ZANCO Journal of Pure and Applied Sciences

TYPE Original article
PUBLISHED 05 /02/2024
DOI http://dx.doi.org/10.21271/ZJPAS.36.1.5

OPEN ACCESS

*Corresponding author
Polla Fattah
polla.fattah@su.edu.krd

RECEIVED :23 /11 /2022
ACCEPTED :27/08/ 2023
PUBLISHED :05/ 02/ 2024

KEYWORDS:
Clustering, clustering validation index, internal validation index, region-based clustering validation index, clustering cohesion.

A New Approach to Cohesion Measurement: Region-Based Clustering Validation

Sakar Salar Salih, Polla Fattah*

Department of Software Engineering and Informatics, College of Engineering, Salahaddin University- Erbil, Kurdistan Region, Iraq

ABSTRACT

Clustering assigns objects to clusters based on similarity, aiming to ensure that objects within the same cluster are similar and those in different clusters are dissimilar. Evaluating clustering quality is crucial and challenging. Thus, researchers have proposed clustering validation indices namely internal and external validation indices. Internal indices assess clustering quality using intrinsic information within a dataset. We focus on internal validation indices for their real-world applicability. In this paper, we have proposed a novel region-based internal validation (RCV) index. Our index incorporates the division of each cluster into three distinct regions which are the inner, middle, and outer regions. according to the clusters' center and their corresponding radius, we split each cluster into the aforementioned regions. The average distance is then computed for each region, and a penalty factor is applied to these average distances. By summing up the three penalized average distances, a Region Cluster Validation (RCV) score is obtained for each cluster. The RCV scores for all clusters are then summed together to yield an overall measure of cluster validity. A lower index value indicates better clustering quality. Experiment results on the synthetic and real-world datasets exhibit the usability and effectiveness RCV index.

1. Introduction

Clustering algorithms are a widely-used technique in various fields of science (Jain et al., 1999, Xie and Beni, 1991, Han et al., 2022). The purpose of clustering is to partition unlabeled data into separate groups based on their similarities and dissimilarities. The essence of the grouping is to uncover the underlying structure or patterns of a dataset (Jain et al., 1999) in such that data elements within a group are similar and in different groups are dissimilar. Since there is no common agreement on what constitutes quality clustering, therefore, assessing the quality of clustering results has become a challenging task. To address this issue, the researchers have attempted to develop and propose various clustering validation indices to evaluate the effectiveness of clustering algorithms. The sole aim of clustering validation measurements is to generate meaningful and valuable clusters (Jain et al., 1999, Halkidi et al., 2001) in a way that the performance of clustering algorithms is determined based on intra-cluster similarity and inter-cluster dissimilarity (Halkidi et al., 2001, Liu et al., 2013). As a result, obtaining the optimal number of clusters (Deborah et al., 2010). In the bellow, we briefly describe the two types of clustering validation indices (Arbelaitz et al., 2013). Internal validation: it assesses the goodness of clustering results based on the intrinsic information that contains in the data itself. External validation: it depends on external information when evaluating the goodness of a clustering structure.

The focus of our research is on internal validation measurement. Internal validation mainly depends on intrinsic information existing within the dataset to assess the quality of clustering results. The internal validation index is preferred over the external index because it does not require prior information on the classes or labels of a given dataset (Jain and Dubes, 1988, Halkidi et al., 2001). For this reason, the internal validation index is more realistic in real-world applications (Clarke, 1974, Hennig, 2015). It might be worth mentioning that the internal validation index directly relies on compactness and separation concepts. According to (Jain and Dubes, 1988, Zhao and Karypis, 2004) separation refers to the degree of dissimilarity of objects of different clusters (inter-cluster dissimilarity) whereas cohesion indicates the similarity of objects within a cluster (intra-cluster similarity).

Despite numerous attempts in the recent decades to propose new internal validation indices including some common ones (Halkidi et al., 2001, Kim and Ramakrishna, 2005) and recent ones (Fu and Wu, 2016, Guo et al., 2016, Wang and Xu, 2019), based on our knowledge the literature, the field still lacks a simple and straightforward index that accounts for the region of clusters when evaluating clustering performance. Existing indices do not fully incorporate the notion of dividing the clusters into separate regions. Thus, we regard our suggested approach as an innovative index that considers the idea of cluster regions during the validation process and performance evaluation of clustering algorithms. Hence, we have labeled our proposed method as the Region-Based Clustering Validation (RCV) Index.

Region-based clustering validation index split each cluster into regions. Firstly, we partition each cluster into three distinct regions, namely the inner, middle, and outer regions, based on their distance from the cluster centroid. We then compute the average distance of the data elements in each region using the Euclidean method (Bishop, 2006). To account for the varying importance of each region in determining the quality of the clustering results, we apply a penalty factor by multiplying the average distances of the inner, middle, and outer regions by one, two, and three, respectively. Notably, the outer region incurs the highest penalty due to its greater average distance from the centroid. Subsequently, we sum up the three penalty-weighted distances to obtain the RCV score for a certain cluster, which serves as a measure of the cohesion of the cluster. The lower the index score, the better the clustering quality and cohesion.

The objective of RCV is to assess and validate the clustering configuration quality. It is worth noting that the proposed index could be used specifically to measure the cohesion of clusters.

To exhibit the applicability and performance of our index we have conducted a series of experiments on both synthetic and real-world data. The results verify the effectiveness of RCV in validation clustering.

The subsequent sections of this paper are organized as follows. In section two we review the current state of the art in the literature of internal validation indices including the most common ones to some recently proposed methods. In section three we have described the proposed index. In section four we have conducted an experiment on our index as well as performing the analytical experiment on the four common indices, then, a detailed finding is presented. Finally, in section five we conclude this paper.

2.Related work

In this section, we aim to provide a review of the most commonly used internal validation indices, as well as some recently developed ones that have undergone testing. To the best of our knowledge, these indices provide a comprehensive representation of validation measures across various disciplines.

One of the earliest works in 1974 is the Dunn index which computes the ratio of the minimum inter-cluster and intra-clusters distances. A larger value indicates better clustering quality (Dunn, 1974). The Dunn index is sensitive to noise and data dimensionality. Other works later in the same were published by (Caliński and Harabasz, 1974). Calinski–Harabasz index calculated the ratio of inter-cluster and inter-cluster variance. The index is sensitive to large-scale datasets and it performs better when there are larger numbers of clusters. In 1975, Baker and Hubert proposed the Gamma index which examines the correlation between two different objects within the same cluster. The range of the index value is between zero to one. A higher value of the index indicated a better clustering configuration (Baker and Hubert, 1975). A balance between cluster sizes is preferable and the index might not perform ideally when the clusters are not extensively overlapped. The Silhouette index, which was proposed by Rousseeuw in 1987, compares compactness and separation. The range of the index value is between a negative one and to positive one in which a higher value indicates a better clustering quality (Rousseeuw, 1987). The dataset features such as scale, shape, and density might be an issue for the Silhouette index. In 1989, the Root-Mean-Square Standard Deviation was proposed by Sharma. The index computes the distance between data points and their corresponding centers. The computed value will be normalized by the standard deviation of the distance. The lower values represent better cluster configuration (Sharma, 1995). This index assumes that the data points are evenly distributed around their corresponding centroids and they work on simple shapes and data structures. The R Squared is also proposed by Sharma which measures the dissimilarity between clusters. The R Squared produces two different values which are one and zero where zero implies that there is no difference between clusters and one indicates otherwise (Sharma, 1995). The R Squared index is sensitive to the size and distribution of the data points.

