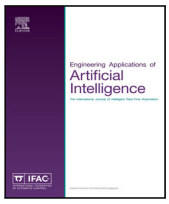
Engineering Applications of Artificial Intelligence 160 (2025) 111861

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Research paper

Statistical significance of cluster membership for categorical data

Lianyu Hu [a], Zerun Li [b], Junjie Dong [b], Mudi Jiang [b], Zengyou He [b,*]

[a] College of Information Science and Engineering, Henan University of Technology, Zhengzhou, China
[b] School of Software, Dalian University of Technology, Dalian, China

ARTICLE INFO

Keywords:
Categorical data
Cluster membership
Fisher's exact test
Meta-analysis
Statistical interpretation
Cluster validation

ABSTRACT

Clustering algorithms partition data samples into distinct groups in an unsupervised manner, which requires subsequent validation. In post-hoc cluster analysis, clustering quality is typically evaluated at the cluster level, with a focus on metrics such as intra- and inter-cluster distances. However, evaluation at the sample level (i.e., cluster membership) is often overlooked. To assess whether a given sample is correctly assigned to its respective cluster, it is crucial to consider the inevitable effects of random assignments or noisy samples. Unfortunately, the statistical cluster membership evaluation is a largely underexplored problem, with almost no previous research efforts in this direction. In this paper, we propose a new method for assessing the statistical significance of cluster membership for categorical data. Under the null hypothesis that there is no association between one sample and a cluster, we can employ the Fisher's exact test to derive a p-value with respect to each attribute. By combining p-values from all attributes via meta-analysis, we can obtain a consensus p-value to quantify the cluster membership, i.e., if the sample is statistically associated with the target cluster. To show the benefit of such a cluster membership evaluation technique, we deploy our algorithm to several applications, ranging from cluster validation to cluster refinement and enhancement. Experimental results on real categorical data sets demonstrate the rationale and effectiveness of our method, including its potential to improve the overall clustering accuracy of both classical and state-of-the-art clustering algorithms.

1. Introduction

Clustering, a fundamental technique in machine learning for exploratory data analysis, is widely utilized in diverse fields such as biomedical research, social sciences, economics, etc (Ezugwu et al., 2022). Its primary aim is to organize heterogeneous data into meaningful groups. To achieve this, various clustering algorithms have been developed for both numerical data (Jain, 2010) and categorical data (Dinh et al., 2025). These algorithms aim to assign samples to distinct clusters, typically by optimizing objective functions such as intra-cluster compactness and inter-cluster separation.

The most commonly used k-means-type clustering algorithms employ iterative optimization to find a local optimum. Many other clustering algorithms follow a similar approach and may fail to achieve a global optimum. These limitations can result in some samples being assigned to suboptimal clusters and may inevitably introduce uncontrolled factors into the reported clustering results. This is particularly problematic when clustering algorithms are applied to random data with insufficient preprocessing, yet the algorithm still optimizes to report a set of "distinct" clusters. Given the unsupervised nature of clustering algorithms, it is crucial to assess whether the assignment of each individual sample to its cluster is correct. This involves determining if one sample truly belongs to a particular cluster, i.e., its authentic cluster membership.

To evaluate the quality of clustering results in post-hoc analysis, numerous cluster validity indexes (CVIs) (Liu et al., 2013), as indicator-based functions of all cluster memberships, have been proposed. These CVIs are applicable to both numerical (Žalik and Žalik, 2011; Lee et al., 2018) and categorical data (Bai and Liang, 2015; Zhao et al., 2017). However, these methods typically use indicators to assess the clusters in their entirety, evaluating the entire partition (cluster-level) rather than the correctness or statistical uncertainty of each sample's cluster membership (sample-level). The latter provides a personalized interpretation for each sample, based on its current assigned cluster or potential assignments to alternative clusters.

Research on cluster membership evaluation is still in its infancy, with statistical inference techniques being employed to quantify the uncertainty between a sample and a target cluster. Although several methods for cluster membership evaluation have been proposed, they are primarily designed for panel data in economics (Dzemski and Okui, 2024) and numerical data in biology (Chung, 2020), rather than for categorical data. From a statistical perspective, different data types require distinct modeling approaches, with corresponding statistics varying

* Correspondence to: Economy & Technology Development Zone, No.321 Tuqiang Street, Dalian, Liaoning, 116620, China
E-mail addresses: lyhu@haut.edu.cn (L. Hu), zyhe@dlut.edu.cn (Z. He).

https://doi.org/10.1016/j.engappai.2025.111861
Received 19 May 2024; Received in revised form 22 June 2025; Accepted 21 July 2025
Available online 7 August 2025

accordingly. For instance, the method proposed in Chung (2020) relies on a test statistic based on the cluster center, a geometric concept that lacks a meaningful interpretation for categorical data. Furthermore, well-established methods that utilize sampling approaches to derive p-values (Chung, 2020) are often time-consuming and unstable, making them less reliable for accurate membership evaluation. This underscores the need for solutions that analytically derive p-values to determine the statistical significance of cluster membership in categorical data.

To our knowledge, this is the first effort to investigate the problem of assessing the statistical **sig**nificance of **c**luster **m**embership for categorical data. The presented method is named as SigCM, which works as follows. Given a target cluster and one candidate sample, the null hypothesis is that the candidate sample has no association with the target cluster (i.e., the sample does not belongs to the cluster). To make the association assessment tractable, we make an assumption on the independence of different attributes. Thereafter, we employ the Fisher's exact test to quantify the statistical significance of cluster membership with respect to each attribute in terms of a p-value. Then, we combine the p-values from all attributes via meta-analysis (Chang et al., 2013; Cinar and Viechtbauer, 2022) to form a consensus p-value. Obviously, a smaller final p-value indicates a more accurate cluster membership assignment.

Our method offers multifaceted potential applications: (1) In the absence of external class labels, the cluster membership validation result can be used to define a measure that is analogous to clustering accuracy, potentially enabling us to validate the clustering result in an unsupervised manner. (2) By reassigning samples to the cluster with the lowest p-value, we can enhance the clustering result under the assumption that no outlying samples are present. (3) By removing samples in a cluster that cannot pass a multiple testing correction procedure (Noble, 2009), we can further improve the purity of clusters under the assumption that outliers are present in the data set.

In summary, the main contributions of this work are as follows:

- For the first time, how to accurately assess the statistical significance of cluster membership for categorical data is introduced into the literature.
- A novel algorithm called SigCM is presented, which yields an analytical p-value based on Fisher's exact test and meta-analysis to quantify the statistical significance of cluster membership.
- Through the ingenious use of cluster membership quantification results in terms of p-values, our algorithm can be employed for both cluster validation and cluster enhancement. It opens up new avenues for validating and improving clustering results from an angle that still remains untouched.

The remainder of this paper is structured as follows: Section 2 highlights the motivation behind our method and discusses recent literature. Section 3 presents our proposed method and its applications. Section 4 presents experimental results on real data sets. Section 5 concludes this paper.

2. Motivation and related works

The following motivations illustrate why our method plays a crucial role in cluster analysis and why it is essential to use it:

- Motivation 1: Our method aims to evaluate cluster membership. What role does this research field play in cluster analysis?
- Motivation 2: Building upon Motivation 1, what are the benefits of our method, particularly its use of statistical techniques?

2.1. Answer to motivation 1

An ideal clustering pipeline is shown in Fig. 1. In cluster analysis, much of the research has focused on generating high-accuracy clustering results, referred to as the in-clustering stage. During this stage, clustering algorithms designed for categorical data have developed from early methods like k-modes (Huang, 1998) to a range of state-of-the-art (SOTA) algorithms, including both hard (Zhang and Cheung, 2022) and fuzzy (Zhang et al., 2023) clustering methods. Due to the complex relationships among attribute values in categorical data (Qian et al., 2016; Jian et al., 2019; Zhu et al., 2022; Park et al., 2024), recent SOTA algorithms have increasingly focused on similarity-based representation learning techniques (Bai and Liang, 2022; Zhang et al., 2022; Jian et al., 2018; Zhang et al., 2025). However, no single SOTA in-clustering algorithm guarantees optimal performance across all data sets, and the resulting clusters still require subsequent validation. Additionally, in the pre-clustering stage, it remains unclear whether the application of advanced categorical data processing methods, such as feature selection (Bandyapadhyay et al., 2023; Yuan et al., 2024; Ling et al., 2025) and outlier detection (Li et al., 2023; Zhao et al., 2024b; Song et al., 2025), is ultimately beneficial or detrimental to subsequent clustering outcomes. These considerations collectively underscore the need for a post-clustering stage in cluster analysis, which aims to evaluate and manage clusters, particularly before applying them in high-risk domains. To further assess the impact of individual samples in the post-clustering stage, our method provides a more direct indicator for each sample, offering finer granularity than cluster-level functions such as CVIs.

2.2. Answer to motivation 2

For non-statistical cluster membership, assignments are typically optimized by clustering algorithms. Since these in-clustering methods always produce a final result, they are limited in reporting the extent of noise and spurious patterns, making it difficult to assess whether the results are significantly better than arbitrary assignments or random partitions. While CVIs can evaluate clustering results, they do not provide an indicator with direct qualitative meaning to determine whether the clustering outcome is valid.

As shown in Fig. 1, our statistical cluster membership evaluation method introduces a p-value indicator to determine whether a sample should be assigned to a cluster based on a predefined significance level. If the p-value for a target cluster is less than or equal to this threshold, the assignment is considered statistically significant and recommended; otherwise, it is not. Analogous to cluster assignments in clustering algorithms, a hard assignment is made if a sample has a statistically significant p-value for only one cluster. If multiple clusters meet the statistical threshold, a fuzzy assignment is made, with weights can be determined by the relative magnitudes of the p-values. A key distinction from non-statistical clustering algorithms is that our method allows for the possibility of no assignment if none of a sample's cluster assignments meet the significance threshold.

It is worth noting that various efforts have been made to enhance the statistical robustness and interpretation of clustering for categorical data; however, these approaches differ fundamentally from our post-clustering method. Prior work includes studies in the in-clustering stage that assess the statistical significance of a partition (Hu et al., 2025d) and evaluate the goodness of each split in a clustering tree (Hu et al., 2025c,b). In the pre-clustering stage, a clusterability test has been proposed (Hu et al., 2025a) to determine whether the categorical data are suitable for clustering before applying any clustering algorithm.

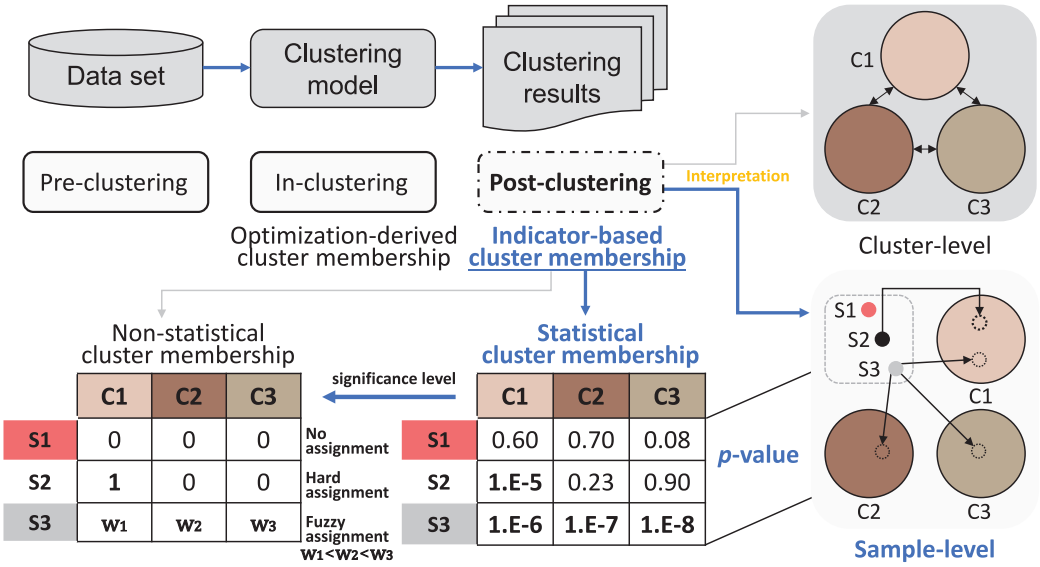


Fig. 1. The role of statistical cluster membership evaluation methods in an ideal cluster analysis pipeline. The pre-clustering stage involves clusterability tests to assess whether inherent cluster structures exist in the data; the in-clustering stage uses clustering algorithms to optimize cluster memberships; and the post-clustering stage focuses on evaluating cluster memberships for subsequent validation or adjustment of clustering results. Our statistical cluster membership evaluation method provides a sample-level *p*-value interpretation for each cluster membership, potentially guiding cluster assignments during post-hoc operations.

3. Algorithm and applications

3.1. Problem formulation

Suppose a categorical data set $D = \{x_1, \ldots, x_N\}$, containing N samples where each sample $x_i = [x_{i1}, \ldots, x_{iM}]$ is composed of M attribute values, is divided into K non-overlapping clusters, denoted as $D = D_1 \cup \cdots \cup D_K$. In the kth cluster D_k, there are $N^{(k)}$ samples $x_i^{(k)} \in \{x_1^{(k)}, \ldots, x_{N^{(k)}}^{(k)}\}$. For a candidate sample $x_i^{(k)}$ from the kth cluster and a target k'-th cluster, we assess whether $x_i^{(k)}$ belongs to cluster k' in terms of a p-value that is denoted by $pval(x_i^{(k\to k')})$. When $k' = k$, the p-value evaluates the cluster membership of $x_i^{(k)}$ with respect to its currently allocated cluster.

