Ibn Al-Haitham Journal for Pure and Applied Sciences
Journal homepage: jih.uobaghdad.edu.iq
PISSN: 1609-4042, EISSN: 2521-3407
IHJPAS. 2025, 38(2)

Finding Optimal Number of Clusters Using Heuristic Clustering Algorithms

Hanin Haqi Ismail[1*], Tareef Kamil Mustafa[2]
[1,2] Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq.
*Corresponding Author.

Received: 13 November 2024 Accepted: 4 February 2025 Published: 20 April 2025
doi.org/10.30526/38.2.4047

Abstract

The problem of estimating the number of clusters, k, is considered one of the major challenges for partition clustering. The k-means algorithm is a division-based clustering method where only objects are entered into a set of K, and the algorithm decides the number group initially. Still there is no specific way to estimate the best number for the cluster. The outcome of the clustering process can be better after being performed in more than one attempt. Therefore, the optimal count of iterations can be identified through a steady method in advance that can determine the time and number of rounds. So, the problem is finding the optimal number (k) in an easy and fast way. The aim of this paper is to use a genetic algorithm (GA) with a new objective function to determine the best number of clusters of different types of datasets. The fitness function optimizes the heterogeneity among clusters and homogeneity within each cluster. Utilizing the gap statistic equation, the optimal number of clusters is determined in four standard datasets.

Keywords: Clustering, K-means, Heuristic, Genetic Algorithm, Gap statistic.

1. Introduction

Data mining is a highly powerful technique for retracting information and is useful for a large amount of technical and commercial data. This involves using a certain number of methods and strategies to check the data with a specified number of classes, clusters, and associations. Technology and various other fields widely use clustering algorithms in many applications. For example, marketing, pattern recognition (PR), information technology (IT), information retrieval (IR), artificial intelligence (AI), image processing, psychology, biology, and bioinformatics (gene expression analysis). Several clustering techniques were proposed to work on a range of the data. Such clustering algorithms could be defined on a category basis: clustering of portioning algorithms, clustering of fuzzy algorithms, clustering algorithms grid-based, clustering algorithms of hierarchy, and clustering algorithms density-based (1). K-means is one clustering technique that may be employed. Clustering falls within the domain of partitioning algorithms. The K-means clustering algorithm divides objects into k groups. To accomplish this clustering, the value of k should be specified in advance. The next step is to calculate the cluster centroid

(2-4). Among the several broad clustering methods, K-means is perhaps the best used. It still, however, has certain intrinsic limitations. One of the most difficult issues when utilizing K-means is identifying the best cluster number symbolized by K (5), where K represents a positive integer. However, K-means has a significant drawback: it frequently converges to a suboptimal solution due to the huge clustering search space. Thus, evolutionary methods such as the genetic algorithm are appropriate for grouping tasks. A good GA examines the search space appropriately while also using superior solutions to obtain the globally optimum solution (6). A typical explanation of the clustering aim is to maintain homogeneity within each cluster while increasing variability between clusters. The GA method will stop the maximum number of generations is created or when the maximum number of objective function evaluations is reached. This paper suggests a new objective method to find the optimal number of clusters using a genetic algorithm (GA) and solve the problem of many traditional methods used to find the best K, as each method gives different results from those applied to the same dataset.

1.1. Literature Review

There is no specific rule for determining the appropriate cluster number. Due to these limitations, the researchers used the evolutionary methods to solve the clustering problem, as follows: El-Shorbagy et al. (7), research on combining K-means with the genetic method has executed clustering using this combined technique. It was discovered that genetic algorithms have the potential to split datasets into a variety of groups that were not recognized previously. By using label-based representation and K-means methods, which will enhance the offspring generated by cluster-based crossover, there will be no need to set the number of clusters. The clustering process is employed to get the central point, which decreases the complexity of the problem. A large-scale problem is divided into multiple minor issues, so these methods represent that GA is good for sub-problems to increase speed and performance while handling tiny-scale combinatorial optimization. Some of the proposed techniques work more effectively and efficiently in terms of complexity and converge to the global optimum. In Hruschka, E.R. and Ebecken, N.F. (8), the study provided a way to find the ideal number for clustering a dataset. It created a genetic algorithm to perform this task. A simple encoding approach that results in constant-length chromosomes is utilized. The homogeneity between and within each cluster is optimized by the objective method. Besides, the clustering genetic algorithm also determines the appropriate group number based on the Average Silhouette Width requirement. It also created genetic operators that are context-sensitive. Four examples are supplied to show the effectiveness of the suggested strategy. The efficacy of the suggested method is illustrated by presenting the results obtained from four different data sets: data that is generated randomly, data on breast cancer, data from Ruspini, and data from the Iris plant. In each simulation, a population of 20 genotypes was examined, which, in turn, reflected the use of 21 clusters. Besides, 200 is set to be the maximum number of generations. The dissimilarities between items were calculated using the Euclidean distance measure. It simulated ten tests on each dataset; the best objective function value was first found. In a study by Liu et al. (9), the so-called clustering of Automatic Genetics, which is basically genetic algorithm-based clustering, is presented in this paper approach to Genetic Clustering for Unknown K (AGCUK). The AGCUK technique could automatically give the cluster number to define the division of clustering. Davies-Bouldin's index is used to evaluate the cluster's accuracy. Real-world and simulated datasets are experimented with to demonstrate the performance of the AGCUK algorithm. In a study by Rahman and Islam (10), a novel GA-based clustering study strategy has

the capability of choosing the ideal number of clusters and genes using a selection approach of a novel initial population. The objective function and gene modification operation create high-quality cluster centers. The centers are then passed to K-means as initial seeds, resulting in an even higher quality clustering solution by enabling the original seeds to change as necessary. The results show that the approach outperforms five contemporary techniques on twenty natural data sets included in this study, according to six evaluation criteria. Roy and Sharma (11) described a clustering technique based on the Genetic K-means paradigm that works well with data that contains both numerical and categorical variables. This study proposed a modified cluster center description to overcome the genetic K-means algorithm's numeric data constraint and give a more complete definition of clusters. The performance of this method was investigated using benchmark data sets. Han and Xiao (12) explain the fundamental premise of the genetic algorithm, which is based on Darwin's theory of evolution's "survival of the fittest", describes the algorithm's primary characteristics, and analyzes its weaknesses. An enhanced adaptive genetic algorithm is developed based on the specific running phases of the genetic algorithm, to address its inadequacies. Finally, an example is provided for simulation. The simulation results demonstrated that the enhanced method has some advantages. In the current paper, we use the genetic algorithm with the negative equation of the gap statistic as the objective function to find the optimal number of clusters.

