Eur. J. Stat. 5 (2025) 8
doi: 10.28924/ada/stat.5.8

Optimal Cluster Determination in K-Means Using Gap Statistic Analysis Across Diverse Datasets

Iliyas Karim Khan[1*], Hanita Binti Daud[1], Nooraini Binti Zainuddin[1], Rajalingam Sokkalingam[1], Mudasar Zafar[2], Soofia Iftikhar[3], Agha Inayat[4], Atta Ullah[1], Abdul Museeb[1]

[1]Fundamental and Applied Science Department, Universiti Teknologi PETRONAS, Perak 32610, Malaysia

[2]School of Mathematics Acturail and Quantitative Studies (SOMAQS), Asia Pacific University of Technology & Innovation (APU), 57000 Bukit Jalil, Malaysia

[3]Department of Statistics Shaheed Benazir Bhutto Women University, LARAMA, Charsadda Road, Peshawar, Pakistan

[4]University of Malakand Chakdara Peshawar, Khyber Pakhtunkhwa, Pakistan

*Correspondence: iliyas_22008363@utp.edu.my

Abstract. Clustering is a fundamental technique in unsupervised machine learning, where selecting the optimal number of clusters (ONC) remains a critical challenge, particularly for datasets with diverse characteristics. The Gap Statistic is a widely adopted method for determining ONC in K-means clustering, yet its performance is influenced by dataset size, feature complexity, and computational efficiency. This study systematically evaluates the accuracy, execution time, and coefficient of determination (R^2) of the Gap Statistic across four distinct datasets sourced from GitHub: Well Log, Time Series, Iris, and Hitters. These datasets vary in size, structure, and domain, providing a comprehensive assessment of the method's robustness. The Iris dataset exhibited the highest accuracy (87.25%) with an R^2 of 0.95, demonstrating the Gap Statistic's superior clustering capability in well-structured datasets. The Time Series dataset followed closely, achieving 68.47% accuracy and $R^2 = 0.88$, reflecting moderate reliability. Conversely, the Well Log dataset attained only 57.98% accuracy ($R^2 = 0.66$), while the Hitters dataset performed the worst, with 48.41% accuracy and $R^2 = 0.53$, indicating poor clustering effectiveness. Notably, datasets with higher ONC values (8 clusters in Well Log and Hitters) exhibited prolonged execution times (2.45 sec and 2.41 sec, respectively), highlighting computational inefficiencies.

1. Introduction

Cluster analysis serves as a fundamental exploration tool widely utilized across diverse domains including biology, sociology, medicine, and business [1-5]. The main aim is to

Received: 2 Mar 2025.

Key words and phrases. K-means clustering; Gap statistic; Unsupervised learning; Cluster analysis; Optimal cluster number.

categorize a collection of data items, known as data points, into clusters based on their similarities. This involves assessing the resemblance between various data points using a designated distance measure. The fundamental concept revolves around aggregating points with minimal distance into the same cluster, while points in separate clusters exhibit greater distances from one another. Clustering techniques typically fall into three main categories: Distance-based, Density-based, and Hierarchical [6-8]. K-means, developed by MacQueen, is an unsupervised learning algorithm that is grounded in distance-based approaches [9]. Renowned for its widespread usage in cluster analysis, it offers a straightforward, recursive approach to allocating data points into clusters based on predefined similarity metrics. One of its key attributes is its linear time and space complexity. Furthermore, numerous variants of k-means exist, known as disk-based variants, as they operate without necessitating the presence of all data points in memory [10-12]. In the K-Means clustering algorithm, which utilizes Euclidean distance to measure similarity, the k data objects that demonstrate the highest separation from each other are deemed more representative than "k" data objects randomly selected [13-15]. This algorithm aims to organize specified objects into clusters, each representing a distinct class. By evaluating similarities among objects based on specific criteria, it addresses the clustering problem by iteratively refining attributes through an alternating fitting process. However, each iteration involves calculating distances, leading to reduced algorithm efficiency and increased processing time. To mitigate this, a simplified data structure is introduced to retain pertinent details across iterations, thereby optimizing subsequent iterations. This approach eliminates the need to compute the distance of every data point from each cluster center in every iteration, resulting in reduced algorithm runtime. Determining the optimal number of clusters is crucial in k-means clustering, as this algorithm requires the number of clusters to be predetermined. However, selecting the right number of clusters can be challenging, as different datasets exhibit varying data features. This complexity arises because the choice of clusters directly impacts the effectiveness and interpretability of the clustering results [16, 17]. Different methods have been devised to automatically determine the optimal number of clusters in k-means, with the Gap statistic emerging as a key method in recent developments. The Gap statistic relies on comparing the logarithm of the expected value of reference data with the logarithm of the original data. However, the presence of a reference dataset poses challenges for the Gap statistic in accurately selecting the optimal number of clusters for different datasets in k-means

clustering. To address this challenge, current research examines the performance of the Gap statistics in terms of accuracy and execution time using diverse datasets with varying features and sizes. The main aim of this study is to assess the efficacy of the Gap Statistic. The objectives are as follows:

1. Evaluate the performance of the Gap Statistic on large datasets to determine its suitability for such data sizes.
2. Investigate whether the Gap Statistic performs better when applied to datasets with varying numbers of features.
3. Analyze the performance of the Gap Statistic in terms of accuracy and execution time across different types of data.
4. Propose strategies to address any challenges encountered in utilizing Gap Statistic for optimal cluster selection in K-means clustering.

The findings reveal that Gap statistics may not be suitable for datasets with differing features when determining the optimal number of clusters in k-means. The remaining paper is organized as follows: Section II delves into related research work. Section III outlines the fundamental concepts of k mean clustering algorithm and Gap statistic, both of which are employed in the proposed approach. Section IV presents the results of the experimental study validating the efficiency of the proposed approach. Lastly, Section V concludes the paper by summarizing the proposed work.

2. Literature Review

A crucial challenge in cluster analysis involves determining the ideal number of clusters that best fits the data under examination. Lu Xin-guo et al. [18] introduced a gene clustering method based on the Most Similarity Tree (CMST) to effectively generate comprehensive global clusters. This method tackles the challenge of distinguishing between various combinations of similarity associations, including a parameter called λ. Their research findings demonstrate that CMST surpasses traditional clustering techniques like K-means and SOM. They suggest utilizing Gap statistics to ascertain the optimal similarity measure λ and advocate for an adaptive gene clustering approach named OS-CMST (Optimal Self-adaptive CMST). Unlike SOM and K-means, which require predefining the number of clusters, the OS-CMST algorithm dynamically determines both the relevant similarity measure threshold and the number of clusters. [19] focused on clustering multidimensional

