

Case study: the reliability of clustering solutions

Mustafa Kadhim

kadhim_mustafa@uestc.edu.cn

University of Electronic Science and Technology of China

Guangxi Lu

University of Electronic Science and Technology of China

Jianbo Wang

University of Electronic Science and Technology of China

Zhao Kang

University of Electronic Science and Technology of China

Ruyi Tang

Phoenix Intelligent Innovation Technology Co., Ltd, Shapingba Industrial Zone A

Songlin Ye

Phoenix Intelligent Innovation Technology Co., Ltd, Shapingba Industrial Zone A

Research Article

Keywords: Pattern Recognition, Exclusive Clustering, Data Overlapping, Machine Learning, Feature Selections

Posted Date: August 30th, 2024

DOI: <https://doi.org/10.21203/rs.3.rs-4690535/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Case study: the reliability of clustering solutions

Mustafa Kadhim^{1*}, Guangxi Lu¹, Jianbo Wang¹, Zhao Kang¹,
Ruyi Tang², Songlin Ye²

¹ School of Computer Science and Engineering (School of Cyber Security), University of Electronic Science and Technology of China, West Hi-Tech Zone, Chengdu, 611731, Sichuan, China .

² Chongqing Seres, Phoenix Intelligent Innovation Technology Co., Ltd, Shapingba Industrial Zone A, Chongqing, 400000, China .

*Corresponding author(s). E-mail(s): kadhim.mustafa@uestc.edu.cn;
Contributing authors: [@uestc.edu.cn](mailto:glu3; 202322081427; zkang);
[@seres.cn](mailto:ruyi.tang; songlin.ye);

Abstract

Unsupervised clustering analysis (CA) models are crucial in various machine learning tasks aiming to discover the data structure and hidden patterns between samples. The performance of CA is affected by overlapping issue (samples that are similar to each other but do not belong to the same cluster). These samples may occur when the data grows over time, which causes the need to redefine features, the distance matrix, centroids, or other parameters. As recent CA models achieved optimal solutions, the details of parameters have been ignored due to the “no-free lunch” theory, which raises the question about their reliability. This study conducts an in-depth investigation of the complex relationships among the given issues, aiming to clarify the key problems associated with unsupervised models that may lead to inaccurate prediction of patterns, inappropriate partitioning, and false confidence in outcomes. We designed a novel experiment* that produced over 271 million possible settings based on various models, equations, and procedures. The experimental results* demonstrate that obtaining optimal parameters for optimal solutions is as challenging as finding a needle in a haystack. Finally, we designed an open issue* that challenges the CA models because our experiments raised doubts over the use of supervised information upfront to determine the optimal solutions, which is against the concept of CA

Keywords: Pattern Recognition, Exclusive Clustering, Data Overlapping, Machine Learning, Feature Selections

*Released at the supplementary files and github.com/MustafaRaadKadhim/CaseStudy

1 Introduction

Clustering analysis (CA) [1] is an important component of machine learning tasks such as pattern recognition, Internet of Things (IoT), computer vision, and natural language processing, as it utilizes an effective unsupervised framework to uncover hidden patterns and categorize data samples according to their similarities [2]. Exclusive clustering (EC) and overlapped clustering (OC) are two significant tasks in CA [3]. In EC, each data point is assigned to a single cluster, whereas in OC, data samples can be assigned to multiple clusters simultaneously, allowing for overlap. The choice between EC and OC depends on the characteristics of the data and the tasks executed. However, the pattern interpretation and implementation of the EC are easier and more straightforward compared to the OC. Still, the EC may experience inconsistent partitioning if the data naturally contains overlap between clusters [4].

As Figure 1 shows different types of data shapes, both the OC and EC models would succeed in the non-overlapped data Figure 1a. Thus, the overlapped data in Figure 1b would make it much more difficult for EC to obtain the correct partitioning where the interpretation of OC would be unclear and it would be difficult to judge the correct belongings of the overlapped area. Thus, the more overlapped samples appear in the dataset, the more likely OC and EC fail to obtain the correct partitioning and interpretation.

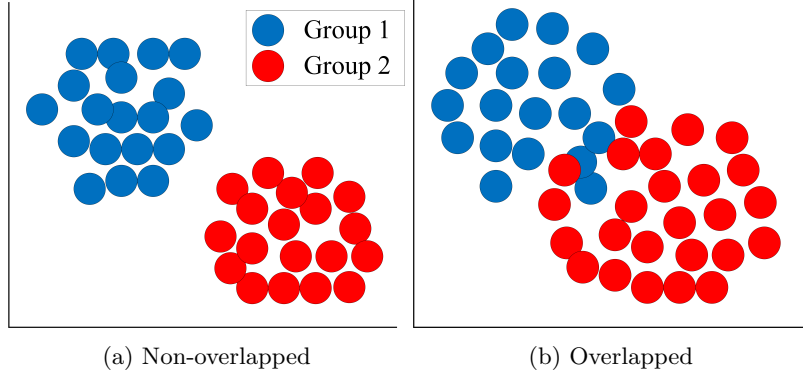


Fig. 1: Datasets types in clustering

Many EC models have been proposed, aiming to provide a solution for the overlapping issue by integrating constraints, consensus functions, Gaussian mixtures, and spectral techniques [5, 6]. These models might contain a multitude of parameters that demand fine tuning to reach the optimal solution for a specific dataset [4]; reaching optimal tuning with unsupervised models can be challenging, which impacts the model's performance. Still, these models suffer from various issues [7], such as complexity and computation, which make them less effective with large datasets.