In 2001, Halkidi and Vazirgiannis proposed the S_Dbw index which is well suited for a dataset that has compact and well-separated groups. The cluster variance and density are used to compute both cluster compactness and separation respectively (Halkidi et al., 2001). The S_Dbw index might not perform well with large data sizes and the objects are spherical. Chou et al. introduced the CS index in 2004. The CS index measures both compactness and separation through cluster diameter and nearest neighbor distance respectively. The smallest value of SC indicates a higher quality of clustering configuration (Chou et al., 2004). This index has high computational overload and it is sensitive to noise. In 2007, Saitta et al. introduced the score function index. it depends on the within and between class distances concepts. This index is capable of detecting a lack of division in a dataset (Saitta et al., 2007). The score function index is sensitive when encountering large-scale datasets and arguably produces high computational overload. The Point symmetry index was proposed by Bandyopadhyay and Saha in 2008. The index computes clustering configuration based on the similarity of the points (Bandyopadhyay and Saha, 2008). It is capable of identifying both convex and nonconvex clusters regardless of clusters' sizes and shapes. In 2010, Gurrutxaga et al. proposed the SEP/COP index which assesses the quality of hierarchical clustering. To do so the index depends on the number of clusters and inter-cluster distances. The index is partially affected by noise (Gurrutxaga et al., 2010). A crisp clustering validation index was proposed by Lago-Fernández and Corbacho in 2010. It measures the average normality of clusters. To determine the normality of a cluster the negentropy is used (Lago-Fernández and Corbacho, 2010). The index needs extensive computational power and might be solid to noise and outliers.

To account for clusters with significant variations in size and density in 2011, Žalik and Žalik introduced the SV-index. The index calculates inter-cluster and intra-cluster distance (Žalik and Žalik, 2011). The index might perform better when it deals with different cluster shapes. In 2016, Fu and Wu proposed an index that assesses the clustering results of high dimensional Boolean data. It can be applied to categorical data that could be transformed into Boolean data. The data are categorized into three classes to represent zero, one, and not the same for all the objects in the dataset (Fu and Wu, 2016). It might not be easy to well categorize complex datasets and structure them into three classes. In 2017, in an attempt to develop a simple and robust validation index, Jauhiainen and Kärkkäinen proposed the kCE index. The authors claim that the index has a maximum coverage (Jauhiainen and Kärkkäinen, 2017). The index can detect the absence of data partitioning which means that the dataset comprises a single cluster. The structural dissimilarity index is proposed by Guo et al. in 2017 to validate clusters of categorical sequences. The method utilizes a probabilistic approach to evaluate the structure dissimilarity between the sequences. The index computes both within-cluster compactness and between-cluster separation (Guo et al., 2016). In 2019, Wang and Xu proposed the Peak Weight Index which incorporates the data separation and aggregation features of the Silhouette and Calinski-Harabasz indices and computes the highest value of the two indices as a reference point to assign a suitable weight within a specific range (Wang and Xu, 2019). In the same year, Misuraca, Spano, and Balbi proposed the BMS index for document clustering. This index is an improved version of the Dunn index in which cosine dissimilarity is applied to the original Dunn index to assess the quality of clustering quality (Misuraca et al., 2019). In 2021, Ncir et al. introduced the Scalable Dunn index. it computes the Dunn index on a scalable and parallel structure. To do so, multiple nodes will be utilized to distribute the computational process. To reduce computational overhead a small sampling technique will be used when dealing with a large dataset (Ncir et al., 2021).

3. Proposed Method

We have proposed a novel region-based internal clustering validation index. Our approach is based on dividing a cluster into regions. Hence, we have named the proposed method the Region-based Cluttering Validation index. To do so, we partition each cluster into three distinct regions, namely the inner, middle, and outer regions, based on their distance from the cluster centroid. We have decided to divide into three regions for two reasons. Firstly, the size of the clusters is small. Secondly, dividing a cluster into three regions requires less computational overhead and time. It is worth to consider that when one deals with large scale data, the number of regions could vary based on the data requirements. To calculate the distance or boundary of each region, we compute the average distance of the data elements in each region using the Euclidean method (Bishop, 2006) in which the distance between the centroid and data points of corresponding clusters will be computed. After that, the maximum distance will be selected as the cluster's radius. The computed maximum distance will be divided by three to generate the three regions. Figure 1 illustrates the concept of partitioning the region of clusters.

