K-Cube Consensus Clustering with Centroid Improvement and Variance-Based Metrics on High-Dimensional Data

Efori Bu'ulolo[1], Poltak Sihombing[2,*], Sutarman[3], Mohammad Andri Budiman[4]

[1,2,4]*Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Dr. T. Mansur str., 9, Medan, Indonesia*

[3]*Faculty Of Mathematics and Natural Sciences, Universitas Sumatera Utara, Dr. T. Mansur str., 9, Medan, Indonesia*

(Received: December 10, 2025; Revised: February 5, 2026; Accepted: March 28, 2026; Available online: April 26, 2026)

Abstract

High-dimensional and multidimensional cube data structures (K-Cube) are posing a significant challenge for conventional clustering algorithms due to the effect of dimensionality, uniform feature weight assumptions, and loss of hierarchical information. Therefore, this study aimed to propose K-Cube Consensus Clustering framework, which integrates Variance-Based Centroid Refinement, Weighted Distance Metrics, and consensus voting mechanism to overcome the challenges of high-dimensional cube data. The proposed method systematically clustered all dimensions and sub-dimensions of cube data, refined centroid by emphasizing more stable low-variance attributes, and applied adaptive distance weighting based on variance-derived feature weights integrated into the distance metric to improve cluster assignment. The final clusters were obtained through majority voting of the clustering results for each dimension. Unlike existing consensus clustering methods that operate on flat data representations or combine independent clustering results, the proposed framework explicitly exploits the hierarchical structure of multidimensional cube data by clustering dimensions and sub-dimensions prior to consensus integration. Moreover, variance-based centroid refinement and weighted distance metrics are jointly embedded within each cube dimension rather than applied as isolated enhancements. This hierarchy-aware design preserves cube semantics while simultaneously improving centroid stability and distance adaptivity, resulting in a distinct and scalable clustering framework for complex high-dimensional cube data. The framework processes cube dimensions independently with iterative convergence control, enabling scalable application to large-scale cube data. The results of synthetic and real-world high-dimensional datasets, including cube data with approximately 2.2 million instances, showed that the proposed method consistently outperformed K-Means, K-Medoids, and Hamiltonian formulations. The method produced lower SSE such as 3,179,328 on Arcene and 1,422.21 on Lung Cancer, higher Silhouette Score of approximately 0.5718 and 0.4905 for consensus results, better cluster stability of 0.9947, and faster convergence. These results confirmed the effectiveness of K-Cube Consensus Clustering in producing stable and meaningful clusters in large-scale high-dimensional data applications.

Keywords: High-Dimensional Clustering, K-Cube Data, Consensus Clustering, Variance-Based Centroid Refinement, Weighted Distance Metric

1. Introduction

The massive growth of data in the digital age is posing a significant challenge in information analysis, particularly with high-dimensional data [1], [2]. To address the challenge, clustering, an important tool in unsupervised learning, serves to group data based on similar characteristics without requiring initial labels [3]. Among various methods available, K-Means is the most popular due to simplicity and computational efficiency [4]. The effectiveness of K-Means reduces significantly when applied to high-dimensional data because of dimensionality, which causes the distances between points to become approximately uniform and difficult to distinguish meaningfully. Additionally, the assumption that all features have the same contribution is not in line with real conditions, where several attributes are irrelevant or redundant [5].

Several approaches have been proposed to address the equal feature contribution assumption of K-Means, including feature selection and dimensionality reduction methods such as principal component analysis. Although effective in reducing dimensionality and noise, these approaches often transform or discard original features, leading to reduced interpretability and loss of hierarchical relationships in multidimensional cube data. In contrast, the proposed K-Cube

***Corresponding author: Poltak Sihombing (poltak@usu.ac.id)**
DOI: https://doi.org/10.47738/jads.v7i2.1209
This is an open access article under the CC-BY license (https://creativecommons.org/licenses/by/4.0/).
© Authors retain all copyrights

Consensus Clustering framework preserves the cube structure and learns feature relevance through variance-based centroid refinement and adaptive distance weighting without collapsing dimensional semantics.

In various modern applications such as GIS, OLAP, recommendation systems, and digital health, data are not presented in flat tables, but rather in multidimensional cube structures containing hierarchical relationships between dimensions and sub-dimensions. This structure, known as cube data or K-Cube, allows information to be represented at several levels of aggregation, making the relationships between attributes essential [6]. However, conventional clustering methods such as K-Means are not designed to handle relational and hierarchical complexities, causing a significant loss of information during the clustering process [7], [8].

K-Cube Clustering method was developed through systematic exploration of all combinations of dimensions and sub-dimensions, alongside the results using ensemble clustering to overcome the limitations of K-Means. This method is capable of capturing various data perspectives, but generates new issues related to centroid formation and distance measurement [9], [10]. Since each sub-dimension has different variances and scales, the use of arithmetic mean-based centroid and Euclidean distances is no longer adequate [11].

Variance-Based Centroid Refinement (VBCR) offers a solution by giving greater weight to low-variance sub-dimensions that are more stable and informative. This method has been proven to improve centroid robustness against noise and outliers in high-dimensional data [12]. Additionally, Weighted Distance Metrics (WDM) allow distances between data to be calculated based on adaptive weights for more relevant features to have a greater influence in cluster formation. Although each method has been discussed in previous studies, there is no information on multidimensional cube data structures, creating a significant knowledge gap [13], [14].

The main gap is in the absence of an integrated method that simultaneously uses the hierarchical structure of K-Cube, VBCR, and WDM. This lack of integration causes conventional methods to fail in capturing hierarchical relationships, thereby leading to inaccurate and unstable clusters in high-dimensional data [15].

Based on the background above, this study proposes K-Cube Consensus Clustering method with the integration of VBCR and WDM to overcome the challenges of high-dimensional cube data. This method uses the entire cube structure, applies adaptive weights based on variance, and calculates distances more relevant to data characteristics. The results are expected to produce more stable, accurate, and semantically meaningful clusters, while providing theoretical and practical contributions to domains such as bioinformatics, e-commerce, recommendation systems, and spatial analysis.

2. Literature Review

High-dimensional data are becoming increasingly common in various fields such as bioinformatics, digital health, recommendation systems, and spatial exploration. However, the increase in the number of features causes the curse of dimensionality problem, where the distance between points becomes less informative and sensitive to noise, leading to the loss of accuracy by traditional clustering algorithms such as K-Means [16], [17]. Equal contributions from features are also assumed by K-Means, although several features are redundant or irrelevant, making centroid formation process prone to distortion [18].