Algorithm 1 SigCM

Input: A sample $x_i^{(k)}$ and a target cluster $D_{k'}$ from the given categorical data set D, and a threshold α.
Output: An analytical p-value $pval(x_i^{(k\to k')})$.
1: **for** $m = 1$ **to** M **do**
2: $\quad pval(x_{im}^{(k\to k')}) \Leftarrow$ Fisher's exact test $\left(x_{im}^{(k)}, D_{k'}, D \setminus D_{k'}\right)$
3: **end for**
4: $pval(x_i^{(k\to k')}) \Leftarrow$ Binomial test $\left(pval(x_{i1:M}^{(k\to k')}), \alpha\right)$
5: **return** $pval(x_i^{(k\to k')})$

3.2. The SigCM algorithm

Given any sample in the data set and a target cluster, our proposed SigCM algorithm calculates a p-value to assess its cluster membership. That is, $pval(x_i^{(k\to k')})$ will be returned from SigCM $\left(x_i^{(k)}, D_{k'}, D\right)$ as shown in Algorithm 1. We calculate p-values for all M attributes (Lines 1∼3) by using the Fisher's exact test. Then, we combine all M p-values denoted as $pval(x_{i1:M}^{(k\to k')})$ into a single p-value via a meta-analysis method. Here we employ the Binomial test (Cinar and Viechtbauer, 2022) with a user-specified threshold α.

3.2.1. Fisher's exact test

For the mth attribute, measuring the association between the sample $x_i^{(k)}$ and the k'-th cluster is equivalent to assessing the statistical association between two binary variables defined below. One is the cluster variable denoted by C, where $C = 1$ if the sample belongs to $D_{k'}$ and $C = 0$ otherwise. Another is the category variable denoted by Q, where $Q = 1$ if the mth attribute value of the sample equals $x_{im}^{(k)}$ and $Q = 0$ otherwise.

Suppose N_q and N_q^{in} samples take the attribute value $x_{im}^{(k)}$ in D and $D_{k'}$, respectively. Then, $N_q = N_q^{in} + N_q^{out}$, where N_q^{out} is the number of samples that take the attribute value $x_{im}^{(k)}$ in $D \setminus D_{k'}$. Meanwhile, the total number of samples in $D \setminus D_{k'}$ is denoted by $\tilde{N}^{(k')}$, with $N = N^{(k')} + \tilde{N}^{(k')}$. Based on above notations and definitions, we can construct the following contingency table:

	$Q=1$	$Q=0$	Total
$C=1$	N_q^{in}	$N^{(k')} - N_q^{in}$	$N^{(k')}$
$C=0$	N_q^{out}	$\tilde{N}^{(k')} - N_q^{out}$	$\tilde{N}^{(k')}$
Total	N_q	$N - N_q$	N

The Fisher's exact test is one of the most widely used methods for testing the association between two binary variables. Under the null hypothesis of no association between the sample $x_i^{(k)}$ and the k'-th cluster with respect to the mth attribute, i.e., the independence between Q and C, the cell count N_q^{in} follows a hypergeometric distribution:

$$P(N_q^{in} | D_{k'}) = \frac{\binom{N^{(k')}}{N_q^{in}} \times \binom{\tilde{N}^{(k')}}{N_q^{out}}}{\binom{N}{N_q}} . \tag{1}$$

To quantify if the null hypothesis of no association is true, the p-value is typically employed in significance testing. The p-value is the probability of obtaining results at least as extreme as the observed results when the null hypothesis is true. In our problem, more extreme results correspond to the cases that the cell count when $Q = 1$ and $C = 1$ is larger than N_q^{in}. This is, it is considered to be more extreme if the attribute value $x_{im}^{(k)}$ is over-expressed in the cluster $D_{k'}$. Hence, the p-value can be calculated as the following cumulative probability:

$$pval(x_{im}^{(k\to k')}) = \sum_{n=0}^{\min\left(N^{(k')} - N_q^{in}, N_q^{out}\right)} P(N_q^{in} + n \mid D_{k'}) , \tag{2}$$

where $\min\left(N^{(k')} - N_q^{in}, N_q^{out}\right)$ indicates the upper bound on the number of samples that can take the attribute value $x_{im}^{(k)}$ within the target cluster $D_{k'}$ in addition to another N_q^{in} samples. A concise illustrative example for Eq. (2) is shown in Fig. 2. Given that an attribute value appears 6

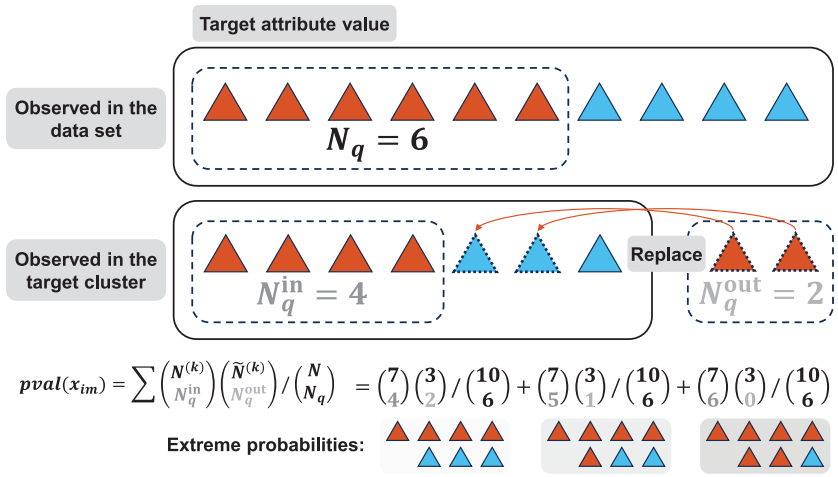


Fig. 2. Illustration of extreme probabilities summed in Eq. (2).

times in a data set of 10 samples, the figure illustrates more extreme cases where it occurs 4 or more times within an observed target cluster of 7 samples.

3.2.2. Binomial test

In the Binomial test, we aim to consolidate p-values from the Fisher's exact test on M attributes into a single p-value. We start by defining a new test statistic that aggregates the statistical significance of M tests:

$$r = \sum_{m=1}^{M} \delta(pval(x_{im}^{(k\to k')}), \alpha) \ , \tag{3}$$

where

$$\delta(pval(x_{im}^{(k\to k')}), \alpha) = \begin{cases} 0 & \text{if } pval(x_{im}^{(k\to k')}) > \alpha \\ 1 & \text{if } pval(x_{im}^{(k\to k')}) \le \alpha \end{cases} \tag{4}$$

is an indicator function for each attribute based on the threshold α. Assuming that all M null hypotheses are true, the number of tests r that leads to rejection follows a Binomial distribution with a rejection probability of α. The cumulative function, which accounts for more significant individual cases, is then calculated to determine the final p-value as follows:

$$pval(x_i^{(k\to k')}) = \sum_{h=r}^{M} \binom{M}{h} \alpha^h \ (1-\alpha)^{M-h} \ , \tag{5}$$

which is the probability of obtaining at least r rejections under the joint null. Eq. (5) aggregates M individual testing results, providing a comprehensive assessment on the cluster membership with respect to all M attributes.

3.2.3. Time complexity analysis of SigCM

To derive the p-value of a target cluster membership for a sample $x_i^{(k)}$ using SigCM (Algorithm 1), the procedure primarily consists of two steps:

- Fisher's exact test step: In the worst case, computing $pval(x_{im}^{(k\to k')})$ in Eq. (2) requires to calculate at most N probabilities. Each $P(N_q^{in}|D_{k'})$ in Eq. (1) can be obtained in $\mathcal{O}(1)$ time using a hash table to store the counts. Thus, each Fisher's exact test is computed in $\mathcal{O}(N)$ time. Since M Fisher's exact tests are performed in Algorithm 1, the worst-case time complexity for this step is $\mathcal{O}(NM)$.
- Binomial test step: Given the M p-values obtained from Fisher's exact test step, the final aggregated p-value is computed by summing up at most M binomial probability mass functions in Eq. (5), leading to a worst-case time complexity of $\mathcal{O}(M)$.

Combining both steps, the overall worst-case time complexity of the SigCM algorithm is $\mathcal{O}(NM) + \mathcal{O}(M) = \mathcal{O}(NM)$.

3.3. Applications

To demonstrate the usefulness of SigCM, we explore three key applications: cluster validation, cluster refinement, and cluster enhancement, as illustrated in Fig. 3.

For cluster validation, we will introduce a new validity index based on a multiple testing correction procedure. More precisely, the Bonferroni correction procedure (Cui et al., 2021) is employed to control the Family-Wise Error Rate (FWER) of each cluster. That is, only the samples with p-values that are smaller than the adjusted significance level are considered to be true members of the corresponding cluster. The validation index named as **C**luster **M**embership **I**ndex (CMI) is defined as the percentage of samples that can pass the FWER control. For cluster refinement and enhancement, we can remove outliers through FWER control or reassign samples to other clusters with smaller p-values to improve the cluster quality. The details of these applications are presented in the following three subsections.

3.3.1. Cluster membership index

Algorithm 2 Validation operation

Input: A categorical data set partitioned into K clusters, $D = D_1 \cup \cdots \cup D_K$, and a threshold α.
Output: A validation index CMI.

```
for i = 1 to N do
    pval(x_i^{(k→k)}) ⇐ SigCM (x_i^{(k)}, D_k)
end for
CMI ⇐ The percentage of x_i^{(k)} passing FWER (pval(x_{i:N}^{(k→k)}), α)
return CMI
```

The proposed validity index, termed CMI, is defined by the proportion of samples whose p-values are no larger than a cluster-specific adjusted significance level:

$$\text{CMI} = \frac{\sum_{k=1}^{K} \sum_{i=1}^{N^{(k)}} \delta\left(pval(x_i^{(k\to k)}), \frac{\alpha}{N^{(k)}}\right)}{N} \ , \tag{6}$$

where

$$\delta\left(pval(x_i^{(k\to k)}), \frac{\alpha}{N^{(k)}}\right) = \begin{cases} 0 & \text{if } pval(x_i^{(k\to k)}) > \frac{\alpha}{N^{(k)}} \\ 1 & \text{if } pval(x_i^{(k\to k)}) \le \frac{\alpha}{N^{(k)}} \end{cases} \tag{7}$$

indicates a binary decision for each sample $x_i^{(k)}$ with respect to its currently allocated cluster, based on the adjusted significance level

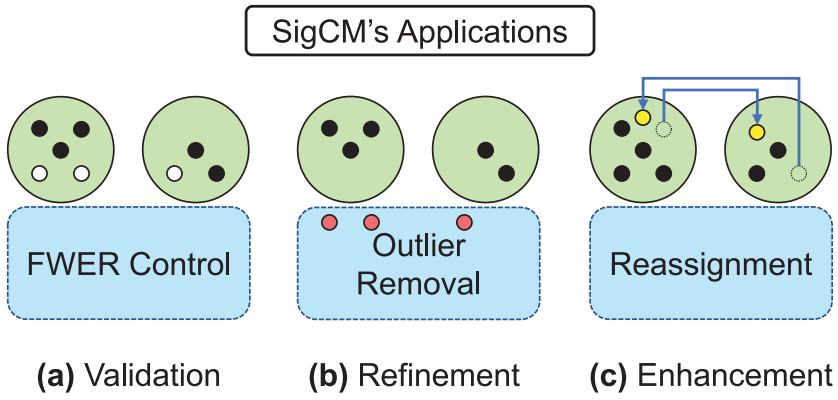


Fig. 3. Three potential applications by employing the cluster membership evaluation capability in SigCM. Taking 8 samples from 2 clusters as an example, **(a)** there are 3 samples colored white that do not pass the FWER control, with CMI = 5/8 calculated based on the proportion of remaining samples to all samples. **(b)** Similar to (a), the three samples colored red can be viewed as outliers and removed to refine clusters. **(c)** Two samples colored yellow are reassigned to another cluster with a smaller cluster membership p-value.

$\frac{\alpha}{N^{(k)}}$. This adjustment accounts for multiple comparisons within D_k. The validation operation based on SigCM is presented in Algorithm 2.

Given a clustering result, the CMI can assess the validity of clusters. Specifically, CMI is analogous to cluster accuracy since it also ranges from 0 to 1 and is defined based on the percentage of "correctly" identified samples. More importantly, our index is an internal one, which can be calculated in an unsupervised manner.

3.3.2. Cluster refinement via outlier removal

Algorithm 3 Refinement operation

Input: A categorical data set partitioned into K clusters, $D = D_1 \cup \cdots \cup D_K$, and a threshold α.
Output: A refined categorical data set with K clusters, $\hat{D} = \hat{D}_1 \cup \cdots \cup \hat{D}_K$.

for $k = 1$ **to** K **do**
  Initialize $\hat{D}_k \leftarrow D_k$
  for $j = 1$ **to** $N^{(k)}$ **do**
    $pval(x_j^{(k\to k)}) \Leftarrow \text{SigCM}\left(x_j^{(k)}, D_k\right)$
    $\hat{D}_k \Leftarrow \hat{D}_k \setminus \{x_j^{(k)} \mid \text{failing to pass FWER } (p\text{val}(x_j^{(k\to k)}), \alpha)\}$
  end for
end for
return $\hat{D} = \hat{D}_1 \cup \cdots \cup \hat{D}_K$

For the kth cluster D_k, the refinement can be achieved by removing specific samples identified as outliers. The resulting new cluster, denoted as $\hat{D}_k$, is defined as follows:

$$\hat{D}_k = D_k \setminus \{x_i^{(k)} \mid \delta\big(pval(x_i^{(k\to k)}), \frac{\alpha}{N^{(k)}}\big) = 0\}\ , \tag{8}$$

where all samples that cannot pass the FWER control are removed. The refinement operation based on SigCM is described in Algorithm 3.

