Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 7s (2025)

Mathematical Insights and Applications of K-Means Clustering in Diabetes Data Analysis

Sofi. A. Francis[1], M. Sangeetha[1], P.Pooranadevi[2], A.Sathyakala[3]

[1]Department of Mathematics, Dr.N.G.P Arts and Science College, Coimbatore, India-641048

[2]Department of Science and Humanities, Info Institute of Engineering, Coimbatore, India-641107

[3]Department of Mathematics, Dr.N.G.P. Institute of Technology, Coimbatore, India-641048 Corresponding author E-mail: m.sangeethaphd@gmail.com

Article History:

Received: *26-10-2024*

Revised:*10-11-2024*

Accepted:*18-12-2024*

Abstract

K-Means clustering is a powerful and adaptable technology that has transformed industries like operations research and healthcare due to the quick development of machine learning and data analytics. Using K-Means, this study examines a real- world dataset of 1050 diabetic patients that has been divided into groups based on important clinical criteria such as HbA1c levels, random blood sugar (RBS), post- prandial blood sugar (PPBS), and fasting blood sugar (FBS). Utilizing operations research methods, our goals are to pinpoint unique patient groups, improve individualized care plans, and maximize the use of available resources. The effectiveness of K-Means clustering in diabetes management is demonstrated in this study, which also provides important insights for individualized healthcare and shows the algorithm’s superiority over other machine learning techniques.

Keywords: FBS, PPBS, RBS, HbA1c, K-Means

1 Introduction

In many different industries, operations research is essential to enhancing efficiency, solving complicated challenges, and streamlining decision-making processes. Patient management, resource allocation, and treatment procedures have significantly improved in the healthcare industry as a result of the use of operations research approaches. One such method, K-Means clustering, has become a potent instrument for dividing data into distinct groups and making it easier to find trends and insights that can improve judgment.

For diabetes management to improve patient outcomes and lessen the burden of this chronic condition, individualized treatment plans are crucial. In order to effectively man- age diabetes, important clinical markers such HbA1c levels, random blood sugar (RBS), postprandial blood sugar (PPBS), and fasting blood sugar (FBS) must be continuously monitored and controlled. Because diabetic patients are diverse, clustering algorithms such as K-Means can be useful in identifying discrete patient groups, each with its own set of traits and requirements for therapy.

In this study, 1050 diabetic patients from a real-world dataset are analyzed using K - Means clustering. Finding hidden patterns and groups based on FBS, PPBS, RBS, and HbA1c levels is the goal. These can help healthcare organizations allocate resources more efficiently and provide individualized treatment strategies. By combining machine learning and operations research approaches, we hope to provide a strong foundation for improving the care of diabetes patients.

This study is innovative in that it tackles the challenges of managing diabetes by fusing rigorous operations research methods with K-Means clustering. This integration helps medical professionals make data-driven decisions for focused interventions in addition to classifying patients according to their glycemic control. The potential of K-Means clustering to better patient care outcomes, encourage effective use of medical resources, and improve individualized diabetes control is demonstrated by our findings.

With this work, we hope to make a significant contribution to the domains of healthcare analytics and operations research by showcasing the effectiveness of sophisticated clustering approaches in practical clinical settings. Our study's findings highlight how crucial data-driven strategies are to treating chronic illnesses and improving healthcare delivery.

In operations research, K-Means clustering is a popular unsupervised learning approach for dividing data into discrete groups according to similarity. The work of [1] Amor et al. offers a thorough bibliometric examination of K-Means clustering and other multi- criteria decision analysis (MCDA) techniques. It highlights prospective study directions, maps research patterns, and finds important articles.

The paper of [2] Ahmed et al. discusses the K-Means method's limits and current advancements while providing an organized summary of the research on the algorithm. The efficiency of different K-Means variants is assessed by experimental study. The measurement errors in K-Means clustering are addressed in the study of [3] Burgard et al. through the use of robust optimization approaches. Specifically designed decomposition techniques are developed to protect clustering outcomes from unstructured mistakes.

