HVLV-Motor-KC: Production Efficiency of HVLV Motor Classification using K-means Clustering

YEJI DO[1,2], CHAEGYU LEE[1,*], JONGPIL JEONG[1,*], JIHO JEONG[1], DONGGEUN BAE[1], INKWON YEO[1], MINGYU KIM[1]
[1]Department of Smart Factory Convergence,
Sungkyunkwan University,
2066 Seobu-ro Jangan-gu, Suwon, 16419,
REPUBLIC OF KOREA

[2]AI Research Lab,
Hygino,
25 Simin-daero 248beon-gil, Dongan-gu, Anyang-si, Gyeonggi-do,
REPUBLIC OF KOREA

Corresponding Authors

Abstract: - This paper aims to introduce the K-means clustering algorithm to complement the Group Technology (GT) methodology as part of a multi-product, low-volume production system. This challenge aims to overcome the limitations of the GT methodology and optimize the production schedule to increase efficiency. We propose a high-variation, low-volume K-means clustering (HVLV-Motor-KC) algorithm, which is a K-means clustering algorithm that focuses on high-variety, low-volume data. This algorithm helps to optimize production by placing motors with similar characteristics in the same cluster.

Key-Words: - Group Technology, HVLV Production, K-means Clustering, Hierarchical Clustering, Silhouette Score.

Received: December 19, 2023. Revised: August 19, 2024. Accepted: September 23, 2024. Published: October 14, 2024.

1 Introduction

The development of intelligent manufacturing and smart machines in modern industry has been fuelled by the integration of artificial intelligence (AI) and big data methodologies, [1]. It is anticipated that this AI-supported Industry 4.0 revolution will enhance industrial productivity by at least 30% within a few years of its implementation, [1]. This innovation employs the use of artificial intelligence (AI) to reduce the incidence of machine failure, enhance quality control, boost productivity, and markedly reduce product costs. Consequently, humanity is undergoing a significant transformation that will fundamentally alter the future and our way of life, [2].

This trend is also bringing significant changes to modern manufacturing. Firstly, the development of smart factories capable of collecting and storing process data in real-time through information automation technology has advanced, [2]. These intelligent factory operation technologies can create value-added data through big data analysis, thereby improving quality within the production process. Furthermore, in the era of the Fourth Industrial Revolution, production methods are shifting from mass production of a few varieties to small-batch production of various varieties, [3]. This illustrates that an era has arrived where it is necessary to secure not only product quality but also service quality and brand quality through the evolution of technologies such as sensors, the Internet of Things, and big data utilization. Consequently, the speed of quality transformation must be swiftly adjusted to meet customer preferences, [3]. In line with these trends, modern manufacturing is increasingly transitioning to a system of small-batch production of various varieties to meet diverse customer demands and respond quickly to market changes.

The objective of this study is to examine the factors contributing to the observed decline in productivity at the Korean motor manufacturer *K*, resulting from the practice of small-batch production for a range of different types of small motors. In the case of mass production of standard items, a standardized production flow can be designed and standardized as the same product is continuously manufactured on a single piece of equipment, [4].

However, in small-batch production processes of various types, products with different characteristics are mixed and manufactured. In one case, a company reported that the model is changed more than 10 times per day on average per production line, [2]. Each time this occurs, the preparation and production flow must be adjusted continuously, resulting in only 5% of the manufacturing time being spent on actual processing and preparation, while the remaining 95% is non-processing time, leading to waste.

Manufacturers producing small-batch, multi-variety products apply the concept of Group Technology (GT) to solve the aforementioned problems and achieve their goals, [5], [6]. By classifying similar parts based on shape, dimensions, or processes, and applying optimized part design, machine allocation, tools, and work methods to each group, this methodology minimizes the radius of action and setup time, thereby enhancing productivity. This methodology reclassifies parts into part families with similar design or manufacturing characteristics to maximize efficiency, [5], [7]. However, there are several issues with the GT methodology.

Firstly, it should be noted that the results may vary depending on the subjective judgment, experience, or preferences of the individual responsible for classifying the parts, [7]. Secondly, the process of analysis and grouping is inherently time-consuming and costly, and it can be challenging to analyse when the data volume is large or complex. Finally, the costs associated with data collection and processing may be considerable. While the GT methodology is suitable for small quantities of data, it may be challenging to apply to large-scale data sets.

To address the shortcomings of the GT methodology, the K-means clustering algorithm, a fundamental tool in machine learning, can be employed.

In this study, the efficacy of the K-means clustering algorithm was assessed using the Silhouette Score as a post-hoc evaluation metric. The analysis yielded a Silhouette Score of 0.9158, indicative of a robust clustering effect. This indicates that the K-means clustering algorithm is an effective method for classifying data and can be used to complement the GT methodology, as previously demonstrated in [8], [9].

The structure of the paper is as follows: Section 2 covers several key topics, Group Technology, K-means Clustering, and Hierarchical Clustering. Section 3 describes the proposed HVLV-Motor-KC framework in detail. Section 4 describes the experimental environment, datasets, evaluation measures, and results of the three experiments. Finally, Section 5 presents the conclusions of the three experiments and future research directions.