mass data using density-based techniques within the MapReduce framework. They highlight the inadequacy of traditional clustering algorithms for efficiently handling modern, high-speed multidimensional data processing needs. Additionally, these algorithms often overlook the intrinsic multidimensional nature of the data. Consequently, their study introduces a novel approach to large-scale multidimensional data clustering, incorporating density and information entropy principles. Inspired by the DBSCAN clustering algorithm, their proposed algorithm aims to address these shortcomings and enhance clustering effectiveness. Introduced an enhanced gap statistics algorithm utilizing the area density statistics method. Their algorithm effectively handles problematic data points. Through observation, it reduces the computational complexity associated with iterative computations, leading to improved computational speed and reduced processing time [20]. They introduced an innovative hybrid clustering method named KHM-ABC, blending K-harmonic means with the ABC algorithm to achieve optimal clustering outcomes. Their results illustrate that this hybrid approach outperforms other algorithms in terms of cluster quality. KHM-ABC utilizes the artificial bee colony algorithm to optimize the K-harmonic means clustering, ensuring globally optimal solutions. Evaluation was conducted across diverse datasets including iris, wine, yeast, and spam, with cluster quality assessed via silhouette index scores. Comparative analysis was carried out against ABC, K-means, K-harmonic means, and PAM algorithms, demonstrating the superior performance of KHM-ABC. The value of k was predetermined during preprocessing, employing the gap statistics method and silhouette width method [21]. Unsupervised Machine Learning techniques were developed to uncover dataset structures without prior information. Validating these structures posed a challenge, with various validation indices proposed. However, few addressed time-dependent data. A new internal index based on Gap Statistic was developed for time series datasets. Modifications included distance measurement, medoid-based clustering, and phase space modeling using Dynamical System tools. Results showed the index accurately clustered chaotic time series [22]. Galaxies in clusters merged over time, explaining the presence of multiple luminous galaxies. Researchers measured the age of these systems using the luminosity gap, the difference in brightness between the top two galaxies. They estimated this gap's distribution in clusters based on dark matter halo mass. "Fossil" groups, where galaxies had merged significantly, were identified. Predictions suggested that 1%-3% of massive clusters and 5%-40% of groups were likely fossil systems. Comparing predictions with Sloan Digital Sky Survey C4 Catalog

data, researchers found agreement, validating theoretical merger probabilities for cluster scales [23]. Archana Singh et al. [11] applied the k-means methodology with three distinct distance metrics: Euclidean, Manhattan, and Minkowski. Through their comparative analysis, the study determined that k-means achieves optimal performance specifically when employing the Euclidean distance metric. [24] provides an in-depth examination of k-means and its primary characteristics. Additionally, the research delves into addressing the limitations of k-means and strategies for mitigation. Emphasized within the study is the importance of accurately estimating the appropriate number of clusters, a critical aspect of cluster analysis. Based on our previous research in big data clustering, the parallel K-Means algorithm has demonstrated remarkable efficiency, requiring minimal time for cluster construction, and boasting easy implementation. However, a drawback of this algorithm is its fixed number of clusters. Unlike traditional K-Means where cluster centers are determined based on data chunks in mappers, resulting in varying clusters across different runs for the same dataset, our work addresses this limitation. Our key contribution lies in automating the determination of the number of clusters formed by this algorithm, achieved through gap statistics evaluation criteria. Applying data mining clustering techniques in Big Data environments is challenging due to the vast volume of data and the complexity of clustering algorithms, which entail significant processing costs [25]. In clustering, objects are grouped based on similarities. K-means, a popular method, struggled to determine the optimal number of clusters (k). Despite attempts to solve this, it remained unresolved. A study introduced a technique enhancing the gap statistic method for selecting k, showing superior performance on diverse datasets. The method's adaptability to various clustering algorithms was noted, promising broader applicability beyond k-means [26]. Limited research addresses determining the number of unknown targets in open-world scenarios within generalized evidence theory (GET), leading to accuracy issues and complex implementations. To overcome this, a novel method combining Isolation Forest and Gap statistic with K-means for bidirectional analysis is proposed. This method completes the defective frame of discernment (FOD) by summing base and novel clusters, ensuring FOD integrity even with highly correlated sample features. Simulation experiments demonstrate its effectiveness and broad applicability [27, 28]. In the field of image watermarking, k-means clustering, and genetic algorithms were established techniques. While k-means clustering allocated pixels into clusters, it did not always achieve optimal results. Genetic algorithms, however, were

known for producing optimal watermarking solutions [29]. Introduced the Hybrid Capuchin Search Algorithm (HCSA) to enhance K-means clustering by overcoming local optima traps and initialization sensitivity. HCSA outperformed K-means and eight other meta-heuristics-based methods across 16 datasets [30]. In signal processing, k-means clustering had encountered challenges with non-spherical clusters. Self-Weighted Euler k-means (SWEKM) was introduced to integrate clustering and feature selection, and it outperformed existing methods on UCI datasets [31]. The paper modified K-means using fuzzy membership, gap statistics, and data density for better clustering in time-of-use tariff partitioning, showing improved performance [32]. The study addressed K-means' difficulty with non-spherical clusters by proposing NDP-K means, which efficiently identified arbitrary-shaped clusters using natural density peaks and graph distance, showing superior performance [33]. Proposed: DBSCAN and k-means combo detects/reduces high-density regions. Applied in fusion plasma simulation, showcasing adaptive reduction based on data density [34]. This letter emphasized the challenge of choosing the optimal k in k-means clustering and suggested alternatives to the "elbow method", urging educators and researchers to reconsider its use [35].

3. Methodology

3.1 K-mean Clustering Algorithm

K-means clustering is a frequently utilized unsupervised machine learning algorithm designed to divide a dataset into a fixed number of clusters, labeled as k. Its goal is to group data points into clusters based on their similarities, where each cluster is identified by its centroid. The process begins with setting initial centroids for the clusters, serving as reference points. Subsequently, the data points are grouped into their respective clusters based on proximity to these predetermined centroids [36-38]. This assignment involves multiple sequential steps as outlined below.