In practice, modern data are not often organized in flat tables. Therefore, several analytical systems use multidimensional or K-Cube data structures containing dimensions and sub-dimensions such as time, location, and product. These structures store hierarchical relationships that are important for data interpretation [19]. However, conventional clustering methods do not consider the relationships, leading to the loss of important information when data is transformed into a flat format, making the clustering results less representative [20], [21].

Several studies have attempted to address the challenges through K-Cube Clustering method, which systematically explores combinations of dimensions and sub-dimensions, alongside the results using ensemble clustering [18], [22]. Although ensemble methods have been proven to improve cluster stability, there are still challenges in centroid formation and distance calculation caused by variance and scale differences between sub-dimensions. In these conditions, arithmetic mean-based centroid and Euclidean distances are no longer adequate [23].

To address the challenges, two methods have been developed separately, namely VBCR and WDM. VBCR provides greater weight to low-variance features to achieve more stable and noise-resistant centroid [24]. Meanwhile, WDM calculates distances based on adaptive weights for relevant features to have a greater contribution [25], [26]. Although effective, the literature shows that both methods have not been explicitly integrated in the context of multidimensional cube data structures.

The lack of information on VBCR and WDM integration into multidimensional cube data structure creates a significant knowledge gap in the literature. Specifically, there is no clustering method that comprehensively combines K-Cube exploration, ensemble clustering, VBCR, and WDM in a single comprehensive framework. To fill the gap, this study proposes K-Cube Consensus Clustering with VBCR and WDM integration to produce more stable, accurate, adaptive cluster to sub-dimensional heterogeneity, and semantically representative of complex multidimensional structures.

3. Methodology

As shown in figure 1, an experimental quantitative analysis was used by focusing on the development and evaluation of clustering algorithms based on high-dimensional cube data structures. The proposed algorithm integrated VBCR, WDM, and a Consensus mechanism. The method was divided into three main stages, namely data pre-processing, the clustering process (VBCR and WDM), and the consensus as well as evaluation stage.

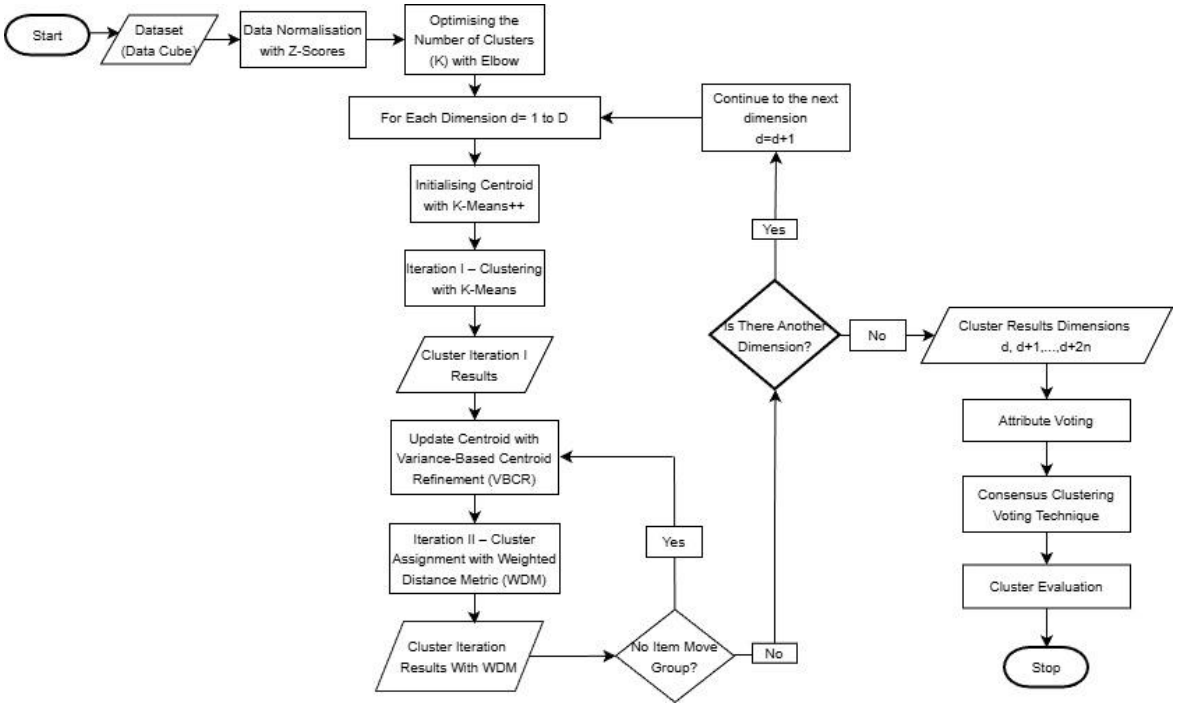

Figure 1. K-Cube Clustering Consensus Method Framework

3.1. Pre-processing

The input consists of cube data with high dimensions and sub-dimensions, as represented in the form:

$$X_i = x_{i1}, x_{i2}, xi_3, \dots, x_n), i = 1,2,3, \dots, m \quad (1)$$

All dimensions and sub-dimensions of the cube data are normalized using z-scores [27]:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \quad (2)$$

Z-score normalization was used because it preserves relative variance by standardizing features to zero mean and unit variance, which is essential for VBCR and WDM. Unlike min–max normalization, which compresses variance and is sensitive to outliers, and robust scaling, which may attenuate inter–sub-dimension variance, z-score better supports adaptive weighting in high-dimensional cube data.

Cluster number (K) optimization was performed using the Elbow method. The dimension used as the basis for determining the K value was randomly selected from all dimensions in the cube data, and K value obtained was applied as the number of clusters for all dimensions of the cube data.

$$WCSS(K) = \sum_{k=1}^{K} \sum_{x_i \epsilon C_k} ||x_i - \mu_k||^2 \quad (3)$$

3.2. Update Centroid and Assignment

Each dimension in the cube data was clustered based on sub-dimension values, starting with the centroid initialization process using K-Means++ method.

$$P(x) = \frac{D(x)^2}{\sum_{x' \epsilon X} D(x')^2} \quad (4)$$

Based on the initial centroid obtained, the first clustering process (Iteration I) was carried out. In this iteration, cluster member assignment was performed using K-Means algorithm with standard Euclidean distance.

$$d(x_i, \mu_k) = \sqrt{\sum_{j=1}^{p} (x_{ij} - \mu_{kj})^2} \quad (5)$$

After Iteration I was completed, the initial cluster results were obtained. Based on these results, the centroid was updated using the VBCR method to produce more representative and stable cluster centers.

$$C_j = \frac{\sum_{x_i \epsilon C_j} w_i . x_i}{\sum_{x_i \epsilon C_j} w_i} \quad (6)$$

with the variance weight defined as:

$$w_j = \frac{1}{\sigma_j^2 + \epsilon} \quad (7)$$

$\sigma^2(x_i)$ is the variance of attribute feature xi in cluster C_j, The constant ϵ ensures numerical stability by avoiding extreme weights for near-zero variances and is fixed at 10^{-6}, with negligible impact on centroid updates, and w_j is the weight for feature x_i based on the stability of its distribution.