The scalable and economical R-k-means algorithm, which offers an ideal solution for improved clustering of large-scale high-dimensional datasets, is proposed in the study of Gul et al. [4]. In order to test for a difference in means across clusters found using K- Means clustering, the study of Yiqun et al. [5] suggests a p-value that depends on all intermediate clustering assignments.

The study of Zubair et al. [6] introduces an enhanced K-Means clustering technique that effectively determines ideal initial centroids while cutting down on execution time and iterations. The work of [7] Ikotun et al. offers a thorough description and categorization of the K-means clustering algorithm and its variants. In addition to discussing suggested future research topics, the study reviews current trends, identifies unresolved concerns and challenges, and digs into the history of K-means.

In the paper, [8] Sinaga et al. identify the ideal number of clusters and create an unsupervised learning framework for the k-means algorithm that does not require parameter selection or initializations. The new unsupervised k-means (U-k-means) clustering method finds the ideal number of groups on its own without the need for initialization or parameter adjustment. The work also examines how computationally hard the suggested U-k-means clustering algorithm is.

For K-means clustering, two new initialization techniques are presented in the study of Joonas et al. [9]. The divide-and-conquer tactic used in both approaches is specific to the K-means initialization procedure. The second approach additionally makes use of several lower-dimensional subspaces created for initialization using the random projection technique.

Amorim et al.'s [10] fundamental claim is that K-means converges with a highly in- formative average number of iterations, represented by τ. They illustrate this finding's importance with two real-world examples. First, they describe the use of τ to improve on current feature selection techniques or to find characteristics in a dataset that are not relevant. Furthermore, they find a substantial correlation between τ and the number of clusters in a dataset, which can be used to estimate the actual number of clusters.

In the work of [11] Fahim , a technique for determining near-optimal initial centers and k values is presented. The preprocessing stage in the suggested method establishes the initial centers and the number of clusters before the k-means algorithm is applied. It computes the mean values for objects inside each cluster to guide the k-means process after using a density-based technique to create initial clusters without knowing the number of clusters beforehand.

Muhima et al. [12] used a hybrid genetic algorithm polygamy with K-means (GAP K-means) for grouping hotspot data in the investigation. The initial centroid of K-means was found using the GA polygamy, which addressed its sensitivity and sped up the process of finding the best answer.

In this paper, it stems from several advanced mathematical elements.

- It integrates a comprehensive mathematical analysis, including the derivation of the within-cluster sum of squares (WCSS) and its implications for clustering quality. The incorporation of rigorous evaluation metrics, such as the Silhouette Score, provides a quantitative measure of cluster cohesion and separation. This metric not only validates the clustering performance but also offers a deeper understanding of the clustering process. The use of both 3D and 2D scatter plots, along with the inclusion of centroids, demonstrates innovative geometric representations that enhance the interpretability of the clustering results.

- The paper discusses the computational complexity of the k-means algorithm, ex- pressed as O(n, k, t, d), highlighting its efficiency and scalability. By addressing practical applications and limitations, such as the algorithm's sensitivity to initial centroid placement and assumptions of spherical clusters, our research bridges mathematical theory and practical utility.

- The suggestions for future research directions, including exploring alternative distance metrics and incorporating additional features, further showcase our forward- thinking approach, paving the way for advancements in clustering methodologies. This comprehensive and balanced view adds significant value to our research, making it a noteworthy contribution to the field.

2 Mathematical Preliminaries

2.1 In terms of Operations Research

In order to guarantee the usefulness of K-Means clustering in operations research, especially in the healthcare industry for managing diabetic patients, we incorporate further optimization methods:

Objective: Determine the optimal number of clusters K.

1. **Elbow Method:** Determine the ”elbow point” by plotting the within-cluster sum of squares (WCSS) against the number of clusters (K).

$$\mathbf{WCSS} = \sum_{k=1}^{K} \sum_{x_i \in C_k} ||x_i - c_k||^2$$

2. **Silhoutte Analysis:** Measure the quality of clustering by calculating the silhoutte score.

$$\textbf{Silhouette Score} = \frac{b(i) - a(i)}{max(a(i), b(i))}$$

where:

- $a(i)$ is the average distance from the i - th point to the other points in the same cluster.
- $b(i)$ is the average distance from the i - th point to the points in the nearest different cluster.