If the CMI is close to 0, or if more than half of the samples in D_k cannot pass the FWER control, i.e., $\left|\{x_i^{(k)} \mid \delta\big(pval(x_i^{(k\to k)}), \frac{\alpha}{N^{(k)}}\big) = 0\}\right| > \frac{N^{(k)}}{2}$, we will not perform this refinement process in such cases, as the cluster is probably not correctly identified.

3.3.3. Cluster enhancement via reassignment

Beyond measures focused on cluster validation, the output of SigCM also enables the reassignment of cluster labels. This is achieved by reassigning each sample to the cluster with the smallest cluster membership p-value, provided that this p-value also passes the FWER control of the newly assigned cluster. The new cluster for any sample $x_i^{(k)}$, denoted as $\hat{k}_{x_i^{(k)}}$, is given as follows:

$$\hat{k}_{x_i^{(k)}} = \arg\min_{k' \in [1:K]}\ pval(x_i^{(k\to k')})\ , \quad \text{subject to} \quad \delta\big(pval(x_i^{(k\to \hat{k})}), \frac{\alpha}{N^{(\hat{k})}}\big) = 1\ . \tag{9}$$

Algorithm 4 Enhancement operation

Input: A categorical data set partitioned into K clusters, $D = D_1 \cup \cdots \cup D_K$, and a threshold α.
Output: Enhanced cluster assignment to N samples, each with label: $\hat{k}_{x_i^{(k)}}$.

for $i = 1$ **to** N **do**
  Initialize $pval_{\text{set}} \leftarrow \emptyset$
  for $k' = 1$ **to** K **do**
    $pval(x_i^{(k\to k')}) \Leftarrow \text{SigCM}\left(x_i^{(k)}, D_{k'}\right)$
    if passing FWER $(p\text{val}(x_i^{(k\to k')}), \alpha)$ **then**
      $pval_{\text{set}} \leftarrow pval_{\text{set}} \cup \{pval(x_i^{(k\to k')})\}$
    end if
  end for
  if $pval_{\text{set}} \neq \emptyset$ **then**
    $\hat{k}_{x_i^{(k)}} \Leftarrow$ cluster label corresponding to the smallest p-value in $pval_{\text{set}}$
  end if
end for
return $\hat{k}_{x_i^{(k)}}$

The enhancement operation based on SigCM is described in Algorithm 4. Note that the reassignment here is a one-step procedure. It involves applying SigCM to all samples in the fixed clustering results, followed by a unified reassignment. Other methods of reassignment that involve multiple iterations, such as local updates or step-by-step reassignment where each step builds upon the results of the previous one, have not yet been explored.

4. Experimental results

To verify the effectiveness of our proposed SigCM in assessing cluster membership from a statistical significance perspective, we first introduce the evaluation strategy and criteria in Section 4.1. Before demonstrating the applications of SigCM, we compare our method

Table 1
The properties of 18 UCI categorical data sets, which are available at https://archive.ics.uci.edu/datasets. $|Q|$ indicates the number of categories from all attributes.

Data set	*Abbr.*	N	M	$\|Q\|$	K
Lenses	Ls	24	4	9	3
Lung Cancer	Lc	32	56	159	3
Soybean (Small)	So	47	21	58	4
Zoo	Zo	101	16	36	7
Promoter Sequences	Ps	106	57	228	2
Hayes-Roth	Hr	132	4	15	3
Lymphography	Ly	148	18	59	4
Heart Disease	Hd	303	13	57	5
Solar Flare	Sf	323	9	25	6
Primary Tumor	Pt	339	17	42	21
Dermatology	De	366	33	129	6
House Votes	Hv	435	16	48	2
Balance Scale	Bs	625	4	20	3
Credit Approval	Ca	690	9	45	2
Breast Cancer	Bc	699	9	90	2
Mammographic Mass	Mm	824	4	18	2
Tic-Tac-Toe	Tt	958	9	27	2
Car Evaluation	Ce	1728	6	21	4

with Jackstraw[1] (Chung, 2020) in Section 4.2, the only comparable method currently available. Jackstraw is designed for numerical data and requires categorical data to be encoded into a numerical format before use. However, as shown by the preliminary experimental results in Section 4.2, Jackstraw fails to meet the validation requirements for cluster membership evaluation on categorical data and is therefore excluded from subsequent experiments. In Sections Sections 4.3–4.5, we individually examine the applications of SigCM, including the proposed CMI, cluster refinement, and cluster enhancement. To demonstrate the scalability of these SigCM-based applications, we apply SigCM algorithm to large-scale categorical data sets in Section 4.6 and propose potential strategies to accelerate computation.

4.1. Data sets and evaluation criteria

In the experiment[2], we choose 18 real categorical data sets from the widely-used UCI repository (Dua and Graff, 2019). The characteristics of these data sets we employed are presented in Table 1.

For each data set D, the ground-truth cluster label $\dot{k}_i$ of each sample x_i in D is available. These labels are used to compute benchmark external evaluation metrics (Rezaei and Fränti, 2016), including Clustering Accuracy (ACC), Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), and F1-score (FSC), which serve as the gold standard for evaluating our declared performance. Higher values of these metrics suggest better clustering quality.

Given the clustering labels D and the ground-truth labels $\dot{D}$ for all samples in D, where each sample x_i has a ground-truth cluster label $\dot{k}_i$. The ACC is defined as follows:

$$\mathrm{ACC} = \frac{\sum_{i=1}^{N} \delta\left(\dot{k}_i, \mathrm{map}(k_i)\right)}{N}, \tag{10}$$

where $\mathrm{map}(k_i)$ is a mapping function that aligns k_i with its equivalent true label $\dot{k}_i$, and $\delta(\dot{k}_i, \mathrm{map}(k_i)) = 1$ if $\dot{k}_i = \mathrm{map}(k_i)$ and 0 otherwise. We use the Kuhn–Munkres algorithm (Cai et al., 2005) to achieve the best mapping.

The NMI is defined a follows:

$$\mathrm{NMI} = \frac{1}{2} \cdot \frac{\mathrm{H}(D) + \mathrm{H}(\dot{D}) - \mathrm{H}(D, \dot{D})}{\mathrm{H}(D) + \mathrm{H}(\dot{D})}, \tag{11}$$

where both types of labels are treated as random variables. Here, $\mathrm{H}(D)$ and $\mathrm{H}(\dot{D})$ denote their respective entropies, while $\mathrm{H}(D, \dot{D})$ represents their joint entropy.

The ARI is defined as follows:

$$\mathrm{ARI} = \frac{\mathrm{RI} - \mathbb{E}[\mathrm{RI}]}{\max(\mathrm{RI}) - \mathbb{E}[\mathrm{RI}]}, \tag{12}$$

where the Rand Index (RI) measures the proportion of correctly assigned sample pairs, while $\mathbb{E}[\mathrm{RI}]$ represents its expected value and $\max(\mathrm{RI})$ denotes its maximum value. The latter two can be computed using a permutation model. The ARI ranges between -1 and 1, adjusting the RI. The RI is defined as follows:

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}}. \tag{13}$$

In Eq. (13), TP (True Positive) represents the number of sample pairs that share the same label in $\dot{D}$ and are correctly assigned to the same cluster in D; FP (False Positive) refers to the number of sample pairs with different labels in $\dot{D}$ that are incorrectly clustered together in D; FN (False Negative) denotes the number of sample pairs that share the same label in $\dot{D}$ but are incorrectly assigned to different clusters in D; TN (True Negative) represents the number of sample pairs with different labels in $\dot{D}$ that are correctly assigned to different clusters in D.

The FSC is defined a follows:

$$\mathrm{FSC} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}, \tag{14}$$

where

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \quad \mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{15}$$

In the upcoming experiments, we demonstrate the efficacy of CMI as an alternative to these metrics and examine how cluster quality, as measured by these metrics, changes before and after applying our cluster refinement and enhancement operations based on SigCM.

All evaluations are conducted based on three types of partitions: (i) random partitions, (ii) partitions obtained by clustering algorithms, and (iii) the ground-truth partition. These partitions inherently reflect different levels of confidence in cluster membership assignments. In random partitions, nearly all samples are assumed to be incorrectly assigned to their true clusters. In contrast, partitions generated by clustering algorithms tend to assign most samples to the correct clusters. For the ground-truth partition, the values of ACC, NMI, ARI, and FSC are 1, although exceptions may occur in practice due to misleading manually annotated cluster labels. The generation process of these partitions is described as follows:

- Random partitions: Each data set undergoes 100 independent random partitioning processes, where the ground-truth labels of all samples are randomly permuted. In general, such partitions result in invalid cluster memberships.
- Algorithm-derived partitions: We run the classical k-modes clustering algorithm (Huang, 1998) and five representative SOTA algorithms, namely CDCDR[3] (Bai and Liang, 2022), Het2Hom[4] (Zhang et al., 2022), CMS[5] (Jian et al., 2018), ADC[6] (Zhang and Cheung, 2023) and COForest (Zhao et al., 2024a), each for 100 independent runs on each data set. Due to algorithmic randomness, these methods produce varying results with different levels of quality. To represent overall performance, we report the average results across all runs.

[1] The 'jackstraw' R package used in our experiments was obtained from https://cran.r-project.org/web/packages/jackstraw. In our implementation, the function `jackstraw_cluster` is used to specify cluster centers externally.

[2] The SigCM algorithm and the code for generating experimental data are available at https://github.com/hulianyu/SigCM.

[3] In our experiments, the autoencoder and joint operation are selected in the proposed framework of the original paper.

[4] Since the data sets used in our experiments do not consider ordinal attributes, the number of ordinal attributes is set to 0 in the implementation.

[5] The parameter related to intra- and inter-attribute couplings is set to 0.5 as recommended in the original paper, and spectral clustering is finally employed.

[6] Consistent with the parameter settings in Het2Hom, the number of nominal attributes is set to M in the implementation.

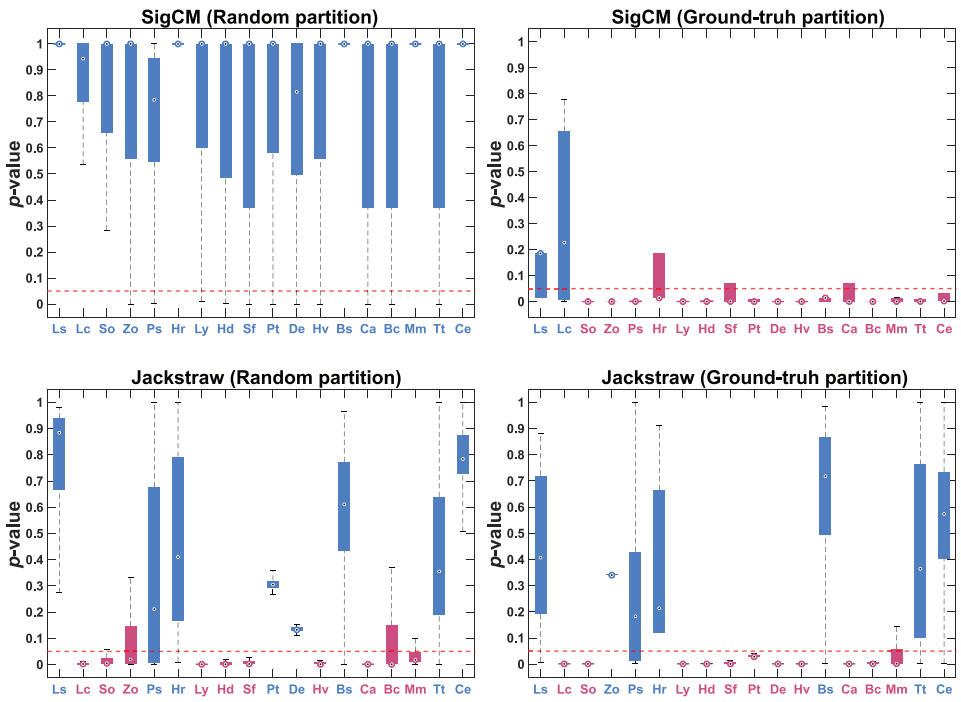


Fig. 4. The boxplot shows the distribution of p-values for assigned cluster membership, as output by SigCM and Jackstraw, across 100 random partitions and the ground-truth partition on 18 data sets. To run Jackstraw, categorical variables are one-hot encoded into numerical data. Jackstraw also requires cluster centers, which are calculated based on the provided partition labels. The boxplot is marked in blue if its median exceeds the significance level of 0.05 (indicated by the horizontal dashed line), and in red if it is less than or equal to 0.05.

4.2. SigCM vs. Jackstraw

To demonstrate the preliminary applicability of SigCM in assessing cluster membership, we employed two distinct types of partitions for each categorical data set: 100 random partitions, regarded as spurious clustering results, and one ground-truth partition, considered the de facto correct clustering result. Based on these partitions, we used SigCM to compute each sample's p-value for its assigned cluster membership. To compare cluster membership p-values under the same setting, we implemented the Jackstraw method, which requires all categorical attributes to be converted into numerical format via one-hot encoding. Additionally, Jackstraw relies on cluster centers, which were obtained by computing the mean of each attribute value based on the given clusters in the partitions.

