symmetry

MDPI

Article

Model Selection Using *K*-Means Clustering Algorithm for the Symmetrical Segmentation of Remote Sensing Datasets

Ishfaq Ali [1], Atiq Ur Rehman [2], Dost Muhammad Khan [1], Zardad Khan [1], Muhammad Shafiq [3,*] and Jin-Ghoo Choi [3,*]

[1] Department of Statistics, Abdul Wali Khan University, Mardan 23200, Pakistan; ishfaqali8380@gmail.com (I.A.); dostmuhammad@awkum.edu.pk (D.M.K.); zardadkhan@awkum.edu.pk (Z.K.)
[2] Department of Mathematics and Statistics, Faculty of Basic and Applied Sciences, International Islamic University, Islamabad 44000, Pakistan; atiq.msst138@iiu.edu.pk
[3] Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Korea
* Correspondence: shafiq@ynu.ac.kr (M.S.); jchoi@yu.ac.kr (J.-G.C.)

Abstract: The importance of unsupervised clustering methods is well established in the statistics and machine learning literature. Many sophisticated unsupervised classification techniques have been made available to deal with a growing number of datasets. Due to its simplicity and efficiency in clustering a large dataset, the *k*-means clustering algorithm is still popular and widely used in the machine learning community. However, as with other clustering methods, it requires one to choose the balanced number of clusters in advance. This paper's primary emphasis is to develop a novel method for finding the optimum number of clusters, *k*, using a data-driven approach. Taking into account the cluster symmetry property, the *k*-means algorithm is applied multiple times to a range of *k* values within which the balanced optimum *k* value is expected. This is based on the uniqueness and symmetrical nature among the centroid values for the clusters produced, and we chose the final *k* value as the one for which symmetry is observed. We evaluated the proposed algorithm's performance on different simulated datasets with controlled parameters and also on real datasets taken from the UCI machine learning repository. We also evaluated the performance of the proposed method with the aim of remote sensing, such as in deforestation and urbanization, using satellite images of the Islamabad region in Pakistan, taken from the Sentinel-2B satellite of the United States Geological Survey. From the experimental results and real data analysis, it is concluded that the proposed algorithm has better accuracy and minimum root mean square error than the existing methods.

Keywords: unsupervised clustering; *k*-means; balanced optimal number of clusters; symmetry; clustering validity indices; remote sensing; root mean square error; satellite images

check for updates

Citation: Ali, I.; Rehman, A.U.; Khan, D.M.; Khan, Z.; Shafiq, M.; Choi, J.-G. Model Selection Using *K*-Means Clustering Algorithm for the Symmetrical Segmentation of Remote Sensing Datasets. *Symmetry* **2022**, *14*, 1149. https://doi.org/10.3390/sym14061149

Academic Editors: Palle E. T. Jorgensen and Mihai Postolache

Received: 4 April 2022
Accepted: 27 May 2022
Published: 2 June 2022

1. Introduction

Due to the rapid advances in computer technology, massive amounts of data are being produced and stored in different areas of study, including earth sciences, medical sciences, social sciences, and engineering. Extracting useful information from such an ocean of data is a challenging task [1–4]. However, it is often costly and unrealistic to look at important aspects of detailed information buried in a dataset. In machine learning, numerous methods and tools are used to deal with such data, making it possible to unearth the prominent symmetrical features hidden in the data. Machine learning algorithms can solve a plethora of complex problems in high-dimensional datasets from various fields.

Clustering is a well-known branch of the machine learning field for uncovering hidden patterns in data in an unsupervised manner, where it can play a critical role in revealing core information at the outset of exploring a massive and unbalanced dataset [5]. It is a technique for partitioning a dataset into significant subgroups, the so-called clusters, in such a way that the items in a cluster are similar/symmetrical to one another but dissimilar/asymmetrical to the items in other clusters. Image segmentation aims to divide images into several non-overlapping regions such that the pixels in each region are akin due to some common features such as intensity, texture or color [6]. Considering the cluster symmetry property, the following points are taken into account while partitioning the data points into k groups $\{G_1, G_2, \ldots, G_k\}$:

- Distribute the data into disjoint/asymmetrical groups, $G_i \cap G_j = \phi$;
- At least one instance should be present in each cluster, $G_k = \phi$;
- Items within each cluster are homogeneous or symmetrical while, externally, they are heterogeneous or asymmetrical.

The clustering approaches usually distinguish items or patterns on the basis of similarity measures, such as Euclidean distance. Due to the unsupervised nature, it has been deployed for data analysis, such as gene expression analysis, heterogeneous data analysis, social network analysis, remotely sensed image segmentation and infrared pedestrian segmentation, etc. [7–9].

A popular and well-known clustering approach is the k-means [10] algorithm, which has great potential to deal with massive datasets [11]. The k-means algorithm starts with the random initialization of clusters' centers, and then iteratively assigns instances to the nearest (closest) cluster centers. Due to its simplicity and computational efficiency, it has been widely used in image segmentation, market segmentation, social network analysis and computer vision, among other fields, and different variants have been developed in the literature [12–15].

Even though the k-means clustering algorithm is the simplest and most commonly used algorithm, it is limited by the required number of clusters, which has to be predetermined. Specifying the correct number of clusters in the k-means algorithm is one of the challenging and fundamental tasks for the researchers. This crucial problem needs urgently to be solved because different numbers of clusters cause different results. In the literature, many variants have been proposed by researchers to improve k-means and to determine the balanced number of clusters [16–19].

A modified k-means algorithm was proposed by Ahmed Shafeeq et al. [20] to improve the clustering quality in finding the needed number of clusters k in a dataset. The k-means algorithm needs the number of clusters k in advance, which is a difficult task with regard to meeting the actual number of clusters of a dataset. This is why the proposed method works well in both situations, for a known and an unknown number of clusters. In the proposed method, the user has the choice to fix the k value in advance or choose the minimum number of clusters.

The "G-means", suggested by Greg Hamerly et al., [21], is an improved algorithm to learn k that is based on a statistical normality test. It states that subsets of data follow a Gaussian (Normal) distribution. A famous method used for evaluating the number of clusters is known as gap statistics [22], which compares the total intra-cluster dispersion with their expected value under an appropriate null distribution for different k values. k will be deemed for which the gap statistic is maximum. It works well for a small number of clusters, whereas it has difficulty when increasing true value of k, as recommended by Feng and Hamerly [23]. Siddheswar Ray et al. [24] elaborated a simple procedure to find the parameter k automatically based on the inter-cluster and intra-cluster distance methods. The basic idea of this method is to partition the images from two groups up to $k_{\max}$ groups, where $k_{\max}$ is the upper limit of the cluster number. The minimum value of the index determines the best clustering.

A modified form of the k-means algorithm was proposed by Nidhi Gupta et al. [25], which does not require the number of clusters in advance. It takes less computational time as compared to the traditional k-means algorithm. A simple and powerful mechanism to capture the knee point in the curve on the basis of curvature was suggested by Zhang Yaqian et al. [26]. They compared the proposed method with six other existing indices and concluded that the proposed method outperforms the others. Similarly, Trupti

M.Kodinariya et al. [27] explained six different techniques to find the optimal value of k for the k-means algorithm.

More recently, a divergent ratio of the sum of squares and Euclidean distance was suggested by Lie et al. [28], which determined the near-optimal number of clusters; moreover, Zhou et al. [8] proposed an enhanced method to determine the true k value based on the cluster centroid and closest neighbor cluster. Jian Di et al. [16] presented an enhanced bisecting k-means algorithm for automatically determining the number of clusters (k) and improved the cluster centroids. Shao et al. [29] introduced a voting mechanism technique to determine the number of clusters (k) in an automatic way in the dataset. A series of k sets was obtained by using different clustering validity indices and they concluded that the highest frequency of k is the true number of clusters.

In addition, in the literature, a vast number of surveys and comparative studies concerning the clustering validity indices have been developed extensively—for example, the Duda Index [30], Calinski–Harabasz (CH) index [31], Dunn index [32], Haritang index [33], Davies–Bouldin (DB) index [34], silhouette method [35] and Krzanowski and Lai's method [36]. Among these, some of them only consider the geometric structural knowledge of the dataset, some are based on compactness and some are based on separability, while some of them consider both compactness as well as separability. For new validity indices, these two measures are applied either by sum or quotient.

The main goal of this paper is to develop a novel method for determining the appropriate number of clusters, k, more accurately in larger datasets. Moreover, in comparison to other approaches that have been developed, the proposed method is unique in that it automatically selects the number of clusters from the data. In other words, the proposed strategy is a data-driven approach that makes it easier to make an objective rather than subjective judgment about the number of clusters. Moreover, the k-means algorithm is computationally expensive when clustering very large datasets, such as satellite images. We scale it up to big data to increase its efficiency and reduce the computational time.

