Understanding and Enhancement of Internal Clustering Validation Measures

Yanchi Liu, Zhongmou Li, Hui Xiong, *Senior Member, IEEE*, Xuedong Gao, Junjie Wu, *Member, IEEE*, and Sen Wu

***Abstract*—Clustering validation has long been recognized as one of the vital issues essential to the success of clustering applications. In general, clustering validation can be categorized into two classes, external clustering validation and internal clustering validation. In this paper, we focus on internal clustering validation and present a study of 11 widely used internal clustering validation measures for crisp clustering. The results of this study indicate that these existing measures have certain limitations in different application scenarios. As an alternative choice, we propose a new internal clustering validation measure, named clustering validation index based on nearest neighbors (CVNN), which is based on the notion of nearest neighbors. This measure can dynamically select multiple objects as representatives for different clusters in different situations. Experimental results show that CVNN outperforms the existing measures on both synthetic data and real-world data in different application scenarios.**

***Index Terms*—Clustering validation index based on nearest neighbors (CVNN), internal clustering validation measure, *k*-nearest neighbor (kNN).**

I. Introduction

CLUSTERING, one of the most important unsupervised learning problems, is the task of dividing a set of objects into clusters such that objects within the same cluster are similar while objects in different clusters are distinct. Clustering is widely used in many fields, such as text mining, image analysis, and bioinformatics [2]–[4]. As an unsupervised learning task, it is necessary to find a way to validate the goodness of partitions after clustering. Otherwise, it would be difficult to make use of different clustering results.

Manuscript received October 12, 2010; revised January 16, 2012 and May 8, 2012; accepted September 6, 2012. Date of publication October 26, 2012; date of current version May 10, 2013. This work was supported in part by the National Science Foundation via Grants CCF-1018151 and IIP-1069258, by the National Natural Science Foundation of China under Grants 70890082, 71028002, 71271027, 70901002, 71171007, 71031001, and 70890080, by Fundamental Research Funds for the Central Universities of China under Grant FRF-TP-10-006B, by the Foundation for the Author of National Excellent Doctoral Dissertation of China under Grant 201189, and by the Program for New Century Excellent Talents in University. This paper was recommended by Associate Editor F. Karray.

Y. Liu is with the Department of Information Systems, New Jersey Institute of Technology, Newark, NJ 07102 USA, and also with the Department of Management Science and Information Systems, Rutgers Business School, Rutgers University, Newark, NJ 07102 USA (e-mail: yl473@njit.edu).

Z. Li and H. Xiong are with the Department of Management Science and Information Systems, Rutgers Business School, Rutgers University, Newark, NJ 07102 USA (e-mail: mosesli@pegasus.rutgers.edu; hxiong@rutgers.edu).

X. Gao and S. Wu are with the Dongling School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China (e-mail: gaoxuedong@manage.ustb.edu.cn; wusen@manage.ustb.edu.cn).

J. Wu is with the Department of Information System and the Beijing Key Laboratory of Emergency Support Simulation Technologies for City Operations, School of Economics and Management, Beihang University, Beijing 100191, China (e-mail: wujj@buaa.edu.cn).

Digital Object Identifier 10.1109/TSMCB.2012.2220543

Clustering validation, which evaluates the goodness of clustering results [5], has long been recognized as one of the vital issues essential to the success of clustering applications [6]. External clustering validation and internal clustering validation are the two main categories of clustering validation. The main difference is whether external information is used for clustering validation. An example of external validation measure is entropy, which evaluates the "purity" of clusters based on the given class labels [7].

Unlike external validation measures, which use external information not present in the data, internal measures evaluate the goodness of a clustering structure without respect to external information [8]–[11]. Since external validation measures know the "true" cluster number in advance, they are mainly used for choosing an optimal clustering algorithm on a specific data set. On the other hand, internal validation measures can be used to choose the best clustering algorithm as well as the optimal cluster number without any additional information. In practice, external information such as class labels is often not available in many application scenarios. Therefore, internal validation measures are the only option for cluster validation when there is no external information available.

In the literature, a number of internal clustering validation measures for crisp clustering have been proposed, such as the Calinski–Harabasz index (CH), the Davies–Bouldin index (DB), and standard deviation index (SD). However, current existing measures can be affected by various data characteristics [12], [13]. For example, noise in data can have a significant impact on the performance of an internal validation measure, if minimum or maximum pairwise distances are used in the measure. More fundamentally, existing measures are likely to perform well only in sphere-shaped clusters. The performance of existing measures in different situations remains unknown. Therefore, we present a study of 11 widely used internal validation measures, as shown in Table I. We study the impact of monotonicity of the first three measures and study others in a very detailed way. We investigate their validation properties in different aspects, such as data with noise, different density, and arbitrary shapes. The study results show that these measures have certain limitations in different scenarios.

As an alternative choice, we propose a new internal clustering validation measure, named clustering validation index based on nearest neighbors (CVNN), which is based on the notion of nearest neighbors. This measure consists of two components which measures intercluster separation and intracluster compactness, respectively. We refer to the idea of k-nearest neighbor (*kNN*) consistency and propose an index by using dynamic multiple objects as representatives for different

TABLE I
INTERNAL CLUSTERING VALIDATION MEASURES

	Measure	Notation	Definition	Optimal value
1	Root-mean-square std dev	$RMSSTD$	$\{\sum_i \sum_{x\in C_i} \parallel x-c_i \parallel^2/[P\sum_i(n_i-1)]\}^{\frac{1}{2}}$	Elbow
2	R-squared	RS	$(\sum_{x\in D} \parallel x-c \parallel^2 - \sum_i \sum_{x\in C_i} \parallel x-c_i \parallel^2)/\sum_{x\in D} \parallel x-c \parallel^2$	Elbow
3	Modified Hubert Γ statistic	Γ	$\frac{2}{n(n-1)} \sum_{x\in D}\sum_{y\in D} d(x,y)d_{x\in C_i, y\in C_j}(c_i,c_j)$	Elbow
4	Calinski-Harabasz index	CH	$\frac{\sum_i n_i d^2(c_i,c)/(NC-1)}{\sum_i \sum_{x\in C_i} d^2(x,c_i)/(n-NC)}$	Max
5	I index	I	$(\frac{1}{NC}\cdot \frac{\sum_{x\in D} d(x,c)}{\sum_i \sum_{x\in C_i} d(x,c_i)} \cdot \max_{i,j} d(c_i,c_j))^p$	Max
6	Dunn's indices	D	$\min_i\{\min_j(\frac{\min_{x\in C_i, y\in C_j} d(x,y)}{\max_k\{\max_{x,y\in C_k} d(x,y)\}})\}$	Max
7	Silhouette index	S	$\frac{1}{NC}\sum_i\{\frac{1}{n_i}\sum_{x\in C_i}\frac{b(x)-a(x)}{\max[b(x),a(x)]}\}$ $a(x)=\frac{1}{n_i-1}\sum_{y\in C_i, y\neq x} d(x,y), b(x)=\min_{j,j\neq i}[\frac{1}{n_j}\sum_{y\in C_j} d(x,y)]$	Max
8	Davies-Bouldin index	DB	$\frac{1}{NC}\sum_i \max_{j,j\neq i}\{[\frac{1}{n_i}\sum_{x\in C_i} d(x,c_i)+\frac{1}{n_j}\sum_{x\in C_j} d(x,c_j)]/d(c_i,c_j)\}$	Min
9	Xie-Beni index	XB	$[\sum_i \sum_{x\in C_i} d^2(x,c_i)]/[n\cdot min_{i,j\neq i} d^2(c_i,c_j)]$	Min
10	SD validity index	SD	$Dis(NC_{max})Scat(NC)+Dis(NC)$ $Scat(NC)=\frac{1}{NC}\sum_i \parallel \sigma(C_i) \parallel / \parallel \sigma(D) \parallel, Dis(NC)=\frac{max_{i,j} d(c_i,c_j)}{min_{i,j} d(c_i,c_j)}\sum_i(\sum_j d(c_i,c_j))^{-1}$	Min
11	S_Dbw validity index	S_Dbw	$Scat(NC)+Dens\_bw(NC)$ $Dens\_bw(NC)=\frac{1}{NC(NC-1)}\sum_i[\sum_{j,j\neq i}\frac{\sum_{x\in C_i\cup C_j} f(x,u_{ij})}{max\{\sum_{x\in C_i} f(x,c_i),\sum_{x\in C_j} f(x,c_j)\}}]$	Min

D: data set; n: number of objects in D; c: center of D; P: attributes number of D; NC: number of clusters; C_i: the i–th cluster; n_i: number of objects in C_i; c_i: center of C_i; $\sigma(C_i)$: variance vector of C_i; $d(x,y)$: distance between x and y; $\parallel X_i \parallel = (X_i^T \cdot X_i)^{\frac{1}{2}}$

clusters in different situations when measuring the intercluster separation. The measurement of the intracluster compactness is similar to the existing measures.

Finally, we provide comparative experiments to evaluate the validation properties and performances of CVNN. Experimental results show that CVNN outperforms the existing measures on both synthetic data and real-world data in different application scenarios. Therefore, CVNN can be used as a complementary measure to the existing measures, in particular, when data have arbitrary shapes.