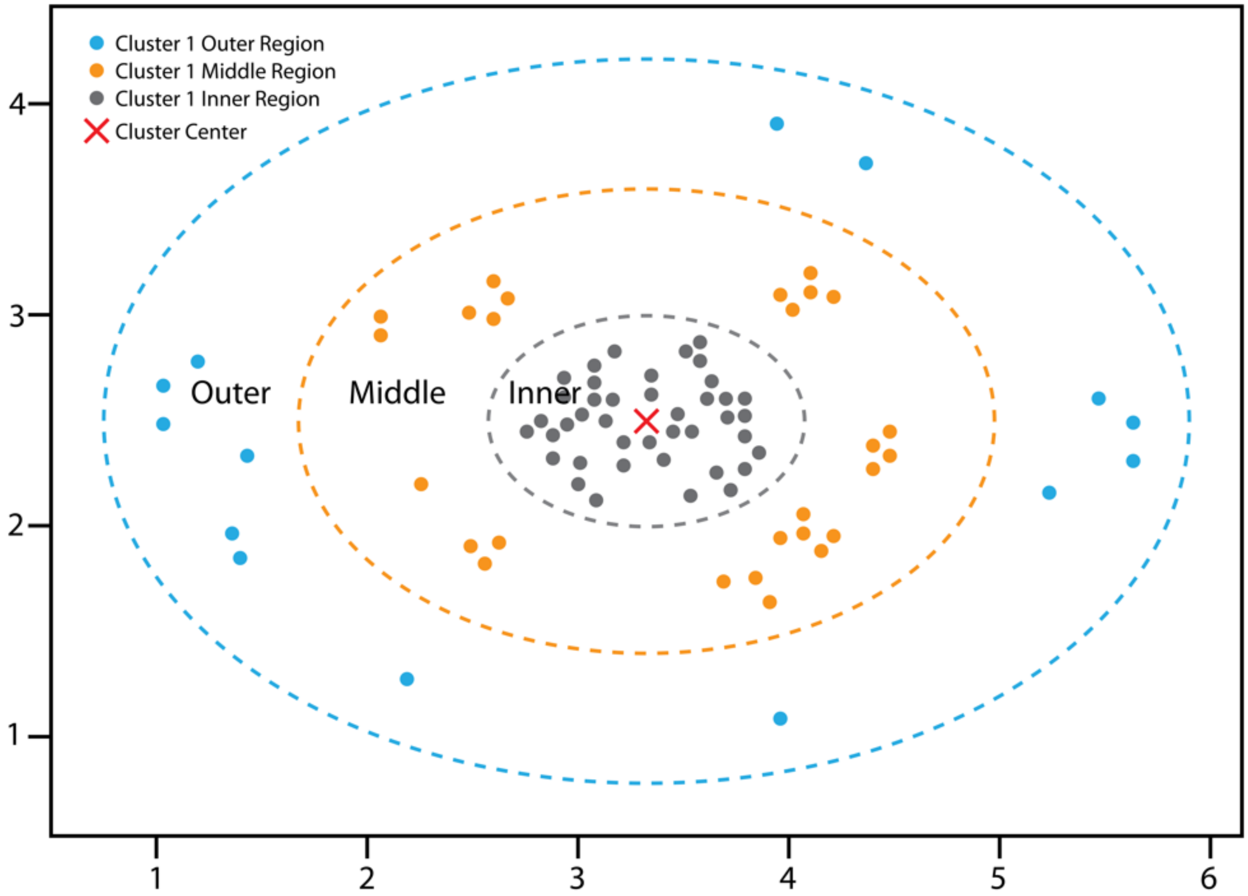

Figure 1: dividing a cluster into three regions

To account for the varying distance indicator of each region, we apply a penalty factor by multiplying the average distances of the inner, middle, and outer regions by one, two, and three, respectively. These penalty factors are hyperparameters that are manually tuned. Notably, the outer region incurred the highest penalty due to its greater average distance from the centroid. Subsequently, we sum up the three penalty-weighted distances to obtain the RCV score for a certain cluster, which serves as a measure of the cohesion of the cluster. The lower the RCV score, the better the clustering quality and cohesion. It is worth mentioning that RCV is capable of validating diverse clustering configurations that might include numeric and multi-dimensional elements.

The algorithmic steps of the RCV approach are described below points: -

Step 1: Using the Elbow method (Thorndike, 1953) to specify the number of cluster k.

Step 2: Clustering with various clustering algorithms to perform data labeling.

Step 3: Define the centroid of the obtained clusters.

Step 4: Using the Euclidean method [28] to find the distance between the centroid and points for each cluster. Then, select the maximum distance as the cluster’s radius. The Euclidean method is calculated according to the following equation.

$$d(a,b) = \sqrt{\sum_{i=1}^{n} (a-b)^2}$$

where a and b are two data points of the dataset:

Step 5: Partition each cluster according to its radius into three distinct and equivalent regions namely inner, middle, and outer regions.

Step 6: Compute the average distance for the inner, middle, and outer regions.

Step 7: Measure the RCV score which computes the cohesion value for each cluster. The RCV score is calculated according to the following equation.

$$\text{Total RCV} = \sum_{c} \sum_{i,w=1}^{3} \operatorname{avg} d(r_i) * w$$

Where: r=region, i= region number, avg d is average distance and w=weight which equals to 1,2,3 according to the inner, middle, and outer regions respectively. And c is a cluster.

Figure 2 illustrates the algorithmic steps of the RCV index in detail.

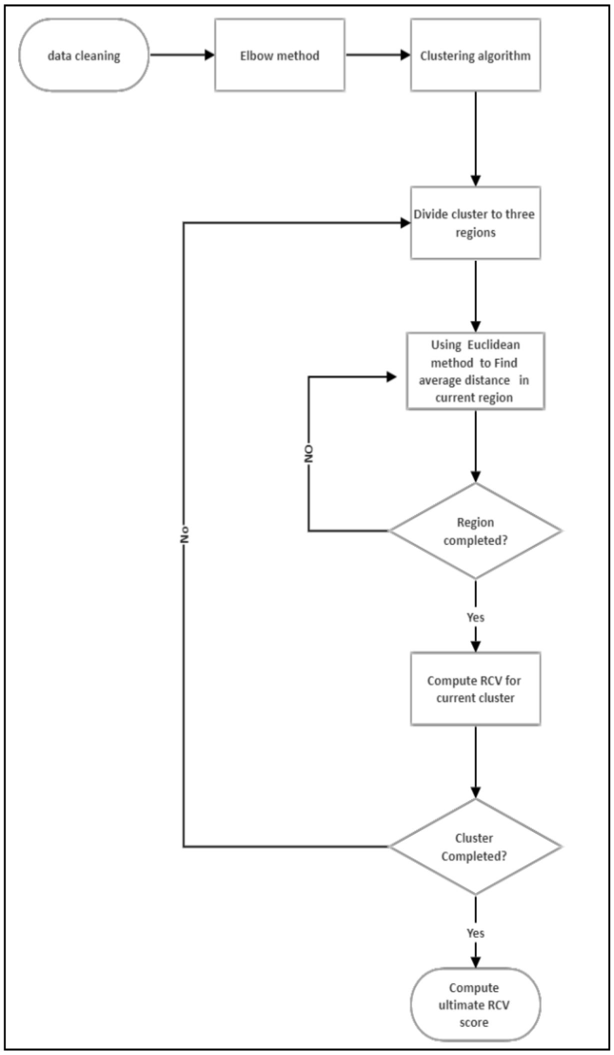
data cleaning
Elbow method
Clustering algorithm
Divide cluster to three regions
Using Euclidean method to Find average distance in current region
NO
No
Region completed?
Yes
Compute RCV for current cluster
Cluster Completed?
Yes
Compute ultimate RCV score

Figure 2: RCV steps

4.Results and Findings

We have categorized our results and findings into two major sections. The first section encompasses the results of four popular indices, namely Dunn, Davies-Bouldin, Calinski-Harabasz, and Silhouette indices which hereinafter we refer them as the four common indices for the purpose of this paper. To ensure inclusivity in our findings, we have divided this section into two subsections based on the dataset utilized. Firstly, we evaluate the performance of k-means and fuzzy c-means clustering algorithms on a synthetic dataset. Then, we employ the same indices to evaluate the performance of the clustering algorithms using a real dataset.

In the second section, we have divided our experiment into three subsections to assess the performance of k-means and fuzzy c-means clustering algorithms by using our proposed method. In the first subsection, we use one synthetic dataset to validate the clustering results. In the following subsection, we have used two synthetic datasets. In each dataset either well compacted or well separated. In the final subsection, we use the IRIS dataset to validate clustering results.