Therefore, the main motivation of this study is to propose a method to determine the number of groups in data in an automatic manner using a modified k-means clustering algorithm. Secondly, the k-means clustering algorithm is computationally expensive for large datasets due to its iterative nature. In this regard, effort has been made using the quantization step to scale the k-means clustering algorithm and control the noise and outliers in the data. Moreover, the proposed method can be used in large datasets such as satellite images, where it is to be utilized in remote sensing in order to identify deforestation and urbanization data for future planning and development.

The proposed idea is to use the k-means clustering algorithm to measure the uniqueness of the clustering results. The clustering centroids are unique if the central values remain constant after applying the clustering algorithm. Our hypothesis for this study is that determining the true number of clusters will yield more stable clustering results.

The remainder of this article is organized as follows. Section 2, briefly elaborates ten cluster validity indices and k-means. A detailed elucidation of our proposed methodology is provided in Section 3. Section 4 includes the experimental results and application, where the performance and effectiveness of the proposed method are evaluated. The concluding remarks are given in the last section.

2. Materials and Methods

In this section, we present an overview of the k-means clustering algorithm and clustering validity indices.

2.1. The k-Means Algorithm

In the k-means clustering algorithm, the dataset N is partitioned into k pre-determined number of clusters, where k will be specified by the user or by applying some heuristic approaches. It randomly initializes the cluster centroids and assigns data points to the nearest (closest) cluster centroid on the basis of the Euclidian distance between data points {x} and centroids {c}. The distance is calculated between any two points by Equation (1) and is defined as

$$\text{Euclidean distance} = d(x,c) = \sqrt{\sum_{i=1}^{n}(x_i - c_i)^2}. \tag{1}$$

When all the points have been placed, we recalculate new cluster centroids and repeat (assign data points to the nearest cluster centroid) the same procedure until no change occurs in the centroid values. There are other methods of computing the distance between points/vectors; however, the most commonly used distance measuring method is Euclidean distance. The main goal of k-means is to minimize the sum of squared error among data points and their respective cluster centers, defined by Equation (2).

$$\text{wss} = \sum_{i=1}^{k} \sum_{x \in G_i} d(x, c_i), \tag{2}$$

where c_i is the ith cluster centroid.

The calculation procedure for the k-means clustering method is summarized below as described by [37].

1. Select k initial cluster center points randomly, such as $c_1(1), c_2(1), \ldots, c_k(1)$.
2. At kth iteration, assign the data points {x} to k clusters using the given relation $x \in G_j(k)\ if \parallel x - c_j(k) \parallel < \parallel x - c_i(k) \parallel$ ∀ (for all) $i = 1,2,\ldots,k;\ i \neq j$, where $G_j(k)$ denotes group of data points whose centers are $c_j(k)$.
3. Recalculate the location of new cluster centers $c_j(k+1);\ j = 1,2,\ldots,k$, so that the sum of squared distance to the new cluster center is minimized from all points in $G_j(k)$. The degree of measurement that serves to minimize the distance is the sample mean of $G_j(k)$. Hence, the new cluster is measured by Equation (3).

   $$G_j(k+1) = \frac{1}{N_j} \sum_{x \in G_i} x, \qquad j = 1,2,\ldots,k, \tag{3}$$

   where N_j is the total number of sample points in $G_j(k)$.
4. If $c_j(k+1) = c_j(k)$, for $j = 1,2\ldots k$, the algorithm stands converged and the process is deemed terminated; otherwise, repeat steps 2 and 3.

Even though the k-means clustering algorithm is the simplest and most commonly used algorithm, it is limited by the required number of clusters, which has to be pre-determined.

2.2. Calinski and Harabasz (CH) Index

The CH index technique was proposed by Calinski and Harabasz [31] to determine the ideal number of clusters k and is defined as

$$\text{CH}(k) = \frac{(b_k)(n-k)}{(w_k)(k-1)}, \tag{4}$$

where w_k is the within-group sum of squared (wss) means dissimilarity within groups and b_k is the between-group sum of squared (bss) means dissimilarity between groups.

2.3. Silhouette Index

The silhouette index was introduced by Kaufman and Rousseeuw [35] and was built to show graphically how well each element is categorized in a given clustering output; it is defined as

$$\text{S(i)} = \frac{b(i) - a(i)}{\max[a(i), b(i)]}, \tag{5}$$

where $a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d_{ij}$ is the average dissimilarity of the ith element to all other elements of cluster C_i and $b(i) = \min_{n \neq i} \frac{1}{|C_n|} \sum_{j \in C_n} d_{ij}$ is the average dissimilarity of the ith element to all elements of cluster C_n. The optimal number of clusters will be the value for which the S(i) is maximum.

2.4. Krzanowski and Lai (KL) Index

Krzanowski and Lai [36] proposed the KL index to determine the ideal number of clusters k and it is defined as

$$kl(k) = \left| \frac{\mathrm{Diff}_k}{\mathrm{Diff}_{k+1}} \right|, \tag{6}$$

where $\mathrm{Diff}_k = (k-1)^{2/p} trace(w_{k-1}) - k^{2/p} trace(w_k)$, w_k is the within-group dispersion matrix for data clustered into k clusters and p is the number of variables. The estimation of k, maximizing $kl(k)$, is viewed as indicating the ideal number of clusters.

2.5. Gap Statistic

Tibshirani et al. [22] proposed a way to determine the ideal number of clusters k in a dataset by the gap statistic. The idea of the gap statistic is to compare the total intra-group dissimilarity w_k for various numbers of k with their expected values generated from a reasonable reference null distribution.

$$Gap(k) = \frac{1}{Z} \sum_{b=1}^{Z} \log(w_{kb}) - \log(w_k), \tag{7}$$

where Z is the reference dataset generated using uniform distribution and w_{kb} is the total intra-group deviation. A value of k that maximizes the gap statistic will be the estimate of the ideal cluster number.

2.6. Dunn Index

The well-separated and compact clusters in a dataset are identified by the Dunn index [32]. The basic purpose of the Dunn index (DI) is to reduce the intra-cluster distances and to maximize the inter-cluster distances. The Dunn validity index (VI_{Dunn}) is defined by Equation (8).

$$VI_{Dunn} = \frac{\min_{1 \leq i < j \leq k} d(G_i, G_j)}{\max_{1 \leq z \leq k} diam(G_z)}, \tag{8}$$

where

- $d(G_i, G_j)$ denotes the dissimilarity between two groups/clusters, i.e., G_i and G_j are defined as $d(G_i, G_j) = \min_{x \in G_i, y \in G_j} d(x, y)$;
- $diam(G)$ is the diameter of a group/cluster G, which measures cluster dispersion, which is defined as $diam(G) = \max_{x,y \in G} d(x, y)$.

For cohesive and well-alienated clusters, the distance among the clusters will be larger and the diameter of the cluster will be smaller. Hence, the selection criterion of k in the dataset depends on the value that maximizes the Dunn index (VI_{Dunn}).

2.7. Duda Index

A ratio criterion $(je(B)/je(A))$, defined by Equation (6), was presented by Duda and Hart [30]. This index can be calculated as

$$VI_{Duda} = \frac{je(B)}{je(A)}, \tag{9}$$

where $Je(B)$ is the sum of squared errors within groups when the datasets are segregated into two groups and $Je(A)$ provides the squared errors when only one group exists. A group is rejected if ratio $(je(B)/je(A))$ is lower than the critical value. The maximum value signals to stop the index and similar groups are well separated according to the specific values of k.

2.8. Pseudot2 Index

In 1973, Duda and Hart [38] proposed another validity index known as pseudot2; the mentioned index can only be applied to hierarchical methods. It is defined as

$$VI_{Pseudot2} = \frac{je(A) - je(B)}{\frac{je(B)}{n_k + n_l - 2}}, \tag{10}$$

where $je(A)$ and $je(B)$ are defined in the Duda index.

