Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.2, April 2025, pp. 547~555
e-ISSN: 2548-6861

Implementation of K-Means Clustering in Grouping Sales Data at Zura Mart

Miranda [1]*, Sriani [2]*
* Computer Science Study Program, Faculty of Science and Technology, State Islamic University of North Sumatra
mirandaoppo22@gmail.com [1], sriani@uinsu.ac.id [2]

Article Info

Article history:

Received 2025-02-17
Revised 2025-02-20
Accepted 2025-02-25

Keyword:

K-Means Algorithm,
Cluster Analysis,
Data Mining,
Adaptive Marketing Strategy.

ABSTRACT

The efficiency of inventory management and targeted marketing strategies relies on understanding sales patterns and stock levels dynamically. This study proposes a K-Means Clustering-based approach combined with a real-time stock monitoring system to classify products adaptively. The dataset consists of 87 products with variables including total sales, average sales, and remaining stock. The analysis process begins with data normalization to standardize parameter scales, followed by the application of the Elbow Method, which determines the optimal number of clusters as three. The clustering results indicate that Cluster C0 (21 products) has high sales but low stock, Cluster C1 (59 products) has stable sales with moderate stock, and Cluster C2 (7 products) has low sales but abundant stock. These findings not only provide strategic insights for inventory optimization but also serve as the foundation for developing an automated recommendation system that links clustering results with adaptive promotional strategies and restock prediction. Thus, this study contributes to enhancing Zura Mart's business efficiency through the integration of data-driven decision-making in inventory management and marketing.

I. INTRODUCTION

The development of information technology has had a significant impact on the retail industry, including Zura Mart, which struggles to utilize sales data effectively for business decision-making. As a growing retail company, Zura Mart faces challenges in understanding sales patterns, particularly in categorizing products based on sales performance and stock availability[1].

Currently, the sales data management system still relies on conventional methods without in-depth analysis, leading to stock imbalances such as stock out (running out of products during high demand) and overstock (excess inventory due to low sales). This issue results in financial losses and reduces customer satisfaction due to suboptimal product availability[2][3].

Grouping products based on average sales can help Zura Mart identify the most and least potential items. By categorizing products into high, medium, and low sales groups, the company can develop more effective strategies. High-selling products can be prioritized for restocking, while low-selling products can be boosted through promotions. This grouping process also helps the company better understand consumer preferences and market trends[4].

One technique that can be used for product grouping is the K-Means Clustering method, a popular clustering algorithm in data analysis[5]. With this algorithm, products can be automatically categorized into best-selling products, moderately selling products, and low-selling products based on their sales patterns. K-Means is chosen for its efficiency in handling medium-sized datasets[6], its ability to provide clear and structured clustering results, and its simplicity in implementation compared to other methods such as DBSCAN, which is more suitable for datasets with varying densities, or Hierarchical Clustering, which has higher computational complexity. This clustering approach provides Zura Mart with a more structured insight into the performance of each product and helps the company make better decisions regarding inventory management and marketing strategies[7].

K-Means Clustering is a better method for this dataset because the data consists of numerical variables suitable for grouping based on averages (centroids), the product distribution pattern can be clearly analyzed using a predetermined number of clusters, and it is faster and more efficient for handling small to medium-sized datasets[8]. Meanwhile, DBSCAN is not suitable since this dataset does not exhibit varying density patterns and does not contain significant noise or outliers. Therefore, K-Means with three clusters is recommended to categorize products into high, moderate, and low sales groups based on Total Sales, Average Sales, and Remaining Stock[9].

The objective of this study is to classify sales data at Zura Mart using the K-Means Clustering method to optimize inventory management. By categorizing products based on product category, average sales, and remaining stock, this research aims to identify sales patterns that can assist the company in making strategic decisions regarding stock management. Through clustering results, Zura Mart is expected to minimize the risk of overstocking or stock shortages, thereby improving operational efficiency and customer satisfaction.

Machine learning is a branch of Artificial Intelligence (AI) that is also defined as the science that enables computers to behave like humans, where computers can make a system intelligent by examining the available data, especially if the program is not explicitly defined. Machine learning aims to discover and apply patterns found in data using statistical techniques to identify those patterns. This data processing is not limited to text but can also include images, sound, video, or user actions while using the internet. Machine learning has several learning techniques, including supervised learning, unsupervised learning, and reinforcement learning[10].

Clustering is a process considered an important approach to finding similarities in data and grouping similar data into clusters. Clustering is regarded as one of the most important unsupervised learning methods, where the problem involves discovering patterns in an unlabeled dataset. Clustering divides a dataset into several groups, where the similarity within a specific group is greater than the similarity between groups[11].

K-means is a clustering algorithm that works based on the principle of partitioned clustering. The working principle of partitioned clustering involves grouping items randomly, influenced by centroids. In each iteration of partitioned clustering, more than one item may be selected to be merged. On the other hand, the working principle of hierarchical clustering is done progressively. In each iteration of hierarchical clustering, only one item will be selected to merge with another item[12][13].

