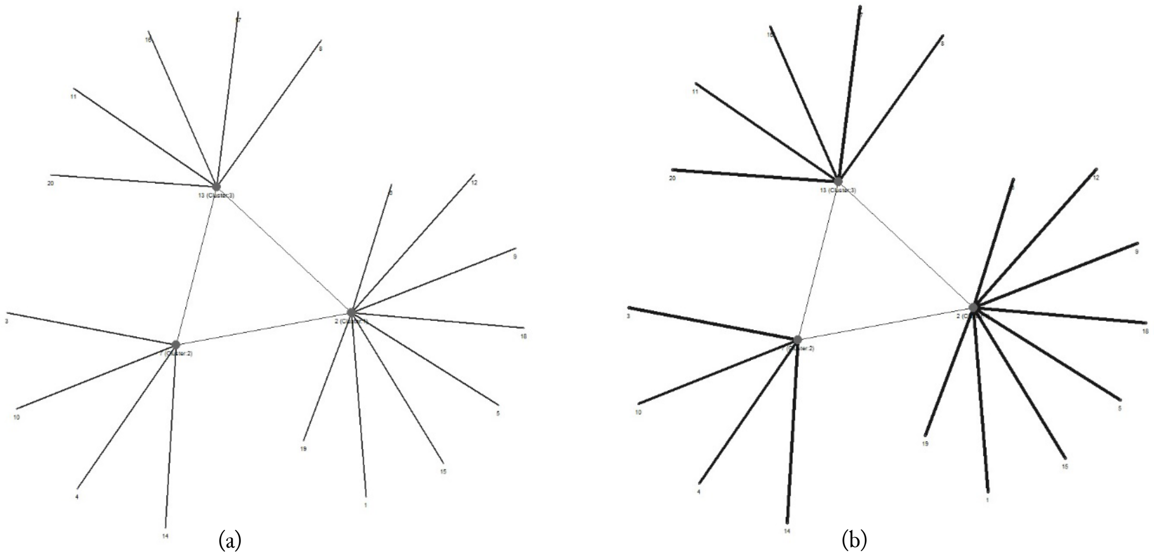
In the all-object visualization, on the other hand, a matrix of *n* x *n* dimension, **O**, is required, where *n* is the number of objects. The matrix **O** is defined as

$$\mathbf{O} = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{bmatrix} \begin{cases} b_{ij} = NA, & j \neq id\ medoid \\ b_{ij} = msv\ (i)_j, & j = id\ medoid \end{cases}, \tag{5}$$

where *msv(i)*$_j$ is the MSV of object *i*, computed by (3), with the corresponding first closest medoids *j*. Thus, there are only a single value, which is an MSV, and (*n* – 1) NAs in each row of the matrix **O**. A non-NA value in each row of the matrix **O** is deemed as a compactness index. Due to the MSV in (3) indicating a closeness of an object to the medoid, a high value in the matrix **O** denote a high closeness (compactness).

For the visualization of the matrix **O** in a network graph, it has also three aesthetic parts; the number of nodes is equal to *n*, the edges connect the objects to a particular medoid, and the NA values assure that each object is only connected to the medoid of the cluster. Then, to set the thickness of the edge, a thick edge indicates high compactness, which is identical to the MSV, in the matrix **O**. If the matrix **O** is visualized in a network graph directly, it produces a graph of connected nodes within a cluster yet disconnected nodes between clusters (medoids).

In order to produce a graph of connected nodes in both within and between clusters, all pieces of the network information from the matrix **M** and **O** have to be combined. Then, it can be directly translated into a network graph of the all-object MSV visualization. Fig. 5 illustrates well-separated clusters in a network graph with all-object visualization where it has a high separation and compactness, indicated by thin edges among medoid nodes and thick edges among object nodes, respectively. A constant (*c*) is also introduced in order to multiply the MSV such that the separation and compactness edges are more visible. The constant *c* is also applicable to the aforementioned medoid visualization such that the matrix **M** (4) is modified into a matrix *c*(**M**).



(a) (b)

Fig. 5. All-object network graph of well-separated clusters with c = 1 (a) and c = 2 (b).

3. Method

To apply the proposed MSV validation and visualization, some simulated data sets are generated. Reference [22] has developed an algorithm to generate numerical data set for clustering algorithm benchmarking with a pre-specified degree of separation [23]. The simulated data sets in this study vary in the separation degree only (well, middle, and poorly separated). The results of these three different separated clusters are compared among the three internal validation indices, i.e. the silhouette, CSV, and MSV, which produce visualizations.