As illustrated in Fig. 4, the boxplots display the distribution of derived p-values for all samples across 18 UCI categorical data sets in two partition groups. The observations from the experimental results, which suggest that SigCM is more reliable than Jackstraw, are summarized as follows:

- **Random partition group**: SigCM algorithm consistently reports larger p-values for each sample's cluster membership, with all randomly assigned cluster memberships deemed invalid, as their p-values exceed the significance level of 0.05. Additionally, SigCM demonstrates statistical robustness against randomness across all 18 data sets. In contrast, Jackstraw generally produces misleading results by assigning smaller p-values, often incorrectly identifying cluster memberships as statistically significant p-values (≤ 0.05) on most data sets, including Lc, So, Zo, Ly, Hd, Sf, Hv, Ca, Bc, and Mm.
- **Ground-truth partition group:** SigCM occasionally reports large p-values (> 0.05) to cluster memberships on two data sets (Ls and Lc). However, even in these exceptional cases, the p-values remain notably smaller than those in the random partition group, reflecting a distinct clustering tendency. In contrast, this trend is frequently violated in Jackstraw's results, typically observed in data sets such as Zo and Bs. Furthermore, Jackstraw generally fails to recognize the correct assignments based on ground-truth labels, reporting unexpected p-values (> 0.05) on 7 data sets: Ls, Zo, Ps, Hr, Bs, Tt, and Ce.

4.3. CMI vs. external metrics

To further demonstrate the effectiveness of SigCM, we utilize its derived validation index, CMI, to evaluate cluster quality without requiring ground-truth labels. We compare the performance of CMI against external metrics across different partitions: invalid (random) partitions, which serve as a baseline reference, and optimized partitions, generated by both classical and SOTA clustering algorithms, as detailed in Table 2. A simple example is shown in Fig. 5 to illustrate the characteristics of CMI. Specifically, under invalid partitions, the CMI value tends toward 0, identifying nearly all samples as being statistically non-significant; whereas under higher-quality partitions, it tends toward 1, identifying more samples as being statistically significant and fewer deemed to be outliers.

In the absence of a quantitative cutoff threshold for determining what value indicates high-quality clustering, we adopt a relatively intuitive distinguishing value of 0.5 (50%) for the following reasons:

- CMI is based on counting the number of samples with statistically significant assigned cluster memberships. If exactly half of the samples in the data set meet or fail to meet the statistical significance criterion, the clustering tendency remains indeterminate.
- Since we consider statistically significant and non-significant samples to contribute equally when assessing clustering validity, we deem clustering to be of high quality if the former outnumbers the latter. Conversely, if fewer than half do, the clustering is deemed to be of low quality.

Table 2
Average CMI and external metrics across diverse sets of 100 partitions generated by random partitioning or different clustering algorithms. The last column shows "#>0.5", representing the number of data sets for which the average metric exceeds 0.5. Values greater than 0.5 are marked in bold.

@Validation		Ls	Lc	So	Zo	Ps	Hr	Ly	Hd	Sf	Pt	De	Hv	Bs	Ca	Bc	Mm	Tt	Ce	#>0.5
CMI	Random	0	0	0.016	0.015	4.E−04	2.E−04	9.E−04	0.001	0.002	4.E−04	0.019	0.009	1.E−04	2.E−04	0.007	0	3.E−04	3.E−05	0
	k-modes	0.019	0.471	**0.994**	**0.968**	0.495	0.445	**0.945**	**0.930**	**0.838**	**0.907**	**0.999**	**0.992**	0.011	0.495	**0.899**	0.304	0.348	0.195	9
	CDCDR	0	0.412	**0.981**	**0.957**	0.439	0.127	**0.926**	**0.731**	**0.890**	**0.826**	**1.000**	**0.945**	0.000	0.258	**0.774**	0.312	0.173	0.009	9
	Het2Hom	0.025	0.391	**1**	**0.934**	0.002	0.227	**0.930**	**0.814**	**0.921**	**0.843**	**1**	**0.991**	0	0.444	**0.960**	0.329	0.238	0.004	9
	CMS	0	**0.780**	**1**	**0.932**	**0.601**	0.227	**0.976**	**0.953**	**0.928**	**0.563**	**1**	**0.993**	0.015	**0.500**	**0.977**	0.309	0	0	12
	ADC	0.025	0.383	**0.991**	**0.958**	**0.563**	0.347	**0.913**	**0.942**	**0.887**	**0.864**	**0.997**	**0.993**	0.009	0.431	**0.953**	0.303	0.332	0.176	10
	COForest	0.013	**0.611**	**0.995**	**0.970**	**0.579**	0.258	**0.842**	**0.769**	**0.876**	**0.893**	**0.999**	**0.993**	0.003	0.378	**0.889**	0.306	0.159	0.009	11
ACC	Random	0.5	0.453	0.371	0.295	**0.536**	0.391	0.494	0.367	0.256	0.172	0.232	**0.527**	0.446	**0.513**	**0.547**	**0.512**	**0.548**	**0.543**	7
	k-modes	**0.555**	**0.523**	**0.851**	**0.721**	**0.562**	0.369	0.443	0.396	0.478	0.296	**0.589**	**0.866**	0.423	**0.754**	**0.904**	**0.818**	**0.549**	0.385	11
	CDCDR	0.471	**0.538**	**0.891**	**0.689**	**0.573**	0.488	0.476	0.388	0.412	0.282	**0.723**	**0.832**	0.449	**0.629**	**0.794**	**0.798**	**0.559**	0.393	10
	Het2Hom	**0.533**	**0.557**	**1**	**0.704**	**0.502**	0.333	0.495	0.349	0.443	0.284	**0.750**	**0.874**	0.467	**0.827**	**0.967**	**0.818**	**0.552**	0.368	11
	CMS	**0.563**	**0.583**	**1**	**0.633**	**0.766**	0.341	**0.502**	0.317	0.441	0.261	**0.811**	**0.878**	0.381	**0.672**	**0.964**	**0.826**	**0.517**	0.400	12
	ADC	**0.526**	**0.508**	**0.891**	**0.721**	**0.615**	0.382	**0.501**	0.423	0.485	0.313	**0.639**	**0.871**	0.442	**0.651**	**0.958**	**0.820**	**0.543**	0.369	12
	COForest	**0.554**	**0.548**	**0.836**	**0.707**	**0.626**	0.408	0.499	0.368	0.417	0.277	**0.708**	**0.876**	0.456	**0.714**	**0.921**	**0.809**	**0.564**	0.383	11
NMI	Random	0.121	0.066	0.086	0.117	0.006	0.014	0.031	0.022	0.027	0.163	0.020	0.001	0.004	8.E−04	8.E−04	6.E−04	8.E−04	0.003	0
	k-modes	0.248	0.199	**0.838**	**0.749**	0.026	0.007	0.112	0.153	0.290	0.339	**0.572**	0.459	0.016	0.210	**0.563**	0.326	0.012	0.069	4
	CDCDR	0.153	0.205	**0.886**	**0.707**	0.032	0.354	0.150	0.170	0.215	0.335	**0.812**	0.393	0.038	0.079	0.268	0.303	0.006	0.075	3
	Het2Hom	0.238	0.249	**1**	**0.789**	0.002	5.E−17	0.160	0.171	0.249	0.346	**0.771**	0.480	0.003	0.357	**0.777**	0.327	0.012	0.063	4
	CMS	0.287	0.232	**1**	**0.645**	0.300	3.E−04	0.183	0.127	0.234	0.310	**0.804**	0.490	0.014	0.127	**0.778**	0.338	5.E−05	0.117	4
	ADC	0.214	0.184	**0.885**	**0.783**	0.081	0.011	0.150	0.178	0.301	0.355	**0.687**	0.475	0.029	0.129	**0.748**	0.328	0.009	0.040	4
	COForest	0.283	0.248	**0.876**	**0.773**	0.085	0.029	0.130	0.135	0.230	0.324	**0.789**	0.482	0.034	0.211	**0.668**	0.320	0.008	0.095	4
ARI	Random	−0.002	0.001	0.002	6.E−04	−0.001	5.E−06	−4.E−04	−0.002	2.E−04	−4.E−04	−0.001	−0.001	2.E−05	−4.E−04	−7.E−04	−4.E−04	3.E−04	−7.E−04	0
	k-modes	0.127	0.121	**0.767**	**0.671**	0.021	−0.009	0.077	0.149	0.215	0.093	0.451	**0.534**	0.016	0.264	**0.670**	0.403	0.016	0.047	4
	CDCDR	0	0.412	**0.981**	**0.957**	0.439	0.127	**0.926**	**0.731**	**0.890**	**0.826**	**1.000**	**0.945**	4.E−04	0.258	**0.774**	0.312	0.173	0.009	9
	Het2Hom	0.099	0.168	**1**	**0.652**	4.E−04	−0.015	0.172	0.127	0.177	0.116	**0.706**	**0.557**	7.E−04	0.441	**0.871**	0.404	0.014	0.030	5
	CMS	0.167	0.168	**1**	0.496	0.280	−0.015	0.192	0.090	0.184	0.090	**0.745**	**0.571**	0.015	0.150	**0.861**	0.426	−0.001	0.077	4
	ADC	0.077	0.101	**0.836**	**0.681**	0.082	−0.004	0.139	0.188	0.229	0.125	**0.554**	**0.550**	0.028	0.154	**0.847**	0.410	0.012	0.026	5
	COForest	0.139	0.146	**0.803**	**0.666**	0.091	0.016	0.121	0.116	0.134	0.089	**0.659**	**0.566**	0.034	0.250	**0.759**	0.399	0.015	0.055	5
FSC	Random	0.437	0.322	0.252	0.234	0.495	0.345	0.467	0.352	0.213	0.108	0.198	**0.524**	0.430	**0.505**	**0.547**	0.4997	**0.547**	**0.542**	5
	k-modes	0.477	0.431	**0.831**	**0.741**	**0.512**	0.352	0.407	0.397	0.387	0.180	**0.567**	**0.773**	0.414	**0.646**	**0.857**	**0.704**	**0.535**	0.396	9
	CDCDR	0.406	0.423	**0.885**	**0.681**	**0.526**	**0.611**	0.433	0.381	0.325	0.168	**0.755**	**0.750**	0.423	**0.589**	**0.756**	**0.695**	**0.535**	0.403	10
	Het2Hom	0.439	0.483	**1**	**0.717**	**0.662**	0.327	0.447	0.360	0.337	0.179	**0.762**	**0.784**	**0.597**	**0.730**	**0.942**	**0.703**	**0.533**	0.394	10
	CMS	0.473	0.457	**1**	**0.589**	**0.667**	0.327	0.453	0.322	0.339	0.152	**0.795**	**0.791**	0.388	**0.598**	**0.936**	**0.713**	**0.526**	0.405	9
	ADC	0.440	0.450	**0.885**	**0.747**	**0.556**	0.343	0.449	0.420	0.392	0.194	**0.654**	**0.781**	0.424	**0.621**	**0.931**	**0.706**	**0.528**	0.386	9
	COForest	0.468	0.452	**0.858**	**0.732**	**0.552**	0.352	0.440	0.360	0.327	0.167	**0.733**	**0.788**	0.438	**0.644**	**0.898**	**0.706**	**0.538**	0.396	9

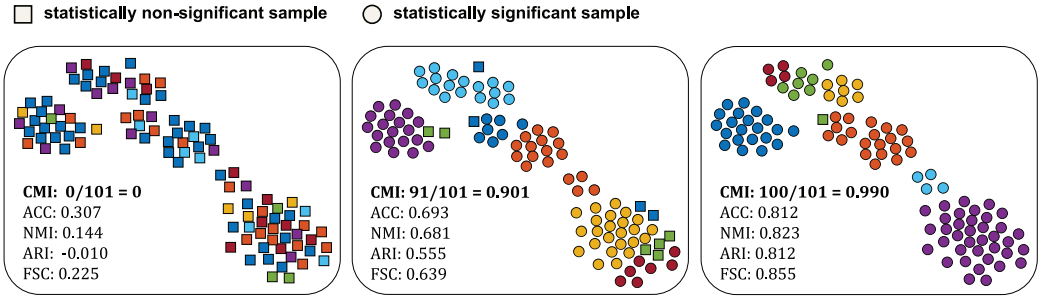


Fig. 5. Illustration of CMI values under a random partition and two partitions generated by clustering algorithms. Two-dimensional scatter plots (t-SNE on Zoo data set with Hamming distance) use color to indicate cluster labels and shape to indicate whether each sample's cluster membership is identified as being statistically significant.

- To further verify this cutoff threshold, we divide algorithm-derived partitions into two groups: those with CMI > 0.5, deemed high-quality, and those with CMI< 0.5, deemed low-quality. The distribution of external metric values for these two groups is shown in Fig. 6. It can be concluded that, in general, partitions with CMI > 0.5 exhibit significantly higher clustering quality compared to those with CMI< 0.5, as consistently supported by all external metrics.