The smallest value of k is the optimal number of clusters specified by Gordon [38], such that

$$VI_{Pseudot2} \leq \left(\frac{1 - Critvalue_{Duda}}{Critvalue_{Duda}} \right) \times n_k + n_l - 2. \tag{11}$$

2.9. Davies–Bouldin (DB) Index

The Davies–Bouldin index [34] is defined as "the ratio of the sum of the intra-cluster dispersion to the inter-cluster separation" and can be calculated using Equation (12).

$$VI_{Davies\text{-}Bouldin} = \frac{1}{K} \sum_{i=1, i \neq j}^{k} \max \left(\frac{S_i + S_j}{d_{ij}} \right), \tag{12}$$

where k is the cluster numbers, S_i is the standard deviation of the distance of point v from the center g_i in cluster G_i, S_j is the standard deviation of the distance of point v from the center g_j in cluster G_j, and d_{ij} is the Euclidean distance, which measures the distance between centroids g_i and g_j of clusters G_i and G_j. A smaller value of the $VI_{Davies\text{-}Bouldin}$ indicates the appropriate number of clusters.

2.10. Rubin Index

An alternative clustering criterion was proposed by Friedman and Rubin (1967) [39], which is based on the ratio of the determinant of the total sum of squares and cross-product matrix to the determinant of the pooled within-cluster matrix. This index can be calculated using Equation (13).

$$VI_{Rubin} = \frac{det(t)}{det(w_k)}. \tag{13}$$

The minimum value of second differences between levels is used to select the optimal number of clusters [19].

2.11. C-Index

The C-index validity of clustering was reviewed by Hubert and Levin [40]. It can be calculated using Equation (14).

$$VI_{C\text{-}index} = \frac{S_{(w)} - S_{(\min)}}{S_{(\max)} - S_{(\min)}}, \quad S_{(\min)} \neq S_{(\max)}, V_{C\text{-}index} \in (0, 1), \tag{14}$$

where

- $S_{(w)}$ is the sum of the Euclidean distance between objects x_i and x_j;
- $S_{(\min)} = \sum_{x_i, x_j \in X} \min(N_w)\ d(x_i, x_j)$;
- $S_{(\max)} = \sum_{x_i, x_j \in X} \max(N_w)\ d(x_i, x_j)$.

A smaller value of the $VI_{C\text{-}index}$ index indicates the appropriate number of clusters.

3. The Proposed Method

In this section, a new method is proposed to determine the true value of k in an automatic way that depends on the central values of the clusters. In this method, multiple numbers of clusters are determined for each value of k using the k-means clustering algorithm in order to check the uniqueness among multiple cluster centroids. The multiple numbers of cluster means the number of repetitions of the k-means algorithm j times, where j depends on the size of the dataset. In this article, the J is fixed to 5 in order to reduce the computational cost.

In the proposed clustering algorithm, all the parameters is kept fixed, except the number of clusters, in order to achieve the best clustering model.

In this method, we choose an optimum value of k from a range of values ($1 \leq k \leq 20$) within which its true value is presumed to lie, and we test the k-means clustering algorithm multiple times for each value of k individually.

For instance, suppose that k = 1, and run the K-means clustering algorithm multiple times to obtain the results of the cluster centroid for each trial. If the cluster centroid values of all the trials (obtained from the K-means clustering algorithm) are the same, we move towards the next value of k, and this process is repeated for different successive values of k until the same centroid values of the clusters are produced. The same process is performed multiple times to obtain the actual number of k groups, thereby deducing the average, and the resultant value is deemed to be the final value of k. A step-by-step Algorithm 1 for selecting the optimum number of clusters is described as follows.

Algorithm 1

1. Consider a dataset $\{x\}$ to be partitioned into k clusters.
2. Define the range of clusters k where its true value is presumed to lie, i.e., $1 \leq k \leq 20$.
3. For each value of k, apply the k-means clustering algorithm multiple times (J times) on the given dataset $\{x\}$.
4. The centroid values of clusters C_{jk} are calculated j times by using the following expression:

$$C_{jk} = \frac{\sum_{i=1}^{n_k} x_{ik}}{n_k}, \qquad \text{for } j = 1, 2, \ldots, J. \tag{15}$$

5. The uniqueness of the centroid values obtained in step 4 is checked.
6. If uniqueness is established, select the next value of k.
7. Repeat the process from step 3 to 6 until a global unique value of k is achieved.
8. The optimum number of clusters k will be the value for which the uniqueness of cluster centroids is observed the maximum number of times in the iterative procedure.

The process of the proposed method is depicted in Figure 1.

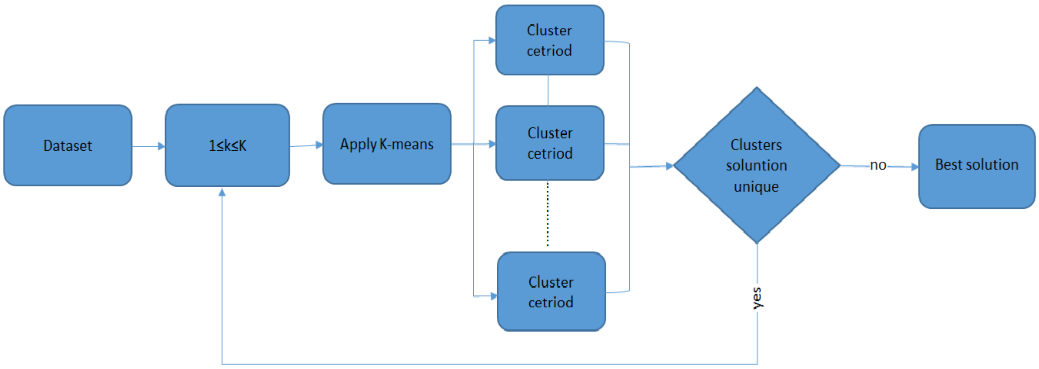

Figure 1. Flow chart of the proposed method.

To elaborate the proposed idea, we utilize a two-dimensional dataset of size 300 consisting of five elliptical-shaped clusters generated from Gaussian distribution with different means and co-variance matrices, as shown in Figure 2a. The *k*-means clustering algorithm is applied multiple times for partitioning it into 5 clusters, where the cluster centroids remain intact (unique), as elucidated in Figure 2b–d. However, if the number of groups is set beyond its true value, i.e., for $k = 6$, then the cluster centroids do not remain the same for each trial of the clustering algorithm, which is elucidated in Figure 2e–g.

In Figure 2e, the top-left cluster contains two groups and the top-right cluster comprises one group, where, in the next trial, the top-left cluster comprises one cluster and the top-right is partitioned into two groups, as shown in Figure 2f, while in Figure 2g, the top-most clusters are partitioned into two groups and the bottom-left two clusters are merged into one cluster, which shows clear disagreement among them. Therefore, for $k = 5$, the uniqueness has been considered as the true value of k. However, uniqueness can also be obtained for the smaller number of groups other than true value of k because the nearest clusters are always combined with each other.

Thus far, where the iteration of the proposed method is concerned, it is computationally expensive in big data analysis, such as satellite images. To enhance the computational power of the proposed method, we reduce the original dataset by a quantization step. In this regard, a sagacious method is to arrange akin objects of a dataset by compressing it into a quantized set of data (average of similar components) that will be representative of all the cluster objects. The quantized data are achieved by deliberately producing centroids of a large number of small clusters through the *k*-means clustering algorithm. Utilizing this method, the most favorable quantized set of data can be generated for the refinement of the proposed algorithm and we gain a significant improvement from days to minutes in processing time. By adopting this method, clustering was performed for satellite images of the Islamabad region in order to obtain substantial information regarding different groups, such as forestry, urbanization, barren land, etc.

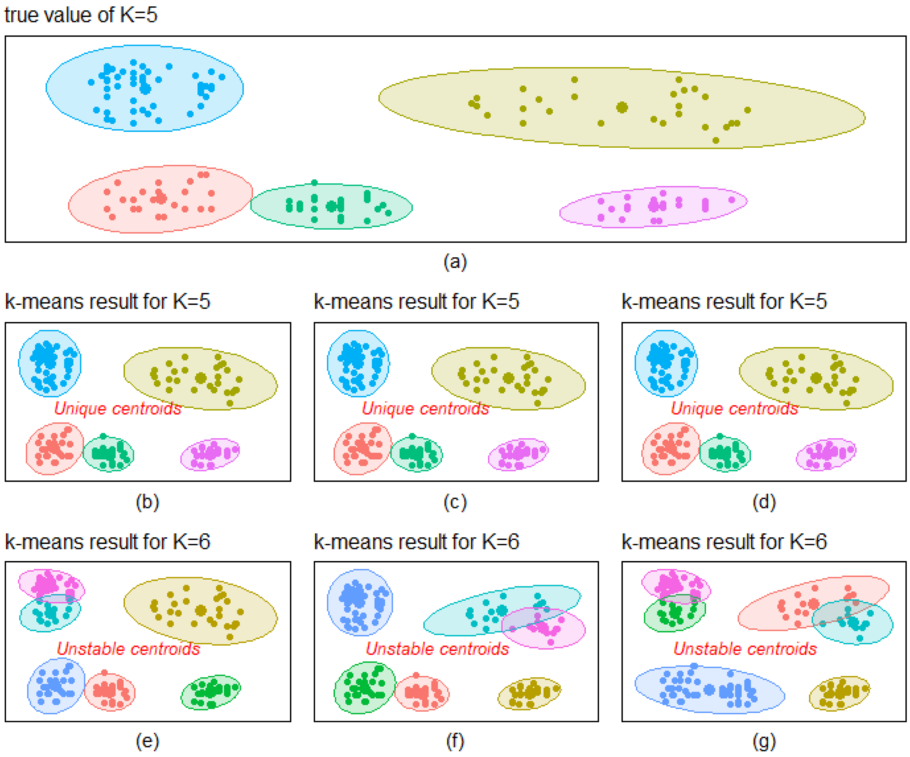

Figure 2. (**a**) shows the original data having true number of clusters $k = 5$. (**b**–**d**) show the same clustering results when $k = 5$ (correct number of clusters). (**e**–**g**) show different centroid values when $k = 6$ (incorrect number of clusters).