Due to the focus of the study on the different settings of degree separation, the variables of *n* (the number of objects), *p* (the number of variables), and *k* (the number of clusters) in the simulated data are fixed such that they are set as 1000, 2, and 5, respectively (Table 2). The algorithm to group the data is also fixed via a popular partitioning around medoid (PAM) algorithm [24], which is one of the medoid-based algorithms. Then, each simulated data set is replicated. Although 50 replications for each simulated data set are fairly precise [25], the strategy to replicate the simulated data in this paper is via subsetting by choosing the number of the subset sample m = n/2, i.e. 1000/2 = 500 replicates.

For real data sets, the data sets from the UCI repository [26], which represent well and poorly separated clusters, are also analyzed. The analyses produced in this article, moreover, are run in an Intel i3 4GB RAM using R software environment [27] using the *clusterGeneration, cluster*, *kmed*, *ggplot2, geomnet,* and *flexclust* packages.

Table 2. The settings of the simulated data sets

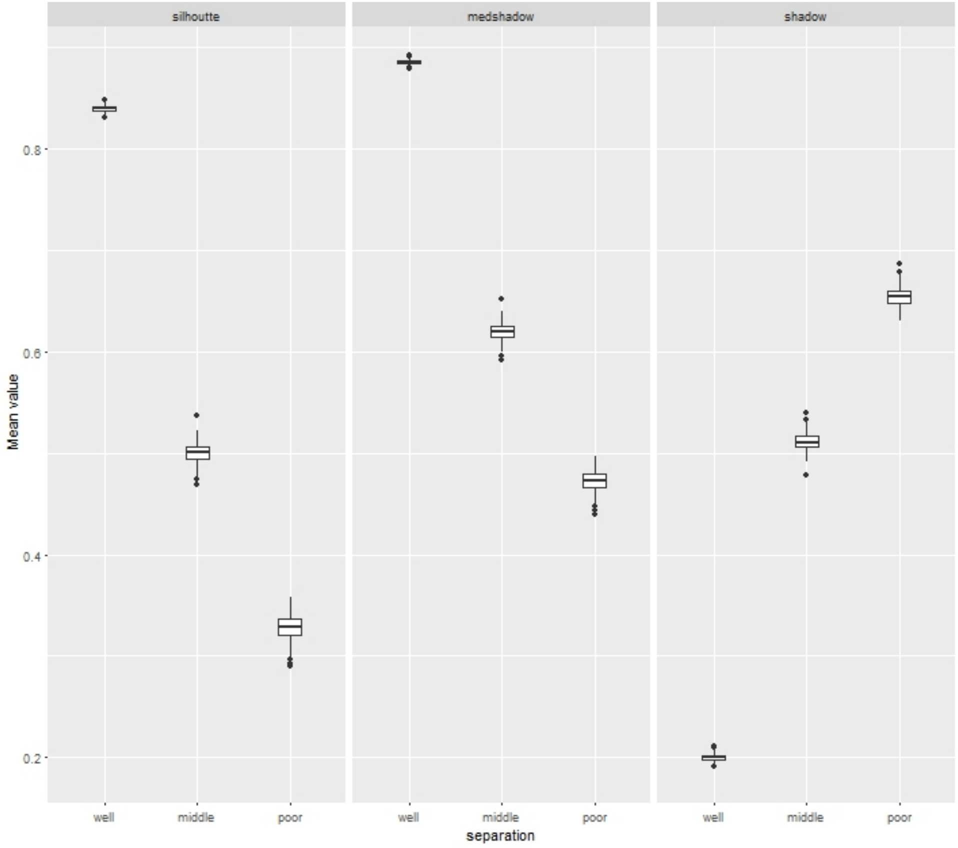
Simulated data set	Separation	*n* number of objects	*p* number of variables	*k* number of clusters
1	0.5 (well)	1000	2	5
2	0.0 (middle)			
3	-0.5 (poor)			

4. Results and Discussion

In this section, the MSV proposed index is applied in simulated data sets and real data sets. The simulated data sets are generated via the *clusterGeneration* package [28]. Meanwhile, the real data sets are two data sets of UCI repository data sets [26] namely the well-known iris data set, and lenses data sets to represent numerical and categorical data sets, respectively, partitioned by PAM via the *cluster* package [29]. The silhouette and CSV, moreover, are obtained by the *kmed* package [30]. While the neighborhood graph is visualized via the *flexclust* [17]-[18] package, a function, created with the assistance of the *ggplot2* [31] and *geomnet* [32] packages, visualizes the MSV network graph.