II. INTERNAL CLUSTERING VALIDATION MEASURES

In this section, we introduce some basic concepts of internal validation measures, as well as a suite of 11 widely used internal validation indices.

As the goal of clustering is to make objects within the same cluster similar and objects in different clusters distinct, internal validation measures are often based on the following two criteria [8], [14], [15].

1) **Compactness**. It measures how closely related the objects in a cluster are. A group of measures evaluate cluster compactness based on variance. Lower variance indicates better compactness. In addition, there are numerous measures that estimate the cluster compactness based on distance, such as maximum or average pairwise distance and maximum or average center-based distance.
2) **Separation**. It measures how distinct or well separated a cluster is from other clusters. For example, the pairwise distances between cluster centers or the pairwise minimum distances between objects in different clusters are widely used as measures of separation. Also, measures based on density are used in some indices.

The general procedure to determine the best partition and optimal cluster number of a set of objects by using internal validation measures is as follows.

Step 1) Initialize a list of clustering algorithms which will be applied to the data set.
Step 2) For each clustering algorithm, use different combinations of parameters to get different clustering results.
Step 3) Compute the corresponding internal validation index of each partition obtained in Step 2).
Step 4) Choose the best partition and the optimal cluster number according to the criteria.

Table I shows a suite of 11 widely used internal validation measures. To the best of our knowledge, these measures represent a good coverage of the validation measures available in different fields, such as data mining, information retrieval, and machine learning. The "Definition" column gives the computation forms of the measures. On the other hand, most indices consider both of the evaluation criteria (compactness and separation) in the way of ratio or summation, such as the index DB, Xie–Beni index (XB), and S_Dbw; some only consider one aspect, such as root-mean-square standard deviation (RMSSTD), R-squared (RS), and Γ. Next, we briefly introduce these measures.

The RMSSTD is the square root of the pooled sample variance of all the attributes [16]. It measures the homogeneity of the formed clusters. RS is the ratio of sum of squares between clusters to the total sum of squares of the whole data set. It measures the degree of difference between clusters [16], [17]. The modified Hubert Γ statistic (Γ) [18] evaluates the difference between clusters by counting the disagreements of pairs of data objects in two partitions.

The index CH [19] evaluates the cluster validity based on the average between- and within-cluster sum of squares. Index I (I) [5] measures the separation based on the maximum distance between cluster centers and measures compactness based on the sum of distances between objects and their cluster center. The index by Dunn (D) [20] uses the minimum pairwise distance between objects in different clusters as the intercluster separation and the maximum diameter among all clusters as the intracluster compactness. These three indices take a form of $\text{Index} = (a \cdot \text{Separation})/(b \cdot \text{Compactness})$, where a and b are weights. The optimal cluster number is determined by maximizing the value of these indices.

The Silhouette index (S) [21] validates the clustering performance based on the pairwise difference of between- and within-cluster distances. In addition, the optimal cluster number is determined by maximizing the value of this index.

The DB [22] is calculated as follows. For each cluster C, the similarities between C and all other clusters are computed, and the highest value is assigned to C as its cluster similarity. Then, the DB index can be obtained by averaging all the cluster similarities. The smaller the index is, the better the clustering result is. By minimizing this index, clusters are the most distinct from each other and, therefore, achieves the best partition. The index XB [23] defines the intercluster separation as the minimum square distance between cluster centers and the intracluster compactness as the mean square distance between each data object and its cluster center. The optimal cluster number is reached when the minimum of XB is found. Kim and Ramakrishna [24] proposed indices DB^{**} and XB^{**} in year 2005 as the improvements of DB and XB. In this paper, we will use these two improved measures.

The idea of index SD [25] is based on the concepts of the average scattering and the total separation of clusters. The first term evaluates compactness based on variances of cluster objects, and the second term evaluates separation difference based on distances between cluster centers. The index SD is the summation of these two terms, and the optimal number of clusters can be obtained by minimizing the value of SD.

The index (S_Dbw) [26] takes density into account to measure the intercluster separation. The basic idea is that, for each pair of cluster centers, at least one of their densities should be larger than the density of their midpoint. The intracluster compactness is the same as it is in SD. Similarly, the index is the summation of these two terms, and the minimum value of S_Dbw indicates the optimal cluster number.

There are some other internal validation measures in the literature [27]–[30]. However, some have poor performance, while some are designed for data sets with specific structures. Take composed density between- and within-cluster index (CDbw) and symmetry distance-based index $(\mathrm{Sym}-\mathrm{index})$ for examples. It is hard for (CDbw) to find the representatives for each cluster, which makes the result of (CDbw) unstable. On the other hand, $\mathrm{Sym}-\mathrm{index}$ can only handle data sets which are internally symmetrical. As a result, we will focus on the aforementioned 11 internal validation measures in the rest of this paper, and we will use the acronyms for these measures.

III. Understanding of Internal Clustering Validation Measures

In this section, we present a study of the 11 internal validation measures mentioned in Section II and investigate the validation properties of different internal validation measures in different aspects, which can be helpful for the index selection. If not mentioned, we use K-means [31] (implemented by CLUstering TOolkit) [32] as the clustering algorithm for experiment.

A. Impact of Monotonicity

The monotonicity of different internal validation indices can be evaluated by the following experiment. We apply the K-means algorithm on the data set *Wellseparated* and get the clustering results for different number of clusters. As shown in Fig. 1, *Wellseparated* is a synthetic data set composed of 1000 data objects, which are well separated to five clusters.

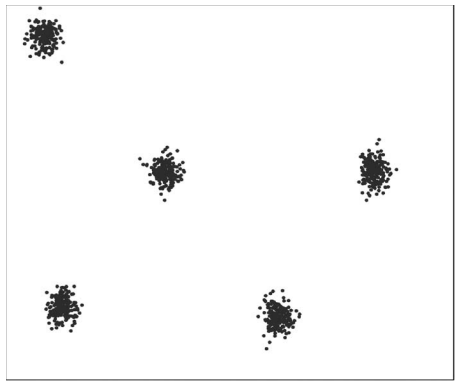


Fig. 1. Data set *Wellseparated*.

As the experiment results show in Table II, the first three indices monotonically increase or decrease as the cluster number NC increases. On the other hand, the rest eight indices reach their maximum or minimum value as NC equals to the true cluster number. There are certain reasons for the monotonicity of the first three indices.

$\mathrm{RMSSTD}=\sqrt{\mathrm{SSE}/P(n-\mathrm{NC})}$, and the sum of square error (SSE) decreases as NC increases. In practice, $\mathrm{NC}\ll n$; thus, $n-\mathrm{NC}$ can be viewed as a constant number. Therefore, RMSSTD decreases as NC increases. Moreover, we also have $\mathrm{RS}=(\mathrm{TSS}-\mathrm{SSE})/\mathrm{TSS}$ (TSS—total sum of squares), and $\mathrm{TSS}=\mathrm{SSE}+\mathrm{SSB}$ (SSB—between group sum of squares) which is a constant number for a certain data set. Thus, RS increases as NC increases.

From the definition of Γ, only data objects in different clusters will be counted in the equation. As a result, if the data set is divided into two equal clusters, each cluster will have $n/2$ objects, and $n^2/4$ pairs of distances will be counted actually. If the data set is divided into three equal clusters, each cluster will have $n/3$ objects, and $n^2/3$ pairs of distances will be counted. Therefore, with the increasing of the cluster number NC, more pairs of distances are counted, which makes Γ increase.

Looking further into these three indices, we can find out that they only take either separation or compactness into account (RS and Γ only consider separation, and RMSSTD only considers compactness). As the property of monotonicity, the curves of RMSSTD, RS, and Γ will be either upward or downward. It is claimed that the optimal cluster number is reached at the shift point of the curves, which is also known as "the elbow" [17]. However, since the judgement of the shift point is very subjective and hard to determine, we will not discuss these three indices in the further sections.

B. Impact of Noise

In order to evaluate the influence of noise on internal validation indices, we have the following experiment on the data set *Wellseparated.noise*. As shown in Fig. 2, *Wellseparated.noise* is a synthetic data set formulated by adding 5% noise to the data set *Wellseparated*. The cluster numbers selected by indices are

TABLE II
EXPERIMENT RESULTS OF THE IMPACT OF MONOTONICITY; TRUE NC = 5

	$RMSSTD$	RS	Γ	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}
2	28.496	0.627	2973	1683	3384	0.491	0.607	0.716	0.215	61.843	0.265
3	20.804	0.801	3678	2016	5759	0.549	0.707	0.683	0.124	0.153	0.374
4	14.829	0.899	4007	2968	11230	0.580	0.825	0.522	0.075	0.059	0.495
5	**3.201**	**0.994**	**4342**	**52863**	**106163**	**2.234**	**0.913**	**0.122**	**0.045**	**0.004**	**0.254**
6	3.081	0.995	4343	45641	82239	0.025	0.718	0.521	0.504	0.066	35.099
7	2.957	0.996	4344	41291	68894	0.017	0.579	0.803	0.486	0.098	35.099
8	2.834	0.996	4346	38580	58420	0.009	0.475	1.016	0.538	0.080	36.506
9	2.715	0.997	4347	36788	50259	0.010	0.391	1.168	0.553	0.113	38.008

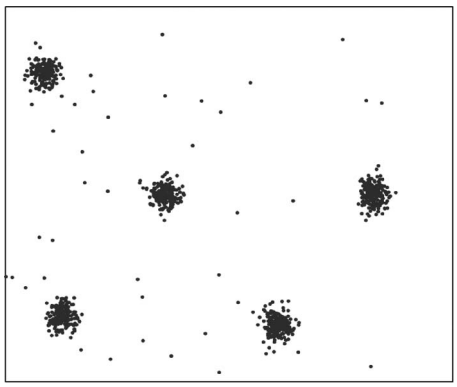


Fig. 2. Data set *Wellseparated.noise*.

