Chapter 23

Clustering Validation Measures

Hui Xiong

Rutgers, The State University of New Jersey
Newark, NJ 07102
hxiong@rutgers.edu

Zhongmou Li

Rutgers, The State University of New Jersey
Newark, NJ 07102
mosesli@pegasus.rutgers.edu

23.1 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 [16, 69, 17]. 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.

Clustering validation, which evaluates the goodness of clustering results [41], has long been recognized as one of the vital issues essential to the success of clustering applications [26]. Despite the vast amount of expert endeavor spent on this problem [18, 19, 5], there is no consistent and conclusive solution to cluster validation. The best suitable measures to use in practice remain unknown. Indeed, there are many challenging validation issues which have not been fully addressed in the clustering literature. For instance, the importance of normalizing validation measures has not been fully established. Also, the relationship between different validation measures is not clear. Moreover, there are important properties associated with validation measures which are important to the selection of the use of these measures but have not been well characterized. Finally, given the fact that different validation measures may be appropriate for different clustering algorithms, it is necessary to provide a focused study of cluster validation measures on specified clustering algorithms.

Clustering validation measures can be categorized into two main types: external clustering validation and internal clustering validation. The main difference is whether or not external information is used for clustering validation. External validation measures use external information not present in the data to evaluate the extent to which the clustering structure discovered by a clustering algorithm matches some external structure, e.g., the one specified by the given class labels. One example of external validation measure is entropy, which evaluates the "purity" of clusters based on the given class labels [54]. On the other hand, internal measures evaluate the goodness of a clustering structure without respect to external information [57, 6, 53, 34]. For example, the Silhouette index [49] validates the clustering performance based only on the pairwise difference of between- and within-cluster distances of all data points.

Both external and internal validation measures are crucial for many application scenarios. Since external validation measures know the "true" cluster number in advance, they can be used for choosing an optimal clustering algorithm on a specific data set. For instance, if external validation measures show that a document clustering algorithm can lead to the clustering results which can match the categorization performance by human experts, there is a good reason to believe this clustering algorithm has a practical impact on document clustering. 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 not available in some real world applications. In that case, internal validation measures are the only option for cluster validation when there is no external information available.

However, there are still scenarios that clustering validations measures have limitations in evaluating the goodness of the clustering results. For example, in the case when external criteria are not available and internal validation measures are not very robust, subjective evaluations such as case studies are often used in many contexts, which is particularly common in network clustering algorithms [56, 70, 66]. Another example would be the case that class labels may not reflect cluster structure well, when the class labels do not necessarily correspond to locality. Some clusters may contain a mixture of objects from different classes, thus objects in widely separated clusters may belong to the same class. In this circumstance, using clustering techniques to reveal the data characteristics may not even be a good idea.

In literature, there is a list of clustering validation measures for soft (fuzzy) clustering algorithms. A fuzzy clustering algorithm generates a fuzzy partition to provide a degree of membership of each object to a given cluster. A fuzzy clustering approach is less prone to local minimum than crisp clustering algorithms since it makes soft decisions in each iteration through the use of membership functions [4]. Kim et al. propose a cluster validation measure for fuzzy partitions obtained from fuzzy C-Means algorithm [31]. The proposed validity index exploits an intercluster proximity between fuzzy clusters, which is used to measure the degree of overlap between clusters. The best fuzzy c-partition is obtained by minimizing the intercluster proximity with respect to c. Smyth proposes a cross-validated likelihood measure to determine the appropriate number of clusters in the context of model-based probabilistic clustering [52]. The Xie-Beni index (XB) [63] defines a fuzzy clustering validation function to measure the overall average compactness and separation of a fuzzy c-partition. The intercluster separation is the minimum square distance between cluster centers, and the intracluster compactness is 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. Gath and Geva also proposed a fuzzy validation index which is based on the concepts of hypervolume and density [14].

Another category of related works is validation measures for subspace clustering algorithms. Subspace clustering is an extension of traditional clustering that seeks to find clusters in different subspaces within a data set [45]. Often in high-dimensional data, many dimensions are irrelevant and can mask existing clusters in noisy data. Subspace clustering algorithms localize the search for relevant dimensions allowing them to find clusters that exist in multiple, possibly overlapping, subspaces. Many of the existing validation measures for traditional clustering approaches, such as entropy [1], F-measure [2], and classification error [46], are also used as validation measures for subspace clustering. Muller et al. [44] provide a systematic and thorough evaluation of subspace clustering paradigms.

This chapter focuses on providing a comprehensive study on various aspects of both the external and internal clustering validation measures for crisp clustering algorithms. The sections of this chapter are organized as follows. In Section 23.2, an organized study on 16 external validation measures for K-means clustering is presented. The importance of measure normalization in cluster evaluation on data with imbalanced class distributions is demonstrated, and the normalization solutions for several measures are also provided. Major properties of the external measures, as well as their interrelationships, are presented [62]. Section 23.3 presents a detailed study on 12 widely used internal validation measures for crisp clustering [37, 38]. Properties of these internal measures in different aspects, such as the impact of data with noise, subclusters, and arbitrary shapes, are well investigated. We conclude this chapter with a summary in Section 23.4.

23.2 External Clustering Validation Measures

In literature, a number of external validation measures for crisp clustering have been proposed. In this section, an organized study on a suite of 16 widely used external clustering validation measures as shown in Table 23.1 is presented for the K-means clustering algorithm. These measures represent a good coverage of the external validation measures available in different fields such as data mining, information retrieval, machine learning, and statistics. A common ground of these measures is that they can be computed by the contingency matrix as follows.

The Contingency Matrix. Given a data set D with n objects, assume that there is a partition $P = \{P_1, \cdots, P_K\}$ of D, where $\bigcup_{i=1}^{K} P_i = D$ and $P_i \bigcap P_j = \phi$ for $1 \leq i \neq j \leq K$, and K is the number of clusters. If the "true" class labels for the data are given, another partition can be generated on D:

TABLE 23.1: External Cluster Validation Measures

	Measure	Definition	Range
1	Entropy (E)	$-\sum_i p_i(\sum_j \frac{p_{ij}}{p_i}\log\frac{p_{ij}}{p_i})$	$[0,\log K']$
2	Purity (P)	$\sum_i p_i(\max_j \frac{p_{ij}}{p_i})$	(0,1]
3	F-measure (F)	$\sum_j p_j \max_i[2\frac{p_{ij}}{p_i}\frac{p_{ij}}{p_j}/(\frac{p_{ij}}{p_i}+\frac{p_{ij}}{p_j})]$	(0,1]
4	Variation of Information (VI)	$-\sum_i p_i\log p_i-\sum_j p_j\log p_j-2\sum_i\sum_j p_{ij}\log\frac{p_{ij}}{p_ip_j}$	$[0,2\log\max(K,K')]$
5	Mutual Information (MI)	$\sum_i\sum_j p_{ij}\log\frac{p_{ij}}{p_ip_j}$	$(0,\log K']$
6	Rand statistic (R)	$[\binom{n}{2}-\sum_i\binom{n_{i\cdot}}{2}-\sum_j\binom{n_{\cdot j}}{2}+2\sum_{ij}\binom{n_{ij}}{2}]/\binom{n}{2}$	(0,1]
7	Jaccard coefficient (J)	$\sum_{ij}\binom{n_{ij}}{2}/[\sum_i\binom{n_{i\cdot}}{2}+\sum_j\binom{n_{\cdot j}}{2}-\sum_{ij}\binom{n_{ij}}{2}]$	[0,1]
8	Fowlkes & Mallows index (FM)	$\sum_{ij}\binom{n_{ij}}{2}/\sqrt{\sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}}$	[0,1]
9	Hubert Γ statistic I (Γ)	$\frac{\binom{n}{2}\sum_{ij}\binom{n_{ij}}{2}-\sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}}{\sqrt{\sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}[\binom{n}{2}-\sum_i\binom{n_{i\cdot}}{2}][\binom{n}{2}-\sum_j\binom{n_{\cdot j}}{2}]}}$	(-1,1]
10	Hubert Γ statistic II (Γ')	$[\binom{n}{2}-2\sum_i\binom{n_{i\cdot}}{2}-2\sum_j\binom{n_{\cdot j}}{2}+4\sum_{ij}\binom{n_{ij}}{2}]/\binom{n}{2}$	[0,1]
11	Minkowski score (MS)	$\sqrt{\sum_i\binom{n_{i\cdot}}{2}+\sum_j\binom{n_{\cdot j}}{2}-2\sum_{ij}\binom{n_{ij}}{2}}/\sqrt{\sum_j\binom{n_{\cdot j}}{2}}$	$[0,+\infty)$
12	classification error (ε)	$1-\frac{1}{n}\max_\sigma\sum_j n_{\sigma(j),j}$	[0,1)
13	van Dongen criterion (VD)	$(2n-\sum_i\max_j n_{ij}-\sum_j\max_i n_{ij})/2n$	[0,1)
14	micro-average precision (MAP)	$\sum_i p_i(\max_j\frac{p_{ij}}{p_i})$	(0,1]
15	Goodman-Kruskal coeff (GK)	$\sum_i p_i(1-\max_j\frac{p_{ij}}{p_i})$	[0,1)
16	Mirkin metric (M)	$\sum_i n_{i\cdot}^2+\sum_j n_{\cdot j}^2-2\sum_i\sum_j n_{ij}^2$	$[0,2\binom{n}{2})$

Note: $p_{ij}=n_{ij}/n,\ p_i=n_{i\cdot}/n,\ p_j=n_{\cdot j}/n$.

TABLE 23.2: The Contingency Matrix.

		Partition C				
		C_1	C_2	$\cdots$	$C_{K'}$	$\sum$
Partition P	P_1	n_{11}	n_{12}	$\cdots$	$n_{1K'}$	$n_{1\cdot}$
	P_2	n_{21}	n_{22}	$\cdots$	$n_{2K'}$	$n_{2\cdot}$
	$\cdot$	$\cdot$	$\cdot$	$\cdots$	$\cdot$	$\cdot$
	P_K	n_{K1}	n_{K2}	$\cdots$	$n_{KK'}$	$n_{K\cdot}$
	$\sum$	$n_{\cdot 1}$	$n_{\cdot 2}$	$\cdots$	$n_{\cdot K'}$	n

$C=\{C_1,\cdots,C_{K'}\}$, where $\bigcup_{i=1}^{K'}C_i=D$ and $C_i\bigcap C_j=\phi$ for $1\le i\ne j\le K'$, where K' is the number of classes. Let n_{ij} denote the number of objects in cluster P_i from class C_j, then the information on the overlap between the two partitions can be written in the form of a contingency matrix, as shown in Table 23.2. Consistent notations in this contingency matrix will be used throughout this section.

23.2.1 An Overview of External Clustering Validation Measures

Table 23.1 shows the list of measures to be studied. The "Definition" column gives the computation forms of the measures by using the notations in the contingency matrix. The 16 measures are briefly introduced as follows.

Entropy and purity are frequently used external measures for K-means [54, 67]. They measure the "purity" of the clusters with respect to the given class labels.

F-measure was originally designed for the evaluation of hierarchical clustering [48, 36], but has also been employed for partitional clustering. It combines the precision and recall concepts from the information retrieval community.

The Mutual Information (MI) and Variation of Information (VI) were developed in the field of information theory [8]. MI measures how much information one random variable can tell about

another one [55]. VI measures the amount of information that is lost or gained in changing from the class set to the cluster set [42].

The Rand statistic [47], Jaccard coefficient, Fowlkes and Mallows index [13], and Hubert's two statistics [23, 24] evaluate the clustering quality by the agreements and/or disagreements of the pairs of data objects in different partitions.

The Minkowski score [3] measures the difference between the clustering results and a reference clustering (true clusters). And the difference is computed by counting the disagreements of the pairs of data objects in two partitions.

The classification error takes a classification view on clustering [6]. It tries to map each class to a different cluster so as to minimize the total misclassification rate. The "σ" in Table 23.1 is the mapping of class j to cluster $\sigma(j)$.

The van Dongen criterion [60] was originally proposed for evaluating graph clustering. It measures the representativeness of the majority objects in each class and each cluster.

Finally, the micro-average precision, Goodman-Kruskal coefficient [15], and Mirkin metric [43] are also popular measures. However, the former two are equivalent to the purity measure and the Mirkin metric is equivalent to the Rand statistic ($M/2\binom{n}{2} + R = 1$). Some discussion on GoodMan–Kruskal and Mirkin can be found in Section 23.2.4.3.

In summary, there are 13 (out of 16) candidate measures. Among them, P, F, MI, R, J, FM, Γ, and Γ' are positive measures—a higher value indicates a better clustering performance. The remainder, however, consists of measures based on the distance notion. The acronyms of these measures will be used throughout this section.

23.2.2 Defective Validation Measures

In this section, some validation measures which produce misleading validation results for K-means on data with skewed class distributions are presented.