4.1. Simulated data

The first simulated data set (well-separated clusters) has high values in both the silhouette (Fig. 6(a)) and MSV (Fig. 6(b)) indices, yet it has low values in the CSV (Fig. 6(c)). The contradictory results of the CSV to the silhouette and MSV, moreover, occur in all types of simulated data set except in the middle-separated clusters where all indices produce comparable results between 0.4 and 0.6. Fig. 6 also shows that the MSV has always had a higher index compared to the silhouette value. It can be explained that the span value of the MSV is shorter than the silhouette value, i.e. [0,1] compared to [-1, 1] [13].

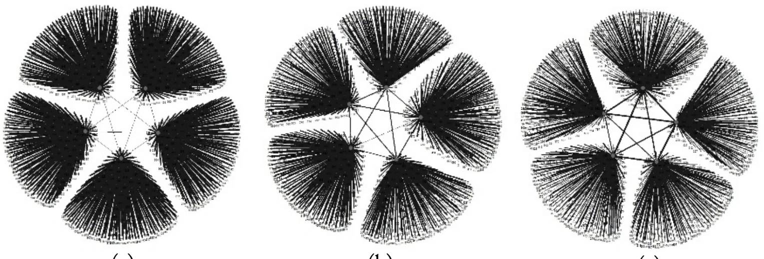


(a) (b) (c)

Fig. 6. Boxplot of the mean value indices of the silhouette (a), MSV (b), and shadow (c) values

As an internal validation index, compared to the CSV that produces in opposite values, the proposed MSV index adapts the behavior of the silhouette values well. Thus, the MSV gains an advantage of the similarity interpretation of the silhouette index, i.e. a high value of the index indicates a well-separated cluster. However, due to a shorter span of the MSV than silhouette index, a different threshold to define the quality of cluster results applies. A value of 0.5 in the silhouette value, for instance, may indicate a middle-separated cluster, while it is a poor-separated cluster in the MSV.

For the network visualization of the simulated data, which are partitioned into 5 clusters, all objects are plotted by comparing the well, middle, and poorly separated cluster data sets. Fig. 7 shows the dissimilarity among them. The well-separated clusters (Fig. 7(a)) have thin edges among the medoids and thicker edges in the within a cluster indicating that they have a high separation and compactness, respectively. Meanwhile, the poorly separated clusters (Fig. 7(c)) have the opposite image where the edges among the medoids are thicker than the edges within a cluster, which represents low separation among the medoids.



(a) (b) (c)

Fig. 7. All-object visualizations of well (a), middle (b), and poorly (c) separated clusters