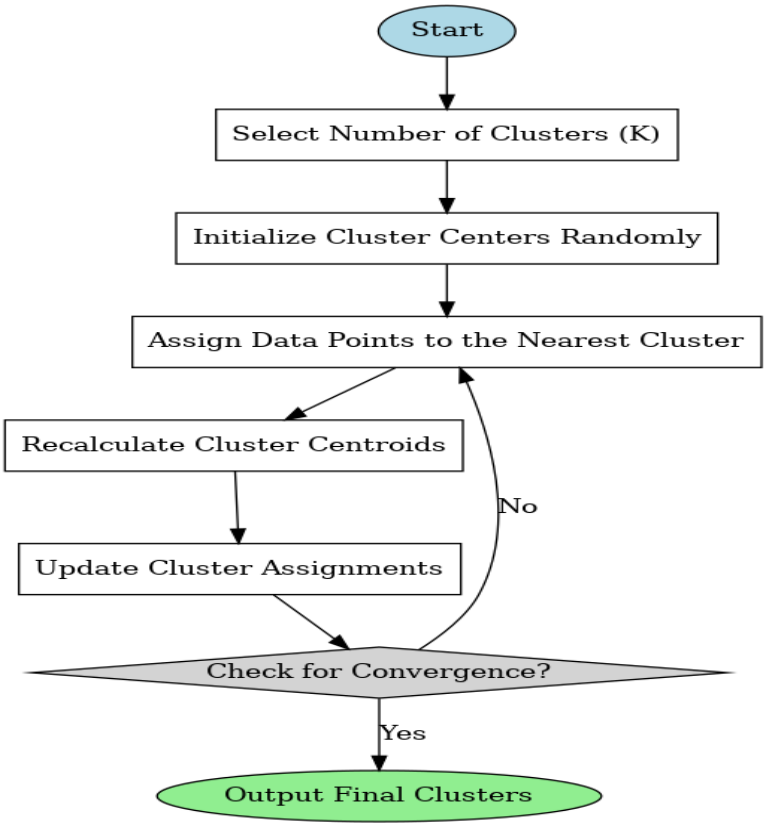


Fig. 1. Flow chart of k-mean clustering algorithm

The process begins with setting initial centroids for the clusters, serving as reference points. Subsequently, the data points are grouped into their respective clusters based on proximity to these predetermined centroids shown in Fig.1. This assignment involves multiple sequential steps as outlined below.

Algorithm: Pseudocode for K-Means Clustering

1. Start
2. Select the number of clusters (K)
3. Initialize K cluster centers randomly
4. Repeat until convergence:
   a. Assign each data point to the nearest cluster centre
   b. Compute new centroids by taking the mean of all data points assigned to each cluster
   c. Update cluster assignments based on new centroids
   d. Check for convergence:
      - If centroids do not change, stop
      - Otherwise, repeat steps a to c
5. Output final clusters
6. End

This pseudocode follows the iterative approach of K-Means clustering, ensuring that cluster centroids are updated until no further changes occur.

The iteration persists through steps 2 and 5 until stability is reached. Stability is achieved when the centroids exhibit minimal to no further change or after a specific number of iterations. Consequently, the outcome comprises clusters along with their individual centroids, signifying the arrangement of similar data points. This iterative method aims to reduce the total variance within clusters or the squared distances of data points to their respective centroids, ensuring the formation of coherent and distinct clusters. In the above step 2 we assigned the data point by using Eq. (1).

$$C_i = arg.min_j \parallel x_i - \mu_j \parallel^2 \quad (1)$$

Where C_i: cluster to which data points. x_i μ_j: centroid of clusters. $\parallel x_i - \mu_j \parallel^2$: Euclidean distance

Updating cluster centroid by applying the formula as in Eq. (2).

$$\mu_j = \frac{1}{\parallel C_j \parallel} \sum_{x_{i \subseteq} c_j} X_i \quad (2)$$

Where $\sum_{x_{i \subseteq} c_j} X_i$: summation of all data points in clusters j

The process will be iterative, and convergence will be achieved when the assignments and centroid stop changing or if a stopping criterion is reached.

3.2 Gap Statistic

The gap statistics, devised by Tibshirani, Walther, and Hastie in 2001, is a method utilized in cluster analysis to assess the ideal number of clusters present in a dataset. Its aim is to gauge the disparity between the dispersion within clusters and the expected dispersion derived from a null reference distribution [20, 39]. The K-means algorithm is used to determine the suitable number of clusters within a provided dataset by evaluating the sum of distances from each object to the cluster mean, termed dispersion. To compute the gap statistic, the algorithm generates several sample datasets from the original data and computes the mean dispersion of these samples. Each gap represents a logarithmic contrast between the mean dispersion of the reference datasets and that of the original dataset. Maximizing the gap involves selecting the minimum value of k [40].

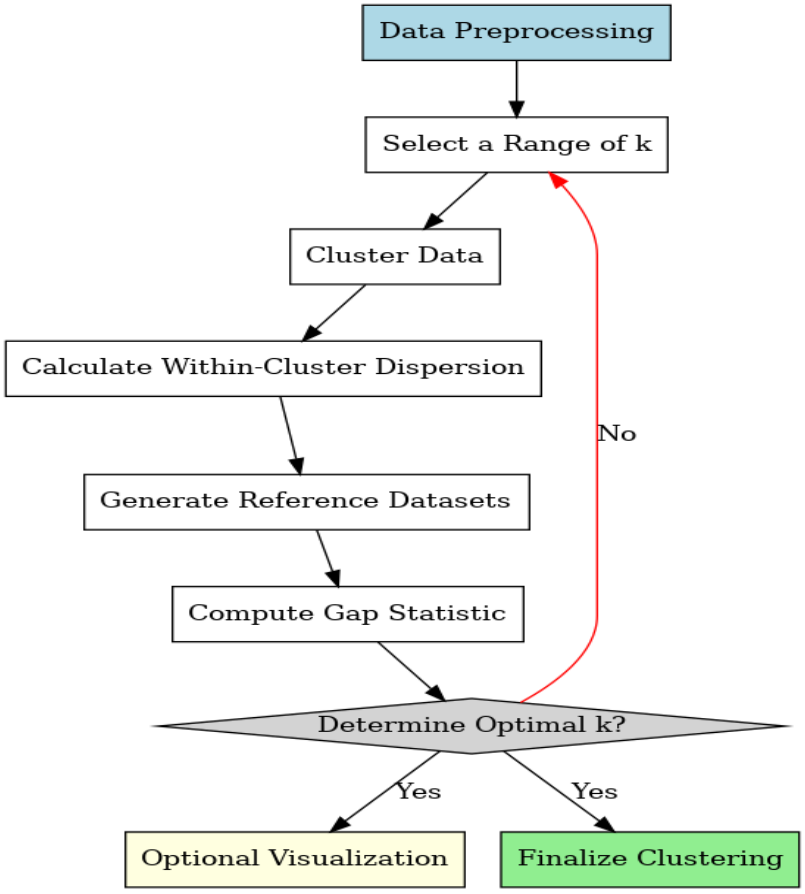


Fig.2. Gap Statistic

Pseudocode for calculation of Gap statistic Fig.2

Algorithm: Gap Statistic for Optimal Cluster Selection

1. Start
2. Data Preprocessing
3. Select a range of k values (number of clusters)
4. Repeat for each k:
   a. Apply clustering algorithm (e.g., K-Means)
   b. Compute within-cluster dispersion (Wk)
   c. Generate reference datasets (randomly distributed points)
   d. Compute within-cluster dispersion for reference datasets (Wk*)
   e. Calculate the Gap Statistic:
      Gap(k) = E[log(Wk*)] - log (Wk)
5. Determine the optimal k:
   a. If the Gap Statistic criterion is met, proceed
   b. Otherwise, adjust k and repeat
6. If visualization is required, generate plots (e.g., Gap Statistic curve)
7. Finalize clustering using the optimal k
8. End

This pseudocode follows the iterative process of selecting the optimal number of clusters using the Gap Statistic method, ensuring the best clustering structure.