23.2.2.1 K-Means: The Uniform Effect

One of the unique characteristic of K-means clustering is the so-called uniform effect; that is, K-means tends to produce clusters with relatively uniform sizes [64]. The coefficient of variation (CV) [10], a statistic which measures the dispersion degree of a random distribution, is used to quantify the uniform effect. CV is defined as the ratio of the standard deviation to the mean. Given a sample of data objects $X = \{x_1, x_2, \ldots, x_n\}$, $CV = s/\bar{x}$, where $\bar{x} = \sum_{i=1}^{n} x_i/n$ and $s = \sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2/(n-1)}$. CV is a dimensionless number that allows the comparison of the variations of populations that have significantly different mean values. In general, the larger the CV value is, the greater the variability in the data.

Example. Let CV_0 denote the CV value of the "true" class sizes and CV_1 denote the CV value of the resultant cluster sizes. The `sports` data set [59] is used to illustrate the uniform effect by K-means. The "true" class sizes of `sports` have $CV_0 = 1.02$. Then the CLUTO implementation of K-means [27] with default settings is employed to cluster `sports` into seven clusters, and the CV value of the resultant cluster sizes is 0.42. Therefore, the CV difference is $DCV = CV_1 - CV_0 = -0.6$, which indicates a significant uniform effect in the clustering result.

Indeed, it has been empirically validated that the 95% confidence interval of CV_1 values produced by K-means is in $[0.09, 0.85]$ [61]. In other words, for data sets with CV_0 values greater than 0.85, the uniform effect of K-means can distort the cluster distribution significantly.

Now the question is: Can these widely used validation measures capture the negative uniform effect by K-means clustering? Next, a necessary but not sufficient criterion is provided to testify whether a validation measure can be effectively used to evaluate K-means clustering.

TABLE 23.3: Two Clustering Results

I	C_1	C_2	C_3	C_4	C_5
P_1	10	0	0	0	0
P_2	10	0	0	0	0
P_3	10	0	0	0	0
P_4	0	0	0	10	0
P_5	0	2	6	0	2

II	C_1	C_2	C_3	C_4	C_5
P_1	27	0	0	2	0
P_2	0	2	0	0	0
P_3	0	0	6	0	0
P_4	3	0	0	8	0
P_5	0	0	0	0	2

23.2.2.2 A Necessary Selection Criterion

Assume that there is a sample document data containing 50 documents from 5 classes. The class sizes are 30, 2, 6, 10 and 2. Thus, $CV_0 = 1.166$, which implies a skewed class distribution.

For this sample data set, assume that there are two clustering results as shown in Table 23.3. In the table, the first result consists of five clusters with extremely balanced sizes. This is also indicated by $CV_1 = 0$. In contrast, for the second result, the five clusters have varied cluster sizes with $CV_1 = 1.125$, much closer to the CV value of the "true" class sizes. Therefore, from a data distribution point of view, the second result should be better than the first one.

Indeed, by taking a closer look on contingency Matrix I in Table 23.3, one can find that the first clustering partitions the objects of the largest class C_1 into three balanced subclusters. Meanwhile, the two small classes C_2 and C_5 have totally "disappeared"—they are overwhelmed in cluster P_5 by the objects from class C_3. In contrast, it is easy to identify all the classes in the second clustering result, since they have the majority of objects in the corresponding clusters. Therefore, it is the conclusion that the first clustering is indeed much worse than the second one.

As shown in Section 23.2.2.1, K-means tends to produce clusters with relatively uniform sizes. Thus the first clustering in Table 23.3 can be regarded as the negative result of the uniform effect. So the first necessary but not sufficient criterion for selecting the measures for K-means is as follows.

Criterion 1 *If an external validation measure cannot capture the uniform effect by K-means on data with skewed class distributions, this measure is not suitable for validating the results of K-means clustering.*

The performance of existing external cluster validation measures for this criterion is presented in next section.

23.2.2.3 The Cluster Validation Results

Table 23.4 shows the validation results for the two clusterings in Table 23.3 by all 13 external validation measures. The better evaluation of each validation measure is highlighted.

As shown in Table 23.4, only three measures, E, P, and MI, cannot capture the uniform effect by K-means and their validation results can be misleading. In other words, these measures are not suitable for evaluating the K-means clustering. These three measures are defective validation measures.

TABLE 23.4: The Cluster Validation Results

	E	*P*	*F*	*MI*	*VI*	*R*	*J*	*FM*	Γ	Γ′	*MS*	ε	*VD*
I	**0.274**	**0.920**	0.617	**1.371**	1.225	0.732	0.375	0.589	0.454	0.464	0.812	0.480	0.240
II	0.396	0.9	**0.902**	1.249	**0.822**	**0.857**	**0.696**	**0.821**	**0.702**	**0.714**	**0.593**	**0.100**	**0.100**

23.2.2.4 The Issues with the Defective Measures

First, the problem of the **entropy** measure lies in the fact that it cannot evaluate the integrity of the classes since $E = -\sum_i p_i \sum_j \frac{p_{ij}}{p_i} \log \frac{p_{ij}}{p_i}$. If a random variable view on cluster P and class C is taken, then $p_{ij} = n_{ij}/n$ is the joint probability of the event: $\{P = P_i \bigwedge C = C_j\}$, and $p_i = n_{i\cdot}/n$ is the marginal probability. Therefore, $E = \sum_i p_i \sum_j -p(C_j|P_i) \log p(C_j|P_i) = \sum_i p_i H(C|P_i) = H(C|P)$, where $H(\cdot)$ is the Shannon entropy [8]. The above implies that the entropy measure is nothing but the conditional entropy of C on P. In other words, if the objects in each large partition are mostly from the same class, the entropy value tends to be small (indicating a better clustering quality). This is usually the case for K-means clustering on highly imbalanced data sets, since K-means tends to partition a large class into several pure subclusters. This leads to the problem that the integrity of the objects from the same class has been damaged. The entropy measure cannot capture this information and penalize it.

The **mutual information** is strongly related to the entropy measure, which is illustrated by the following lemma.

Lemma 23.2.1 *The mutual information measure is equivalent to the entropy measure for cluster validation.*

PROOF. By information theory, $MI = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{p_i p_j} = H(C) - H(C|P) = H(C) - E$. Since $H(C)$ is a constant for any given data set, MI is essentially equivalent to E. □

The **purity** measure works in a similar fashion as the entropy measure. That is, it measures the "purity" of each cluster by the ratio of the objects from the majority class. Thus, it has the same problem as the entropy measure for evaluating K-means clustering.

In summary, entropy, purity, and mutual information are defective measures for validating K-means clustering.

23.2.2.5 Improving the Defective Measures

In this section, the improved versions of the above three defective measures entropy, mutual information, and purity are provided.

Lemma 23.2.2 *The Variation of Information measure is an improved version of the entropy measure.*

PROOF. Cluster P and class C can be viewed as two random variables, and it has been shown that $VI = H(C) + H(P) - 2MI = H(C|P) + H(P|C)$ [42]. The component $H(C|P)$ is nothing but the entropy measure, and the component $H(P|C)$ is a valuable supplement to $H(C|P)$. That is, $H(P|C)$ evaluates the integrity of each class along different clusters. □

Since MI is equivalent to E according to Lemma 23.2.1, therefore, VI is also an improved version of MI.

Lemma 23.2.3 *The van Dongen criterion is an improved version of the purity measure.*

PROOF. $VD = \frac{2n - \sum_i \max_j n_{ij} - \sum_j \max_i n_{ij}}{2n} = 1 - \frac{1}{2}P - \frac{\sum_j \max_i n_{ij}}{2n}$. Apparently, $\sum_j \max_i n_{ij}/n$ reflects the integrity of the classes and is a supplement to the purity measure. □

23.2.3 Measure Normalization

In this section, discussions on the importance of measure normalization and the normalization solutions of different measures are presented.

23.2.3.1 Normalizing the Measures

Generally speaking, normalizing techniques can be divided into two categories. One is based on a statistical view, which formulates a baseline distribution to correct the measure for randomness. A clustering can then be termed "valid" if it has an unusually high or low value, as measured with respect to the baseline distribution. The other technique uses the minimum and maximum values to normalize the measure into the [0,1] range. From a statistical view, it is equivalent to view this technique with the assumption that each measure takes a uniform distribution over the value interval.

***The Normalizations of* R, FM, Γ, Γ′, J, *and* MS**. The normalization scheme can take the form:

$$S_n = \frac{S - E(S)}{\max(S) - E(S)} \tag{23.1}$$

where $\max(S)$ is the maximum value of the measure S and $E(S)$ is the expected value of S based on the baseline distribution. Some measures derived from the statistics community, such as R, FM, Γ, and Γ', usually take this scheme.

Specifically, Hubert and Arabie (1985) [24] suggested using the multivariate hypergeometric distribution as the baseline distribution in which the row and column sums are fixed in Table 23.2, but the partitions are randomly selected. This determines the expected value as follows:

$$E(\sum_i \sum_j \binom{n_{ij}}{2}) = \frac{\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}}{\binom{n}{2}} \tag{23.2}$$

Based on this value, it is easy to compute the expected values of R, FM, Γ, and Γ' since they are the linear functions of $\sum_i \sum_j \binom{n_{ij}}{2}$ under the hypergeometric distribution assumption. Furthermore, although the exact maximum values of the measures are computationally prohibited under the hypergeometric distribution assumption, it is still reasonable to approximate them by 1. Then, according to Equations (23.1) and (23.2), the normalized R, FM, Γ, and Γ' measures can be calculated as shown in Table 23.5.

The normalization of J and MS is a little bit complex, since they are not linear to $\sum_i \sum_j \binom{n_{ij}}{2}$. Nevertheless, one can still normalize the equivalent measures converted from them. Let $J' = \frac{1-J}{1+J} = \frac{2}{1+J} - 1$ and $MS' = MS^2$.

It is easy to show $J' \Leftrightarrow J$ and $MS' \Leftrightarrow MS$. Then, based on the hypergeometric distribution assumption, the normalized J' and MS' can be calculated as shown in Table 23.5. Since J' and MS' are

TABLE 23.5: The Normalized Measures

	Measure	Normalization
1	R_n	$(m - m_1m_2/M)/(m_1/2 + m_2/2 - m_1m_2/M)$
2	FM_n	$(m - m_1m_2/M)/(\sqrt{m_1m_2} - m_1m_2/M)$
3	Γ_n	$(mM - m_1m_2)/\sqrt{m_1m_2(M-m_1)(M-m_2)}$
4	Γ'_n	$(m - m_1m_2/M)/(m_1/2 + m_2/2 - m_1m_2/M)$
5	J'_n	$(m_1 + m_2 - 2m)/(m_1 + m_2 - 2m_1m_2/M)$
6	MS'_n	$(m_1 + m_2 - 2m)/(m_1 + m_2 - 2m_1m_2/M)$
7	VI_n	$1 + 2\frac{\sum_i \sum_j p_{ij} \log(p_{ij}/p_i p_j)}{(\sum_i p_i \log p_i + \sum_j p_j \log p_j)}$
8	VD_n	$\frac{(2n - \sum_i \max_j n_{ij} - \sum_j \max_i n_{ij})}{(2n - \max_i n_{i\cdot} - \max_j n_{\cdot j})}$
9	F_n	$(F - F_-)/(1 - F_-)$
10	ε_n	$(1 - \frac{1}{n}\max_\sigma \sum_j n_{\sigma(j),j})/(1 - 1/\max(K, K'))$

Note: (1) $m = \sum_{i,j} \binom{n_{ij}}{2}$, $m_1 = \sum_i \binom{n_{i\cdot}}{2}$, $m_2 = \sum_j \binom{n_{\cdot j}}{2}$, $M = \binom{n}{2}$.
(2) $p_i = n_{i\cdot}/n$, $p_j = n_{\cdot j}/n$, $p_{ij} = n_{ij}/n$.
(3) Refer to Table 23.1 for F, and Procedure 1 for F_-.

negative measures—a lower value implies a better clustering—they are normalized by modifying Equation (23.1) as $S_n = (S - \min(S))/(E(S) - \min(S))$.

Finally, there are some interrelationships between these measures as follows.

Proposition 23.2.1

(1) $(R_n \equiv \Gamma'_n) \Leftrightarrow (J'_n \equiv MS'_n)$
(2) $\Gamma_n \equiv \Gamma$

The above proposition indicates that the normalized Hubert Γ statistic I (Γ_n) is the same as Γ. Also, the normalized Rand statistic (R_n) is the same as the normalized Hubert Γ statistic II (Γ'_n). In addition, the normalized Rand statistic (R_n) is equivalent to J'_n, which is the same as MS'_n. Therefore, there are only three independent normalized measures, R_n, FM_n, and Γ_n, needed for further study. The proposition can be easily proved by mathematical transformation, and due to the space limitation, the proof is omitted.

The Normalizations of VI and VD. Another normalization scheme is formalized as

$$S_n = \frac{S - \min(S)}{\max(S) - \min(S)} \tag{23.3}$$

Some measures, such as VI and VD, often take this scheme. However, to know the exact maximum and minimum values is often impossible. So it usually turns to a reasonable approximation, e.g., the upper bound for the maximum or the lower bound for the minimum.