4.2. Real data sets

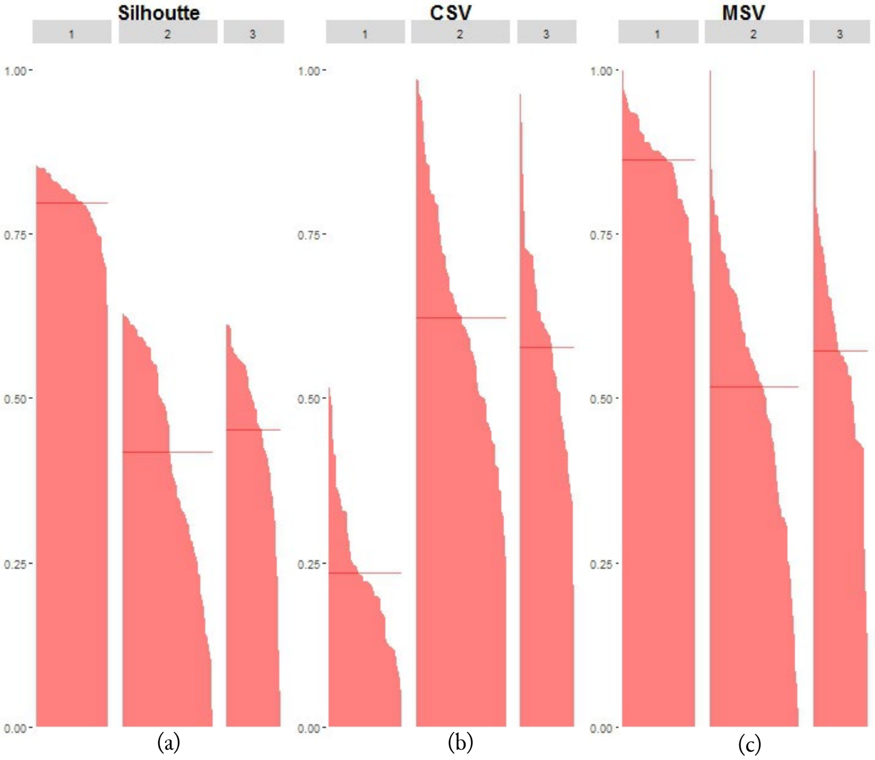
4.2.1. Iris data set

The iris data set is a well-known data set that consists of 150 objects divided into three species of iris (Setosa, Versicolor, and Virginica) with four numerical variables. To compare the silhouette, CSV, and MSV indices, the PAM algorithm in the Euclidean distance matrix of this data is applied with the number of clusters *k* equal to 3. The accuracy rate of the PAM algorithm is 86.67% (Table 3), which is 100% correct achieved in cluster 1 (Setosa class).

Table 3. The misclassification table of the PAM algorithm in the iris data set

	Setosa	Versicolor	Virginica
Cluster 1	50	0	0
Cluster 2	0	48	18
Cluster 3	0	2	36

When the internal validations with the silhouette, CSV, and MSV indices are plotted (Fig. 8), they produce similar results. However, it requires extra caution when the CSV figure is illustrated due to its dissimilarity to the other two. Cluster 1 has the best result indicated by a high peak of the silhouette and MSV, and a small peak of the CSV. Table 4 confirms the identical result. It also shows that cluster 2 and 3 are poorly separated.



(a) (b) (c)

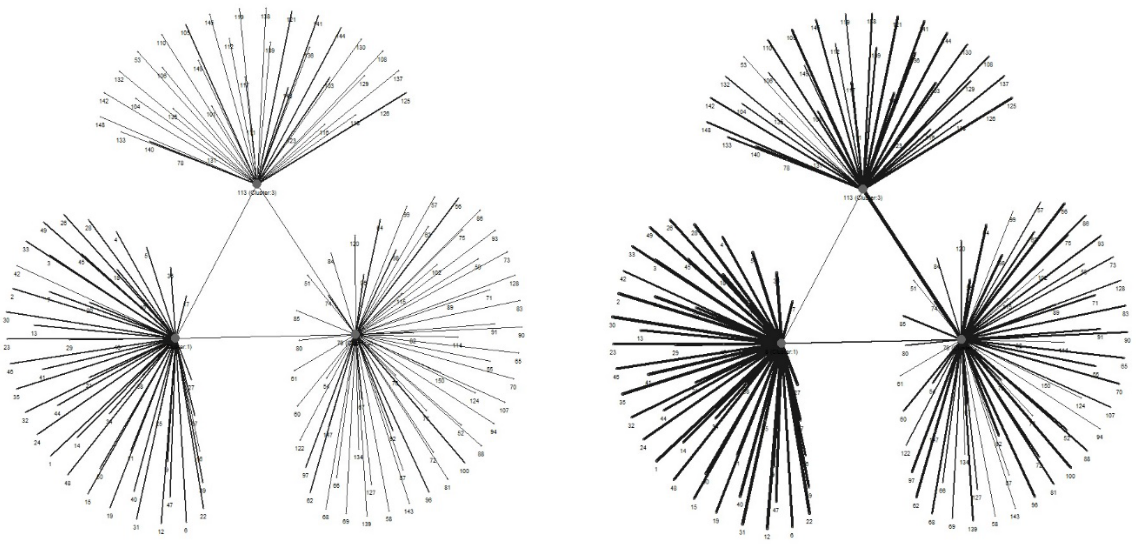
Fig. 8. Silhouette (a), CSV (b), and MSV (c) plots of the iris data set

Table 4. The means of the internal validation indices of the PAM algorithm in the iris data set

	Silhouette	CSV[1]	MSV[2]
Cluster 1	0.80	0.23	0.86
Cluster 2	0.42	0.62	0.52
Cluster 3	0.45	0.58	0.57