The Gap Statistic is a clustering validation technique used to determine the optimal number of clusters (ONC) by comparing the within-cluster dispersion of observed data to that of a reference (random) dataset. The mathematical formulation of the Gap Statistic is given by:

$$Gap(k) = E_n\{\log(W_{k^*})\} - \log(W_k) \quad (3)$$

$$Gap(k) = \frac{1}{B}\sum_{b=1}^{B} logW_k^b - \log(W_k) \quad (4)$$

Where:

k is the number of clusters being evaluated. Wk is the total within-cluster variation for k clusters. $Wk*$ is the total within-cluster variation for a reference set of clusters. W_k^b: Within-cluster dispersion for the b-th reference dataset (randomly generated) and B: Number of bootstrapped reference datasets used for comparison. The optimal number of clusters is typically chosen as the value of k that maximizes the gap statistic. The proposed model consists of three main phases, depicted in Figure 3. The initial phase, referred to as the partitioning phase, focuses on directly handling large-scale data. Here, the data is divided into multiple chunks based on the available hardware resources. Upon completion of this phase, the extensive dataset is transformed into smaller datasets ready to be transferred to the mapper phase. In the mapper phase, which constitutes the second stage, these data chunks are received and distributed across a group of mappers. The primary task during this phase is to execute the k-means algorithm on each mapper. Consequently, the data chunks are clustered locally using the optimal number of clusters determined by the proposed optimized k-means algorithm. Finally, in the third phase, known as the reducer phase, local key-value pairs produced by each mapper are gathered and merged to form a global cluster center. Further elaboration on each phase will be provided in subsequent sections.

3.1 Partitioning Data

The large input dataset is divided among the mappers. Each map function receives input data chunks in the form of data points.

3.2 Optimized K-Means Clustering Approach

The K-means algorithm is used on each data chunk with varying cluster numbers, from 2 up to the maximum allowed. Gap Statistics clustering evaluation is employed to identify the optimal number of clusters for each dataset. Initially, distances are calculated as the total sum of Euclidean distances between pairs of data points within each cluster k.

3.3 Accuracy

The key metric for evaluation is accuracy, which gauges the proximity of a result to the true value and evaluates the performance of the proposed approach. Greater accuracy denotes superior clustering performance. It is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN \quad + FP + FN} \tag{5}$$

- True Positive (TP): Gap Statistic correctly identifies the optimal number of clusters (ONC).
- True Negative (TN): Correctly rejects non-optimal cluster numbers.
- False Positive (FP): Incorrectly suggests a non-optimal ONC (over/under-segmentation).
- False Negative (FN): Fails to detect the correct ONC.

The accuracy formula measures how well the Gap Statistic aligns with expected clustering results, with higher accuracy indicating better ONC selection and lower accuracy suggesting potential misclassification of clusters.

3.4 Time Taken

The second measure is the duration of execution, recorded in seconds. This time can vary across different machines due to varying configurations.

3.5 Speedup:

This metric represents the comparative performance of two methods addressing the same problem. It reflects the increase in execution speed between two similar tasks performed using different approaches. Speedup is utilized to evaluate the performance of the proposed approach,

$$Speedup = \frac{T_c}{T_p} \tag{6}$$

where Tc denotes the execution time of the current method, and Tp represents the execution time of the classical k-means.

4. Data Analysis

In this study, four extensive datasets were examined, sourced from the UCI repository, with their statistical characteristics outlined as follows:

4.1 Dataset

Different datasets, including well log, time series, iris, and hitters' data, are utilized to assess the performance of the Gap Statistic in selecting the optimal number of clusters in K-means. Each dataset is comprehensively explained in detail below table 1.

Well Log Data:

- The Delta-T (DT) measurements indicate an average travel time of 75.46 microseconds per foot, with a standard deviation of 8.96. The distribution is slightly right-skewed (skewness = 0.50) and exhibits fewer extreme values (kurtosis = -0.09). The range of DT values spans from 54.40 to 108.98 microseconds per foot.
- Gamma Ray (GR) readings show an average of 44.72 API units, with a wider variability as indicated by a higher standard deviation of 26.60. The distribution is right skewed with a longer tail (skewness = 1.08) and demonstrates more extreme values (kurtosis = 0.72). The range of GR values extends from 11.68 to 130.67 API units.

Time Series Data:

- Temperature data reveals a mean temperature of 8.33°C, with a standard deviation of 4.68. The distribution is right-skewed (skewness = 0.81) and displays fewer extreme values (kurtosis = -1.22). Temperature ranges from 4.21 to 15.83°C.
- Humidity levels have a mean of 70.14%, with a standard deviation of 14.08. The distribution is heavily right-skewed (skewness = 2.14) and very leptokurtic (kurtosis = 11.79), indicating a concentration of values around the mean. Humidity ranges from 30% to 150.7%.
- Wind speed data exhibits a mean of 1.32 meters per second and a high standard deviation of 5.95, indicating significant variability. The distribution is heavily right-skewed (skewness = 6.89) and extremely leptokurtic (kurtosis = 51.40). Wind speed ranges from 0.07 to 50.08 meters per second.
- General diffuse radiation has a mean of 164.63, with a wide standard deviation of 197.44. The distribution is right-skewed (skewness = 0.61) and exhibits fewer extreme values (kurtosis = -1.37). General diffuse radiation ranges from 0.03 to 498.8.

Iris Dataset:

- Sepal length has a mean of 5.84 cm and a standard deviation of 0.83, with a slight right skew (skewness = 0.31) and a platykurtic distribution (kurtosis = -0.55). Sepal length ranges from 4.3 to 7.9 cm.

- Sepal width has a mean of 3.05 cm and a standard deviation of 0.43, with a slightly right-skewed distribution (skewness = 0.33) and slightly leptokurtic (kurtosis = 0.29). Sepal width ranges from 2 to 4.4 cm.
- Petal length shows a mean of 3.76 cm and a standard deviation of 1.76, with a left-skewed distribution (skewness = -0.27) and platykurtic (kurtosis = -1.40). Petal length ranges from 1 to 6.9 cm.