When the cluster structure matches the class structure perfectly, $VI = 0$. So, $\min(VI) = 0$. However, finding the exact value of $\max(VI)$ is computationally infeasible. Meila [42] suggested using $2\log\max(K, K')$ to approximate $\max(VI)$, so the normalized VI is $\frac{VI}{2\log\max(K,K')}$.

The VD in Table 23.1 can be regarded as a normalized measure. In this measure, $2n$ has been taken as the upper bound [60], and $\min(VD) = 0$.

However, the above normalized VI and VD cannot well capture the uniform effect of K-means, because the proposed upper bound for VI or VD is not tight enough. Two new tighter upper bounds are introduced as follows.

Lemma 23.2.4 *Let random variables C and P denote the class and cluster sizes, respectively, and $H(\cdot)$ be the entropy function; then $VI \leq H(C) + H(P) \leq 2\log\max(K', K)$.*

Lemma 23.2.4 gives a tighter upper bound $H(C) + H(P)$ than $2\log\max(K', K)$ which was provided by Meila [42]. With this new upper bound, the normalized VI_n can be calculated as shown in Table 23.5. In addition, if $H(P)/2 + H(C)/2$ is used as the upper bound to normalize mutual information, the VI_n can be equivalent to the normalized mutual information MI_n ($VI_n + MI_n = 1$).

Lemma 23.2.5 *Let $n_{i\cdot}$, $n_{\cdot j}$, and n be the values in Table 23.2, then $VD \leq (2n - \max_i n_{i\cdot} - \max_j n_{\cdot j})/2n \leq 1$.*

Due to space limitations, the proofs are omitted. The above two lemmas imply that the tighter upper bounds of VI and VD are the functions of the class and cluster sizes. Using these two new upper bounds, the normalized VI_n and VD_n can be derived as in Table 23.5.

The Normalization of F and ε have seldom been discussed in the literature. Since $\max(F) = 1$, now the goal is to find a tight lower bound for F, which can be found by Procedure 1.

With Procedure 1, the following lemma can find a lower bound for F.

Lemma 23.2.6 *Given F_- computed by Procedure 1, $F \geq F_-$.*

Procedure 1: The computation of F_-.

Let $n^* = \max_i n_{i\cdot}$.
Sort the class sizes so that $n_{\cdot[1]} \le n_{\cdot[2]} \le \cdots \le n_{\cdot[K']}$.
Let $a_j = 0$, for $j = 1, 2, \cdots, K'$.
for $j = 1 : K'$
  if $n^* \le n_{\cdot[j]}$, $a_j = n^*$, **break**.
  else $a_j = n_{\cdot[j]}$, $n^* \leftarrow n^* - n_{\cdot[j]}$.
$F_- = (2/n)\sum_{j=1}^{K'} a_j/(1 + \max_i n_{i\cdot}/n_{\cdot[j]})$.

PROOF. It is easy to show

$$F = \sum_j \frac{n_{\cdot j}}{n} \max_i \frac{2n_{ij}}{n_{i\cdot} + n_{\cdot j}} \ge \frac{2}{n} \max_i \sum_j \frac{n_{ij}}{n_{i\cdot}/n_{\cdot j} + 1} \tag{23.4}$$

Consider an optimization problem as follows:

$$\min_{x_{ij}} \sum_j \frac{x_{ij}}{n_{i\cdot}/n_{\cdot j} + 1}$$

$$s.t. \sum_j x_{ij} = n_{i\cdot};\ \forall j,\ x_{ij} \le n_{\cdot j};\ \forall j,\ x_{ij} \in \mathbb{Z}_+$$

For this optimization problem, to have the minimum objective value, as many objects as possible need to be assigned to the cluster with highest $n_{i\cdot}/n_{\cdot j} + 1$, or equivalently, with smallest $n_{\cdot j}$. Let $n_{\cdot[0]} \le n_{\cdot[1]} \le \cdots \le n_{\cdot[K']}$ where the virtual $n_{\cdot[0]} = 0$, and assume $\sum_{j=0}^{l} n_{\cdot[j]} < n_{i\cdot} \le \sum_{j=0}^{l+1} n_{\cdot[j]},\ l \in \{0, 1, \cdots, K' - 1\}$. The optimal solution is given as

$$x_{i[j]} = \begin{cases} n_{\cdot[j]},\ 1 \le j \le l \\ n_{i\cdot} - \sum_{k=1}^{l} n_{\cdot[k]},\ j = l+1 \\ 0,\ l+1 < j \le K' \end{cases}$$

Therefore, according to (23.4), $F \ge \frac{2}{n} \max_i \sum_{j=1}^{K'} \frac{x_{i[j]}}{n_{i\cdot}/n_{\cdot[j]}+1}$.

Let $F_i = \frac{2}{n}\sum_{j=1}^{K'} \frac{x_{i[j]}}{n_{i\cdot}/n_{\cdot[j]}+1} = \frac{2}{n}\sum_{j=1}^{K'} \frac{x_{i[j]}/n_{i\cdot}}{1/n_{\cdot[j]}+1/n_{i\cdot}}$. Denote "$x_{i[j]}/n_{i\cdot}$" by "$y_{i[j]}$", and "$\frac{1}{1/n_{\cdot[j]}+1/n_{i\cdot}}$" by "$p_{i[j]}$", then $F_i = \frac{2}{n}\sum_{j=1}^{K'} p_{i[j]} y_{i[j]}$. Next, it remains to show

$$\arg\max_i F_i = \arg\max_i n_{i\cdot}$$

Assume $n_{i\cdot} \le n_{i'\cdot}$, and for some l, $\sum_{j=0}^{l} n_{\cdot[j]} < n_{i\cdot} \le \sum_{j=0}^{l+1} n_{\cdot[j]},\ \ l \in \{0, 1, \cdots, K' - 1\}$. This implies that

$$y_{i[j]} \begin{cases} \ge y_{i'[j]},\ 1 \le j \le l \\ \le y_{i'[j]},\ l+1 < j \le K' \end{cases}$$

Since $\sum_{j=1}^{K'} y_{i[j]} = \sum_{j=1}^{K'} y_{i'[j]} = 1$ and $j\uparrow\ \Rightarrow\ p_{i[j]}\uparrow$, thus $\sum_{j=1}^{K'} p_{i[j]} y_{i[j]} \le \sum_{j=1}^{K'} p_{i[j]} y_{i'[j]}$. Furthermore, according to the definition of $p_{i[j]}$, $p_{i[j]} \le p_{i'[j]},\ \forall\ j \in \{1, \cdots, K'\}$. Therefore,

$$F_i = \frac{2}{n}\sum_{j=1}^{K'} p_{i[j]} y_{i[j]} \le \frac{2}{n}\sum_{j=1}^{K'} p_{i[j]} y_{i'[j]} \le \frac{2}{n}\sum_{j=1}^{K'} p_{i'[j]} y_{i'[j]} = F_i'$$

which implies that $n_{i\cdot} \le n_{i'\cdot}$ is the sufficient condition for $F_i \le F_i'$. Therefore, by Procedure 1, $F_- = \max_i F_i$, which finally leads to $F \ge F_-$. □

Therefore, $F_n = (F - F_-)/(1 - F_-)$, as listed in Table 23.5. Finally, the following lemma provides an upper bound of ε.

Lemma 23.2.7 *Given $K' \leq K$, $\varepsilon \leq 1 - 1/K$.*

PROOF. Assume $\sigma_1 : \{1, \cdots, K'\} \to \{1, \cdots, K\}$ is the optimal mapping of the classes to different clusters, i.e.,

$$\varepsilon = 1 - \frac{\sum_{j=1}^{K'} n_{\sigma_1(j),j}}{n}$$

Then construct a series of mappings $\sigma_s : \{1, \cdots, K'\} \mapsto \{1, \cdots, K\}$ $(s = 2, \cdots, K)$ which satisfy

$$\sigma_{s+1}(j) = \text{mod}(\sigma_s(j), K) + 1, \ \forall j \in \{1, \cdots, K'\}$$

where "mod(x, y)" returns the remainder of positive integer x divided by positive integer y. By definition, σ_s $(s = 2, \cdots, K)$ can also map $\{1, \cdots, K'\}$ to K' different indices in $\{1, \cdots, K\}$ as σ_1. More importantly, $\sum_{j=1}^{K'} n_{\sigma_1(j),j} \geq \sum_{j=1}^{K'} n_{\sigma_s(j),j}$, $\forall s = 2, \cdots, K$, and $\sum_{s=1}^{K} \sum_{j=1}^{K'} n_{\sigma_s(j),j} = n$. Therefore, $\sum_{j=1}^{K'} n_{\sigma_1(j),j} \geq \frac{n}{K}$, which implies $\varepsilon \leq 1 - 1/K$. The proof is completed. □

Therefore, $1 - 1/K$ can be used as the upper bound of ε, and the normalized ε_n is shown in Table 23.5.

23.2.3.2 The *DCV* Criterion

In this section, some experiments are presented to show the importance of *DCV* $(CV_1 - CV_0)$ for selecting validation measures.

Experimental Data Sets. Some synthetic data sets were generated as follows. Assume there is a two-dimensional mixture of two Gaussian distributions. The means of the two distributions are [-2,0] and [2,0]. And their covariance matrices are exactly the same as $[\sigma^2\ 0;\ 0\ \sigma^2]$.

Therefore, given any specific value of σ^2, one can generate a simulated data set with 6000 instances, n_1 instances from the first distribution, and n_2 instances from the second one, where $n_1 + n_2 = 6000$. To produce simulated data sets with imbalanced class sizes, set a series of n_1 values: {3000, 2600, 2200, 1800, 1400, 1000, 600, 200}. If $n_1 = 200$, $n_2 = 5800$, the data set is highly imbalanced with $CV_0 = 1.320$. For each mixture model, 8 simulated data sets were generated with CV_0 ranging from 0 to 1.320. Further, to produce data sets with different clustering tendencies, set a series of σ^2 values: {0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5}. As σ^2 increases, the mixture model tends to be more unidentifiable. Finally, for each pair of σ^2 and n_1, the sampling was repeated 10 times to have the average performance evaluation. In summary, $8 \times 10 \times 10 = 800$ data sets were produced. Figure 23.1 shows a sample data set with $n_1 = 1000$ and $\sigma^2 = 2.5$.

A sampling on a real-world data set `hitech` was also conducted to get some sample data sets with imbalanced class distributions. This data set was derived from the San Jose Mercury newspaper articles [59], which contains 2301 documents about computers, electronics, health, medical, research, and technology. Each document is characterized by 126373 terms, and the class sizes are 485, 116, 429, 603, 481, and 187. Carefully setting the sampling ratio for each class, 8 sample data sets were extracted with the class-size distributions (CV_0) ranging from 0.490 to 1.862, as shown in Table 23.6. For each data set, the sampling was repeated 10 times to observe the averaged clustering performance.

Experimental Tools. The MATLAB 7.1 [40] and CLUTO 2.1.2 [27] implementations of *K*-means were employed for the experiment. The MATLAB version with the squared Euclidean distance is suitable for low-dimensional and dense data sets, while CLUTO with the cosine similarity is used to handle high-dimensional and sparse data sets. Note that the number of clusters, i.e., K, was set to match the number of "true" classes.

The Application of Criterion 1. Here, how Criterion 1 can be applied for selecting measures is

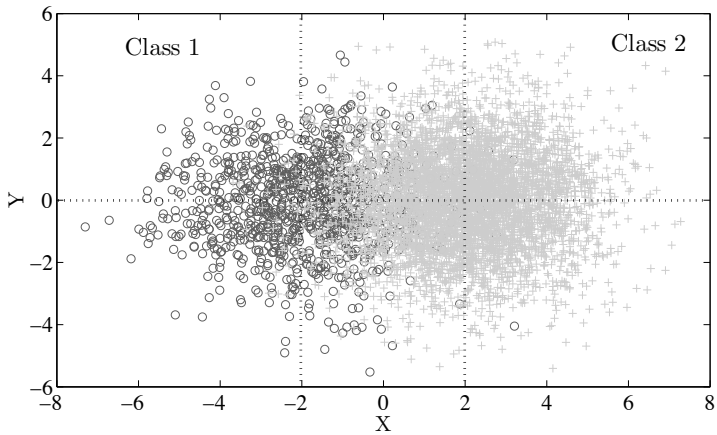


FIGURE 23.1 (See color insert): A simulated data set ($n_1 = 1000$, $\sigma^2 = 2.5$).

presented. As pointed out in Section 23.2.2.1, *K*-means tends to have the uniform effect on imbalanced data sets. This implies that for data sets with skewed class distributions, the clustering results by *K*-means tend to be away from "true" class distributions.