Recent studies have investigated the impacts on the performance of EC models from various perspectives. Article [8] conducted a study on how the growing state

affects performance and found that it may require readjusting the initial status. Thus, [4] studied the effects of overlapping issues and the challenges associated with distinguishing between the overlapping samples. However, there is no in-depth study of the tangled sensitivity between the initialization status, growing state, overlapped samples, feature selection, and distance matrix toward the performance of the models. These aspects are highly sensitive to each other, potentially causing the model to adjust its status and selection once it experiences a growing state that may contain overlapped samples [7, 9]. Thus, [10, 11] unsupervised-based studies depended on the “no-free lunch” theorem [12, 13] in their developments, where reaching the determined parameters and settings might be unreliable for unsupervised solutions.

We have suggested an experiment that involves the supervised generation of all possible combinations of growing state, initial status, features, and distance matrix. This experiment aims to conduct an in-depth analysis and provide a comprehensive understanding of the performance of EC-based models in each of these circumstances. We selected the Iris dataset for its availability of overlapped samples and its simplicity, making it an optimal choice for benchmarking and validating models. Our experimental results reinforce that determining the optimal centroids and parameters for unsupervised clustering would be an impossible task for some models. Also, every growing state has its own optimal settings because they might be improper for data in the next growing state where overlapped or outlier samples might exist. The contribution of this work is detailed as follows:

- 1- The first comprehensive study of the tangled relationships between growing status and overlapped samples had an impact on determining the centroids, features, and distance matrix of clustering solutions.
- 2- We introduce the first supervised experimental study to validate the strength of clustering models in different real-world scenarios. As a result, the experimental validation of the concept of the “no-free lunch” theory towards obtaining the optimal solution by the unsupervised clustering models.
- 3- Provide an open issue related to overlapped samples, partitioning, and confidence in results. Also, provide over 271 million performance outcomes for further analysis and investigation.

The remainder of the paper consists of five sections: related work, concerns assumption and formulation, experimental settings and design, experimental results and discussion, and finally the conclusion.

2 Related Work

2.1 Exclusive Clustering

EC aims to identify natural patterns within complex datasets by categorizing data samples into distinct, meaningful clusters. Let X denote a dataset that contains N samples and M features, where $1 \leq i \leq N$ and $1 \leq j \leq M$. Each $x_{i,j}$ represents the value of the j -th feature in the i -th sample. The EC aims to partition X into K groups, with each group represented by k containing a certain number of samples.

The clustering outcomes, denoted as $\Lambda^p = [\lambda_1^p, \lambda_2^p, \dots, \lambda_N^p]$, where λ^p is also termed as predicted labels, are expected to match the ground truth labels Λ^g .

However, the “no-free lunch” theorem [12, 13] asserts that no individual clustering algorithm can be universally optimal for all datasets, as each algorithm possesses its own unique strengths and weaknesses, and none of the clustering models can uncover all possible cluster shapes and structures. Therefore, every algorithm necessitates specific parameters, including cluster numbers, distance metrics, and centroid settings. Determining these parameters can be challenging for users, resulting in numerous results for a given dataset.

Article [4] possesses a dataset that includes overlapping samples, and the objective requires accurate results. As for unsupervised clustering, there will be no further information about Λ^g and K . In this scenario, the “no-free lunch” theorem fails to be fair due to the lack of knowledge regarding the data structure, which limits our ability to adopt the most optimal clustering algorithm that will efficiently cluster the data. Thus, the reliability of the reliability of the outcomes will be questionable.

2.2 Baselines approaches

We chose nine well-known unsupervised EC models to provide a real-world scenario of unsupervised clustering where the nature of the data is ambiguous. We assigned $\mathcal{M}$ to refer to these models.

The density peaks (DP) clustering algorithm [11] is valued for its simple parameter determination and execution, effectively addressing non-spherical clustering problems based on distances between data samples. It identifies cluster centers by their higher density compared to neighbors and significant distance from points with higher densities. This approach allows for intuitive cluster number determination, automatic outlier detection, and shape-independent clustering.

The affinity propagation (AP) clustering algorithm [14] enables “message passing” between data points without predefining the number of clusters. It autonomously identifies cluster centers by treating each data point as a potential center, using negative Euclidean distance to measure similarities. As a data point’s affinity with others increases, its likelihood of being selected as a cluster center rises. AP is efficient for datasets with many clusters or complex patterns. However, it has significant temporal complexity, making it unsuitable for very large datasets, and its results are highly sensitive to parameter settings, impacting its effectiveness and reliability.

The MPFC algorithm introduces [15] a method for fair clustering by balancing data points across clusters, ensuring no cluster is overly represented by any color. It uses a center-based approach, assigning objects—each colored one of two specified colors—to clusters via a binary linear optimization program that enforces fairness constraints. Users can choose from clustering targets like k-means (MKM), k-medians (MKMDA), and k-medoids (MKMDO). For larger datasets, MPFC employs a two-step scaling strategy: generating potentially unfair sub-clusters and then merging them into fair clusters. A key advantage of MPFC is its ability to incorporate additional constraints, enhancing its versatility for various applications.

Finally, kmedoids-fasterpam (KFP) [16] is a modified version of the k-medoids clustering algorithm, designed to specifically enhance the PAM (Partitioning Around

Medoids) algorithm. The main improvement of KFP is the alteration of the SWAP phase, which leads to a decrease in the computational expense of PAM by a factor of $O(k)$. The kmedoids-fastpam1 (KPF1) algorithm [17] enhances the KFP algorithm by incorporating the concept of eager swapping, resulting in a reduced number of iterations needed for convergence. In addition, kmedoidsfastermc (KFRM) and kmedoids-fastmcs (KFM) [18] incorporate principles from the original Silhouette method with the KFP and KPF1 algorithms, resulting in a runtime acceleration of $O(k^2)$.