Hitters Dataset:

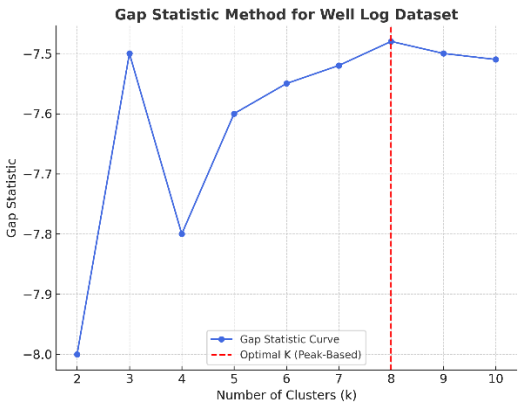
- The Hitters Dataset (abbreviated as "Hitters") contains baseball player performance metrics, including AB (At-Bats), H (Hits), HR (Home Runs), R (Runs), RBI (Runs Batted In), and BB (Walks). It offers insights into player batting abilities and contributions to team success.
- At-bats have a mean of 380.93, with a standard deviation of 153.41. The distribution is slightly left-skewed (skewness = -0.08) and platykurtic (kurtosis = -0.89). At-bats range from 16 to 687.
- Hits have a mean of 101.02, with a standard deviation of 46.45. The distribution is slightly right-skewed (skewness = 0.29) and platykurtic (kurtosis = -0.50). Hits range from 1 to 238.
- Home runs have a mean of 10.77, with a standard deviation of 8.71. The distribution is right-skewed (skewness = 0.90) and mesokurtic (kurtosis = 0.04). Home runs range from 0 to 40.
- Runs have a mean of 50.91, with a standard deviation of 26.02. The distribution is slightly right-skewed (skewness = 0.42) and platykurtic (kurtosis = -0.52). Runs range from 0 to 130.
- RBIs have a mean of 48.03, with a standard deviation of 26.17. The distribution is right-skewed (skewness = 0.61) and platykurtic (kurtosis = -0.30). RBIs range from 0 to 121.
- Walks have a mean of 38.74, with a standard deviation of 21.64. The distribution is right-skewed (skewness = 0.62) and platykurtic (kurtosis = -0.26). Walks range from 0 to 105.

Table 1. Statistical Summary of dataset

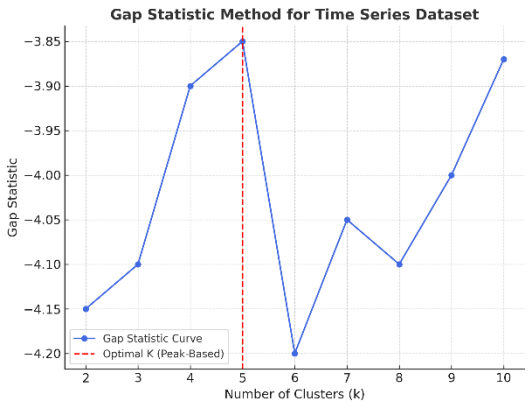
Dataset		Mean	S. D	Skewness	Kurtosis	Minimum	Maximum	Total
Well Log	DT	75.455	8.955	0.498	-0.09	54.402	108.97	2435
	GR	44.720	26.600	1.082	0.721	11.6842	130.66	
Time Series	Temperature	8.3250	4.680	0.805	-1.21	4.212	15.83	100
	Humidity	70.1449	14.08	2.138	11.79	30	150.7	
	WindSpeed	1.3201	5.945	6.89	51.40	0.073	50.08	
	GeneralDiffuse	164.63	197.4	0.608	-1.37	0.033	498.8	
Iris	SepalLength	5.8433	0.828	0.31	-0.55	4.3	7.9	150
	SepalWidth	3.054	0.433	0.33	0.29	2	4.4	
	PetalLength	3.758	1.76	-0.274	-1.40	1	6.9	
Hitters	AtBat	380.92	153.45	-0.07	-0.88	16	687	322
	Hits	101.02	46.45	0.291	-0.50	1	238	
	HmRun	10.770	8.709	0.904	0.03	0	40	
	Runs	50.90	26.0	0.41	-0.51	0	130	
	RBI	48.02	26.16	0.60	-0.30	0	121	
	Walks	38.74	21.63	0.620	-0.25	0	105	

4.2 Optimal number of clusters in k means

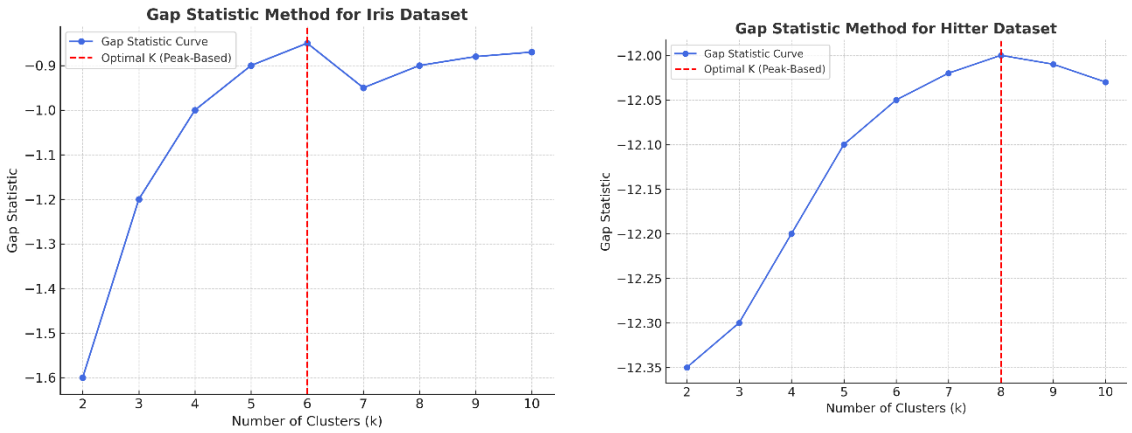
In K-means clustering, the gap statistic is a widely used method for determining the optimal number of clusters. In the plot below, the curve exhibiting a peak indicates the optimal number of clusters, as determined by this criterion.



(a)



(b)



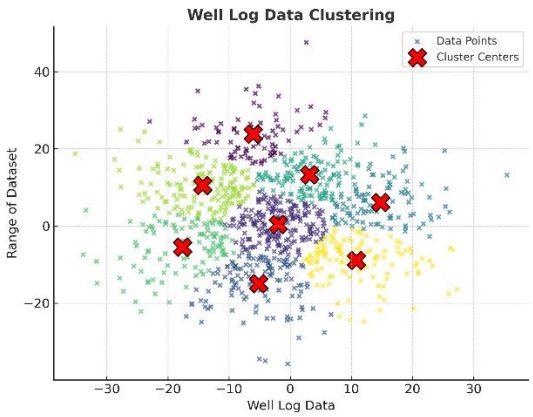
(c) (d)

Fig.2 Selection ONC by using GS

In Fig 2(a), the optimal number of clusters (ONC) for the Well Log dataset is determined to be 8, as identified using the Gap Statistic method. Fig 2(b) illustrates the ONC for the Time Series dataset, which is found to be 5, indicating a more compact clustering structure compared to the Well Log dataset. Moving to Fig 2(c), the Iris dataset exhibits an ONC of 6, suggesting a well-defined clustering pattern that aligns with the inherent structure of the dataset. Finally, Fig 2(d) demonstrates that the ONC for the Hitters dataset is 8, like the Well Log dataset. This suggests that a higher number of clusters is necessary to capture the variations within the dataset. These Fig collectively highlight the variability in the optimal number of clusters across different datasets, emphasizing the importance of selecting an appropriate clustering validation method based on dataset characteristics.