The process continued to Iteration II for cluster assignment using WDM, based on centroid that had been updated through VBCR method.

$$d_w(x_i, y_i) = \sqrt{\sum_{k=1}^{n} w_k \left(x_{ik} - x_{jk}\right)^2} \quad (8)$$

W_k is the weight for the kth attribute dimension, and n is the total number of attribute features in the K-Cube feature vector. Subsequently, cluster results in Iteration II are compared with the results of Iteration I to determine whether there is any cluster member movement. When there is still movement, the process continues to the next iteration with centroid updates using VBCR and cluster assignment using WDM until convergence is achieved. However, in the absence of movement, the iteration process in that dimension is stopped and continued to the clustering process in the next (d+2n) until all clusters are produced in all dimensions. The iterative process is considered converged when no cluster membership changes occur between successive iterations.

Although the proposed method processes all dimensions and sub-dimensions of the cube structure, it does not exhaustively enumerate dimensional combinations. Each dimension is clustered independently through an iterative process that stops at convergence, thereby avoiding combinatorial explosion. As a result, the computational complexity scales linearly with the number of cube dimensions and clustering iterations, ensuring scalability for large-scale cube datasets.

3.3. K-Cube Consensus Clustering Voting Method

After all dimensions have been subjected to clustering process, each dimension would produce distinct clusters. To obtain the final clusters from the cube data, a simple voting method was used, where each clustering method or iteration

 ISSN 2723-6471

cast one vote on the cluster membership of a data point. The final decision was determined based on the majority vote, assuming there were M clustering methods or iterations that contributed to the voting process.

$$C_{final}(x_i) = mode\{C_1(x_i), C_2(x_2), \dots, C_m(x_i)\} \quad (9)$$

Consensus clustering is performed via majority voting across cube dimensions, with ties resolved using minimum average weighted distance to VBCR-refined centroids. Aggregation across dimensions limits the influence of noisy assignments, while feature relevance is handled within each dimension through variance-based centroid refinement and weighted distance metrics.

To clearly describe the computational procedure of the proposed K-Cube Consensus Clustering framework, the algorithmic steps are summarized in the following pseudocode.

Algorithm 1. K-Cube Consensus Clustering

Input: Cube dataset D, number of clusters K
Output: Final cluster labels
1: Normalize data using Z-score
2: For each dimension d in D do
3: Initialize centroids using K-Means++
4: repeat
5: Assign data to clusters using Euclidean distance
6: Update centroid using VBCR
7: Reassign clusters using WDM
8: until convergence
9: end for
10: Apply majority voting across all dimensions
11: Return final cluster labels

The pseudocode above summarizes the main computational steps of the proposed method, providing a clear and structured representation of the overall clustering process.

4. Results and Discussion

This section presents the results of experiments and comparative analyses of K-Cube Consensus Clustering method with established algorithms, namely K-Means, K-Medoids, and Hamiltonian Formulation. The evaluations were carried out to measure the effectiveness and reliability of the proposed method. The variables used to compare the performance of each algorithm included the number of iterations, Sum of Squared Error (SSE), Squared Error (SE), clustering result stability, and the ability to detect outliers. Subsequently, dataset consisted of synthetic and test data obtained from the main source, namely Kaggle.com. With these variations in data characteristics, the evaluation was expected to be more representative and comprehensive in measuring the capabilities of the proposed method.

4.1. Data pre-processing

This study used synthetic dataset obtained from a computational process, consisting of 247 variables and 240 records arranged in 19 dimensions with 13 sub-dimensions in each dimension. All variables were grouped into three variance categories High (65 variables), Medium (78 variables), and Low (104 variables), each with different dispersion characteristics. This variance-based structure was designed to support the evaluation of VBCR and WDM methods under synthetic data conditions with varying levels of dispersion, and the statistical summary is presented in table 1.

Table 1. Synthetic Dataset

Category of Variance	Number of Variables	Mean	Standard Deviation	Variance	Min	Max
High	65	4.166	1008.203	1,016,474.097	−3775.506	4678.949
Medium	78	−9.184	995.289	990,600.052	−3976.719	4611.257

Low	104	−2.909	997.938	995,879.596	−4121.696	3938.301

The dataset in table 1 was normalized using the Z-Score method to equalize the scale of all dimensions and sub-dimensions. This allows each feature to have an equal influence in the cluster calculation process. The variance categories in table 1 are used solely for descriptive analysis of feature dispersion. Features are grouped into high, medium, and low variance using quantile-based partitioning, and these categories are not used as clustering inputs, as all clustering is performed on normalized continuous features to avoid bias. A summary of the dataset statistics after the Z-Score normalization process is presented in table 2.

Table 2. Summary Statistics of the Dataset After Z-Score Normalization

Statistics	Value
Mean	-3.44×10^{-19} (≈ 0)
Standard Deviation	1.000
Minimum	−4.254
Maximum	4.681

As shown in table 2, the dataset exhibits substantial variance heterogeneity across attributes. This motivates the use of variance-aware clustering, while the optimal cluster number is determined using the Elbow Method shown in figure 2. Cluster number (K) optimization was performed using the Elbow method applied to one of the randomly selected dataset dimensions. In this study, dimension 1 was selected as the basis for determining the K value, with the results shown in figure 2.

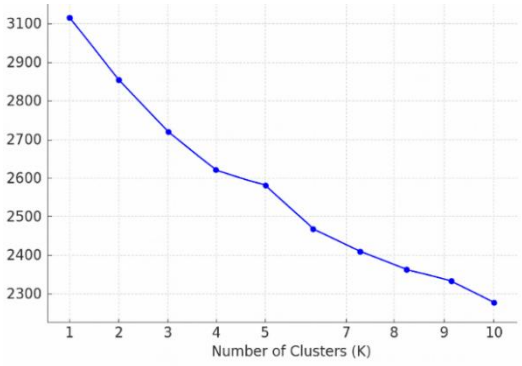

Figure 2. Optimal Cluster Number (K)

The 1-dimensional Elbow Method graph for the first cube data showed a sharp decline in WCSS up to K = 3, where the elbow point was clearly formed. After K = 3, the decline in WCSS became gradual, as adding clusters only provided minimal improvement. This result shows that K = 3 can be considered the optimal number of clusters.