2 Related Work

2.1 High-Variety Low-Volume Production

The advent of Industry 4.0 has ushered in a new era of technological advancement, with market trends shifting from mass, low-mix production to high-mix, low-volume (HMLV) production, [8]. Small-batch, multi-variety production refers to the method of producing multiple types of products in small quantities, diversifying the production process to be flexible and not limited to a single type, [6]. This approach enables manufacturers to respond flexibly to fluctuations in market demand for customized products. In a small-batch, multi-variety production environment, it is common for order sizes to be small, the number of orders to be high, and the variability of products to increase. This enables the enhancement of production efficiency and maximization of resource utilization.

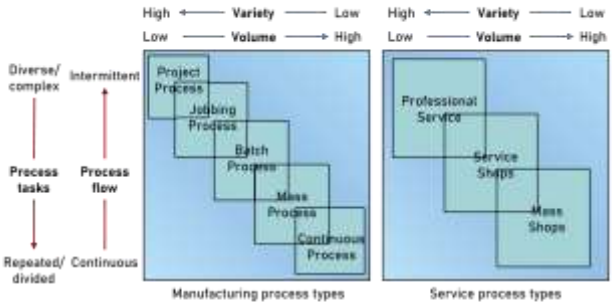

Fig. 1: Types of Manufacturing and Service Processes Based on the Number of Varieties and Production Volume

Figure 1 illustrates the types of manufacturing and service processes based on the diversity of varieties and production volumes in manufacturing. Four characteristics—Volume, Variety, Variation, and Visibility—are interrelated, [10]. Volume and Variety generally have an inverse relationship within a single operational process, determining the position of a specific operational process along this continuum. Both manufacturing and service process types can be identified along this continuum. Manufacturing processes are categorized into Project, Jobbing, Batch, Mass, and Continuous processes, while service processes are divided into Professional, Service, and Mass services. These process types are determined by the inverse correlation between Volume and Variety. After determining the appropriate process type, another crucial factor is the layout. The choice of layout is closely linked to the process type and is generally determined based on the process type. The most common layout types are Fixed Position, Functional, Cellular, and Product layouts.

As consumer demands become more diverse and specific, manufacturers are introducing a variety of products and launching high-performance customized products. Small-batch, multi-variety production reflects this trend and is utilized in various fields such as clothing, jewellery, cosmetics, ships, and robotic devices. In small-batch, multi-variety production, customers directly determine the design, specifications, production volume, and delivery time of products, making management for companies very complex and uncertain. Furthermore, frequent job changes and different specifications and standards require advanced technology. These challenges necessitate manufacturers to adopt more flexible and efficient management systems.

[8], discusses the importance of small-batch, multi-variety systems and the technical requirements to support them. This study explains how small-batch, multi-variety production systems should be designed and operated to meet the demand for customized products. Additionally, [11], emphasizes the optimization of production schedules and the efficiency of personalized product manufacturing by introducing digital twin technology.

2.2 Group Technology

GT is a concept based on the principle of processing similar products in a similar manner.

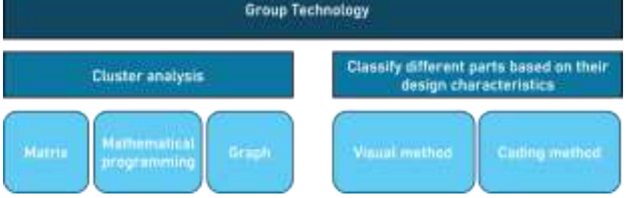

Fig. 2: Group Technology

Figure 2 illustrates two major methodologies in GT. The first methodology is cluster analysis, which groups objects into similar clusters based on their characteristics, [12]. This method is used to minimize setup times and tool change times according to the types and quantities of manufacturing parts, [7]. Through cluster analysis, machines and workstations can be rearranged. In frequently changing production environments, virtual rearrangement can provide a number of benefits. Formal methods for clustering machines and parts include matrix, mathematical programming, and graph methods. The second methodology classifies parts into groups based on their design characteristics. This approach includes visual methods and coding methods. The visual method groups similar parts based on their geometric shapes. This method is suitable when the number of parts is small, but can vary depending on the subjective judgment of the classifier. The coding method assigns numerical or alphabetic codes based on characteristics such as geometric shape, complexity, and machining precision of parts or products. The Opitz coding system is a representative example.

Thus, GT rationalizes design and allocates appropriate production facilities and tools to each classified group, thereby reducing setup times, inter-process transportation, and machining waiting times, [5]. This increases the lot size compared to a disordered production method, achieving an effect similar to mass production and enhancing productivity. Additionally, in production preparation, for previously designed and produced repeat or similar parts, GT allows the calculation of part design, process planning, manufacturing cell design, and estimated manufacturing costs based on data retrieved from part production information, [13].

2.3 K-means Clustering

Although clustering algorithms have been developed for decades, the K-means algorithm remains widely used due to its simple principles, convenience, and high efficiency, [14]. The K-means clustering algorithm, [15], is one of the unsupervised learning algorithms used in machine learning. It groups similar data points by leveraging their characteristics, [8], [15]. This algorithm partitions a dataset so that data points with similar attributes belong to the same cluster, [8]. It provides a method to divide points in a multi-dimensional space into k clusters. The algorithm classifies n points into k clusters, ensuring that points within each cluster share similar characteristics and exhibit different attributes from points in other clusters. This results in the formation of clusters where similar data points are grouped together.