2.3 Distance matrix

Distance metrics [19] play a crucial role in addressing various pattern recognition problems, including classification and clustering. The objective is to establish the disparity (or distance) among the different entities within a given dataset. Since all clustering procedures depend on a distance measure, the method used to compute the distance matrix can significantly influence the results of the clustering.

Thus, there are several well-known metrics: (D1) Bray-Curtis distance, (D2) Canberra distance, (D3) Chebyshev distance, (D4) City Block distance, (D5) Correlation distance, (D6) Cosine distance, (D7) Euclidean distance, (D8) Hamming distance, (D9) Jaccard distance, (D10) Jensen-Shannon distance, (D11) Mahalanobis distance, (D12) matching distance, (D13) Minkowski distance, (D14) standardized Euclidean distance, and (D15) squared Euclidean distance.

2.4 Feature selection

Feature selection [20] is a data processing strategy that entails choosing a reduced and distinctive set of significant features from a larger dataset. The objective is to enhance the performance of the machine learning model by employing filter, wrapper, and embedding techniques to eliminate irrelevant and redundant inputs. This will result in improved accuracy, reduced overfitting, and decreased training time. Feature selection is crucial for alleviating the negative impact of the “curse of dimensionality,” which diminishes the effectiveness of clustering algorithms as the number of features grows. Clustering algorithms can enhance pattern recognition and create more well-defined and meaningful clusters by prioritizing important characteristics.

The three primary approaches for feature selection are Univariate feature selection [20], recursive feature elimination [21], and sequential feature selection [22]. Nevertheless, unsupervised feature selection techniques frequently encounter obstacles such as overfitting to particular clustering algorithms, disregarding feature interdependencies, and being vulnerable to noise and outliers. These problems can greatly hinder the efficiency of feature selection and the effectiveness of subsequent models.

2.5 Iris dataset

The Iris dataset [1] is a benchmark real-world dataset widely used in machine learning and data science. It consists of 150 samples of iris flowers, each characterized by four features: sepal length (SL), sepal width (SW), petal length (PL), and petal width (PW). These features are used to classify the flowers into three species: Setosa,

Versicolor, and Virginica. As depicted in Figure 2, the dataset’s well-structured and balanced nature makes it ideal for testing and validating various classification algorithms. Its simplicity allows for easy visualization and intuitive understanding, yet it provides enough complexity to challenge and refine model performance. Many researchers still evaluating their models performances using the Iris dataset [2, 8, 10, 11].

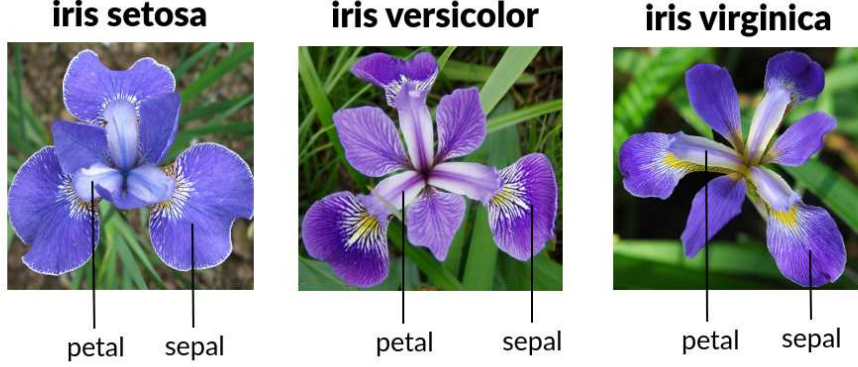


Fig. 2: Species samples of the Iris dataset

3 Concerns assumption and formulation

3.1 growing state

Given that the Iris dataset comprises 150 samples, $N = [1, 2, \dots, 150]$, each of which belong to a single k -group. We assumed in our investigation that the data group would grow by 10% in each execution, with the initial growing state starting at 50%. These percentages will be used for fair picking up, where in every growing state the number of samples of each group will be equal; the growing state is represented as $\mathcal{G}$. The purpose of this separation is to ensure that our experiments will be real-world scenarios in which applications gain more data over time.

As this paper is based on supervised validation, the growing state strategy aims to find the data indices based on Λ^g . The equation is represented as:

$$I^k = \{i \mid \Lambda_{[i]}^g = k, i \in \{1, 2, \dots, N\}\} \quad (1)$$

where k represents a specific clustering group, and $I^k \subseteq \{1, 2, \dots, N\}$. Then, the V function will be responsible for the task of growing the fraction based on the obtained I^k . The indices I^K are defined as:

$$I^K = \{\{I^1\}, \dots, \{I^k\}\} \quad (2)$$

The V function is given as:

$$V(k, \mathcal{G}) = \{I_{[j]}^k \mid j \in \{1, 2, \dots, \lceil \#I^k \cdot \mathcal{G} \rceil\}\} \quad (3)$$

where $\mathcal{G}$ is the growing state fraction. Finally, the processed data $\mathcal{X}$ that will be used in different evaluations is generated as:

$$\mathcal{X} = \bigcup_{k=1}^K V(k, \mathcal{G}) \quad (4)$$

3.2 Centroids concern

Suppose the EC-based model has determined specific centroids $\mathcal{C}$ for $\mathcal{X}$ in a specific period of $\mathcal{G}$. Thus, in the next $\mathcal{G}$ the determined $\mathcal{C}$ might become inappropriate for the new data. As a result, this issue might cause misprediction patterns and demand redetermine $\mathcal{C}$ in every growing. In this work, we have depended on Cartesian product function [23] depending on the subsets in I^K , and it defined as

$$V(1, \mathcal{G}) \times \dots \times V(k, \mathcal{G}) = \{(c_1, \dots, c_k) \mid c_1 \in V(1, \mathcal{G}), \dots, c_k \in V(k, \mathcal{G})\} \quad (5)$$

where $(c_1, \dots, c_k)$ represents a centroids set and each c a center of a specific clustering group. Equation (5) will product $\#V(1, \mathcal{G}) \times \dots \times \#V(k, \mathcal{G})$ centroids possibilities which are going to be sorted into $\mathcal{C}$ in every growing state. Finally, the aim of generating these centroids is to obtain every single possibility that would assist into obtaining the best performance.

3.3 Distance matrix concern

The selected EC-based models are based on calculating the distance between samples. However, utilizing different distance metrics leads to different results, so choosing the optimal distance metric is a challenging task, especially with unsupervised EC. Suppose $\mathcal{D}$ is the distance metrics representor; the EC model in a specific $\mathcal{G}$ would require a specific $\mathcal{D}$ to obtain optimal outcomes. Therefore, in the next $\mathcal{G}$ status, the selected $\mathcal{D}$ might fail to obtain optimal outcomes, necessitating a change to another $\mathcal{D}$.

3.4 Overlapping concern

Many datasets contain a number of samples that contains very similar patterns to each other but naturally belong to different groups. Suppose, $\mathcal{X}_i$ and $\mathcal{X}_j$ are two overlapped data samples that cannot link to each other, $\lambda_i^g \neq \lambda_j^g$. The clustering models or distance matrix outcomes might mistakenly group them together.

$$\mathcal{D}(\mathcal{X}_i, \mathcal{X}_j) \text{ or } \mathcal{M}(x_{i,:}, x_{j,:}) \implies \lambda_i^p = \lambda_j^p, \quad (6)$$

where $\lambda_i^p \neq \hat{\lambda}_i^g$ or $\lambda_j^p \neq \hat{\lambda}_j^g$. This overlapping presents challenges for clustering tasks, as mispredictions can lead to erroneous confidence in the outcomes of unsupervised models.

3.5 Feature concern

Feature selection models have been proposed to be part of addressing the overlapping issue by providing insights into the dataset’s nature. Figure 3 shows that the feature selection possibilities $\mathcal{F}$ of the Iris datasets that were obtained by combining SL, SW, PL, and PW features in a couple, triple, and quarter forms generated 11 $\mathcal{F}$ sets, which are (F1) SL-SW-PL-PW, (F2) SL-SW, (F3) SL-PL, (F4) SL-PW, (F5) SL-SW-PL, (F6) SL-SW-PW, (F7) SL-PL-PW, (F8) SW-PL, (F9) SW-PW, (F10) SW-PL-PW, and (F11) PL-PW.

Figure 3 indicates that Versicolor and Virginica exhibit overlap across every $\mathcal{F}$ possibilities. Further, the overlapping samples may not be present during the initial stages of development but will gradually occur over time. Suppose $\mathcal{U}$ represent the overlapping existence status, the assumed overlapping checker is given as

$$\mathcal{U} = \sum_{i \neq j} 1(\mathcal{X}_{i,\mathcal{F}} \approx \mathcal{X}_{j,\mathcal{F}} \text{ and } \lambda_i^g \neq \lambda_j^g) \quad (7)$$

where $\mathcal{F}$ represents the selected features by the feature selection models. Thus, if $\mathcal{U} > 0$ then there is possibilities of overlapped samples. Figures A1–A11 show that the number of samples that overlapped increased over time. This gives evidence for our concern about the overlapping as well as increases the doubt of the unsupervised feature selection solutions.

3.6 Research questions

These concerns are discussed individually; however, our concern is more complicated because it puts down some important questions that need to be discussed:

Q1: What will be the influence of $\mathcal{G}$, $\mathcal{C}$, and $\mathcal{U}$ on the selection and stability of $\mathcal{D}$?

Q2: What will be the influence of $\mathcal{G}$, $\mathcal{C}$, and $\mathcal{U}$ on the selection and stability of $\mathcal{F}$?

Q3: What is the relationship between $\mathcal{F}$ and $\mathcal{D}$ when $\mathcal{U}$ exists and $\mathcal{C}$ is diversified?

4 Experimental settings and design

To validate the proof of the concerns assumptions in Section 3, we designed an experimental based on the Iris dataset, 15 distance metrics, 11 feature possibilities, and 9 EC-based models. That means the concerns of each $\mathcal{G}$, $\mathcal{C}$, $\mathcal{D}$, $\mathcal{F}$, $\mathcal{M}$, and $\mathcal{U}$ have been considered. If the experiment utilized with other datasets which contain missing data, the imputation algorithms can be used to recover the data [24, 25]. Figure 4 presents three experimental designs, which are distance-based, feature-based, and features- and distance-based validations.