4.2. Updating Centers with K-Means++ and Clustering with Standard K-Means

After the optimal number of clusters (K) was determined, clustering process in the first dimension started with centroid initialization using K-Means++. The first centroid (C1) was selected randomly and generated ID 103. The second centroid (C2) was selected based on the weighted probability of the squared distance to C1, and ID 193 was picked. Meanwhile, the selection of the third centroid (C3) was based on the minimum distance to the existing centroid. Table 3 shows centroid resulting from the K-Means++ initialization. The reported centroid IDs denote data instances selected by the standard K-Means++ procedure and are provided only for illustration and reproducibility, not as ID-based selection criteria.

Table 3. K-Means++ Initialization Centroid

ID / Centroid	D1.1	D1.2	D1.3	D1.4	D1.5	D1.6	D1.7	D1.8	D1.9	D1.10	D1.11	D1.12	D1.13
103/C1	-0.551	0.651	-0.836	-0.895	0.227	-0.202	-0.853	-1.045	0.699	0.358	0.359	-1.182	-0.512
193/C2	-0.296	-1.751	-0.818	1.615	-0.485	0.904	0.719	-1.015	-1.075	0.976	-0.804	0.412	-0.194
48/C3	-0.584	-0.034	-0.163	0.161	-1.425	0.466	0.287	1.156	0.292	0.280	2.728	0.727	1.939

Based on the obtained centroid, the process continued to Iteration I to cluster the data using K-Means algorithm with Euclidean distance calculations. This stage produced the initial cluster formation as shown in table 4.

Table 4. Cluster Results of the First Dimension (Iteration I)

Cluster	Number of Items	Member ID
1	117	['1', '2', '3', '8', '10', '11', '12', '13', '15', '16', '20', '23', '24', '26', '28', '29', '35', '36', '37', '38', '40', '41', '42', '44', '45', '46', '49', '51', '54', '56', '58', '60', '62', '66', '67', '69', '70', '72', '73', '76', '78', '82', '83', '84', '90', '91', '93', '96', '99', '103', '104', '105', '106', '107', '108', '109', '110', '112', '119', '120', '128', '130', '131', '132', '133', '134', '137', '138', '139', '140', '144', '145', '146', '147', '151', '153', '156', '162', '163', '165', '166', '167', '168', '170', '171', '173', '176', '177', '180', '181', '185', '186', '189', '192', '198', '199', '200', '203', '204', '205', '206', '210', '215', '217', '218', '219', '226', '228', '229', '230', '231', '232', '233', '235', '237', '239', '240']
2	68	['4', '5', '6', '7', '9', '14', '17', '21', '25', '27', '30', '32', '39', '59', '65', '74', '77', '81', '86', '92', '95', '98', '100', '101', '111', '113', '114', '115', '117', '121', '122', '126', '129', '136', '141', '142', '149', '152', '157', '160', '161', '164', '169', '174', '178', '182', '183', '184', '188', '190', '193', '195', '196', '197', '201', '202', '207', '208', '209', '211', '212', '214', '216', '221', '222', '223', '224', '238']
3	55	['18', '19', '22', '31', '33', '34', '43', '47', '48', '50', '52', '53', '55', '57', '61', '63', '64', '68', '71', '75', '79', '80', '85', '87', '88', '89', '94', '97', '102', '116', '118', '123', '124', '125', '127', '135', '143', '148', '150', '154', '155', '158', '159', '172', '175', '179', '187', '191', '194', '213', '220', '225', '227', '234', '236']

4.3. Update Centroid with VBCR Centroid and Assignment with WDM

Based on the cluster results in Iteration I, the process continued with centroid updating using VBCR method in each cluster. This stage produced a new variance-weighted centroid, as presented in table 5.

Table 5. VBCR Weighted Variance Centroid

Centroid	D1.1	D1.2	D1.3	D1.4	D1.5	D1.6	D1.7	D1.8	D1.9	D1.10	D1.11	D1.12	D1.13
C1	-0.032	0.216	-0.120	-0.430	0.233	-0.103	-0.273	-0.105	0.252	-0.071	-0.091	-0.373	-0.262
C2	0.043	-0.667	0.088	0.567	-0.061	-0.050	0.337	-0.435	-0.446	0.035	-0.357	0.517	0.044
C3	-0.124	0.426	0.103	0.328	-0.395	0.163	0.183	0.681	-0.121	0.038	0.781	0.033	0.549

According to centroid generated through VBCR, the clustering process continued to Iteration II. At this stage, cluster member assignment was performed using WDM, where each feature was weighted based on the level of relevance in calculating the distance between the data and the centroid. Subsequently, iterative process continued with centroid updating using VBCR and WDM-based reassignment until convergence was achieved. The series of iterations showed a pattern of increasingly stable cluster member shifts with each repetition. The clustering results in Iteration II are presented in table 6.

Table 6. Clustering results using WDM

Iteration	Cluster	Number of Items	Member ID	Update Centroid Values with VBCR
II	1	117	['1', '2', '3', '8', '10', '11', '12', '13', '15', '16', '20', '23', '24', '26', '28', '29', '35', '36', '37', '38', '40', '41', '42', '44', '45', '46', '49', '51', '54', '56', '58', '60', '62', '66', '67', '69', '70', '72', '73', '76', '78', '82', '83', '84', '90', '91', '93', '96', '99', '103', '104', '105', '106', '107', '108', '109', '110', '112', '119', '120', '128', '130', '131', '132', '133', '134', '137', '138', '139', '140', '144', '145', '146', '147', '151', '153', '156', '162', '163', '165', '166', '167', '168', '170', '171', '173', '176', '177', '180', '181', '185', '186', '189', '192', '198', '199', '200', '203', '204', '205', '206', '210', '215', '217', '218', '219', '226', '228', '229', '230', '231', '232', '233', '235', '237', '239', '240']	[-0.032, 0.216, -0.12, -0.43, 0.233, -0.103, -0.273, -0.105, 0.252, -0.071, -0.091, -0.373, -0.262]
II	2	68	['4', '5', '6', '7', '9', '14', '17', '21', '25', '27', '30', '32', '39', '59', '65', '74', '77', '81', '86', '92', '95', '98', '100', '101', '111', '113', '114', '115', '117', '121', '122', '126', '129', '136', '141', '142', '149', '152', '157', '160', '161', '164', '169', '174', '178', '182', '183', '184', '188', '190', '193', '195', '196', '197', '201', '202', '207', '208', '209', '211', '212', '214', '216', '221', '222', '223', '224', '238']	[0.043, -0.667, 0.088, 0.567, -0.061, -0.05, 0.337, -0.435, -0.446, 0.035, -0.357, 0.517, 0.044]
II	3	55	['18', '19', '22', '31', '33', '34', '43', '47', '48', '50', '52', '53', '55', '57', '61', '63', '64', '68', '71', '75', '79', '80', '85', '87', '88', '89', '94', '97', '102', '116', '118', '123', '124', '125', '127', '135', '143', '148', '150', '154', '155', '158', '159', '172', '175', '179', '187', '191', '194', '213', '220', '225', '227', '234', '236']	[-0.124, 0.426, 0.103, 0.328, -0.395, 0.163, 0.183, 0.681, -0.121, 0.038, 0.781, 0.033, 0.549]