2. Materials and Methods

This section will present the types of clusters and algorithms used for them, as well as the algorithms used in this paper.

2.1. Clustering types

A clustering approach seeks to group comparable data items and allocate heterogeneous data to various clusters using an objective function, model, grid computing, or density computing. As a result, hundreds of articles include numerous clustering techniques. These approaches are divided into five types: partitioning, density-based, hierarchical, and grid-based methods. Each strategy has advantages and disadvantages (13).

1. **Hierarchical clustering:** It is a cluster analysis technique that aims to create a hierarchical cluster structure. A hierarchical clustering technique may be defined as a collection of standard (flat) clustering algorithms grouped in a tree form as the shape in **Figure 1.** which provides a dendrogram for visual interpretation. These techniques create clusters by recursively splitting the items in either top-down or bottom-up mode, so the number of clusters does not need to be provided beforehand. The advantage of this type is capturing clusters at different scales. However, large datasets are computationally costly, making it difficult to find the right number of clusters from the dendrogram; they cannot provide the (k) beforehand and are not suitable for huge datasets (14).

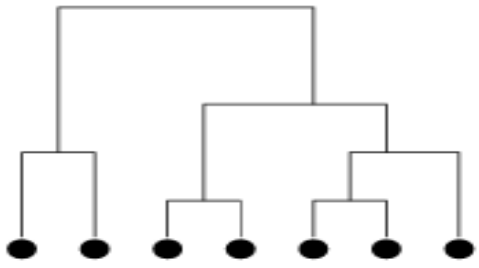

Figure 1. Hierarchical clustering (15)

2. **Density-based cluster**: It appears in geographical applications, such as clusters of points related to rivers, highways, electricity lines, or any linked form in image segmentation (16). The main advantage of density-based clustering is that the number of clusters is not required, and clusters of any shape may be discovered. Many density-based clustering methods have been developed. The most common is Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (17). This type is robust to noise and outliers; it identifies randomly formed clusters, and there is no need for a certain number of clusters to be specified. However, it is sensitive to the density parameter settings and unsuitable for clusters with varying densities. An example of density-based clusters is shown in **Figure 2**.

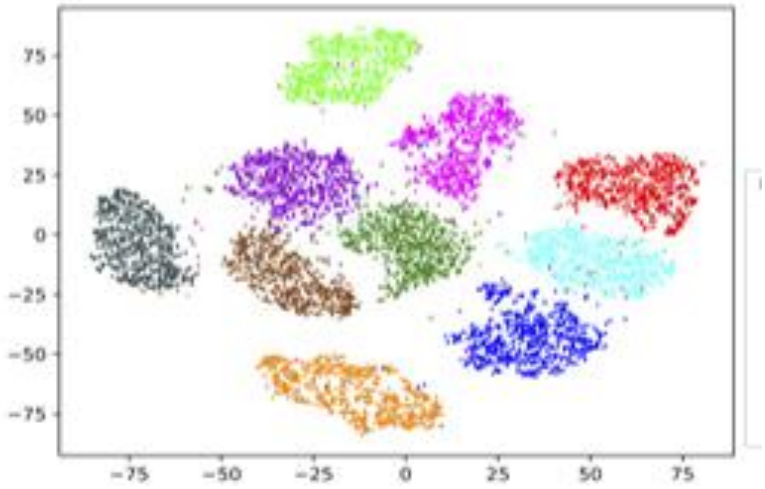

Figure 2. Density-based clusters (source (18))

3. **Gaussian Mixture Models (GMM):** This strong nonlinear model accurately fits varied data distributions and may successfully develop solutions of higher quality based on the distributions. **Figure 3.** shows the Expectation-Maximization (EM) algorithm with Gaussian Mixture Models (19). The Gaussian can model complex cluster shapes and provide probabilistic cluster assignments. However, it is sensitive to initialization values, it is possible to converge to local optimum values, and it is more computationally expensive than K-means.

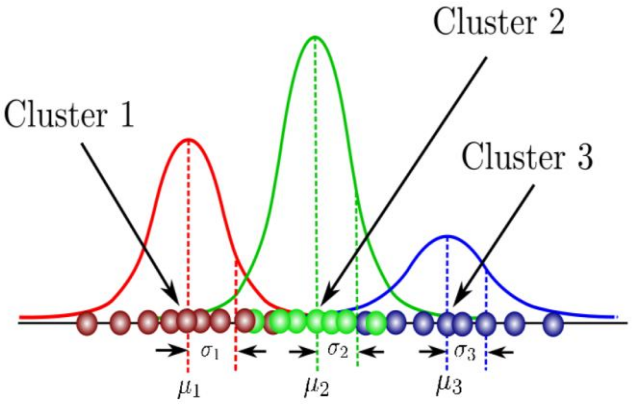

Figure 3. Gaussian Mixture Models

4. **Fuzzy clustering:** It is an important type of clustering where items or objects can belong to many clusters with varying degrees of membership. Instead of hard assignments, where each data point belongs to only one cluster, as shown in **Figure 4**. fuzzy clustering assigns membership values to data points indicating the degree to which they belong to each cluster. The Fuzzy C-Means (FCM) algorithm allows for soft clustering where data points can belong to multiple clusters. It handles noise and outliers better than traditional hard clustering methods and provides more flexibility in capturing the uncertainty in data. Interpretation of fuzzy clusters can be more complex than traditional hard clusters (20).

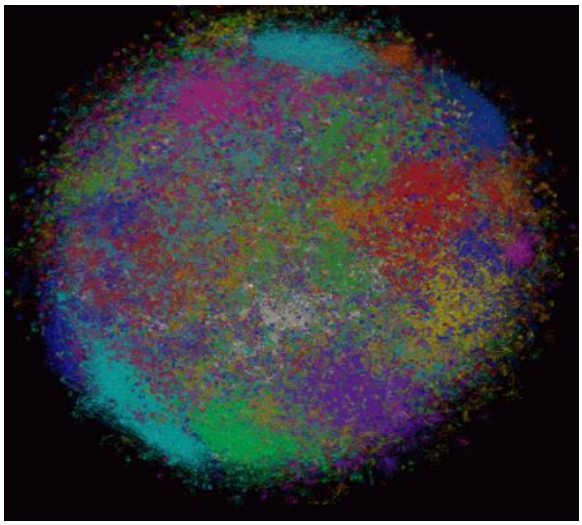

Figure 4. Fuzzy clustering

5. **Partition clustering:** As shown in **Figure 5**. the K-means algorithm is easy and simple to implement, computationally cost-effective, works well with spherical clusters, and is sensitive to the initial cluster centers. However, the number of clusters (k) must be set in advance, and it is not suitable for clusters with different sizes and densities.