Distance validation: It is based on implementing the growing state on X using Equation 4, generating centroids possibilities from the fractioned $\mathcal{X}$ using Equation 5, and selecting the distance metric starting from **D1** to **D15**. Finally, these settings will be sent to every selected model $\mathcal{M}$ to evaluate its performance. This validation assists us in understanding the real impact of $\mathcal{G}$, $\mathcal{C}$, $\mathcal{D}$, $\mathcal{M}$, and $\mathcal{U}$ on each other, especially to detail the impact of growing and overlapping on the model performance and selecting the optimal $\mathcal{D}$.

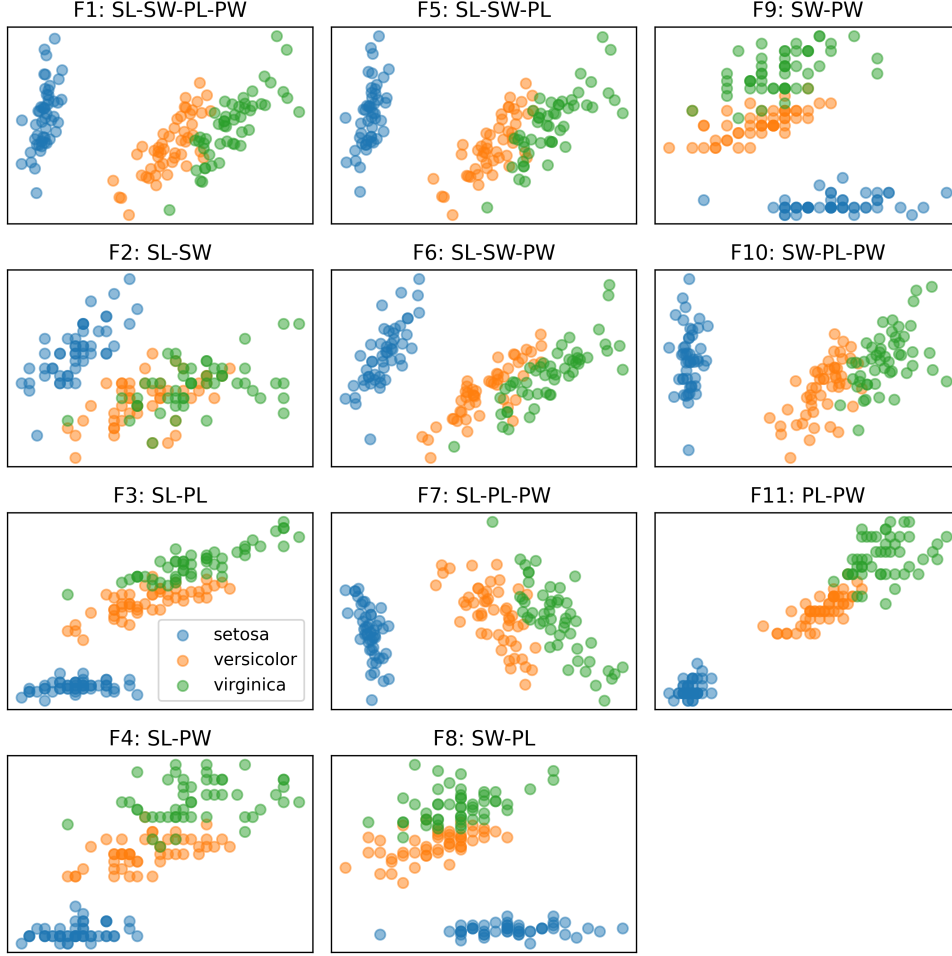


Fig. 3: Species samples of the Iris dataset

Features validation: This validation is similar to the distance-based validation, where the only difference is replacing the $\mathcal{D}$ with the feature possibilities. The purpose of this experiment is to know the challenges of selecting the optimal features in the unsupervised form and how growing and overlapping will impact the model performance and selecting the optimal $\mathcal{F}$.

Distance and feature validation: We ignored the growing state concern in this experiment part and focused on validating the impact of $\mathcal{C}$, $\mathcal{F}$, $\mathcal{D}$, $\mathcal{M}$, and $\mathcal{U}$ on each other. The steps start with the dataset X , predict centroids possibilities, then select each feature set and distance matrix, and finally evaluate every model. Further, this validation can assess the **overall validation** of both the unsupervised and semi-supervised versions of the models. The objective is to confirm the dependability and attainability of the optimal performance achieved.

We executed the experiment on a server with an Intel (R) Xeon (R) Gold 6248R CPU operating at 3 GHz, 384 GB of memory, and Nvidia T4 graphics on the Python platform, consuming over 1,250 hours of time. The outcomes of the experiment contain 271,181,250 clustering outcomes, with further details presented in Table 1. Finally, Appendix A presents the data distribution of various feature selection methods under different growing states.