ISSN 2723-6471

...	...	...	...	...
VI	1	109	['3', '8', '10', '11', '12', '13', '15', '16', '20', '21', '23', '24', '26', '28', '29', '33', '35', '36', '37', '38', '40', '41', '42', '44', '45', '49', '51', '52', '56', '58', '60', '62', '66', '67', '69', '70', '72', '76', '78', '84', '90', '91', '93', '96', '99', '103', '104', '105', '106', '107', '109', '110', '112', '119', '120', '128', '130', '131', '132', '133', '134', '137', '138', '139', '140', '144', '145', '146', '147', '151', '153', '156', '162', '163', '165', '166', '167', '170', '171', '173', '177', '180', '181', '185', '186', '189', '192', '198', '199', '200', '203', '204', '205', '206', '210', '215', '217', '218', '219', '226', '228', '229', '230', '231', '232', '235', '237', '239', '240']	[-0.01, 0.219, -0.11, -0.5, 0.33, -0.074, -0.305, -0.159, 0.31, -0.072, -0.109, -0.3, -0.25]
VI	2	69	['4', '5', '6', '7', '9', '14', '17', '18', '25', '27', '30', '31', '32', '39', '50', '59', '64', '74', '77', '86', '92', '95', '98', '100', '101', '111', '113', '114', '115', '117', '121', '122', '123', '126', '129', '136', '141', '142', '149', '152', '157', '160', '161', '164', '169', '174', '178', '182', '183', '188', '190', '193', '195', '196', '197', '201', '202', '207', '208', '209', '211', '212', '213', '214', '216', '221', '222', '224', '238']	[0.1, -0.606, 0.142, 0.506, -0.129, 0.002, 0.227, -0.473, -0.383, 0.04, -0.33, 0.648, 0.096]
VI	3	62	['1', '2', '19', '22', '34', '43', '46', '47', '48', '53', '54', '55', '57', '61', '63', '65', '68', '71', '73', '75', '79', '80', '81', '82', '83', '85', '87', '88', '89', '94', '97', '102', '108', '116', '118', '124', '125', '127', '135', '143', '148', '150', '154', '155', '158', '159', '168', '172', '175', '176', '179', '184', '187', '191', '194', '220', '223', '225', '227', '233', '234', '236']	[-0.209, 0.335, -0.019, 0.383, -0.413, 0.02, 0.28, 0.761, -0.234, 0.02, 0.695, -0.327, 0.382]

Convergence was consistently reached within a small number of iterations across dimensions, indicating stable and efficient refinement. The visualization of clustering results in the final iteration (Iteration IV) is shown in figure 3.

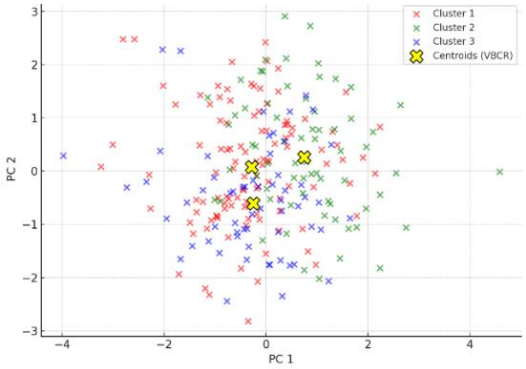

Figure 3. First Dimension Cluster Visualization Based on PCA.

Each point in figure 3 represents a single data instance, with different colors for each cluster (red for Cluster 1, green for Cluster 2, and blue for Cluster 3) as the final result of VBCR and WDM processes until convergence. The yellow X points show the final centroid obtained through VBCR, which are located at mass center of each cluster in the PCA projection space. The distribution of points shows that cluster members tend to cluster around the centroid, although there is still a small area of overlap showing pattern proximity in high-dimensional space. This visualization shows that the combination of VBCR and WDM can produce clearer and more representative cluster separation.

After the first dimension of the cube data converged on the 6th iteration and final cluster formation, the same process was applied to the second and third until all dimensions of the cube data obtained the respective cluster results. Subsequently, the results obtained per dimension are combined to form a consensus label, in the form of a table containing attribute IDs and cluster labels. Summary of Cluster Frequency per Attribute (C1, C2, C3) in table 7.

Table 7. Summary of Cluster Frequency per Attribute (C1, C2, C3)

Atribut	C1	C2	C3	Atribut	C1	C2	C3
D1	88	67	85	D11	94	70	76
D2	68	55	117	D12	61	90	89
D3	74	85	81	D13	176	24	40
D4	71	86	83	D14	70	99	71
D5	78	78	84	D15	62	12	166
D6	100	46	94	D16	77	86	77
D7	102	67	71	D17	43	19	178

D8	70	97	73	D18	161	57	22
D9	70	97	73	D19	138	83	19
D10	75	26	139				

4.4. K-Cube Consensus Clustering

Table 7 shows frequency distribution of cluster labels (C1, C2, and C3) for each attribute from the consensus process. The distribution pattern obtained shows variations in cluster dominance between dimensions. Some attributes such as D13, D17, D18, and D19, have a strong tendency towards certain clusters, while others tend to be more balanced. This information forms the basis in consensus clustering stage to determine the final clusters through a voting mechanism, with the results shown in table 8.

Table 8. Final consensus voting results

ID	Consensus
1	C2
2	C1
3	C3
4	C3
5	C2
…	…
236	C1
237	C1
238	C3
239	C1
240	C3

Based on the majority vote results for 240 instances across all data dimensions, the final cluster distribution is obtained as shown in table 9.

Table 9. Cluster Distribution

Cluster	Number of Members	Label Clustering
C1	37	Medium range and homogeneous
C2	101	Relatively small average and low variance
C3	102	Dominant and high variance

Figure 4 shows a visualization of K-Cube consensus clustering results in the form of a two-dimensional PCA plot. The distribution of points appears to be concentrated around the projection centre, with a number of points scattered to more extreme areas. The colors or cluster labels show overlapping patterns in various regions, without clear boundaries between clusters. This shows that the separation of clusters in the PCA projection space is weak, and cluster structure in high-dimensional space is not fully projected separately in two-dimensional space.

