

A Review on Analysis of K-Means Clustering Machine Learning Algorithm based on Unsupervised Learning

Zahraa Radhi Waad^{1*}

Computer Science Department, College of Computers Sciences and Mathematics¹

zahraa.waad@utq.edu.iq^{1*}

ARTICLE INFO

Received: 20 September 2026;

Revised: 10 October 2026;

Revised: 20 October 2026;

Accepted: 22 October 2026;

Volume 4, Number 1

2027, pp 1-16

<https://doi.org/10.61401/rmaps.v4i1>

Corresponding author:

Zahraa Radhi Waad

Computer Science Department, College
of Computers Sciences and Mathematics

E-mail: zahraa.waad@utq.edu.iq

ABSTRACT

Purpose: This study revisits the K-means clustering algorithm as a core technique of unsupervised machine learning and explains, in plain terms, how it groups unlabeled data into meaningful clusters.

Methodology: The paper combines a narrative review of studies published mostly between 2021 and 2026 with a manual, step-by-step computation of K-means on a small one-dimensional dataset of nine values, using Euclidean distance and $K=2$. The final partition is then checked with the within-cluster sum of squares and the silhouette coefficient for K values from one to four.

Results: The algorithm stabilized after three reassignment rounds and produced the clusters $\{2, 3, 4, 10, 11, 12\}$ and $\{20, 25, 30\}$, with centroids of 7 and 25. The within-cluster sum of squares fell from 348.00 to 150.00, and the mean silhouette coefficient reached 0.660, although $K=3$ gave a higher value of 0.711.

Conclusions: K-means is simple, fast, and easy to interpret, yet its output depends strongly on the chosen number of clusters and on the starting centroids.

Limitations: The worked example relies on a small one-dimensional dataset and does not test high-dimensional, noisy, or streaming data.

Contributions: The paper offers a transparent teaching case that links the classic procedure to current validation practice and to recent applied studies.

Keywords: *Clustering Algorithm, Euclidean Distance, K-Means Algorithm, Machine Learning, Unsupervised Learning*

How to Cite: Waad, Z. R. (2027). A Review on Analysis of K-Means Clustering Machine Learning Algorithm based on Unsupervised Learning. *Review of Multidisciplinary Academic and Practice Studies*, 4(1), 1-16.

1. Introduction

Few ideas in computing have moved from the laboratory into daily life as quickly as machine learning. Recommendation engines, fraud alerts, medical image readers, and traffic cameras now rely on models that learn patterns from data rather than following rules written line by line by a programmer. [Sarker \(2021\)](#) describes machine learning as a family of methods that give computers

the ability to improve their performance through experience, and he groups these methods into supervised, unsupervised, semi-supervised, and reinforcement learning. Each family answers a different kind of question, and choosing the wrong one often wastes both data and effort.

Supervised learning works when every training example carries a label. A model that learns to separate ripe from unripe fruit, for instance, is shown thousands of images that a person has already tagged, and it gradually learns which visual cues predict each tag. Recent applied work illustrates how far this approach has spread. [Gunawan and Artaye \(2026\)](#) trained a convolutional neural network to judge banana freshness from photographs, [Muhammad and Fauzi \(2026\)](#) used a support vector machine to classify air quality levels in Jakarta, and [Purdadi and Agus \(2025\)](#) applied the YOLO detector to estimate traffic density in Bandar Lampung. These studies share one requirement: somebody had to label the data first.

Labels, however, are expensive. They demand time, domain knowledge, and often several rounds of checking. In many settings, such as customer transaction logs, sensor streams, or collections of student records, the data arrive in large volumes with no labels at all. Unsupervised learning was developed for exactly this situation. Instead of predicting a known answer, an unsupervised model looks for structure that is already present in the inputs, such as groups of similar records, unusual observations, or hidden dimensions that summarize many variables. [Sathya and Abraham \(2013\)](#) showed early on that the two families of learning behave very differently when the same pattern classification task is approached with and without labels, and that distinction still shapes how analysts design their projects today.

Clustering is the most widely used unsupervised task. Its goal is to divide a set of objects into groups so that objects in the same group resemble one another more than they resemble objects in other groups. [Xu and Tian \(2015\)](#) organized the large body of clustering methods into partition-based, hierarchical, density-based, grid-based, and model-based families. More recent surveys by [Ezugwu et al. \(2022\)](#) and [Wani \(2024\)](#) confirm that this taxonomy still holds, although each family has grown new branches to handle big data, streaming inputs, and mixed data types. Among all of these methods, K-means remains the one most often taught in introductory courses and most often used as a baseline in applied research.

The popularity of K-means is easy to understand. The algorithm asks the user for a single number, K , which states how many clusters are expected. It then places K centroids, assigns each object to its nearest centroid, recomputes each centroid as the mean of its members, and repeats the last two steps until the assignments stop changing. [Ikotun et al. \(2023\)](#) note that this procedure is fast, scales almost linearly with the number of objects, and produces results that non-specialists can read without difficulty. Those strengths explain why K-means appears in studies of poverty mapping ([Tadi & Ningsi, 2023](#)), malaria incidence ([Sroyer et al., 2022](#)), social aid distribution ([Utami et al., 2023](#)), and small business revenue ([Sunarti et al., 2025](#)), to name only a few recent Indonesian examples.

The same simplicity, though, hides several weaknesses that beginners often overlook. The result depends on where the first centroids are placed, so two runs on the same data can produce different groupings ([Romanuke, 2023](#)). The user must decide K in advance, and a poor choice can merge distinct groups or split natural ones apart. Because the method relies on distances to a mean, it prefers compact, roughly spherical clusters of similar size, and it reacts badly to outliers and to clusters of uneven spread ([Raeisi & Sesay, 2022](#)). These limits do not make K-means obsolete, but they do mean that its output should never be accepted without some form of validation.

A large share of published tutorials and review papers on K-means stop at describing the procedure. They list the steps, show a diagram, and move on to applications. Fewer papers walk the reader through a complete numerical example, show how the centroids move from one iteration to the next, and then test whether the final partition is actually a good one. Even fewer connect that small example to the validation tools that current research treats as standard, such as the elbow curve built from the within-cluster sum of squares and the silhouette coefficient ([Shutaywi & Kachouie, 2021](#)). As a result, students and early career researchers often know how to run K-means in a software

library but cannot explain why it produced a particular answer or whether that answer can be trusted.