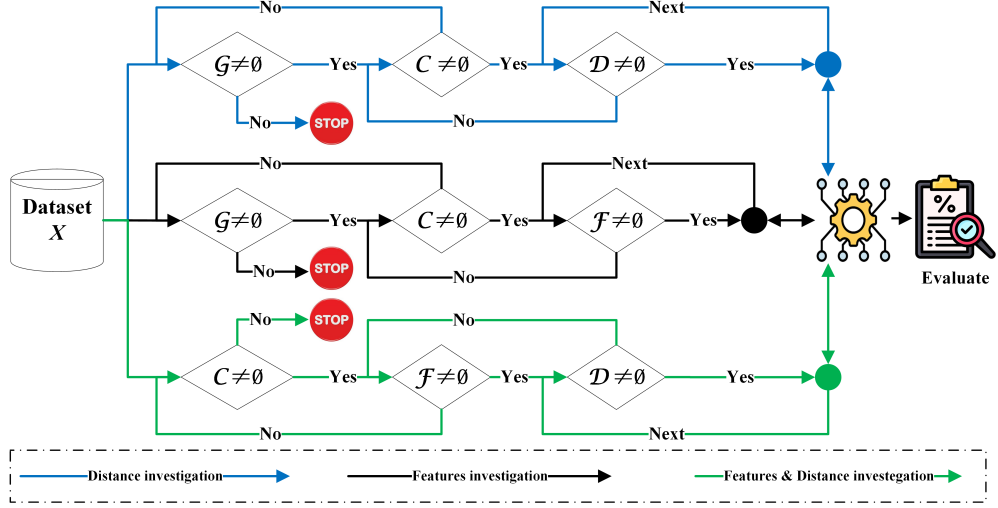


Fig. 4: Investigation process of this study

Table 1: Details of number of obtained outcomes by the experimental

Impacts	$\mathcal{G}$	$\mathcal{C}$	$\mathcal{D}$	$\mathcal{F}$	$\mathcal{M}$	Total runs $\mathcal{C} \times \mathcal{D} \times \mathcal{F} \times \mathcal{M}$
Distance	50%	15,625	15	-	9	2,109,375
	60%	27,000	15	-	9	3,645,000
	70%	42,875	15	-	9	5,788,125
	80%	64,000	15	-	9	8,640,000
	90%	91,125	15	-	9	12,301,875
	100%	125,000	15	-	9	16,875,000
Features	50%	15,625	-	11	9	1,546,875
	60%	27,000	-	11	9	2,673,000
	70%	42,875	-	11	9	4,244,625
	80%	64,000	-	11	9	6,336,000
	90%	91,125	-	11	9	9,021,375
	100%	125,000	-	11	9	12,375,000
Distance and Feature	100%	125,000	15	11	9	185,625,000
271,181,250						

5 Experimental results and discussion

5.1 Distance validation

In real-world applications, the growing state of data might impact both the performance of models and the correlation between the distance matrix and the models. Figure 5 shows the performance of various algorithms on different distance matrices under different data growing states. The results suggest that models need to select distinct distance matrixes for different statuses in order to consistently reach the best performance. This validates our concern that the performance of models is impacted by the growing state and that the distance matrix must be dynamically adjusted to optimize performance.

Figures 5c, 5d, and 5e demonstrate that the maximum performance approaches nearly 1.0 as the data increases to 60%. However, when the data surpasses 70%, the maximum performance experiences a substantial decline. This suggests that the performance of the model does not improve consistently during data growth. Conversely, increased data might result in a performance decline. Moreover, as shown in Figure 5g, the performance exhibits significant instability as the data grows to 60%. The performance decreases when the data reaches 70% and increases again once the data reaches 80%. A significant similarity or duplication between the newly added data and the existing data can cause performance instability and decline. Figure A1 assists in understanding the stability issue in the outcomes. We can see that the overlapping samples increased significantly when the data grew to 70%, which is unlike the data in its 80% growth where more non-overlapped samples were added. This variation in the proportion of overlapping samples in the data-growing state corresponds to the observed fluctuations in performance.

Table 2: Investigating the compatibility of distance matrix with every model each model

Distance matrix	Models								
	DP	AP	MKMDA	MKMDO	MKM	KFP	KPF1	KFRM	KFM
D1	0	0	0.67	0.67	0.67	0	0	0	0
D2	0.17	0	0.17	0.17	0.17	0	0	0	0
D3	0.17	0	0	0	0	1.00	0	0.67	0
D4	0	0	0.17	0.17	0.17	0	0	0.17	0.50
D5	0	0	0	0	0	0	0.33	0	0.17
D6	0.33	0	0.50	0.50	0.83	0.33	0.83	0	0
D7	0	1.00	0.83	0.67	0.83	0	0	0	0
D8	0	0	0	0	0	0	0	0	0
D9	0	0	0	0	0	0	0	0	0
D10	1.00	0	0.17	0.17	0.17	0.17	0.50	0	0
D11	0	0	0	0	0	0	0	0.17	0.50
D12	0	0	0	0	0	0	0	0	0
D13	0	1.00	0.83	0.67	0.83	0	0	0	0
D14	0	0	0.33	0.33	0.33	0	0	0	0
D15	0	0	0.67	0.67	0.83	0	0	0	0