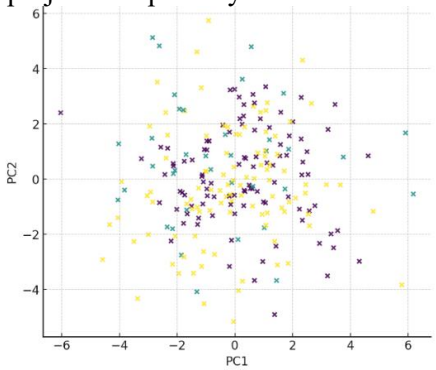

Figure 4. Distribution of Consensus Voting Results Clusters

ISSN 2723-6471

An evaluation was conducted using silhouette scores (SS) to assess the suitability and consistency of cluster members. The PCA-based visualizations in figure 3 and 4 are used to provide an intuitive two-dimensional representation of the clustering results. The first two principal components explain approximately 73% of the total variance (PC1: 76%, PC2: 71%), indicating that the projection captures a substantial portion of the data variability while not fully representing the high-dimensional structure. In the synthetic dataset used, the scores from the Final Consensus Cluster were compared with those on dimensions D1, D10, and D18, with results shown in figure 5.

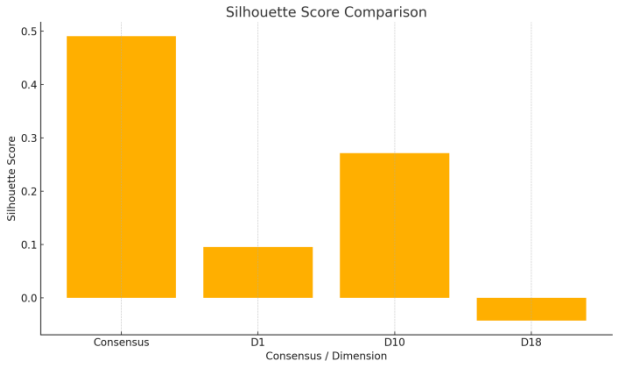

Figure 5. SS Comparison

Based on the comparison results, Consensus clusters had the highest SS (0.4905), showing better separation quality and a clearer structure compared to other dimensions. D10 dimension showed moderate clustering quality (0.2714), while D1 had a relatively low score (0.0954). D18 dimension produced a negative score (-0.0425), showing significant cluster overlap. The negative Silhouette Score for D18 reflects strong cluster overlap caused by high variance heterogeneity and noise in this dimension. This observation underscores the limitation of single-dimension clustering and supports the role of the consensus mechanism in stabilizing the final clustering result. This graph shows that K-Cube clustering consensus mechanism can produce more stable and separate clusters compared to each dimension individually.

4.5. Evaluation of WBCR and WDM

To evaluate the performance of K-Cube Consensus Clustering, a series of tests was conducted using several datasets and compared with the K-Means, K-Medoids, and Hamiltonian Formulation. The evaluation was based on numerous metrics, namely the number of iterations, SSE, SS, stability, and outlier detection. The performance testing was divided into two parts, namely testing of VBCR and WDM methods using high-dimensional but non-cubic datasets to assess the effectiveness of the two components independently. K-Cube Consensus Clustering was also tested using high-dimensional cubic datasets to evaluate the overall performance of the mechanism. Independent testing of VBCR and WDM methods was conducted using the dataset described in table 10.

Table 10. Dataset for Evaluation of VBCR and WDM

No	Dataset	Row	Column	Description
1	Arcene	700	10.000	Very high-dimensional data
2	Colon Labeled	62	2.000	Small dataset with big features
3	Lung Cancer	32	57	Very small medical data
4	Ionosphere	350	35	Balanced dataset between the number of rows and attributes
5	Breast Cancer	569	33	Medium-sized and relatively balanced dataset

The results of the dataset evaluation in table 10 are presented in figure 6.

ISSN 2723-6471

Figure 6. Performance Comparison Across Clustering Algorithms on Multiple Metrics

Based on the test results in table 10 dataset, VBCR and WDM methods showed the most consistent and efficient performance. These methods produced lower SSE in high-dimensional datasets and higher SS, showing better cluster separation quality. The stability was also high in most datasets, showing resilience to initialization variations. Furthermore, VBCR and WDM required fewer iterations, leading to faster convergence and limited outliers, which produced cleaner and more representative clusters. The evaluation in figure 6 focuses on centroid-based clustering methods (K-Means, K-Medoids, and Hamiltonian), as VBCR and WDM are designed to enhance centroid representation and distance computation. Density-based and spectral methods were excluded due to parameter sensitivity in high-dimensional spaces and scalability limitations for large datasets.

4.6. Evaluation Of K-Cube Consensus Clustering

Performance evaluation was conducted by comparing the results of K-Cube Consensus Clustering with several widely used algorithms, namely K-Means, K-Medoids, and Hamiltonian Formulation. Testing was carried out on various cube-shaped datasets, where each dimension had a number of subdimensions, as shown in table 11.

Table 11. Dataset for Evaluating K-Cube Consensus Clustering

No.	Dataset	Instance	Dimension	Sub-Dimension	Description
1	Synthetic	240	19	13	Multidimensional
2	SeasFire Datacube	100.000	21	59	Multidimensional
3	Mave	2.200.000	2.535	1.000	Multidimensional

 ISSN 2723-6471

The results of the dataset evaluation in table 11 are presented in figure 7.