TABLE III
EXPERIMENT RESULTS OF THE IMPACT OF NOISE; TRUE NC = 5

	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}
2	1626	3213	0.0493	0.590	0.739	0.069	20.368	0.264
3	1846	5073	0.0574	0.670	0.721	0.061	0.523	0.380
4	2554	9005	**0.0844**	0.783	0.560	0.050	0.087	0.444
5	10174	**51530**	0.0532	**0.870**	**0.183**	**0.045**	**0.025**	**0.251**
6	**14677**	48682	0.0774	0.802	0.508	0.046	0.044	0.445
7	12429	37568	0.0682	0.653	0.710	0.055	0.070	0.647
8	11593	29693	0.0692	0.626	0.863	0.109	0.052	2.404
9	11088	25191	0.0788	0.596	0.993	0.121	0.056	3.706

shown in Table III. The experiment results show that D and CH choose the wrong cluster number. From our point of view, there are certain reasons that D and CH are significantly affected by noise.

D uses the minimum pairwise distance between objects in different clusters $[(\min_{x\in C_i, y\in C_j} d(x,y)]$ as the intercluster separation and the maximum diameter among all clusters $(\max_k\{\max_{x,y\in C_k} d(x,y)\})$ as the intracluster compactness. Moreover, the optimal number of clusters can be obtained by maximizing the value of D. When noise is introduced, the intercluster separation can decrease sharply since it only uses the minimum pairwise distance, rather than the average pairwise distance, between objects in different clusters. Thus, the value of D may change dramatically, and the corresponding optimal cluster number will be influenced by the noise.

Since $\mathrm{CH} = (\mathrm{SSB}/\mathrm{SSE}) \cdot ((n - \mathrm{NC})/(\mathrm{NC} - 1))$, and $((n - \mathrm{NC})/(\mathrm{NC} - 1))$ is constant for the same NC, we can just focus on the $(\mathrm{SSB}/\mathrm{SSE})$ part. By introducing noise, SSE increases in a more significant way compared with SSB. Therefore, for the same NC, CH will decrease by the influence of noise, which makes the value of CH instable. Finally, the optimal cluster number will be affected by noise.

TABLE IV
EXPERIMENT RESULTS OF THE IMPACT OF DENSITY; TRUE NC = 3

	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}
2	1172	**120.1**	0.0493	0.587	0.658	0.705	0.603	0.408
3	**1197**	104.3	**0.0764**	**0.646**	**0.498**	**0.371**	**0.275**	**0.313**
4	1122	93.5	0.0048	0.463	1.001	0.672	0.401	3.188
5	932	78.6	0.0049	0.372	1.186	0.692	0.367	3.078
6	811	59.9	0.0049	0.312	1.457	0.952	0.312	6.192
7	734	56.1	0.0026	0.278	1.688	1.192	0.298	9.082
8	657	44.8	0.0026	0.244	1.654	1.103	0.291	8.897
9	591	45.5	0.0026	0.236	1.696	1.142	0.287	8.897

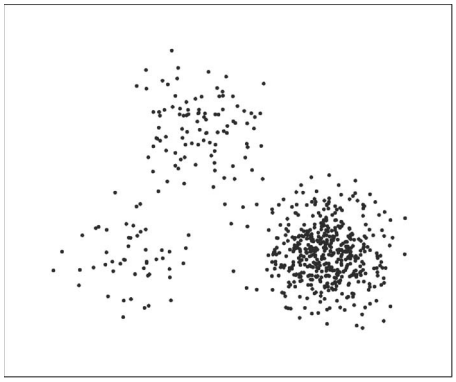


Fig. 3. Data set *Differentdensity*.

Moreover, the other indices rather than CH and D will also be influenced by noise in a less sensitive way. Comparing Table III with Table II, we can observe that the values of other indices more or less change. If we add 20% noise to the data set *Wellseparated*, the optimal cluster number suggested by I will also be incorrect. Thus, in order to minimize the adverse effect of noise, in practice, it is always good to remove noise before clustering.

C. Impact of Density

Data set with various densities is challenging for many clustering algorithms. Therefore, we are very interested in whether it also affects the performance of the internal validation measures. An experiment is done on a synthetic data set with different density, which is named *Differentdensity*. The results listed in Table IV show that only I suggests the wrong optimal cluster number. *Differentdensity* totally has 650 data objects, and the details of *Differentdensity* are shown in Fig. 3.

The reason why I does not give the right cluster number is not easy to tell. We can observe that I keeps decreasing as cluster number NC increases. One possible reason by our

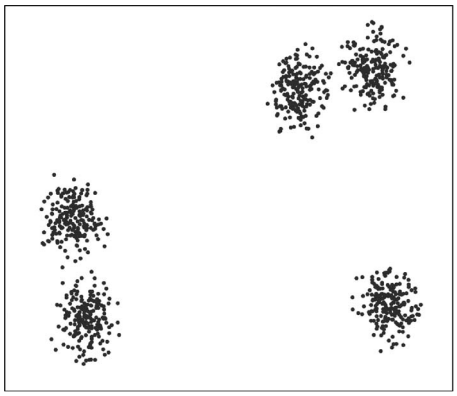


Fig. 4. Data set *Subcluster*.

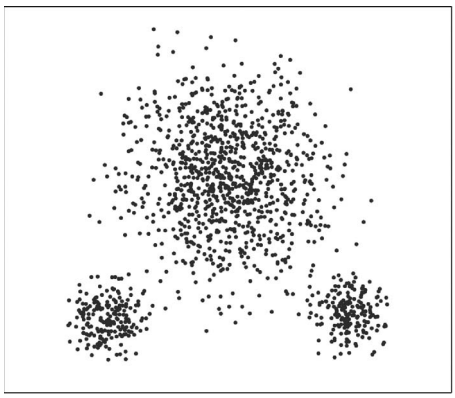


Fig. 5. Data set *Skewdistribution*.

TABLE V
EXPERIMENT RESULTS OF THE IMPACT OF SUBCLUSTERS; TRUE NC = 5

	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}
2	3474	2616	0.7410	0.736	0.445	0.156	0.207	0.378
3	7851	5008	**0.7864**	**0.803**	**0.353**	**0.096**	0.056	**0.264**
4	8670	5594	0.0818	0.737	0.540	0.164	0.039	1.420
5	**16630**	**9242**	0.0243	0.709	0.414	0.165	**0.026**	1.215
6	14310	7021	0.0243	0.587	0.723	0.522	0.063	12.538
7	12900	5745	0.0167	0.490	0.953	0.526	0.101	12.978
8	11948	4803	0.0167	0.402	1.159	0.535	0.105	14.037
9	11354	4248	0.0107	0.350	1.301	0.545	0.108	14.858

guessing is the uniform effect of K-means algorithm, which tends to divide objects into relatively equal sizes [33]. I measures compactness based on the sum of distances between objects and their cluster center. When NC is small, objects with high density are likely in the same cluster, which makes the sum of distances almost remain the same. Since most of the objects are in one cluster, the total sum will not change too much. Therefore, as NC increases, I will decrease since NC is in the denominator.

D. Impact of Subclusters

Subclusters are clusters that are closed to each other. Fig. 4 shows a synthetic data set *Subcluster* which contains five clusters, and four of them are subclusters since they can form two pairs of clusters, respectively. The total number of data objects in *Subcluster* is 1000.

The experiment results presented in Table V evaluate whether the internal validation measures can handle data set with subclusters. For this data set, D, S, DB^{**}, SD, and XB^{**} get the wrong optimal cluster numbers, while I, CH, and S_Dbw suggest the correct ones. Intercluster separation is supposed to have a sharp decrease when cluster number changes from $\mathrm{NC}_{\mathrm{optimal}}$ to $\mathrm{NC}_{\mathrm{optimal}}+1$ [24]. However, for D, S, DB^{**}, SD, and XB^{**}, sharper decreases can be observed at $\mathrm{NC}<\mathrm{NC}_{\mathrm{optimal}}$. The reasons are as follows.

S uses the average minimum distance between clusters as the intercluster separation. For data set with subclusters, the intercluster separation will achieve its maximum value when subclusters close to each other are considered as one big cluster. Therefore, the wrong optimal cluster number will be chosen due to subclusters. XB^{**} uses the minimum pairwise distance between cluster centers as the evaluation of separation. For data set with subclusters, the measure of separation will achieve its maximum value when subclusters closed to each other are considered as a big cluster. As a result, the correct cluster number will not be found by using XB^{**}. The reasons for D, SD, and DB^{**} are very similar to the reason of XB^{**}; we will not elaborate them here due to the limit of space.