Various studies have examined the application of K-Means Clustering in sales analysis. Nugraha et al. (2022) found that this method can identify 99 best-selling items and 23 unsold items, helping with restocking strategies[12]. Sallaby et al. (2022) applied K-Means at Toko Widya Bengkulu using Visual Basic .Net and SQL Server 2008, resulting in three clusters: very popular, moderately popular, and less popular items[14]. Chaerunisa et al. (2021) applied K-Means to honey sales at Toko Teras Palawi, finding 6 optimal clusters with a DBI value of -0.365, indicating good proximity between cluster members[15]. Triyandana et al.'s study applied K-Means clustering to sales data from Dpom Coffee, identifying three clusters with a DBI value of -0.457. Cluster 1 contained 8 low-selling items, offering insights for Dpom Coffee to evaluate and optimize sales strategies and stock management. This method helps businesses identify popular and less popular menu items based on sales and pricing[16]. Aria et al. (2023) analyzed pharmaceutical data and found the accuracy of K-Means based on the Davies Bouldin Index to be 0.513. These findings show that K-Means is effective in product clustering and stock management strategies across various retail sectors[17].

II. METHOD

A research framework is a conceptual structure that plans and organizes the research, helping researchers understand the relationships between variables and providing guidance in data design and analysis. A clear framework maintains consistency, ensures relevance between theory and data, and simplifies the interpretation of results. Additionally, the research framework helps identify gaps, formulate hypotheses, and supports the design of data collection instruments and appropriate methodologies. With a well-structured framework, research effectiveness and efficiency are enhanced.

The stages of the research framework are as follows:

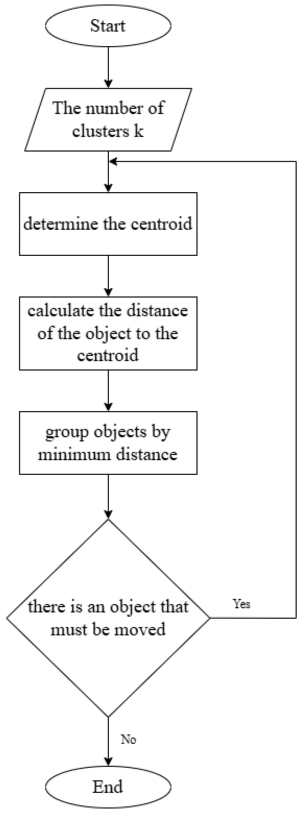


Figure 1 Research Framework

A. Data Representation

This study uses data obtained from Zura Mart's sales records, collected from October 2024 to November 2024. The dataset consists of 87 entries, including product name, category, total sales, average sales, and remaining stock. The data is analyzed to identify sales patterns and group products with similar characteristics using the K-Means Clustering method. This approach aims to gain deeper insights into sales dynamics, as well as provide strategic guidance for stock management, marketing strategies, and decision-making. Additionally, the clustering results can be used to optimize product offerings based on customer needs and improve operational efficiency.

TABLE 1
DATA REPRESENTATION

No	Product Name	Category	Total Sales	Average Sales	Remaining Stock
1	5 Kg Rice	Staple Food	481	40	168
2	1 Kg Sugar	Staple Food	650	54	158
3	1 L Cooking Oil	Staple Food	310	26	166
4	250 gr Butter	Staple Food	223	19	89
5	1 Kg Glutinous Rice Flour	Staple Food	207	17	66
6	1 Dozen Chicken Eggs (30)	Staple Food	802	67	189
7	1 L UHT Milk	Supporting Item	566	47	169
8	1 Kg Shallots	Staple Food	443	37	187
9	1 Kg Salt	Staple Food	628	52	182
10	1 Kg Red Rice	Staple Food	177	15	79
11	2 L Cooking Oil	Staple Food	230	19	74
12	500 gr Butter	Staple Food	314	26	126
13	500 gr Fried Crackers	Supporting Item	309	26	155
14	1 Dozen Duck Eggs	Staple Food	267	22	95
15	397 gr Sweetened Condensed Milk	Supporting Item	335	28	197
16	1 Kg Garlic	Staple Food	324	27	146
17	5 Kg Sugar	Staple Food	569	47	125
18	5 L Cooking Oil	Staple Food	438	37	140
19	250 gr Beef Floss	Complementary Item	424	35	73
20	250 gr Sausage	Supporting Item	381	32	80
21	200 gr Cheese	Supporting Item	402	34	74
22	500 gr Mayonnaise	Complementary Item	375	31	86
23	500 ml Sweet Soy Sauce	Complementary Item	439	37	79
24	500 ml Tomato Sauce	Seasoning	394	33	95
25	500 ml Vinegar	Complementary Item	416	35	104

B. Data Normalization

The initial step in the analysis is data normalization, aimed at standardizing the scale of each parameter to ensure equal weight for all variables in the clustering process. Normalization is performed using the Min-Max Scaling method, transforming each parameter's value into the range [0, 1]. This process is crucial to avoid bias, as differences in scale among parameters could disproportionately affect the clustering results. Normalization ensures balanced contribution from each parameter, leading to more consistent, accurate, and representative clustering outcomes. Additionally, it enhances the efficiency of the K-Means algorithm, enabling better handling of data, especially in scenarios with diverse attributes.