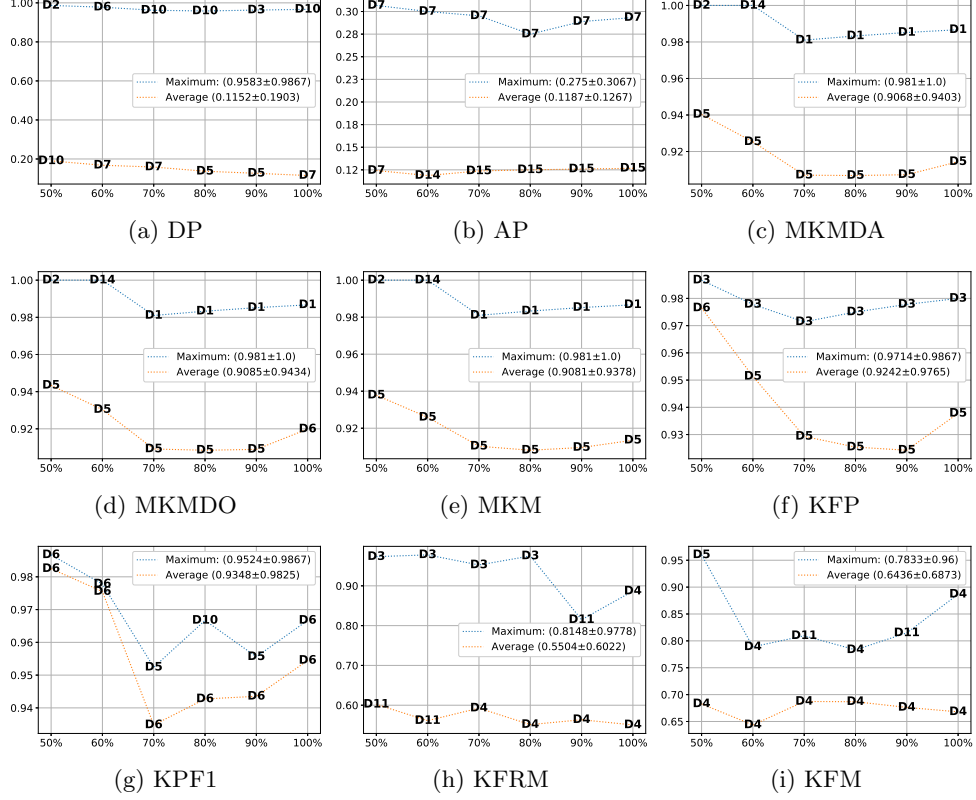


Fig. 5: Investigating the impact of growing data on the chosen distance matrix

Considering the gap between maximum and average performance is crucial for comprehending the model’s sensitivity to parameters. A wider gap indicates that the model is more sensitive to parameters $\mathcal{C}$, $\mathcal{D}$, $\mathcal{D}$, and $\mathcal{U}$. Conversely, a narrower gap signifies lower sensitivity. Moreover, the model’s sensitivity has a direct impact on the difficulty of achieving ideal performance. Models with high sensitivity respond strongly to parameter changes, making them more challenging to fine-tune and optimize in practical applications. Specifically, Figure 5a appears to be difficult to achieve optimal performance, whereas Figure 5g exhibits low sensitivity.

Table 2 displays the proportion of times each distance matrix achieved the best performance under six growing states with every model. If the difference between the model and the distance matrix is less than 1.00, it indicates that they should not be associated, as the distance matrix may not be the optimal choice in every growing state. To clarify this, as shown in Figure 5d, the MKMDO model has obtained the highest maximum and average performance; however, it demands changing the distance matrix in several growing states.

The presented results in Figure 5 and Table 2 prove that the growing state of data would contain overlapping samples, which increases the demand for re-determining

the distance matrix and parameters to ensure the stability of obtaining optimal performance. This proves our concern and raises the challenges of addressing the issues of incorporation partitioning and false confidence.

5.2 Features validation

Figure 6 demonstrates that the DP and KFRM models have achieved the highest level of stability in terms of performance. Nevertheless, achieving stability in the feature settings necessitates the demanding task of readjustment, which involves retraining and evaluating the model’s outcomes. If there is a request for altering the features, it may be necessary to also adjust the centroids, as different feature settings can result in different clustering structures. By examining F7, F10, and F11 in Figure 3, it becomes evident that the data have distinct shapes. Still, each of the models has achieved a performance of over 90%, indicating that certain models like DP and KFRM aren’t impacted by the data shape. Based on this validation, we can conclude that if the model consistently produces stable outcomes using different features, then the model is robust in terms of data shapes. Moreover, this conclusion could be used only in supervised validation, where it is very challenging to implement unsupervised models.

The MKMDA, MKMDO, and MKM models achieved a perfect performance score of 1.0 in the 50% growth state. However, their performance begins to decline once it reaches a 70% growth state. By comparing F1 in Figure A1 and F10 in Figure A10 at their 50% growth state, it is evident that there are only two samples that overlap, which are dependent on the Versicolor and Virginica species. The MKMDA, MKMDO, and MKM models effectively clustered the samples into their respective classes. However, these models encounter difficulties when there are a higher number of overlapping samples in the growing state. Therefore, models that can handle overlapping samples must take into account a diverse range of overlapped samples to demonstrate their ability to address these challenges.

The average accuracy of the DP model, ranging from 0.1168 to 0.1828, reflects the challenges in achieving optimal results from a total of 4,021,875 runs, as calculated from Table 1. In contrast to KFP and KFP1, which had an average of approximately 0.8957 ± 0.9397 . The average performance in Figures 6a-6i can refer to the complexity of achieving the optimal outcomes.

The presented results in Figure 6 prove that most of the models have suffered from feature demands and stability in each growing state. To clarify that, Table 3 represents the model features demand to reach the maximum performance, where the 1.00 value refers to the demand of reaching the maximum performance with the same features set. Thus, MKMDA with F1, MKMDO with F1, and KFM with F11 have recorded higher performance stability. However, DP, AP, and other models didn’t reach feature stability, where they required different feature settings to obtain optimal solutions; AP demanded F1, F7, and F10 to obtain the whole thing. Finally, unsupervised models would make it much more difficult to make the correct feature selections for further growing states, specific centroids, and distance matrix.