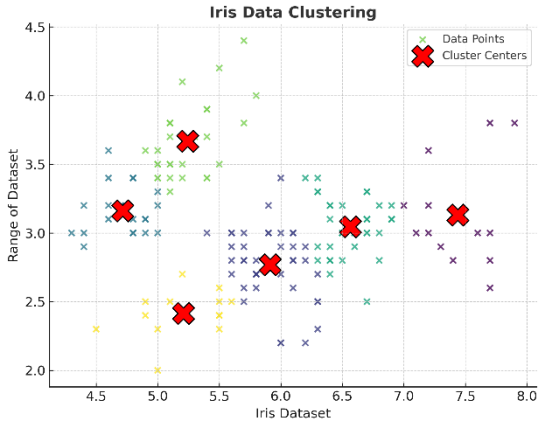
4.3 K-mean clustering

K-means clustering is a commonly adopted unsupervised machine learning algorithm utilized to partition a dataset into 'K' separate, non-overlapping clusters. Its goal is to group similar data points together while ensuring dissimilar points remain separate. Here's a breakdown of its operation:

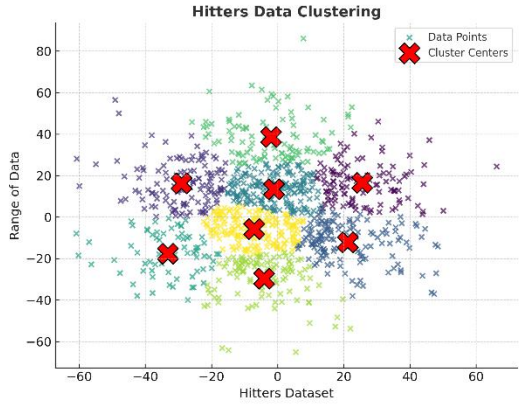


(a)

(b)



(c)



(d)

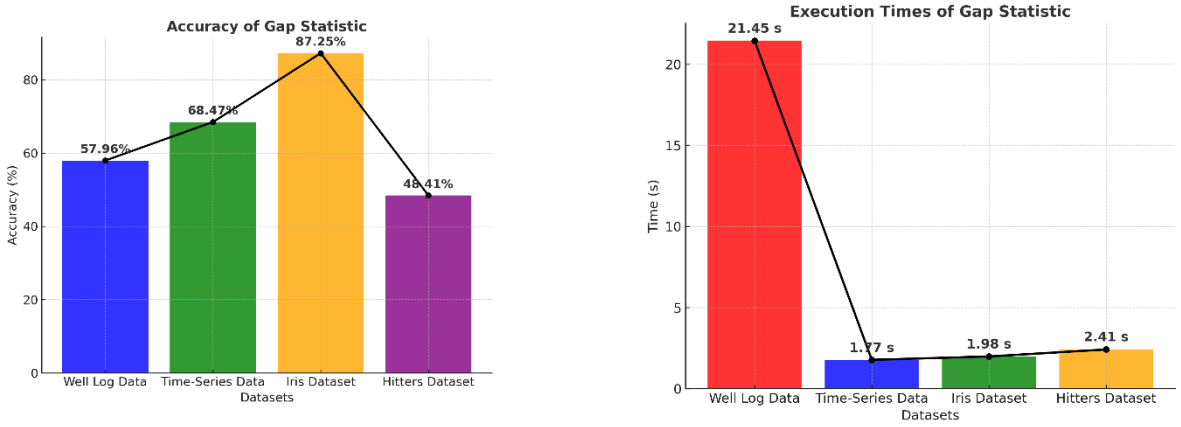
Fig. 3 k means clustering for different datasets

After employing the gap statistics to select the optimal number of clusters (ONC), k-means clustering was conducted as depicted in Fig 2. Figure 3 illustrates the results of k-means clustering across various datasets: (a) well log data, (b) time series data, (c) iris data, and (d) Hitter's dataset. Each graph provides a clear representation of the k-means clustering outcomes for the respective datasets.

4.4 Precision and computational efficacy

The accuracy and precision refer to the ability of the Gap Statistic method to reliably identify the optimal number of clusters in various datasets. This involves assessing how well the identified number of clusters aligns with the true underlying structure of the data.

Computational efficiency pertains to the speed and resource requirements of the Gap Statistic algorithm in determining the ONC. By applying the Gap Statistic method to different datasets, such as well log data, time series data, iris data, and the Hitters dataset, we can evaluate its effectiveness across a range of data types and structures. The analysis involves comparing the resulting clustering solutions to established benchmarks or ground truth labels, where available, to assess the accuracy of the identified ONC. Additionally, precision refers to the consistency and reproducibility of the ONC selection across multiple runs or iterations of the clustering algorithm. Overall, evaluating the accuracy and precision of the Gap Statistic method across diverse datasets provides insights into its robustness and reliability in practical applications of clustering analysis.



(a) (b)

Fig 3. Accuracy and execution time for selection the ONC

Fig 3(a) illustrates the process of selecting the optimal number of clusters (ONC) using the Gap Statistic method across various datasets including well log, time series, iris, and Hitter's data. In Fig 3(b), the corresponding execution times for applying the Gap Statistic in k-means clustering on the mentioned datasets are depicted.

4.5 Coefficient of determination (R-Square)

The coefficient of determination (R2) is used in k-means clustering to assess the quality of clustering solutions and select the optimal number of clusters (ONC). It measures the proportion of variance in the data explained by the clustering. A higher R2 value indicates better separation between clusters, helping to identify the point at which additional clusters do not significantly improve the explanation of variance, thus suggesting the ONC

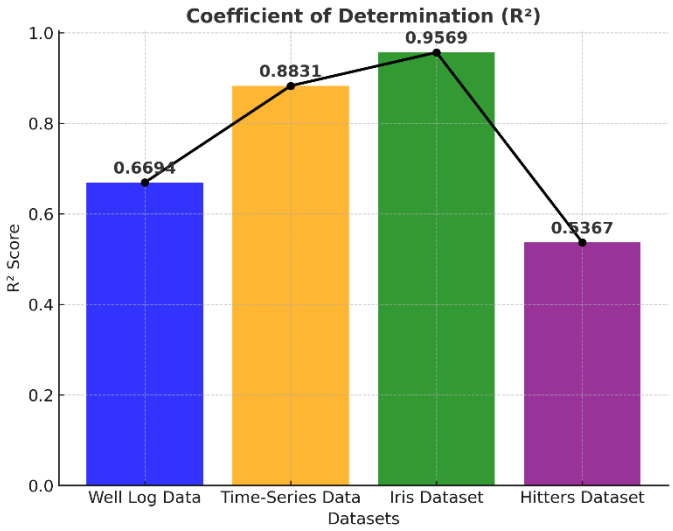


Fig 4. Coefficient of determination

In Fig 4 above, the coefficient of determination is depicted across various datasets, including well log, time series, iris, and Hitters. This measure assesses the strength of relationships between variables. A higher R-squared value indicates better performance, suggesting a stronger association between variables.