E. Impact of Skewed Distributions

It is common that clusters in a data set have unequal sizes. Fig. 5 shows a synthetic data set *Skewdistribution* with skewed distributions, which contains 1500 data objects. It consists of one large cluster and two small ones. Since K-means has the uniform effect which tends to divide objects into relatively equal sizes, it does not have a good performance when dealing with skewed distributed data sets [33]. In order to demonstrate this statement, we employ four widely used algorithms from four different categories: K-means (prototype based), density-based spatial clustering of applications with noise (DBSCAN) (density based) [34], Agglo based on average link (hierarchical) [6], and Chameleon (graph based) [35]. We apply each of them on *Skewdistribution* and divide the data set into three clusters, since three is the true cluster number. As shown in Fig. 6, K-means performs the worst, while Chameleon is the best.

An experiment is done on the data set *Skewdistribution* to evaluate the performance of different indices on data set with skewed distributions. We use Chameleon as the clustering algorithm. The experiment results listed in Table VI show that only CH cannot give the right optimal cluster number, since $\mathrm{CH}=(\mathrm{TSS}/\mathrm{SSE}-1)\cdot((n-\mathrm{NC})/(\mathrm{NC}-1))$ and TSS is a constant number of a certain data set. Thus, CH is essentially based on SSE, which shares the same basis with K-means algorithm. As mentioned earlier, K-means cannot handle skewed distributed data sets. Therefore, the similar conclusion can be applied to CH.

F. Impact of Arbitrary Shapes

Data set with arbitrary shapes is always hard to handle. Fig. 7 shows a synthetic data set *T4.8k.modified* which consists of six irregular shapes of clusters. It is generated by removing 10% noise from the original data set *T4.8k* which contains 8000 objects [36]. Similarly, as in the last section, we employ the

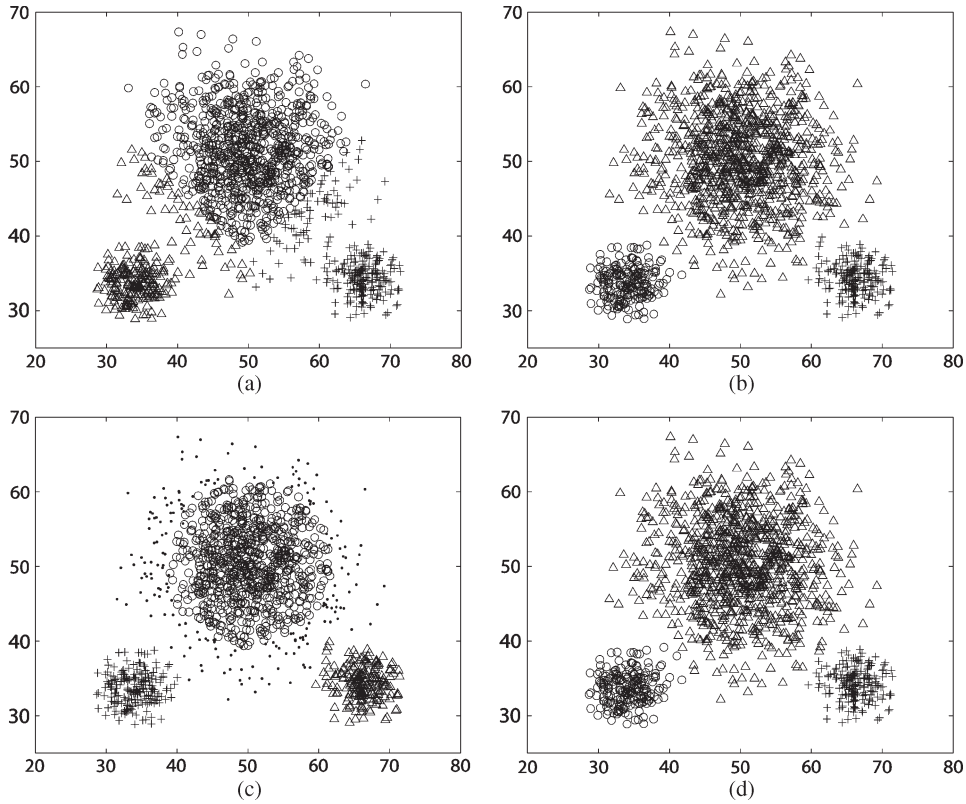


Fig. 6. Clustering results on the data set *Skewdistribution* by different algorithms where NC = 3. (a) Clustering by K-means. (b) Clustering by Agglo. (c) Clustering by DBSCAN. (d) Clustering by Chameleon.

TABLE VI
EXPERIMENT RESULTS OF THE IMPACT OF SKEWED DISTRIBUTIONS; TRUE NC = 3

	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}
2	788	232.3	0.0286	0.621	0.571	0.327	0.651	0.369
3	1590	**417.9**	**0.0342**	**0.691**	**0.466**	**0.187**	**0.309**	**0.264**
4	1714	334.5	0.0055	0.538	0.844	0.294	0.379	1.102
5	**1905**	282.9	0.0069	0.486	0.807	0.274	0.445	0.865
6	1886	226.7	0.0075	0.457	0.851	0.308	0.547	1.305
7	1680	187.1	0.0071	0.371	1.181	0.478	0.378	3.249
8	1745	172.9	0.0075	0.370	1.212	0.474	0.409	3.463
9	1317	125.5	0.0061	0.301	1.875	0.681	0.398	7.716

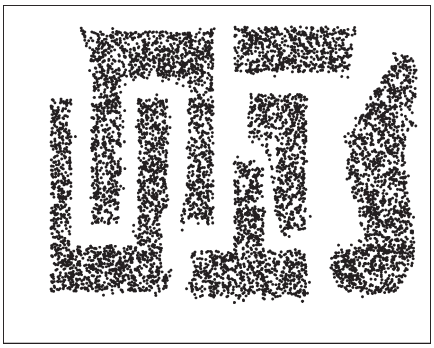


Fig. 7. Data set *T4.8k.modified*.

same four algorithms and run each of them on *T4.8k.modified* to divide the data set into six clusters, since six is the true cluster number. As shown in Fig. 8, Chameleon performs the best among these four clustering algorithms.

We finish an experiment on the data set *T4.8k.modified* to evaluate whether the eight internal validation indices can handle data set with arbitrary shapes. We use Chameleon as the clustering algorithm. Experiment results listed in Table VII show that none of the existing measures can deal with data set with arbitrary structures.

D uses the minimum pairwise distance between objects in different clusters to measure the intercluster separation. When dealing with arbitrary-shaped data sets, this can be misleading. For example, consider cluster A and cluster B' shown in Fig. 9(a). The minimum pairwise distance between these two clusters is almost zero while they are still separable.

For CH, I, DB^{**}, SD, S_Dbw, and XB^{**}, these six indices use the cluster center of each cluster as the representative for that cluster when evaluating the intercluster separation. In addition, S uses the average minimum pairwise distance between objects in each cluster as the separation measurement, which can be viewed as equivalent as the minimum pairwise distance between cluster centers in a sense. Since it is meaningful to use the center to represent for the entire cluster only for the sphere-shaped cluster, it implies that these indices can only work in the hypersphere condition. Fig. 9(a) shows an illustration for this argument. In this figure, both clusters A and B have an arcuate structure, and the cluster centers are not even in the clusters. If we move cluster B from the real-line place to the dash-line place B', A and B are getting closer, while the distance between their centers becomes larger. In this case, it is meaningless and incorrect to make cluster center representative for the entire cluster.

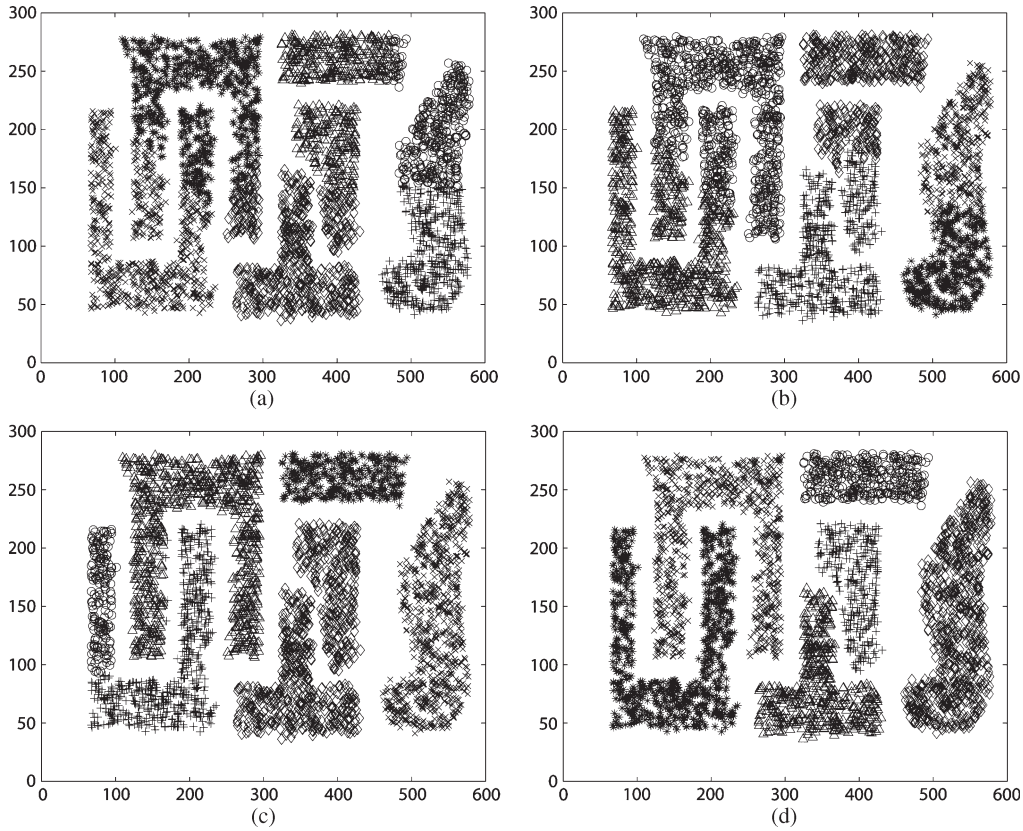


Fig. 8. Clustering results on the data set *T4.8k.modified* by different algorithms where NC = 6. (a) Clustering by K-means. (b) Clustering by Agglo. (c) Clustering by DBSCAN. (d) Clustering by Chameleon.