TABLE 2
DATA NORMALIZATION

No	Product Name	Total Sales	Average Sales	Remaining Stock
1	5 Kg Rice	0,4864	0,48077	0,78832
2	1 Kg Sugar	0,7568	0,75	0,71533
3	1 L Cooking Oil	0,2128	0,21154	0,77372
4	250 gr Butter	0,0736	0,07692	0,21168
5	1 Kg Glutinous Rice Flour	0,048	0,03846	0,0438
6	1 Dozen Chicken Eggs (30)	1	1	0,94161
7	1 L UHT Milk	0,6224	0,61538	0,79562
8	1 Kg Shallots	0,4256	0,42308	0,92701
9	1 Kg Salt	0,7216	0,71154	0,89051
10	1 Kg Red Rice	0	0	0,13869
11	2 L Cooking Oil	0,0848	0,07692	0,10219
12	500 gr Butter	0,2192	0,21154	0,48175
13	500 gr Fried Crackers	0,2112	0,21154	0,69343
14	1 Dozen Duck Eggs	0,144	0,13462	0,25547
15	397 gr Sweetened Condensed Milk	0,2528	0,25	1
16	1 Kg Garlic	0,2352	0,23077	0,62774
17	5 Kg Sugar	0,6272	0,61538	0,47445
18	5 L Cooking Oil	0,4176	0,42308	0,58394
19	250 gr Beef Floss	0,3952	0,38462	0,09489
20	250 gr Sausage	0,3264	0,32692	0,14599
21	200 gr Cheese	0,36	0,36538	0,10219
22	500 gr Mayonnaise	0,3168	0,30769	0,18978

23	500 ml Sweet Soy Sauce	0,4192	0,42308	0,13869
24	500 ml Tomato Sauce	0,3472	0,34615	0,25547
25	500 ml Vinegar	0,3824	0,38462	0,32117

C. The number of clusters k

This study applies the K-Means Clustering algorithm to group sales items based on specific characteristics. One of the main challenges in applying this algorithm is determining the optimal number of clusters. To address this, the study uses the Elbow method, which helps identify the point where adding more clusters no longer significantly improves the total variation explained by the model. The Elbow method is a common approach to determine the best number of clusters, aiming to divide the data optimally without creating too many irrelevant or redundant clusters.

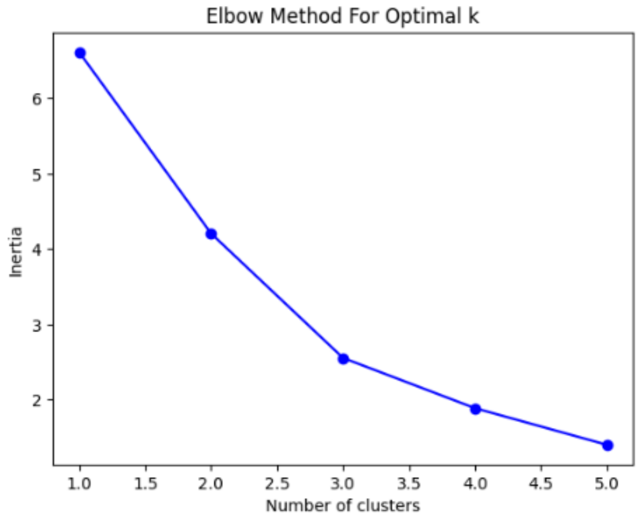


Figure 2 Elbow Method Graph

Based on the Elbow method graph above, it is clear that after k = 3, the reduction in inertia begins to slow down, indicating that the elbow point at k = 3 is the optimal number of clusters. However, if the number of clusters is increased to 4 or 5, inertia continues to decrease, but not significantly. Increasing the number of clusters to 4 or 5 may provide more specific segmentation, but it is important to assess whether this division still has clear business relevance. If the differences between clusters become too small or irrelevant from a business perspective, then choosing k = 3 remains preferable for balancing the variation within the clusters and maintaining ease of interpretation for stock management and marketing strategies.

III. Results And Discussion

A. Determine The Centroid

Initializing the initial centroids randomly serves as the starting point for the clustering process. In this case, the initial centroids are selected from the normalized data, with C0 taken from the 5th data point, C1 from the 3rd data point, and C2 from the 1st data point. This initial centroid selection acts as a reference point for the K-Means algorithm to begin grouping the data into clusters. The details can be seen in the following table.

TABLE 3
INITIAL CENTROID

Centroid Awal	Total Penjualan	Rata-Rata Penjualan	Sisa Stok
Centroid 0	0,048	0,03846	0,0438
Centroid 1	0,2128	0,21154	0,77372
Centroid 2	0,4864	0,48077	0,78832

B. Calculate The Distance Of The Object To The Centroid

After setting the initial centroids, the next step is to calculate the distance between each centroid and the data using the Euclidean Distance formula, as shown below.

$$D(ij) = \sqrt{(x_{1i} - x_{1j})^2 + (x_{2i} - x_{2j})^2 + \cdots + (x_{ki} - x_{kj})^2}$$