Figure 5. Partition clustering

2.2. Clustering Algorithms

There are many clustering algorithms used to divide a specific dataset into a set of clusters, but the most important of them is the K-Means algorithm (21).

2.2.1. K-Means Algorithm

The K-means algorithm is a technique of dividing N items into K clusters based on the least distance between the center of the cluster and the other points. K-means uses various distance functions to measure the similarity among the objects. The most used functions are Euclidean Distance and Manhattan Distance (City Block Distance). **Figure 6.** shows the steps of the K-Means algorithm.

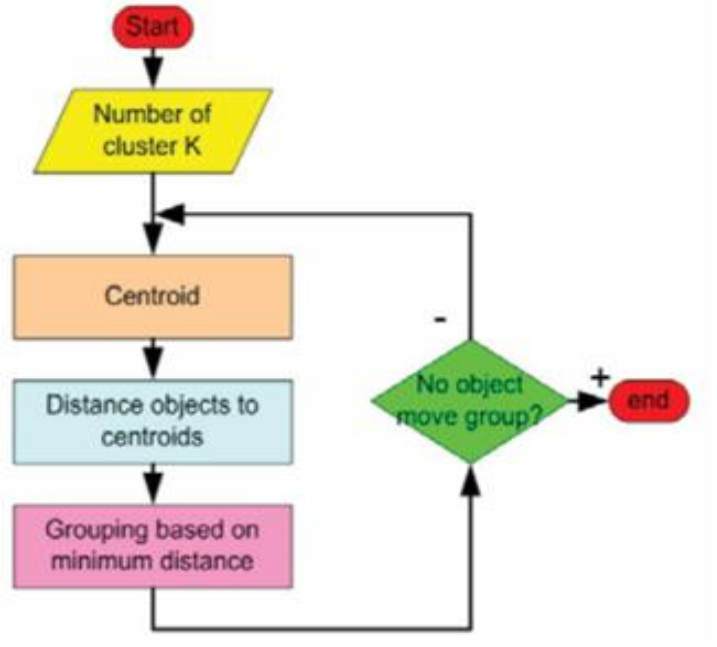

Figure 6. Flowchart of K-means (22)

2.2.2. Genetic Algorithm

Genetic algorithms are random search and optimization strategies driven by ideas of evolution and natural genetics. They are typically used to find solutions to optimization and search problems by mimicking the process of natural selection to evolve toward better solutions. Because this algorithm is used for optimization, it will use it to find the optimal number of clusters. **Figures 7** and **8.** show the steps of this algorithm.

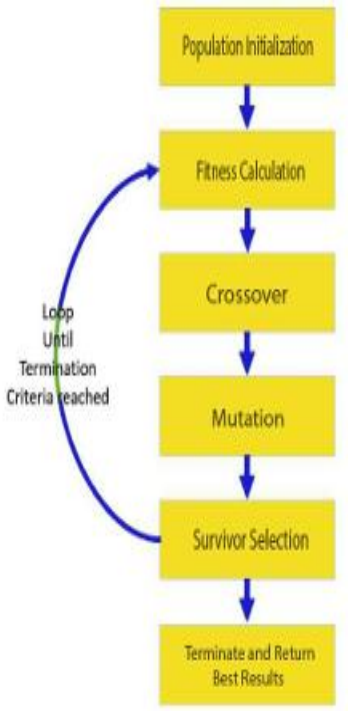

Figure 7. Genetic Algorithm (23)

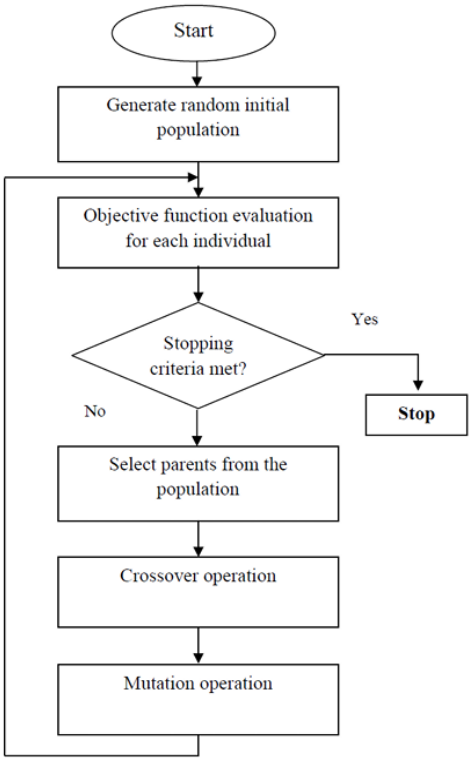

Figure 8. Flowchart of genetic (24)

2.3. How to find the optimal number of clusters

Finding the optimal number of clusters in a dataset is a crucial step in clustering analysis. Several methods can help predict the optimal number of clusters without relying solely on subjective judgment. Here are some common techniques used to find the correct number of clusters:

1. **Elbow Method**: It provides a visual cue to determine a reasonable number of clusters (25). It is simple and easy to implement and can calculate the Within-Cluster Sum of Squares (WCSS) for a range of different numbers of clusters and look for the "elbow point" where the rate of decrease in WCSS slows down. However, it may not always produce a clear elbow (smooth elbow).
2. **Silhouette coefficient**: The Silhouette Coefficient (SC) was established to quantify cluster density and separation (26). It is limited to the interval [-1, 1].
   It calculates the average silhouette score for a range of different numbers of clusters. The silhouette score indicates how similar an object is to its own cluster compared to other clusters:
   $$si = \frac{bi-ai}{\max(ai,bi)} \quad (1)$$
   Negative values for a point indicate that the instance is in the wrong cluster.
   Positive numbers imply accurate and dense clustering, with larger values suggesting a bigger ratio of bi values to (ai) values. This means that instance i is more similar to other instances in its own cluster than to instances in the next closest cluster. Values around zero suggest that the clusters are overlapping (27). It is, computationally, more expensive than the elbow method.
3. **Gap Statistic:** It compares the within-cluster dispersion for different numbers of clusters with what would be expected by random chance (28). The right number of clusters is where the gap statistic is maximized:

$$Gap\ (k) = \log\ (\mathrm{Wk}*) - \log(Wk) \quad (2)$$

- **Advantages**:
  i. Accounts for the randomness in the data.
  ii. Helps avoid overfitting.
- **Disadvantages**:
  i. Computationally intensive.
  ii. Requires generating random data for comparison.