K-means clustering has the advantage of being easy to apply to large-scale and high-dimensional data. However, it has the disadvantage that the number of clusters must be predetermined by the analyst. In addition, the randomness of the initial centroid selection can lead to different results each time it is run. To mitigate these drawbacks, initialization methods such as K-means++ and various modified algorithms have been proposed to achieve more stable and consistent clustering results.

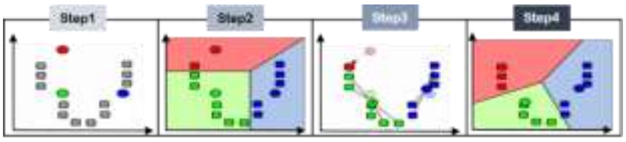

Fig. 3: Steps in Performing K-means Clustering

Figure 3 shows the steps involved in performing K-means clustering. There are four steps in the K-means clustering process. Firstly, a number of clusters or a value of *k* are determined, chosen by empirical methods or domain knowledge. In the second step, initial centroids are randomly assigned and all data points are assigned to the most closely. The third step is to calculate the average of the data points classified to the respective clusters and to update the centroids in accordance with this calculation. Finally, the fourth step repeats the third and fourth steps until the centroids stabilize, meaning the algorithm continues iterating until there are no changes in the centroids, confirming convergence. Through this process, the final cluster for each data point is determined.

2.4 Hierarchical Clustering

Hierarchical clustering is an algorithm that merges or splits data samples into groups based on their similarity to form a hierarchical structure, [16].

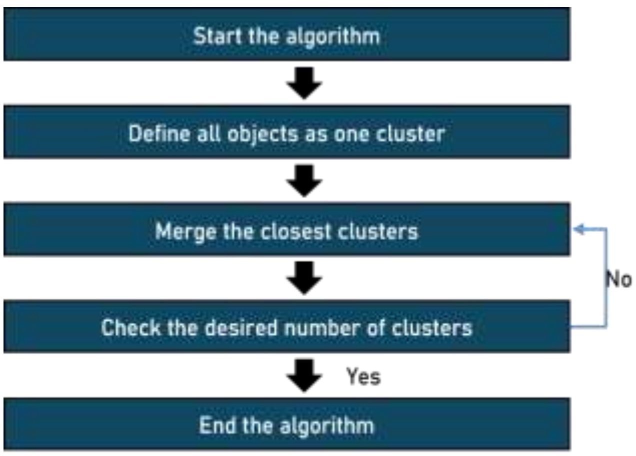

Fig. 4: Hierarchical Clustering Algorithm

As shown in Figure 4, this algorithm captures various levels of relationships between clusters and creates clusters through a step-by-step merging process of data samples. For example, when there are n samples, initially each sample starts as an individual cluster. Subsequently, the most similar pair of clusters is repeatedly merged to reduce the number of clusters, continuing this process until only one cluster remains. This continuous merging process is central to hierarchical clustering and is useful for understanding the structure of clusters based on data similarity, [17], [18].

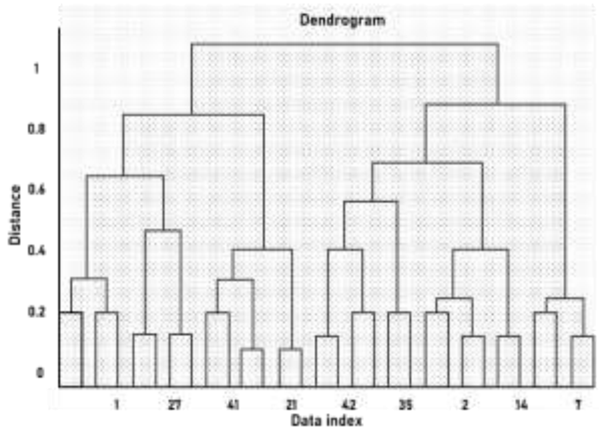

Fig. 5: Hierarchical Clustering Dendrogram

The aforementioned process allows for the generation of a dendrogram, as illustrated in Figure 5, which provides a visual representation of the complete clustering process and offers an intuitive understanding of the relationships between clusters. This hierarchical approach is an effective method for discovering and understanding the inherent structure of the data.

The field of hierarchical clustering is characterized by two principal approaches: agglomerative clustering, as outlined in [19] and divisive clustering, as detailed in [20]. In the context of agglomerative clustering, the process commences with each data point constituting an independent cluster. Thereafter, the most analogous clusters are repeatedly merged. In the initial stage of the process, each data point is represented by a distinct cluster. The procedure continues until all points are merged into a single cluster. In contrast, divisive clustering commences with all data points in a single cluster and proceeds to repeatedly divide the most disparate clusters, continuing until each data point becomes its own distinct cluster. Figure 6 illustrates the process of agglomerative hierarchical clustering, accompanied by a visual representation of the resulting dendrogram.

Hierarchical clustering forms clusters based on the Euclidean distance between data points or clusters without the use of an objective function. This method circumvents the issue of initial parameter determination, thereby distinguishing it from K-means clustering. The outcome of K-means clustering is contingent upon the initial parameter settings, which may result in disparate clustering outcomes for the same dataset. In contrast, hierarchical clustering is not susceptible to the influence of initial parameter settings, thereby ensuring the generation of consistent clustering results. Nevertheless, hierarchical clustering is not without its limitations, particularly in regard to its applicability to large datasets. The computational complexity of this method increases exponentially with the number of data points, due to the necessity of calculating and storing the distance for every pair of data points. Consequently, this significantly increases the demand on memory and the time required for computation, rendering it inefficient for large datasets.