Calculating the distance from Data Point 1 to Centroid 0 with attribute values (0.048; 0.03846; 0.0438):

$$C0 = \sqrt{\begin{array}{c}(0{,}048 - 0{,}4864)^2 + (0{,}03846 - 0{,}48077)^2 \\ +(0{,}0438 - 0{,}78832)^2\end{array}}$$

$$C0 = 0{,}970640369$$

Calculating the distance from Data Point 1 to Centroid 1 with attribute values (0.2128; 0.21154; 0.77372):

$$C1 = \sqrt{\begin{array}{c}(0{,}2128 - 0{,}4864)^2 + (0{,}21154 - 0{,}48077)^2 \\ +(0{,}77372 - 0{,}78832)^2\end{array}}$$

$$C1 = 0{,}384128771$$

Calculating the distance from Data Point 1 to Centroid 2 with attribute values (0.4864; 0.48077; 0.78832):

$$C2 = \sqrt{\begin{array}{c}(0{,}4864 - 0{,}4864)^2 + (0{,}48077 - 0{,}48077)^2 \\ +(0{,}78832 - 0{,}78832)^2\end{array}}$$

$$C2 = 0$$

Calculating the distance from Data Point 3 to Centroid 0 with attribute values (0.048; 0.03846; 0.0438):

$$C0 = \sqrt{\begin{array}{c}(0{,}048 - 0{,}2128)^2 + (0{,}03846 - 0{,}21154)^2 \\ +(0{,}0438 - 0{,}77372)^2\end{array}}$$

$$C0 = 0{,}768048783$$

Calculating the distance from Data Point 3 to Centroid 1 with attribute values (0.2128; 0.21154;

$$C1 = \sqrt{\begin{array}{c}(0{,}2128 - 0{,}2128)^2 + (0{,}21154 - 0{,}21154)^2 \\ +(0{,}77372 - 0{,}77372)^2\end{array}}$$

$C1 = 0$

Calculating the distance from Data Point 3 to Centroid 2 with attribute values (0.4864; 0.48077; 0.78832):

$$C2 = \sqrt{\begin{array}{c}(0{,}4864 - 0{,}2128)^2 + (0{,}48077 - 0{,}21154)^2 \\ +(0{,}78832 - 0{,}77372)^2\end{array}}$$

$C2 = 0{,}384128771$

Calculating the distance from Data Point 5 to Centroid 0 with attribute values (0.048; 0.03846; 0.0438):

$$C0 = \sqrt{\begin{array}{c}(0{,}048 - 0{,}048)^2 + (0{,}03846 - 0{,}03846)^2 \\ +(0{,}0438 - 0{,}0438)^2\end{array}}$$

$C0 = 0$

Calculating the distance from Data Point 5 to Centroid 1 with attribute values (0.2128; 0.21154; 0.77372):

$$C1 = \sqrt{\begin{array}{c}(0{,}2128 - 0{,}048)^2 + (0{,}21154 - 0{,}03846)^2 \\ +(0{,}77372 - 0{,}0438)^2\end{array}}$$

$C1 = 0{,}768048783$

Calculating the distance from Data Point 5 to Centroid 2 with attribute values (0.4864; 0.48077; 0.78832):

$$C2 = \sqrt{\begin{array}{c}(0{,}4864 - 0{,}048)^2 + (0{,}48077 - 0{,}03846)^2 \\ +(0{,}78832 - 0{,}0438)^2\end{array}}$$

$C2 = 0{,}970640369$

1. Iterasi 1

Based on the nearest centroid distance calculations performed above, the results for Iteration 1 have been obtained. The details can be seen in the following table.

TABLE 4
ITERATION RESULT 1

Data	Centroid Distance Iteration 1			Cluster
	C0	C1	C2	
1	0,970640369	0,384128771	0	2
2	1,208155268	0,767648724	0,388495165	2
3	0,768048783	0	0,384128771	1
4	0,174121297	0,594464588	0,816095798	0
5	0	0,768048783	0,970640369	0
6	1,623861437	1,126738685	0,746245655	2
7	1,108154032	0,575620114	0,191491885	2
8	1,03468556	0,336943372	0,162048425	2
9	1,274241528	0,722853612	0,344987868	2
10	0,113080872	0,70235042	0,943262365	0
11	0,079011415	0,69674894	0,891712823	0
12	0,501065194	0,292040136	0,487714873	1
13	0,691816495	0,080305941	0,396514748	1
14	0,251529192	0,528425197	0,721794088	0
15	1,000505198	0,23298457	0,39068168	1
16	0,642660797	0,148935265	0,389085307	1
17	0,923996076	0,651441969	0,369403396	2
18	0,759136301	0,350297959	0,223232526	2
19	0,492934857	0,723903733	0,705979764	0
20	0,413713098	0,648276528	0,679601097	0
21	0,455664436	0,704476491	0,707153582	0
22	0,40749036	0,600871655	0,6457329	0
23	0,542886817	0,700437886	0,655639469	0
24	0,478537318	0,552056405	0,566946476	0
25	0,555503576	0,513344279	0,488149511	2

After calculating the Euclidean Distance, data points are assigned to clusters based on their minimum distance to the nearest centroid. The clustering results from Iteration 1 are as follows:

Previous Cluster: (0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0)

New Cluster: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 2)

Since changes occurred in the clustering, the process continues to the next iteration. In this step, centroids are updated based on the average values of the newly formed clusters. Iterations will proceed until stability is reached, meaning no further changes in cluster assignments. This ensures accurate and optimal clustering results.

2. Iterasi 2

Based on the nearest centroid distance calculations performed above, the results for Iteration 2 have been obtained. They can be seen in the following table.