The state of the art in this area has moved in several directions at once. [Ikotun et al. \(2023\)](#) catalogued dozens of K-means variants that change the initialization, the distance measure, or the update rule. Hybrid designs now pair K-means with metaheuristics such as firefly optimization ([Ezugwu et al., 2022](#)) or particle swarm optimization ([Masoud, 2023](#)), and other studies combine it with deep text representations ([Setiawan et al., 2025](#)) or with supervised classifiers for prediction tasks ([Xiahou & Harada, 2022](#)). Reviews of rapid and stream clustering ([Mahnoor et al., 2024](#); [Nordahl et al., 2025](#)) show that evaluation itself has become a research topic. What remains less developed is material that bridges this advanced literature and the basic mechanics that newcomers need to master first.

The novelty of the present paper lies in that bridge. It pairs a review of literature published mostly in the last five years with a fully worked K-means example in which every assignment, every centroid update, and every intermediate value is shown. It then goes one step further than a typical classroom example by measuring the quality of the result with the within-cluster sum of squares and the silhouette coefficient for several values of K . This reveals something that the original two-cluster solution hides, namely that a three-cluster solution fits the same data better. In this way the paper turns a simple demonstration into a small case study of why validation matters.

The aim of the study is therefore to explain the K-means algorithm within the broader context of unsupervised learning, to demonstrate its behavior on a transparent dataset, and to relate that behavior to recent research findings. The rest of the paper is organized as follows. Section 2 reviews related work on unsupervised learning, clustering families, and K-means applications. Section 3 describes the clustering methodology, the distance measure, and the algorithmic steps. Section 4 presents the worked example together with the validation results and a discussion of their meaning. Section 5 closes the paper with conclusions, limitations, and directions for future study.

2. Literature Review and Hypothesis/es Development

2.1 Related Work

Research on unsupervised learning has grown so quickly that any review must be selective. This section focuses on four strands that bear directly on the present study, namely general surveys of machine learning and clustering, studies that examine the internal mechanics of K-means, comparisons between K-means and competing methods, and applied work that shows how the algorithm performs in real settings.

2.1.1 Machine Learning and Unsupervised Learning

[Sarker \(2021\)](#) provides one of the most cited overviews of modern machine learning. His review explains how classification, regression, clustering, association rule learning, and dimensionality reduction support applications in cybersecurity, smart cities, health care, e-commerce, and agriculture. A key point in his discussion is that the choice of algorithm must follow the nature of the data and the goal of the analysis, not the popularity of the method. For unlabeled data, clustering becomes the natural starting point because it allows the analyst to see how records group together before any further modeling is attempted.

Unsupervised learning is also spreading into fields that once relied almost entirely on expert judgment. [Partamies, Dol, Teissier, Juusola, Syrjäso, and Mulders \(2024\)](#), for example, used unsupervised learning to detect auroral breakups in large archives of all-sky camera images that had never been labeled by hand. [Abed and Abdullah \(2024\)](#) applied unsupervised algorithms to detect and localize fingerprint regions in low-quality palm images, a task in which labeled examples are scarce. [dos Santos and Bruni \(2024\)](#) used unsupervised learning to examine infrared spectra of new psychoactive substances generated by density functional theory, showing that the method can reveal chemical groupings that are not obvious from the raw spectra. These cases share a common thread: the analyst begins with data that have no reliable labels and needs a method that can propose a structure first.

The same trend is visible in the journals that publish applied artificial intelligence research in Indonesia. [Widyastuti and Tarumingkeng \(2025\)](#) examined how artificial intelligence in telemedicine shapes customer satisfaction, while [Khonqulova \(2026\)](#) proposed a framework in which large language models drive real-time sentiment analysis in customer service dashboards. [Ahludzikri, Hasibuan, Aziz, and Triloka \(2025\)](#) used bibliometric mapping to trace the growth of lung cancer detection research, an approach that itself relies on grouping documents by shared terms and citations. Such studies show that intelligent methods are now part of everyday decision making in business, health, and public services, and that many of them depend on grouping data before any prediction is made.

2.2 Families of Clustering Methods

[Xu and Tian \(2015\)](#) produced an early and detailed survey that remains a common reference point for classifying clustering algorithms. They distinguished methods that partition data around representative points, methods that build a tree of nested clusters, methods that grow clusters through dense regions, and methods that assume a statistical model. [Ezugwu et al. \(2022\)](#) updated this picture by reviewing hundreds of studies and adding new categories for nature-inspired, deep, and automatic clustering. They also highlighted open problems that remain unsolved, including how to choose the number of clusters, how to handle high-dimensional data, and how to evaluate results when no ground truth exists.

[Wani \(2024\)](#) focused on the limitations of popular clustering algorithms and on the remedies proposed for each. His analysis shows that partition-based methods such as K-means are efficient but sensitive to initialization and outliers, that hierarchical methods are easy to visualize but expensive for large datasets, and that density-based methods find clusters of arbitrary shape but struggle when density varies across the data. [Mahnoor et al. \(2024\)](#) reached a similar conclusion in their review of rapid clustering approaches, adding that speed and accuracy often pull in opposite directions when datasets become very large. For data that arrive continuously, [Nordahl et al. \(2025\)](#) argued that evaluation must also change, because a partition that is good at one moment may degrade as new records arrive.

Text data introduce further challenges. [Ahmed et al. \(2022\)](#) surveyed short text clustering and found that sparse word vectors, informal language, and very short documents weaken the distance measures on which most algorithms depend. Their review explains why recent studies increasingly combine clustering with pretrained language models. [Setiawan et al. \(2025\)](#) offer a clear example: they encoded student complaints with an Indonesian BERT model and then grouped the resulting vectors with K-means, which produced topic clusters that matched human judgment more closely than word-count features did.

2.3 Mechanics and Improvements of K-Means

[Ikotun et al. \(2023\)](#) wrote the most comprehensive recent review dedicated to K-means. They traced the algorithm from its origins to dozens of modern variants and grouped the improvements into changes to initialization, to the number of clusters, to the distance measure, and to computational efficiency for big data. Their review stresses that most variants keep the basic assign-and-update loop intact, which is why understanding the original algorithm remains essential.

Initialization has received particular attention. [Romanuke \(2023\)](#) showed that partitioning a large set of flat points in advance and selecting initial centroids from that preliminary partition can speed up K-means considerably while keeping the quality of the final clusters. [Ikotun and Ezugwu \(2022\)](#) took a different path, using a firefly algorithm with adaptive mutation to search for good starting centroids and to estimate the number of clusters automatically in high-dimensional datasets. [Masoud \(2023\)](#) combined K-means with particle swarm optimization to plan the recharging of shared electric scooters, demonstrating that hybrid designs can carry the algorithm into routing and logistics problems.