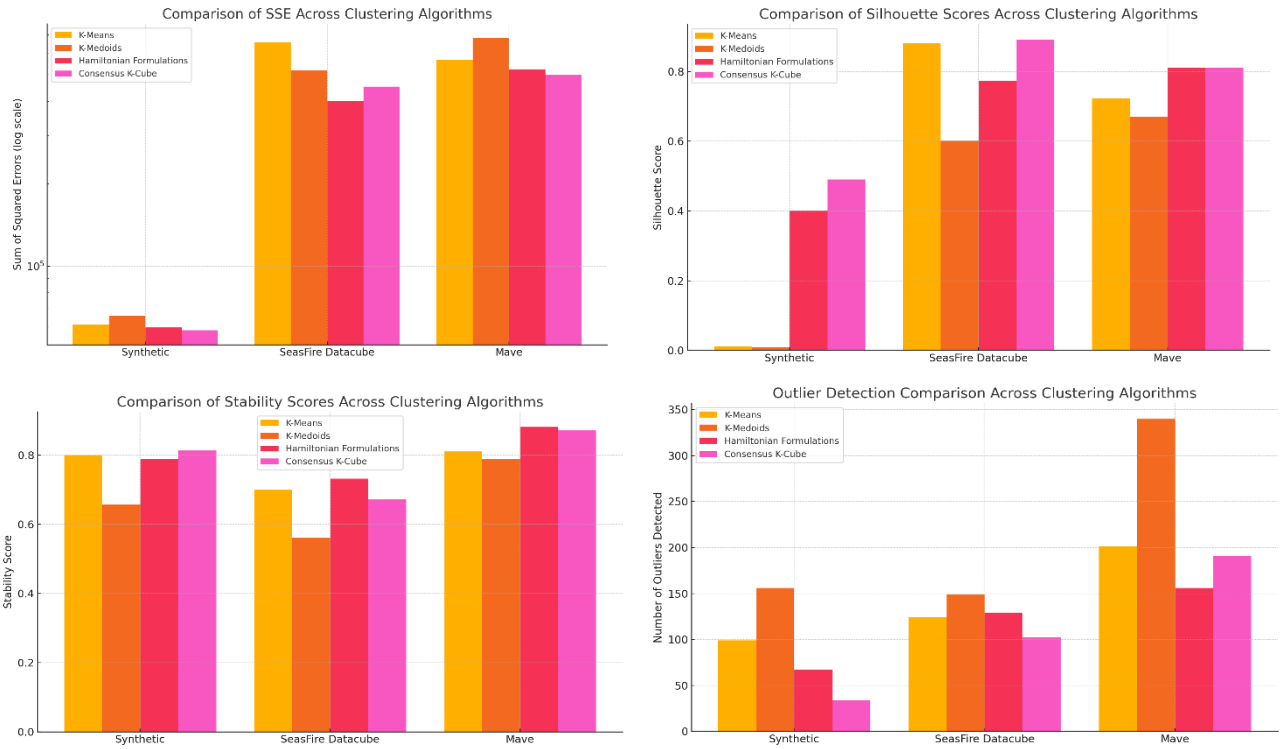

Figure 7. Performance Comparison Between Clustering Algorithms on Various Metrics On cube-shaped datasets

K-Cube Consensus Clustering shows superior performance compared to K-Means, K-Medoids, and Hamiltonian formulations on all four main evaluation metrics. On the SSE graph, this method consistently produces lower error values, showing the ability to form more compact clusters across the entire dataset. Regarding the SS metric, K-Cube Consensus Clustering achieves the highest or near-highest score on every dataset, signifying better cluster separation and cohesive cluster structure. Stability evaluation also shows that the method provides more consistent results across iterations and data samples, suggesting robustness to data variation. In outlier analysis, a lower number of outliers is detected in most datasets, showing effectiveness in forming stable clusters without separating many points as anomalies. These combined results show that K-Cube Consensus Clustering offers better, more stable, and reliable clustering quality in high-dimensional cube datasets. This study focuses on empirical performance trends across multiple datasets rather than formal statistical significance testing; therefore, the results should be interpreted as indicative rather than definitive superiority.

5. Conclusion

In conclusion, K-Cube Consensus Clustering successfully presents a clustering framework capable of using the multidimensional cube structure (K-Cube) optimally to maintain the relationships between attributes and sub-attributes previously lost in the flat data method. The application of VBCR has been proven to improve centroid representation by considering variance between dimensions, thereby producing a more accurate distribution of data within each cluster. The integration of WDM enables adaptive dimension weighting to precisely distinguish the relevance of each attribute. The combination of K-Cube structure, VBCR, and WDM produces more stable, accurate, and relevant clusters, overcoming the limitations of conventional K-Means algorithms and previous separate methods.

VBCR and WDM show the most competitive performance on high-dimensional cube data, as indicated by the lowest SSE on Arcene (3,179,328), Lung Cancer (1,422.21), and Ionosphere (5,703.06), alongside the highest SS on Arcene (0.5718) and Ionosphere (0.3582). The stability of this method is also high, reaching 0.9914 on Ionosphere and 0.9947 on Breast Cancer, with a relatively low number of outliers on datasets such as Arcene (20) and Colon Labeled (0). VBCR and WDM show better iteration efficiency, achieving only 5 iterations on Arcene and 3 on Lung Cancer.

However, the performance declines slightly on certain datasets, such as lower stability on Lung Cancer (0.6875) and a higher number of outliers on Ionosphere (16). The reduced performance on the Lung Cancer dataset is mainly due to its extremely small sample size relative to dimensionality, which results in unstable variance estimation and sensitivity to noise and attribute redundancy. These results show that VBCR and WDM are generally superior in producing compact, stable, and efficient clusters, but remain sensitive to variations in certain data structures.

Compared to cube-shaped datasets, K-Cube Consensus consistently outperforms K-Means, K-Medoids, and Hamiltonian Formulations on high-dimensional cube data. Quantitatively, this method produces the lowest SSE across all datasets (58,447 on Synthetic and 498,277 on Mave), the highest or near-highest SS (0.8912 on SeasFire and 0.8112 on Mave), and a better stability score (up to 0.8722). For datasets such as Arcene and Ionosphere, baseline methods (K-Means and K-Medoids) generally produce low Silhouette Scores and higher SSE, reflecting the difficulty of clustering high-dimensional data. The higher Silhouette Scores and reduced SSE achieved by the proposed methods therefore represent meaningful relative improvements rather than absolute performance claims. The number of detected outliers is also lower in several datasets, such as 34 in Synthetic, showing the ability to form more compact, clearly separated, stable clusters with minimal anomalies. These results confirm that K-Cube Consensus is a more effective and reliable method for clustering high-dimensional cube data.

5.1. Author Contributions

Conceptualization: E.B., P.S., S; Methodology: E.B., M.A.B.; Software: E.B.; Validation: P.S., S.; Formal Analysis: E.B.; Investigation: E.B., M.A.B; Resources: S., M.A.B.; Data Curation: E.B.; Writing Original Draft Preparation: E.B.; Writing Review and Editing: P.S., S., M.A.B; Visualization: E.B. This article is an official output of the doctoral dissertation of E.B., under the supervision of P.S., S., and M.A.B. All authors have read and agreed to the published version of the manuscript.

5.2. Data Availability Statement

The original dataset used in this study was obtained from Kaggle (https://www.kaggle.com/datasets, while the synthetic dataset was sourced from a publicly accessible spreadsheet available at https://docs.google.com/spreadsheets/d/1oIvq2piW67ETJh_xHNABmvvcGvj-n8mC/edit?usp=sharing&ouid=112124156289122863688&rtpof=true&sd=true

5.3. Funding

This study was conducted as part of the doctoral dissertation of E.B. at Universitas Sumatera Utara and received no external funding.

5.4. Institutional Review Board Statement

Not applicable. This study used publicly available secondary data and did not include human participants directly.

5.5. Informed Consent Statement

Not applicable.

5.6. Declaration of Competing Interest

The authors declare that there are no competing financial interests or personal relationships regarding this study.