3 Algorithm

The algorithm of K-means is discussed in the works of [13], [14], [15], [16]. We introduce a new mathematical model that combines operations research methods with K-Means clustering to improve patient outcome as efficiently as possible.

Model Formulation

The objective of K-Means clustering is to partition a set of n diabetic patients into K clusters, minimizing the within-cluster sum of squares (WCSS). Each patient i is characterized by a feature vector x_i that includes fasting blood sugar (FBS), post- prandial blood sugar (PPBS), random blood sugar (RBS), and HbA1c levels.

Objective Function:

Minimize the total within-cluster variance: Minimize

$$\sum_{k=1}^{K} \sum_{x_i \in C_k} ||x_i - c_k||^2$$

Where:

c_k is the centroid of cluster k, C_k is the set of patients assigned to cluster k, $\| x_i - c_k \|^2$ represents the squared Euclidean distance between patient i and the centroid of cluster k.

Constraints:

Each patient must be assigned to exactly one cluster:

$$\sum_{k=1}^{K} y_{ik} = 1 \quad \forall\, i \in 1, 2, \ldots\ldots n.$$

Ensure that each cluster contains atleast one patient:

$$\sum_{i=1}^{n} y_{ik} \geq 1 \forall\, k \in 1, 2, \ldots\ldots K.$$

Where y_{ik} is a binary variable indicating whether patient i is assigned to cluster k:

$$y_{ik} = \begin{cases} 1, \text{if patient } i \text{ is assigned to cluster } k \\ 0, \quad \text{Otherwise} \end{cases}$$

Centroid Update

The centroid of each cluster is recalculated as the mean of all patients assigned to that cluster:

$$c_k = \frac{1}{|C_k|} \sum_{x_i \in c_k} x_i$$

Where:

c_k is the new centroid for cluster k

C_k is the set of data points assigned to cluster k

|.| denotes the cardinality of the set, i.e., the number of data points in cluster k.

3.1 K-Means Clustering Algorithm from Scratch

This is a step-by-step algorithm for K-Means clustering from scratch.

Step 1: Random Initialization

Objective: Randomly select K data points from the dataset as initial centroids to start the clustering process.

$c_k^0 \in X, k = 1, 2, \qquad K.$

K-means++ Initialization:(for better convergence):

- Choose at random the first centroid.
- Calculate the distance $D(x_i)$ to the closest centroid for each data point x_i.
- ^{i}Choose the subsequent centroid whose probability is proportional to $D(x^2)$.
- Continue until centroids are chosen.

Mathematical Concept:

To enhance the quality of the initial centroids, apply the k-means++ initialization technique. In order to improve clustering and accelerate convergence, this approach selects initial centroids in a way that disperses them over the feature space.

$$P(x_i) = \frac{D(xi)^2}{\sum_j D(xj)^2}$$

where:

$D(x_i)$ is the distance from data point x_i to the nearest already chosen centroid.

$P(x_i)$ is the probability of selecting x_i as the next centroid.

Step 2: Assignment Step

Objective: Assign each data point to the nearest centroid.

Mathematical Concept:

Calculate the Euclidean distance between each data point x_i and each centroid c_k:

$$d(x_i, c_k) = \sqrt{\sum_{j=1}^{n} (x_{ij} - c_{ij})^2}$$

Assign each data point x_i to the cluster with the nearest centroid:

$$\boldsymbol{y_{ik}} = \begin{cases} 1, \text{if } k = argmin_k \, d(x_i, c_k) \\ 0, \qquad\qquad \text{Otherwise} \end{cases}$$

where y_{ik} is a binary variable indicating cluster assignment.

Step 3: Upadate Step

Objective: Update the centroids with respect to the current cluster assignments.