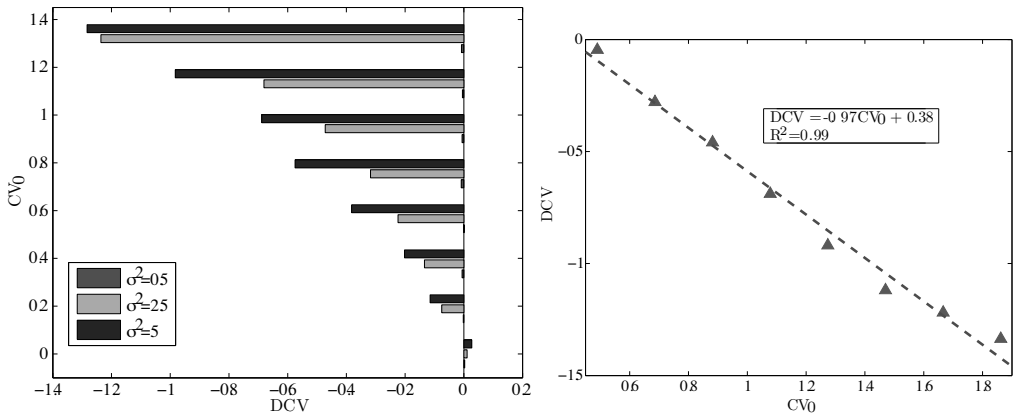
To further illustrate this, take a look at Figure 23.2(a) of the simulated data sets. As can be seen, for the extreme case of $\sigma^2 = 5$, the *DCV* values decrease as the CV_0 values increase. Note that *DCV* values are usually negative since *K*-means tends to produce clustering results with relative uniform cluster sizes ($CV_1 < CV_0$). This means that when data become more skewed, the clustering results by *K*-means tend to be worse. Therefore, the selection of measures can be done by observing the relationship between the measures and the *DCV* values. As the *DCV* values go down, the good measures are expected to show worse clustering performances. In this experiment, the MATLAB version of *K*-means was applied.

A similar trend can be found in Figure 23.2(b) of the sampled data sets. That is, as the CV_0 values go up, the *DCV* values decrease, which implies worse clustering performances. Indeed, *DCV* is a good indicator for finding the measures which cannot capture the uniform effect by *K*-means clustering. In this experiment, the CLUTO version of *K*-means clustering was applied.

In Section 23.2.4, the Kendall's rank correlation is used (κ) [30] to measure the relationships between external validation measures and DCV. Note that, $\kappa \in [-1, 1]$. $\kappa = 1$ indicates a perfect positive rank correlation, whereas $\kappa = -1$ indicates an extremely negative rank correlation.

TABLE 23.6: The Sizes of the Sampled Data Sets

Data Set	1	2	3	4	5	6	7	8
Class 1	100	90	80	70	60	50	40	30
Class 2	100	90	80	70	60	50	40	30
Class 3	100	90	80	70	60	50	40	30
Class 4	250	300	350	400	450	500	550	600
Class 5	100	90	80	70	60	50	40	30
Class 6	100	90	80	70	60	50	40	30
CV_0	0.49	0.686	0.88	1.078	1.27	1.47	1.666	1.862



(a) Simulated Data Sets.

(b) Sampled Data Sets.

FIGURE 23.2: Relationship of CV_0 and DCV.

23.2.3.3 The Effect of Normalization

The importance of measure normalization is presented in this section. Along this line, K-means clustering is first applied on the simulated data sets with $\sigma^2 = 5$ and the sampled data sets from `hitech`. Then, both unnormalized and normalized measures are used for cluster validation. Finally, the rank correlation between DCV and the measures are computed and the results are shown in Table 23.7.

As can be seen in the table, if the unnormalized measures are used to do cluster validation, only three measures, namely R, Γ, Γ', have strong consistency with DCV on both groups of data sets. VI, VD and MS even show strong conflict with DCV on the sampled data sets, since their κ values are all close to -1 on sampled data. In addition, notice that F, ε, J, and FM show weak correlation with DCV.

Table 23.7 shows the rank correlations between DCV and the normalized measures. As can be seen, all the normalized measures show perfect consistency with DCV except for F_n and ε_n. This indicates that the normalization is crucial for evaluating K-means clustering. The proposed bounds for the measures are tight enough to capture the uniform effect in the clustering results.

In Table 23.7, it can be observed that both F_n and ε_n are not consistent with DCV. This indicates that normalization does not help F and ε too much. The reason is that the proposed lower bound for F and upper bound for ε are not very tight. Indeed, the normalizations of F and ε are very challenging due to the fact that they both exploit relatively complex optimization schemes in

TABLE 23.7: The Correlation between DCV and the Validation Measures

κ	VI	VD	MS	ε	F	R	J	FM	Γ	Γ'
Simulated Data	***-0.71***	***0.79***	***-0.79***	1.00	1.00	1.00	0.91	***0.71***	1.00	1.00
Sampled Data	***-0.93***	***-1.00***	***-1.00***	***0.50***	***0.21***	1.00	***0.50***	***-0.43***	0.93	1.00
κ	VI_n	VD_n	MS'_n	ε_n	F_n	R_n	J'_n	FM_n	Γ_n	Γ'_n
Simulated Data	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Sampled Data	1.00	1.00	1.00	***0.50***	***0.79***	1.00	1.00	1.00	0.93	1.00

Note: Poor or even negative correlations have been highlighted by the bold and italic fonts.

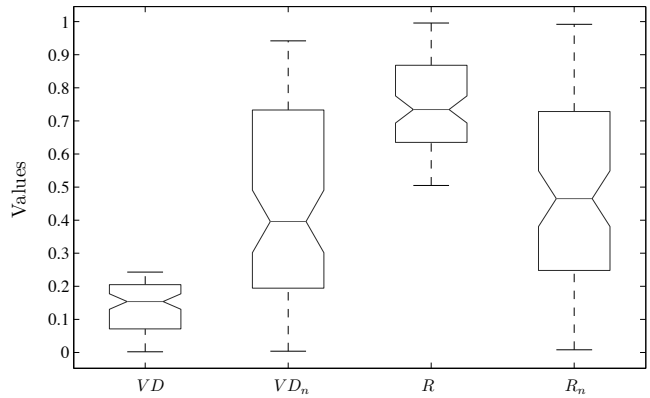


FIGURE 23.3: Unnormalized and normalized measures.

the computations. As a result, it is not easy to compute the expected values from a multivariate hypergeometric distribution perspective, and it is also difficult to find tighter bounds.

Nevertheless, the above experiments show that the normalization is very valuable. In addition, Figure 23.3 shows the cluster validation results of the measures on all the simulated data sets with σ^2 ranging from 0.5 to 5. It is clear that the normalized measures have much wider value range than the unnormalized ones along [0, 1]. This indicates that the values of normalized measures are more spread in [0, 1].

In summary, to compare cluster validation results across different data sets, normalized measures should be used.

23.2.4 Measure Properties

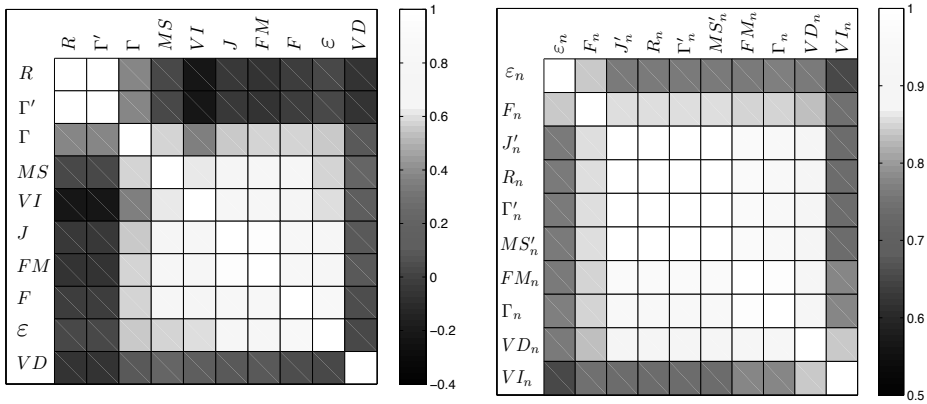
Measure properties, which can serve as the guidance for the selection of measures, are investigated in this section.

23.2.4.1 The Consistency Between Measures

Here, the consistency between a pair of measures is defined as the similarity between their rankings on a series of clustering results. The similarity is measured by the Kendall's rank correlation. And the clustering results are produced by the CLUTO version of K-means clustering on 29 benchmark real-world data sets listed in Table 23.8. In the experiment, for each data set, the cluster number is set to be the same as the "true" class number.

Figures 23.4(a) and 23.4(b) show the correlations between the unnormalized and normalized measures, respectively. One interesting observation is that the normalized measures have a much stronger consistency than the unnormalized measures. For instance, the correlation between VI and R is merely -0.21, but it reaches 0.74 for the corresponding normalized measures. This observation indeed implies that the normalized measures tend to give more robust validation results, which also agrees with previous analysis.

According to the colors in Figure 23.4(b) on the normalized measures, it can be roughly found that R_n, Γ'_n, J'_n, MS'_n, FM_n, and Γ_n are more similar to one another, while VD_n, F_n, VI_n, and ε_n show inconsistency with others in varying degrees. To gain the precise understanding, a hierarchical



(a) Unnormalized Measures. (b) Normalized Measures.

FIGURE 23.4 (See color insert): Correlations of the measures.

clustering on the measures is performed by using their correlation matrix. The resultant hierarchy can be found in Figure 23.5 ("s" means the similarity). As mentioned before, R_n, Γ'_n, J'_n, and MS'_n are equivalent, so they have perfect correlation with one another and form the first group. The second group contains FM_n and Γ_n. These two measures behave similarly, and have just slightly weaker consistency with the measures in the first group. Finally, VD_n, F_n, ε_n, and VI_n have obviously weaker consistency with other measures in a descending order.

Furthermore, the data sets in Table 23.8 are divided into two repositories to explore the source of the inconsistency among the measures, where $\Re_1$ contains data sets with $CV_0 < 0.8$, and $\Re_2$ contains the rest. After computing the correlation matrices of the measures on the two repositories (denoted by $M(\Re_1)$ and $M(\Re_2)$), their difference $(M(\Re_1) - M(\Re_2))$ can be calculated as shown in Table 23.9. As can be seen, roughly speaking, all the measures except VI_n show weaker consistency with one another on data sets in $\Re_2$. In other words, while VI_n acts in the opposite way, most measures tend to disagree with one another on data sets with highly imbalanced classes.

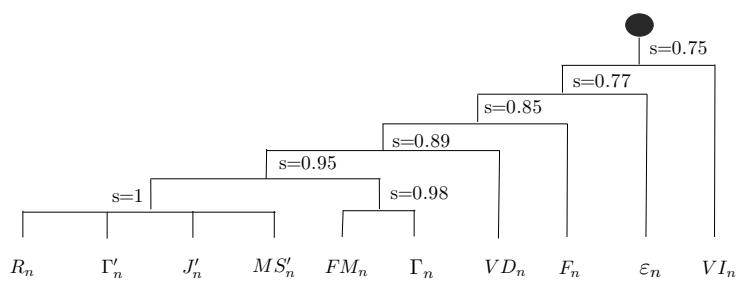


FIGURE 23.5: The measure similarity hierarchy.

TABLE 23.8: The Benchmark Data Sets

Data Set	Source	#Class	#Case	#Feature	CV_0
cacmcisi	CA/CI	2	4663	41681	0.53
classic	CA/CI	4	7094	41681	0.55
cranmed	CR/ME	2	2431	41681	0.21
fbis	TREC	17	2463	2000	0.96
hitech	TREC	6	2301	126373	0.50
k1a	WebACE	20	2340	21839	1.00
k1b	WebACE	6	2340	21839	1.32
la1	TREC	6	3204	31472	0.49
la2	TREC	6	3075	31472	0.52
la12	TREC	6	6279	31472	0.50
mm	TREC	2	2521	126373	0.14
ohscal	OHSUMED	10	11162	11465	0.27
re0	Reuters	13	1504	2886	1.50
re1	Reuters	25	1657	3758	1.39
sports	TREC	7	8580	126373	1.02
tr11	TREC	9	414	6429	0.88
tr12	TREC	8	313	5804	0.64
tr23	TREC	6	204	5832	0.93
tr31	TREC	7	927	10128	0.94
tr41	TREC	10	878	7454	0.91
tr45	TREC	10	690	8261	0.67
wap	WebACE	20	1560	8460	1.04
DLBCL	KRBDSR	3	77	7129	0.25
Leukemia	KRBDSR	7	325	12558	0.58
LungCancer	KRBDSR	5	203	12600	1.36
ecoli	UCI	8	336	7	1.16
pageblocks	UCI	5	5473	10	1.95
letter	UCI	26	20000	16	0.03
pendigits	UCI	10	10992	16	0.04
MIN	-	2	77	7	0.03
MAX	-	26	20000	126373	1.95

Note: CA-CACM, CI-CISI, CR-CRANFIELD, ME-MEDLINE.

TABLE 23.9: $M(\Re_1) - M(\Re_2)$

	R_n	FM_n	Γ_n	VD_n	F_n	ε_n	VI_n
R_n	0.00	0.09	0.13	0.08	0.10	0.26	-0.01
FM_n	0.09	0.00	0.04	0.00	0.10	0.22	-0.10
Γ_n	0.13	0.04	0.00	0.04	0.14	0.22	-0.06
VD_n	0.08	0.00	0.04	0.00	0.05	0.20	-0.18
F_n	0.10	0.10	0.14	0.05	0.00	0.08	-0.08
ε_n	0.26	0.22	0.22	0.20	0.08	0.00	0.04
VI_n	-0.01	-0.10	-0.06	-0.18	-0.08	0.04	0.00

23.2.4.2 Properties of Measures

In this section, some key properties of external clustering validation measures are discussed.