The distance measure is another frequent target for improvement. [Raeisi and Sesay \(2022\)](#) proposed a distance metric designed for clusters of uneven size and spread, a case in which standard Euclidean distance tends to pull boundaries toward the larger groups. [Candra et al. \(2023\)](#) compared

the simple matching coefficient with Euclidean distance for classifying tourism destinations and found that the better choice depends on whether the attributes are categorical or numerical. [Haryati and Surono \(2022\)](#) reported a similar sensitivity to distance measures in fuzzy subtractive clustering. Together, these findings show that the familiar Euclidean formula is a reasonable default but not a universal answer.

Choosing K is the third major issue. [Shutaywi and Kachouie \(2021\)](#) examined silhouette analysis as a tool for judging clustering quality and showed that it can identify the number of clusters more reliably than visual inspection alone. The silhouette value compares how close each object is to members of its own cluster with how close it is to the nearest other cluster, so values near one indicate compact and well separated groups. The elbow method, which plots the within-cluster sum of squares against K and looks for the point where further increases bring little improvement, is often used alongside it. Both tools are applied in the worked example of Section 4.

2.4 K-Means Compared With Other Methods

Several recent studies compare K-means with alternative algorithms on the same data. [Kenger et al. \(2023\)](#) clustered cities according to their smart city performance using fuzzy C-means, K-means, and K-medoids and found that the three methods agreed on broad patterns but differed on borderline cities. [Siagian et al. \(2022\)](#) compared K-means and K-medoids for customer segmentation in e-commerce transactions and reported that K-medoids handled extreme purchase values better, because it uses actual data points as cluster centers instead of averages. [Setyawan and Purwatiningsih \(2024\)](#) compared K-means with agglomerative clustering for maritime skill certification data and showed that the hierarchical approach gave a useful view of nested groups, while K-means was faster and easier to interpret. [Sroyer et al. \(2022\)](#) reached a comparable conclusion when they compared single linkage with K-means for malaria data in Papua.

These comparisons suggest that K-means rarely wins on every criterion. Its strength lies in speed, simplicity, and clarity, and its weakness lies in its assumptions about cluster shape and its sensitivity to outliers. In practice, many analysts run K-means first to obtain a quick baseline and then test whether a more flexible method adds enough value to justify its extra cost.

2.5 Applications of K-Means

Customer analytics is one of the most active application areas. [Anitha and Patil \(2022\)](#) combined the recency, frequency, and monetary model with K-means to describe customer purchase behavior, and [Hui et al. \(2025\)](#) used K-means to segment Chinese tourists according to their intentions to buy duty-free luxury goods. [Xiahou and Harada \(2022\)](#) first grouped business-to-consumer e-commerce customers with K-means and then used a support vector machine to predict churn within each group, which improved prediction accuracy. [Hafidz and Lestari \(2023\)](#) applied K-means to ease the sparsity and scalability problems of collaborative filtering, showing that clustering can make recommendation systems more efficient. In the same spirit, [Gustina and Wibowo \(2026\)](#) designed a book recommendation system for academic libraries, and [Nisa and Ali \(2025\)](#) used fuzzy matching to improve chatbot responses in an online fashion store. Although those two studies did not use K-means, they show the growing demand for systems that group users and items by similarity.

Education is another field where K-means is widely used. [Maryono et al. \(2022\)](#) grouped programming students according to their learning styles so that teachers could form more balanced classes, and [Khomarudin and Hidayat \(2022\)](#) used K-means to examine how Quran memorization activities related to students' learning outcomes. [Wardhana and Winarno \(2024\)](#) applied the algorithm to identify the best football players from performance statistics, an example that shows how the same logic can move from classrooms to sports analytics.

In finance and public policy, [Ridwan et al. \(2022\)](#) used K-means and average linkage clustering to select stocks for a mean value at risk portfolio, while [Tadi and Ningsi \(2023\)](#) and [Utami et al. \(2023\)](#) used the algorithm to group regions by poverty indicators and households by eligibility for social assistance. [Sunarti et al. \(2025\)](#) grouped micro, small, and medium enterprises in Jakarta by the number of businesses and their revenue. Such studies give policymakers a structured way to target programs at the groups that need them most.

Image and signal analysis complete the picture. [Osman and Yaba \(2022\)](#) combined color space conversion with K-means to segment subtypes of acute lymphocytic leukemia in blood images, [Nasor and Obaid \(2021\)](#) used K-means as one stage of a pipeline for segmenting osteosarcoma in magnetic resonance images, and [Wu et al. \(2021\)](#) proposed a convex version of K-means that improves color image segmentation. [Alalayah et al. \(2023\)](#) used K-means together with t-distributed stochastic neighbor embedding to prepare electroencephalography signals for early detection of epileptic seizures. Finally, [Syauqi et al. \(2023\)](#) showed that the quality of K-means results depends on how missing values are handled, since imputation with nearest neighbors and with means led to different clusters on the same benchmark data.

Taken together, the literature shows a clear pattern. K-means continues to serve as a baseline, a building block, and an applied tool across many disciplines, and the main research effort now goes into fixing its known weaknesses rather than replacing it. At the same time, few papers combine a step-by-step demonstration of the algorithm with a formal check of the result. The present study addresses that gap.

3. Research Methodology

This study follows a two-part design. The first part is a narrative review of literature on unsupervised learning and K-means clustering, drawn mainly from national and international journals published between 2021 and 2026, with a small number of earlier sources kept for historical context. Sources were selected for their direct relevance to the mechanics, validation, comparison, or application of K-means. The second part is a demonstration in which the algorithm is applied by hand to a small dataset so that each step can be observed and checked. The demonstration is followed by a validation step that uses the within-cluster sum of squares and the silhouette coefficient. This section explains the ideas behind that demonstration.

3.1 The Idea of Clustering

Unsupervised learning relies heavily on clustering, which means assembling sets of data points that are statistically similar to one another. A simple example makes the idea concrete. Imagine a basket that contains a mix of fruits, and suppose the task is to separate them so that each kind ends up in its own pile. A clustering algorithm does not know the names of the fruits. It simply looks for shared properties. It might first notice color, so that yellow, red, and blue items begin to drift apart, and then refine those piles by size and shape. When the process ends, each pile contains fruits that look alike, even though no one ever told the algorithm what a mango or a plum is.

Supervised learning handles the same basket in a very different way. Before training begins, a person describes each fruit with a set of features and attaches a label to it. An apple might be recorded as red, round, about twenty grams, and five centimeters tall, with the label apple. A mango might be recorded as yellow, oval, and sweet, with the label mango. The model studies these pairs of features and labels and learns which combinations predict each fruit. When a new fruit arrives, it uses what it has learned to predict the label. The key difference is that supervised learning needs those labels from the start, while unsupervised learning must discover the groups on its own ([Sathya & Abraham, 2013](#); [Sarker, 2021](#)).