Mathematical Concept:

Recalculate the centroid of each cluster to be the mean of all points assigned to that cluster:

$$c_k = \frac{1}{|C_k|} \sum_{x_i \in c_k} x_i$$

where: C_k is the set of points assigned to cluster k.

$|C_k|$ is the number of points in cluster k.

Step 4: Iteration

Objective: Repeat the assignment and update steps till convergence.

Mathematical Concept:

Repeat Steps 2 and 3 until the centroids no longer change significantly or a predefined number of iterations is reached.

Convergence Criterion:

$$||\boldsymbol{c_k^{t+1}} - \boldsymbol{c_K^{t}}|| < \boldsymbol{\mathcal{E}}$$

$\boldsymbol{c_k^{t+1}}$ is the updated centroid at iteration t+1.

$\boldsymbol{c_K^{t}}$ is the centroid at iteration t.

$\boldsymbol{\mathcal{E}}$ is a small positive threshold.

3.2 Working Model Algorithm

First Step: Initialization and Importing Libraries

Import Libraries: Import the necessary libraries for data manipulation, preprocessing and clustering. The code in Python sets the stage for data processing and model implementation by importing all required libraries. This foundational step is crucial for any subsequent data analysis, clustering, or regression tasks.

Second Step: Loading and Displaying the Dataset

-**Load the Dataset**: Load the dataset from a specified file path and display the first few rows to verify the data. This step involves reading the CSV file containing our dataset and printing the first few rows to ensure that the data has been loaded correctly. Adjust the file path variable to match the location of our CSV file on your computer.

Moreover, the paper discusses the computational complexity of the k-means algorithm, expressed as O(n, k, t, d), highlighting its efficiency and scalability. By addressing practical applications and limitations, such as the algorithm's sensitivity to initial centroid placement and assumptions of spherical clusters, our research bridges mathematical theory and practical utility. The suggestions for future research directions, including exploring alternative distance metrics and incorporating additional features, further showcase our forward-thinking approach, paving the way for advancements in clustering methodologies. This comprehensive and balanced view adds significant value to our research, making it a noteworthy contribution to the field.

0: Hyper (greater than Normal) 1: Hypo (less than Normal)

2: Normal

Third Step: Extracting Features and Target, Standardizing Data

Extract Input Features and Target, Standardize the Data: Extract the input features (first four columns) and the target column. Standardize the features for k- means clustering.

Extract Input Features: Select the first four columns of the dataset as input features (FBS, PPBS, RBS, Gender).

Extract Target Column: Select the 'Outcome' column as the target variable for classification tasks.

Standardize the Data: Use Standard Scaler to standardize the input features. This ensures that each feature contributes equally to the distance calculations in k-means clustering, as standardized features have a mean of 0 and a standard deviation of 1.

This step is crucial for preparing the data for k-means clustering and subsequent classification tasks.

Fourth Step: K-Means Clustering

K-Means Clustering: Initialize the k-means algorithm with the desired number of clusters, fit it to the scaled data, assign cluster labels, and display the clustering results.

Initialize K-Means: The K-Means object is initialized with the specified number of clusters (e.g., 3) and a random state for reproducibility.

Fit the Model: The k-means algorithm is applied to the standardized data (X_{scaled}), which clusters the data into the specified number of clusters.

Assign Cluster Labels: The resulting cluster labels are assigned to the original dataset in a new column named 'Cluster'.

Display Clustering Results: The distribution of data points across the clusters is displayed using the value counts () method.

This step completes the k-means clustering process, allowing us to analyze the clustering results within our dataset.

Fifth Step: Visualizing Clusters with a 3D Scatter Plot

3D Scatter Plot for K-Means Clustering: Create a 3D scatter plot to visualize the clusters formed by the k-means algorithm, including the cluster centers.

Import Libraries: Import the Axes3D module from mpl toolkits.mplot3d and matplotlib.pyplot for plotting.

Create 3D Scatter Plot:

Initialize a figure and a 3D subplot.

Plot the clusters using a scatter plot where each data point is color-coded based on its cluster label.

Plot the cluster centroids with distinct markers.