[1] Centroid-based shadow value, [2] Medoid-based shadow value

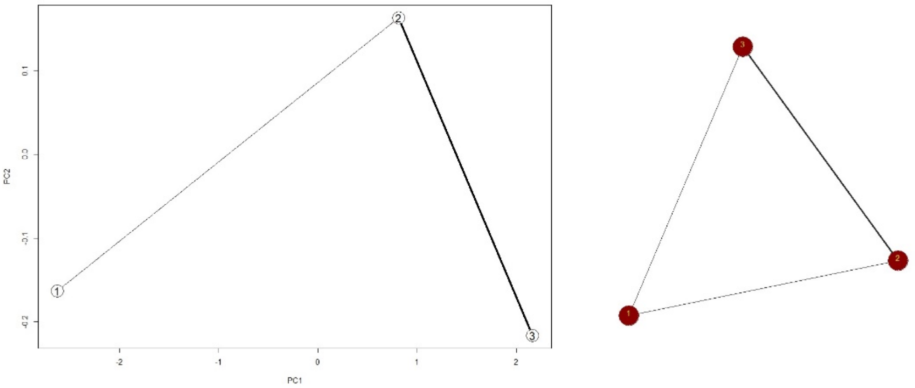
In a network graph, Fig. 9 illustrates network graphs of iris data set based on the all-object MSV visualization with two different multiplication constant (*c*). By adjusting the value of *c* into 2 (Fig. 9(b)), clusters 2 and 3 are discernable that they have a low separation, i.e. a thick edge. It also shows that cluster 1 has the highest compactness among the three clusters portrayed by having the thickest edges within cluster 1.



(a) (b)

Fig. 9. All-object visualization of the iris data set with c = 1 (a) and c = 2 (b)

The value of *c* = 2 is then adopted in the medoid visualization (Fig. 10(b)). It shows that cluster 1 is separable to cluster 2 and 3. If it is compared to the neighborhood graph (Fig. 10(a)), it depicts a similar image where clusters 2 and 3 have a low separation.



(a) (b)

Fig. 10. The neighborhood graph (a) and medoids visualization c = 2 (b) of the iris data set

The other difference between a neighborhood graph and a medoids visualization is that the former has interpretable axes. They can be the first and second principal components. In addition, they are replaceable by the variables. In the iris data case, for instance, the combination of x_1 and x_2 for the *x* and *y* axes can be selected. Hence, the total combination of the *x* and *y* axes that can be produced is six combinations due to the four variables involved. This is a good property of the neighborhood graph because all combinations can be applied and a final graph for a suitable visualization can be determined subsequently. Meanwhile, at the same time, too many variables involved in a data set cause impractical choices.

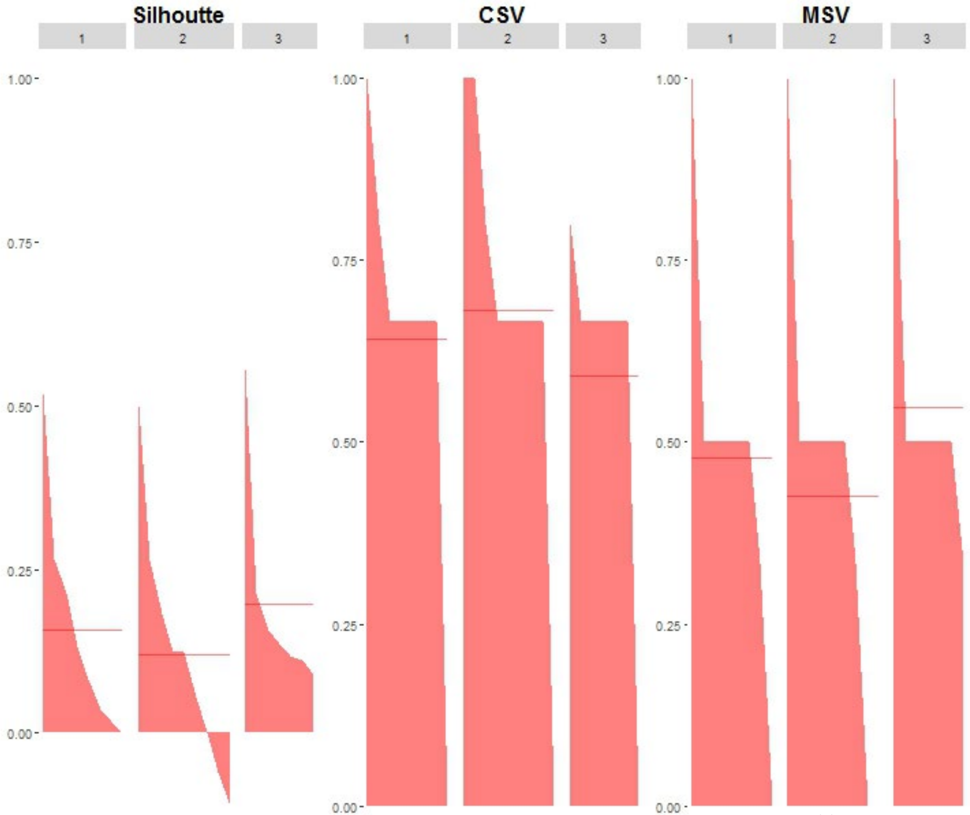
The neighborhood graph also has fewer edges than the medoids visualization, because the former draws an edge between two nodes if only at least one object has the closest and second closest to those nodes [17]. Meanwhile, the medoids visualization is based on the squared matrix **M** such that the number of edges is equal to $_kC_2$. With the number of clusters (*k*) equal to 5, for instance, there are 10 edges drawn among the medoids. Although a loaded of the edges can occupy the space of the graph when the number of clusters is large, a *c* parameter is a key to set the edge visibility of the image.