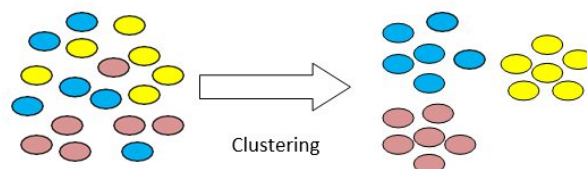


Figure 1. Generating diverse groups through the application of clustering algorithms

Figure 1 summarizes this idea. On the left side, colored points representing different objects are scattered without any order. After a clustering algorithm is applied, shown by the arrow, the same points are rearranged into three compact groups, each containing objects of one color. The figure

does not show labels because the algorithm never sees them. It recovers the groups only from the similarity of the objects, which is the essence of unsupervised learning.

3.2 The K-Means Algorithm

K-means is the best known partition-based clustering algorithm. It processes data points and assigns each one to a cluster so that items with similar attributes end up together, while items with different attributes are placed in separate clusters. Each cluster is represented by a centroid, which is the average position of all the points that belong to it. Because the centroid is an average, it does not have to coincide with any real data point, a feature that distinguishes K-means from K-medoids ([Kenger et al., 2023](#); [Siagian et al., 2022](#)).

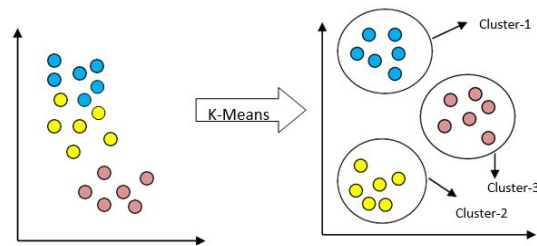


Figure 2. Illustration of data distribution before and after K-Means clustering

Figure 2 shows a set of points before K-means is applied. Blue, yellow, and red points overlap in the same space, and no group boundaries are visible. Three clusters now stand out, labeled Cluster 1, Cluster 2, and Cluster 3, and each circle encloses points that are close to one another. The comparison makes clear that K-means does not create new information. It reorganizes what is already present so that the structure of the data becomes visible.

The algorithm can be described as a short sequence of steps, which are summarized in Table 1. The procedure starts by choosing K and placing K initial centroids, either at random or by a more careful method. It then alternates between two operations. In the assignment step, every data point is linked to the centroid that lies closest to it. In the update step, each centroid is moved to the mean of the points assigned to it. These two steps repeat until no point changes its cluster, at which point the algorithm is said to have converged.

Table 1. Steps of the K-means clustering algorithm

Step	Action	Purpose
1	Choose the number of clusters K	Defines how many groups the algorithm will form
2	Select K initial centroids from the data	Provides starting points for the search
3	Assign each data point to the nearest centroid	Forms provisional clusters based on distance
4	Recompute each centroid as the mean of its members	Moves each center toward the middle of its cluster
5	Reassign every data point to the nearest updated centroid	Corrects points that now lie closer to another center
6	Return to Step 4 if any point changed cluster, otherwise stop	Repeats the loop until assignments are stable
7	Report the final clusters and centroids	Delivers the partition for interpretation

Table 1 lists the seven steps together with the role that each one plays. The first two steps prepare the search, the middle steps form the repeating core of the algorithm, and the last step delivers the result. The table also makes the main weakness of the procedure easy to see. The final clusters

depend on the choices made in Steps 1 and 2, so a different value of K or a different set of starting centroids can lead to a different answer ([Romanuke, 2023](#); [Ikotun & Ezugwu, 2022](#)).

3.3 Distance Measure

The assignment step needs a way to measure how close a point is to a centroid. The most common choice is the Euclidean distance, which is the straight-line distance between two points. For two-dimensional data it is calculated with the following Formula 1.

$$D = \sqrt{(X1 - C1)^2 + (X2 - C2)^2} \quad (1)$$

where $X1$ and $X2$ are the coordinates of a data point, and $C1$ and $C2$ are the coordinates of the centroid of the cluster being considered.

The formula squares the difference along each axis, adds the squares, and takes the square root of the sum. A small value of D means that the point lies close to the centroid, and a large value means that it lies far away. @figureillustrates the idea with two locations, P and q , joined by a straight line. The length of that line is the Euclidean distance between them. In the one-dimensional example used in Section 4, the formula reduces to the absolute difference between a value and a centroid, which makes each calculation easy to follow.

Euclidean distance is the default in most K-means software, but it is not the only option. [Candra et al. \(2023\)](#) found that the simple matching coefficient works better for categorical attributes, and [Raeisi and Sesay \(2022\)](#) showed that clusters of uneven size benefit from a modified metric. Because K-means minimizes squared Euclidean distances to the mean, replacing the distance measure usually also requires a change in how the cluster centers are updated.

3.4 Validation Measures

Two measures are used to judge the quality of the final partition. The first is the within-cluster sum of squares, often abbreviated as WCSS. It adds up the squared distance between every point and the centroid of its own cluster. A smaller WCSS means tighter clusters, but this measure always decreases as K grows, so it cannot choose K on its own. Analysts therefore plot WCSS against K and look for an elbow, the point after which adding more clusters brings only small gains.

The second measure is the silhouette coefficient ([Shutaywi & Kachouie, 2021](#)). For each point, it compares the average distance to other members of the same cluster with the average distance to members of the nearest neighboring cluster. The result ranges from minus one to one. Values close to one indicate that the point sits comfortably inside its cluster, values near zero indicate that it lies on a boundary, and negative values suggest that it may belong to another cluster. The mean silhouette over all points gives a single summary of clustering quality and can be compared across different values of K .

The K-means clustering technique partitions items into distinct clusters whose number is set by K . If $K=2$, for instance, the objects are divided into two clusters, which in the example below are called K_1 and K_2 . The next section applies this full methodology to a concrete dataset.