4. **Davies–Bouldin Index Method:** Computes the average similarity measure between each cluster and its most similar one. The right number of clusters minimizes this index. It is a measure of the clustering quality and can be used to evaluate the clustering results. It is based on the average similarity between each cluster and its most similar one, and the average dissimilarity between cluster centers (29).

   The Davis-Bouldin Index is calculated using the following formula:

$$DB = \frac{1}{\mathrm{k}} \sum_{i=1}^{k} max\left(\frac{\sigma i+\sigma j}{d(ci,cj)}\right) \quad (3)$$

   Where : k is the number of clusters.
   σi is the average distance between pointsin cluster i and the centroid of cluster i.
   σj is the average dustance between points cluster j and the centriod of cluster j.
   D(ci,cj) is the distance between the centroids of clusters i and j.

- **Advantages**:
  i. Provides a measure of the clustering quality.
  ii. Encourages clusters to be well separated.
- **Disadvantages**:
  i. Requires distance/similarity measure between clusters.
  ii. Not as widely used as some other methods.

2.4. Our Proposal

This section describes the steps to discover the appropriate number of clusters. Genetic Algorithm (GA) implementation for finding the ideal number of clusters (k), the individual representation is based on the choice of the number of clusters (k) for each member of the population. Each individual (solution) in the population represents a potential solution, which is a possible value for the number of clusters. The Genetic Algorithm uses an integer encoding for the representation of individuals. Each gene in the chromosome represents a potential value for the number of clusters (k) within the specified range (k_min to k_max). The chromosome represents an individual in the population. Each gene in the chromosome represents a potential value for the number of clusters (k). The range of possible values for each gene is between k_min and k_max. The encoding is based on an integer representation where each gene corresponds to a specific value for k. The population is initialized with random integers within the range 1 to Len (gaps), representing potential values for k. **Each individual's fitness** is evaluated based on the gap statistic values. Individuals with better fitness values are selected for the next generation. During the crossover stage, selected individuals are combined to produce offspring with a mix of genes from the selected parents. Then, the mutation randomly changes some genes in the offspring to introduce diversity in the population. In the Gap Statistic strategy for determining the optimal number of clusters in a

dataset, generating appropriate reference data is crucial for comparing the within-cluster dispersion of the original data by using Uniform Distribution, which is the points that are randomly distributed within the same range as the original data, ensuring that reference data. The Gap Statistic is calculated as the difference between the log of the within-cluster dispersion for the original data and the expected log dispersion for the reference data. Let's denote the within-cluster dispersion for K clusters in the original data as (Wk), and the expected within-cluster dispersion for K clusters in the reference data as (Wk∗).

The fitness function is: max (Gap (K)= - (log(wk*)-log(wk)))

The Gap statistic is a measure used to compare the within-cluster dispersion of a given dataset with that of reference datasets. The Gap statistic formula takes the difference between the logarithm of the within-cluster dispersion of the reference datasets and the logarithm of the within-cluster dispersion of the original dataset. If the average within-cluster dispersion of the reference datasets is greater than the within-cluster dispersion of the original dataset, the gap statistic will be negative. So, by using the gap statistic equation in the genetic algorithm's objective function, we can find the optimal number of clusters for standard and synthetic datasets. The steps of the Genetic Algorithm are executed, and upon reaching the objective function, Algorithm 1 is executed to find the max value of the objective function, which is considered the optimal number of clusters.

Algorithm1: Steps to find the result of objective function

Input: Original Data
Output: best number of cluster
Set Initial Parameters:
    Define the maximum number of clusters (K_max)
    Choose the range of K values to evaluate (from 2 to K_max)
Initialize K = 2
K_max = 10
Loop:
    Fit K-Means: Apply K-means clustering to the data for K clusters
    Calculate Within-Cluster Dispersion (Wk): Compute the within-cluster dispersion for K cluster
    Generate Reference Data:
        Create reference data using a uniform distribution
        Fit K-means on the reference data for K cluster
        Calculate expected within-cluster dispersion (Wk*)
Compute Gap Statistic:
  Calculate Gap Statistic: Gap(K) = log(Wk*) - log(Wk)
Increment K
Check Termination Criteria:
If K reaches K_max, exit loop
End of Loop
Select Optimal K: Analyze Gap Statistic values then multiply the result with negative 1 to determine the optimal number of clusters
Calculate the standard error
End

3. Results and Discussion

In the beginning, we show the results of four standard datasets (the Iris dataset, the Wine dataset, the Digits dataset, and the Breast cancer dataset) by applying the genetic algorithm on

them and finding the optimal number of each dataset by utilizing the Gap statistic using Python code on Colab Google.

3.1. Experimental results

A Genetic Algorithm was implemented on standard and synthetic datasets to find the optimal number of clusters and show the results of four examples of standard datasets below.

Example 1.

This method is applied to the iris dataset. The dataset is commonly used for classification tasks, clustering algorithms, and data visualization techniques. It contains 150 instances, with 50 instances for each class. The dataset contains measurements of four features of three species of iris flowers (30). **Figure 9**. shows the plot of sepal width and length in the iris dataset.