To ensure consistency in subsequent analyses, all indices are interpreted in the same manner, with overall performance evaluated based on the cutoff threshold presented in the last column of Table 2. Based on experimental observations, CMI exhibits the following two intuitive characteristics, neither of which is simultaneously satisfied by all external metrics:

- For random partitions, all evaluated values are expected to be less than 0.5, often approaching 0. This holds for CMI, NMI, and ARI, whereas ACC and FSC indicate a tendency toward moderate clustering quality in such random assignments, with values exceeding 0.5 on 7 and 5 data sets, respectively, introducing ambiguity in interpretation.
- For algorithm-derived partitions, the evaluated values are generally expected to align with the majority of other indices (typically three or four) in identifying moderate-quality clustering. For example, in CMS partitions on Zo, when most other indices with values exceeding 0.5 indicate moderate-quality clustering, CMI also maintains consistency with a value greater than 0.5. This holds for CMI, ACC, and FSC, whereas NMI and ARI exhibit inconsistencies in certain cases. Typically, NMI conflicts with all other indices across all algorithm-derived partitions on Hv and CDCDR partitions on Bc, while ARI conflicts with all other indices in *k*-modes partitions on De and CMS partitions on Zo.

Moreover, with respect to the overall "#>0.5" counts across the seven partition groups in Table 2, CMI exhibits a high level of consistency with other indices. Specifically, CMI shares the same count with NMI and ARI in random partitions, FSC in *k*-modes partitions, ARI in CDCDR partitions, and ACC in CMS and COForest partitions. This motivates a further examination of the relationship between CMI values and other external metric values, as shown in Table 3.

To further highlight the strength of CMI in aligning with all these different external metrics, we additionally provide a comparison using three widely used internal evaluation metrics (Bai and Liang, 2015; Sulc

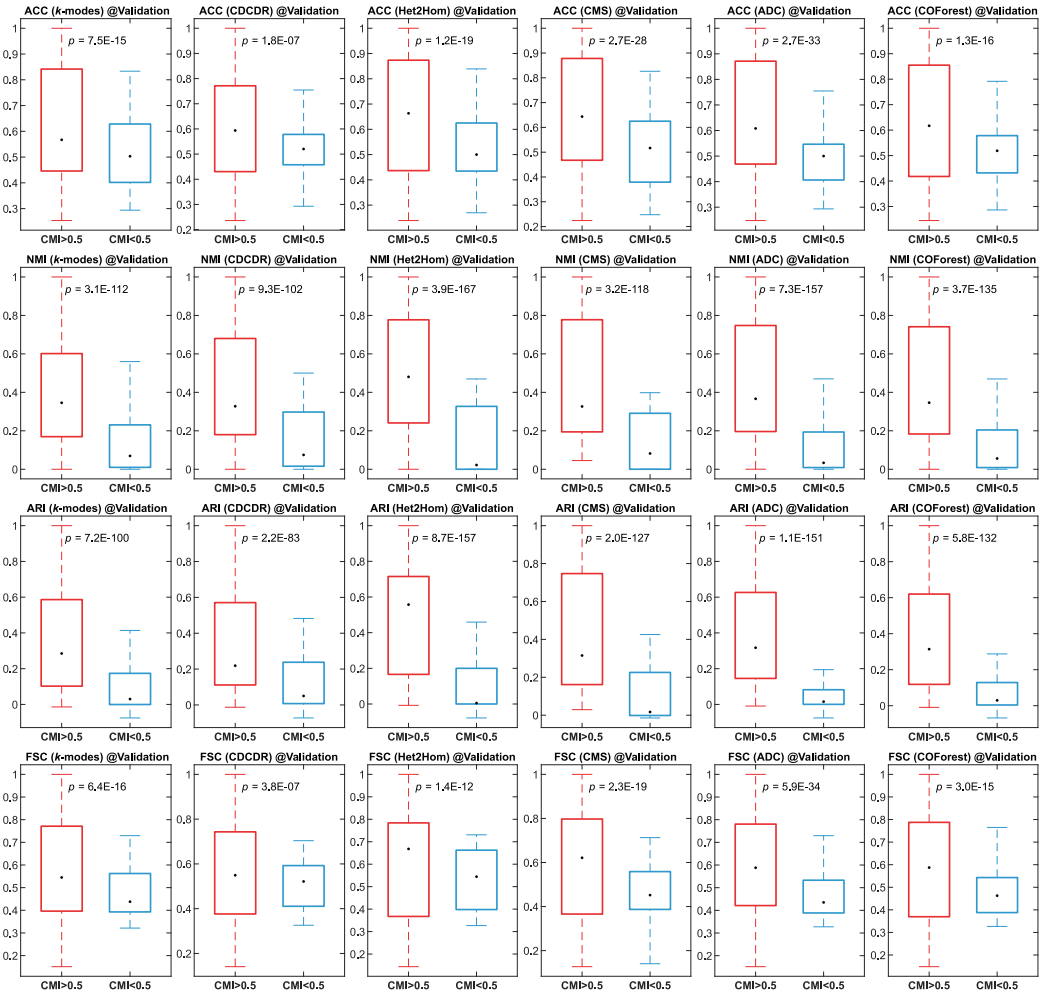


Fig. 6. Each boxplot contains 18 × 100 clustering results, where different metrics are represented in rows and different partition types in columns. The results in each boxplot are divided into two groups: one corresponding to clustering results with CMI > 0.5, and the other with CMI < 0.5. In all cases, the metrics in the former group tend to be significantly larger than those in the latter group, as determined by the Mann–Whitney U test (one-tailed).

Table 3
Comparison among CMI and three widely used internal evaluation metrics in terms of their correlation coefficients with external evaluation metrics. For ease of comparison, absolute values are taken for KMF, Entropy, and CU, which are negatively correlated with the external metrics. All CMI/KMF/EE/CU-to-ACC/NMI/ARI/FSC sample pairs are derived from 7 × 100 partitions, as presented in Table 2. The strongest correlation coefficient across the internal evaluation metrics is highlighted in bold, while statistically significant coefficients are underlined.

	Pearson				Spearman				Kendall			
	ACC	NMI	ARI	FSC	ACC	NMI	ARI	FSC	ACC	NMI	ARI	FSC
CMI	**0.44**	0.71	0.69	**0.40**	**0.46**	**0.77**	**0.80**	**0.43**	**0.31**	0.57	**0.59**	**0.28**
KMF	0.20	0.60	0.48	0.09	0.20	0.74	0.60	0.10	0.14	0.55	0.42	0.07
EE	0.13	0.59	0.44	0.03	0.13	0.69	0.53	0.04	0.10	0.53	0.37	0.03
CU	0.38	**0.80**	**0.73**	0.37	0.29	**0.77**	0.70	0.26	0.18	**0.61**	0.53	0.15

et al., 2024): the k-modes objective function (KMF) (Huang, 1998), Expected Entropy (EE) (Li et al., 2004), and Category Utility (CU) (Mirkin, 2001). These internal metrics are calculated based on their original formulations in the literature, with normalization adjustments applied. Specifically, KMF is divided by NM, while EE and CU are divided by M, to mitigate the influence of data scale and facilitate meaningful comparisons across data sets.

As shown in Table 3, the correlation coefficients indicate that CMI exhibits significant correlations with ACC, NMI, ARI, and FSC, whereas among the internal evaluation metrics, only CU is comparable to CMI. In contrast, KMF lacks a significant correlation with FSC, while EE shows no significant correlation with either ACC or FSC. Notably, CMI shows the highest overall consistency with all external metrics according to both Spearman's and Kendall's rank correlation coefficients.

4.4. Cluster refinement

To refine the algorithm-derived clustering results in Table 2, we focus on individual clusters that are already statistically valid, with a majority of samples in the cluster passing the FWER control. If none of the individual clusters in a given partition are valid, we do not perform any refinement operations on such partitions.

The ACC of refined clustering results, after removing n samples specified in a set $o = \{o_1, \ldots, o_n\}$, is calculated as $\mathrm{ACC} = \frac{\sum_{i=1}^{N} \delta(k_i, \mathrm{map}(k_i)) \cdot \mathbb{1}_{\{i \notin o\}}}{N-n}$, where $\mathbb{1}$ is an indicator function that equals 1 if the ith sample in

Table 4
Average external metrics for the refinement operation (@Refinement), based on input partitions generated by clustering algorithms. The Δ represents the percentage increase relative to the original value, with the original value shown in Table 2 serving as the baseline for comparison. Δ values exceeding 10% are marked in bold. The Δ value in the final column shows the overall performance improvement relative to the original mean of each metric across all 18 data sets.