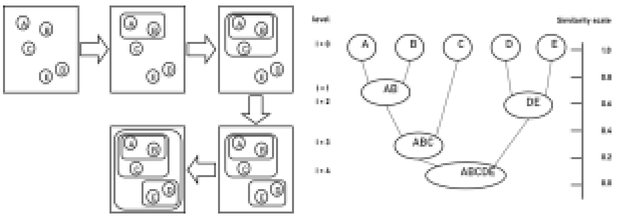

Fig. 6: Agglomerative Hierarchical Clustering & Dendrogram

3 HVLV-Motor-KC

3.1 Methodology

HVLV-Motor-KC proposes a method that can efficiently and flexibly handle complex datasets, such as those of high-variety low-volume motor data. Figure 7 provides an explanation of the methodology.

Previously, the GT method was used. This method mainly relies on subjective judgment and experience, which poses limitations when dealing with complex patterns or large volumes of data. Managing data through Excel sheets is time-consuming and costly, and it is difficult to apply this method across various environments or fields. Therefore, classifying high-variety, low-volume motor items through GT is challenging.

The framework of HVLV-Motor-KC has the following features. First, it automates the clustering process to minimize human intervention, improving processing speed and efficiency. Second, K-means clustering groups data points more accurately based on similarity. Third, it performs clustering considering the characteristics of diverse data, making it applicable to various types of data. Fourth, it provides the ability to process large-scale datasets quickly and effectively. Finally, it is designed to respond swiftly to changes in data.

This methodology can be particularly useful in areas such as production line optimization, inventory management, and product development. The data insights obtained through clustering can enhance production efficiency, improve product customization, and contribute to a more accurate understanding of customer needs.

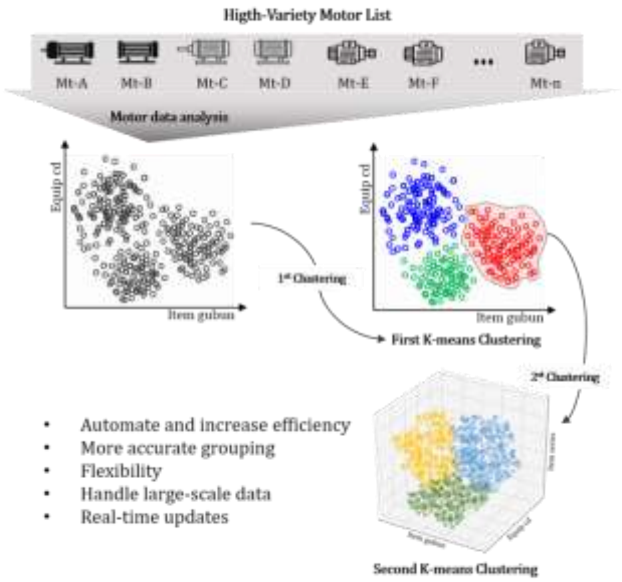

Fig. 7: HVLV-Motor-KC Framework

3.2 K-means Clustering

HVLV-Motor-KC algorithm is a K-means algorithm that clusters data points through multiple stages. Figure 8 explains the algorithm.

Figure 8 compares the general K-means clustering algorithm with the HVLV-Motor-KC algorithm to help clearly understand the differences and characteristics of each algorithm. The data input for the general K-means clustering algorithm is based on the initial dataset. This algorithm is executed to divide the given dataset into *k* clusters. In Figure 8, the number of clusters is set to 3. As a result, the data points are grouped into three clusters, with each cluster represented in a different color.

The data input for the HVLV-Motor-KC algorithm is based on the same dataset as the general K-means clustering algorithm. In Figure 8, the number of clusters is set to 3 for all levels. In the first level of clustering, the initial dataset is divided into three clusters. Each cluster formed at the first level becomes the dataset for the second level. Therefore, the number of clusters for the next level is the same as the *k* value set in the previous clustering, which is 3 in Figure 8. Within each cluster, repeated clustering allows for further refinement in motor classification. To classify these multi-variety motors, we iteratively train the K-means algorithm to find patterns in each motor and classify them by their unique characteristics.

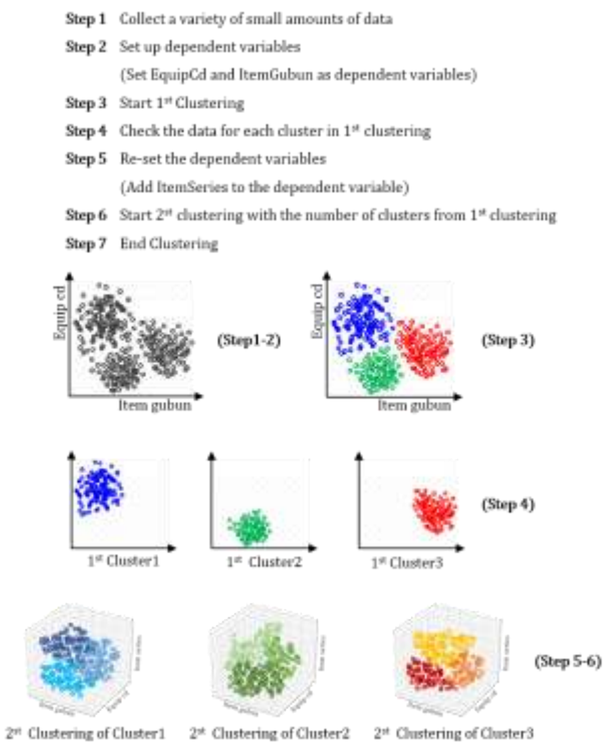

Fig. 8: K-means Clustering Algorithm

4 Experiment and Results

4.1 Experimental Environments

As shown in Table 1, the hardware configuration is based on a 12th Gen Intel® Core™ i7-1260P CPU, Intel® Iris® GPU, 16 gigabytes (GB) of RAM, and an M.2 SSD for storage.