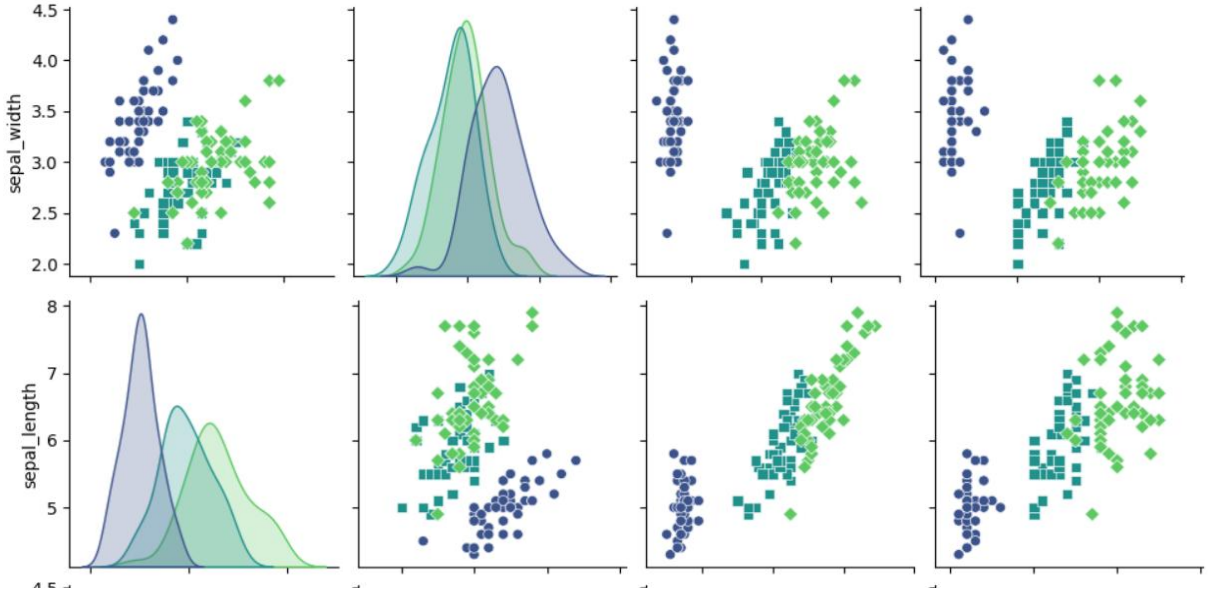

Figure 9. Plot of sepal width and length of iris dataset

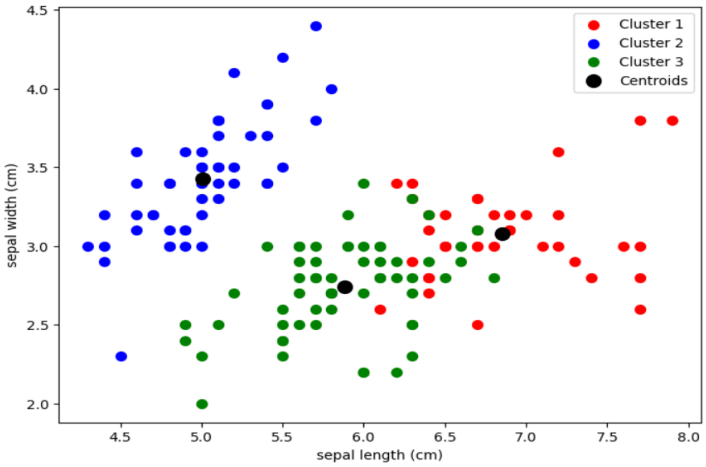

Figure 10. Distribution of iris dataset

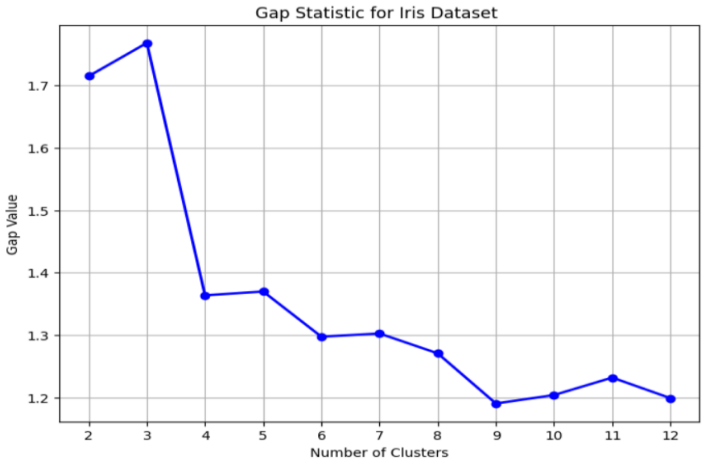

Figure 11. Gap statistic of iris dataset

Figure 10. shows the distribution of the iris dataset, and **Figure 11**. shows the plot of the gap statistic of the iris dataset. The ideal number of clusters is 3, which is determined by using the genetic algorithm, as shown in **Tables 1** and **2.**

Example 2.

The Wine Quality dataset is a classic dataset used for classification tasks. While it is primarily designed for classification rather than clustering, it is available from the UCI Machine Learning Repository and includes data for both red and white wines. The goal of the dataset is to model wine quality based on physicochemical tests. It contains 4,898 instances and 11 features: (1-fixed acidity, 2 - volatile acidity, 3 - citric acid, 4 - residual sugar, 5 – chlorides, 6 - free sulfur dioxide, 7 - total sulfur dioxide 8 - density 9 - pH 10 - sulphates 11 – alcohol). **Figure 12** shows the distribution of the wine dataset before finding the optimal number of clusters.

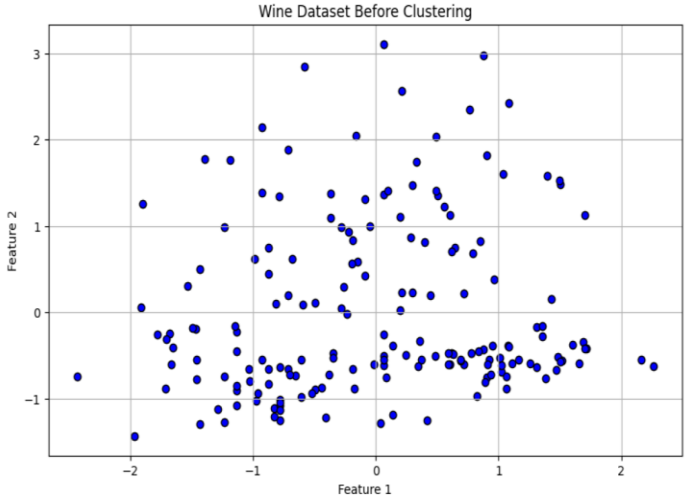

Figure 12. Distribution of wine dataset before finding the optimal number of clusters