4. Results and Discussions

4.1 Worked Example of K-Means

The K-means method is easiest to understand through a simple example. As shown in Formula 2, a dataset consisting of nine values is used to demonstrate the clustering process, with the number of clusters set to two ($K=2$).

$$[2, 3, 4, 10, 11, 12, 20, 25, 30], \quad K = 2 \quad (2)$$

The first task is to choose starting centroids. Because $K=2$, two initial mean values are needed. In this example they are taken directly from the dataset, as shown in Formula 3.

$$M_1 = 4, \quad M_2 = 12 \quad (3)$$

With these centroids, two clusters K_1 and K_2 are formed. A value joins K_1 if it lies closer to 4 and joins K_2 if it lies closer to 12. The values 2, 3, and 4 are closest to M_1 , while the value 10 lies two units from 12 and six units from 4, so it joins K_2 together with all the larger values. The resulting initial cluster assignments are presented in Formula 4.

$$K_1 = [2, 3, 4], \quad K_2 = [10, 11, 12, 20, 25, 30] \quad (4)$$

Because the starting centroids were chosen arbitrarily, they must now be updated. As shown in Formula 5, M_1 is recalculated as the mean of the values assigned to cluster K_1 .

$$M_1 = \frac{2 + 3 + 4}{3} = \frac{9}{3} = 3 \quad (5)$$

As shown in Formula 6, M_2 becomes the mean of the values in K_2 .

$$M_2 = \frac{10 + 11 + 12 + 20 + 25 + 30}{6} = \frac{108}{6} = 18 \quad (6)$$

The new means of 3 and 18 differ from the starting values of 4 and 12, so the points must be reassigned. The value 10 now lies seven units from M_1 and eight units from M_2 , so it moves to K_1 . The value 11 lies eight units from M_1 and seven units from M_2 , so it stays in K_2 . The updated cluster assignments are presented in Formula 6, while the recalculation of centroid M_1 is shown in Formula 7.

$$K_1 = [2, 3, 4, 10], \quad K_2 = [11, 12, 20, 25, 30] \quad (7)$$

As shown in Formula 8, M_1 is recomputed as the mean of the updated K_1 cluster.

$$M_1 = \frac{2 + 3 + 4 + 10}{4} = 4.75 \approx 5 \quad (8)$$

As shown in Formula 9, M_2 is recomputed as the mean of the updated K_2 cluster.

$$M_2 = \frac{11 + 12 + 20 + 25 + 30}{5} = 19.6 \approx 20 \quad (9)$$

The centroids have moved again, so another reassignment is required. The values 11 and 12 are now closer to M_1 than to M_2 , since 12 lies 7.25 units from M_1 and 7.6 units from M_2 . Both values therefore move to K_1 .

The recalculation of centroid M_l based on the updated cluster assignments is presented in Formula 10.

$$K_1 = [2, 3, 4, 10, 11, 12], \quad K_2 = [20, 25, 30] \quad (10)$$

As shown in Formula 11 M_l is recomputed as the mean of the K_l cluster.

$$M_1 = \frac{2 + 3 + 4 + 10 + 11 + 12}{6} = 7 \quad (11)$$

As shown in Formula 12 M_2 is recomputed as the mean of the K_2 cluster.

$$M_2 = \frac{20 + 25 + 30}{3} = 25 \quad (12)$$

With centroids of 7 and 25, every value is assigned once more to its nearest centroid. The boundary between the two clusters now lies at 16, halfway between the centroids, so all values from 2 to 12 stay in K_1 and all values from 20 to 30 stay in K_2 . To confirm convergence, centroid M_l is recalculated using the unchanged cluster assignments, as shown in Formula 13.

$$K_1 = [2, 3, 4, 10, 11, 12], \quad K_2 = [20, 25, 30] \quad (13)$$

To confirm convergence, the means are calculated again for the unchanged clusters.

$$M_1 = \frac{2 + 3 + 4 + 10 + 11 + 12}{6} = 7 \quad (14)$$

$$M_2 = \frac{20 + 25 + 30}{3} = 25 \quad (15)$$

The means in Formula 14 and Formula 15 are identical to those in Formula 11 and 12. No point has changed its cluster, so the algorithm stops, and the final result is given in Formula 16.

$$K_1 = \{2, 3, 4, 10, 11, 12\}, \quad K_2 = \{20, 25, 30\} \quad (16)$$

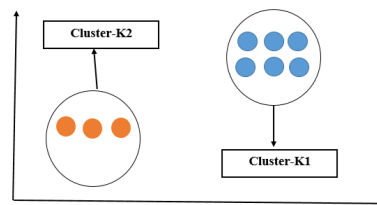


Figure 4. K1 comprises 6 data points (2, 3, 4, 10, 11, 12) and K2 comprises 3 data points (20, 25, 30)

Figure 4 shows the final partition. Cluster K_1 , on the right, contains the six smaller values 2, 3, 4, 10, 11, and 12, while Cluster K_2 contains the three larger values 20, 25, and 30. The two circles do not overlap, which reflects the clear gap between 12 and 20 in the data. The figure also shows that the clusters are not equal in size, a result that K-means allows as long as each point is closer to its own centroid than to the other one.

4.2 Convergence of the Algorithm

Table 2 summarizes how the clusters and centroids changed during the run. Each row corresponds to one assignment round, and the last column reports the within-cluster sum of squares calculated with the exact, unrounded means.

Table 2. Summary of K-means iterations for the example dataset with $K = 2$

Round	Centroids Used for Assignment	Cluster K1	Cluster K2	Updated Means	WCSS
1	4 and 12	2, 3, 4	10, 11, 12, 20, 25, 30	3 and 18	348.00
2	3 and 18	2, 3, 4, 10	11, 12, 20, 25, 30	4.75 and 19.6	307.95
3	4.75 and 19.6	2, 3, 4, 10, 11, 12	20, 25, 30	7 and 25	150.00
4	7 and 25	2, 3, 4, 10, 11, 12	20, 25, 30	7 and 25	150.00

Table 2 shows that the WCSS fell from 348.00 in the first round to 307.95 in the second and to 150.00 in the third, after which it stayed constant. This steady decline is not a coincidence. Each assignment step and each update step can only lower the WCSS or leave it unchanged, which is why K-means always reaches a stable state after a finite number of rounds. The table also shows that the largest gain came in the third round, when the values 11 and 12 moved to K1. That move corrected the poor starting position of M2, which had been placed at 12, well inside the group of small values.

The example confirms an observation made by [Ikotun et al. \(2023\)](#) and [Romanuke \(2023\)](#). The first centroids matter. If the starting values had been 3 and 25, the algorithm would have reached the final partition in a single round. A poor start does not always prevent convergence to a good answer, but it can slow the process and, in more complex data, lead to a weaker local solution.

4.3 Validating the Number of Clusters

The original example assumed $K=2$, which is a common choice in teaching. Yet that assumption deserves a test. Table 3 reports the WCSS and the mean silhouette coefficient for K values from one to four. For each value of K , the partition with the lowest WCSS was identified by checking every possible split of the sorted data, which is feasible for a dataset of nine values.