Table 1. Hardware Configuration

Item	Description
CPU	12th Gen Intel(R) Core(TM) i7-1260P
GPU	Intel(R) Iris(R) Xe Graphics
RAM	16.0GB
Storage	M.2 SSD

The software environment was configured as follows. The operating system used was Windows 11, and Python 3.10.12 was adopted as the programming language. For data analysis and visualization, libraries such as Numpy 1.25.2, Pandas 2.0.3, scikit-learn, UMAP, Matplotlib, and Seaborn were utilized. The development environment used was Jupyter Notebook, and the virtual environment was set up through Anaconda. This software configuration facilitates the efficient execution of various data processing and analysis tasks. Table 2 summarizes the software configuration environment.

Table 2. Software Configuration

Item	Description
Operating System	Windows 11
Programming Language	Python 3.10.12
Libraries	NumPy 1.25.2, Pandas 2.0.3
Dev Environment	Google Colaboratory

Experiments were conducted in the hardware and software environment configured as described, enabling smooth execution of various data processing and analysis tasks. This environment configuration enhances the reproducibility of the experiments and is suitable for meeting diverse data processing and analysis requirements.

4.2 Datasets

The dataset collected for this study is based on the production data of a Korean small and medium-sized enterprise (SME) *K*, a manufacturer of high-variety, low-volume motors, primarily consisting of small, medium, and ultra-small motor product families.

Table 3. Motor Dataset

ITEM_GUBN	ITEM_SERIES	EQUIP_CD
SMALL	KAFZ	EQ004
SMALL	KAFZ	EQ004
MEDIUM	PSMR	EQ001
ULTRA-SMALL	PSMS	EQ007
MEDIUM	RTSE	EQ015

As shown in Table 3, the items used to differentiate the data in the actual field include ITEM_CD (item code), ITEM_REV (item revision), ITEM_NM (item name), PROD_GUBUN (production classification), ITEM_GUBUN (item classification), ITEM_SERIES (item series), LEAD_TIME (production time), and EQUIP_CD (equipment code).

A total of 909 motor data records were collected. Of these, 727 records (80%) were used as training data, while the remaining 182 records (20%) were utilized as test data for performance evaluation.

To train the HVLV-Motor-KC algorithm on the motor data, we performed several preprocessing steps. First, we removed rows that contained missing values for EQUIP_CD and ITEM_GUBUN, which are essential fields for the analysis.

Then, we converted the categorical data, EQUIP_CD, ITEM_GUBUN, and ITEM_SERIES, into a and converted them to numeric codes by applying data scaling.

The main variables used in the experiment are ITEM_GUBUN, EQUIP_CD, and ITEM_SERIES. Table 4 shows the variables used for each primary and secondary clustering.

Table 4. Variables for Clustering 1 and 2

Clustering	ItemGubun	EquipCd	ItemSeries
1	●	●	x
2	●	●	●

4.3 Evaluation Metrics

The silhouette coefficient is an indicator of how closely a data point is clustered based on its distance from data points in similar clusters, [21] and how far it is distributed from data in other clusters. The silhouette coefficient can have a value between -1 and 1, with values closer to 1 indicating better clustering. The HVLV-Motor-KC algorithm was evaluated using the silhouette coefficient above.

$$a(i) = \frac{1}{|C| - 1} \sum_{i \in C, i \neq j} \mathrm{d(i,j)} \qquad (1)$$

$$\mathrm{b(i)} = \frac{min}{C` \neq C} \frac{1}{|C`|} \sum_{i \in C`} d(i,j) \qquad (2)$$

$$\mathrm{s(i)} = \frac{b(i) - a(i)}{\max\left(a(i), b(i)\right)} \; where - 1 \leq s(i) \leq 1 \qquad (3)$$

$$\text{Silhouette Score} = \frac{1}{n} \sum_{i=1}^{n} s(i) \qquad (4)$$

4.4 Visualization of Clustering Numbers

To find the optimal number of clusters for the HVLV-Motor-KC algorithm, we used the silhouette and elbow methods.

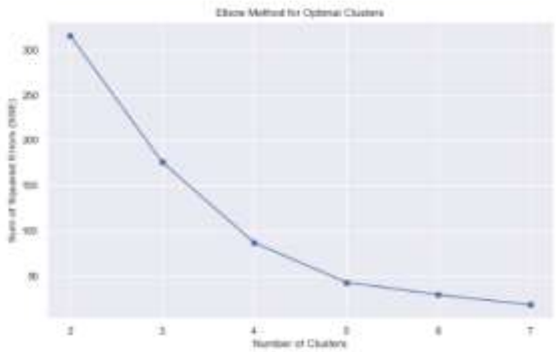

Fig. 9: Elbow Method

The first silhouette measured the number of clusters with the highest silhouette score, while the second elbow method measured the number of clusters with a point where the Within Cluster Sum of Squares (WCSS) decreases sharply.