Add Titles and Labels: Set the title and labels for each axis to make the plot understandable.

Display the Legend and Grid: Add a legend to distinguish the centroids from the data points and enable the grid for better visualization.

Show the Plot: Render the plot on the screen.

This step will help us to visualize the clustering results in a 3D space, making it easier to analyze the clustering performance and understand the data distribution.

Sixth Step: 2D Visualization of K-Means Clustering

(FBS vs PPBS) Plotting the Clusters Using FBS and PPBS as the Axes: Create a 2D scatter plot to visualize the clusters formed by the k-means algorithm using FBS and PPBS as the axes. Create 2D Scatter Plot: Initialize a 2D scatter plot with a specified figure size.

Plot Clusters:

The first scatter plot displays the data points, colored by their cluster labels.

The second scatter plot shows the cluster centroids with distinct markers and colors. Add Titles and Labels: Titles and axis labels are added to clearly describe the plot.

Display Legend and Grid: A legend is included to differentiate between data points and centroids. The grid is enabled for better visualization.

Show the Plot: Render the plot on the screen.

This visualization helps in understanding the distribution of clusters with respect to FBS and PPBS, providing clear insights into the clustering results.

Seventh Step: 2D Visualization of K-Means Clustering (PPBS vs RBS)

This visualization focuses on the relationship between PPBS and RBS, providing another perspective on the clustering results.

Eighth Step: 2D Visualization of K-Means Clustering(FBS vs RBS)

This visualization exhibits the relation between FBS and RBS.

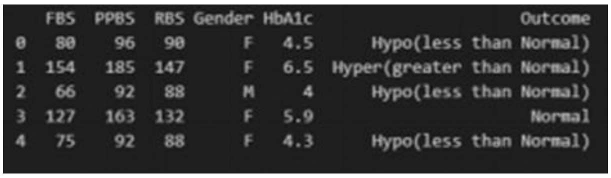
3.3 Algorithm in Table Format

Table 1: Algorithm Steps

Step No.	Algorithm Idea	Illustration
Step 1	Initialization and Importing Libraries	In Coding
Step 2	Loading and Displaying the Dataset	csv file of our dataset(Figure 2 shows 5 patient’s report)
Step 3	Extracting Features and Target, Standardizing Data	FBS, PPBS, RBS, Gender
Step 4	K-Means Clustering	Figure 3
Step 5	Visualizing CLusters with a 3D Scatter Plot	Figure 4
Step 6	2D Visualization of K- Means Clustering (FBS vs PPBS)	Figure 6
Step 7	2D Visualization of K- Means Clustering (PPBS vs RBS)	Figure 7
Step 8	2D Visualization of K- Means CLustering (FBS vs RBS)	Figure 8

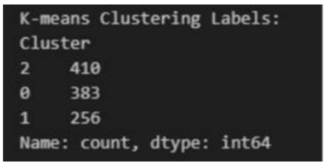
4 Observations

In the observation section we are displaying the first 5 rows of the datasets in figure 1 as the complete exhibition (1050 patients reports) is a little bit difficult task.



```
   FBS  PPBS  RBS Gender HbA1c                      Outcome
0   80    96   90      F   4.5       Hypo(less than Normal)
1  154   185  147      F   6.5  Hyper(greater than Normal)
2   66    92   88      M     4       Hypo(less than Normal)
3  127   163  132      F   5.9                       Normal
4   75    92   88      F   4.3       Hypo(less than Normal)
```

Figure 1: Datasets of the first 5 patients



```
K-means Clustering Labels:
Cluster
2    410
0    383
1    256
Name: count, dtype: int64
```