4. Experiments and Results

In this section, we describe the datasets, working environment and results obtained using the proposed method and also discuss its comparison with other existing CVIs.

4.1. Datasets and Experimental Procedure

Here, the performance of the proposed method is evaluated, where different experiments are described for synthetic (simulated) datasets as well as real-world datasets that have a known number of groups. To examine the efficiency, experiments were conducted to compare the results of the proposed method with ten other common existing cluster validity indices (CVIs), as discussed in the previous section. The k value has been fixed for each algorithm, ranging from 1 to 20 ($1 \leq K \leq 20$) for all the datasets during the experimentation, because the minimum value of k for a synthetic dataset is 3, while the maximum is 7. The goodness of best fit for each CVI is determined by the root mean square error (RMSE) between the true number of groups and the predicted value of k. The CVI is superior if the value of RMSE is smaller.

Obviously, for the simulation study, we utilize different synthetically created datasets generated from Gaussian distribution that contain different numbers of clusters k regarding their sizes, shapes and dimensions, with different means and co-variance matrices. The experiments are conducted for three distinct configurations of k, i.e., 3, 5 and 7, and each dataset contains a different number of instances, i.e., 1000, 5000 and 10,000, with 2, 4 and 6 dimensions. For every configuration, 100 synthetic datasets are generated from Gaussian distribution.

We also test the proposed method on 10 real datasets taken from the UCI machine learning repository [41] and compare its results with other existing CVIs. These datasets have prior known numbers of clusters; therefore, they are mostly used by the researchers for assessment purposes. Table 1 summarizes the characteristics of the 10 real-world datasets. It is of the utmost importance that for the real datasets, the obtained results should be analyzed with care, as these datasets are always deemed to be utilized with supervised learning, and this is why they are sparingly used in clustering-related problems [42].

Table 1. The basic characteristics of real-world datasets.

Dataset	Number of Instances	Number of Attributes	Number of Clusters
Iris	150	5	3
Wine	178	13	3
Tripadvisor review	980	11	2
Aggregation	788	3	7
Flame	240	3	2
Control	600	-	6
Glass	214	9	7
Pathbased	300	3	3
Breast	699	10	2
Vechile	946	18	4

Furthermore, the proposed method is also applied on satellite images of Islamabad, the capital city of Pakistan, and its surroundings—that is, parts of the Rawalpindi and Khanpur districts—to identify various numbers of clusters with the aim of remote sensing, such as deforestation, urbanization, etc. For this purpose, two images are acquired, one of June 2018 and another of June 2021, from the Sentinel-2B satellite of the United States Geological Survey. The acquired images are converted into workable data files using the 'raster' package [43] in R software.

We used 8 spectral bands with the central wavelength, bandwidth and spatial resolution shown in Table 2. There are other spectral bands; however, these bands are not appropriate for our study. Band 9 is, for example, useful for detecting clouds, which are not relevant to our research. Similarly, bands 10 and 11 are used to provide precise surface temperatures, and we are not interested in determining temperature. The spatial resolution for bands 5–7 and 8A is 20 m, and they were resampled to a 10-m resolution.

Table 2. Eight spectral bands.

Band Number	Central Wavelength (nm)	Bandwidth (nm)	Spatial Resolution
2	490	65	10
3	560	35	10
4	665	30	10
5	705	15	20
6	740	15	20
7	783	20	20
8	842	115	10
8A	865	20	20

The experimental results obtained from these datasets are discussed below. The simulation experiments were carried out in *R* software for Windows 3.5.2 and used the NbClust function of the NbClust package, to find the "best" CVI. The specifications of the computer used for the execution of experiments are as follows: Inter (R) Core (TM) i7-5500U CPU @ 2.40 GHz; RAM 16 GB with Windows 10, 64-bit operating system. Furthermore, for the clustering of satellite images, the Google Cloud Platform (GCP) was utilized with RAM of 32 GB.

4.2. Results

All experiments included a comprehensive comparative analysis, where we engaged a number of clustering validity indices studied in the literature. Here, we put forward the results of various experiments actually conducted on the synthetic datasets. The value of N has been set in such a way as to check the performance of the proposed algorithm for small as well as for large sizes, dimensions and clusters. The success hits of each CVI are properly optimized in Table 3 and their percentages of success are graphically portrayed in Figure 3 with respect to different sample sizes, dimensions and clusters. It can be observed from the results given in Table 3 that the proposed method achieved the maximum success score for all the k values, as depicted in Figure 3. Moreover, it can also be observed that all the existing indices suffered in terms of successes with respect to the increase in the value of k. With regard to dimension, the fact remains unchanged that the performance and utilization of the proposed method is superior. Moreover, it was also observed that there is a keen and stable relationship between sample size and dimension. In the detection of the true value of k, the success score of the proposed method became stabilized with the increase in sample size and varied dimensions. However, it has been observed that the computational complexity of all the indices increases significantly with the increase in the sample size and dimensions.

In the simulation study, the proposed method was followed by the CH index with respect to the peak performance in all the datasets. The percentage of success for the small value of k (i.e., k = 3) remained as 1:2 on average, and when increasing the value of k, this method reaches beyond less than 200 percent compared to the proposed method. It was observed for the CH index that by increasing the sample size to 10,000, the peak performance is minimized significantly. There is the eminent inverse relationship between sample size and dimension, which means that the success score of the CH index shows better performance with the enhancement in the dimensions and decrease in sample size.