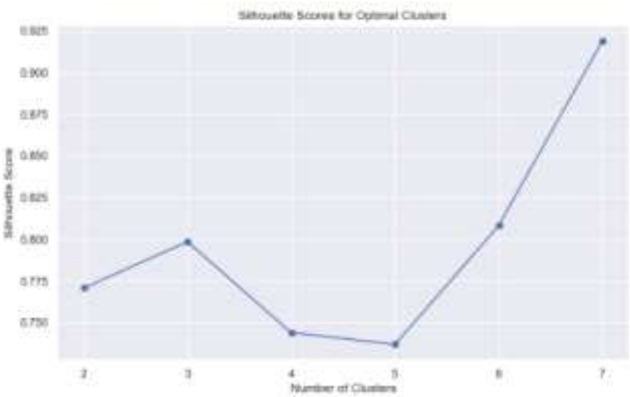

Fig. 10: Silhouette

We experimented with finding the optimal number of clusters using the above method and determined that seven clusters was the best number, as shown in Figure 9 and Figure 10.

Table 5 shows the SSE (Within-Cluster Sum of Squares) and silhouette scores for different numbers of clusters. When the number of clusters is *7*, the SSE sharply decreases to 18.0726, indicating that data points are more closely grouped around each cluster center. The silhouette score is the highest at 0.9188, indicating the best separation between clusters.

Additionally, the initial values for the cluster centers (init) were set using the k-means++ algorithm. This approach selects the initial cluster centers in a better way to improve the performance of clustering. The number of times to use different initializations was set to the default value of 10. The maximum number of iterations (max_iter) was set to 300 to allow sufficient iterations for the algorithm to converge.

Table 5. Comparison of Elbow Method and Silhouette

n-Clusters	SSE	Silhouette Score
2	315.73772788646966	0.7707042591059704
3	175.85620616762893	0.79852165813859
4	86.75635309788157	0.7439876446148647
5	42.993683685000555	0.7369224511116048
6	29.25106647321429	0.8081641831658453
7	18.072619525625903	0.9188765290433094

In summary, by setting the optimal number of clusters to *7*, the motor data could be grouped most effectively according to similar characteristics.

4.5 First Classification

Figure 11 presents the results of the initial clustering process. Each cluster is delineated by a distinct colour, facilitating the identification of its distribution and boundaries in relation to other clusters. The size of the data points is adjusted proportionally to the number of samples within the corresponding cluster; that is, the more samples in a cluster, the larger the size of the data point. This visualization allows for an immediate understanding of the relative size and density of each cluster.

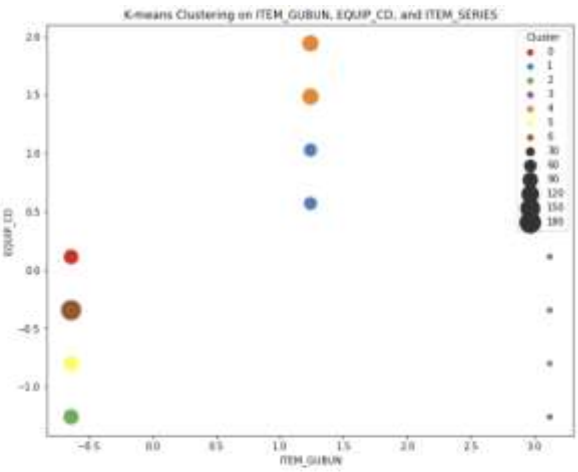

Fig. 11: First Classification on *ItemGubun* and *EquipCd*

Table 6 shows the exact number of data points belonging to each cluster, providing specific information on the cluster distribution. This helps to clearly understand the composition and distribution of each cluster.

Table 6. Comparison of Elbow Method and Silhouette

n-Clusters	Sample Count
0	102
1	84
2	109
3	21
4	122
5	105
6	185

Cluster 6, consisting of 185 data points, forms the largest cluster and accounts for a significant portion of the total data. The next largest cluster is Cluster 4, which includes 122 data points. In comparison, the data points in cluster 3 show the smallest number of data points, 21. The first clustering result from K-means gives a silhouette score of 0.9188, which is very close to 1, indicating that the boundaries between clusters are clear and the data is well distributed.

4.6 Second Classification

Based on the actual classification of motors in Company K, we performed primary clustering by equipment code and item classification, and then secondary clustering by item series. When we made multiple classifications, such as primary, secondary, and so on, we were able to make more detailed and sophisticated clustering. In the second stage, the ITEM_SERIES variable was incorporated into the process to facilitate a more refined clustering approach. Any missing values in the ITEM_SERIES field were treated as *Other*.