It is worth mentioning that for the real data we have used the IRIS dataset. IRIS dataset is available online for experimental purposes on the Machine Learning Repository, Center for Machine Learning and Intelligent Systems from the University of California Irvine. In the following sections, we present the results and findings of our experiment.

4.1. Validating the performance of k-means and fuzzy c-means by using four common indices

In this section, we attempt to validate the performance of k-means and c-means clustering algorithms by using the four common indices. As previously mentioned in the first stage we use a synthetic dataset followed by a real dataset. The following subsection presents more details about our findings.

4.1.1. Using Synthetic Data to validate k-means and fuzzy c-means

We created a two-featured dataset in order to assess the performance of k-means and fuzzy c-means clustering algorithms by using the four common indices. We used k-means and Fuzzy C-means clustering algorithms to classify the dataset. Figure 3 shows the representation of the clustering of the dataset.

Typically, the aforementioned indices are experimented with the aim of obtaining an optimal number of clusters for a given k value.

According to the assessment conducted using the k-means algorithm, the results yielded by the Dunn index indicated that the dataset contains five clusters, which is a misleading indication. We are already aware that the dataset only comprises two clusters. Despite this anomaly, the remaining clustering indices performed well, correctly identifying the appropriate number of clusters, which is two. Table 1 shows the results of the assessment for the k-means algorithm.

We applied the same four indices to evaluate the performance of the fuzzy c-means algorithm. Notably, the Dunn index performed slightly better in comparison to its results when applied to the k-means algorithm. However, it yielded a slight discrepancy as it indicated the presence of three clusters rather than two. Despite this minor anomaly, the remaining indices accurately identified the optimal number of clusters, which is two. Table 2 presents the results of the assessment.

In general, when k is set to 2 for both the k-means and c-means algorithms, they perform similarly, generating comparable scores across all indices, with only minor exceptions observed in the Dunn index. The unsatisfactory performance of the Dunn index could be attributed to its susceptibility to noise and outliers that may exist in the dataset. Moreover, it can be challenging to definitively determine which algorithm outperforms the other. Thus, we have concluded that both clustering algorithms are equally suitable for this type of dataset. To further illustrate the findings, the charts are presented in Figure 4

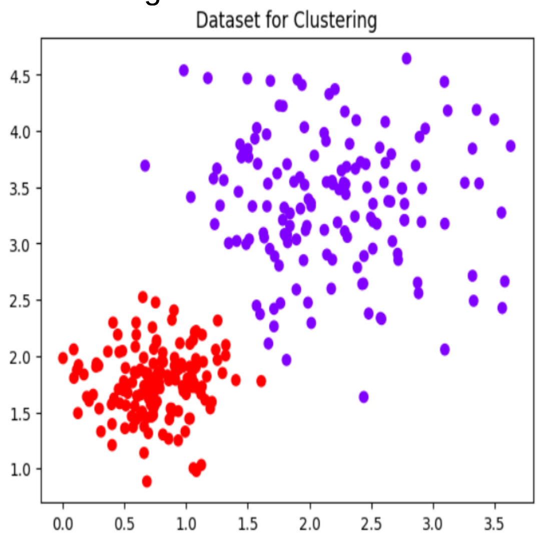

Figure 3. Synthetic dataset

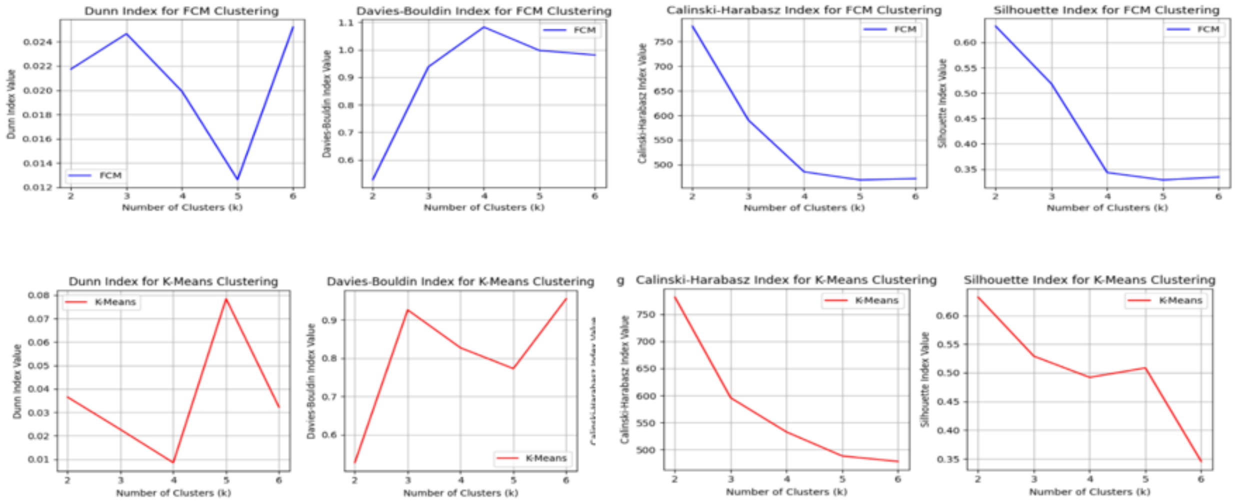

Figure 4. Analysis the 4 indices for k-means and fuzzy c-means in synthetic dataset

Table 1. Results of k-means on synthetic dataset

k	Dunn	Davies-Bouldin	Calinski-Harabasz	Silhouette
2	0.036463	0.527181	781.473532	0.632042
3	0.022625	0.925935	595.861747	0.529001
4	0.008474	0.826952	532.952627	0.492097
5	0.078445	0.772733	488.657698	0.508589
6	0.032173	0.955061	478.566068	0.345388

Table 2. Results of fuzzy c-means on synthetic dataset

k	Dunn	Davies-Bouldin	Calinski-Harabasz	Silhouette
2	0.021747	0.528163	781.181973	0.631880
3	0.024658	0.937837	590.887466	0.518372
4	0.019902	1.082470	485.587450	0.343317
5	0.012626	0.997604	469.123769	0.328814
6	0.025209	0.981206	471.774371	0.334467

4.1.2. Using Real Dataset to validate k-means and fuzzy c-means

In this subsection, an IRIS dataset is used to evaluate the performance of the k-means and fuzzy c-means clustering algorithms. To do so, the four common indices are used. The outcome of clustering algorithms is depicted in Figure 5.