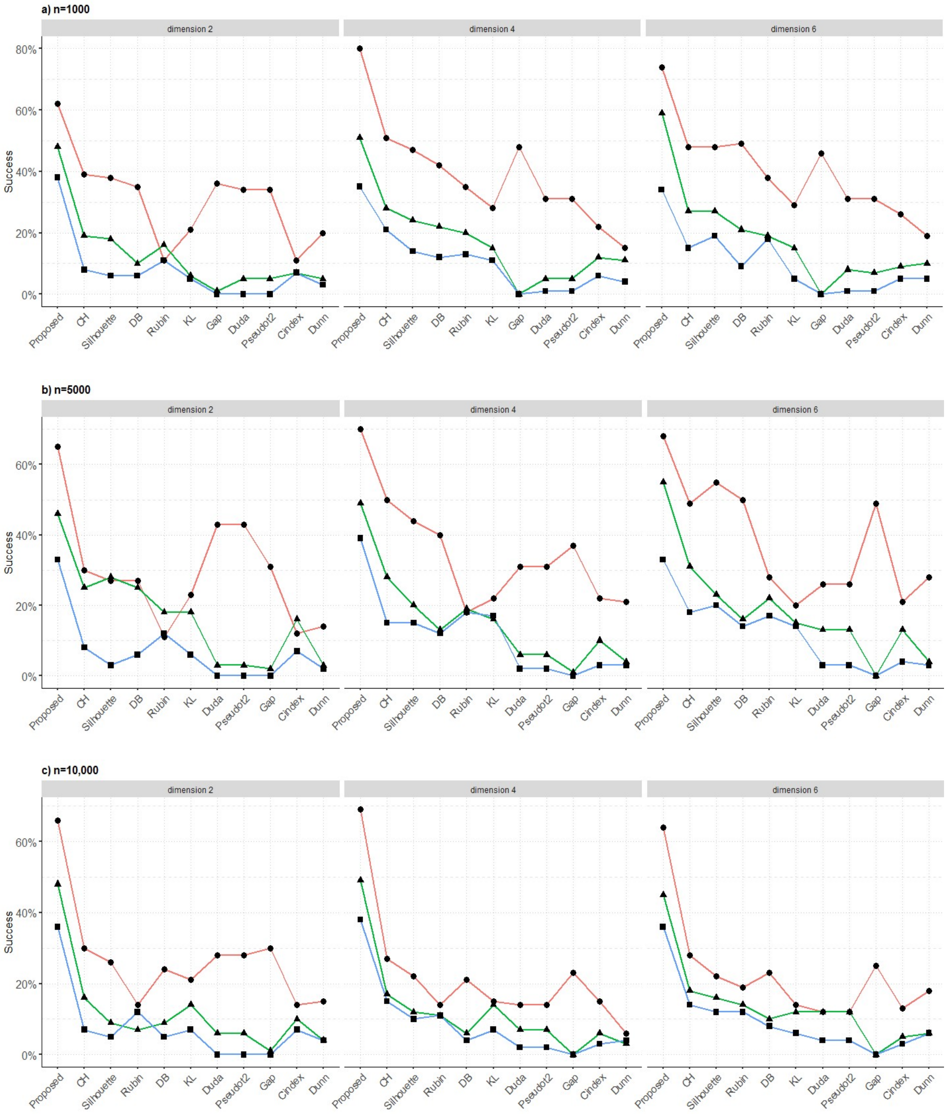

Figure 3. The number of successive hits for eleven validity indices for different sample sizes, dimensions and numbers of clusters. Methods are shown on the x-axis. Lines coloured as red, green and blue show number of clusters equal to 3, 5 and 7, respectively.

Similarly, the success of the silhouette index for $k = 3$ remains unchanged due to the enhancement in dimensions and sample size, whereas for $k = 7$, it leads to a higher success score due to the increase in the dimensions. Nevertheless, viewing collectively with regard to dimension for $k = 3$ and $k = 5$, the success score in the silhouette index indicates an improvement as compared to $k = 7$.

Moreover, the gap, DB, pseudot2 and Duda indices show similar numbers of successes for $k = 3$. For a small value of k (=3) in relation to sample size and dimension, it does not affect the better performance of the gap, pseudot2 and Duda indices. Meanwhile, for a large value of k (=7), the success score of the above-evaluated method tends towards lower performance (failure). It was observed in these indices that for the maximum true value of k (5 and 7), they opt for the lower value of k (2 and 3). This is for the obvious reason that the peak performance of these indices is at its lowest value of percentage. Meanwhile, the DB index provides good performance for a maximum value of k (5 and 7) after silhouette. However, there is an inverse relation between sample size and dimension in the evaluation of the DB index, so that the success score of the DB index increases with an increase in dimension and decreases with an increase in sample size.

Table 3. Summary of the hits for each validity index on synthetic datasets.

N	D	k*	k+										
			Proposed	**CH**	**KL**	**Silhouette**	**Gap**	**C-Index**	**Pseudot2**	**DB**	**Duda**	**Dunn**	**Rubin**
1000	2	3	62	39	21	38	36	11	34	35	34	20	11
		5	48	19	6	18	1	7	5	10	5	5	16
		7	38	8	5	6	0	7	0	6	0	3	11
	4	3	80	51	28	47	48	22	31	42	31	15	35
		5	51	28	15	24	0	12	5	22	5	11	20
		7	35	21	11	14	0	6	1	12	1	4	13
	6	3	74	48	29	48	46	26	31	49	31	19	38
		5	59	27	15	27	0	9	7	21	8	10	19
		7	34	15	5	19	0	5	1	9	1	5	18
5000	2	3	65	30	23	27	31	12	43	27	43	14	11
		5	46	25	18	28	2	16	3	25	3	3	18
		7	33	8	6	3	0	7	0	6	0	2	12
	4	3	70	50	22	44	37	22	31	40	31	21	18
		5	49	28	16	20	1	10	6	13	6	4	19
		7	39	15	17	15	0	3	2	12	2	3	18
	6	3	68	49	20	55	49	21	26	50	26	28	28
		5	55	31	15	23	0	13	13	16	13	4	22
		7	33	18	14	20	0	4	3	14	3	3	17
1000	2	3	66	30	21	26	30	14	28	24	28	15	14
		5	48	16	14	9	1	10	6	9	6	4	7
		7	36	7	7	5	0	7	0	5	0	4	12
	4	3	69	27	15	22	23	15	14	21	14	6	14
		5	49	17	14	12	0	6	7	6	7	3	11
		7	38	15	7	10	0	3	2	4	2	4	11
	6	3	64	28	14	22	25	13	12	23	12	18	19
		5	45	18	12	16	0	5	12	10	12	6	14
		7	36	14	6	12	0	3	4	8	4	6	12

Note: N, size of dataset; D, dimension of dataset; k*, true number of clusters; k^+ denotes the number of clusters out of 100 that were correctly identified by different existing CVIs.

During the evaluation of the Dunn, C-index and KL indices to capture the true clusters, their performance was deemed to be unsatisfactory. The performance of the Dunn index with respect to sample size was the lowest in rank among them. Meanwhile, it was observed for the C-index and KL index that in order to find the true value of k, there is a tendency toward its higher value (redundant clusters), due to which, for $k = 5$ and $k = 7$, the success score increases. Apart from this, due to an increase in dimension, the performance of KL becomes inhibited, and due to an increase in sample size, the performance becomes somewhat better.

Additionally, in the arena of performance, the results obtained with the help of the Rubin index for $k = 3$ are quite unsatisfactory; however, for an increased value of k (5 and 7), its performance tends toward satisfaction. Similarly, with an increase in dimension, the success score of the Rubin index becomes better, and with an alteration in sample size, it is not impacted.

Keeping in view all the results related to k, the sample size and dimension, the proposed method was found to be the best among all existing CVIs.

To assess the performance of the proposed and other CVI methods, ten real-world datasets were used, listed in Table 1. All the datasets were clustered by the CVIs, including the proposed one, and their results are summarized in Table 4 and graphically shown in Figure 4. It is noted that the proposed index determined the correct optimal number of clusters for 7 out of 10 datasets. The CH, silhouette, DB and Rubin indices achieved the same results for four datasets. The correct optimal number of clusters for three datasets was achieved by the gap, Duda and pseudot2 indices. The KL index and C-index performed similarly for two datasets, while Dunn worked well only for one dataset.

To test the performance of each CVI, we determined the root mean square error (RMSE) between the actual k values and the estimated k values. The percentage of success and RMSE of each validity index are summarized in Table 4. The percentage of success is shown in Figure 4. We can see that the computed value of the RMSE of the proposed method is low, while the percentage of success is maximum among all the CVIs. Hence, it can be concluded that the proposed method has correctly partitioned the real datasets.

All the methods, including the proposed method, showed almost the same results for both the simulation study and real data applications. It can be deduced from the results that those CVIs that produce poor results for the simulated datasets also fell short in the real dataset applications regarding the ranking, as depicted in Figures 3 and 4. Concentrating on the good performance, we can see that the proposed index remains the best for synthetic datasets as well as for real datasets.

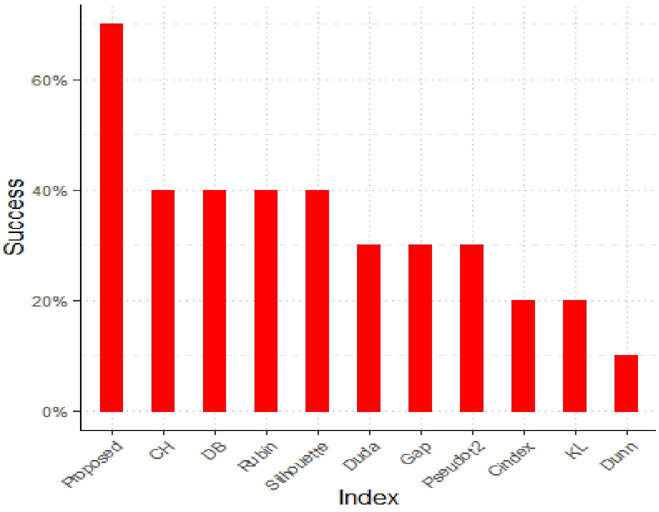

Figure 4. Graphical depiction of the percentage of success of each validity index for real datasets.

Table 4. Summary of the hits of each validity index for real datasets.