4.6 Summary of results

The given table 2 presents the results of clustering different datasets using the Gap Statistic to determine the Optimal Number of Clusters (ONC). The key performance metrics include Accuracy (%), Execution Time (sec), and the Coefficient of Determination (R^2). Let's interpret the results:

- Well Log Dataset: ONC = 8, meaning the Gap Statistic identified 8 as the optimal number of clusters. Accuracy = 57.98%, which indicates moderate clustering performance. Execution Time = 2.45 sec, suggesting a relatively higher computational cost. R^2 = 0.66, implying a moderate level of explained variance in the data.
- Time Series Dataset: ONC = 5, meaning the optimal number of clusters determined is 5. Accuracy = 68.47%, showing a relatively better clustering performance compared to the Well Log dataset.Execution Time = 1.77 sec, making it computationally efficient. R^2 = 0.88, which is high and indicates a strong fit of the clustering model to the data.
- Iris Dataset: ONC = 6, meaning the Gap Statistic found 6 clusters optimal instead of the standard 3 clusters. Accuracy = 87.25%, which is the highest accuracy among

all datasets, suggesting effective clustering. Execution Time = 1.98 sec, indicating reasonable computational efficiency. $R^2 = 0.95$, which is the highest, showing a strong relationship between the clustering structure and the dataset characteristics.

- Hitters Dataset: ONC = 8, meaning 8 clusters were found optimal. Accuracy = 48.41%, which is the lowest among all datasets, indicating clustering challenges. Execution Time = 2.41 sec, suggesting a higher computational cost. $R^2 = 0.53$, which is relatively low, implying weaker clustering performance

Dataset	ONC	Accuracy (%)	Execution Times/sec	Coefficient of determination
Well log	8	57.98	2.45	0.66
Time series	5	68.47	1.77	0.88
Iris	6	87.25	1.98	0.95
Hitters	8	48.41	2.41	0.53

Table 2. Summary of all the results for selection the ONC

Iris dataset performed the best in terms of accuracy (87.25%) and model fit ($R^2 = 0.95$), suggesting that the Gap Statistic effectively identified meaningful clusters. Time Series dataset also showed strong clustering performance with relatively high accuracy (68.47%) and an R^2 of 0.88, making it an effective dataset for clustering. Well Log and Hitters datasets performed the worst, with lower accuracy (57.98% and 48.41%) and lower R^2 values (0.66 and 0.53), indicating that clustering might not be as effective for these datasets.

5. Conclusion

This study evaluates the performance of the Gap Statistic in determining the optimal number of clusters across different datasets. The results indicate that the method performs well on smaller and structured datasets but struggles with larger or diverse datasets. For instance, the Iris dataset showed the best clustering performance with 87.25% accuracy and an R^2 of 0.95, confirming the suitability of the Gap Statistic for well-structured datasets. In contrast, the Hitters dataset performed the worst, with 48.41% accuracy and $R^2 = 0.53$, suggesting poor clustering effectiveness. Additionally, execution time varied, with the Well Log dataset taking 2.45 sec, reflecting increased computational costs for larger datasets. The findings suggest that the Gap Statistic is more effective for smaller datasets with clear structure but may be less reliable for complex datasets, where alternative clustering

validation methods should be considered.

Competing interests: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] A. Gondeau, Z. Aouabed, M. Hijri, P. Peres-Neto, V. Makarenkov, Object Weighting: A New Clustering Approach to Deal with Outliers and Cluster Overlap in Computational Biology, IEEE/ACM Trans. Comput. Biol. Bioinform. 18 (2021), 633–643. https://doi.org/10.1109/TCBB.2019.2921577.

[2] J.S. Ahlquist, C. Breunig, Model-Based Clustering and Typologies in the Social Sciences, Polit. Anal. 20 (2012), 92–112. https://doi.org/10.1093/pan/mpr039.

[3] A. Ghosal, A. Nandy, A.K. Das, S. Goswami, M. Panday, A Short Review on Different Clustering Techniques and Their Applications, in: J.K. Mandal, D. Bhattacharya (Eds.), Emerging Technology in Modelling and Graphics, Springer Singapore, Singapore, 2020: pp. 69–83. https://doi.org/10.1007/978-981-13-7403-6_9.

[4] H. Güçdemir, H. Selim, Integrating Multi-Criteria Decision Making and Clustering for Business Customer Segmentation, Ind. Manag. Data Syst. 115 (2015), 1022–1040. https://doi.org/10.1108/IMDS-01-2015-0027.

[5] Y. Fan, S. Lehmann, A. Blok, Extracting the Interdisciplinary Specialty Structures in Social Media Data-Based Research: A Clustering-Based Network Approach, J. Informetrics 16 (2022), 101310. https://doi.org/10.1016/j.joi.2022.101310.

[6] M. Van De Velden, A. Iodice D'Enza, A. Markos, Distance-based Clustering of Mixed Data, WIREs Comput. Stat. 11 (2019), e1456. https://doi.org/10.1002/wics.1456.

[7] P. Bhattacharjee, P. Mitra, A Survey of Density Based Clustering Algorithms, Front. Comput. Sci. 15 (2021), 151308. https://doi.org/10.1007/s11704-019-9059-3.

[8] F. Murtagh, P. Contreras, Algorithms for Hierarchical Clustering: An Overview, II, WIREs Data Min. Knowl. Discov. 7 (2017), e1219. https://doi.org/10.1002/widm.1219.

[9] C. Tan, H. Zhao, H. Ding, Statistical Initialization of Intrinsic K-Means Clustering on Homogeneous Manifolds, Appl. Intell. 53 (2023), 4959–4978. https://doi.org/10.1007/s10489-022-03698-8.

[10] P. Vora, B. Oza, A Survey on K-Mean Clustering and Particle Swarm Optimization, Int. J. Sci. Mod. Eng. 1 (2013), 24–26.

[11] A. Singh, A. Yadav, A. Rana, K-Means with Three Different Distance Metrics, Int. J. Comput. Appl. 67 (2013), 13–17. https://doi.org/10.5120/11430-6785.

[12] J. Yadav, M. Sharma, A Review of K-Mean Algorithm, Int. J. Eng. Trends Technol. 4 (2013), 2972–2976.

[13] I.K. Khan, H.B. Daud, N.B. Zainuddin, et al. Determining the Optimal Number of Clusters by Enhanced Gap Statistic in K-Mean Algorithm, Egypt. Inform. J. 27 (2024), 100504. https://doi.org/10.1016/j.eij.2024.100504.

[14] I.K. Khan, H.B. Daud, N.B. Zainuddin, et al. Addressing Limitations of the K-Means Clustering Algorithm: Outliers, Non-Spherical Data, and Optimal Cluster Selection, AIMS Math. 9 (2024), 25070–25097. https://doi.org/10.3934/math.20241222.

[15] I.K. Khan, H.B. Daud, R. Sokkalingam, et al. Numerical Solution by Kernelized Rank Order Distance (KROD) for Non-Spherical Data Conversion to Spherical Data, AIP Conf. Proc. 3123 (2024), 020011. https://doi.org/10.1063/5.0223847.