Based on the results, it appears that k-means outperforms fuzzy c-means by a small margin when using the IRIS dataset. Among the various values of k when k is equal to four can be considered the optimal choice for k-means clustering as it generates the highest scores for both the Calinski-Harabasz and Silhouette indices. However, in terms of the Dunn and Davies-Bouldin indices, the best performance is achieved when k equals three, which can be regarded as an optimal value. The outcomes of the evaluation of the k-means and fuzzy c-means algorithms are presented in Table 3 and Table 4

The visual representation of the analysis and findings described above can be observed in Figure 6

Table 3. Results of k-means on IRIS dataset

k	Dunn	Davies-Bouldin	Calinski-Harabasz	Silhouette
2	0.076506	0.404293	513.924546	0.681046
3	0.098807	0.661972	561.627757	0.552819
4	0.136543	0.780307	530.765808	0.498051
5	0.082339	0.805965	495.541488	0.488749
6	0.082903	0.925770	473.515454	0.367846

Table 4. Results of fuzzy c-means on IRIS dataset

k	Dunn	Davies-Bouldin	Calinski-Harabasz	Silhouette
2	0.076506	0.404293	513.924546	0.681046
3	0.104973	0.669247	560.223502	0.549518
4	0.037346	0.863877	415.325179	0.412729
5	0.082339	0.816398	494.504045	0.487436
6	0.060302	0.926120	473.379885	0.364001

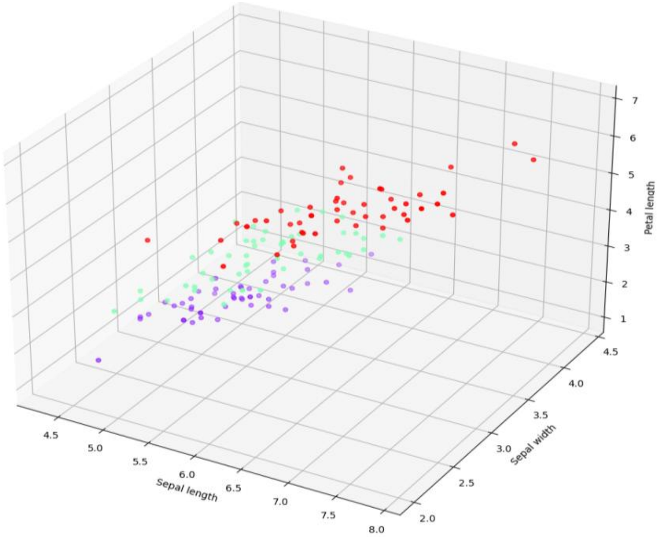

Figure 5: iris dataset

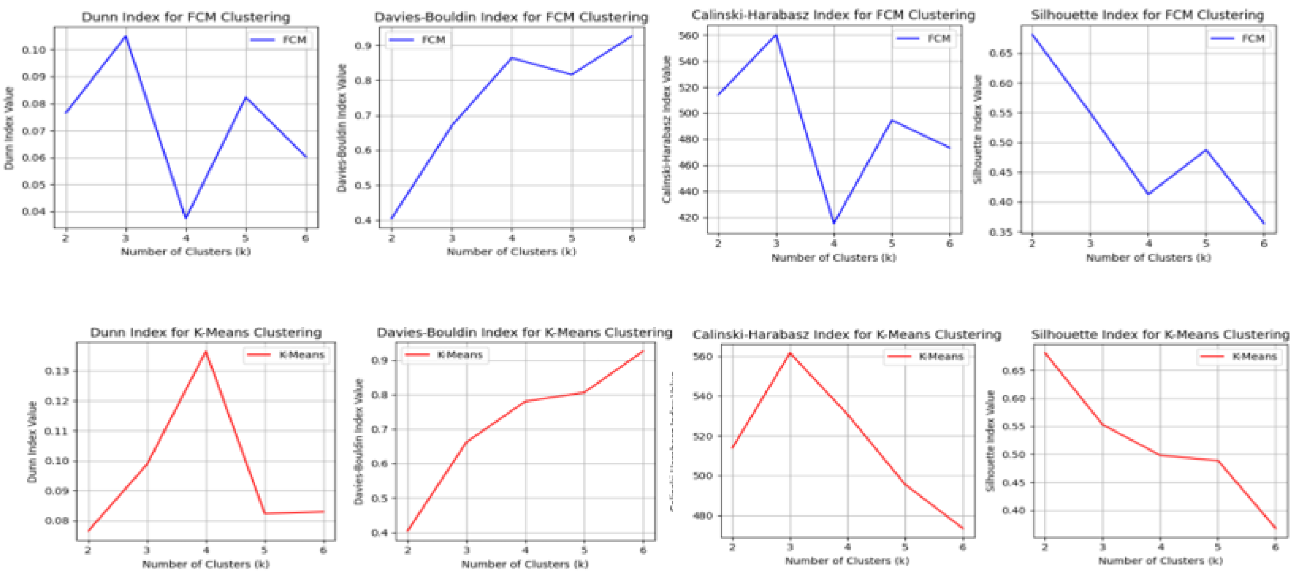

Figure 6: Analysis the 4 indices for k-means and fuzzy c-means in Iris dataset

4.2. Validating the performance of k-means and fuzzy c-means by applying RCV

In this section, we have divided our experiment into three subsections to assess the performance of k-means and fuzzy c-means clustering algorithms by using our proposed method. In the first subsection, we use one synthetic dataset to validate the clustering results of the clustering algorithms. In the following subsection, we have used two synthetic datasets. In each dataset either well compacted or well separated. In the final subsection, we use the IRIS dataset to validate clustering results.

4.2.1. Using a synthetic dataset to validate k-means and fuzzy c-means

We have used the same synthetic dataset as in section (4.1.1). To evaluate the clustering performance of k-means and fuzzy c-means, we use the RCV index. RCV divides the cluster into three distinct regions: the inner, middle, and outer regions, as shown in Figures 7 and 8. Our findings revealed that both the k-means and fuzzy c-means clustering algorithms produced satisfactory results, with only minor discrepancies between them. Because as shown in Figure 8 after applying fuzzy c—means, one data point will be added to cluster1 the RCV Cohesion score is changed from 4.352 to 7.793.

For cluster 1, the k-means algorithm computes a lower cohesion score and average distance across all three regions as compared to the fuzzy c-means algorithm. Therefore, k-means appears to be a more suitable option for the given clustering scenario. The results are presented in Table 5.

For cluster 2, both algorithms have produced similar results, with fuzzy c-means having a slightly lower cohesion and average distance in the inner and middle regions. However, k-means has a larger radius and therefore may be well suited for identifying outliers in this cluster. The results are shown in Table 6.

For more clear illustration the previous results are graphed in Figure 9 the cohesion score for each cluster is presented for each clustering algorithm namely the k-means and fuzzy c-means.