Datasets	**k***	**k$^+$**										
		Proposed	**CH**	**KL**	**Silhouette**	**Gap**	**C-Index**	**Pseudot2**	**DB**	**Duda**	**Dunn**	**Rubin**
Iris	3	3	3	12	2	2	3	2	2	2	12	3
Wine	3	3	3	3	3	2	2	3	3	3	19	3
Tripadvisor review	2	2	2	2	2	2	6	2	2	2	15	2
Aggregation	7	7	17	19	4	2	16	2	4	2	17	17
Flame	2	3	4	15	4	2	5	2	4	2	20	9
Control	6	4	3	12	2	2	20	2	2	2	13	3
Glass	7	6	6	18	3	2	7	2	7	2	2	8
Pathbased	3	3	19	9	3	2	11	2	2	2	20	12
Breast	2	2	2	18	2	2	14	3	2	3	2	2
Vechile	4	4	3	10	2	2	2	3	2	3	19	8
Success		7	4	2	4	3	2	3	4	3	1	4
RMSE		0.77	6.10	9.38	2.27	2.70	7.17	2.64	1.87	2.64	12.32	5.06

Note: The k* in the second column denotes the true number of clusters in the real-world datasets and k$^+$ denotes the number of clusters estimated by different existing CVIs.

The proposed algorithm was used to identify the number of groups in the images for 2018 and 2021, and it determined five clusters and six clusters, respectively. The actual and clustered images are shown in Figures 5 and 6. The k-means clustering algorithm was then applied on both the images with a total of 30 initial configurations, where each group was assigned a separate color.

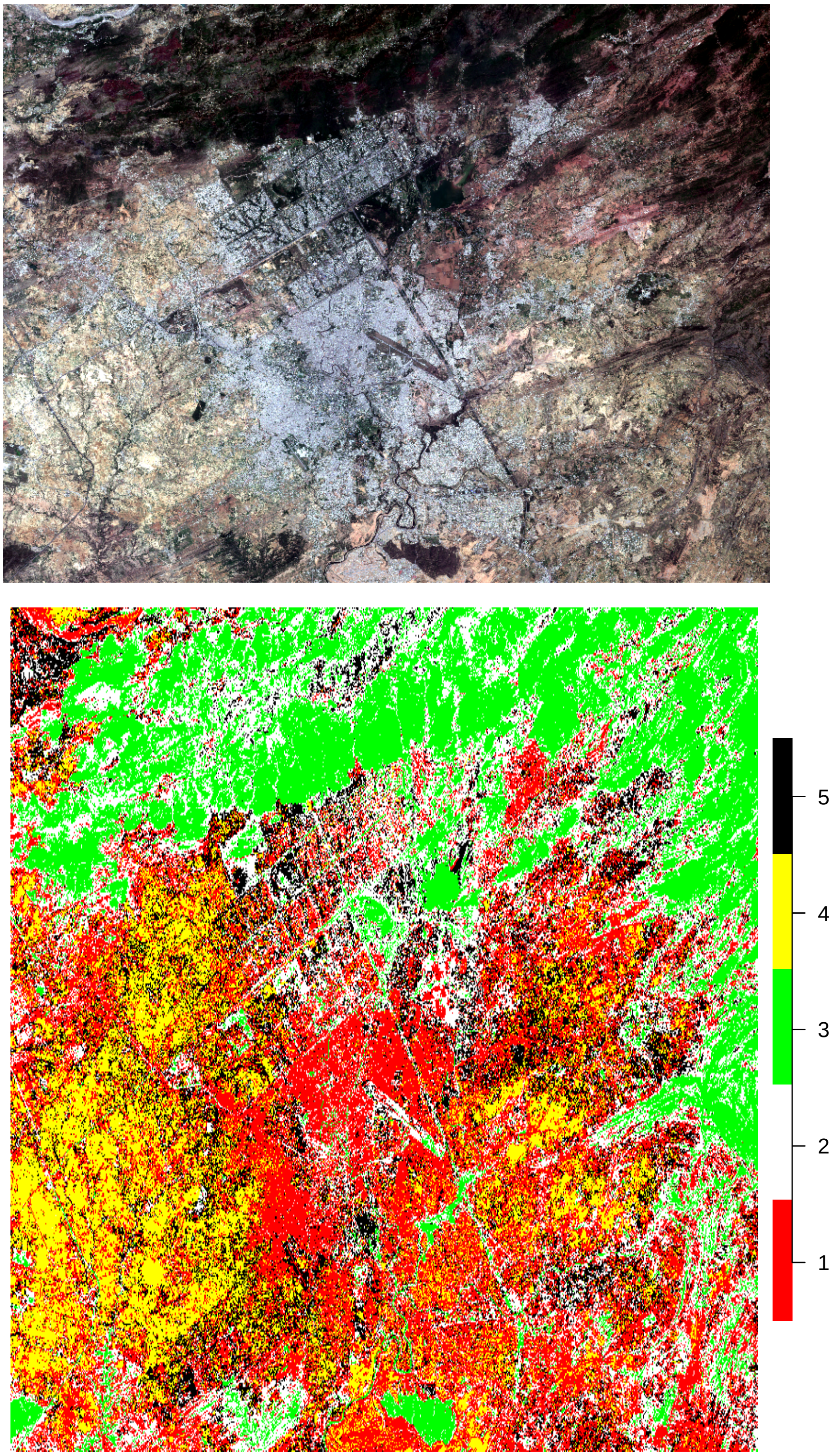

Figure 5. Snapshot of a cropped satellite image of 28 June 2018 of Islamabad and its surroundings, and its clustered image with a total of five clusters.

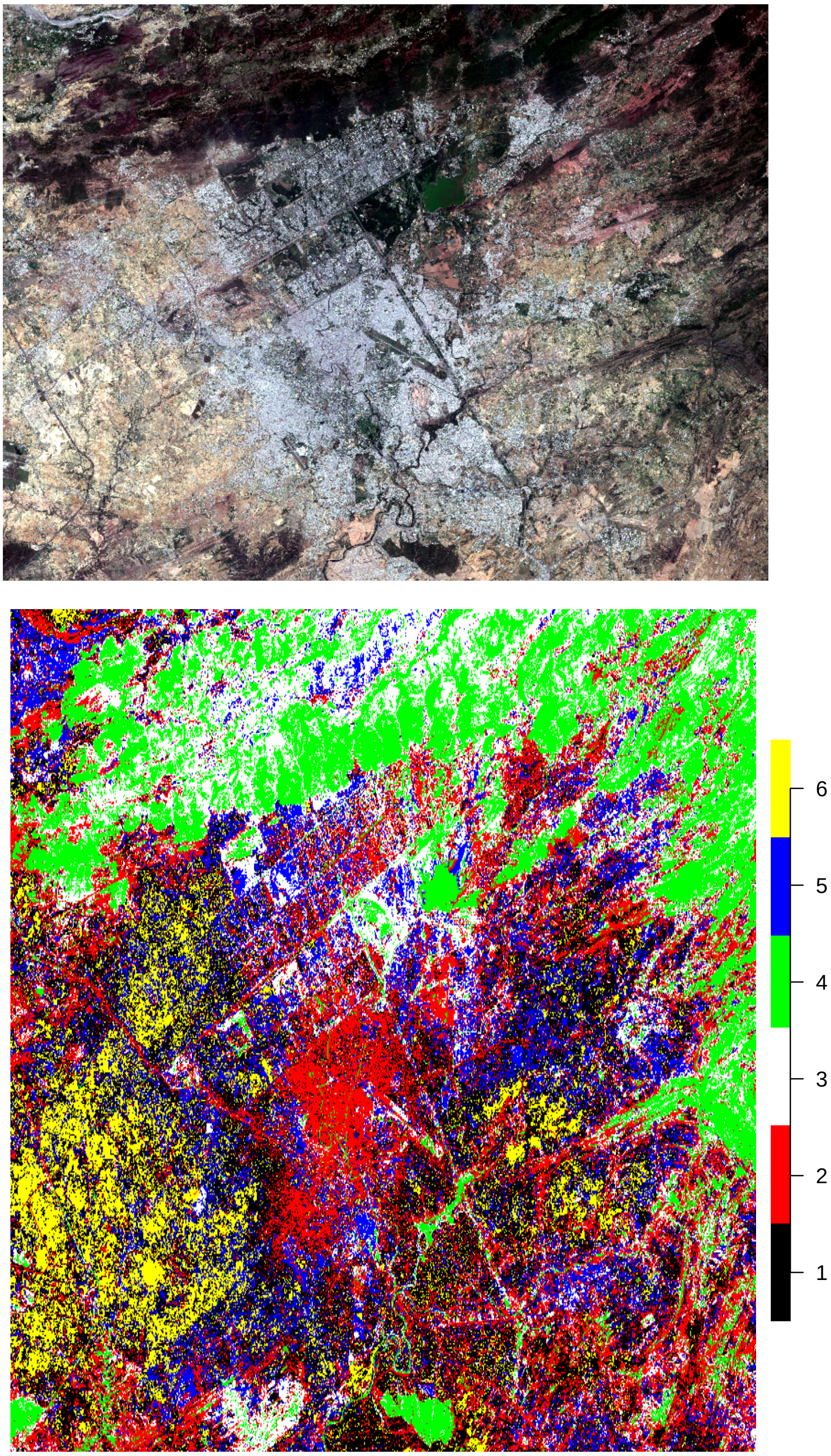

Figure 6. Snapshot of a cropped satellite image of 29 June 2021 of Islamabad and its surroundings, and its clustered image with a total of six clusters.