TABLE 5
ITERATION RESULT 2

Data	Centroid Distance Iteration 2			Cluster
	C0	C1	C2	
1	0,698737831	0,383059594	0,263195961	2
2	0,885392437	0,684760373	0,463485829	2
3	0,616350199	0,274127048	0,420129783	1
4	0,27259434	0,433129791	0,621698989	0
5	0,334776928	0,591102768	0,756054886	0

6	1,299712648	1,089675687	0,876364907	2
7	0,807775616	0,540980708	0,355034054	2
8	0,79774063	0,448023529	0,400522732	2
9	0,969149861	0,708254793	0,519105328	2
10	0,375111985	0,560229695	0,750510566	0
11	0,266799311	0,512651814	0,677855692	0
12	0,327412253	0,111561507	0,339955921	1
13	0,536898123	0,203519451	0,37996309	1
14	0,200576963	0,340312651	0,521528346	0
15	0,838340476	0,480942829	0,551231199	1
16	0,468025457	0,132054616	0,327162826	1
17	0,593300023	0,47250216	0,243557518	2
18	0,476048506	0,196015825	0,07269738	2
19	0,189330645	0,450387171	0,442965398	0
20	0,089087051	0,379780469	0,422552295	0
21	0,151012692	0,432614147	0,44534955	0
22	0,072835441	0,333875832	0,393322286	0
23	0,222492573	0,426513776	0,392610949	0
24	0,148988403	0,278764386	0,312098642	0
25	0,231554354	0,241333351	0,229833638	2

After completing the Euclidean Distance calculations, the data is grouped into clusters based on the minimum distance to the nearest centroid. The clustering results for Iteration 2 are as follows:

Previous cluster: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 2)

New cluster: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 2)

Since there is no change in the clustering results, the process has reached stability. At this stage, the clustering process is considered complete, ensuring accurate and optimal results.

3. *Iterasi 3*

Based on the nearest centroid distance calculations performed above, the results for Iteration 3 have been obtained. They can be seen in the following table.

TABLE 6
ITERATION RESULT 3

Data	Centroid Distance Iteration 3			Cluster
	C0	C1	C2	
1	0,680344702	0,377957377	0,226305187	2
2	0,862104644	0,674350932	0,422385333	2
3	0,609651378	0,284017905	0,42761285	1
4	0,292704571	0,43824192	0,665276505	0
5	0,358820285	0,594329826	0,801116494	0
6	1,276231803	1,08024611	0,833706155	2
7	0,786500989	0,533000603	0,309556643	2
8	0,782285933	0,447638757	0,371637032	2
9	0,947309532	0,700434906	0,473724798	2
10	0,397737832	0,565860191	0,794207522	0
11	0,290461312	0,515677911	0,722821815	0
12	0,323334581	0,121641605	0,375947659	1
13	0,530733057	0,214773911	0,395225755	1
14	0,216252052	0,344197471	0,565321558	0
15	0,829280373	0,48766299	0,539949254	1
16	0,460890828	0,143866312	0,348471969	1
17	0,569447532	0,459608728	0,224854384	2
18	0,457391154	0,186077024	0,080496788	2
19	0,173119829	0,441590691	0,482733362	0
20	0,070547082	0,373025572	0,466237288	0
21	0,135888161	0,424666954	0,486839507	0
22	0,048839375	0,327650862	0,437964786	0
23	0,20202198	0,41644371	0,430480815	0
24	0,125848105	0,270308061	0,35653504	0
25	0,208976834	0,230194277	0,273725499	0

After completing the Euclidean Distance calculations, the data is grouped into clusters based on the minimum distance to the nearest centroid.

Iteration 3 results:

Previous clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 2)

New clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 0)

Since changes in clustering still occur, the process continues to the next iteration. At this stage, the centroid positions are updated based on the average of the data within each newly formed cluster. Iterations proceed until the clustering stabilizes, ensuring an accurate and optimal clustering result.

4. *Iterasi 4*

After completing the Euclidean Distance calculations, the data is grouped into clusters based on the minimum distance to the nearest centroid.

Iteration 4 results:

Previous clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 0)

New clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 0)

Since there are still changes in the clustering, the process will continue to the next iteration. At this stage, the centroid positions will be updated based on the average data within each newly formed cluster. The iteration will continue until the clustering stabilizes, ensuring an accurate and optimal clustering result.

5. Iterasi 5

After completing the Euclidean Distance calculations, the data is grouped into clusters based on the minimum distance to the nearest centroid.
Iteration 5 results:
Previous clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 2 0 0 0 0 0 0 0)
New clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 1 0 0 0 0 0 0 0)
Since there are changes in the clustering, the process will continue to the next iteration. At this stage, the centroid positions will be updated based on the average data within each newly formed cluster. The iteration will continue until the clustering stabilizes, ensuring an accurate and optimal clustering result.

6. Iterasi 6

After completing the Euclidean Distance calculations, the data is grouped into clusters based on the minimum distance to the nearest centroid.
Iteration 5 results:
Previous clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 1 0 0 0 0 0 0 0)
New clusters: (2 2 1 0 0 2 2 2 2 0 0 1 1 0 1 1 2 1 0 0 0 0 0 0 0)
No changes occurred in the data grouping, so the data processing is stopped. The final result shows that cluster 0 contains 21 samples, cluster 1 includes 59 samples, and cluster 2 contains 7 samples, as shown in the table below.