Table 5. RCV for cluster1 in synthetic dataset

indicators	k-means	Fuzzy c-means
Average distance for inner region	0.247	0.343
Average distance for middle region	0.577	0.845
Average distance for outer region	0.983	1.920
RCV Cohesion score	4.352	7.793

Table 6. RCV for cluster 2 in synthetic dataset

	k-means	Fuzzy c-means
Average distance for inner region	0.409	0.405
Average distance for middle region	0.925	0.922
Average distance for outer region	1.610	1.576
RCV Cohesion score	7.091	6.977

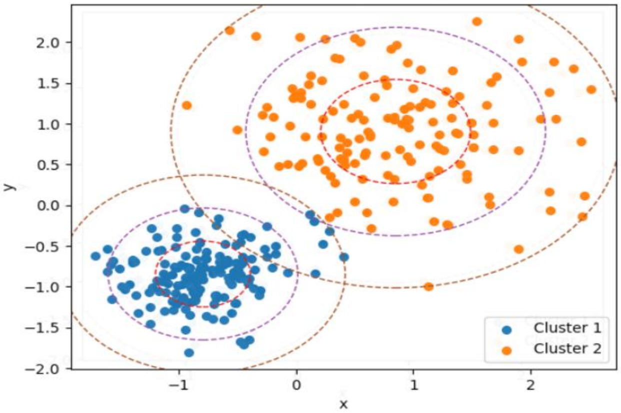

Figure 7. RCV after apply k-means clustering

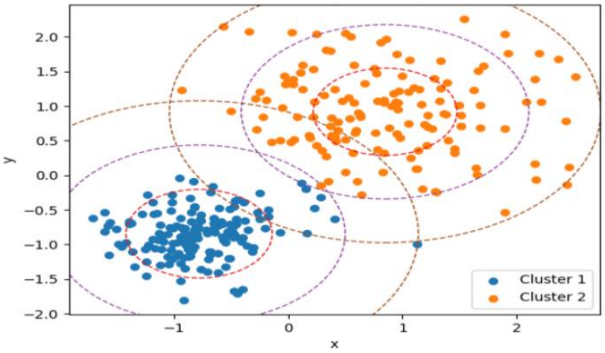

Figure 8. RCV after apply fuzzy c--means clustering

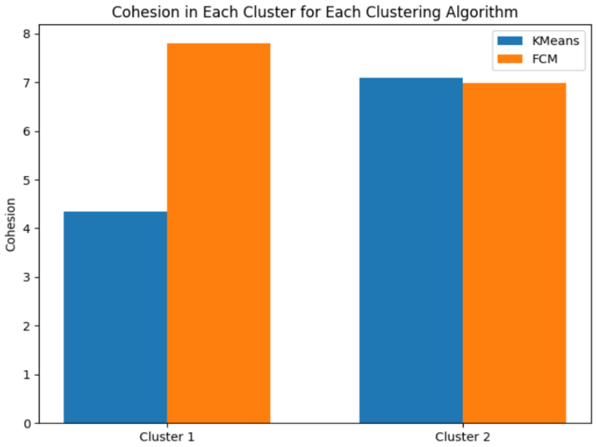

Figure 9: RCV Cohesion score for k-means and fuzzy c-means

4.2.2. RCV Discriminatory Analysis for Compacted and Well-Separated Datasets

We have generated two synthetic datasets by using Python namely data1 and data2. In the first dataset (data1), we deliberately arranged the objects to be well-compacted. Conversely, in the second dataset (data2), the clusters exhibit significant distribution and separation of their objects, as depicted in Figure 10. Both data1 and data2 comprise two clusters and random values are chosen for cluster coordination.

The clusters of the data1 are generated without the scaling factor. Each cluster is centered around a different mean point. The first cluster of data1 is centered around [3,3,3] and the second cluster is centered around [6,6,6]. The clusters of data2 are generated by a scaling factor 4 and shifted by a specific mean point for each cluster in which the first cluster is centered around [3, 3, 3], while the second cluster is centered around [20, 20, 20].

The experimental results for both the k-means and fuzzy c-means algorithms, as presented in Table 7, demonstrate a higher RCV score for data2, while the cohesion score for data1 is notably lower. This observation clearly distinguishes the compactness of clusters in data1 and the separation of clusters in data2 which is considered a significant observation.

The RCV scores from Table 8 reveal that the k-means and fuzzy c-means algorithms exhibit similar performance in terms of RCV score for data1. This suggests that both algorithms effectively compute the compactness of the clusters. Conversely, for data 2, a notable disparity in RCV scores emerges. The k-means algorithm yields a lower RCV score compared to the fuzzy c-means algorithm. This disparity indicates that the k-means outperforms the fuzzy c-means in assessing the cohesion of the clusters. It is worth noting that a lower RCV score indicates a more accurate assessment of cluster cohesion. Figure 11 illustrates the results for both clusters in each dataset.