In the satellite image of 2018, the first cluster, red, depicts the urbanization, which includes commercial and residential areas, with a total of 27.1%, having the largest percentage of pixels and area covered compared to other clusters (as shown in Figure 5). The red-colored cluster also includes some of the Haro River downstream of the Khanpur Dam, located in Khanpur, and it can be seen in the top-left corner of the actual image in

Figure 5. The white color illustrates infertile land (or barren land) at 23.8%, which is mostly visible near the forest region; this cluster also contains roads and walking tracks. The forest cluster, mixed with parks, playgrounds and water (Rawal Lake, which has been grouped with the forest cluster), is represented by the green color, and it covers 19.96% of the total area. In the original image, the forest area called Margala Hills is mostly located at the top of the image. The cluster with the yellow color depicts the arable area, with 14.6%, nearly half of the urban area, which is mostly located outside the urban area of Islamabad and Rawalpindi. Meanwhile, the last cluster (colored black) represents non-forest vegetation in the given image, with 14.46%, which includes crops.

The second satellite image of 2021, shown in Figure 6, has been clustered into six subgroups. It can be observed that the forest area has decreased from 19.96% to 16.46%, as indicated by cluster number 4 with the green color in Figure 6. This green cluster also includes water—as the color of water in the given actual image shown in Figure 6 resembles that of the forest, they have been assigned to the same group—in the Rawal Lake and Nullah Leh, a small river passing through the city of Islamabad and Rawalpindi. The cluster with the red color shows urbanization, including both residential and commercial areas, with a total of 27.1% in 2018, which decreased to 20.97% in 2021.

However, some of the urban area has been assigned to another cluster, as shown by the cluster with the red and black color. However, as solar panels have been installed on the rooftops of some buildings, they have been clustered into one separate group, colored in black. This black color also represents buildings with dark-gray-colored roofs (old residential areas) as well as scrub land. Besides these, the urbanization cluster also covers roads. The white-colored cluster indicates the grassy area, with a total of 17.2%. The blue-colored cluster shows the infertile land, which decreased to 13.5% compared to 2018. The last cluster, colored in yellow, shows the agricultural area, which decreased to 9% in 2021, and can be seen in the bottom-left corner of the clustered image in Figure 6.

From the above real-world application, we have observed that our algorithm for selecting the number of clusters, k, is successful. The reason is that it clearly chooses and divides the images into meaningful groups and, most importantly, it can identify new objects in the same image over time. For example, we quantified in percentages whether the groups change over a period of time, and our algorithm clearly calculates these changes. Moreover, it clustered the solar panels that were installed on roofs in 2021, which were not available in 2018.

4.3. Discussion

Clustering validity indices (CVIs) are utilized to validate the clustering results and find the correct number of clusters in a dataset. The CVIs are intended to indicate the intensity of separation and/or compactness among clusters. However, the terms compactness and separability are contradictory because the association between the clusters regarding compactness is generally positive while, with separation, it is negative. Often, CVIs use the same methodology to calculate compactness, which is the displacement between every data point and its cluster centroids. Apart from numerous problems related to compactness and separation, the evaluation of CVIs is mostly dependent upon the behavior of the datasets used for experimentation. As far as high-dimensional datasets (satellite images/remote sensing) are concerned, where cluster overlapping can exist significantly, the same quality for such datasets must be considered when using CVIs.

In light of the experiments on the synthetic datasets as well as real-world datasets, it was clear that most of the CVIs performed poorly (see Tables 3 and 4). In most cases, the CVIs (e.g., gap, pseudot2 and Duda) were found to underestimate the number of clusters. This is because of the fact that they are constructed on the idea/presumption that clusters are situated far away from each other and the affinity of each object to its group is at the maximum as compared to other clusters. Consequently, high compactness can be observed to merge groups into very few groups; therefore, the mixing of groups leads to ambiguity and a loss of correct and realistic information.

Some CVIs (e.g., CH, silhouette and DB) gave better performance when handling a smaller number of clusters but were far away from satisfactory in determining the true number of clusters. Apart from this, some indices (e.g., KL, Dunn, C-index and Rubin) overestimated the number of clusters with regard to its true number. As a result, vast separation could be observed when splitting groups into too many groups, which leads to redundant information. Thus, overestimation and underestimation of the number of clusters severely affect the clustering output. Remote sensing datasets are generally overlapped between clusters. Thus, most CVIs are not applicable for such remote sensing datasets when clusters are overlapped significantly.

In light of the challenges faced in calculating the true number of clusters, effort was made in the proposed method to tackle this issue. It was found in the simulation study that the proposed method yielded good results and performed better as compared to other existing CVIs (see Figure 3). Similarly, in the real-world datasets, the success rate of the proposed method was also high (see Figure 4). On the basis of good performance, the proposed method was applied to two different satellite images of the Islamabad region and its surroundings in order to determine the actual number of clusters. However, the proposed method, in the case of big data, is computationally costly due to its iterative nature. Thus, to tackle a high-dimensional dataset (such as remote sensing), we determined the best number of centroids (deliberately over-clustering) to shrink the complex data by utilizing the *k*-means algorithm [44]. Therefore, this step reduces the computational time and is particularly appropriate for handling multifaceted remote sensing data.

The *k*-means clustering algorithm was applied to the same images in order to determine various groups to achieve the main objective, such as forestry, urban areas, etc. (see Figures 5 and 6). Secondly, it was also examined whether, for example, the percentage of deforestation and urbanization was altered over a period of time or not.

5. Conclusions

The *k*-means clustering method is one of the most critical unsupervised learning techniques, which has been widely used in different fields due to its simplicity and efficiency. When data are large in size, clustering provides meaningful groups that can be processed and analyzed in different ways. Besides its simplicity, *k*-means has good potential to deal with large-scale datasets. Many variants of *k*-means have been put forward continuously by researchers due to its accuracy and efficacy.

In the *k*-means algorithm, the automatic estimation of the ideal number of clusters *k* in a dataset is a difficult and challenging task. Finding the best value of *k* has been a critical and interesting area of research for data analysts. This research study has focused on solving the uncertainty of the ideal number of clusters. Here, a unique procedure is proposed for the determination of the true value of *k* in an automatic manner. The performance accuracy of the proposed method was assessed on synthetic and real datasets. The experimental results show that the performance of all the existing CVIs is poor when detecting the true number of groups. It seems that the size of the cluster affects the percentage of the results, while the dimensions have no effect on the clustering results. However, the proposed method produced better segmentation results consistently in almost all situations. Moreover, based on the better performance, the proposed method was also applied to real satellite images of the Islamabad region in order to identify deforestation and urbanization data for future planning and development. In order to highlight the limitations and drawbacks of the proposed method, a more exhaustive assessment is required.

For future work, the proposed algorithm will be mainly designed to cluster objects and not attributes, so that improvements can be made in clustering attributes instead of objects. Similarly, to scale the *k*-means clustering algorithm using the quantization step, more research work is required on the best size of quantized data to represent the overall data, because the size of quantized data makes a significant contribution to the clustering of large datasets.

Author Contributions: Conceptualization, I.A., D.M.K., A.U.R. and Z.K.; methodology, M.S. and J.-G.C.; software, I.A., D.M.K., A.U.R. and Z.K.; validation, M.S. and J.-G.C.; formal analysis, I.A., D.M.K., A.U.R. and Z.K.; investigation, M.S. and J.-G.C.; resources, I.A., D.M.K., A.U.R. and Z.K.; data curation, M.S. and J.-G.C.; writing—original draft preparation, I.A., D.M.K., A.U.R. and Z.K.; writing—review and editing, M.S. and J.-G.C.; visualization, I.A., D.M.K., A.U.R. and Z.K.; supervision, M.S. and J.-G.C.; project administration, I.A., D.M.K., A.U.R. and Z.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Basic Science Research Program through the National Research Foundation (NRF) of Korea funded by the Ministry of Education under Grant 2018R1D1A1B07048948.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict to interest.