TABLE VII
EXPERIMENT RESULTS OF THE IMPACT OF ARBITRARY SHAPES; TRUE NC = 6

	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}
2	301	1808	0.0110	0.231	2.927	0.0442	1.790	7.824
3	5484	32080	0.0117	0.401	0.984	0.0219	0.579	1.271
4	**8213**	**34532**	**0.0183**	**0.438**	**0.769**	**0.0197**	0.680	**1.143**
5	6838	24902	0.0142	0.384	0.828	0.0299	0.509	3.032
6	7560	24721	0.0074	0.333	1.038	0.0286	∞	2.685
7	7151	20753	0.0080	0.343	0.984	0.0290	0.426	2.674
8	6445	16922	0.0072	0.367	0.896	0.0293	0.416	2.892
9	6636	22365	0.0067	0.376	0.865	0.0312	**0.281**	2.755

Furthermore, sometimes, it may be also inappropriate to use cluster centers to compute the intercluster separation even for the sphere-shaped clusters. Here is an example. There are two small clusters (A and B) and two big clusters (C and D) in Fig. 9(b). If we use the distance between cluster centers as the intercluster separation, then we have $\mathrm{dis}_{AB} = \mathrm{dis}_{CD}$. However, it is clear that clusters A and B are better separated than clusters C and D. Therefore, we need to find a new internal validation measure for this scenario, and we will discuss it in the next section.

IV. ENHANCEMENT OF INTERNAL CLUSTERING VALIDATION MEASURES

In this section, we propose a new internal validation measure based on the notion of nearest neighbors, as a complementary to the existing measures.

A. Intercluster Separation Based on Nearest Neighbors

An internal validation measure is generally based on intercluster separation and intracluster compactness. In the literature, some researchers believe that intercluster separation should play a more important role than intracluster compactness in clustering validation. A large number of research works emphasize the development of intercluster separation measures [27], [37], [38]. In general, existing validation measures of intercluster separation can be categorized into six classes [39].

Let C_i and C_j be two clusters in a data set, c_i and c_j be the cluster centers of C_i and C_j, and n_i and n_j be the numbers of objects in C_i and C_j, respectively. The following are the six classes.

I) $\mathrm{Sep}(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y)$.
II) $\mathrm{Sep}(C_i, C_j) = \max_{x \in C_i, y \in C_j} d(x, y)$.
III) $\mathrm{Sep}(C_i, C_j) = (n_i \cdot n_j)^{-1} \Sigma_{x \in C_i, y \in C_j} d(x, y)$.
IV) $\mathrm{Sep}(C_i, C_j) = d(c_i, C_j)$.
V) $\mathrm{Sep}(C_i, C_j) = \max\{\delta(C_i, C_j), \delta(C_i, C_j)\}$, where $\delta(C_i, C_j) = \max_{x \in C_i}\{\min_{y \in C_j} d(x, y)\}$ and $\delta(C_j, C_i) = \max_{y \in C_j}\{\min_{x \in C_i} d(x, y)\}$.
VI) $\mathrm{Sep}(C_i, C_j) = \mathrm{Dens}(c_{ij}) / \max\{\mathrm{Dens}(c_i), \mathrm{Dens}(c_j)\}$, where c_{ij} is the midpoint of c_i and c_j and $\mathrm{Dens}(c)$ is the density of c, usually computed by counting the number of objects within a certain distance from c.

Looking deeper into these six categories, we find that one single object is used to represent the entire cluster when calculating the intercluster separation in categories I), II), IV), V),

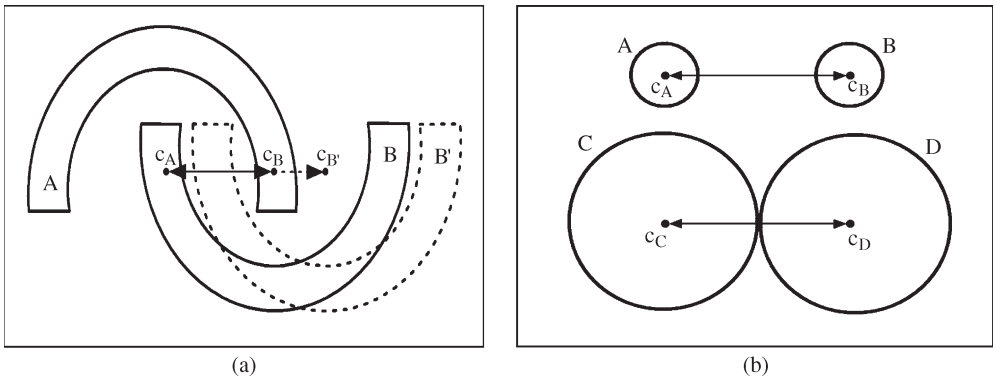


Fig. 9. Intercluster separation between clusters. (a) Arcuate shape. (b) Sphere shape.

and VI). However, using single representative for the entire cluster is questionable since one single object cannot keep the geometrical information of the whole cluster. The measurement in category III) is based on the average pairwise distance between objects in different clusters, which only considers the positions of the objects in clusters but not the object distributions which form the geometrical information [39]. Thus, there are certain limitations of the existing intercluster separation measures.

The criterion widely accepted to measure the intercluster separation present is to use cluster center as the representative for the entire cluster. However, as discussed earlier, as well as in Section III, this idea has its limitation since it loses the geometrical information and can only work on data sets with sphere-shaped clusters and is sometimes doubtable even in the spherical situation. Therefore, we propose a new intercluster separation measure as an alternative choice.

The basic idea of our measure is straightforward. We evaluate the intercluster separation only based on objects that carry the geometrical information of each cluster. Since one single object cannot reveal the geometrical information of the entire cluster, we use multiple objects as representatives; since, for different clusters, the object distributions and the relative positions to other clusters are different, we use different objects for the same cluster to present the geometrical information in different situations. In sum, we use dynamic multiple objects as representatives for different clusters in different situations when measuring the intercluster separation.

Which objects shall we take into account when evaluating the intercluster separation? Before going any further, we will review the concept of nearest neighbor consistency first.

1) Cluster kNN Consistency: For any data object in a cluster, its kNNs should also be in the same cluster [40]. *kNN* was initially proposed as the foundation of the *kNN* classification algorithm 50 years ago [41]. Later, it was realized that, since the goal of clustering is to divide objects into clusters, such that objects are more similar to objects within the cluster than to objects in other clusters, similar objects often tend to be close to each other. Thus, it is highly likely that an object and its nearest neighbors are all in the same cluster. In this sense, *kNN* consistency can be extended from supervised learning tasks (classification) to unsupervised learning tasks (clustering).

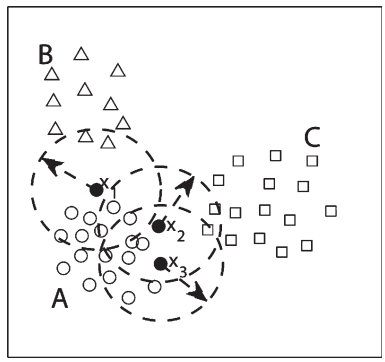
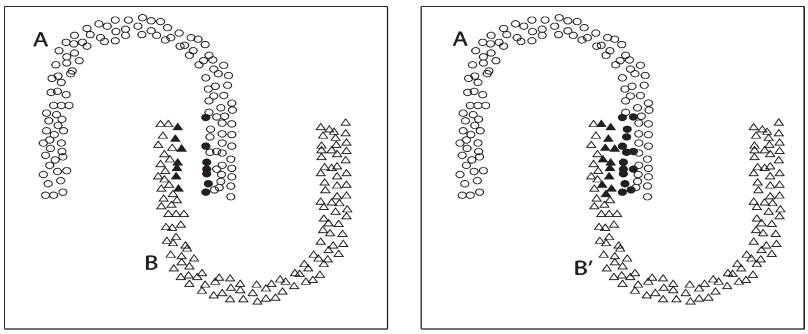


Fig. 10. Example of intercluster separation calculation.