Figure 2: Display of Clustering Value Count

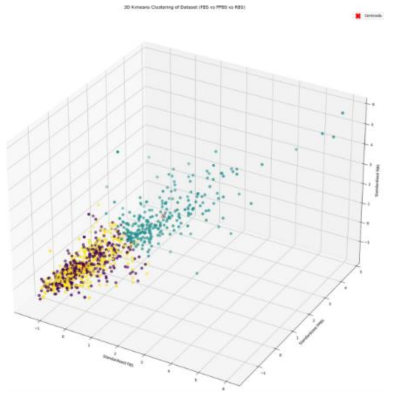
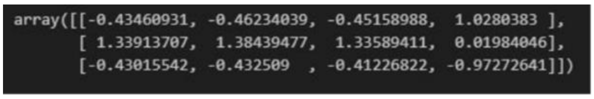


Figure 3: FBS vs PPBS vs RBS



```
array([[-0.43460931, -0.46234039, -0.45158988,  1.0280383 ],
       [ 1.33913707,  1.38439477,  1.33589411,  0.01984046],
       [-0.43015542, -0.432509  , -0.41226822, -0.97272641]])
```

Figure 4: K Means Cluster Centres

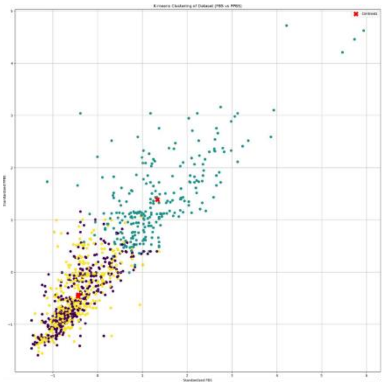


Figure 5: FBS vs PPBS

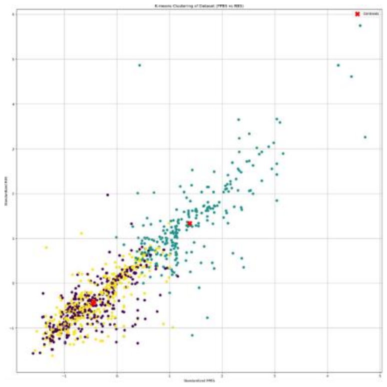


Figure 6: PPBS vs RBS

Results and Discussion

The application of k-means clustering to the diabetes dataset has revealed distinct groupings of patients based on their FBS, PPBS, RBS, and Gender values. By minimizing the within-cluster sum of squares (WCSS), the algorithm effectively identified three clusters, each represented by a centroid. The centroids, as calculated, provide a clear mathematical representation of the central tendencies of each cluster, offering insights into the typical values and variation within these patient groups.

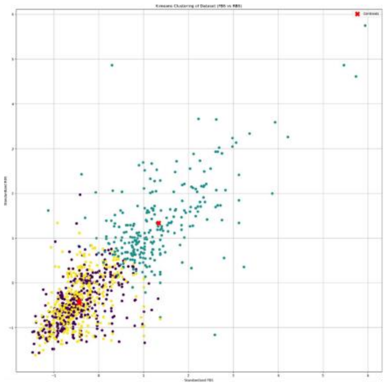


Figure 7: FBS vs RBS

The visualizations, including both 3D and 2D scatter plots, further illustrate the spatial distribution of data points across the clusters. These plots, which show the clustering boundaries and centroids, allow for a better understanding of how the data is structured and how well the clusters are separated. The calculated Silhouette Score quantitatively validates the clustering performance, indicating well-defined clusters with clear separations, thereby confirming the efficacy of the k-means algorithm in this context.

From a mathematical standpoint, the k-means algorithm's efficiency and convergence properties make it a robust tool for partitioning datasets. Its computational complexity O(n.k.t.d.)[The notation O(n.k.t.d.) represents the time complexity of the k-means clustering algorithm. In this context: n is the number of data points, k is the number of clusters, t is the number of iterations the algorithm runs, d is the number of dimensions (features) in the dataset], ensures suitability for large datasets, while its iterative nature guarantees convergence to a local minimum. Despite its sensitivity to initial centroid placement and assumptions of spherical clusters, the algorithm provides a valuable method for uncovering patterns and correlations within medical data. These insights can inform personalized treatment plans, improving patient outcomes by targeting the specific needs identified within each cluster. Future research could focus on refining the algorithm's robustness and exploring its application to different medical datasets to enhance healthcare decision-making.