TABLE 7
CLUSTER RESULT

Number of Clusters		
C0	C1	C2
21	59	7

D. Implementation of the K-Means Clustering Algorithm in Python.

1. Import Data

At this stage, the imported data consists of sales data from Zura Mart, comprising 87 entries with five main attributes: Product Name, Product Category, Total Sales, Average Sales, and Remaining Stock. This data contains essential information for further analysis. It will undergo preprocessing steps such as cleaning, normalization, and preparation to ensure it is ready for processing using the K-Means Clustering method, which will be implemented in Python.

```
#Import Data
dataInput = pd.read_csv('data_penjualan.csv')
print("Menampilkan Data:")
pd.set_option('display.max_rows', 10)
print(dataInput)
```

```
Menampilkan Data:
          Nama Barang          Kategori  Total Penjualan  \
0          Beras 5 Kg       Bahan Pokok              481
1      Gula Pasir 1 Kg      Bahan Pokok              650
2    Minyak Goreng 1 L      Bahan Pokok              310
3       Mentega 250 gr      Bahan Pokok              223
4     Tepung Ketan 1 Kg     Bahan Pokok              207
..                 ...              ...              ...
82     Mie Instan 85 gr  Bahan Pendamping            384
83  Kecap Manis 500 ml  Bahan Pelengkap              411
84        Sirup 600 ml  Bahan Pendamping             329
85   Roti Tawar 400 gr  Bahan Pendamping             342
86        Pasta 500 gr  Bahan Pendamping             351

    Rata-Rata Penjualan  Sisa Stok
0                    40        168
1                    54        158
2                    26        166
3                    19         89
4                    17         66
..                  ...        ...
82                   32        133
83                   34        124
84                   27        116
85                   29        132
86                   29        114

[87 rows x 5 columns]
```

Figure 3 Import Data

2. Data Normalization

```
#Normalisasi Data
from sklearn.preprocessing import MinMaxScaler
cols = dataInput.columns
ms = MinMaxScaler()

X = ms.fit_transform(dataInput)
X = pd.DataFrame(X, columns=[cols])
df_rounded = X.round(5)
df_rounded.to_csv('normalisasi.csv', index=False)
print(df_rounded)
```

```
   Total Penjualan Rata-Rata Penjualan Sisa Stok
0           0.4864             0.48077   0.78832
1           0.7568             0.75000   0.71533
2           0.2128             0.21154   0.77372
3           0.0736             0.07692   0.21168
4           0.0480             0.03846   0.04380
..             ...                 ...       ...
82          0.3312             0.32692   0.53285
83          0.3744             0.36538   0.46715
84          0.2432             0.23077   0.40876
85          0.2640             0.26923   0.52555
86          0.2784             0.26923   0.39416

[87 rows x 3 columns]
```

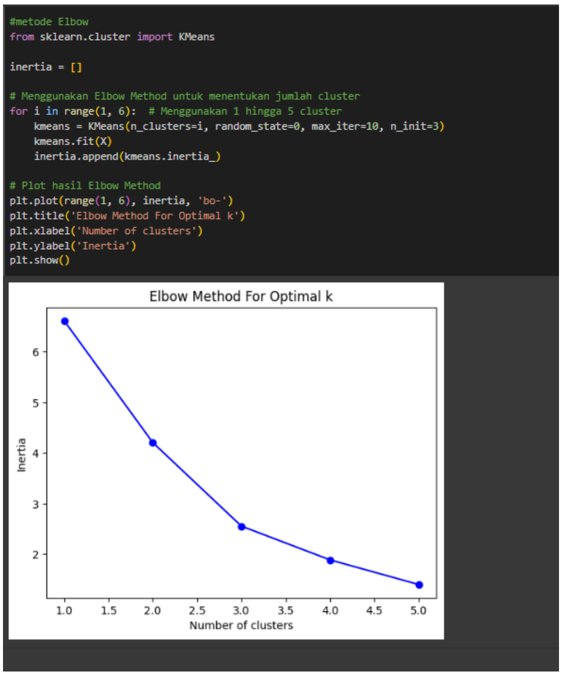
Figure 4 Data Normalization

The next step is data normalization, which aims to align attribute values within a consistent scale. This process enhances analytical efficiency, ensures accurate clustering results, and facilitates model interpretation. Using Min-Max

Scaling, data values are transformed into a range of [0,1], ensuring equal contribution from each attribute. The normalized data, prepared for the clustering process, is shown in the following figure 4.

3. *Elbow Method*

The optimal number of clusters is determined using the Elbow Method with a random state of 0. This algorithm identifies the point where adding more clusters no longer significantly improves variance explanation. It ensures efficient clustering while preventing overfitting. The following image visualizes the Elbow Method used to determine the optimal number of clusters.

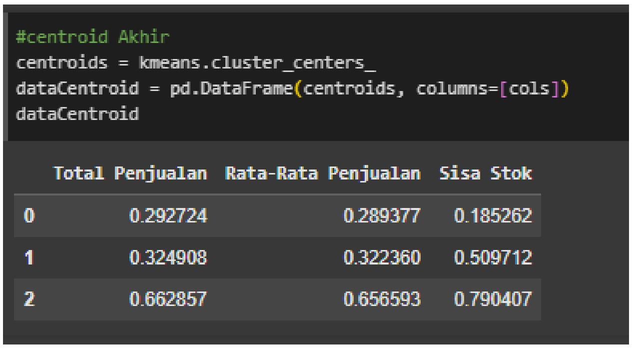


```
#metode Elbow
from sklearn.cluster import KMeans

inertia = []

# Menggunakan Elbow Method untuk menentukan jumlah cluster
for i in range(1, 6):  # Menggunakan 1 hingga 5 cluster
    kmeans = KMeans(n_clusters=i, random_state=0, max_iter=10, n_init=3)
    kmeans.fit(X)
    inertia.append(kmeans.inertia_)

# Plot hasil Elbow Method
plt.plot(range(1, 6), inertia, 'bo-')
plt.title('Elbow Method For Optimal k')
plt.xlabel('Number of clusters')
plt.ylabel('Inertia')
plt.show()
```


Figure 5 Elbow Method

4. *Final Centroids*

A centroid is the central point of each cluster, calculated as the average position of all data points within the cluster. It represents the cluster's center and serves as a reference for further data grouping. The centroid is determined by averaging the attribute values of all data points in the cluster, making it a general representation of the cluster's characteristics.