4.2.2. Lenses data set

The lenses data set consists of 24 patients with four categorical variables. The patients are classified into three groups: hard contact lenses, soft contact lenses, and none of those two types of lenses. The PAM algorithm in the simple matching distance matrix of this data is applied with *k* equal to 3. The accuracy rate is low, i.e. 50% (Table 5), which indicates poor separated clusters. Moreover, Fig. 11 shows that the three internal criteria validations indicate poorly separated clusters, i.e. low peaks in both the silhouette and MSV and high peaks in the CSV. Table 6 emphasizes the result, which all clusters are poorly separated.

Table 5. The misclassification table of the PAM algorithm in the lenses data set

	Hard	Soft	None
Cluster 1	3	1	5
Cluster 2	1	3	4
Cluster 3	0	1	6



(a) (b) (c)

Fig. 11. Silhoutte (a), CSV (b), and MSV (c) plots of the lenses data set

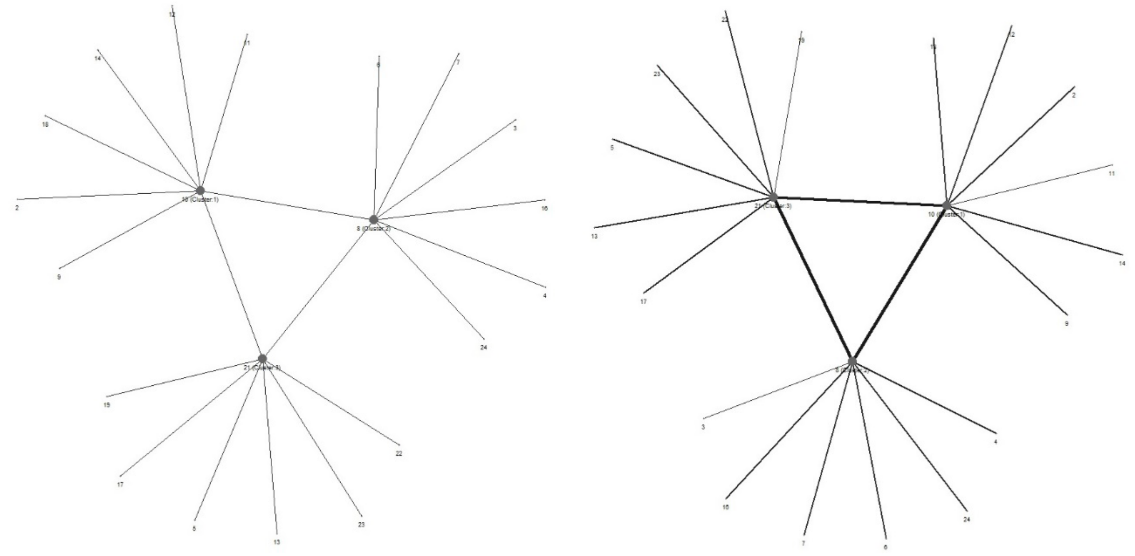
Table 6. The means of the internal validation indices of the PAM algorithm in the lenses data set

	Silhouette	CSV[1]	MSV[2]
Cluster 1	0.16	0.64	0.48
Cluster 2	0.12	0.68	0.43
Cluster 3	0.20	0.59	0.55

[1] Centroid-based shadow value, [2] Medoid-based shadow value

When all objects are visualized in a network graph with c = 2, all clusters have low separation indices indicated by thick edges among the medoids (Fig. 12(b)). The compactness within a cluster is also low representing by thin edges within a cluster. Fig. 13, moreover, illustrates the medoid visualization with c = 2, adapted from the all-object network graph, in which all medoids are close to each other, i.e. poorly separated. On the other hand, the neighborhood graph version of this plot is absent due to non-numerical variable data set.