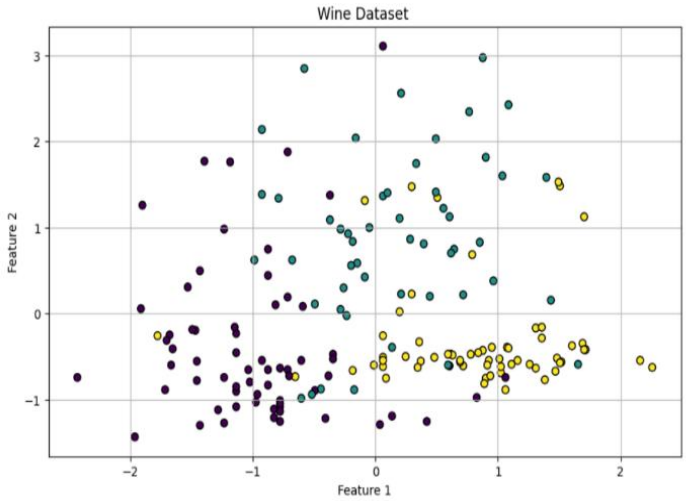

Figure 13. Distribution of wine dataset after finding the optimal number of clusters

Figure 13. shows the distribution of the wine dataset after finding the optimal number of clusters, and **Figure 14**. shows the plot of the gap statistic of the wine dataset. The optimal number of clusters of the wine dataset is 2, as shown in **Tables 1** and **5**.

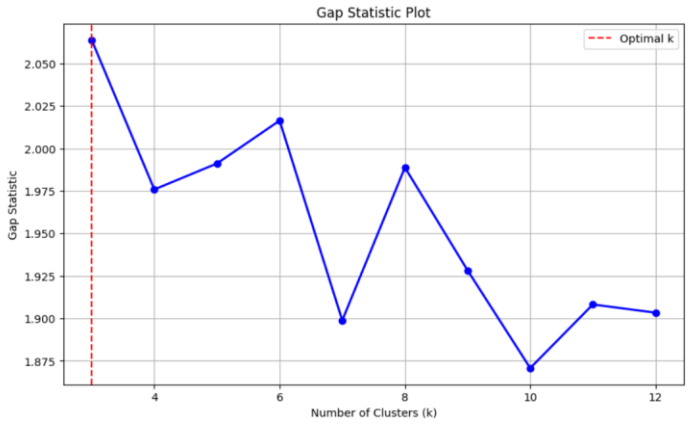

Figure 14. Gap statistic of wine dataset

Example 3.

The digits dataset is a commonly used dataset in machine learning for practicing classification algorithms. It consists of 8x8 pixel images of handwritten digits (0-9). Each sample in the dataset represents an image of a handwritten digit. Each pixel value is an integer ranging from 0 to 16, representing grayscale values. The dataset has a total of 1797 samples (31). **Figure 15.** shows the plot of the gap statistic of the Digits dataset.

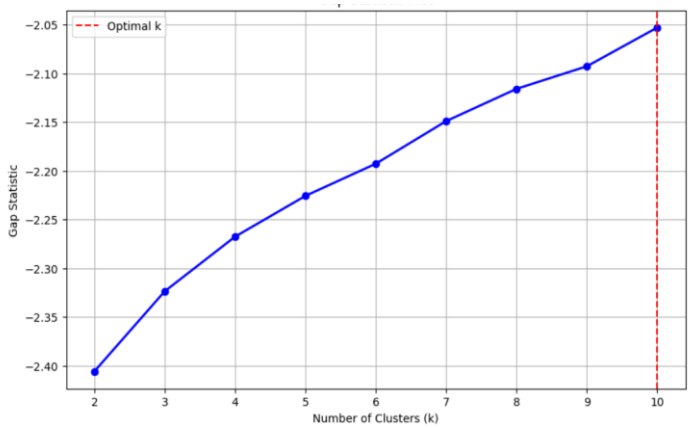

Figure 15. Gap statistic of Digits dataset

The optimal number of clusters of the digits dataset is 10, as shown in **Table 1**.

Example 4.

The Breast Cancer dataset is a common dataset used in machine learning for classification tasks, particularly in the context of predicting whether a tumor is benign or malignant based on features derived from images of cell nuclei. The dataset has features extracted from digital images of breast masses that describe the properties of cell nuclei found in the image. Each sample in the dataset represents information related to a specific breast mass. It contains 7,909 breast cancer images. The main job for this dataset is binary classification, where the objective is to guess from the given information whether a tumor is harmless or harmful (32). **Figure 16**. shows the distribution of the breast cancer dataset after clustering into two sets, as in **Figure 17**. That means the optimal number of clusters is 2, as shown in **Table 1**.

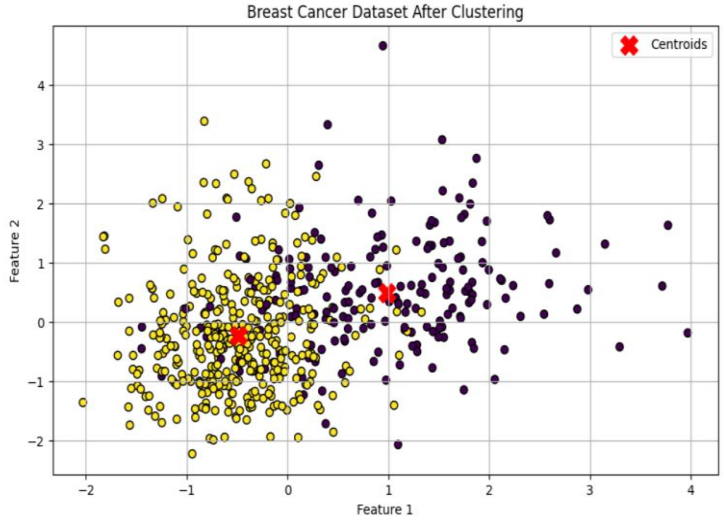

Figure 16. Distribution of Breast Cancer dataset

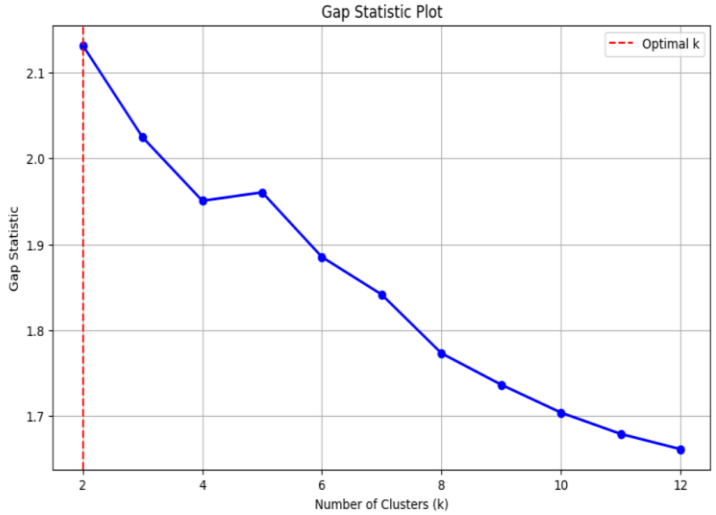

Figure 17. Gap statistic of Breast Cancer dataset

Table 1. will present the results. Determine the optimal number of clusters for four datasets within a specified cluster range. Other methods show the optimal number of clusters in **Tables 2, 3, 4, and 5.** So the comparison is clear. **Table 6.** illustrates the comparison clearly.