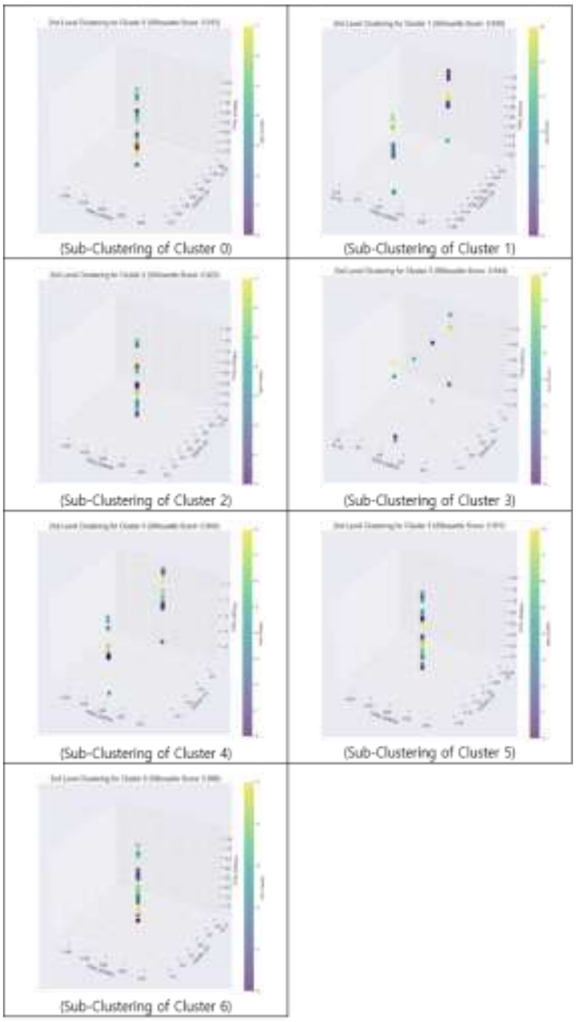

Fig. 12: Second Classification on *ItemGubun*, *EquipCd* and *ItemSeries*

Yeji Do, Chaegyu Lee, Jongpil Jeong,
Jiho Jeong, Donggeun Bae,
Inkwon Yeo, Mingyu Kim

In the second stage of the clustering process, each cluster generated in the initial stage was further subdivided in order to identify more detailed patterns within the motor production data. A further iteration of K-means clustering was conducted within each cluster, incorporating the ITEM_SERIES variable. In order to ascertain the optimal number of clusters, the elbow method and silhouette analysis were employed, in a manner analogous to that undertaken in the initial stage of clustering. The initial value settings and other optimal algorithm settings were maintained in accordance with those employed in the initial stage of clustering.

Figure 12 visualizes the results of the second-stage clustering based on the first-stage clustering. Each cluster is represented in a different color and visualized in 3D. This visualization allows the silhouette scores of each cluster to be confirmed, enabling a more detailed grouping of the motors.

Table 7. Comparison of Silhouette

n-Clusters	Silhouette Score
0	0.913
1	0.840
2	0.923
3	0.944
4	0.848
5	0.911
6	0.888

Table 7 shows the number of clusters and silhouette scores for the second-stage clustering. The number of clusters was determined to be between 6 and 9, and all clusters had silhouette scores above 0.8, indicating that the clustering was very successful. Particularly, Cluster 3 recorded the highest silhouette score of 0.9444, indicating that the data points within this cluster are very densely packed and clearly distinguished from other clusters.

4.7 Results

We applied the HVLV-Motor-KC algorithm to optimize the production of 900 small, medium, and ultra-small motors by classifying them into equipment codes, item categories, and item series. As the results of the experiment show in Table 8, both primary and secondary clustering in the training data showed the highest silhouette score of 0.91742, which is consistent with the process of classifying multi-variety motors into seven motors by field workers in the actual field, thus increasing the applicability in the real industry.

Table 8. Comparison of Silhouette Score

	Silhouette Score
train_data	0.9174270361483706
test_data	0.9195868772687471

Figure 13 shows, keeping the number of clusters at 7, applying the test data resulted in a high silhouette score of 0.91958, demonstrating that the proposed HVLV-Motor-KC algorithm can classify new varieties into similar types of varieties even when new varieties are introduced.

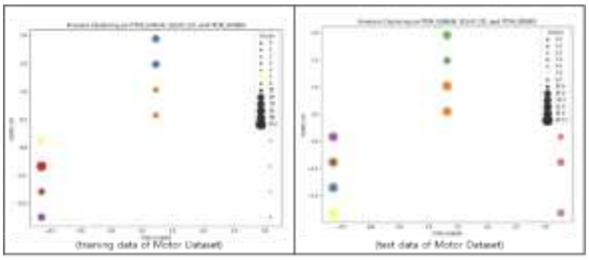

Fig. 13: Clustering Comparison of Training and Test Data

5 Conclusion

The HVLV-Motor-KC algorithm demonstrated high performance on training and test data from a large variety of low-volume motors, showing that it can efficiently classify complex and large volumes of motor data to optimise production management.

Further research is required to enhance the algorithm's performance by conducting a comparative analysis with alternative data types and clustering techniques. This will assist in further verifying the versatility and stability of the HVLV-Motor-KC algorithm. Moreover, further experiments will be conducted with the objective of enhancing the practicality of the algorithm and advancing the research in order to increase its applicability in real industrial environments.

Finally, a user-friendly interface will be developed to facilitate straightforward utilisation in practical applications. In order to achieve this, tools that provide a visual representation of the algorithm's results will be developed, thereby enabling users to directly control and analyse the clustering process.

Acknowledgement:
This research was supported by the SungKyunKwan University and the BK21 FOUR (Graduate School Innovation) funded by the Ministry of Education (MOE, Korea) and National Research Foundation of Korea (NRF). Moreover, This work was supported by ICT Creative Consilience Program through the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT)(IITP-2024-2020-0-01821).