Now, back to the question that which objects best represent for the entire clusters when evaluating the intercluster separation. Here are our thoughts: If an object is located in the center of a cluster and is surrounded by objects in the same cluster, it is well separated from other clusters and thus contributes little to the intercluster separation; if an object is located at the edge of a cluster and is surrounded mostly by objects in other clusters, it connects to other clusters tightly and thus contributes a lot to the intercluster separation. It shares the same idea of *kNN* consistency. Along this line, we propose a new measurement of the intercluster separation based on the notion of nearest neighbors, which is different from the existing measures.

2) Intercluster Separation (Sep)*:* $\mathrm{Sep}(\mathrm{NC}, k) = \max_{i=1,2,\ldots,\mathrm{NC}}((1/n_i)\sum_{j=1,2,\ldots,n_i}(q_j/k))$. NC is the cluster number, k is the number of nearest neighbors, n_i is the number of objects in the ith cluster C_i, O_j is the jth object in C_i, and q_j is the number of nearest neighbors of O_j which are not in cluster C_i. We define the intercluster separation mainly in four steps. First, for each object in each cluster, find out whether at least one of its *kNNs* is in other clusters. Second, for objects with positive answers, assign a weight to each of them (q_j/k). These are the objects that best represent for the entire clusters; for objects with negative answers, the weight is zero. Third, calculate the average weight of objects in the same cluster. Finally, take the maximum average weight among all clusters as the intercluster separation. Note that a lower value of Sep indicates a better intercluster separation.

Fig. 10 shows an example of the intercluster separation calculation process. There are three clusters in this figure, which are



(a) (b)

Fig. 11. Illustration of the dynamic effect of cluster representatives. (a) Before movement. (b) After movement.

represented by triangle, square, and circle, respectively. We set $k = 10$ here. For objects in cluster A, we find out that only objects $x_1, x_2,$ and x_3 have two, two, and one neighbor(s) in other clusters out of their ten-nearest neighbors separately. These three objects are the representatives for cluster A when calculating the intercluster separation. Since there are 20 objects in cluster A, $\mathrm{Sep}_A(3, 10) = (1/20) \cdot (2/10 + 2/10 + 1/10) = 0.025$. The same process can be applied to calculate Sep_B and Sep_C, and the maximum value is the intercluster separation index.

Fig. 11 shows the dynamic effect of how representatives for clusters evolve in different situations when measuring the intercluster separation. In this example, both clusters A and B have an arcuate structure, and solid objects are the representatives selected by our measure. Comparing Fig. 11(a) with Fig. 11(b), we can see that A and B' are closer than A and B, which indicates that the intercluster separation is getting worse. Meanwhile, the numbers of representatives for both clusters are growing as well as our intercluster separation measure Sep, which agree with the indication that clusters are getting worse separated. This example illustrates the dynamic effect of our intercluster separation measure well, since the representatives for the same two clusters in different situations are different.

B. Intracluster Compactness

Intracluster compactness is the other indispensable part of internal validation measures. Usually, the compactness shows a monotonically decreasing tendency when cluster number approaches to the number of objects in the data set [42]. Existing validation measures of intracluster compactness can be categorized into the following five classes.

Let C_i be one cluster in a data set, x and y be the two different objects in C_i, c_i is the cluster center of C_i, and n_i is the number of objects in C_i. The following are the five classes.

I) $\mathrm{Com}(C_i) = (1/n_i)\Sigma_{x \in C_i} d^2(x, c_i)$.
II) $\mathrm{Com}(C_i) = \max_{x,y \in C_i} d(x, y)$.
III) $\mathrm{Com}(C_i) = (2/n_i \cdot (n_i - 1))\Sigma_{x,y \in C_i} d(x, y)$.
IV) $\mathrm{Com}(C_i) = (1/n_i)\Sigma_{x \in C_i} d(x, c_i)$.
V) $\mathrm{Com}(C_i) = \|\sigma(C_i)\|$.

As discussed in Section III-F, in order for an internal validation measure to deal with arbitrary-shaped data set, we should avoid using the center point to represent for the entire cluster, since it only works in the hypersphere condition. As a result, we can eliminate the candidacies of categories I), IV), and V). In addition, the compactness should also not be determined by the distance between a single pair of objects in the cluster, which excludes category II) as well. Therefore, the only option left as our measurement of intracluster compactness is category III).

TABLE VIII
EXPERIMENT RESULTS OF CVNN IN DIFFERENT ASPECTS

	2	3	4	5	6	7	8	9
*Wellseparated - **5***	1.00	0.64	0.38	**0.12**	0.74	1.05	1.11	1.10
*Wellseparated.noise - **5***	1.01	0.69	0.44	**0.19**	0.51	0.92	1.15	0.98
*Differentdensity - **3***	1.03	**0.84**	0.92	1.22	1.32	1.28	1.26	1.59
*Subcluster - **5***	1.00	0.54	0.47	**0.43**	0.79	1.26	1.25	1.06
*Skewdistribution - **3***	1.04	**0.73**	0.85	0.87	0.96	1.10	1.03	1.41
*T4.8k.modified - **6***	1.03	0.75	0.65	0.62	**0.58**	0.89	1.39	1.36

TABLE IX
OVERALL PERFORMANCE OF DIFFERENT INDICES

	Mono.	Noise	Dens.	Subc.	Skew Dis.	Arbi. Shape
$RMSSTD$	×	–	–	–	–	–
RS	×	–	–	–	–	–
Γ	×	–	–	–	–	–
CH		×			×	×
I			×			×
D		×		×		×
S				×		×
DB^{**}				×		×
SD				×		×
S_Dbw						×
XB^{**}				×		×
$CVNN$						

1) Intracluster Compactness (Com)*:* $\mathrm{Com(NC)} = \sum_i [(2/n_i \cdot (n_i - 1))\Sigma_{x,y \in C_i} d(x, y)]$, where NC is the cluster number, n_i is the number of objects in the ith cluster C_i, and x and y are two different objects in C_i. This measure is mainly based on the average pairwise distance between objects in the same cluster. Note that a lower value of Com indicates a better intracluster compactness.

C. CVNN Index

Based on the intercluster separation and intracluster compactness, we have the definition of our internal CVNN.

1) CVNN Index: $\mathrm{CVNN(NC}, k) = \mathrm{Sep}_{\mathrm{norm}}(\mathrm{NC}, k) + \mathrm{Com}_{\mathrm{norm}}(\mathrm{NC})$, where $\mathrm{Sep}_{\mathrm{norm}}(\mathrm{NC}, k) = \mathrm{Sep}(\mathrm{NC}, k)/(\max_{\mathrm{NC}_{\min} \le \mathrm{NC} \le \mathrm{NC}_{\max}} \mathrm{Sep}(\mathrm{NC}, k))$ and $\mathrm{Com}_{\mathrm{norm}}(\mathrm{NC}) = \mathrm{Com}(\mathrm{NC})/(\max_{\mathrm{NC}_{\min} \le \mathrm{NC} \le \mathrm{NC}_{\max}} \mathrm{Com}(\mathrm{NC}))$. This index takes a form of the summation of the intercluster separation

TABLE X
EXPERIMENT RESULTS OF CVNN WITH DIFFERENT k's

	2	3	4	5	6	7	8	9
k=1								
*Wellseparated - **5***	1.00	0.64	0.38	**0.13**	0.83	0.83	0.89	1.10
*Wellseparated.noise - **5***	1.01	0.68	0.45	**0.21**	0.56	1.15	1.15	1.15
*Differentdensity - **3***	1.09	**0.94**	0.97	0.98	1.21	1.18	1.20	1.59
*Subcluster - **5***	1.00	0.54	**0.44**	0.53	0.53	1.26	1.00	1.15
*Skewdistribution - **3***	1.09	0.78	**0.75**	0.84	1.18	1.18	1.40	1.12
*T4.8k.modified - **6***	1.00	0.67	0.65	0.62	**0.59**	0.88	1.39	1.36
k=5								
*Wellseparated - **5***	1.00	0.64	0.38	**0.12**	0.74	1.05	1.11	1.10
*Wellseparated.noise - **5***	1.01	0.69	0.44	**0.19**	0.51	0.92	1.15	0.98
*Differentdensity - **3***	1.03	**0.82**	0.92	1.22	1.32	1.28	1.26	1.59
*Subcluster - **5***	1.00	0.54	0.47	**0.42**	0.79	1.26	1.25	1.06
*Skewdistribution - **3***	1.04	**0.73**	0.85	0.87	0.96	1.10	1.03	1.41
*T4.8k.modified - **6***	1.03	0.75	0.65	0.62	**0.58**	0.89	1.39	1.36
k=10								
*Wellseparated - **5***	1.00	0.64	0.38	**0.12**	0.83	1.08	1.08	1.10
*Wellseparated.noise - **5***	1.01	0.69	0.44	**0.19**	0.87	1.07	1.15	1.14
*Differentdensity - **3***	1.03	**0.84**	0.91	1.11	1.22	1.19	1.17	1.59
*Subcluster - **5***	1.00	0.54	0.47	**0.43**	0.96	1.14	1.19	1.24
*Skewdistribution - **3***	1.03	**0.73**	0.85	0.88	0.96	1.10	1.03	1.41
*T4.8k.modified - **6***	1.03	0.75	0.65	0.62	**0.58**	0.89	1.39	1.36
k=20								
*Wellseparated - **5***	1.00	0.64	0.38	**0.12**	0.83	1.09	1.11	1.10
*Wellseparated.noise - **5***	1.01	0.69	0.44	**0.20**	0.91	1.01	1.15	1.09
*Differentdensity - **3***	1.05	**0.92**	0.95	1.45	1.65	1.62	1.61	1.59
*Subcluster - **5***	1.00	0.54	**0.45**	**0.45**	0.92	1.25	1.18	1.24
*Skewdistribution - **3***	1.03	**0.73**	0.86	0.92	0.94	1.07	1.06	1.41
*T4.8k.modified - **6***	1.03	0.81	0.68	0.65	**0.59**	0.85	1.40	1.37
k=50								
*Wellseparated - **5***	1.00	0.64	0.38	**0.12**	1.04	1.05	1.11	1.10
*Wellseparated.noise - **5***	1.01	0.69	0.44	**0.19**	0.97	1.02	1.04	1.15
*Differentdensity - **3***	1.03	**0.89**	0.90	1.10	1.16	1.13	1.22	1.59
*Subcluster - **5***	1.00	0.54	**0.44**	0.46	0.97	1.26	1.25	1.24
*Skewdistribution - **3***	1.02	**0.72**	0.86	0.91	0.92	1.09	1.09	1.41
*T4.8k.modified - **6***	1.04	0.81	0.82	0.79	**0.74**	0.82	1.40	1.37
k=100								
*Wellseparated - **5***	1.00	0.64	0.38	**0.12**	0.95	1.01	0.95	1.10
*Wellseparated.noise - **5***	1.01	0.69	0.44	**0.19**	1.12	1.12	1.15	1.15
*Differentdensity - **3***	1.11	**1.29**	1.41	**1.29**	1.35	1.37	1.48	1.59
*Subcluster - **5***	1.00	0.54	**0.46**	0.47	1.07	1.12	1.11	1.24
*Skewdistribution - **3***	1.01	**0.72**	0.97	1.03	1.16	1.28	1.34	1.41
*T4.8k.modified - **6***	1.08	**0.74**	0.86	0.83	0.78	0.98	1.38	1.37