With this type of data set, i.e. categorical data set, the centroid calculation is unfeasible in the CSV context. Although a transformation of a medoid-based into a centroid-based algorithm in order to produce a CSV is applicable [17], a neighborhood graph is unachievable due to the absence of the centroid values. Thus, the medoids visualization gains an advantage compared to a neighborhood graph in a non-numerical data set, i.e. a categorical and mixed variables data set.



(a) (b)

Fig. 12. All-objects visualization of the lenses data set with c = 1 (a) and c = 2 (b)

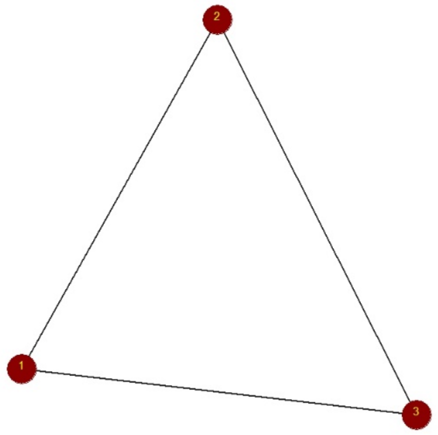


Fig. 13. The medoid visualization of the lenses data set with c = 2

5. Conclusion

In this paper, we proposed an internal criteria validation for clustering results, namely the medoid-based shadow value (MSV). The MSV index imitated the silhouette index behavior where the higher value of the index, the better the clustering result, i.e. it had identical interpretation to the silhouette index. On the other hand, the value of the MSV was always higher than the silhouette index due to a shorter span of the MSV such that a particular threshold to determine the quality of cluster results applied. For the visualization of the MSV, a medoids graph of the MSV produced a similar figure to a neighborhood graph of the CSV. An all-object visualization was able to be created from the MSV as well. Both the medoid and all-object network visualizations of the MSV had a parameter *c* to regulate the visibility of the edges. It was suggested to first apply the all-object network graph with multiple values of *c*. Then, the *c* obtained from the all-object visualization is adapted as the suitable *c* for the medoids graph. The important difference between the medoids graph of the MSV and the neighborhood graph was meaningless axes of the medoids graph. With this property, a medoids graph of the MSV was more preferred and suitable than a neighborhood graph in any type of data set, especially a non-numerical data set.

Acknowledgment

This research was supported by the Ministry of Research, Technology, and Higher Education of Indonesia, Kemenristekdikti, and Osterreichischer Austauschdienst, OeAD, under the Indonesia Austria Scholarship Program.

References

[1] A.R. Webb and K. Copsey, *Statistical Pattern Recognition*, 3rd ed. West Sussex, UK: John Wiley and Sons, 2011, doi: 10.1002/9781119952954.

[2] A.K. Jain and J. V. Moreau, "Bootstrap Technique in Cluster Analysis," *Pattern Recognit.*, vol. 20, pp. 547–568, 1987, doi: 10.1016/0031-3203(87)90081-1 .

[3] Y. Fang and J. Wang, "Selection of the number of clusters via the bootstrap method," *Comput. Stat. Data Anal.*, vol. 56, no. 1, pp. 468–477, 2012, doi: 10.1016/j.csda.2011.09.003.

[4] S. Monti, P. Tamayo, J. Mesirov, and T. Golub, "Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data," *Mach. Learn.*, vol. 52, pp. 91–118, 2003, doi: 10.1023/A:1023949509487.

[5] J. Handl, J. Knowles, and D. B. Kell, "Computational cluster validation in post-genomic data analysis," *Bioinformatics*, vol. 21, no. 15, pp. 3201–3212, 2005, doi: 10.1093/bioinformatics/bti517.

[6] J. Ji, T. Bai, C. Zhou, C. Ma, and Z. Wang, "An improved k-prototypes clustering algorithm for mixed numeric and categorical data," *Neurocomputing*, vol. 120, pp. 590–596, 2013, doi: 10.1016/j.neucom.2013.04.011.

[7] X. Wu *et al.*, "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, 2008, doi: 10.1007/s10115-007-0114-2.

[8] K. Waiyamai and T. Kangkachit, "Constraint-based discriminative dimension selection for high-dimensional stream clustering," *Int. J. Adv. Intell. Informatics*, vol. 4, no. 3, pp. 167–179, Nov. 2018, doi: 10.26555/ijain.v4i3.271.

[9] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J.M. Perez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognit.*, vol. 46, no. 1, pp. 243–256, 2013, doi: 10.1016/j.patcog.2012.07.021.

[10] W. M. Rand, "Objective Criteria for the Evaluation of Clustering Methods," *J. Am. Stat. Assoc.*, vol. 66, no. 336, pp. 846–850, 1971, doi: 10.1080/01621459.1971.10482356.