Declaration of Generative AI and AI-assisted Technologies in the Writing Process

During the preparation of this work the authors used ChatGPT in order to improve the readability of the paper. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References:

[1] Jagatheesaperumal, S. K., Rahouti, M., Ahmad, K., Al-Fuqaha, A., & Guizani, M., The duo of artificial intelligence and big data for industry 4.0: Applications, techniques, challenges, and future research directions, *IEEE Internet of Things Journal*, Vol. 9, No. 15, 2022, pp. 12861-12885.

[2] Riew, M. C., & Lee, M. K., A Case Study of the Construction of Smart Factory in a Small Quantity Batch Production System: Focused on IDIS Company, *Journal of Korean Society for Quality Management*, Vol. 46, No. 1, 2018, pp. 11-26.

[3] Chong, H. R., Bae, K. H., Lee, M. K., Kwon, H. M., & Hong, S. H., Quality strategy for building a smart factory in the fourth industrial revolution, *Journal of Korean Society for Quality Management*, Vol. 48, No. 1, 2020, pp. 87-105.

[4] Im, K.-H., Rule-based Process Control System for multi-product, small-sized production, *Journal of Korea Society of Industrial Information Systems*, Vol. 15, No. 1, 2010, pp. 47–57.

[5] Park, H. K., & Oh, C. J., Integration of design and process planning using group technology, *Proceedings of the Korean Society for Intelligent Information Systems Conference*, 1997, pp. 107-112.

[6] Gödri, I., Improving Delivery Performance in High-Mix Low-Volume Manufacturing by Model-Based and Data-Driven Methods, *Applied Sciences*, Vol. 12, No. 11, 2022, pp. 5618.

[7] Park, G. J., & Park, J. W., A Study on the Application of Group Technology for Naval Ship Design and Manufacturing, *Journal of the military operations research society of Korea*, Vol. 32, No. 2, 2006, pp. 78-91.

[8] Ahmed, M., Seraj, R., & Islam, S. M. S, The k-means Algorithm: A Comprehensive Survey and Performance Evaluation, *Electronics*, Vol. 9, No. 8, 2020, pp. 1295.

[9] Capó, M., Pérez, A., & Lozano, J. A., An efficient K-means clustering algorithm for tall data, *Data mining and knowledge discovery*, Vol. 34, 2020, pp. 776-811.

[10] Brown, S., Blackmon, K., Cousins, P., & Maylor, H., *Operations management: policy, practice and performance improvement*, Routledge, 2013

[11] Sit, S. K., & Lee, C. K., Design of a Digital Twin in Low-Volume, High-Mix Job Allocation and Scheduling for Achieving Mass Personalization, *Systems*, Vol. 11, No. 9, 2023, pp. 454.

[12] Shahin, A., & Janatyan, N., Group Technology (GT) and Lean Production: A Conceptual Model for Enhancing Productivity, *International Business Research*, Vol. 3, No. 4, 2010, pp. 105-117.

[13] Askin, R. G., & Chiu, K. S., A graph partitioning procedure for machine assignment and cell formation in group technology, *The International Journal of Production Research*, Vol. 28, No. 8, 1990, pp. 1555-1572.

[14] Hu, H., Liu, J., Zhang, X., & Fang, M., An effective and adaptable K-means algorithm for big data cluster analysis, *Pattern Recognition*, Vol. 139, 2023, pp. 109404.

[15] Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhaija, B., & Heming, J., K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data, *Information Sciences*, Vol. 622, 2023, pp. 178-210.

[16] Pal, S. S., Mukhopadhyay, J., & Sarkar, S., Finding hierarchy of clusters, P*attern Recognition Letters*, Vol. 178, 2024, pp. 7-13.

[17] Nielsen, F., & Nielsen, F., Hierarchical clustering, *Introduction to HPC with MPI for Data Science*, 2016, pp. 195-211.

[18] Campello, R. J., Moulavi, D., & Sander, J., *Density-based clustering based on hierarchical density estimates, Pacific-Asia Conference on Knowledge Discovery and*

Data Mining, Berlin, Heidelberg, Springer Berlin Heidelberg, Vol. 7819, 2013, pp. 160-172.

[19] Rodriguez, M. Z., Comin, C. H., Casanova, D., Bruno, O. M., Amancio, D. R., Costa, L. D. F., & Rodrigues, F. A., Clustering algorithms: A comparative approach, *PloS one*, Vol. 14, No. 1, 2019, pp. e0210236.

[20] Ran, X., Xi, Y., Lu, Y., Wang, X., & Lu, Z., Comprehensive survey on hierarchical clustering algorithms and the recent developments, *Artificial Intelligence Review*, Vol. 56, No. 8, 2023, pp. 8219-8264.

[21] Minh, H. L., Sang-To, T., Wahab, M. A., & Cuong-Le, T., A new metaheuristic optimization based on K-means clustering algorithm and its application to structural damage identification, *Knowledge-Based Systems*, Vol. 251, 2022, pp. 109189.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

This research was supported by the SungKyunKwan University and the BK21 FOUR (Graduate School Innovation) funded by the Ministry of Education (MOE, Korea) and National Research Foundation of Korea (NRF). Moreover, This work was supported by ICT Creative Consilience Program through the Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT)(IITP-2024-2020-0-01821).

Conflict of Interest

The authors have no conflicts of interest to declare.