@Refinement		Ls	Lc	So	Zo	Ps	Hr	Ly	Hd	Sf	Pt	De	Hv	Bs	Ca	Bc	Mm	Tt	Ce	Overall
k-modes	ACC	0.555	0.549	0.856	0.737	0.570	0.373	0.453	0.416	0.522	0.314	0.590	0.867	0.423	0.768	0.982	0.818	0.555	0.385	0.596
	Δ	0%	4.88%	0.50%	2.30%	1.34%	0.95%	2.25%	5.21%	9.24%	6.23%	0.08%	0.14%	0%	1.83%	8.61%	0%	1.15%	0%	2.39%
	NMI	0.248	0.250	0.841	0.761	0.035	0.009	0.124	0.172	0.344	0.347	0.573	0.462	0.016	0.212	0.858	0.326	0.015	0.069	0.315
	Δ	0%	**25.66%**	0.40%	1.49%	**36.01%**	**26.78%**	**10.87%**	**12.68%**	**18.85%**	2.44%	0.19%	0.72%	0%	1.19%	**52.33%**	0%	**27.91%**	0%	9.18%
	ARI	0.127	0.173	0.772	0.696	0.029	−0.010	0.087	0.175	0.277	0.110	0.452	0.537	0.016	0.284	0.927	0.403	0.019	0.047	0.284
	Δ	0%	**42.34%**	0.73%	3.73%	**36.55%**	−6.77%	**12.70%**	**17.00%**	**28.61%**	**18.41%**	0.23%	0.66%	0%	7.91%	**38.24%**	0%	**21.01%**	0%	**10.53%**
	FSC	0.477	0.475	0.836	0.762	0.521	0.355	0.417	0.423	0.439	0.200	0.568	0.775	0.414	0.691	0.968	0.704	0.537	0.396	0.553
	Δ	0%	**10.37%**	0.55%	2.87%	1.59%	0.82%	2.40%	6.67%	**13.39%**	**11.06%**	0.13%	0.24%	0%	7.03%	**12.89%**	0%	0.41%	0%	3.65%
CDCDR	ACC	0.471	0.571	0.903	0.707	0.614	0.488	0.497	0.474	0.447	0.319	0.723	0.861	0.449	0.639	0.984	0.798	0.559	0.393	0.605
	Δ	0%	6.08%	1.36%	2.58%	7.23%	0%	4.29%	**22.18%**	8.50%	**13.23%**	0.04%	3.51%	0%	1.52%	**23.99%**	0%	0%	0%	4.93%
	NMI	0.153	0.260	0.894	0.728	0.064	0.354	0.164	0.290	0.236	0.363	0.813	0.432	0.038	0.079	0.826	0.303	0.006	0.075	0.338
	Δ	0%	**26.63%**	0.90%	3.06%	**98.48%**	0%	9.18%	**70.85%**	9.82%	8.40%	0.01%	9.95%	0%	0.70%	**208.24%**	0%	0%	0%	**17.32%**
	ARI	0.041	0.177	0.855	0.627	0.061	0.282	0.163	0.261	0.147	0.119	0.689	0.511	0.043	0.107	0.919	0.372	0.013	0.038	0.301
	Δ	0%	**47.32%**	1.85%	4.56%	**117.11%**	0%	9.74%	**83.98%**	**17.14%**	**28.55%**	0.05%	**10.89%**	0%	1.82%	**188.41%**	0%	0%	0%	**21.69%**
	FSC	0.406	0.470	0.897	0.706	0.542	0.611	0.451	0.500	0.354	0.202	0.755	0.778	0.423	0.593	0.978	0.695	0.535	0.403	0.572
	Δ	0%	**11.11%**	1.36%	3.72%	2.99%	0%	4.20%	**31.35%**	8.83%	**19.78%**	0.04%	3.77%	0%	0.65%	**29.38%**	0%	0%	0%	5.69%
Het2Hom	ACC	0.533	0.560	1	0.720	0.502	0.333	0.517	0.401	0.474	0.321	0.750	0.877	0.467	0.829	0.979	0.818	0.556	0.368	0.611
	Δ	0%	0.50%	0%	2.33%	0%	0%	4.35%	**14.75%**	7.05%	**13.08%**	0%	0.40%	0%	0.27%	1.24%	0%	0.78%	0%	1.69%
	NMI	0.238	0.288	1	0.778	0.002	5.E−17	0.188	0.246	0.281	0.384	0.771	0.493	0.003	0.359	0.846	0.327	0.013	0.063	0.349
	Δ	0%	**15.59%**	0%	−1.35%	0%	0%	**17.85%**	**43.94%**	**12.62%**	**11.06%**	0%	2.54%	0%	0.53%	8.90%	0%	7.47%	0%	4.76%
	ARI	0.099	0.211	1	0.649	4.E−04	−0.015	0.198	0.214	0.212	0.150	0.706	0.568	7.E−04	0.447	0.917	0.404	0.015	0.030	0.323
	Δ	0%	**25.86%**	0%	−0.41%	0%	0%	**15.01%**	**68.59%**	**19.56%**	**29.64%**	0%	1.87%	0%	1.25%	5.26%	0%	**12.15%**	0%	5.18%
	FSC	0.439	0.530	1	0.722	0.662	0.327	0.467	0.431	0.367	0.217	0.762	0.789	0.597	0.736	0.963	0.703	0.536	0.394	0.591
	Δ	0%	9.64%	0%	0.77%	0%	0%	4.52%	**19.86%**	8.79%	**21.39%**	0%	0.66%	0%	0.84%	2.25%	0%	0.59%	0%	2.38%
CMS	ACC	0.563	0.625	1	0.664	0.844	0.341	0.507	0.326	0.465	0.367	0.811	0.880	0.381	0.718	0.977	0.826	0.517	0.400	0.623
	Δ	0%	7.29%	0%	4.94%	**10.09%**	0%	1.03%	2.81%	5.60%	**40.39%**	0%	0.17%	0%	6.95%	1.28%	0%	0%	0%	3.28%
	NMI	0.287	0.349	1	0.690	0.464	3.E−04	0.181	0.137	0.275	0.406	0.804	0.498	0.014	0.206	0.835	0.338	5.E−05	0.117	0.367
	Δ	0%	**50.70%**	0%	7.03%	**54.66%**	0%	−0.87%	8.35%	**17.60%**	**30.91%**	0%	1.57%	0%	**62.27%**	7.38%	0%	0%	0%	**10.30%**
	ARI	0.167	0.244	1	0.553	5.E−01	−0.015	0.193	0.098	0.212	0.174	0.745	0.575	0.015	0.253	0.908	0.426	−0.001	0.077	0.340
	Δ	0%	**45.57%**	0%	**11.37%**	**78.15%**	0%	0.76%	9.45%	**15.51%**	**93.39%**	0%	0.78%	0%	**68.21%**	5.42%	0%	0%	0%	**11.42%**
	FSC	0.473	0.520	1	0.639	0.765	0.327	0.455	0.331	0.367	0.244	0.795	0.793	0.388	0.667	0.958	0.713	0.526	0.405	0.576
	Δ	0%	**13.82%**	0%	8.51%	**14.55%**	0%	0.44%	2.78%	8.16%	**60.35%**	0%	0.29%	0%	**11.45%**	2.33%	0%	0%	0%	4.36%
ADC	ACC	0.526	0.518	0.898	0.741	0.659	0.382	0.512	0.441	0.516	0.346	0.640	0.873	0.442	0.672	0.973	0.820	0.552	0.369	0.604
	Δ	0%	1.94%	0.71%	2.81%	7.03%	0.02%	2.28%	4.24%	6.44%	**10.68%**	0.17%	0.16%	0%	3.08%	1.53%	0%	1.75%	0%	2.07%
	NMI	0.214	0.198	0.886	0.793	0.137	0.011	0.173	0.198	0.336	0.385	0.687	0.483	0.029	0.150	0.827	0.328	0.016	0.040	0.327
	Δ	0%	7.76%	0.12%	1.28%	**69.73%**	4.39%	**15.18%**	**10.78%**	**11.84%**	8.50%	−0.02%	1.60%	0%	**16.29%**	**10.68%**	0%	**76.75%**	0%	5.46%
	ARI	0.077	0.114	0.842	0.700	0.157	−0.003	0.155	0.212	0.266	0.151	0.555	0.555	0.028	0.178	0.902	0.410	0.023	0.026	0.297
	Δ	0%	**12.62%**	0.63%	2.80%	**91.96%**	5.41%	**12.05%**	**12.95%**	**16.50%**	**21.17%**	0.22%	0.77%	0%	**15.54%**	6.59%	0%	**96.43%**	0%	6.25%
	FSC	0.440	0.470	0.890	0.764	0.595	0.343	0.463	0.444	0.427	0.226	0.655	0.783	0.424	0.645	0.956	0.706	0.535	0.386	0.564
	Δ	0%	4.43%	0.51%	2.36%	6.89%	0.12%	3.12%	5.76%	9.07%	**16.08%**	0.18%	0.29%	0%	3.81%	2.73%	0%	1.24%	0%	2.47%
COForest	ACC	0.554	0.579	0.839	0.718	0.665	0.408	0.518	0.438	0.460	0.300	0.709	0.880	0.456	0.724	0.940	0.809	0.567	0.383	0.608
	Δ	0%	5.63%	0.35%	1.65%	6.32%	0.15%	3.89%	**18.96%**	**10.35%**	**8.21%**	0.07%	0.43%	0%	1.46%	2.07%	0%	0.54%	0%	2.60%
	NMI	0.283	0.302	0.876	0.776	0.145	0.029	0.168	0.235	0.285	0.348	0.789	0.500	0.034	0.215	0.761	0.320	0.010	0.095	0.343
	Δ	0%	**21.66%**	0.07%	0.32%	**71.35%**	−0.17%	**29.16%**	**73.49%**	**23.90%**	7.42%	0.00%	3.67%	0%	**1.89%**	**13.91%**	0%	**20.13%**	0%	7.86%
	ARI	0.139	0.203	0.806	0.672	0.167	0.016	0.148	0.221	0.187	0.109	0.660	0.577	0.034	0.274	0.828	0.399	0.019	0.055	0.306
	Δ	0%	**38.66%**	0.32%	0.93%	**82.35%**	0.03%	**21.94%**	**89.90%**	**39.96%**	**22.01%**	0.10%	2.02%	0%	**9.73%**	9.07%	0%	**22.02%**	0%	8.96%
	FSC	0.468	0.501	0.860	0.739	0.595	0.353	0.464	0.461	0.369	0.191	0.734	0.794	0.438	0.670	0.928	0.706	0.541	0.396	0.567
	Δ	0%	**10.70%**	0.28%	1.02%	7.80%	0.27%	5.42%	**27.96%**	**12.97%**	**14.44%**	0.07%	0.72%	0%	4.05%	3.40%	0%	0.44%	0%	3.64%

D is not in the outlier set and 0 otherwise. This formula ensures that only the samples not identified as outliers contribute to the ACC calculation. Other external metrics, including NMI, ARI, and FSC, are computed in the same manner based on the remaining samples and their corresponding cluster labels after outlier removal.

The refinement performance, expressed in terms of the average ACC, NMI, ARI, and FSC over 6 × 100 clustering results for each data set, grouped by the *k*-modes, CDCDR, Het2Hom, CMS, ADC, and COForest algorithms, is shown in Table 4, along with the percentage improvement compared to the original metric values. Additionally, we record the frequency of metric improvement or deterioration in Fig. 7, where each row of subfigures represents the results for a specific partition group. The observed refinement performance is summarized as follows:

- **Overall performance**: As shown in the last column of Table 4, except for the relatively minor improvements in Het2Hom, ADC, and COForest partitions, other partition groups achieved an improvement of over 10% in at least one metric. Specifically, *k*-modes partitions showed a 10.5% increase in ARI, CDCDR partitions were improved by 17.3% in NMI and 21.7% in ARI, while CMS partitions exhibited gains of 10.3% in NMI and 11.4% in ARI. Furthermore, as depicted in Fig. 7, after the refinement operation, the proportion of cases with improved metrics exceeded those with deteriorated metrics across almost all partitions and data sets. Even in Het2Hom partitions, where NMI and ARI decreased in the Zo data set, the corresponding stacked bars for ACC and FSC still showed absolute improvements, likely due to the potential bias of different metrics.
- **Analysis of 0% improvement cases**: This phenomenon occurs when no outliers are detected within any individual cluster of a given partition, preventing refinement operations from being performed. There are two distinct scenarios: (i) our refinement operation is only applied to statistically valid clusters, excluding partitions where all clusters are invalid; (ii) the partition consists entirely of compact, well-formed clusters with no outliers. This is reflected in the CMI values shown in Table 2. For example, in the CMS partition group, which follows the same interpretation as other partition groups, 0% improvement cases fall into two categories: (i) results observed in data sets Ls, Hr, Bs, Mm, Tt, and Ce, where all clusters are invalid, with CMI values being low or near zero; (ii) results observed in data sets So and De, where no outliers exist in any individual cluster, have CMI = 1. Notably, in these latter outlier-free cases, external metrics also indicate high clustering quality.

In summary, our refinement operation is generally effective in improving cluster quality when a partition contains valid but not perfectly compact clusters, allowing for adjustments based on sufficient clustering structure information.

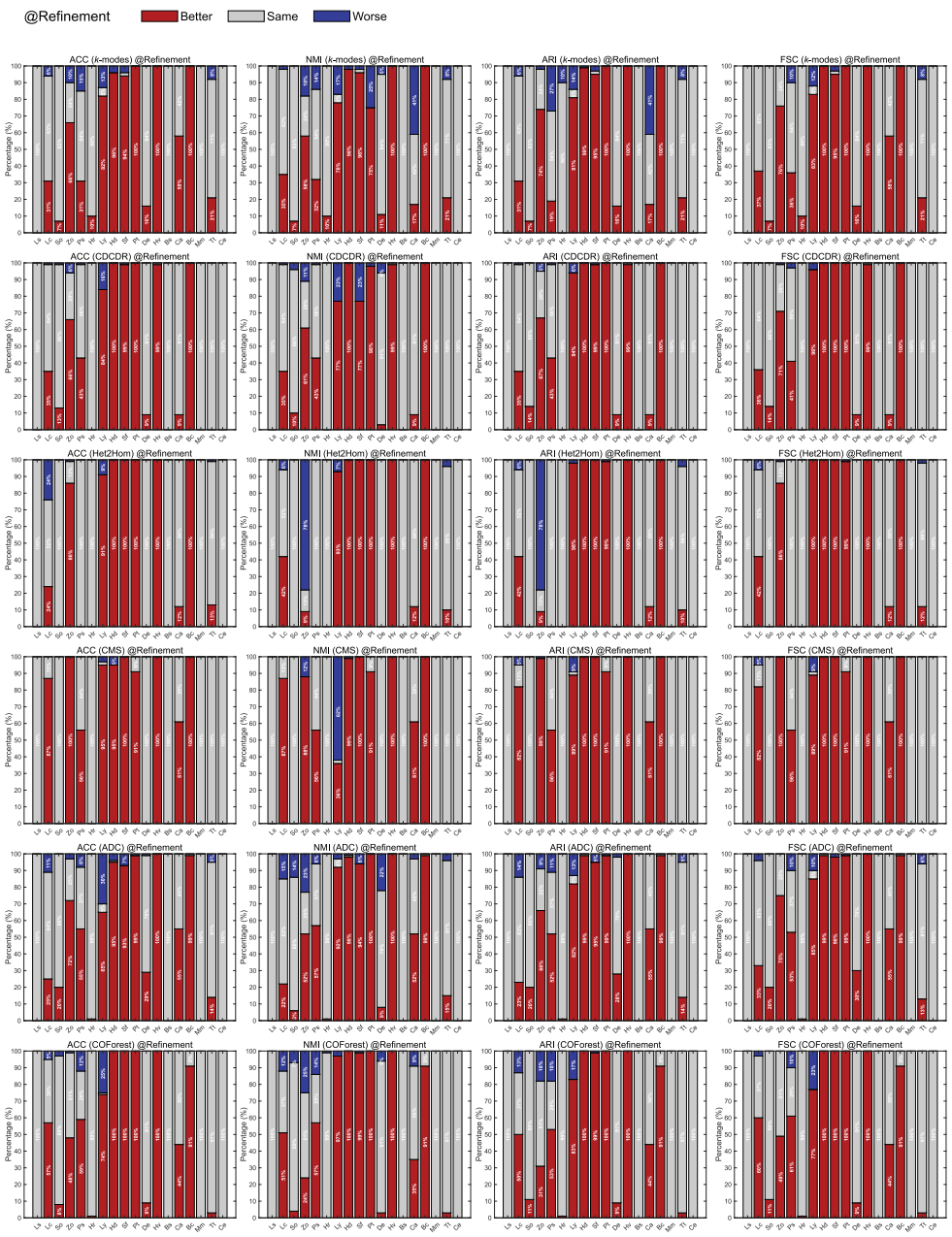


Fig. 7. The stacked barplots depicting the detailed performance of cluster refinement for each data set. These plots illustrate the percentage of the 100 clustering results per data set, categorized based on whether the external metric is better than, worse than, or the same as the original one after removing outliers.

4.5. Cluster enhancement

Another approach we explore to improve the quality of the given optimized partitions involves reassigning each sample to the target cluster with the minimal cluster membership p-value, provided that this p-value passes the FWER control for the target cluster.

This refinement operation differs from the previous refinement approach in several aspects: (i) it retains all samples in the original data set without removing any, even if some are identified as outliers for all clusters. In such cases, the sample maintains its original cluster membership without reassignment, leading to 0% improvement cases; (ii) Even if a sample has a statistically significant p-value for its current cluster membership, it will still be reassigned to a target cluster if a smaller cluster membership p-value exists. In this regard, this approach takes an additional step forward to potentially improve clustering quality.

As the enhancement operation is not well-built and simply determines the optimal assignment for all samples before reassigning them simultaneously, this naïve one-step reassignment approach inevitably carries a risk of deteriorating clustering quality. This suggests that the reassignment of a single sample can influence the entire partition and impact the assignments of other samples, similar to how clustering algorithms typically adopt an iterative optimization strategy based on sample assignments from the previous round. In some of the following clustering quality deterioration cases, this issue fundamentally arises from the limitations of our indicator-based approach, which functions as a post-clustering method and cannot replace optimization-based clustering algorithms.

The performance gain after the cluster enhancement operation is presented in Table 5 in the same format as Table 4. Since all samples are retained and external metrics are computed on the same data set before and after enhancement, we explicitly highlight the metrics

Table 5
Average external metrics for the enhancement operation (@Enhancement), with the same display format and Δ calculation method as in Table 4 for improvement comparison. Enhanced clustering results on 18 data sets for metrics showing significant improvement over their original values in Table 2 are underlined (marked on the metric names). This is verified by the Wilcoxon signed rank test (one-tailed) with a 95% confidence interval. Metrics that decline by more than 5% from their original values are marked in blue.