The Sensitivity. The measures have different sensitivity to the clustering results. It can be illustrated by an example. For two clustering results in Table 23.10, the differences between them are

TABLE 23.10: Two Clustering Results

I	C_1	C_2	C_3	$\sum$
P_1	**3**	**4**	12	19
P_2	**8**	**3**	12	23
P_3	12	12	0	24
$\sum$	23	19	24	66

II	C_1	C_2	C_3	$\sum$
P_1	**0**	**7**	12	19
P_2	**11**	**0**	12	23
P_3	12	12	0	24
$\sum$	23	19	24	66

TABLE 23.11: The Cluster Validation Results

	R_n	FM_n	Γ_n	VD_n	F_n	ε_n	VI_n
I	0.16	0.16	0.16	0.71	0.32	0.77	0.78
II	0.24	0.24	0.24	0.71	0.32	0.70	0.62

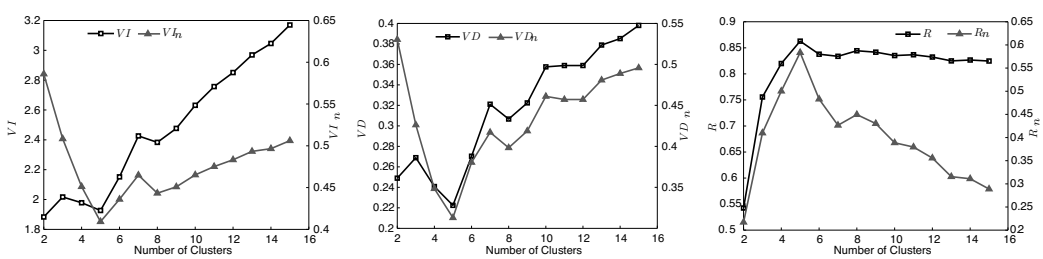
the numbers in bold. Validation results of the measures on these two clusterings are shown in Table 23.11. As can be seen, all the measures show different validation results for the two clusterings except for VD_n and F_n. This implies that VD_n and F_n are less sensitive than other measures. This is due to the fact that both VD_n and F_n use maximum functions, which may lose some information in the contingency matrix. Furthermore, VI_n is the most sensitive measure, since the difference of VI_n values for the two clusterings is the largest.

Impact of the Number of Clusters. The impact of the number of clusters on the validation measures is evaluated on data set `la2` in Table 23.8. Here, the cluster number ranges from 2 to 15. As shown in Figure 23.6, the measurement values for all the measures will change as the cluster numbers increase. However, the normalized measures including VI_n, VD_n, and R_n can capture the same optimal cluster number 5. Similar results can also be observed for other normalized measures, such as F_n, FM_n, and Γ_n.

Property 23.2.1 (n-Invariance) *For a contingence matrix M and a positive integer* λ*, a measure O is n-invariant, if* $O(\lambda M) = O(M)$*, where n is the number of objects.*

A Summary of Math Properties. Five math properties of measures are listed as follows (see Table 23.12). Due to space limitation, the proofs are omitted here.

Property 23.2.2 (Symmetry) *A measure O is symmetric, if* $O(M^T) = O(M)$ *for any contingence matrix M.*



(a) (b) (c)

FIGURE 23.6: Impact of the number of clusters.

TABLE 23.12: Math Properties of Measures

	F_n	VI_n	VD_n	ε_n	R_n	FM_n	Γ_n
P1	No	Yes	Yes	Yes**	Yes	Yes	Yes
P2	Yes	Yes	Yes	Yes	No	No	No
P3	Yes*	Yes*	Yes*	Yes*	No	No	No
P4	No	Yes	Yes	No	No	No	No
P5	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Note: Yes* — Yes for the unnormalized measures.
Yes** — Yes for $K = K'$.

The *symmetry* property treats the predefined class structure as one of the partitions. Therefore, the task of cluster validation is the same as the comparison of partitions. This means transposing two partitions in the contingency matrix should not bring any difference to the measure value. This property is not true for F_n which is a typical measure in asymmetry. Also, ε_n is symmetric if and only if $K = K'$.

Intuitively, a mathematically sound validation measure should satisfy the *n-invariance* property. However, three measures, namely, R_n, FM_n, and Γ_n cannot fulfill this requirement. Nevertheless, they can still be treated as the asymptotically n-invariant measures, since they tend to be n-invariant with the increase of n.

Property 23.2.3 (Convex additivity) *Let $P = \{P_1, \cdots, P_K\}$ be a clustering, P' be a refinement of P[1], and P'_l be the partitioning induced by P' on P_l. Then a measure O is convex additive, if $O(M(P,P')) = \sum_{l=1}^{K} \frac{n_l}{n} O(M(I_{P_l}, P'_l))$, where n_l is the number of data points in P_l, I_{P_l} represents the partitioning on P_l into one cluster, and $M(X,Y)$ is the contingency matrix of X and Y.*

The *convex additivity* property was introduced by Meila [42]. It requires the measures to show additivity along the lattice of partitions. Unnormalized measures including F, VD, VI, and ε hold this property. However, none of the normalized measures studied in this chapter holds this property.

Property 23.2.4 (Left-domain-completeness) *A measure O is left-domain-complete, if for any contingence matrix M with statistically independent rows and columns,*

$$O(M) = \begin{cases} 0, & O \text{ is a positive measure} \\ 1, & O \text{ is a negative measure} \end{cases}$$

When the rows and columns in the contingency matrix are statistically independent, the poorest values of the measures are expected to be seen, i.e., 0 for positive measures and 1 for negative measures. Among all the measures, however, only VI_n and VD_n can meet this requirement.

Property 23.2.5 (Right-domain-completeness) *A measure O is right-domain-complete, if for any contingence matrix M with perfectly matched rows and columns,*

$$O(M) = \begin{cases} 1, & O \text{ is a positive measure} \\ 0, & O \text{ is a negative measure} \end{cases}$$

This property requires measures to show optimal values when the class structure matches the cluster structure perfectly. The above normalized measures hold this property.

[1] "P' be a refinement of P" means P' is the descendant node of node P in the lattice of partitions. See [42] for details.

23.2.4.3 Discussions

In a nutshell, among 16 external validation measures shown in Table 23.1, it is first known that Mirkin metric (M) is equivalent to Rand statistic (R), and micro-average precision (MAP) and Goodman–Kruskal coefficient (GK) are equivalent to the purity measure (P) by observing their computational forms. Therefore, the scope of the study reduces from 16 measures to 13 measures. In Section 23.2.2, analysis shows that P, mutual information (MI), and entropy (E) are defective measures for evaluating K-means clustering. Also, it is proved that variation of information (VI) is an improved version of MI and E, and van Dongen criterion (VD) is an improved version of P. As a result, the selection pool is further reduced to 10 measures.

In addition, as shown in Section 23.2.3, it is necessary to use the normalized measures for evaluating K-means clustering, since the normalized measures can capture the uniform effect by K-means and allow the evaluation of different clustering results on different data sets. Proposition 23.2.1 on page 579 shows that the normalized Rand statistic (R_n) is the same as the normalized Hubert Γ statistic II (Γ'_n). Also, the normalized Rand statistic is equivalent to J'_n, which is the same as MS'_n. Therefore, only R_n needs further consideration and J'_n, Γ'_n as well as MS'_n can be excluded. The results in Section 23.2.3 show that the normalized F-measure (F_n) and classification error (ε_n) cannot well capture the uniform effect by K-means. Also, these two measures do not satisfy some math properties in Table 23.12. As a result, they are excluded as well. Now, there are only five normalized measures left: VI_n, VD_n, R_n, FM_n, and Γ_n. Figure 23.5 shows that the validation performances of R_n, FM_n, and Γ_n are very similar to each other. Therefore, only R_n needs to be considered.

Based on the above study, it is most suitable to use the normalized van Dongen criterion (VD_n) in most of the general cases, since VD_n has a simple computation form, satisfies all mathematically sound properties as shown in Table 23.12, and can measure well on the data with imbalanced class distributions. However, for the case that the clustering performances are hard to distinguish, one may want to use the normalized variation of information (VI_n) instead,[2] since VI_n has high sensitivity on detecting the clustering changes. Finally, R_n can also be used as a complementary to the above two measures.

23.3 Internal Clustering Validation Measures

In the literature, a number of internal validation measures for crisp clustering have been proposed. In this section, an organized study on a suite of 12 widely used internal clustering validation measures as shown in Table 23.13 are provided for different clustering algorithms. These measures represent a good coverage of the internal validation measures available in different fields such as data mining, information retrieval, machine learning, and statistics. The properties of these validation measures are investigated in six different aspects: monotonicity, noise, density, subclusters, skewed distribution, and arbitrary shapes data. For each aspect, a synthetic data set which best represents the property is generated for studies. Results and discussions will be presented at the end of this section. First, some basic concepts of internal clustering validation measures, as well as the suite of 12 widely used internal clustering validation measures, are introduced.

23.3.1 An Overview of Internal Clustering Validation Measures

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 [57, 68, 33].

[2] Note that the normalized variation of information is equivalent to the normalized mutual information.

TABLE 23.13: Internal Clustering Validation Measures.

	Measure	Definition
1	$RMSSTD$[1]	$\{\sum_i \sum_{x\in C_i} \parallel x-c_i \parallel^2/[P\sum_i(n_i-1)]\}^{\frac{1}{2}}$
2	R-squared (RS)	$(\sum_{x\in D} \parallel x-c \parallel^2 - \sum_i \sum_{x\in C_i} \parallel x-c_i \parallel^2)/\sum_{x\in D} \parallel x-c \parallel^2$
3	Modified Hubert Γ statistic (Γ)	$\frac{2}{n(n-1)} \sum_{x\in D} \sum_{y\in D} d(x,y) d_{x\in C_i, y\in C_j}(c_i,c_j)$
4	Calinski-Harabasz index (CH)	$\frac{\sum_i n_i d^2(c_i,c)/(NC-1)}{\sum_i \sum_{x\in C_i} d^2(x,c_i)/(n-NC)}$
5	I index (I)	$(\frac{1}{NC} \cdot \frac{\sum_{x\in D} d(x,c)}{\sum_i \sum_{x\in C_i} d(x,c_i)} \cdot \max_{i,j} d(c_i,c_j))^p$
6	Dunn's indices (D)	$\min_i\{\min_j(\frac{\min_{x\in C_i, y\in C_j} d(x,y)}{\max_k\{\max_{x,y\in C_k} d(x,y)\}})\}$
7	Silhouette index (S)	$\frac{1}{NC}\sum_i\{\frac{1}{n_i}\sum_{x\in C_i}\frac{b(x)-a(x)}{\max[b(x),a(x)]}\}$ $a(x)=\frac{1}{n_i-1}\sum_{y\in C_i, y\neq x} d(x,y), b(x)=\min_{j,j\neq i}[\frac{1}{n_j}\sum_{y\in C_j} d(x,y)]$
8	Davies-Bouldin index (DB)	$\frac{1}{NC}\sum_i \max_{j,j\neq i}\{[\frac{1}{n_i}\sum_{x\in C_i} d(x,c_i)+\frac{1}{n_j}\sum_{x\in C_j} d(x,c_j)]/d(c_i,c_j)\}$
9	Xie-Beni index (XB)	$[\sum_i \sum_{x\in C_i} d^2(x,c_i)]/[n\cdot min_{i,j\neq i} d^2(c_i,c_j)]$
10	SD validity index (SD)	$Dis(NC_{max})Scat(NC)+Dis(NC)$ $Scat(NC)=\frac{1}{NC}\sum_i \parallel \sigma(C_i) \parallel / \parallel \sigma(D) \parallel$ $Dis(NC)=\frac{max_{i,j} d(c_i,c_j)}{min_{i,j} d(c_i,c_j)}\sum_i(\sum_j d(c_i,c_j))^{-1}$
11	S_Dbw validity index (S_Dbw)	$Scat(NC)+Dens\_bw(NC)$ $Dens\_bw(NC)=\frac{1}{NC(NC-1)}\sum_i[\sum_{j,j\neq i}\frac{\sum_{x\in C_i\cup C_j} f(x,u_{ij})}{max\{\sum_{x\in C_i} f(x,c_i),\sum_{x\in C_j} f(x,c_j)\}}]$
12	$CVNN$[2] index	$Sep(NC,k)/\max_{NC} Sep(NC,k)+Com(NC)/\max_{NC} Com(NC)$ $Com(NC)=\sum_i[\frac{2}{n_i\cdot(n_i-1)}\sum_{x,y\in C_i} d(x,y)]$ $Sep(NC,k)=\max_i(\frac{1}{n_i}\sum_{j=1,2,\ldots,n_i}\frac{q_j}{k})$

Note: 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 ith cluster; n_i: number of objects in C_i; c_i: center of C_i;
k: number of nearest neighbors; q_j: number of C_i's jth object's nearest neighbors which are not in cluster C_i;
$\sigma(C_i)$: variance vector of C_i; $d(x,y)$: distance between x and y; $\parallel X_i \parallel = (X_i^T \cdot X_i)^{1/2}$.
[1] $RMSSTD$: Root-mean-square standard deviation.
[2] $CVNN$: Clustering Validation index based on Nearest Neighbors.