References

1. Caraka, R.E.; Chen, R.C.; Huang, S.W.; Chiou, S.Y.; Gio, P.U.; Pardamean, B. Big data ordination towards intensive care event count cases using fast computing GLLVMS. *BMC Med. Res. Methodol.* **2022**, *22*, 77.
2. Bhadani, A.K.; Jothimani, D. Big data: Challenges, opportunities, and realities. In *Effective Big Data Management and Opportunities for Implementation*; IGI Global: Hershey, PA, USA, 2016; pp. 1–24.
3. Fahad, A.; Alshatri, N.; Tari, Z.; Alamri, A.; Khalil, I.; Zomaya, A.Y.; Foufou, S.; Bouras, A. A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE Trans. Emerg. Top. Comput.* **2014**, *2*, 267–279. [CrossRef]
4. Silipo, R.; Adae, I.; Hart, A.; Berthold, M. *Seven Techniques for Dimensionality Reduction*; KNIME: Zurich, Switzerland, 2014; pp. 1–21.
5. Martín-Fernández, J.D.; Luna-Romera, J.M.; Pontes, B.; Riquelme-Santos, J.C. Indexes to Find the Optimal Number of Clusters in a Hierarchical Clustering. In Proceedings of the International Workshop on Soft Computing Models in Industrial and Environmental Applications, Seville, Spain, 13–15 May 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 3–13.
6. Tang, Y.; Ren, F.; Pedrycz, W. Fuzzy C-means clustering through SSIM and patch for image segmentation. *Appl. Soft Comput.* **2020**, *87*, 105928. [CrossRef]
7. Zhang, Y.; Bai, X.; Fan, R.; Wang, Z. Deviation-Sparse Fuzzy C-Means With Neighbor Information Constraint. *IEEE Trans. Fuzzy Syst.* **2019**, *27*, 185–199. [CrossRef]
8. Zhou, S.; Xu, Z. A novel internal validity index based on the cluster centre and the nearest neighbour cluster. *Appl. Soft Comput.* **2018**, *71*, 78–88. [CrossRef]
9. Ye, F.; Chen, Z.; Qian, H.; Li, R.; Chen, C.; Zheng, Z. New approaches in multi-view clustering. In *Recent Applications in Data Clustering*; IntechOpen: London, UK, 2018; p. 195.
10. MacQueen, J. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Oakland, CA, USA, 27 December 1965–7 January 1966; Volume 1, pp. 281–297.
11. Jain, A.K. Data clustering: 50 years beyond K-means. *Pattern Recognit. Lett.* **2010**, *31*, 651–666. [CrossRef]
12. Maldonado, S.; Carrizosa, E.; Weber, R. Kernel penalized k-means: A feature selection method based on kernel k-means. *Inf. Sci.* **2015**, *322*, 150–160. [CrossRef]
13. Du, L.; Zhou, P.; Shi, L.; Wang, H.; Fan, M.; Wang, W.; Shen, Y.D. Robust multiple kernel k-means using l21-norm. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015.
14. Wang, S.; Gittens, A.; Mahoney, M.W. Scalable kernel k-means clustering with nystrom approximation: Relative-error bounds. *arXiv* **2017**, arXiv:1706.02803.
15. Liu, X.; Zhu, X.; Li, M.; Wang, L.; Zhu, E.; Liu, T.; Kloft, M.; Shen, D.; Yin, J.; Gao, W. Multiple kernel k-means with incomplete kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 1191–1204. [CrossRef]
16. Di, J.; Gou, X. Bisecting K-means Algorithm Based on K-valued Selfdetermining and Clustering Center Optimization. *J. Comput.* **2018**, *13*, 588–595. [CrossRef]
17. Kingrani, S.; Levene, M.; Zhang, D. Estimating the number of clusters using diversity. *Artif. Intell. Res.* **2017**, *7*, 15. [CrossRef]
18. Zhou, S.; Xu, Z.; Liu, F. Method for Determining the Optimal Number of Clusters Based on Agglomerative Hierarchical Clustering. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 3007–3017. [CrossRef]
19. Milligan, G.W.; Cooper, M.C. An examination of procedures for determining the number of clusters in a data set. *Psychometrika* **1985**, *50*, 159–179. [CrossRef]
20. Shafeeq, A.; Hareesha, K. Dynamic clustering of data with modified k-means algorithm. In Proceedings of the 2012 Conference on Information and Computer Networks, Singapore, 26–28 February 2012; pp. 221–225.

21. Hamerly, G.; Elkan, C. Learning the k in k-means. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2004; pp. 281–288.
22. Tibshirani, R.; Walther, G.; Hastie, T. Estimating the number of clusters in a data set via the gap statistic. *J. R. Stat. Soc. Ser. B* **2001**, *63*, 411–423. [CrossRef]
23. Feng, Y.; Hamerly, G. PG-means: Learning the number of clusters in data. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2007; pp. 393–400.
24. Ray, S.; Turi, R.H. Determination of number of clusters in k-means clustering and application in colour image segmentation. In Proceedings of the 4th International Conference on Advances in Pattern Recognition and Digital Techniques, Calcutta, India, 27–29 December 1999; pp. 137–143.
25. Gupta, N.; Ujjwal, R. An efficient incremental clustering algorithm. *World Comput. Sci. Inf. Technol. J* **2013**, *3*, 97–99.
26. Zhang, Y.; Mańdziuk, J.; Quek, C.H.; Goh, B.W. Curvature-based method for determining the number of clusters. *Inf. Sci.* **2017**, *415*, 414–428. [CrossRef]
27. Kodinariya, T.M.; Makwana, P.R. Review on determining number of Cluster in K-Means Clustering. *Int. J.* **2013**, *1*, 90–95.
28. Li, X.; Liang, W.; Zhang, X.; Qing, S.; Chang, P.C. A cluster validity evaluation method for dynamically determining the near-optimal number of clusters. *Soft Comput.* **2020**, *24*, 9227–9241. [CrossRef]
29. Shao, X.; Lee, H.; Liu, Y.; Shen, B. Automatic K selection method for the K—Means algorithm. In Proceedings of the 2017 4th International Conference on Systems and Informatics (ICSAI), Hangzhou, China, 11–13 November 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1573–1578.
30. Duda, R.O.; Hart, P.E. *Pattern Classification and Scene Analysis*; Wiley: New York, NY, USA, 1973; Volume 3.
31. Caliński, T.; Harabasz, J. A dendrite method for cluster analysis. *Commun. Stat. Theory Methods* **1974**, *3*, 1–27. [CrossRef]
32. Dunn, J.C. Well-separated clusters and optimal fuzzy partitions. *J. Cybern.* **1974**, *4*, 95–104. [CrossRef]
33. Hartigan, J.A. *Clustering Algorithms*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 1975.
34. Davies, D.L.; Bouldin, D.W. A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.* **1979**, *PAMI-1*, 224–227. [CrossRef]
35. Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [CrossRef]
36. Krzanowski, W.J.; Lai, Y. A criterion for determining the number of groups in a data set using sum-of-squares clustering. *Biometrics* **1988**, *44*, 23–34. [CrossRef]
37. Tou, J.T.; Gonzalez, R.C. *Pattern Recognition Principles*; Addison-Wesley Publishing Company: Boston, MA, USA, 1974.
38. Gordon, A. *Classification*; Chapman and Hall: New York, NY, USA, 1999.
39. Friedman, H.P.; Rubin, J. On some invariant criteria for grouping data. *J. Am. Stat. Assoc.* **1967**, *62*, 1159–1178. [CrossRef]
40. Hubert, L.J.; Levin, J.R. A general statistical framework for assessing categorical clustering in free recall. *Psychol. Bull.* **1976**, *83*, 1072. [CrossRef]
41. Dua, D.; Graff, C. *UCI Machine Learning Repository*; University of California Irvine: Irvine, CA, USA, 2017.
42. Guyon, I.; Von Luxburg, U.; Williamson, R.C. Clustering: Science or art. In *NIPS 2009 Workshop on Clustering Theory*; NIPS: Vancouver, BC, Canada, 2009; pp. 1–11.
43. Hijmans, R.J. Raster: Geographic Data Analysis and Modeling. R Package. 2021. Available online: https://CRAN.R-project.org/package=raster (accessed on 3 April 2012).
44. Ullah, I.; Mengersen, K. Bayesian mixture models and their Big Data implementations with application to invasive species presence-only data. *J. Big Data* **2019**, *6*, 29. [CrossRef]