References

[1] X. Chen, Y. Ye, X. Xu, and J. Z. Huang, "A Feature Group Weighting Method for Subspace Clustering of High-Dimensional Data," *Pattern Recognit*., vol. 45, no. 1, pp. 434–446, 2012, doi: 10.1016/j.patcog.2011.06.004.

[2] M. M. Mariani, R. Baggio, M. Fuchs, and W. Höepken, "Big data and analytics in tourism and hospitality: A perspective article," *Tour. Manag. Perspect*., vol. 25, no. 1, pp. 63–68, 2018, doi: 10.1016/j.tmp.2017.12.002.

[3] W. Qu, X. Xiu, H. Chen, and L. Kong, "A Survey on High-Dimensional Subspace Clustering A Survey on High-Dimensional Subspace Clustering," *Mathematics*, vol. 11, no. 2, pp. 1–39, 2023, doi: 10.3390/math11020436.

[4] S. Chakraborty, P. Dey, D. Das, N. Shah, H. Wang, and S. Sra, "Clustering High-dimensional Data with Ordered Weighted Regularization," in *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics (AISTATS), PMLR*, vol. 2023, no. 1, pp. 10483–10496, 2023.

[5] A. G. Oskouei, "Feature-weighted fuzzy clustering methods: An experimental review," *Neurocomputing,* vol. 619, no. 1, pp. 1-16, 2024, doi: 10.1016/j.neucom.2024.129176.

[6] J. Leprince, C. Miller, and W. Zeiler, "Data mining cubes for buildings: a generic framework for multidimensional analytics of building performance data," *Energy Build.,* vol. 248, no. 1, pp. 1-15, 2021, doi: 10.1016/j.enbuild.2021.111195.

[7] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," *Electronics,* vol. 9, no. 8, pp. 1-15, 2020, doi: 10.3390/electronics9081295.

[8] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: : A comprehensive review, variants analysis, and advances in the era of big data," *Inf. Sci*., vol. 622, no. 1, pp. 178–210, 2023, doi: 10.1016/j.ins.2022.11.139.

[9] J. Zamora and J. Sublime, "An Ensemble and Multi-View Clustering Method Based on Kolmogorov Complexity," *Entropy*, vol. 25, no. 2, pp. 371-389, 2023, doi: 10.3390/e25020371.

[10] X. Zhao, X. Niu, Y. Ma, and J. Zhang, "A Multi-View Ensemble Clustering Approach Using Joint Entropy," *Expert Syst. Appl*., vol. 255, no. 1, pp. 1-23, 2024, doi: 10.1016/j.eswa.2024.124683.

[11] Z. Liu, H. Qiu, T. Senapati, M. Lin, L. Abualigah, and M. Deveci, "Enhancements of Evidential C-Means Algorithms: A Clustering Framework via Feature-Weight Learning," *Expert Syst. Appl.*, vol. 259, no. 1, pp. 1-16, 2025, doi: 10.1016/j.eswa.2024.125246.

[12] F. Galis and Galis, F., Onchis, D. M., and Istin, C., "Refining Filter Global Feature Weighting for Fully Unsupervised Clustering," *Applied Sciences*, vol. 15, no. 16, pp. 1-14, 2025, doi: 10.3390/app15169072.

[13] Z. Ning, Z. Dai, H. Zhang, Y. Chen, and Z. Yuan, "A Clustering Method for Small scRNA-seq Data Based on Subspace and Weighted Distance (SSWD)," *PeerJ*, vol. 2023, no. 1, pp. 1-16, 2023, doi: 10.7717/peerj.14706.

[14] Z. Liu, H. Qiu, T. Senapati, M. Lin, L. Abualigah, and M. Deveci, "A Clustering Framework via Feature-Weight Learning," *Appl. Soft Comput.,* vol. 150, no. 1, pp. 1-15, 2025, doi: 10.1016/j.asoc.2023.111015.

[15] T. ALASALI and Y. ORTAKCI, "Clustering Techniques in Data Mining: A Survey of Methods, Challenges, and Applications," *Comput. Sci.*, vol. 2024, no. 1, pp. 32–50, 2024, doi: 10.53070/bbd.1421527.

[16] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *International Conference on Database Theory (ICDT), Springer,* vol. 2001, no. 1, pp. 420–434. doi: 10.1007/3-540-44794-6_27.

[17] I. Hussain, "Weighted Multiview K-Means Clustering with L2 Regularization," *Symmetry (Basel)*., vol. 16, no. 12, pp. 1-16, 2024, doi: 10.3390/sym16121646.

[18] H.-P. Kriegel, P. Z. Kröger, and A. Zimek, "High-dimensional data clustering: A survey on subspace clustering, pattern-based clustering, and correlation clustering," *ACM Trans. Knowl. Discov. from Data*, vol. 3, no. 1, pp. 1–34, 2009, doi: 10.1145/1497577.1497578.

[19] A. Cuzzocrea, I.-Y. Song, and K. C. Davis, "Analytics over big multidimensional data: The big cube analytics," *Inf. Syst.*, vol. 103, no. 1, pp. 1-16, 2021, doi: 10.1016/j.is.2021.101706.

[20] M. Sabri et al., "A Novel Classification Algorithm Based on the Synergy Between Dynamic Clustering with Adaptive Distances and K-Nearest Neighbors," *J. Classif.*, vol. 41, no. 1, pp. 264–288, 2024, doi: 10.1007/s00357-024-09471-5.

[21] X. Du, Y. Zhang, X. Chen, and Y. Peng, "A robust and high-dimensional clustering algorithm based on non-Euclidean distance combining feature weights and entropy weights," *Entropy*, vol. 25, no. 3, pp. 510-524, 2023, doi: 10.3390/e25030510.

[22] W. Li, P. Chen, Y. Zhang, X. Chen, and Y. Peng, "An Ensemble Clustering Framework Based on Hierarchical Clustering Ensemble Selection," *Int. J. Gen. Syst*., vol. 54, no. 5, pp. 741–766, 2023, doi: 10.1080/01969722.2022.2073704.

[23] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta, "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *ACM Comput. Surv*., vol. 54, no. 7, pp. 1–36, 2021, doi: doi: 10.1145/3471310.

ISSN 2723-6471

[24] W. M. El-Kenawy, A. Ibrahim, A. E. Mohamed, and S. Mirjalili, "WeDIV – An improved k-means clustering algorithm with a weighted distance and a novel internal validation index," *Egypt. Informatics J.*, vol. 23, no. 4, pp. 133–144, 2022, doi: 10.1016/j.eij.2022.09.002.

[25] J. Sun, Y. Zhang, and B. Liu, "Adaptive distance metrics for improved clustering in high-dimensional spaces," *Knowledge-Based Syst.*, vol. 264, no. 1, pp. 1-19, 2023, doi: 10.1016/j.knosys.2023.110229.