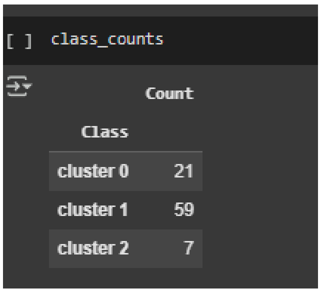
```
#centroid Akhir
centroids = kmeans.cluster_centers_
dataCentroid = pd.DataFrame(centroids, columns=[cols])
dataCentroid
```

	Total Penjualan	Rata-Rata Penjualan	Sisa Stok
0	0.292724	0.289377	0.185262
1	0.324908	0.322360	0.509712
2	0.662857	0.656593	0.790407

Figure 6 Final Centroids

5. *Cluster Result*

The image below presents the final clustering results, illustrating the optimal number of clusters determined. This visualization provides insights into the distribution of data within each cluster, helping to understand the patterns and characteristics of each group. These results can be further utilized for stock management and marketing strategy analysis.



```
class_counts
```

Class	Count
cluster 0	21
cluster 1	59
cluster 2	7

Figure 7 Cluster Result

6. *Scatter Plot*

This scatter plot illustrates each data point that has been clustered by the K-Means algorithm, with different colors representing each cluster. Additionally, the centroid of each cluster is shown as a red dot on the plot, representing the average position of the data within that cluster.

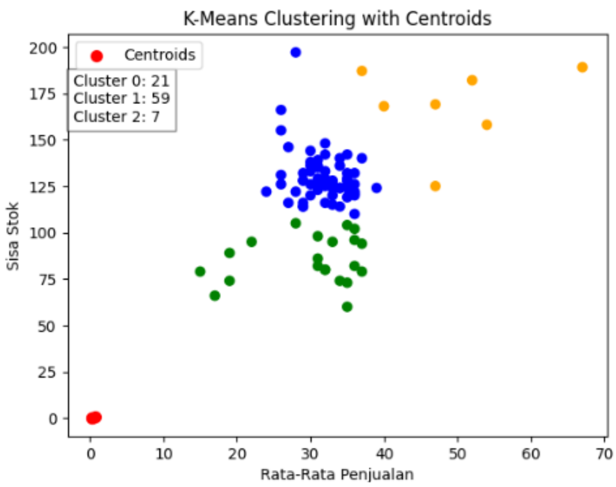


Figure 8 Scatter Plot

The graph above shows the product grouping based on sales patterns and stock using K-Means clustering. The analysis reveals three main clusters. Cluster C0 contains 21 products with high sales but low stock, indicating high demand and the need for proper restocking strategies. Cluster C1 includes 59 products with stable sales and sufficient stock, reflecting a balanced demand-supply relationship. Cluster C2 consists of 7 products with low sales but ample stock, suggesting the need for more aggressive marketing strategies or a reevaluation of the products.

7. DBI (Davies-Bouldin Index)

The image below represents the Davies-Bouldin Index (DBI) values obtained in this study, indicating that the quality of product clustering has been evaluated using DBI as a validation metric. DBI measures the extent to which the formed clusters exhibit good internal density (intra-cluster similarity) and an optimal level of separation from other clusters (inter-cluster difference).

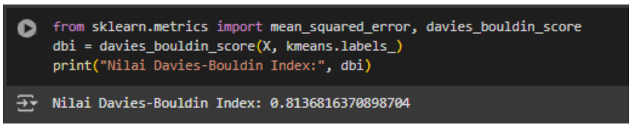


Figure 9 Davies-Bouldin Index (DBI) value.

The Davies-Bouldin Index (DBI) obtained in this study is 0.8137. This value indicates that the clustering quality is fairly good, with a balance between intra-cluster similarity and inter-cluster difference. A lower DBI value signifies better clustering quality, as it shows that each cluster has high internal uniformity and is well-separated from other clusters.

IV. CONCLUSION

The K-Means clustering algorithm effectively grouped sales data into three distinct clusters based on characteristics like sales and stock levels. C0 consists of 21 products with high sales but low stock, highlighting the need for better stock management; C1 contains 59 products with stable sales and moderate stock, reflecting a balanced demand and supply; and C2 comprises 7 products with low sales but high stock, indicating a need for improved marketing strategies. These insights help in making strategic decisions for stock planning, promotions, and inventory management.

REFERENCES

[1] R. Riadi and Mesran, "Penerapan Data Mining Menggunakan Algoritma K-Means Untuk Analisa Penjualan Parfume," *J. Informatics, Electr. Electron. Eng.*, vol. 2, no. 4, pp. 138–145, 2023, doi: 10.47065/jieee.v2i4.1181.