@Enhancement		Ls	Lc	So	Zo	Ps	Hr	Ly	Hd	Sf	Pt	De	Hv	Bs	Ca	Bc	Mm	Tt	Ce	Overall
k-modes	ACC	0.555	0.530	0.890	0.779	0.562	0.370	0.473	0.464	0.487	0.286	0.712	0.874	0.423	0.752	0.958	0.818	0.550	0.385	0.604
	Δ	0%	1.25%	4.55%	8.05%	−0.03%	0.14%	6.61%	**17.40%**	1.89%	−3.19%	**20.85%**	0.95%	0%	−0.34%	5.95%	0%	0.23%	−0.01%	3.67%
	NMI	0.248	0.210	0.887	0.734	0.026	0.008	0.136	0.187	0.274	0.226	0.657	0.464	0.016	0.215	0.731	0.326	0.012	0.069	0.301
	Δ	0%	5.54%	5.84%	−2.01%	0.08%	5.31%	**21.65%**	**22.14%**	**−5.42%**	**−33.13%**	**14.84%**	1.11%	0%	2.35%	**29.72%**	0%	3.37%	0.01%	4.59%
	ARI	0.127	0.136	0.850	0.744	0.021	−0.009	0.087	0.215	0.202	0.060	0.645	0.558	0.016	0.258	0.836	0.403	0.017	0.047	0.290
	Δ	0%	**11.84%**	**10.81%**	**10.76%**	0.08%	1.25%	**13.17%**	**44.18%**	**−5.94%**	**−35.04%**	**43.00%**	4.53%	0%	−2.17%	**24.72%**	0%	4.61%	−0.11%	**12.50%**
	FSC	0.477	0.441	0.894	0.804	0.512	0.351	0.436	0.473	0.395	0.198	0.726	0.785	0.414	0.635	0.926	0.704	0.536	0.396	0.561
	Δ	0%	2.33%	7.51%	8.61%	−0.06%	−0.36%	7.05%	**19.33%**	2.14%	9.81%	**28.03%**	1.58%	0%	−1.77%	7.99%	0%	0.10%	−0.02%	5.15%
CDCDR	ACC	0.471	0.537	0.921	0.742	0.575	0.488	0.513	0.507	0.426	0.282	0.785	0.853	0.449	0.641	0.862	0.800	0.565	0.393	0.601
	Δ	0%	−0.17%	3.32%	7.67%	0.35%	0%	7.69%	**30.76%**	3.44%	0.17%	8.55%	2.55%	0%	1.90%	8.59%	0.24%	0.97%	0%	4.09%
	NMI	0.153	0.206	0.894	0.701	0.033	0.354	0.159	0.228	0.210	0.221	0.784	0.430	0.038	0.105	0.434	0.304	0.008	0.075	0.297
	Δ	0%	0.11%	0.96%	−0.79%	1.94%	0%	6.04%	**34.03%**	−2.38%	**−33.94%**	−3.45%	9.47%	0%	**33.99%**	**62.16%**	0.42%	**37.52%**	0%	3.05%
	ARI	0.041	0.121	0.872	0.671	0.029	0.282	0.158	0.264	0.135	0.065	0.744	0.511	0.043	0.130	0.516	0.372	0.017	0.038	0.278
	Δ	0%	0.76%	3.83%	**11.87%**	3.03%	0%	6.89%	**86.61%**	8.16%	**−30.15%**	8.06%	**10.94%**	0%	**23.87%**	**62.02%**	0.25%	**28.46%**	0%	**12.42%**
	FSC	0.406	0.425	0.910	0.749	0.525	0.611	0.459	0.522	0.365	0.203	0.802	0.773	0.423	0.598	0.811	0.694	0.536	0.403	0.568
	Δ	0%	0.35%	2.81%	**10.08%**	−0.15%	0%	6.01%	**37.07%**	**12.41%**	**20.73%**	6.23%	3.08%	0%	1.62%	7.29%	−0.09%	0.18%	0%	4.84%
Het2Hom	ACC	0.533	0.557	1	0.706	0.503	0.333	0.512	0.456	0.472	0.303	0.813	0.874	0.467	0.821	0.966	0.818	0.553	0.368	0.614
	Δ	0%	0%	0%	0.38%	0.30%	0%	3.41%	**30.69%**	6.52%	6.55%	8.43%	0%	0%	−0.67%	−0.15%	0%	0.15%	0%	2.15%
	NMI	0.238	0.249	1	0.769	0.003	5.E−17	0.170	0.215	0.288	0.259	0.799	0.475	0.003	0.355	0.770	0.327	0.012	0.063	0.333
	Δ	0%	0.13%	0%	−2.55%	**53.60%**	0%	6.66%	**25.99%**	**15.48%**	**−25.04%**	3.64%	−1.03%	0%	−0.68%	−0.95%	0%	1.81%	0%	0.03%
	ARI	0.099	0.168	1	0.650	0.002	−0.015	0.180	0.245	0.196	0.076	0.770	0.557	7.E−04	0.426	0.866	0.404	0.014	0.030	0.315
	Δ	0%	0.31%	0%	−0.29%	**547%**	0%	4.75%	**93.53%**	**10.47%**	**−34.32%**	9.01%	−3.84E−05	0%	−3.34%	−0.62%	0%	3.90%	0%	2.72%
	FSC	0.439	0.484	1	0.724	0.662	0.327	0.460	0.481	0.364	0.203	0.816	0.784	0.597	0.723	0.939	0.703	0.534	0.394	0.591
	Δ	0%	0.10%	0%	0.95%	−0.02%	0%	2.82%	**33.85%**	7.95%	**13.55%**	7.02%	0.01%	0%	−0.95%	−0.26%	0%	0.05%	0%	2.28%
CMS	ACC	0.563	0.600	1	0.784	0.766	0.341	0.511	0.385	0.489	0.274	0.856	0.874	0.381	0.672	0.966	0.826	0.517	0.400	0.623
	Δ	0%	2.90%	0%	**23.83%**	0%	0%	1.87%	**21.69%**	**10.88%**	4.85%	5.54%	−0.52%	0%	0.06%	0.15%	0%	0%	0%	3.21%
	NMI	0.287	0.342	1	0.715	0.263	3.E−04	0.182	0.154	0.284	0.217	0.855	0.475	0.014	0.129	0.780	0.338	5.E−05	0.117	0.342
	Δ	0%	**47.56%**	0%	**10.84%**	**−12.17%**	0%	−0.40%	**21.23%**	**21.38%**	**−29.86%**	6.34%	−3.06%	0%	2.11%	0.30%	0%	0%	0%	2.81%
	ARI	0.167	0.263	1	0.727	0.279	−0.015	0.193	0.145	0.206	0.059	0.843	0.557	0.015	0.150	0.866	0.426	−0.001	0.077	0.331
	Δ	0%	**56.52%**	0%	**46.38%**	−0.14%	0%	0.65%	**61.35%**	**12.12%**	**−33.84%**	**13.22%**	−2.43%	0%	−0.60%	0.62%	0%	0%	0%	8.41%
	FSC	0.473	0.549	1	0.793	0.658	0.327	0.458	0.385	0.384	0.194	0.878	0.784	0.388	0.598	0.939	0.713	0.526	0.405	0.581
	Δ	0%	**20.30%**	0%	**34.67%**	−1.38%	0%	1.11%	**19.46%**	**13.09%**	**27.76%**	**10.32%**	−0.85%	0%	0.04%	0.28%	0%	0%	0%	5.24%
ADC	ACC	0.526	0.511	0.912	0.801	0.618	0.382	0.520	0.469	0.495	0.292	0.715	0.874	0.442	0.662	0.963	0.820	0.543	0.369	0.606
	Δ	0%	0.68%	2.29%	**11.14%**	0.41%	0%	3.84%	**10.89%**	2.11%	**−6.74%**	**12.04%**	0.26%	0%	1.57%	0.47%	0%	0.03%	−0.02%	2.39%
	NMI	0.214	0.189	0.888	0.755	0.084	0.010	0.169	0.200	0.285	0.234	0.695	0.475	0.029	0.142	0.758	0.328	0.009	0.039	0.306
	Δ	0%	2.74%	0.30%	−3.61%	3.89%	−1.49%	**12.12%**	**12.16%**	**−5.25%**	**−34.14%**	1.22%	0.00%	0%	**10.44%**	1.37%	0%	−1.73%	−0.12%	−1.48%
	ARI	0.077	0.108	0.856	0.796	0.086	−0.004	0.142	0.236	0.217	0.063	0.642	0.557	0.028	0.160	0.856	0.410	0.012	0.026	0.293
	Δ	0%	6.67%	2.30%	**16.99%**	5.12%	−4.20%	2.61%	**25.68%**	−4.89%	**−49.38%**	**15.83%**	1.24%	0%	4.29%	1.10%	0%	1.75%	−0.33%	4.68%
	FSC	0.440	0.455	0.900	0.846	0.556	0.342	0.471	0.479	0.407	0.199	0.726	0.784	0.424	0.618	0.935	0.706	0.528	0.386	0.567
	Δ	0%	0.94%	1.68%	**13.31%**	−0.06%	−0.06%	4.91%	**13.93%**	3.76%	2.60%	**10.96%**	0.45%	0%	−0.46%	0.45%	0%	0.04%	−0.04%	2.98%
COForest	ACC	0.554	0.549	0.855	0.768	0.628	0.408	0.527	0.451	0.466	0.289	0.774	0.871	0.456	0.698	0.943	0.809	0.570	0.383	0.611
	Δ	0%	0.11%	2.34%	8.66%	0.30%	0.15%	5.62%	**22.59%**	**11.76%**	4.21%	9.29%	−0.59%	−0.01%	−2.21%	2.37%	0%	1.08%	−0.13%	3.07%
	NMI	0.283	0.249	0.871	0.750	0.086	0.029	0.155	0.196	0.239	0.227	0.768	0.466	0.034	0.185	0.699	0.320	0.011	0.094	0.315
	Δ	0%	0.48%	−0.54%	−2.97%	1.52%	−0.72%	**18.89%**	**44.90%**	4.03%	**−29.88%**	−2.60%	−3.31%	0.07%	**−12.33%**	4.60%	0%	**32.91%**	−0.24%	−1.00%
	ARI	0.139	0.148	0.819	0.740	0.093	0.016	0.146	0.226	0.166	0.068	0.728	0.550	0.034	0.206	0.797	0.399	0.020	0.055	0.297
	Δ	0%	1.01%	1.93%	**11.10%**	2.12%	−0.73%	**20.15%**	**94.15%**	**24.12%**	**−23.59%**	**10.45%**	−2.75%	0.01%	**−17.48%**	4.98%	0%	**30.86%**	−0.17%	5.73%
	FSC	0.468	0.453	0.870	0.800	0.552	0.352	0.471	0.474	0.365	0.202	0.790	0.781	0.438	0.623	0.911	0.706	0.539	0.396	0.566
	Δ	0%	0.28%	1.44%	9.30%	0.02%	−0.04%	7.02%	**31.47%**	**11.71%**	**21.07%**	7.78%	−0.97%	0.00%	−3.24%	1.51%	0%	0.18%	−0.09%	3.47%

with statistically significant improvements. The observed enhancement performance is summarized as follows:

- **Overall performance**: As indicated by the underlined metrics in Table 5, significant improvements are observed in ACC and FSC across all partition groups. Additionally, strong evidence of ARI gains, which also achieve a 10% overall improvement, is specifically observed in the *k*-modes and CDCDR partition groups. Therefore, applying our enhancement operation to these two algorithms is recommended to further improve clustering quality. Although Het2Hom and CMS, as representative SOTA clustering algorithms, have already achieved higher clustering performance than classical and many other methods, our enhancement operation still yields noticeable improvements in certain data sets, such as Het2Hom partitions on Ps, Hd, and Sf, and CMS partitions on Lc, Zo, Hd, Sf, and De. Compared to the refinement operation, which typically failed to refine cases such as Het2Hom partitions on Ps and CMS partitions on De (0% improvement), as shown in Table 4, the enhancement operation can further improve cluster quality in these outlier-free cases by reassigning samples to target clusters with smaller *p*-values.
- **Analysis of deterioration cases**: Some enhancement results exhibit metric declines, which are marked in Table 5 where the decrease exceeds 5% to exclude trivial deviations. In each partition where deterioration occurs, at most two metrics show a noticeable decline, primarily in NMI and ARI, while other metrics, such as FSC, still display notable improvements in different partition groups of Pt. Considering metric biases and potential conflicts among different metrics, these results cannot be deemed complete failures as long as not all metrics indicate deterioration. The only clear failures are observed in CMS partitions on Ps, ADC partitions on Pt, and COForest partitions on Ca, in contrast to the refinement operation, which successfully improved all metrics on these partitions, as shown in Table 4. This highlights the crucial role of outliers, as their disruptive impact on cluster reassignment often makes removal more effective than reassignment, especially in our non-iterative, non-optimized enhancement operation. Moreover, the enhancement operation carries a higher risk of deterioration in partitions of Pt (which contains 21 clusters), as the large number of clusters increases the cumulative likelihood of reassignment errors, with both individual and multiple samples having more possible cluster membership assignments. In contrast, Hd (5 clusters) and Sf (6 clusters) exhibit the most similar characteristics to Pt, where partitions with CMI indicate high quality while nearly all external metrics suggest low quality, as evidenced in Table 2. However, enhanced partitions of these two data sets show much better performance with fewer declining metrics. Notably, Hd, with fewer clusters than the other two, allows the enhancement operation to achieve the most substantial quality improvements across all partition groups.