Table 3. Within-cluster sum of squares and silhouette coefficient for $K = 1$ to 4

K	Best partition	WCSS	Mean silhouette
1	{2, 3, 4, 10, 11, 12, 20, 25, 30}	798.00	Not defined
2	{2, 3, 4, 10, 11, 12}, {20, 25, 30}	150.00	0.660
3	{2, 3, 4}, {10, 11, 12}, {20, 25, 30}	54.00	0.711
4	{2, 3, 4}, {10, 11, 12}, {20}, {25, 30}	16.50	0.607

Table 3 shows that the two-cluster solution found by hand is in fact the best possible two-cluster partition, since no other split of the data gives a lower WCSS. That result confirms that the manual computation was correct. The table also shows something more interesting. Moving from one cluster to two reduces the WCSS by 648 units, and moving from two to three reduces it by a further 96 units. After that, the gain drops to 37.5 units. The elbow therefore appears at $K=3$ rather than at $K=2$.

The silhouette coefficient tells the same story. The mean value rises from 0.660 for two clusters to 0.711 for three clusters and then falls to 0.607 for four. The three-cluster solution separates the values 10, 11, and 12 into their own group, which matches what the eye sees when the data are written on a number line: three tight groups separated by gaps of six and eight units. In the two-cluster solution, these middle values are forced into K_1 , where they sit far from the small values 2, 3, and 4. The individual silhouette value for the point 12 is only 0.538 in that solution, the second lowest of all nine points, which signals that the point is only loosely attached to its cluster.

This finding does not mean that the two-cluster answer is wrong. If the purpose of the analysis requires exactly two groups, for example to separate low and high priority cases, then $K=2$ is a legitimate choice, and K-means delivered the best possible split for that purpose. The finding does mean that the choice of K should be justified by the goal of the study and checked with a validation measure, not taken for granted. [Shutaywi and Kachouie \(2021\)](#) make the same argument, and the small example here shows how easily a reasonable assumption can hide a better structure.

4.4 Discussion

The results of the worked example connect directly to the patterns reported in recent literature. The first connection concerns simplicity. Every step in the example could be checked with basic arithmetic, and the final clusters could be explained in a single sentence. This transparency is the main reason why K-means remains popular in applied studies in public policy, education, and business ([Tadi & Ningsi, 2023](#); [Maryono et al., 2022](#); [Anitha & Patil, 2022](#)). Decision makers are more likely to act on a segmentation they can understand.

The second connection concerns sensitivity to the starting point. In the example, a starting centroid placed inside the group of small values delayed convergence by two rounds. In large, high-dimensional datasets, the same problem can trap the algorithm in a poor local solution. This explains why so much recent research focuses on smarter initialization, either through preliminary partitioning ([Romanuke, 2023](#)) or through metaheuristic search ([Ikotun & Ezugwu, 2022](#); [Masoud, 2023](#)). For practitioners, the simplest safeguard is to run K-means several times with different starting centroids and keep the solution with the lowest WCSS.

The third connection concerns the number of clusters. The gap between the assumed $K=2$ and the better supported $K=3$ mirrors a problem highlighted in nearly every review of clustering ([Ezugwu et al., 2022](#); [Wani, 2024](#); [Mahnoor et al., 2024](#)). When K is chosen by habit, the resulting clusters may mix groups that should be separate. Validation measures such as the elbow curve and the silhouette coefficient cost little to compute and can prevent misleading conclusions.

The fourth connection concerns the type of data. The example used clean, one-dimensional values with clear gaps. Real data are rarely so cooperative. Missing values can change the clusters, as [Syauqi et al. \(2023\)](#) showed, and categorical or text attributes may require different distance measures or new representations altogether ([Candra et al., 2023](#); [Ahmed et al., 2022](#); [Setiawan et al., 2025](#)). Outliers pose another risk because a single extreme value can pull a centroid away from the bulk of its cluster. In such cases, K-medoids or density-based methods may be more appropriate, as several comparative studies suggest ([Siagian et al., 2022](#); [Kenger et al., 2023](#); [Setyawan & Purwatiningsih, 2024](#)).

The last connection concerns the role of K-means inside larger systems. Many recent studies do not stop at clustering. They use the clusters as input for prediction, recommendation, or targeted policy ([Xiahou & Harada, 2022](#); [Hafidz & Lestari, 2023](#); [Hui et al., 2025](#)). Image and signal studies use K-means as a preprocessing stage before more complex models ([Osman & Yaba, 2022](#); [Nasor & Obaid, 2021](#); [Alalayah et al., 2023](#)). This pattern suggests that the value of K-means today lies not only in its standalone results but also in its ability to organize data for the methods that come after it. Applied studies published in Indonesian outlets, including work on intelligent decision support and data-driven services ([Gustina & Wibowo, 2026](#); [Muhammad & Fauzi, 2026](#); [Purdadi & Agus, 2025](#)), point to a growing local demand for such combined pipelines.

Seen together, these points support a balanced view. K-means is not the most powerful clustering algorithm available, but it is one of the most useful. Its results are easy to compute, easy to explain, and easy to check, and when its assumptions are respected and its output is validated, it provides a reliable first picture of the structure hidden in unlabeled data.

5. Conclusions

This study set out to explain the K-means clustering algorithm within the wider framework of unsupervised machine learning and to show, through a fully worked example, how the algorithm behaves and how its results can be checked. Unsupervised learning deals with data that carry no labels, so the model must find structure on its own. Clustering is the most common way to do this, and K-means is the clustering method that most analysts meet first.

The worked example showed that K-means reaches a stable answer through a short loop of assignment and update steps. Starting from centroids of 4 and 12, the algorithm needed three rounds to settle on the clusters {2, 3, 4, 10, 11, 12} and {20, 25, 30}, with final centroids of 7 and 25. The within-cluster sum of squares fell at every round, from 348.00 to 150.00, which illustrates why the procedure always converges. An exhaustive check confirmed that this partition is the best possible split of the data into two groups.

The validation step added an important insight. Both the elbow in the within-cluster sum of squares and the peak of the silhouette coefficient pointed to three clusters rather than two, with a mean silhouette of 0.711 compared with 0.660. The two-cluster answer is still valid when the purpose of the analysis calls for two groups, but the example shows that the number of clusters should be justified and tested rather than assumed.

Overall, K-means remains a practical and transparent tool. It is fast, easy to implement, and easy to explain to people who are not specialists. Its main weaknesses, namely its dependence on the starting centroids, on a fixed value of K , and on compact clusters of similar shape, can be managed when the analyst runs the algorithm several times, checks the result with validation measures, and considers alternative methods when the data contain outliers or irregular shapes.

Acknowledgement

The author thanks the Computer Science Department, College of Computer Science and Mathematics, University of Thi-Qar, for the academic support and facilities provided during the preparation of this article. The author is also grateful to the anonymous reviewers and the editors for their constructive comments.