I. Compactness. This measures how closely related the objects in a cluster are. A group of measures evaluates cluster compactness based on variance. Lower variance indicates better compactness. In addition, numerous measures estimate the cluster compactness based on distance, such as maximum or average pairwise distance, and maximum or average center-based distance.

II. Separation. This measures how distinct or well-separated a cluster is from other clusters. For example, the pairwise distances between cluster centers and 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 which was obtained in Step 2.

Step 4: Choose the best partition and the optimal cluster number according to the criteria.

Table 23.13 lists the measures to be studied in this section. The "Definition" column gives the computation forms of the measures. While most indices, such as *DB*, *XB*, and *S_Dbw*, consider both of the evaluation criteria (compactness and separation) in the way of ratio or summation some, such as *RMSSTD*, *RS*, and Γ consider only one aspect. The 12 measures are briefly introduced as follows.

The root-mean-square standard deviation (*RMSSTD*) is the square root of the pooled sample variance of all the attributes [51]. It measures the homogeneity of the formed clusters. R-squared (*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 [51, 19]. The Modified Hubert Γ statistic (Γ) [25] evaluates the difference between clusters by counting the disagreements of pairs of data objects in two partitions.

The Calinski–Harabasz index (*CH*) [7] evaluates the cluster validity based on the average between- and within-cluster sum of squares. Index *I* (*I*) [41] measures separation based on the maximum distance between cluster centers, and measures compactness based on the sum of distances between objects and their cluster center. Dunn's index (*D*) [11] 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 $Index = (a \cdot Separation)/(b \cdot Compactness)$, where *a* and *b* are weights. The optimal cluster number is determined by maximizing the value of these indices.

The Silhouette index (*S*) [49] 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 Davies–Bouldin index (*DB*) [9] 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, achieve the best partition. The Xie-Beni index (*XB*) [63] 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 [32] proposed indices DB^{**} and XB^{**} in 2005 as the improvements of *DB* and *XB*. The two improved measures will be used in this study.

The idea of SD index (*SD*) [22] 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 *SD* index is the summation of these two terms, and the optimal number of clusters can be obtained by minimizing the value of *SD*.

The S_Dbw index (*S_Dbw*) [20] 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.

Different from the existing measures, the Clustering Validation index based on Nearest Neighbors (*CVNN*) [38] evaluates the intercluster separation based on objects that carry the geometrical information of each cluster. Sharing the same idea with kNN consistency, *CVNN* uses dynamic multiple objects as representatives for different clusters in different situations when measuring the intercluster separation. 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. *CVNN* also employs the average pairwise distance between objects in the same cluster as the measurement of intracluster compactness. Finally, the *CVNN* index takes a form of the summation of the intercluster separation and the intracluster compactness after the normalization for both of them.

There are some other internal validation measures in the literature [50, 21, 58, 35]. However, some have poor performance while some are designed for data sets with specific structures. Take Composed Density between and within clusters index (*CDbw*) and Symmetry distance-based index (*Sym–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, *Sym–index* can handle only data sets which are internally symmetrical. A focused study on the above mentioned 12 internal validation measures will be presented in the following sections, and acronyms for these measures will be used.

23.3.2 Understanding of Internal Clustering Validation Measures

In this section, a study of the 12 internal validation measures mentioned in Section 23.3.1 is presented to investigate the validation properties of different internal validation measures in different aspects, which can be helpful for the index selection. If not mentioned, K-means is used [39] (implemented by CLUTO) [27] as the clustering algorithm, and the parameter k is set to be 10 for *CVNN*.

23.3.2.1 The Impact of Monotonicity

The monotonicity of different internal validation indices is studied in this subsection. K-means algorithm is applied on the data set *Wellseparated* to get the clustering results for different numbers of clusters. As shown in Figure 23.7, *Wellseparated* is a synthetic data set composed of 1000 data objects, which are well separated into five clusters.

As the results shown in Table 23.14, the first three indices monotonically increase or decrease as the cluster number *NC* increases. On the other hand, the remaining nine indices reach their maximum or minimum value as *NC* equals the true cluster number. There are certain reasons for the monotonicity of the first three indices.

$RMSSTD = \sqrt{SSE/P(n-NC)}$, and *SSE* (Sum of Square Error) decreases as *NC* increases. In practice $NC \ll n$; thus, $n - NC$ can be viewed as a constant number. Therefore, *RMSSTD* decreases

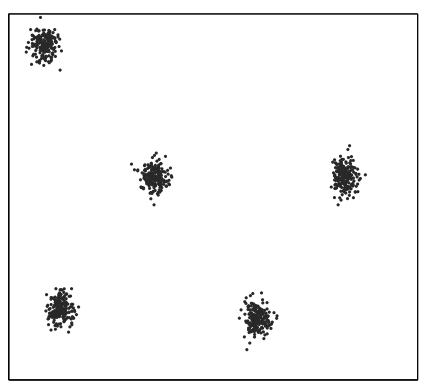


FIGURE 23.7: The data set *Wellseparated*.

TABLE 23.14: Results of the Impact of Monotonicity, True $NC = 5$

	RMSSTD	*RS*	Γ	*CH*	*I*	*D*	*S*	*DB***	*SD*	*S_Dbw*	*XB***	*CVNN*
2	28.50	0.63	2973	1683	3384	0.49	0.61	0.72	0.22	61.84	0.27	1.00
3	20.80	0.80	3678	2016	5759	0.55	0.71	0.68	0.12	0.15	0.37	0.64
4	14.83	0.90	4007	2968	11230	0.58	0.83	0.52	0.08	0.06	0.50	0.38
5	**3.20**	**0.99**	**4342**	**52863**	**106163**	**2.23**	**0.91**	**0.12**	**0.05**	**0.004**	**0.25**	**0.12**
6	3.08	1.00	4343	45641	82239	0.03	0.72	0.52	0.50	0.07	35.10	0.74
7	2.96	1.00	4344	41291	68894	0.02	0.58	0.80	0.49	0.10	35.10	1.05
8	2.83	1.00	4346	38580	58420	0.01	0.48	1.02	0.54	0.08	36.51	1.11
9	2.72	1.00	4347	36788	50259	0.01	0.39	1.17	0.55	0.11	38.01	1.10

as *NC* increases. Also $RS = (TSS - SSE)/TSS$ (TSS-Total Sum of Squares), and $TSS = SSE + SSB$ (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 actually counted. 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, one can discover that they only take either separation or compactness into account. (*RS* and Γ consider only separation, and *RMSSTD* considers only 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" [19]. However, since the judgment of the shift point is very subjective and hard to determine, these three measures are excluded from future studies. The focus will be on the remaining 9 measures.

23.3.2.2 The Impact of Noise

The following study on the data set *Wellseparated.noise* evaluates the influence of noise on internal validation indices. As shown in Figure 23.8, *Wellseparated.noise* is a synthetic data set formulated by adding 5% noise to the data set *Wellseparated*. The cluster numbers selected by indices are shown in Table 23.15. Results show that *D* and *CH* choose the wrong cluster number. 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

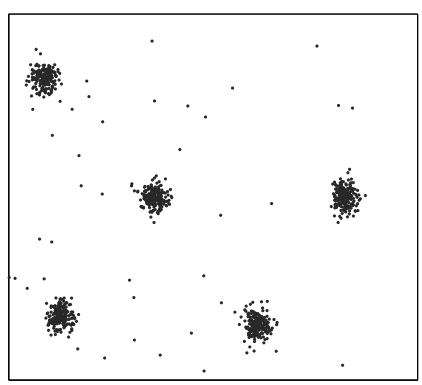


FIGURE 23.8: The data set *Wellseparated.noise*.

TABLE 23.15: Results of the Impact of Noise, True $NC = 5$

	CH	*I*	*D*	*S*	*DB***	*SD*	*S_Dbw*	*XB***	*CVNN*
2	1626	3213	0.0493	0.590	0.739	0.069	20.368	0.264	1.01
3	1846	5073	0.0574	0.670	0.721	0.061	0.523	0.380	0.69
4	2554	9005	**0.0844**	0.783	0.560	0.050	0.087	0.444	0.44
5	10174	**51530**	0.0532	**0.870**	**0.183**	**0.045**	**0.025**	**0.251**	**0.19**
6	**14677**	48682	0.0774	0.802	0.508	0.046	0.044	0.445	0.51
7	12429	37568	0.0682	0.653	0.710	0.055	0.070	0.647	0.92
8	11593	29693	0.0692	0.626	0.863	0.109	0.052	2.404	1.15
9	11088	25191	0.0788	0.596	0.993	0.121	0.056	3.706	0.98

$(\max_k\{\max_{x,y\in C_k} d(x,y)\})$ as the intracluster compactness. And 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 uses only 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 $CH = (SSB/SSE)\cdot((n-NC)/(NC-1))$ and $((n-NC)/(NC-1))$ is constant for the same NC, only (SSB/SSE) needs to be considered. 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.

Moreover, the indices other than CH and D will also be influenced by noise in a less sensitive way. Comparing Table 23.15 with Table 23.14, it is clear that the values of other indices change to some degree. If adding 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.

23.3.2.3 The Impact of Density

A data set with various densities is challenging for many clustering algorithms. Therefore, it is a very interesting topic whether data with different densities also affect the performance of the internal validation measures. A study is conducted on a synthetic data set with different density

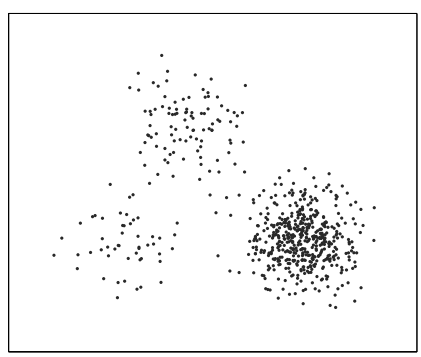


FIGURE 23.9: The data set *Differentdensity*.

TABLE 23.16: Results of the Impact of Density, True $NC = 3$

	CH	*I*	*D*	*S*	*DB***	*SD*	*S_Dbw*	*XB***	*CVNN*
2	1172	**120.1**	0.0493	0.587	0.658	0.705	0.603	0.408	1.03
3	**1197**	104.3	**0.0764**	**0.646**	**0.498**	**0.371**	**0.275**	**0.313**	**0.84**
4	1122	93.5	0.0048	0.463	1.001	0.672	0.401	3.188	0.92
5	932	78.6	0.0049	0.372	1.186	0.692	0.367	3.078	1.22
6	811	59.9	0.0049	0.312	1.457	0.952	0.312	6.192	1.32
7	734	56.1	0.0026	0.278	1.688	1.192	0.298	9.082	1.28
8	657	44.8	0.0026	0.244	1.654	1.103	0.291	8.897	1.26
9	591	45.5	0.0026	0.236	1.696	1.142	0.287	8.897	1.59

named *Differentdensity*. *Differentdensity* has 650 data objects and the details are shown in Figure 23.9. The results listed in Table 23.16 show that only *I* selects the wrong optimal cluster number.

The reason *I* does not choose the right cluster number is not easy to explain. One can observe that *I* keeps decreasing as cluster number *NC* increases. One possible reason is the uniform effect of the *K*-means algorithm, which tends to divide objects into relatively equal sizes [64]. *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 remain almost 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.

23.3.2.4 The Impact of Subclusters

Subclusters are clusters that are close to each other. Figure 23.10 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.

Results presented in Table 23.17 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*, *S_Dbw*, and *CVNN* have the correct ones. Intercluster separation is supposed to have a sharp decrease when cluster number changes from $NC_{optimal}$ to $NC_{optimal+1}$ [32]. However, for *D*, *S*, *DB***, *SD*, and *XB***, sharper decreases can be observed at $NC < NC_{optimal}$. The reasons are as follows.

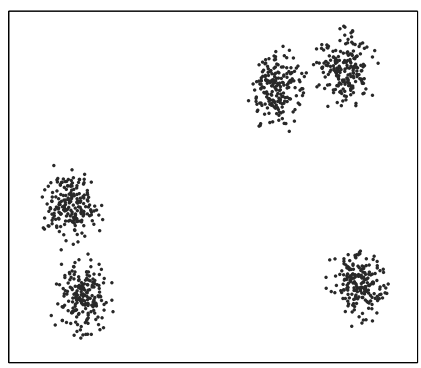


FIGURE 23.10: The data set *Subcluster*.