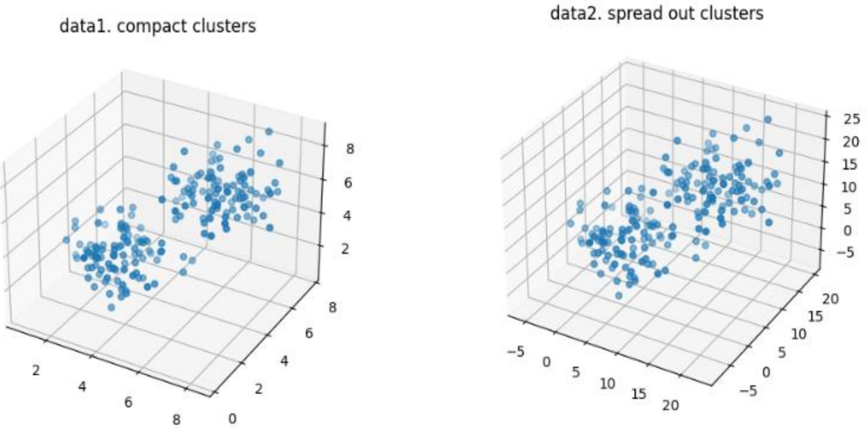

Figure 10: Compacted and Well-Separated Datasets

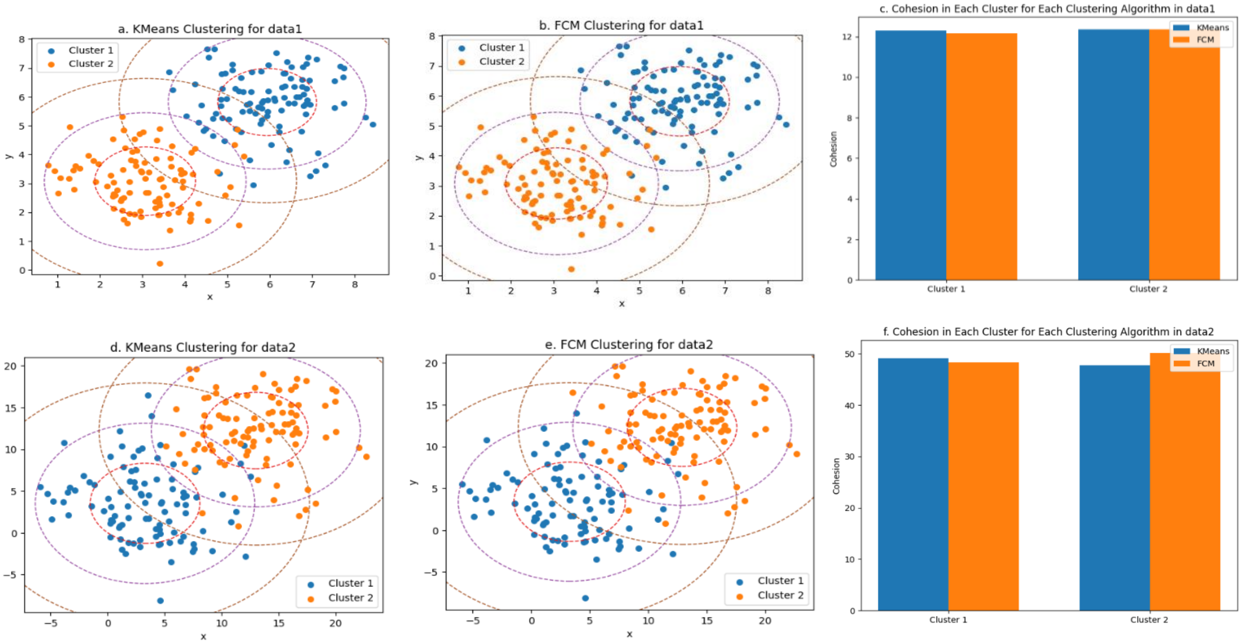

***Figure 11:** results for both clusters in each dataset*

Table 7. RCV score for both datasets

Clustering algorithm	Data1 total RCV cohesion	Data2 total RCV cohesion
K-means	24.635	96.912
Fuzzy c-means	24.504	98.442

Table 8. RCV for cluster 1 in real dataset

	k-means	Fuzzy c-means
Average distance for inner region	0.411	0.404
Average distance for middle region	0.779	0.786
Average distance for outer region	1.420	1.528
RCV Cohesion score	6.228	6.561

4.2.3. Using real data to validate k-means and fuzzy c-means

In this subsection, we have evaluated the performance of the two clustering algorithms by using the IRIS dataset. We have found that the cohesion scores produced for k-means and fuzzy c-means algorithms are different. Fuzzy c-means have scored slightly higher value, hence, is considered less satisfactory. Therefore, k-means outperforms fuzzy c-means in terms of our measurement for cohesion value.

As demonstrated in Figures 12 and 13, three distinct clusters were identified using k-means and fuzzy c-means algorithms, after applying the RCV method.

For cluster 1, both k-means and fuzzy c-means algorithms generated similar average distances for the three regions. However, in terms of cohesion score, k-means computed a slightly lower score, indicating higher cluster compactness. The results are presented in Table 8.

For cluster 2, both algorithms have produced similar values for cohesion and average distance for each region which indicates that the clusters are explicitly and consistently defined across both k-means and fuzzy c-means algorithms. The results are shown in Table 9.

For cluster 3, the k-means algorithm has a lower cohesion value and smaller radius which means that the cluster is tightly packed together and well-defined compared to fuzzy c-means. The results are shown in Table 10.

The results of the previous tables are depicted in Figure 14 for a clearer presentation. By using the RCV method we have clearly differentiated between k-means and fuzzy c-means algorithms for each cluster.

Table 9. RCV for cluster 2 in real dataset

	k-means	Fuzzy c-means
Average distance for inner region	0.281	0.269
Average distance for middle region	0.573	0.559
Average distance for outer region	1.013	1.013
RCV Cohesion score	4.466	4.427

Table 10. RCV for cluster 3 in real dataset

	k-means	Fuzzy c-means
Average distance for inner region	0.339	0.391
Average distance for middle region	0.708	0.730
Average distance for outer region	1.298	1.337
RCV Cohesion score	5.649	5.861