References

- Abed, A. J., & Abdullah, D. A. (2024). An automated fingerprint image detection and localization approach-based unsupervised learning algorithms using low-quality biometrics palm data. *JOIV: International Journal on Informatics Visualization*, 8(3-2), 1635-1642. <https://doi.org/10.62527/joiv.8.3-2.1665>
- Ahludzikri, F., Hasibuan, M. S., Aziz, R. A., & Triloka, J. (2025). Bibliometric analysis of detection lung cancer. *Applied AI and Machine Learning Journal*, 1(1), 61-71. <https://doi.org/10.35912/aiml.v1i1.3776>
- Ahmed, M. H., Tiun, S., Omar, N., & Sani, N. S. (2022). Short text clustering algorithms, application and challenges: A survey. *Applied Sciences*, 13(1), 342. <https://doi.org/10.3390/ap13010342>
- Alalayah, K. M., Senan, E. M., Atlam, H. F., Ahmed, I. A., & Shatnawi, H. S. A. (2023). Effective early detection of epileptic seizures through EEG signals using classification algorithms based on t-distributed stochastic neighbor embedding and K-means. *Diagnostics*, 13(11), 1957. <https://doi.org/10.3390/diagnostics13111957>
- Anitha, P., & Patil, M. M. (2022). RFM model for customer purchase behavior using K-Means algorithm. *Journal of King Saud University: Computer and Information Sciences*, 34(5), 1785-1792. <https://doi.org/10.1016/j.jksuci.2019.12.011>

- Candra, A., Purwanto, P., & Farikhin, F. (2023). Comparison of simple matching coefficient and Euclidean distance in K-means algorithm for tourism destination classification. *E3S Web of Conferences*, 448, 02011. <https://doi.org/10.1051/e3sconf/202344802011>
- dos Santos, C., & Bruni, A. T. (2024). Evaluation of density functional theory-generated data for infrared spectroscopy of novel psychoactive substances using unsupervised learning. *Psychoactives*, 3(2), 265-284. <https://doi.org/10.3390/psychoactives3020017>
- Ezugwu, A. E., Ikotun, A. M., Oyelade, O. O., Abualigah, L., Agushaka, J. O., Eke, C. I., & Akinyelu, A. A. (2022). A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Engineering Applications of Artificial Intelligence*, 110, 104743. <https://doi.org/10.1016/j.engappai.2022.104743>
- Gunawan, A., & Artaye, K. (2026). Website-based convolutional neural network for banana freshness detection to prevent health risks. *Jurnal Kecerdasan Buatan dan Pembelajaran Mesin*, 1(1), 1-22. <https://doi.org/10.35912/jkbpm.v1i1.6025>
- Gustina, P., & Wibowo, H. (2026). Fuzzy Mamdani-based book recommendation system for academic library services: Design, implementation, and evaluation. *Jurnal Kecerdasan Buatan dan Pembelajaran Mesin*, 1(1), 63-74. <https://doi.org/10.35912/jkbpm.v1i1.6765>
- Hafidz, M. F., & Lestari, S. (2023). Solution to scalability and sparsity problems in collaborative filtering using K-means clustering and weight point rank (WP-Rank). *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 7(4), 743-750. <https://doi.org/10.29207/resti.v7i4.4543>
- Haryati, A. E., & Surono, S. (2022). Comparative study of distance measures on fuzzy subtractive clustering. *Media Statistika*, 14(2), 137-145. <https://doi.org/10.14710/medstat.14.2.137-145>
- Hui, F., Loahavilai, P., Chakpitak, N., & Chandarasupsang, T. (2025). K means clustering analysis for Chinese tourists' segmentation on duty-free personal luxury goods purchasing intentions in the Asia-Pacific region. *Asia Marketing Journal*, 27(2), 148-170. <https://doi.org/10.53728/2765-6500.1653>
- Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhaija, B., & Heming, J. (2023). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information Sciences*, 622, 178-210. <https://doi.org/10.1016/j.ins.2022.11.139>
- Kenger, O. N., Kenger, Z. D., Ozceylan, E., & Mrugalska, B. (2023). Clustering of cities based on their smart performances: A comparative approach of fuzzy C-means, K-means, and K-medoids. *IEEE Access*, 11, 134446-134459. <https://doi.org/10.1109/ACCESS.2023.3333753>
- Khomarudin, A. N., & Hidayat, A. (2022). K-means clustering algorithm to see the correlation of tahfidz activities with student's learning outcomes. *Knowbase: International Journal of Knowledge in Database*, 2(1), 1-13. <https://doi.org/10.30983/ijokid.v2i1.5672>
- Khonqulova, N. R. (2026). LLM-driven sentiment analysis in customer service dashboards: A framework for real-time feedback intelligence. *Applied AI and Machine Learning Journal*, 2(1), 1-10. <https://doi.org/10.35912/aiml.v2i1.4258>
- Mahnoor, Shafi, I., Chaudhry, M., Montero, E. C., Alvarado, E. S., Diez, I. D. L. T., Samad, M. A., & Ashraf, I. (2024). A review of approaches for rapid data clustering: Challenges, opportunities, and future directions. *IEEE Access*, 12, 138086-138120. <https://doi.org/10.1109/ACCESS.2024.3461798>
- Maryono, D., Budiyanto, C. W., & Pamungkas, A. A. (2022). Implementation of K-means clustering for optimization of student grouping based on index of learning styles in programming classes. *IJIE (Indonesian Journal of Informatics Education)*, 6(2), 84-89. <https://doi.org/10.20961/ijie.v6i2.68151>