[16] C. Shi, B. Wei, S. Wei, et al. A Quantitative Discriminant Method of Elbow Point for the Optimal Number of Clusters in Clustering Algorithm, EURASIP J. Wirel. Commun. Netw. 2021 (2021), 31. https://doi.org/10.1186/s13638-021-01910-w.

[17] Y. Januzaj, E. Beqiri, A. Luma, Determining the Optimal Number of Clusters Using Silhouette Score as a Data Mining Technique, Int. J. Online Biomed. Eng. 19 (2023), 174–182. https://doi.org/10.3991/ijoe.v19i04.37059.

[18] X. Lu, Y. Lin, X. Li, et al. Gene Cluster Algorithm Based on Most Similarity Tree, in: Eighth International Conference on High-Performance Computing in Asia-Pacific Region, IEEE, Beijing, China, 2005: p. 5 pp. – 656. https://doi.org/10.1109/HPCASIA.2005.41.

[19] K. Cao, I. Musa, J. Liu, Y. Zhang, An Adaptive Density Clustering Algorithm for Massive Data, in: 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, IEEE, Guilin, 2017: pp. 1700–1707. https://doi.org/10.1109/FSKD.2017.8393022.

[20] A.M. El-Mandouh, L. A., H. A., M. H., Optimized K-Means Clustering Model Based on Gap Statistic, Int. J. Adv. Comput. Sci. Appl. 10 (2019), 183–188. https://doi.org/10.14569/IJACSA.2019.0100124.

[21] N. Arora, I. Budhiraja, D. Garg, S. Garg, B.J. Choi, M.S. Hossain, Revolutionizing Facial Image Retrieval: Multi-Block and Mean Based Local Binary Patterns with Sign and Magnitude Analysis, Alexandria Eng. J. 116 (2025), 601–608. https://doi.org/10.1016/j.aej.2024.12.003.

[22] R.G. Ribeiro, R. Rios, Temporal Gap Statistic: A New Internal Index to Validate Time Series Clustering, Chaos Solitons Fractals 142 (2021), 110326. https://doi.org/10.1016/j.chaos.2020.110326.

[23] M. Milosavljević, C.J. Miller, S.R. Furlanetto, A. Cooray, Cluster Merger Variance and the Luminosity Gap Statistic, Astrophys. J. 637 (2006), L9–L12. https://doi.org/10.1086/500547.

[24] K. Singh, D. Malik, N. Sharma, Evolving Limitations in K-Means Algorithm in Data Mining and Their Removal, Int. J. Comput. Eng. Manag. 12 (2011), 105–109.

[25] B. Zerhari, A.A. Lahcen, S. Mouline, Big data clustering: Algorithms and challenges, in: International Conference on Big Data, Cloud and Applications (BDCA'15), Morocco, 2015.

[26] J. Yang, J.Y. Lee, M. Choi, Y. Joo, A New Approach to Determine the Optimal Number of Clusters Based on the Gap Statistic, in: S. Boumerdassi, É. Renault, P. Mühlethaler (Eds.), Machine Learning for Networking, Springer International Publishing, Cham, 2020: pp. 227–239. https://doi.org/10.1007/978-3-030-45778-5_15.

[27] H. Cui, Y. Chang, H. Zhang, X. Mi, B. Kang, Determine the Number of Unknown Targets in the Open World from the Perspective of Bidirectional Analysis Using Gap Statistic and Isolation Forest, Inf. Sci. 623 (2023), 832–856. https://doi.org/10.1016/j.ins.2022.12.034.

[28] M. Iqbal, N. Zainuddin, H. Daud, R. Kanan, H. Soomro, R. Jusoh, A. Ullah, I. Karim Khan, A Modified Basis of Cubic B-Spline with Free Parameter for Linear Second Order Boundary Value Problems: Application to Engineering Problems, J. King Saud Univ. – Sci. 36 (2024), 103397. https://doi.org/10.1016/j.jksus.2024.103397.

[29] Z.F. Hassan, F. Al-Shareefi, H.Q. Gheni, A Coloured Image Watermarking Based on Genetic K-Means Clustering Methodology, J. Adv. Inf. Technol. 14 (2023), 242–249. https://doi.org/10.12720/jait.14.2.242-249.

[30] A. Qtaish, M. Braik, D. Albashish, M.T. Alshammari, A. Alreshidi, E.J. Alreshidi, Optimization of K-Means Clustering Method Using Hybrid Capuchin Search Algorithm, J. Supercomput. 80 (2024), 1728–1787. https://doi.org/10.1007/s11227-023-05540-5.

[31] H. Xin, Y. Lu, H. Tang, R. Wang, F. Nie, Self-Weighted Euler k-Means Clustering, IEEE Signal Process. Lett. 30 (2023), 1127–1131. https://doi.org/10.1109/LSP.2023.3305909.

[32] I.K. Khan, H. Daud, N. Zainuddin, et al. Exploring K-Means Clustering Efficiency: Accuracy and Computational Time across Multiple Datasets, J. Adv. Res. Appl. Sci. Eng. Technol. 65 (2026), 1-13

[33] D. Cheng, J. Huang, S. Zhang, S. Xia, G. Wang, J. Xie, K-Means Clustering With Natural Density Peaks for Discovering Arbitrary-Shaped Clusters, IEEE Trans. Neural Netw. Learn. Syst. 35 (2024), 11077–11090. https://doi.org/10.1109/TNNLS.2023.3248064.

[34] B.J.J. Kremers, J. Citrin, A. Ho, K.L. Van De Plassche, Two-step Clustering for Data Reduction Combining DBSCAN and k -means Clustering, Contrib. Plasma Phys. 63 (2023), e202200177. https://doi.org/10.1002/ctpp.202200177.

[35] E. Schubert, Stop Using the Elbow Criterion for K-Means and How to Choose the Number of Clusters Instead, ACM SIGKDD Explor. Newsl. 25 (2023), 36–42. https://doi.org/10.1145/3606274.3606278.

[36] D.T. Pham, S.S. Dimov, C.D. Nguyen, Selection of K in K-Means Clustering, Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci. 219 (2005), 103–119. https://doi.org/10.1243/095440605X8298.

[37] A. Bansal, M. Sharma, S. Goel, Improved K-Mean Clustering Algorithm for Prediction Analysis Using Classification Technique in Data Mining, International Journal of Computer Applications 157 (2017), 35–40. https://doi.org/10.5120/ijca2017912719.

[38] T. Kanungo, D.M. Mount, N.S. Netanyahu, C. Piatko, R. Silverman, A.Y. Wu, The Analysis of a Simple k -Means Clustering Algorithm, in: Proceedings of the Sixteenth Annual Symposium on Computational Geometry, ACM, Clear Water Bay Kowloon Hong Kong, 2000: pp. 100–109. https://doi.org/10.1145/336154.336189.

[39] J. Wang, R. Zuo, An Extended Local Gap Statistic for Identifying Geochemical Anomalies, J. Geochem. Explor. 164 (2016), 86–93. https://doi.org/10.1016/j.gexplo.2016.01.002.