TABLE 23.17: Results of the Impact of Subclusters, True $NC = 5$

	CH	*I*	*D*	*S*	*DB***	*SD*	*S_Dbw*	*XB***	*CVNN*
2	3474	2616	0.7410	0.736	0.445	0.156	0.207	0.378	1.00
3	7851	5008	**0.7864**	**0.803**	**0.353**	**0.096**	0.056	**0.264**	0.54
4	8670	5594	0.0818	0.737	0.540	0.164	0.039	1.420	0.47
5	**16630**	**9242**	0.0243	0.709	0.414	0.165	**0.026**	1.215	**0.43**
6	14310	7021	0.0243	0.587	0.723	0.522	0.063	12.538	0.79
7	12900	5745	0.0167	0.490	0.953	0.526	0.101	12.978	1.26
8	11948	4803	0.0167	0.402	1.159	0.535	0.105	14.037	1.25
9	11354	4248	0.0107	0.350	1.301	0.545	0.108	14.858	1.06

S uses the average minimum distance between clusters as the intercluster separation. For a 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 a data set with subclusters, the measure of separation will achieve its maximum value when subclusters close 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^{**}, which will not be elaborated here due to the limit of space.

23.3.2.5 The Impact of Skewed Distributions

It is common that clusters in a data set have unequal sizes. Figure 23.11 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 of tending to divide objects into relatively equal sizes, it does not have a good performance when dealing with skewed distributed data sets [65]. In order to demonstrate this statement, four widely used algorithms are employed from four different categories: *K*-means (prototype-based), DBSCAN (density-based) [12], Agglo (based on average-link, hierarchical) [26], and Chameleon (graph-based) [29]. Each of them are applied on *Skewdistribution* to divide the data set into three clusters, which is the true cluster number. As shown in Figure 23.12, *K*-means performs the worst while Chameleon is the best.

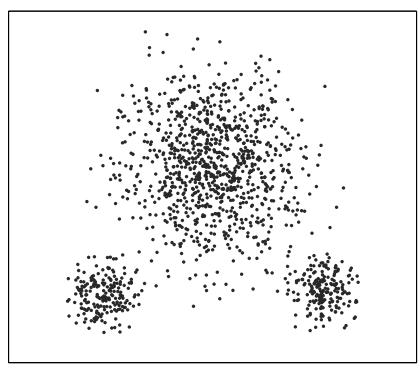
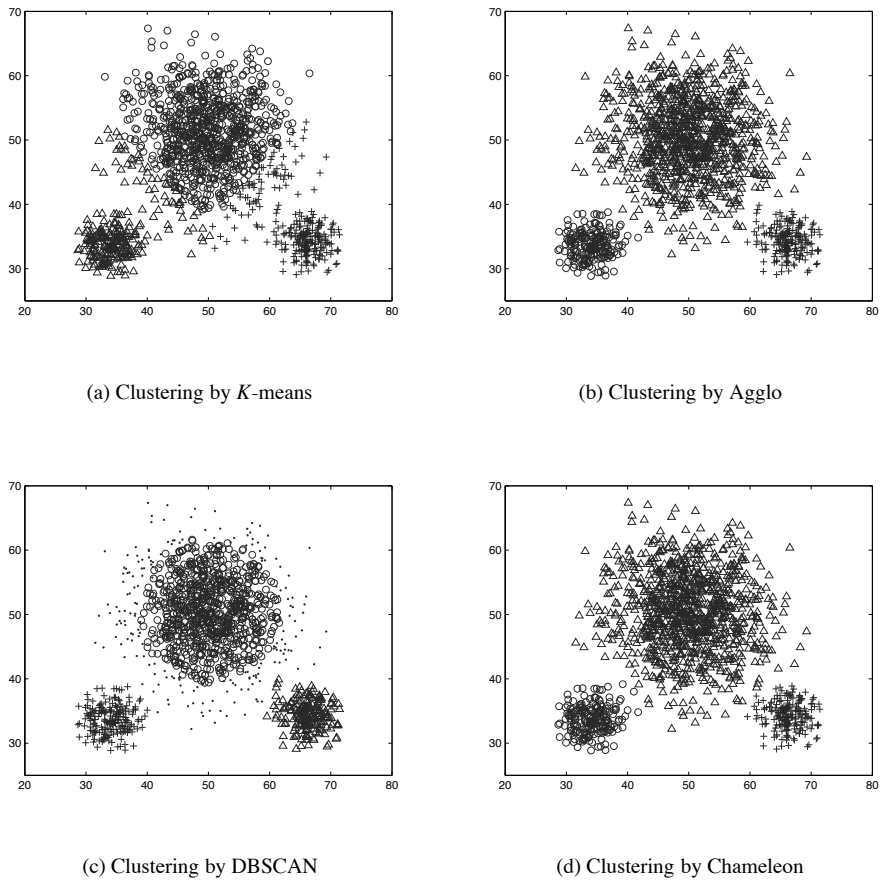


FIGURE 23.11: The data set *Skewdistribution*.



(a) Clustering by *K*-means

(b) Clustering by Agglo

(c) Clustering by DBSCAN

(d) Clustering by Chameleon

FIGURE 23.12: Clustering results on data set *Skewdistribution* by different algorithms where $NC = 3$.

TABLE 23.18: Results of the Impact of Skewed Distributions, True $NC = 3$

	CH	*I*	*D*	*S*	DB^{**}	*SD*	*S_Dbw*	XB^{**}	*CVNN*
2	788	232.3	0.0286	0.621	0.571	0.327	0.651	0.369	1.04
3	1590	**417.9**	**0.0342**	**0.691**	**0.466**	**0.187**	**0.309**	**0.264**	**0.73**
4	1714	334.5	0.0055	0.538	0.844	0.294	0.379	1.102	0.85
5	**1905**	282.9	0.0069	0.486	0.807	0.274	0.445	0.865	0.87
6	1886	226.7	0.0075	0.457	0.851	0.308	0.547	1.305	0.96
7	1680	187.1	0.0071	0.371	1.181	0.478	0.378	3.249	1.10
8	1745	172.9	0.0075	0.370	1.212	0.474	0.409	3.463	1.03
9	1317	125.5	0.0061	0.301	1.875	0.681	0.398	7.716	1.41

A study was conducted on the data set *Skewdistribution* to evaluate the performance of different indices on a data set with skewed distributions. Chameleon is applied as the clustering algorithm. Results listed in Table 23.18 show that only *CH* cannot give the right optimal cluster number. $CH = (TSS/SSE - 1) \cdot ((n - NC)/(NC - 1))$ and TSS is a constant number of a certain data set. Thus, *CH*

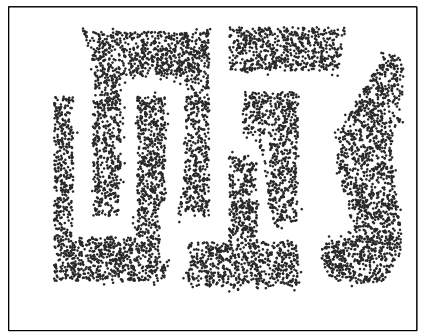


FIGURE 23.13: The data set *T4.8k.modified*.

is essentially based on *SSE*, which shares the same basis with *K*-means algorithm. As mentioned above, *K*-means cannot handle skewed distributed data sets. Therefore, the similar conclusion can be applied to *CH*.

23.3.2.6 The Impact of Arbitrary Shapes

A data set with arbitrary shapes is always hard to handle. Figure 23.13 shows a synthetic data set *T4.8k.modified* which consists of six irregular shape of clusters. It is generated by removing 10% noise from the original data set *T4.8k* which contains 8000 objects[28]. As in the last subsection, the same four algorithms are employed to run on *T4.8k.modified* to divide the data set into six clusters, which is the true cluster number. As shown in Figure 23.14, Chameleon performs the best among these four clustering algorithms.

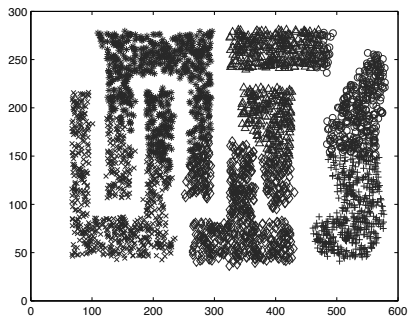
A study on the data set *T4.8k.modified* was performed to evaluate whether the nine internal validation indices could handle a data set with arbitrary shapes. Chameleon is applied as the clustering algorithm. Results listed in Table 23.19 show that only *CVNN* can deal with data set with arbitrary structures. The reasons are as follows.

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 Figure 23.15. The minimum pairwise distance between these two clusters is almost zero while they are still separable.

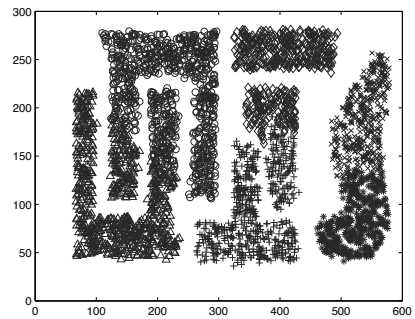
For *CH*, *I*, DB^{**}, *SD*, *S_Dbw*, and XB^{**}, these six indices use the cluster center of each cluster as

TABLE 23.19: Results of the Impact of Arbitrary Shapes, True $NC = 6$

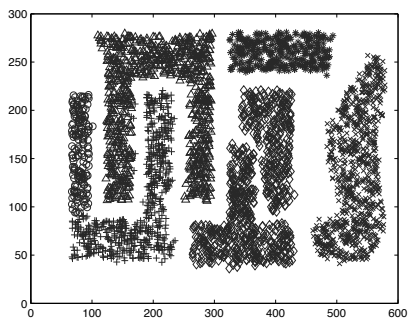
	CH	*I*	*D*	*S*	DB^{**}	*SD*	*S_Dbw*	XB^{**}	*CVNN*
2	301	1808	0.0110	0.231	2.927	0.0442	1.790	7.824	1.03
3	5484	32080	0.0117	0.401	0.984	0.0219	0.579	1.271	0.75
4	**8213**	**34532**	**0.0183**	**0.438**	**0.769**	**0.0197**	0.680	**1.143**	0.65
5	6838	24902	0.0142	0.384	0.828	0.0299	0.509	3.032	0.62
6	7560	24721	0.0074	0.333	1.038	0.0286	∞	2.685	**0.58**
7	7151	20753	0.0080	0.343	0.984	0.0290	0.426	2.674	0.89
8	6445	16922	0.0072	0.367	0.896	0.0293	0.416	2.892	1.39
9	6636	22365	0.0067	0.376	0.865	0.0312	**0.281**	2.755	1.36



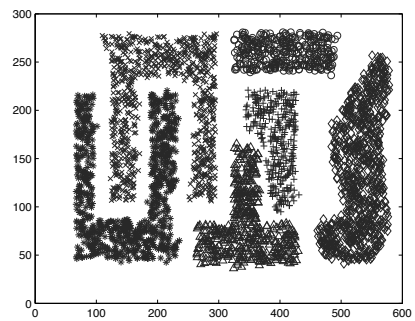
(a) Clustering by *K*-means



(b) Clustering by Agglo



(c) Clustering by DBSCAN



(d) Clustering by Chameleon

FIGURE 23.14: Clustering results on data set *T4.8k.modified* by different algorithms where $NC = 6$.

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 to the minimum pairwise distance between cluster centers in a sense. Since it is meaningful to use the center to represent the entire cluster only for the sphere-shaped cluster, this implies that these indices can work only in the hypersphere condition. Figure 23.15 gives 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 one moves 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.

The above 8 measures use either the average (minimum) pairwise distance between objects in different clusters or one single object (the cluster center) as the representative of the entire cluster when calculating the intercluster separation. These measures consider only the positions of the objects in clusters and fail to take into account the object distributions which form the geometrical information of the cluster. On the other hand, $CVNN$ evaluates the intercluster separation based on objects that carry the geometrical information of each cluster. Sharing the same idea with kNN consistency, $CVNN$ uses dynamic multiple objects as representatives for different clusters in different

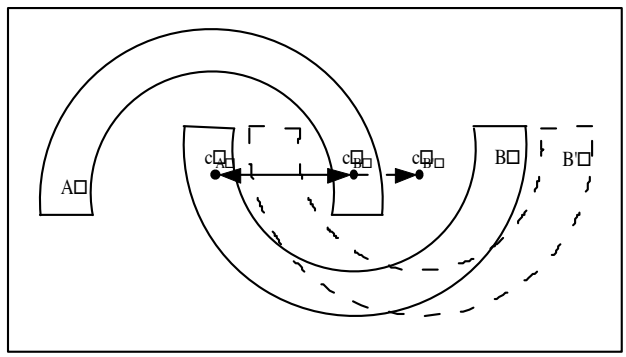
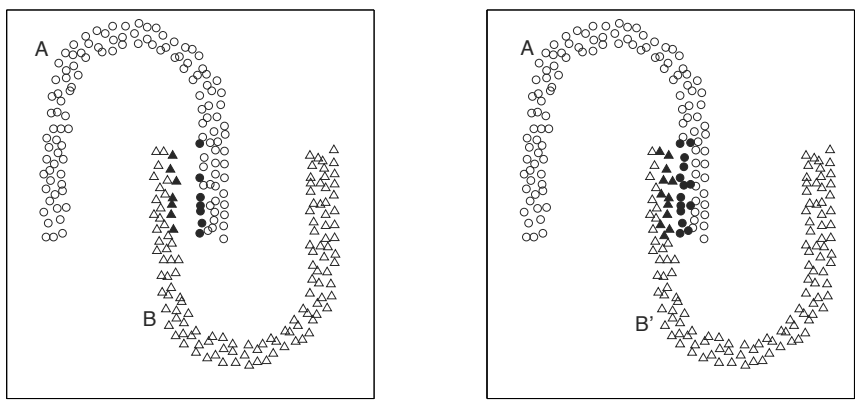


FIGURE 23.15: Arcuate shape intercluster separation.