and the intracluster compactness. Note that we normalize them to the same range before adding them up, since they should have the same order of magnitude. A lower value of CVNN indicates a better clustering result.

The computational complexity of CVNN is determined by the complexities of both Sep and Com. For Sep, the main computational cost is the search of the *kNNs* for each object in the data set. This is a one-time effort, and the result of the *kNN* search can be stored for future use. The computational complexity of the *kNN* search has been well studied. The brute force method gives a complexity of $O(dN^2)$, where N is the total number of objects in the data set and d is the number of dimensions. Furthermore, some space-partition-based methods, such as k-d tree [43] and R-tree [44], have been developed to reduce the computational complexity down to $O(dN \log N)$. When computing Sep with the result of the *kNN* search, for each object O, we have to decide how many O *kNNs* share the same cluster with it. Considering the variation of NC, the total complexity of Sep should be $O(dN \log N) + O(N) \cdot k \cdot (\mathrm{NC}_{\max} - \mathrm{NC}_{\min}) = O(dN \log N)$. For Com, since we have to compute the average pairwise distance of objects within the same cluster, the computational complexity is $O(dN^2)$. Therefore, the complexity of CVNN is $O(dN^2)$, which makes it affordable for large-scale and high-dimensional data sets.

TABLE XI
EXPERIMENT RESULTS OF CVNN FOR DIFFERENT CLUSTERING ALGORITHMS

	2	3	4	5	6	7	8	9
*Skewdistribution - **3***								
K-means	1.19	0.93	0.86	1.39	1.52	1.37	1.45	1.42
Agglo	1.02	0.74	0.90	1.19	1.14	1.09	1.35	1.38
DBSCAN	1.01	**0.56**	1.54	1.39	–	–	–	–
Chameleon	1.04	**0.73**	0.85	0.87	0.96	1.10	1.03	1.41
*T4.8k.modified - **6***								
K-means	1.04	1.87	1.54	0.91	1.04	1.05	1.18	1.12
Agglo	1.08	0.94	1.21	1.11	1.10	1.32	1.33	1.44
DBSCAN	–	–	–	1.33	1.81	1.89	–	–
Chameleon	1.03	0.75	0.65	0.62	**0.58**	0.89	1.39	1.36

TABLE XII
DATA SET CHARACTERISTICS

Data Set	Class #	Instance #	Attribute #
Iris	3	150	4
Cancer	2	699	10
Wine	3	178	13
Letter	26	20000	16
Satimage	6	6435	36
Glass	6	214	9
Yeast	10	1484	8
Ecoli	8	336	7
Magic	2	19020	11
Vehicle	4	846	18

V. EXPERIMENTAL EVALUATIONS

In this section, we provide comparative experiments to evaluate the validation properties and performances of CVNN. We set the parameter $k = 10$ in CVNN for the experiments if not specified.

A. Validation Properties of CVNN

Here, we investigate the validation properties of CVNN in the same problem settings as in Section III. Table VIII shows the experimental results, and the true cluster number NC of each data set is also given in the table. In the experiments, we can observe that CVNN successfully suggests the correct cluster numbers in all six aspects, which none of the existing indices can achieve.

Table IX lists the properties of different internal validation measures in different aspects, which can be helpful for the index selection. "$-$" stands for property not tested, and "$\times$" denotes situation cannot be handled. As mentioned earlier, CVNN outperforms the existing measures in all six aspects.

B. Sensitivity Analysis of the Value of k

In this section, we study how the change in parameter k influences the validation result. Table X lists the values of CVNN with different k's that vary from 1 to 100, on the same six data sets as in Section III. From the table, we can

TABLE XIII
EXPERIMENT RESULTS ON REAL-WORLD DATA SETS

	CH	I	D	S	DB^{**}	SD	S_Dbw	XB^{**}	$CVNN$
*Iris - **3***	2	**3**	2	2	2	2	2	2	**3** - 10
*Cancer - **2***	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2** - 10
*Wine - **3***	4	5	6	2	2	2	7	2	**3** - 5
*Letter - **26***	28	22	22	28	28	25	28	28	**26** - 20
*Satimage - **6***	2	3	2	2	2	2	2	2	**6** - 20
*Glass - **6***	8	7	3	2	2	2	9	2	4 - 5
*Yeast - **10***	14	14	14	7	7	7	14	7	**10** - 10
*Ecoli - **8***	13	6	7	3	2	7	13	2	6 - 5
*Magic - **2***	**2**	4	**2**	**2**	**2**	6	8	**2**	**2** - 20
*Vehicle - **4***	9	7	6	2	2	2	7	2	**4** - 10
Accuracy	0.2	0.2	0.2	0.2	0.2	0.1	0.1	0.2	0.8

**Accuracy = number of data sets with correct results / total number of data sets.*
**The values in the last column stand for (suggested cluster number - k value).*

observe a slight change in the value of CVNN for different k's. When the value of k is small, e.g., $k = 1$, CVNN cannot suggest the correct cluster numbers for data sets *Subcluster* and *Skewdistribution*. With the increasing of k, the value of CVNN gradually decreases and reaches its steadily minimum value. Meanwhile, CVNN uncovers the optimal cluster numbers of all six data sets. However, when k keeps increasing after passing its optimal value, e.g., $k = 50$ or 100, the value of CVNN increases again, and it recommends the wrong cluster numbers for data sets *Subcluster* and *T4.8k.modified*. In summary, the value of CVNN has a U-shape curve as the increase of k values, and it suggests the correct cluster number when it achieves its minimum value. Therefore, we can obtain the minimum value of CVNN by searching a wide range of k values and thus identify the optimal cluster number.

C. CVNN for Best Partition

For the same data set, different clustering algorithms give different partitions. A good internal clustering validation measure should not only be able to suggest the correct cluster number but also be capable to find out the best partitions given by different clustering algorithms.

An experiment is done on data sets *Skewdistribution* and *T4.8k.modified* by applying the four algorithms mentioned in Section III: K-means, DBSCAN, Agglo based on average link, and Chameleon. Table XI lists the values of CVNN for different partitions by different algorithms. Note that DBSCAN does not require cluster number as an input parameter; on the contrary, it takes in two parameters, the value of the searching radius and the minimum number of points to form a cluster, and then outputs the clustering result as well as the number of clusters. Thus, DBSCAN may not be able to divide a set of objects into a specific number of clusters. For example, we cannot separate objects in *T4.8k.modified* data set into two, three, four, eight, or nine clusters by using DBSCAN.

The experiment results indicate that CVNN gives the best partitions for both data sets. For *T4.8k.modified*, CVNN suggests that the best partition is given by Chameleon with cluster number 6, which consists of the results in Fig. 8. For *Skewdistribution*, CVNN indicates that the best partition is given by DBSCAN with cluster number 3. As shown in Fig. 6(c), DBSCAN treats a certain portion of objects in the data set as noises; thus, the value of CVNN for DBSCAN tends to be smaller than that for other algorithms. Since the second best partition is Chameleon with cluster number 3, we can say that both DBSCAN and Chameleon indicate the best partitions in different senses, which meets the conclusion from Fig. 6.