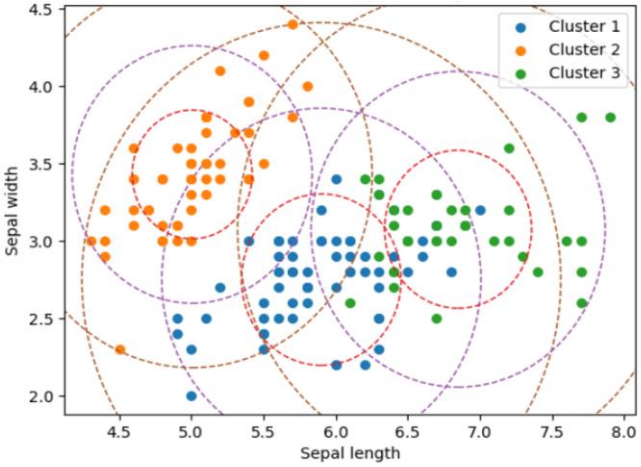

Figure 12: RCV after apply k-means clustering for Iris dataset

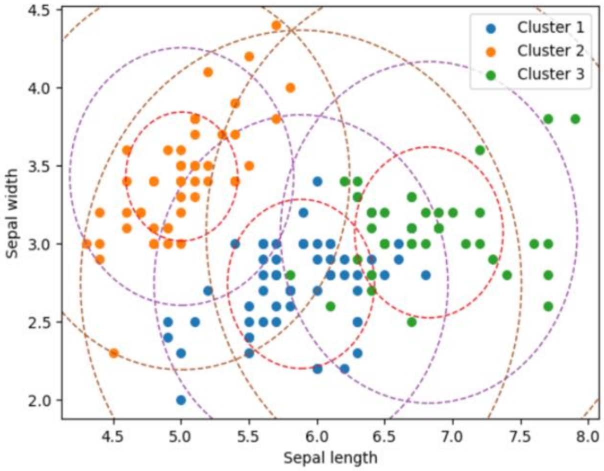

Figure 13: RCV after apply fuzzy c-means clustering for Iris dataset

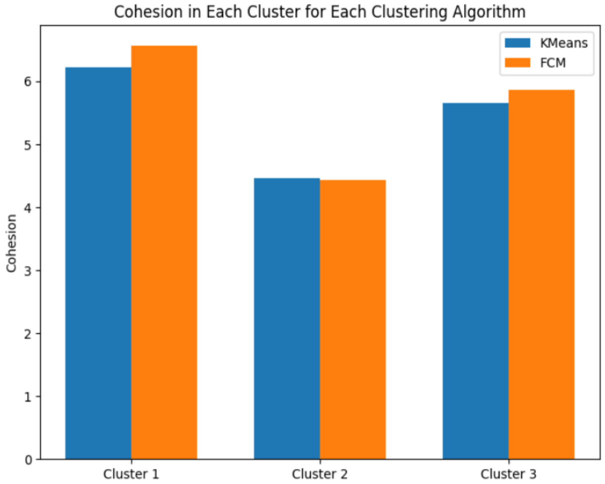

Figure 14: RCV Cohesion score for k-means and fuzzy c-means

Conclusions

In conclusion, evaluating clustering quality remains a challenging task. There is no consensus or guarantee among researchers to agree on a credible and reliable validation index. Yet, they acknowledge that the internal validation index is more applicable to real-world scenarios as it depends on the intrinsic information of a dataset when assessing or evaluating clustering configuration. For this reason, we have focused on internal validation indices. In an unprecedented attempt, we have proposed a novel region-based clustering validation approach under the name of the RCV index. RCV considers each cluster as the regions of data namely inner, middle, and outer regions. RCV relies on a penalty factor to define the boundaries of each region and its significance in clustering configuration. Then, we compute the RCV score to determine the clustering cohesion for each cluster to clarify the clustering results. Based on our experimental results obtained from both synthetic and real-world datasets, we have derived the following findings. Firstly, the validation process of the RCV index is more efficient in terms of time and computational overhead compared to the four commonly used indices. This advantage stems from RCV's ability to determine a single K value using the Elbow method, eliminating the need for a predefined set or sequence of K values. Furthermore, RCV demonstrates superior accuracy and precision in detecting subtle differences in clustering formations produced by the k-means and fuzzy c-means algorithms. In other words, RCV is highly sensitive to small changes or variations in the clustering results. This is evident through the emergence differentiations observed in the RCV scores. Lastly, our experiments on synthetic datasets have shown that the distance or proximity between clusters does not significantly impact the RCV value or the clustering validation process. However, the standard deviation of the clusters does affect the validation process.

In future work, we attempt to use RCV in an unsupervised classification environment, using it in different applications and data analysis. More precisely we intend to work on the separation feature of the index and then compare its performance with the four commonly used indices.

Acknowledgment: Not applicable.

Financial support: No financial support.

Potential conflicts of interest. All authors report no conflicts of interest relevant to this article.

References

Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J. M. & Perona, I. 2013. An extensive comparative study of cluster validity indices. *Pattern recognition,* 46**,** 243-256.

Baker, F. B. & Hubert, L. J. 1975. Measuring the power of hierarchical cluster analysis. *Journal of the American Statistical Association,* 70**,** 31-38.

Bandyopadhyay, S. & Saha, S. 2008. A point symmetry-based clustering technique for automatic evolution of clusters. *IEEE Transactions on Knowledge and Data Engineering,* 20**,** 1441-1457.

Bishop, C. 2006. Pattern recognition and machine learning. *Springer google schola,* 2**,** 5-43.

Caliński, T. & Harabasz, J. 1974. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods,* 3**,** 1-27.

Chou, C.-H., Su, M.-C. & Lai, E. 2004. A new cluster validity measure and its application to image compression. *Pattern Analysis and Applications,* 7**,** 205-220.

Clarke, M. 1974. Pattern classification and scene analysis. Wiley Online Library.

Deborah, L. J., Baskaran, R. & Kannan, A. 2010. A survey on internal validity measure for cluster validation. *International Journal of Computer Science & Engineering Survey,* 1**,** 85-102.

Dunn, J. C. 1974. Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics,* 4**,** 95-104.

Fu, L. & Wu, S. 2016. An internal clustering validation index for Boolean data. *Cybernetics and Information Technologies,* 16**,** 232-244.

Guo, G., Chen, L., Ye, Y. & Jiang, Q. 2016. Cluster validation method for determining the number of clusters in categorical sequences. *IEEE transactions on neural networks and learning systems,* 28**,** 2936-2948.

Gurrutxaga, I., Albisua, I., Arbelaitz, O., Martín, J. I., Muguerza, J., Pérez, J. M. & Perona, I. 2010. SEP/COP: An efficient method to find the best partition in hierarchical clustering based on a new cluster validity index. *Pattern Recognition,* 43**,** 3364-3373.

Halkidi, M., Batistakis, Y. & Vazirgiannis, M. 2001. On clustering validation techniques. *Journal of intelligent information systems,* 17**,** 107-145.

Han, J., Pei, J. & Tong, H. 2022. *Data mining: concepts and techniques*, Morgan kaufmann.

Hennig, C. 2015. What are the true clusters? *Pattern Recognition Letters,* 64**,** 53-62.

Jain, A. K. & Dubes, R. C. 1988. *Algorithms for clustering data*, Prentice-Hall, Inc.

Jain, A. K., Murty, M. N. & Flynn, P. J. 1999. Data clustering: a review. *ACM computing surveys (CSUR),* 31**,** 264-323.

Jauhiainen, S. & Kärkkäinen, T. A simple cluster validation index with maximal coverage. European symposium on artificial neural networks, computational intelligence and machine learning, 2017. ESANN.

Kim, M. & Ramakrishna, R. 2005. New indices for cluster validity assessment. *Pattern Recognition Letters,* 26**,** 2353-2363.

Lago-Fernández, L. F. & Corbacho, F. 2010. Normality-based validation for crisp clustering. *Pattern Recognition,* 43**,** 782-795.

Liu, Y., Li, Z., Xiong, H., Gao, X., Wu, J. & Wu, S. 2013. Understanding and enhancement of internal clustering validation measures. *IEEE transactions on cybernetics,* 43**,** 982-994.

Misuraca, M., Spano, M. & Balbi, S. 2019. BMS: An improved Dunn index for Document Clustering validation. *Communications in statistics-theory and methods,* 48**,** 5036-5049.

Ncir, C.-E. B., Hamza, A. & Bouaguel, W. 2021. Parallel and scalable Dunn Index for the validation of big data clusters. *Parallel Computing,* 102**,** 102751.

Rousseeuw, P. J. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics,* 20**,** 53-65.

Saitta, S., Raphael, B. & Smith, I. F. A bounded index for cluster validity. Machine Learning and Data Mining in Pattern Recognition: 5th International Conference, MLDM 2007, Leipzig, Germany, July 18-20, 2007. Proceedings 5, 2007. Springer, 174-187.

Sharma, S. 1995. *Applied multivariate techniques*, John Wiley & Sons, Inc.

Thorndike, R. L. 1953. Who belongs in the family? *Psychometrika,* 18**,** 267-276.

Wang, X. & Xu, Y. An improved index for clustering validation based on Silhouette index and Calinski-Harabasz index. IOP Conference Series: Materials Science and Engineering, 2019. IOP Publishing, 052024.

Xie, X. L. & Beni, G. 1991. A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis & Machine Intelligence,* 13**,** 841-847.

Žalik, K. R. & Žalik, B. 2011. Validity index for clusters of different sizes and densities. *Pattern Recognition Letters,* 32**,** 221-234.

Zhao, Y. & Karypis, G. 2004. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine learning,* 55**,** 311-331.