(a) Before movement (b) After movement

FIGURE 23.16: An illustration of the dynamic effect of cluster representatives of *CVNN*.

situations when measuring the intercluster separation. Figure 23.16 illustrates the dynamic effect of how representatives of clusters evolve in different situations. In this example, both clusters A and B have an arcuate structure and solid objects are the representatives selected by the *CVNN* measure. Comparing subfigure (a) with (b), one 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 the intercluster separation measure *Sep* within *CVNN*, which agree with the indication that clusters are becoming more separated. This example illustrates the dynamic effect of *CVNN*'s intercluster separation measure, since the representatives for the same two clusters in different situations are different.

23.3.3 Properties of Measures

Table 23.20 summarizes the properties of different internal validation measures in different aspects, which can be helpful for the index selection. "–" indicates property not tested, and "×"

TABLE 23.20: Overall Performance of Different Measures

Measure	Monotonicity	Noise	Density	Subcluster	Skew Distr.	Arbit. Shape
RMSSTD	×	–	–	–	–	–
RS	×	–	–	–	–	–
Γ	×	–	–	–	–	–
CH		×			×	×
I			×			×
D		×		×		×
S				×		×
*DB***				×		×
SD				×		×
S_Dbw						×
*XB***				×		×
CVNN						

denotes situation cannot be handled. It suggests that, while the other 11 measures have certain limitations in different scenarios, especially in the aspect of handling data set with arbitrary structures, *CVNN* performs well in all six aspects. *CVNN* exploits the notion of nearest neighbors and uses dynamic multiple objects as representatives for different clusters in different situations, which makes it particularly useful when the data set includes clusters with arbitrary shapes. Thus, *CVNN* can play an important role as a valuable complementary measure in the suite of internal clustering validation measures.

23.4 Summary

This chapter presents detailed studies on 16 external validation measures and 12 internal validation measures in a comprehensive way on various aspects of these validation measure. For the external validation part, different measures are compared and contrasted for K-means clustering. As results revealed, it is necessary to normalize validation measures before they can be employed for clustering validation, since unnormalized measures may lead to inconsistent or even misleading results. This is particularly true for data with imbalanced class distributions. Normalization solutions are also provided for some validation measures. Furthermore, lemmas are provided to show that some validation measures are mathematically equivalent and some measures have very similar validation performances. Finally, key properties of the 16 measures are summarized. These properties should be considered before deciding what is the right measure to use in practice.

For the internal validation part, the validation properties of a suite of 12 existing internal clustering validation measures for crisp clustering are studied in six different aspects: monotonicity, noise, density, subclusters, skewed distribution, and arbitrary shapes data. Six synthetic data sets which best represent the above six aspects are used to evaluate the performance of the 12 validation measures. The results of the studies demonstrate that all measures except for the *CVNN* index have certain limitations in different application scenarios, especially showing difficulties in handling data set with arbitrary structures. On the other hand, *CVNN* exploits the notion of nearest neighbors and uses dynamic multiple objects as representatives for different clusters in different situations, which makes it particularly useful when the data set includes clusters with arbitrary shapes. The summarized validation properties of the 12 internal validation measures may serve as a guide for index selection in practice.

Bibliography

[1] I. Assent, R. Krieger, E. Muller, and T. Seidl. DUSC: Dimensionality unbiased subspace clustering. In *Seventh IEEE International Conference on Data Mining, 2007, ICDM 2007*, pages 409–414. IEEE, 2007.

[2] I. Assent, R. Krieger, E. Muller, and T. Seidl. InSCY: Indexing subspace clusters with in-process-removal of redundancy. In *Eighth IEEE International Conference on Data Mining, 2008. ICDM'08*, pages 719–724. IEEE, 2008.

[3] A. Ben-Hur and I. Guyon. Detecting stable clusters using principal component analysis. In M.J. Brownstein and A.Kohodursky (Eds.) *Methods in Molecular Biology*, pages 159–182, Humana Press, 2003.

[4] J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, 1981.

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

[6] Marcel Brun, Chao Sima, Jianping Hua, James Lowey, Brent Carroll, Edward Suh, and Edward R. Dougherty. Model-based evaluation of clustering validation measures. *Pattern Recognition*, 40:807–824, March 2007.

[7] T. Calinski and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27, 1974.

[8] T.M. Cover and J.A. Thomas. *Elements of Information Theory* (2nd Edition). Wiley-Interscience, 2006.

[9] D.L. Davies and D.W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):224–227, 1979.

[10] M. DeGroot and M. Schervish. *Probability and Statistics* (3rd Edition). Addison Wesley, 2001.

[11] J.C. Dunn. Well separated clusters and optimal fuzzy partitions. *Cybernetics and Systems*, 4(1):95–104, 1974.

[12] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231, 1996.

[13] E.B. Fowlkes and C.L. Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78:553–569, 1983.

[14] I. Gath and A.B. Geva. Unsupervised optimal fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):773–780, 1989.

[15] L.A. Goodman and W.H. Kruskal. Measures of association for cross classification. *Journal of the American Statistical Association*, 49:732–764, 1954.

[16] Naiyang Guan, Dacheng Tao, Zhigang Luo, and Bo Yuan. NeNMF: An optimal gradient method for nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 60:2882–2898, 2012.

[17] Naiyang Guan, Dacheng Tao, Zhigang Luo, and Bo Yuan. Online non-negative matrix factorization with robust stochastic approximation. *IEEE Transactions on Neural Networks and Learning Systems*, 23:1087–1099, 2012.

[18] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. Cluster validity methods: Part I. *SIGMOD Record*, 31(2):40–45, 2002.

[19] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2–3):107–145, 2001.

[20] Maria Halkidi and Michalis Vazirgiannis. Clustering validity assessment: Finding the optimal partitioning of a data set. In *Proceedings of the 2001 IEEE International Conference on Data Mining*, pages 187–194, 2001.

[21] Maria Halkidi and Michalis Vazirgiannis. Clustering validity assessment using multi-representatives. In *Proceedings of the SETN*, pages 237–248, 2002.

[22] Maria Halkidi, Michalis Vazirgiannis, and Yannis Batistakis. Quality scheme assessment in the clustering process. In *PKDD '00*, pages 265–276, London, UK, 2000. Springer-Verlag.

[23] L. Hubert. Nominal scale response agreement as a generalized correlation. *British Journal of Mathematical and Statistical Psychology*, 30:98–103, 1977.

[24] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[25] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

[26] Anil K. Jain and Richard C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[27] George Karypis. *Cluto—Software for clustering high-dimentional datasets*. version 2.1.2, 2006.

[28] George Karypis. *Karypis Lab*. http://glaros.dtc.umn.edu/gkhome/.

[29] George Karypis, Eui-Hong (Sam) Han, and Vipin Kumar. Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32(8):68–75, 1999.

[30] M.G. Kendall. *Rank Correlation Methods*. New York: Hafner Publishing Co., 1955.

[31] D.W. Kim, K.H. Lee, and D. Lee. Fuzzy cluster validation index based on inter-cluster proximity. *Pattern Recognition Letters*, 24(15):2561–2574, 2003.

[32] Minho Kim and R. S. Ramakrishna. New indices for cluster validity assessment. *Pattern Recognition Letters*, 26(15):2353–2363, 2005.

[33] Johann M. Kraus, Christoph Mssel, Gnther Palm, and Hans A. Kestler. Multi-objective selection for collecting cluster alternatives. *Computational Statistics*, 26:341–353, 2011.

[34] Hardy Kremer, Philipp Kranen, Timm Jansen, Thomas Seidl, Albert Bifet, Geoff Holmes, and Bernhard Pfahringer. An effective evaluation measure for clustering on evolving data streams. In *ACM SIGKDD*, pages 868–876, 2011.

[35] Benson S. Y. Lam and Hong Yan. A new cluster validity index for data with merged clusters and different densities. In *IEEE ICSMC*, pages 798–803, 2005.

[36] B. Larsen and C. Aone. Fast and effective text mining using linear-time document clustering. In *KDD*, pages 16–22, 1999.

[37] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, and Junjie Wu. Understanding of internal clustering validation measures. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, pages 911–916, 2010.

[38] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, Junjie Wu, and Sen Wu. Understanding and enhancement of internal clustering validation measures. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, pages 982–994, 2013.

[39] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of BSMSP*, pages 281–297. University of California Press, 1967.

[40] MathWorks. *K*-means clustering in statistics toolbox.

[41] Ujjwal Maulik and Sanghamitra Bandyopadhyay. Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:1650–1654, 2002.

[42] M. Meila. Comparing clusterings—An axiomatic view. In *ICML*, pages 577–584, 2005.

[43] B. Mirkin. *Mathematical Classification and Clustering*. Kluwer Academic Press, 1996.

[44] E. Müller, S. Günnemann, I. Assent, and T. Seidl. Evaluating clustering in subspace projections of high dimensional data. *Proceedings of the VLDB Endowment*, 2(1):1270–1281, 2009.

[45] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: A review. *ACM SIGKDD Explorations Newsletter*, 6(1):90–105, 2004.

[46] A. Patrikainen and M. Meila. Comparing subspace clusterings. *IEEE Transactions on Knowledge and Data Engineering*, 18(7):902–916, 2006.

[47] W.M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66:846–850, 1971.

[48] C.J.V. Rijsbergen. *Information Retrieval* (2nd Edition). Butterworths, London, 1979.

[49] Peter Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computation and Applied Mathematics*, 20(1):53–65, 1987.

[50] Sriparna Saha and Sanghamitra Bandyopadhyay. Application of a new symmetry-based cluster validity index for satellite image segmentation. *IEEE Geoscience and Remote Sensing Letters*, pages 166–170, 2002.

[51] Subhash Sharma. *Applied Multivariate Techniques*. John Wiley & Sons, Inc., New York, USA, 1996.

[52] P. Smyth. Model selection for probabilistic clustering using cross-validated likelihood. *Statistics and Computing*, 10(1):63–72, 2000.

[53] Mingzhou (Joe) Song and Lin Zhang. Comparison of cluster representations from partial second- to full fourth-order cross moments for data stream clustering. In *IEEE ICDM*, pages 560–569, 2008.

[54] Michael Steinbach, George Karypis, and Vipin Kumar. A comparison of document clustering techniques. In *Workshop on Text Mining, KDD*, 2000.

[55] A. Strehl, J. Ghosh, and R.J. Mooney. Impact of similarity measures on web-page clustering. In *Workshop on Artificial Intelligence for Web Search, AAAI*, pages 58–64, 2000.

[56] Y. Sun, Y. Yu, and J. Han. Ranking-based clustering of heterogeneous information networks with star network schema. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806. ACM, 2009.

[57] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*, (First Edition). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

[58] Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B*, 63(2):411–423, 2001.

[59] TREC. Text retrieval conference. October 2007.

[60] S. van Dongen. Performance criteria for graph clustering and Markov cluster experiments. *TRINS= R0012*, Centrum voor Wiskunde en Informatica. 2000.

[61] J. Wu, H. Xiong, J. Chen, and W. Zhou. A generalization of proximity functions for *k*-means. In *ICDM*, pages 361–370, 2007.

[62] Junjie Wu, Hui Xiong, and Jian Chen. Adapting the right measures for *k*-means clustering. In *ACM SIGKDD*, pages 877–886, 2009.

[63] Xuanli Lisa Xie and Gerardo Beni. A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):841–847, 1991.

[64] H. Xiong, J. Wu, and J. Chen. *K*-means clustering versus validation measures: A data distribution perspective. In *KDD*, pages 318–331, 2006.

[65] Hui Xiong, Junjie Wu, and Jian Chen. *K*-means clustering versus validation measures: A data distribution perspective. *IEEE Transactions on Systems, Man, and Cybernetics Part B*, 39(2):318–331, 2009.

[66] X. Xu, N. Yuruk, Z. Feng, and T.A.J. Schweiger. SCAN: A structural clustering algorithm for networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 824–833. ACM, 2007.

[67] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis. *Machine Learning*, 55(3):311–331, 2004.

[68] Ying Zhao and George Karypis. Evaluation of hierarchical clustering algorithms for document datasets. In *CIKM '02*, pages 515–524, New York, USA, 2002. ACM.

[69] Tianyi Zhou, Dacheng Tao, and Xindong Wu. Manifold elastic net: A unified framework for sparse dimension reduction. *Data Mining and Knowledge Discovery*, 20:340–371, 2010.

[70] Y. Zhou, H. Cheng, and J.X. Yu. Graph clustering based on structural/attribute similarities. *Proceedings of the VLDB Endowment*, 2(1):718–729, 2009.