Table 1. Results of (2 to 10) range to find the optimal number of clusters for the four datasets

	2	3	4	5	6	7	8	9	10
Iris dataset	1.727	**1.773**	1.424	1.276	1.292	1.422	1.275	1.121	1.230
Wine dataset	**2.255**	2.029	2.027	1.999	2.000	1.989	1.969	1.933	1.917
Digits dataset	-2.405	-2.327	-2.268	-2.224	-2.193	-2.150	-2.119	-2.093	**-2.059**
Breast cancer dataset	**2.119**	2.014	1.943	1.970	1.907	1.838	1.759	1.738	1.702

Table 2. Finding the optimal number of clusters by using silhouette and Davies-Bouldin on the iris dataset

	2	3	4	5	6	7	8	9	10
Silhouette score	0.581	0.479	0.385	0.345	0.333	0.266	0.341	0.324	0.335
Davies – bouldin	0.593	0.789	0.869	0.943	0.993	1.124	0.990	0.977	0.975

Table 3. Finding the optimal number of clusters by using silhouette and Davies-Bouldin on the Breast cancer dataset

	2	3	4	5	6	7	8	9	10
Silhouette score	0.344	0.315	0.274	0.164	0.145	0.146	0.161	0.143	0.147
Davies - bouldin	1.309	1.539	1.492	1.429	1.503	1.499	1.555	1.502	1.511

Table 4. Finding the optimal number of clusters by using silhouette and Davies-Bouldin on the Digits dataset

	2	3	4	5	6	7	8	9	10
Silhouette score	0.386	0.104	0.106	0.102	0.104	0.111	0.127	0.130	0.135
Davies - bouldin	1.562	2.292	2.406	2.252	2.232	2.117	2.055	1.893	1.806

Table 5. Finding the optimal number of clusters by using silhouette and Davies-Bouldin on the wine dataset

Results	2	3	4	5	6	7	8	9	10
Silhouette score	0.265	0.284	0.254	0.183	0.168	0.172	0.162	0.173	0.139
Davies - bouldin	1.494	1.389	1.695	1.912	1.930	1.701	1.843	1.643	1.719

Table 6. The optimal number of clusters for four datasets by using the three ways

	Genetic algorithm	Silhouette score	Davies-bouldin
Iris dataset	3	2	2
Wine dataset	2	3	3
Breast cancer dataset	2	2	2
Digits dataset	10	2	2

Table 7. Four datasets and the values of the standard error for each optimal number of clusters

	Optimal number	Standard error
Iris dataset	3	0.711
Wine dataset	2	0.234
Breast cancer dataset	2	0.224
Digits dataset	10	0.0362

Table 7. shows the results of the optimal number of clusters for the four datasets and the standard error of each one of them.

3.2. Discussion

When using the silhouette score to find the optimal number of clusters, first apply the K-means algorithm to divide the dataset into the range of clusters. For example, when using the silhouette score to find the optimal number of clusters on a specific dataset, after the data was divided into 2 to 10 clusters, the biggest result (close to 1) represents the optimal number of clusters. When applied to multiple datasets, the results were correct, as with the breast cancer dataset. T**able 3** displays the results. Similarly, for the Davies Bouldin index, a smaller value indicates a more optimal clustering solution (approaching 0). This index is used to find the optimal number of clusters but does not give the right results in all tested datasets, as applied to the iris dataset as shown in **Table 2.** the minimum number is 2 clusters, but the right number is 3 clusters. So, a genetic algorithm was needed to find the best number of clusters across all datasets that were tested, compare these approaches, and find the right answers, as shown in **Table 6.** The right number of clusters of the iris dataset is 3, the best number of the wine quality dataset is 2, and the breast cancer dataset has 10 clusters of digits.

4. Conclusion

In clustering analysis, many procedures necessitate the designer to supply the number of clusters. The cluster algorithms may not have the ability to locate the appropriate number of clusters in advance. This paper suggests a new goal function for the genetic algorithm (GA)-based gap statistic clustering method based on the clustering partition. The genetic algorithm (GA) is an optimization technique that can present the optimal number of clusters for four different datasets: the Iris dataset, the Wine dataset, the Digits dataset, and the Breast **C**ancer dataset). The genetic algorithm (GA) with a new objective function found the right number of clusters in four datasets.

Acknowledgment

Thanks to the Department of Computer Sciences, College of Sciences, University of Baghdad, for their support and encouragement.

Conflict of Interest

The authors declare that they have no conflicts of interest.

Funding

There is no funding for this research.