- Masoud, M. (2023). A hybrid K-means and particle swarm optimization technique for solving the rechargeable e-scooters problem. *IEEE Access*, 11, 132472-132482. <https://doi.org/10.1109/ACCESS.2023.3336810>
- Muhammad, H. T. U., & Fauzi, C. (2026). Air quality prediction in DKI Jakarta using support vector machine: A comprehensive classification approach. *Jurnal Kecerdasan Buatan dan Pembelajaran Mesin*, 1(1), 49-61. <https://doi.org/10.35912/jkbpm.v1i1.6764>
- Nasor, M., & Obaid, W. (2021). Segmentation of osteosarcoma in MRI images by K-means clustering, Chan-Vese segmentation, and iterative Gaussian filtering. *IET Image Processing*, 15(6), 1310-1318. <https://doi.org/10.1049/ipr2.12106>
- Nisa, S. R., & Ali, R. (2025). Application of fuzzy matching in chatbot development to improve user experience on e-commerce sites (Case study: Cutiw Fashion Store). *Applied AI and Machine Learning Journal*, 1(1), 51-60. <https://doi.org/10.35912/aiml.v1i1.3775>
- Nordahl, C., Boeva, V., Grahn, H., & Persson-Netz, M. (2025). On evaluation of data stream clustering algorithms: A survey. *IEEE Access*, 13, 139524-139546. <https://doi.org/10.1109/ACCESS.2025.3596435>
- Osman, H. M., & Yaba, S. P. (2022). Automated segmentation of acute lymphocytic leukemia (ALL) subtypes by the combination of color space conversion and K-means cluster. *Zanco Journal of Pure and Applied Sciences*, 34(3), 11-20. <https://doi.org/10.21271/ZJPAS.34.3.2>
- Partamies, N., Dol, B., Teissier, V., Juusola, L., Syrjäso, M., & Mulders, H. (2024). Auroral breakup detection in all-sky images by unsupervised learning. *Annales Geophysicae*, 42, 103-115. <https://doi.org/10.5194/angeo-42-103-2024>
- Purdadi, I. G., & Agus, I. (2025). Traffic density prediction using the YOLO algorithm to improve traffic management in Bandar Lampung City. *Applied AI and Machine Learning Journal*, 1(1), 41-49. <https://doi.org/10.35912/aiml.v1i1.3774>
- Raeisi, M., & Sesay, A. B. (2022). A distance metric for uneven clusters of unsupervised K-means clustering algorithm. *IEEE Access*, 10, 86286-86297. <https://doi.org/10.1109/ACCESS.2022.3198992>
- Ridwan, A. F., Napitupulu, H., & Sukono, S. (2022). Decision-making in formation of mean-VaR optimal portfolio by selecting stocks using K-means and average linkage clustering. *Decision Science Letters*, 11(4), 431-442. <https://doi.org/10.5267/j.dsl.2022.7.002>
- Romanuke, V. (2023). Speedup of the k-means algorithm for partitioning large datasets of flat points by a preliminary partition and selecting initial centroids. *Applied Computer Systems*, 28(1), 1-12. <https://doi.org/10.2478/acss-2023-0001>
- Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, 2(3), 160. <https://doi.org/10.1007/s42979-021-00592-x>
- Sathya, R., & Abraham, A. (2013). Comparison of supervised and unsupervised learning algorithms for pattern classification. *International Journal of Advanced Research in Artificial Intelligence*, 2(2), 34-38. <https://doi.org/10.14569/IJARAI.2013.020206>
- Setiawan, G. H., Pranata, M. D. A., Arimbawa, I. B. A., Giri, I. W. P., & Dayani, N. P. L. C. (2025). Topic clustering of student complaints based on semantic meaning using the indobERT and K-means models. *Journal of Applied Informatics and Computing*, 9(4), 1715-1721. <https://doi.org/10.30871/jaic.v9i4.10080>
- Setyawan, D. A., & Purwatiningsih, A. (2024). A comparative analysis between K-means and agglomerative clustering techniques in maritime skill certification. *Compiler*, 13(1), 45-54. <https://doi.org/10.28989/compiler.v13i1.2158>

- Shutaywi, M., & Kachouie, N. N. (2021). Silhouette analysis for performance evaluation in machine learning with applications to clustering. *Entropy*, 23(6), 759. <https://doi.org/10.3390/e23060759>
- Siagian, R., Sirait, P., & Halim, A. (2022). The implementation of K-means dan K-medoids algorithm for customer segmentation on e-commerce data transactions. *Sistemasi: Jurnal Sistem Informasi*, 11(2), 260-270. <https://doi.org/10.32520/stmsi.v11i2.1337>
- Sroyer, A. M., Mandowen, S. A., & Reba, F. (2022). Analisis cluster penyakit malaria Provinsi Papua menggunakan metode single linkage dan K-means (Cluster analysis of malaria in Papua Province using single linkage and K-means methods). *Jurnal Teknologi dan Sistem Informasi*, 7(3), 147-154. <https://doi.org/10.25077/TEKNOSI.v7i3.2021.147-154>
- Sunarti, S., Alawiah, E. T., & Pahlevi, O. (2025). Penggunaan algoritma K-means dalam mengelompokkan jumlah usaha dan hasil pendapatan UMKM DKI Jakarta (Using the K-means algorithm to cluster the number of businesses and revenue of MSMEs in DKI Jakarta). *Techno.Com*, 24(4), 1280-1291. <https://doi.org/10.62411/tc.v24i4.14925>
- Syauqi, R. M., Sabrina, P. N., & Santikarama, I. (2023). K-means clustering with KNN and mean imputation on CPU benchmark compilation data. *Journal of Applied Informatics and Computing*, 7(2), 231-239. <https://doi.org/10.30871/jaic.v7i2.6491>
- Tadi, M., & Ningsi, B. A. (2023). Analisis klaster kemiskinan kabupaten kota di Provinsi Banten menggunakan metode K-means (Cluster analysis of poverty in regencies and cities of Banten Province using the K-means method). *Jurnal Lebesgue*, 4(1), 374-385. <https://doi.org/10.46306/lb.v4i1.224>
- Utami, D. K., Irawati, N., & Sumantri, S. (2023). Analysis of the k-means method in clustering acceptance of PKH aid in Pulau Rakyat Tua Village. *Sistemasi: Jurnal Sistem Informasi*, 12(3), 953-961. <https://doi.org/10.32520/stmsi.v12i3.3236>
- Wani, A. A. (2024). Comprehensive analysis of clustering algorithms: Exploring limitations and innovative solutions. *PeerJ Computer Science*, 10, e2286. <https://doi.org/10.7717/peerj-cs.2286>
- Wardhana, F. P., & Winarno, S. (2024). Analisis pemain terbaik sepak bola dengan menggunakan algoritma K-means (Analysis of the best football players using the K-means algorithm). *Edumatic: Jurnal Pendidikan Informatika*, 8(2), 409-417. <https://doi.org/10.29408/edumatic.v8i2.27105>
- Widyastuti, L. A., & Tarumingkeng, R. C. (2025). The effect of artificial intelligence (AI) and customer experience (CX) use in telemedicine on customer satisfaction moderated by service duration. *Applied AI and Machine Learning Journal*, 1(1), 1-20. <https://doi.org/10.35912/aiml.v1i1.3763>
- Wu, T., Gu, X., Shao, J., Zhou, R., & Li, Z. (2021). Colour image segmentation based on a convex K-means approach. *IET Image Processing*, 15(8), 1596-1606. <https://doi.org/10.1049/ipr.2.12128>
- Xiahou, X., & Harada, Y. (2022). B2C e-commerce customer churn prediction based on K-means and SVM. *Journal of Theoretical and Applied Electronic Commerce Research*, 17(2), 458-475. <https://doi.org/10.3390/jtaer17020024>
- Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2), 165-193. <https://doi.org/10.1007/s40745-015-0040-1>