D. Experiment Results on Real-World Data Sets

In this section, we compare the optimal cluster number suggested by CVNN with the cluster numbers obtained by existing indices on ten real-world data sets from the University of California, Irvine (UCI) machine learning repository [45]: *Iris, Cancer, Wine, Letter, Satimage, Glass, Yeast, Ecoli, Magic*, and *Vehicle*. Table XII lists the basic statistics of these data sets. We use K-means as the clustering algorithm, which has been proved as an effective algorithm in real-world applications. As discussed in Section V-B, we reach the minimum value of CVNN by assigning different values to k (5, 10, 20, 50, and 100 in this case) and obtain the optimal cluster number accordingly.

As the experiment results show in Table XIII, the accuracy rate of CVNN is far higher than other indices, where the accuracy is the ratio of the number of data sets with correct results to the total number of testing data sets. In addition, CVNN can handle all data sets which the other indices can deal with, while the other indices can handle none of the data sets which CVNN cannot deal with. Thus, it is clear that CVNN performs the best among all indices, which consists of the conclusions obtained from previous sections.

In order to perform a test of statistical significance whether the accuracy rate of CVNN is better than that of the existing measures, we assume that the accuracy rates of the existing internal validation measures follow a Students's t-distribution, since the sample size is small and their accuracies are similar on the ten UCI data sets. Thus, the estimation of the mean is $\hat{\mu} = \bar{x} = 0.175$, and the estimation of the standard deviation is $\hat{\sigma} = s = 0.046$. The result of a standard one-tailed hypothesis test indicates that CVNN performs significantly better than the existing internal clustering validation measures at the significance level of 0.001. From this test of significance, we can infer that CVNN is statistically significant and outperforms other existing measures.

Based on the experimental results discussed earlier, we can conclude that CVNN is an effective and efficient internal measure for cluster validation, particularly when data include clusters having arbitrary shapes. Therefore, CVNN can be

a valuable complementary measure to the existing suite of internal clustering validation measures.

VI. Concluding Remarks

In this paper, we have investigated the validation properties of a suite of 11 existing internal clustering validation measures in different aspects. As demonstrated by the experiment results, these measures are only applicable in certain situations. In particular, none of them performs well on data sets with arbitrary shapes. As a complementary measure to these existing measures, we proposed a new internal clustering validation measure, named CVNN, which exploits the notion of nearest neighbors and uses dynamic multiple objects as representatives for different clusters in different situations. Experimental results show that CVNN is capable to suggest the correct number of clusters as well as the best partition on various synthetic and real-world data sets, including the data set with arbitrary cluster shapes.

Acknowledgment

The authors would like to thank the anonymous referees from the IEEE International Conference on Data Mining and the IEEE Transactions on Systems, Man, and Cybernetics—Part B for their valuable and constructive comments on this paper. A preliminary version of this work has been accepted for publication as a six-page short paper in ICDM 2010 [1].

References

[1] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *Proc. IEEE Int. Conf. Data Mining*, 2010, pp. 911–916.

[2] N. Guan, D. Tao, Z. Luo, and B. Yuan, "NeNMF: An optimal gradient method for nonnegative matrix factorization," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2882–2898, Jun. 2012.

[3] T. Zhou, D. Tao, and X. Wu, "Manifold elastic net: A unified framework for sparse dimension reduction," *Data Mining Knowl. Discov.*, vol. 20, no. 3, pp. 340–371, May 2011.

[4] N. Guan, D. Tao, Z. Luo, and B. Yuan, "Online non-negative matrix factorization with robust stochastic approximation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1087–1099, Jul. 2012.

[5] U. Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 12, pp. 1650–1654, Dec. 2002.

[6] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Upper Saddle River, NJ: Prentice-Hall, 1988.

[7] J. Wu, H. Xiong, and J. Chen, "Adapting the right measures for k-means clustering," in *Proc. ACM SIGKDD*, 2009, pp. 877–886.

[8] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, 1st ed. Boston, MA: Addison-Wesley, 2005.

[9] M. Brun, C. Sima, J. Hua, J. Lowey, B. Carroll, E. Suh, and E. R. Dougherty, "Model-based evaluation of clustering validation measures," *Pattern Recogn.*, vol. 40, no. 3, pp. 807–824, Mar. 2007.

[10] M. J. Song and L. Zhang, "Comparison of cluster representations from partial second- to full fourth-order cross moments for data stream clustering," in *Proc. IEEE ICDM*, 2008, pp. 560–569.

[11] H. Kremer, P. Kranen, T. Jansen, T. Seidl, A. Bifet, G. Holmes, and B. Pfahringer, "An effective evaluation measure for clustering on evolving data streams," in *Proc. ACM SIGKDD*, 2011, pp. 868–876.

[12] W. Sheng, S. Swift, L. Zhang, and X. Liu, "A weighted sum validity function for clustering with a hybrid niching genetic algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1156–1167, Dec. 2005.

[13] G. W. Milligan, "A Monte Carlo study of thirty internal criterion measures for cluster analysis," *Psychometrika*, vol. 46, no. 2, pp. 187–199, Jun. 1981.

[14] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *Proc. CIKM*, 2002, pp. 515–524.

[15] J. M. Kraus, C. Müssel, G. Palm, and H. A. Kestler, "Multi-objective selection for collecting cluster alternatives," *Comput. Stat.*, vol. 26, no. 2, pp. 341–353, Jun. 2011.

[16] S. Sharma, *Applied Multivariate Techniques*. New York: Wiley, 1996.

[17] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *J. Intell. Inf. Syst.*, vol. 17, no. 2/3, pp. 107–145, Dec. 2001.

[18] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, Dec. 1985.

[19] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Stat.*, vol. 3, no. 1, pp. 1–27, 1974.

[20] J. Dunn, "Well separated clusters and optimal fuzzy partitions," *Cybern. Syst.*, vol. 4, no. 1, pp. 95–104, 1974.

[21] P. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, no. 1, pp. 53–65, Nov. 1987.

[22] D. Davies and D. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979.

[23] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 8, pp. 841–847, Aug. 1991.

[24] M. Kim and R. S. Ramakrishna, "New indices for cluster validity assessment," *Pattern Recogn. Lett.*, vol. 26, no. 15, pp. 2353–2363, Nov. 2005.

[25] M. Halkidi, M. Vazirgiannis, and Y. Batistakis, "Quality scheme assessment in the clustering process," in *Proc. PKDD*, 2000, pp. 265–276.

[26] M. Halkidi and M. Vazirgiannis, "Clustering validity assessment: Finding the optimal partitioning of a data set," in *Proc. IEEE Int. Conf. Data Mining*, 2001, pp. 187–194.

[27] S. Saha and S. Bandyopadhyay, "Application of a new symmetry-based cluster validity index for satellite image segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 166–170, Apr. 2002.

[28] M. Halkidi and M. Vazirgiannis, "Clustering validity assessment using multi-representatives," in *Proc. SETN*, 2002, pp. 237–248.

[29] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *J. R. Stat. Soc., Ser. B (Stat. Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.

[30] B. S. Y. Lam and H. Yan, "A new cluster validity index for data with merged clusters and different densities," in *Proc. IEEE ICSMC*, 2005, pp. 798–803.

[31] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. BSMSP*, 1967, pp. 281–297.

[32] G. Karypis, *Cluto—Software for Clustering High-Dimentional Datasets*. Minneapolis, MN: Karypis Lab, 2006, version 2.1.2.

[33] H. Xiong, J. Wu, and J. Chen, "K-means clustering versus validation measures: A data distribution perspective," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 2, pp. 318–331, Apr. 2009.

[34] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. KDD*, 1996, pp. 226–231.

[35] G. Karypis, E.-H. S. Han, and V. Kumar, "Chameleon: Hierarchical clustering using dynamic modeling," *Computer*, vol. 32, no. 8, pp. 68–75, Aug. 1999.

[36] G. Karypis, Karypis Lab. [Online]. Available: http://glaros.dtc.umn.edu/gkhome/

[37] J. C. Bezdek and N. R. Pal, "Some new indexes of cluster validity," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 28, no. 3, pp. 301–315, Jun. 1998.

[38] T. Lange, V. Roth, M. L. Braun, and J. M. Buhmann, "Stability-based validation of clustering solutions," *Neural Comput.*, vol. 16, no. 6, pp. 1299–1323, Jun. 2004.

[39] S. Yue, J.-S. Wang, T. Wu, and H. Wang, "A new separation measure for improving the effectiveness of validity indices," *Inf. Sci.*, vol. 180, no. 5, pp. 748–764, Mar. 2010.

[40] C. Ding and X. He, "K-nearest-neighbor consistency in data clustering: Incorporating local information into global optimization," in *Proc. SAC*, New York, 2004, pp. 584–589.

[41] E. Fix and J. J. L. Hodges, "Discriminatory analysis, nonparametric discrimination: Consistency properties," USAF Sch. Aviation Med., Randolph Field, TX, Tech. Rep. 4, 1951.

[42] N. R. Pal and J. C. Bezdek, "On cluster validity for the fuzzy c-means model," *IEEE Trans. Fuzzy Syst.*, vol. 3, no. 3, pp. 370–379, Aug. 1995.

[43] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, Sep. 1975.

[44] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 1984, pp. 47–57.

[45] A. Frank and A. Asuncion, *UCI Machine Learning Repository*. Irvine, CA: Univ. California, Irvine, 2010. [Online]. Available: http://archive.ics.uci.edu/ml