References

1. Zhu E, Zhang Y, Wen P, Liu F. Fast and stable clustering analysis based on Grid-mapping K-means algorithm and new clustering validity index. Neurocomputing. 2019;363:149-70. https://doi.org/10.1016/j.neucom.2019.07.048
2. Abdullah D. Determining a Cluster Centroid of Kmeans Clustering Using Genetic Algorithm. International Journal of Computer Science and Software Engineering (IJCSSE). 2015;4(6):160-4.
3. Punhani A, Faujdar N, Mishra KK, Subramanian M. Binning-based silhouette approach to find the optimal cluster using K-means. IEEE Access. 2022;10:115025-32. https://doi.org /10.1109/ACCESS.2022.3215568
4. Sultan Alalawi SJ, Mohd Shaharanee IN, Mohd Jamil J. Clustering student performance data using k-means algorithms. Journal of Computational Innovation and Analytics (JCIA). 2023;2(1):41-55. https://doi.org/10.32890/jcia2023.2.1.3
5. Yang J, Lee J-Y, Choi M, Joo Y, editors. A new approach to determine the optimal number of clusters based on the gap statistic. International Conference on Machine Learning for Networking; 2019; lecture notes in computer science, vol 12081: Springer, Cham. https://doi.org/ 10.1007/978-3-030-45778-5_15
6. Mor M, Gupta P, Sharma P. A genetic algorithm approach for clustering. Int J Eng Comput Sci. 2014;3(6):6442-7.
7. El-Shorbagy MA, Ayoub A, Mousa A, El-Desoky I. An enhanced genetic algorithm with new mutation for cluster analysis. Computational statistics. 2019;34:1355-92. https://doi.org/10.1007/s00180-019-00871-5
8. Hruschka ER, Ebecken NF. A genetic algorithm for cluster analysis. Intelligent data analysis. 2003;7(1):15-25. https://doi.org/10.3233/IDA-2003-7103
9. Liu, Y.; Ye, M.; Peng, J.; Wu, H. Finding the Optimal Number of Clusters Using Genetic Algorithms. In Proceedings of the 2008 IEEE Conference on Cybernetics and Intelligent Systems; 2008; pp. 1325–1330. https://doi.org/10.1109/ICCIS.2008.4670864
10. Rahman MA, Islam MZ. A hybrid clustering technique combining a novel genetic algorithm with K-Means. Knowledge-Based Systems. 2014;71:345-65. https://doi.org/10.1016/j.knosys.2014.08.011
11. Roy DK, Sharma LK. Genetic k-means clustering algorithm for mixed numeric and categorical data sets. International Journal of Artificial Intelligence & Applications. 2010;1(2):23-8. https://doi.org/ 10.5121/ijaia.2010.1203
12. Han S, Xiao L, editors. An improved adaptive genetic algorithm. SHS web of conferences; 2022: EDP Sciences. https://doi.org/10.1051/shsconf/202214001044
13. Fahim A. K and starting means for k-means algorithm. Journal of Computational Science. 2021;55:101445. https://doi.org/ 10.1016/j.jocs.2021.101445
14. Nazari Z, Kang D, Asharif MR, Sung Y, Ogawa S, editors. A new hierarchical clustering algorithm. 2015 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS); 2015: IEEE. https://doi.org/10.1109/ICIIBMS.2015.7439517
15. Fluck E. Tangles and hierarchical clustering. SIAM Journal on Discrete Mathematics. 2024;38(1):75-92. https://doi.org/10.1137/22M1484936
16. Moulavi D, Jaskowiak PA, Campello RJ, Zimek A, Sander J, editors. Density-based clustering validation. Proceedings of the 2014 SIAM international conference on data mining; 2014: SIAM. https://doi.org/10.1137/1.9781611973440.96
17. Behadil SF, Mhalhal NK. Mobility Prediction Based on Deep Learning Approach Using GPS Phone Data. Ibn AL-Haitham Journal For Pure and Applied Sciences. 2024;37(4):423-38. https://doi.org/10.30526/37.4.3916

18. Ren Y, Wang N, Li M, Xu Z. Deep density-based image clustering. Knowledge-Based Systems. 2020;197:105841. https://doi.org/10.1016/j.knosys.2020.105841
19. Wang F, Liao F, Li Y, Wang H. A new prediction strategy for dynamic multi-objective optimization using Gaussian Mixture Model. Information Sciences. 2021;580:331-51. https://doi.org/10.1016/j.ins.2021.08.065
20. Mahmoudi MR, Baleanu D, Mansor Z, Tuan BA, Pho K-H. Fuzzy clustering method to compare the spread rate of Covid-19 in the high risks countries. Chaos, Solitons & Fractals. 2020;140:110230. https://doi.org/10.1016/j.chaos.2020.110230
21. Amma NG N, Amma NG B. Pclusba: A novel partition clustering-based biometric authentication mechanism. IETE Journal of Research. 2024;70(1):467-72. https://doi.org/10.1080/03772063.2022.2108917
22. Kapil S, Chawla M, Ansari MD, editors. On K-means data clustering algorithm with genetic algorithm. 2016 Fourth international conference on parallel, distributed and grid computing (PDGC); 2016: IEEE. https://doi.org/10.1109/PDGC.2016.7913145
23. Haldurai L, Madhubala T, Rajalakshmi R. A study on genetic algorithm and its applications. Int J Comput Sci Eng. 2016;4(10):139-43.
24. Zeebaree DQ, Haron H, Abdulazeez AM, Zeebaree S. Combination of K-means clustering with Genetic Algorithm: A review. International Journal of Applied Engineering Research. 2017;12(24):14238-45.
25. Maulana I, Roestam R. Optimizing KNN Algorithm Using Elbow Method for Predicting Voter Participation Using Fixed Voter List Data (DPT). Jurnal Sosial Teknologi. 2024;4(7):441-51. https://doi.org/10.59188/jurnalsostech.v4i7.1308
26. Vardakas G, Papakostas I, Likas A. Deep clustering using the soft silhouette score: Towards compact and well-separated clusters. arXiv preprint arXiv:240200608. 2024. https://doi.org/10.48550/arXiv.2402.00608
27. Layton R, Watters P, Dazeley R. Evaluating authorship distance methods using the positive Silhouette coefficient. Natural Language Engineering. 2013;19(4):517-35. https://doi.org/ 10.1017/S1351324912000241
28. Sagala NT, Gunawan AAS. Discovering the optimal number of crime cluster using elbow, silhouette, gap statistics, and nbclust methods. ComTech: Computer, Mathematics and Engineering Applications. 2022;13(1):1-10. https://doi.org/ 10.21512/comtech.v13i1.7270
29. Xiao J, Lu J, Li X. Davies Bouldin Index based hierarchical initialization K-means. Intelligent Data Analysis. 2017;21(6):1327-38. https://doi.org/10.3233/IDA-163129
30. Gupta T, Panda SP. A comparison of k-means clustering algorithm and clara clustering algorithm on iris dataset. International Journal of Engineering & Technology. 2018;7(4):4766-8. https://doi.org/10.14419/ijet.v7i4.21472
31. Seewald AK. Digits-a dataset for handwritten digit recognition. Austrian research institut for artificial intelligence technical report, Vienna (Austria). 2005;7.
32. Spanhol FA, Oliveira LS, Petitjean C, Heutte L. A dataset for breast cancer histopathological image classification. Ieee transactions on biomedical engineering. 2015;63(7):1455-62. https://doi.org/10.1109/TBME.2015.2496264