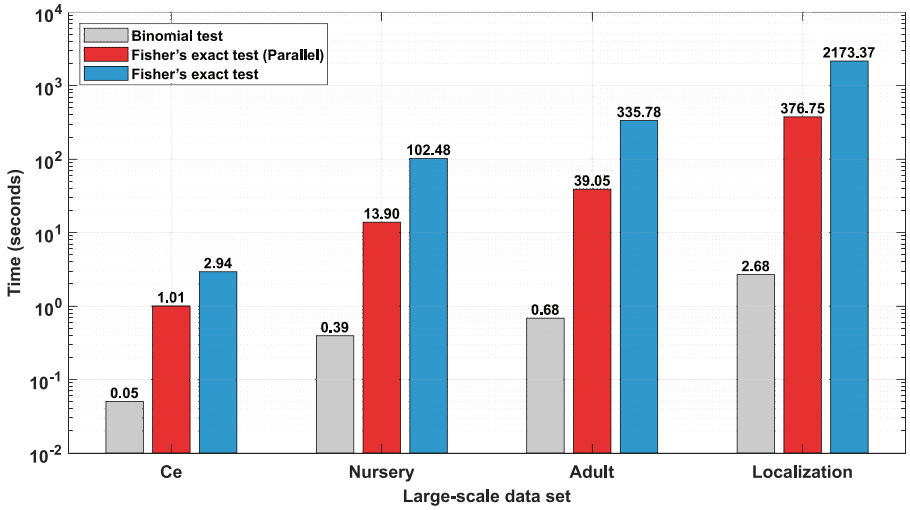


Fig. 8. The execution time of SigCM's two steps, Fisher's exact test step and the Binomial test step, on large-scale categorical data sets using an Intel i7-12700K@3.60 GHz personal computer. All samples in each data set were processed based on their ground-truth assignments. For the parallel execution of Fisher's exact test step, 12 workers were utilized, managed by MATLAB 2023b. The data sets used include Nursery, Adult, and Localization, with N equal to 12960, 32561, and 164860, and M equal to 8, 8, and 2, respectively.

4.6. Time efficiency on large-scale data

Finally, to evaluate the running efficiency of our SigCM-based applications and provide guidance on the expected and affordable execution time for different data scales, we executed the SigCM algorithm on all samples in each data set based on the ground-truth partition. This involved running the SigCM algorithm once for each of the N samples in a data set, resulting in an overall time complexity of $\mathcal{O}(N^2M)$. The actual runtime for the large-scale data sets, including the largest Ce data set used in former experiments and three newly added data sets, is presented in Fig. 8.

From the observed runtime, the Binomial test step in SigCM is time-efficient, whereas the Fisher's exact test step becomes increasingly time-consuming as data scale grows, leading to a runtime exceeding 2000 s when $N = 164860$. Since SigCM runs independently for each sample without computational dependencies, parallel computing serves as a practical approach to reduce runtime, as shown by the red bars in Fig. 8. A more fundamental approach for improving the running efficiency of Fisher's exact test step is to approximate p-values using an upper bound for Eq. (2), utilizing a formula proposed in Hämäläinen (2016). This method reduces computational complexity by considering fewer terms in the calculation, providing a computationally efficient alternative to the original cumulative calculation, albeit with some loss of precision.

5. Conclusions

To ascertain the potential true cluster labels for each categorical sample in an unsupervised manner, we have proposed a novel algorithm for assessing cluster membership. This algorithm leverages hypothesis testing techniques, specifically Fisher's exact test and meta-analysis, to provide a statistical significance assessment of cluster memberships. The analytical p-value derived is then applied to improve the quality of clustering results via refining and enhancing individual clusters.

Although our experimental findings highlight the effectiveness of our method, there is potential for further advancement: (1) Theoretically, we currently assume attribute independence for a straightforward meta-analysis approach to derive the final p-value. This assumption generally does not align with practical scenarios. Exploring more comprehensive meta-analysis methods that account for attribute dependencies is an essential issue for future exploration. (2) The use of p-values for refinement and enhancement can be further improved. For example, integrating the two processes into a unified strategy or designing more sophisticated approaches, such as multi-step heuristic reassignments constrained by specific rules, may yield more robust improvements in cluster quality. Additionally, we aim to develop a new clustering algorithm that incorporates accurately evaluated cluster membership p-values as a guideline for discovering meaningful clusters. (3) Given the limited availability of outlier detection methods for categorical data (Cabral et al., 2025), and the fact that our cluster membership evaluation method defines outliers relative to reference clusters, it cannot yet serve as a standalone outlier detection method. Extending the p-value-based approach for this purpose is a promising direction for future research.

CRediT authorship contribution statement

Lianyu Hu: Writing – original draft, Visualization, Software, Methodology. **Zerun Li:** Data curation. **Junjie Dong:** Investigation. **Mudi Jiang:** Validation. **Zengyou He:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been supported by the Natural Science Foundation of China under Grant No. 62472064.

Data availability

Data will be made available on request.

References

Bai, L., Liang, J., 2015. Cluster validity functions for categorical data: a solution-space perspective. Data Min. Knowl. Discov. 29 (6), 1560–1597.

Bai, L., Liang, J., 2022. A categorical data clustering framework on graph representation. Pattern Recognit. 128, 108694.

Bandyapadhyay, S., Fomin, F.V., Golovach, P.A., Simonov, K., 2023. Parameterized complexity of feature selection for categorical data clustering. ACM Trans. Comput. Theory 15 (3–4), 1–24.

Cabral, E.F., Sánchez Vinces, B.V., Silva, G.D., Sander, J., Cordeiro, R.L., 2025. Efficient outlier detection in numerical and categorical data. Data Min. Knowl. Discov. 39 (3), 1–46.

Cai, D., He, X., Han, J., 2005. Document clustering using locality preserving indexing. IEEE Trans. Knowl. Data Eng. 17 (12), 1624–1637.

Chang, L.-C., Lin, H.-M., Sibille, E., Tseng, G.C., 2013. Meta-analysis methods for combining multiple expression profiles: comparisons, statistical characterization and an application guideline. BMC Bioinform. 14 (1), 1–15.

Chung, N.C., 2020. Statistical significance of cluster membership for unsupervised evaluation of cell identities. Bioinformatics 36 (10), 3107–3114.

Cinar, O., Viechtbauer, W., 2022. The poolr package for combining independent and dependent p values. J. Stat. Softw. 101, 1–42.

Cui, X., Dickhaus, T., Ding, Y., Hsu, J.C., 2021. Handbook of Multiple Comparisons. CRC Press.

Dinh, T., Wong, H., Fournier-Viger, P., Lisik, D., Ha, M.-Q., Dam, H.-C., Huynh, V.-N., 2025. Categorical data clustering: 25 years beyond K-modes. Expert Syst. Appl. 272, 126608.

Dua, D., Graff, C., 2019. UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences.

Dzemski, A., Okui, R., 2024. Confidence set for group membership. Quant. Econ. 15 (2), 245–277.

Ezugwu, A.E., Ikotun, A.M., Oyelade, O.O., Abualigah, L., Agushaka, J.O., Eke, C.I., Akinyelu, A.A., 2022. A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. Eng. Appl. Artif. Intell. 110, 104743.

Hämäläinen, W., 2016. New upper bounds for tight and fast approximation of Fisher's exact test in dependency rule mining. Comput. Statist. Data Anal. 93, 469–482.

Hu, L., Dong, J., Jiang, M., Liu, Y., He, Z., 2025a. Clusterability test for categorical data. Knowl. Inf. Syst. 67 (5), 4113–4138.

Hu, L., Jiang, M., Dong, J., Liu, X., He, Z., 2025b. Interpretable categorical data clustering via hypothesis testing. Pattern Recognit. 162, 111364.

Hu, L., Jiang, M., Liu, X., He, Z., 2025c. Significance-based decision tree for interpretable categorical data clustering. Inform. Sci. 690, 121588.

Hu, L., Jiang, M., Liu, Y., Zou, Q., He, Z., 2025d. Clustering categorical data via multiple hypothesis testing. ACM Trans. Knowl. Discov. Data 19 (5), 109. http://dx.doi.org/10.1145/3735977.

Huang, Z., 1998. Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min. Knowl. Discov. 2 (3), 283–304.

Jain, A.K., 2010. Data clustering: 50 years beyond K-means. Pattern Recognit. Lett. 31 (8), 651–666.

Jian, S., Cao, L., Lu, K., Gao, H., 2018. Unsupervised coupled metric similarity for non-iid categorical data. IEEE Trans. Knowl. Data Eng. 30 (9), 1810–1823.

Jian, S., Pang, G., Cao, L., Lu, K., Gao, H., 2019. CURE: Flexible categorical data representation by hierarchical coupling learning. IEEE Trans. Knowl. Data Eng. 31 (5), 853–866.

Lee, S.-H., Jeong, Y.-S., Kim, J.-Y., Jeong, M.K., 2018. A new clustering validity index for arbitrary shape of clusters. Pattern Recognit. Lett. 112, 263–269.

Li, T., Ma, S., Ogihara, M., 2004. Entropy-based criterion in categorical clustering. In: Proceedings of the Twenty-First International Conference on Machine Learning. p. 68.

Li, Z., Zhu, Y., Van Leeuwen, M., 2023. A survey on explainable anomaly detection. ACM Trans. Knowl. Discov. Data 18 (1), 1–54.

Ling, Z., Wu, J., Zhang, Y., Zhou, P., Wu, X., Yu, K., Wu, X., 2025. Label-aware causal feature selection. IEEE Trans. Knowl. Data Eng. 37 (3), 1268–1281.

Liu, Y., Li, Z., Xiong, H., Gao, X., Wu, J., Wu, S., 2013. Understanding and enhancement of internal clustering validation measures. IEEE Trans. Cybern. 43 (3), 982–994.

Mirkin, B., 2001. Reinterpreting the category utility function. Mach. Learn. 45 (2), 219–228.

Noble, W.S., 2009. How does multiple testing correction work? Nature Biotechnol. 27 (12), 1135–1137.

Park, K., Choe, Y.J., Jiang, Y., Veitch, V., 2024. The geometry of categorical and hierarchical concepts in large language models. In: ICML 2024 Workshop on Theoretical Foundations of Foundation Models.

Qian, Y., Li, F., Liang, J., Liu, B., Dang, C., 2016. Space structure and clustering of categorical data. IEEE Trans. Neural Netw. Learn. Syst. 27 (10), 2047–2059.

Rezaei, M., Fränti, P., 2016. Set matching measures for external cluster validity. IEEE Trans. Knowl. Data Eng. 28 (8), 2173–2186.

Song, Y., Liu, J., Zhang, J., 2025. Attribute grouping-based categorical outlier detection using causal coupling weight. Complex Intell. Syst. 11 (6), 240.

Sulc, Z., Hornicek, J., Rezankova, H., Cibulkova, J., 2024. Comparison of internal evaluation criteria in hierarchical clustering of categorical data. Adv. Data Anal. Classif.

Yuan, K., Miao, D., Pedrycz, W., Ding, W., Zhang, H., 2024. Ze-HFS: Zentropy-based uncertainty measure for heterogeneous feature selection and knowledge discovery. IEEE Trans. Knowl. Data Eng. 36 (11), 7326–7339.

Žalik, K.R., Žalik, B., 2011. Validity index for clusters of different sizes and densities. Pattern Recognit. Lett. 32 (2), 221–234.

Zhang, C., Chen, L., Zhao, Y.-P., Wang, Y., Chen, C.L.P., 2023. Graph enhanced fuzzy clustering for categorical data using a Bayesian dissimilarity measure. IEEE Trans. Fuzzy Syst. 31 (3), 810–824.

Zhang, Y., Cheung, Y.-m., 2022. Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes. IEEE Trans. Pattern Anal. Mach. Intell. 44 (7), 3560–3576.

Zhang, Y., Cheung, Y.-M., 2023. Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data. IEEE Trans. Neural Netw. Learn. Syst. 34 (9), 6530–6544.

Zhang, Y., Cheung, Y.-m., Zeng, A., 2022. Het2Hom: Representation of heterogeneous attributes into homogeneous concept spaces for categorical-and-numerical-attribute data clustering. In: Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence. pp. 3758–3765.

Zhang, R., Zhang, H., Qian, Y., 2025. Three-way space structure and clustering of categorical data. Internat. J. Approx. Reason. 184, 109457.

Zhao, M., Feng, S., Zhang, Y., Li, M., Lu, Y., Cheung, Y.-M., 2024a. Learning order forest for qualitative-attribute data clustering. In: 27th European Conference on Artificial Intelligence. pp. 1943–1950.

Zhao, X., Liang, J., Dang, C., 2017. Clustering ensemble selection for categorical data based on internal validity indices. Pattern Recognit. 69, 150–168.

Zhao, Z., Wang, R., Huang, D., Li, Z., 2024b. Outlier detection for partially labeled categorical data based on conditional information entropy. Internat. J. Approx. Reason. 164, 109086.

Zhu, C., Cao, L., Yin, J., 2022. Unsupervised heterogeneous coupling learning for categorical representation. IEEE Trans. Pattern Anal. Mach. Intell. 44 (1), 533–549.