[11] L. Hubert and P. Arabie, "Comparing Partitions," *J. Classif.*, vol. 2, no. 1, pp. 193–218, 1985, doi: 10.1007/BF01908075.

[12] M. Charrad, N. Ghazzali, V. Boiteau, and A. Niknafs, "NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set," *J. Stat. Softw.*, vol. 61, no. 6, pp. 1–36, 2014, doi: 10.18637/jss.v061.i06.

[13] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, 1987, doi: 10.1016/0377-0427(87)90125-7.

[14] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the Number of Clusters in a Data Set Via the Gap Statistic," *J. R. Stat. Soc. B*, vol. 63, no. 2, pp. 411–423, 2001, doi: 10.1111/1467-9868.00293.

[15] F. Leisch, "Handbook of Data Visualization," Chen, Hardle, and A. Unwin, Eds. Springer Verlag, 2008, pp. 561–587, doi: 10.1007/978-3-540-33037-0_22.

[16] G. Brock, V. Pihur, S. Datta, and S. Datta, "clValid: An R Package for Cluster Validation," *J. Stat. Softw.*, vol. 25, no. 4, 2008, doi: 10.18637/jss.v025.i04.

[17] F. Leisch, "A toolbox for K-centroids cluster analysis," *Comput. Stat. Data Anal.*, vol. 51, pp. 526–544, 2006, doi: 10.1016/j.csda.2005.10.006.

[18] F. Leisch, "Neighborhood graphs, stripes and shadow plots for cluster visualization," *Stat. Comput.*, vol. 20, pp. 457–469, 2010, doi: 10.1007/s11222-009-9137-8.

[19] G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis, "Algorithm for drawing graphs: An annotated bibliography," *Comput. Geom.*, vol. 4, no. 235–282, 1994, doi: 10.1016/0925-7721(94)00014-X.

[20] T. Kamada and S. Kawai, "An Algorithm for Drawing General Undirected Graphs," *Inf. Process. Lett.*, vol. 31, pp. 7–15, Apr. 1989, doi: 10.1016/0020-0190(89)90102-6.

[21] T. M. Fruchterman and E. M. Reingold, "Graph Drawing by Force-directed Placement," *Software-Practice Exp.*, vol. 21, no. 11, pp. 1129–1164, Nov. 1991, doi: 10.1002/spe.4380211102.

[22] Qiu and H. Joe, "Generation of Random Clusters with Specified Degree of Separation," *J. Classif.*, vol. 23, pp. 315–34, 2006, doi: 10.1007/s00357-006-0018-y.

[23] W. Qiu and H. Joe, "Separation Index and Partial Membership for Clustering," *Comput. Stat. Data Anal.*, vol. 50, no. 3, pp. 585–603, 2006, doi: 10.1016/j.csda.2004.09.009.

[24] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data*. New York, USA: John Wiley and Sons, 1990, doi: 10.1002/9780470316801.

[25] C. Hennig, "Cluster-wise Assement of Cluster Stability," *Comput. Stat. Data Anal.*, vol. 52, pp. 258–271, 2007, doi: 10.1016/j.csda.2006.11.025.

[26] M. Lichman, *UCI Machine Learning Repository*. 2013, available at: http://archive.ics.uci.edu/ml.

[27] R Core Team, *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing, 2015, available at: https://www.r-project.org/.

[28] W. Qiu and H. Joe, *clusterGeneration: Random Cluster Generation (with Specified Degree of Separation). R package version 1.3.4*. 2015, available at: https://CRAN.R-project.org/package=clusterGeneration.

[29] M. Maechler, P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik, *cluster: Cluster Analysis Basics and Extensions. R package version 2.0.6 --- For new features, see the "Changelog" file (in the package source)*. 2017, available at: https://cran.r-project.org/package=cluster.

[30] W. Budiaji, *kmed: Distance-Based k-Medoids. R package version 0.2.0*. 2019, available at: https://cran.r-project.org/package=kmed.

[31] H. Wickham, *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer-Verlag, 2016, doi: 10.1007/978-3-319-24277-4_9.

[32] S. Tyner and H. Hofmann, *geomnet: Network Visualization in the "ggplot2" Framework. R package version 0.2.0*. 2016, available at: https://cran.r-project.org/package=geomnet.

Supplementary Material

All materials used in this paper are deposited in the OSF author account to guarantee open access and reproducible research in the Medoid-Based Shadow Value Validation and Visualization project (doi: 10.17605/OSF.IO/9XAH8).