5 Acknowledgement:

Sowparnika Ayurnikethan, SreeRamakrishna Lane, Poonkunnam,Thrissur.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] Sarah Ben Amor, Fateh Belaid, Ramzi Benkraiem, Boumediene Ramdani & Khaled Guesmi.”Multi-criteria classification, sorting, and clustering: a biblio- metric review and research agenda.”Annals of Operations Research, Vol. 325, pages 771-793, (2023)

[2] Mohiuddin Ahmed, Raihan Seraj and Syed Mohammed Shamsul Islam.”The k-means Algorithm: A Comprehensive Survey and Performance Evalua- tion.”Electronics,2020, 9, 1295

[3] Jan Pablo Burgard, Carina Moreira Costa, Martin Schmidt.”Robustification of the k-means clustering problem and tailored decomposition methods: when more conservative means more accurate.”Annals of Operations Research, Vol.339, pages 1525-1568, (2024)

[4] Marina Gul and M. Abdul Rehman. ”Big data: an optimized approach for cluster initialization” Journal of Big Data, 10, Article number: 120(2023)

[5] Yiqun T. Chen, Daniela M.Witten.”Selective Inference for K-Means Clustering .”Journal of Machine Learning Research, Vol.24, (2023), 1-20.

[6] Md. Zubair, MD. Asif Iqbal, Avijeet Shil, M. J. M. Chowdhury, Mohammad Ali Moni, Iqbal H. Sarker.”An Improved K-means Clustering Algorithm To- wards an Efficient Data-Driven Modeling”Annals of Data Science, Vol.11, pages 1525-1544, (2024)

[7] Abiodun M. Ikotun, Absalom E. Ezugwu, Laith Abualigah, Belal Abuhaija, Jia Heming.”K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data.”Information Sciences, vol.622, April 2023, pages 178-210.

[8] KRISTINA P. SINAGA AND MIIN-SHEN YANG.”Unsupervised K-Means Clustering Algorithm”IEEE Access,vol.8, 2020.

[9] Joonas Hlen, Tommi Kken and Tuomo Rossi.”Improving Scalable K- Means++”Algorithms, Vol.14, Issue 1.

[10] Renato Cordeiro de Amorim, Vladimir Makarenkov.”On k-means iterations and Gaussian clusters” Neurocomputing, Vol.533, 7 October 2023, 126547.

[11] Ahmed Fahim.”K and starting means for k-means algorithm” Journal of Com- putational Science,Vol.55, October 2021, 101445.

[12] Rani Rotul Muhima, Muchamad Kurniawan, Septiyawan Rosetya Wardhana, Anton Yudhana, Sunardi, Mitra Adhimukti ”An improved clustering based on K-means for hotspots data” Indonesian Journal of Electrical Engineering and Computer Science, Vol.31, No.2, August 2023, pp. 1109-1117.

[13] Liang P., Bayatti H. A. ”Automatic Knowledge Integration Method of English Translation Corpus Based on Kmeans Algorithm” Applied Mathematics and Nonlinear Sciences, 8(1)(2023), 381-388.

[14] Abdulnassar A. A., Nair L.R. ”Performance analysis of Kmeans with modified initial centroid selection algorithms and developed Kmeans9+model.” Measurement: Sensors(2023, Volume 25, February 2023,

100666.

[15] Guo C.F., Wang X.M. Shi Chenyu, Jin Hong . ”Corn leaf image segmentation based on improved Kmeans algorithm.” Journal of North University of China (Natural Science Edition),Volume(Issue):2021,42(6).

[16] Shyr-Shen Yu, Shao-Wei Chu, Chuin-Mu Wang, Yung-Kuan Chan, Ting-Cheng Chang.”Two improved k-means algorithms”, Applied Soft Computing, Volume 68, July 2018, Pages 747-755.

[17] Sri Ngudi Wahyuni, Nazmun Nahar Khanom, Yuli Astuti. ”K-Means Algorithm Analysis for Election Cluster Prediction”, INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION, Visualization, 7(1) - March 2023, 1-6