[2] L. Latifah, M. Yusup, and Supiyandi, "Desain Dan Pembangunan Aplikasi Penjualan Buku Berbasis Website Di Toko Buku Murah Medan," vol. 9, no. 2, pp. 105–113, 2024.

[3] Asnita, A. Khalid, and A. Salam, "Penerapan Praktik Akuntansi Manajemen Pada UMKM di Kota Makassar," vol. 7, no. 3, pp. 332–343, 2024.

[4] M. R. Sulistio, N. Suarna, and O. Nurdiawan, "Analisa Penerapan Metode Clustering X-Means Dalam Pengelompokan Penjualan Barang," *J. Teknol. Ilmu Komput.*, vol. 1, no. 2, pp. 37–42, 2023, doi: 10.56854/jtik.v1i2.49.

[5] W. W. Kristianto, "Penerapan Data Mining Pada Penjualan Produk Menggunakan Metode K-Means Clustering (Studi Kasus Toko Sepatu Kakikaki)," *J. Pendidik. Teknol. Inf.*, vol. 5, no. 2, pp. 90–98, 2022, doi: 10.37792/jukanti.v5i2.547.

[6] R. Mubarok, A. A. Syahputra, A. T. Permana, L. Sholiah, and Tarwoto, "Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi) Implementasi Data Mining untuk Clustering Lowongan," vol. 9, no. June, pp. 703–712, 2025.

[7] R. Adha, N. Nurhaliza, U. Sholeha, and M. Mustakim, "Perbandingan Algoritma DBSCAN dan K-Means Clustering untuk Pengelompokan Kasus Covid-19 di Dunia," *SITEKIN J. Sains, Teknol. dan Ind.*, vol. 18, no. 2, pp. 206–211, 2021, [Online]. Available: https://ejournal.uin-suska.ac.id/index.php/sitekin/article/view/12469

[8] H. Darmawan and N. Yudistira, "Penggunaan Model Tabular Retrieval Model (TabR) dengan K-Means Clustering untuk Efisiensi Klasifikasi dan Regresi Data Tabular," vol. 1, no. 1, pp. 1–9, 2022.

[9] S. Mutiah, Y. Hasnataeni, A. Fitrianto, and L. M. R. D. Jumansyah, "Perbandingan Metode Klastering K-Means dan DBSCAN dalam Identifikasi Kelompok Rumah Tangga Berdasarkan Fasilitas Sosial Ekonomi di Jawa Barat vol. 09, no. September, pp. 247–260, 2024.

[10] M. H. Rifai, D. A. Pramudya, and R. R. Narfandi, "Analisis peran teknologi kecerdasan buatan dalam mengoptimalkan proses deteksi terhadap serangan siber," pp. 495–502, 2024.

[11] A. P. Adistya, N. Lutfiyani, P. Tara, Rifaldi, R. Adriyan, and P. Rosyani, "Klasterisasi Menggunakan Algoritma K-Means Clustering Untuk Memprediksi Kelulusan Mata Kuliah Mahasiswa," *OKTAL J. Ilmu Komput. dan Sci.*, vol. 2, no. 8, pp. 2301–2306, 2023.

[12] M. R. Nugroho, I. E. Hendrawan, and P. P. Purwantoro, "Penerapan Algoritma K-Means Untuk Klasterisasi Data Obat Pada Rumah Sakit ASRI," *Nuansa Inform.*, vol. 16, no. 1, pp. 125–133, 2022, doi: 10.25134/nuansa.v16i1.5294.

[13] S. N. Br Sembiring, H. Winata, and S. Kusnasari, "Pengelompokan Prestasi Siswa Menggunakan Algoritma K-Means," *J. Sist. Inf. Triguna Dharma (JURSI TGD)*, vol. 1, no. 1, p. 31, 2022, doi: 10.53513/jursi.v1i1.4784.

[14] A. Fikri Sallaby, R. Tri Alinse, V. Novita Sari, and T. Ramadani, "Pengelompokan Barang Menggunakan Metode K-Means Clustering Berdasarkan Hasil PenjualanDi Toko Widya Bengkulu Pengelompokan Barang Menggunakan Metode K-Means Clustering Berdasarkan Hasil Penjualan Di Toko Widya Bengkulu," *J. Media Infotama*, vol. 18, no. 1, p. 2022, 2022.

[15] Danya Rizki Chaerunisa, Nining Rahaningsih, Fadhil Muhammad Basysyar, Ade Irma Purnamasari, and Nana Suarna, "Pengelompokan Penjualan Madu Menggunakan Algoritma K-Means," *KOPERTIP J. Ilm. Manaj. Inform. dan Komput.*, vol. 5, no. 1, pp. 23–28, 2021, doi: 10.32485/kopertip.v5i1.144.

[16] G. Triyandana, L. A. Putri, and Y. Umaidah, "Penerapan Data Mining Pengelompokan Menu Makanan dan Minuman Berdasarkan Tingkat Penjualan Menggunakan Metode K-Means," *J. Appl. Informatics Comput.*, vol. 6, no. 1, pp. 40–46, 2022, doi: 10.30871/jaic.v6i1.3824.

[17] T. Asy Aria, M. Julkarnain, and F. Hamdani, "KLIK: Kajian Ilmiah Informatika dan Komputer Penerapan Algoritma K-Means Clustering Untuk Data Obat," *Media Online*, vol. 4, no. 1, pp. 649–657, 2023, doi: 10.30865/klik.v4i1.1117.