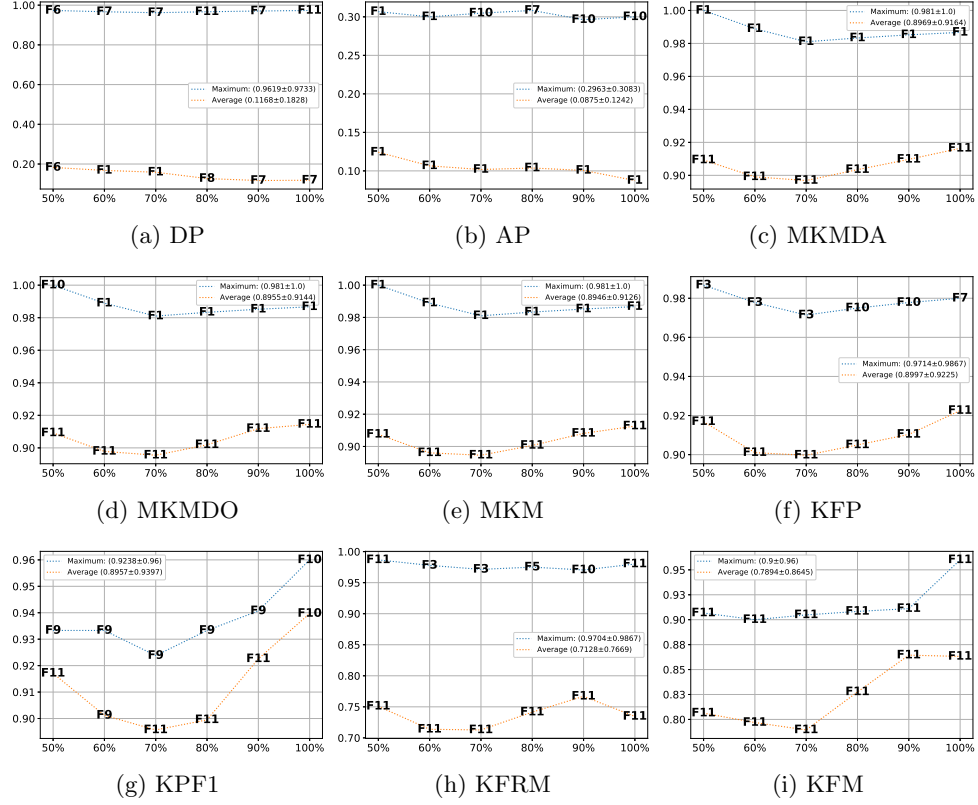


Fig. 6: Investigating the impact of growing data on the selected features

Table 3: Investigating the compatibility of features set with every model each model

Selected Features	Models									
	DP	AP	MKMDA	MKMDO	MKM	KFP	KPF1	KFRM	KFM	
F1	0	0.33	1.00	0.83	1.00	0	0	0	0	
F2	0	0	0	0	0	0	0	0	0	
F3	0	0	0	0	0	0.50	0	0.33	0	
F4	0	0	0	0	0	0	0	0	0	
F5	0	0	0	0	0	0.17	0	0.17	0	
F6	0.17	0	0	0	0	0	0	0	0	
F7	0.50	0.17	0.17	0.50	0.50	0.17	0	0	0	
F8	0	0	0	0	0	0	0	0	0	
F9	0	0	0	0	0	0.17	0.83	0	0	
F10	0	0.50	0.33	0.33	0.33	0.50	0.17	0.17	0	
F11	0.67	0	0	0	0	0.33	0.50	0.50	1.00	

5.3 Features and distance validation

This section aims to simultaneously determine the actual effects on both features and distance. Table 4 displays the highest and mean accuracy achieved by each individual

model exclusively in the 100% growing state. Therefore, it is evident that the highest level of performance achieved by all models was below 0.9867. Consequently, even when all possible configurations are considered, the models consistently make incorrect predictions due to the problem of overlapping between data samples. As an example, the values for F1 and D1 are 0.9867 and 0.8735, respectively. This result resulted from the execution of all nine models, specifically focusing on the F1 and D1 metrics. The expression $125,000\mathcal{C} \times 9\mathcal{M}$ results in 1,125,000 Λ^p outcomes. These outcomes represent both the average and maximum values obtained. This demonstrates the challenge of achieving satisfactory results.

Table 4 provides valuable insights into the optimal feature sets for the Iris dataset. It also helps scholars identify the most suitable distance matrix settings to rely on for their development. Therefore, it is recommended to use the D15 and F1 settings for the Iris dataset when considering it for development. From an alternative standpoint, the table showcases the model’s proficiency in handling diverse data structures, along with the appropriately aligned distance matrix that effectively addresses the problem identified in the Iris dataset.

As this section contains over 185,625,000 outcomes, we implemented an overall analysis to count and display the best, not the highest, clustering results for the 100% growing state of the Iris dataset. We considered the unique performance and counted its duplication time, where the minimum performance is above 0.8. Figure 7 tells us that the *reachable* performance was 0.9467, which can be predicted with most CA models. However, the possibility of reaching 0.9467 is only 2%, where reaching 0.9867 is 8.4E-06% only. The results prove that clustering models always result in inconsistent partitioning and false-confident outcomes.

Since these results are achieved through a supervised approach, it raises two questions: Can these models achieve similar or better performance in their unsupervised form without the provision of any information about Λ^g ? And how can we ensure the stability of the models in real-world applications?

5.4 Overall validation

It is important to recognize the consequences of a state that is expanding and may have overlapping samples over time. This expansion may require the reevaluation of centroids, which in turn necessitates the selection of a new distance matrix and features. These impacts can cause various problems in the model’s performance over time, resulting in the inclusion of partitioning and inaccurate confidence in the outcomes.

In order to answer the questions in Section 5.3, it is necessary to compare the implemented models in their unsupervised form with the actual ability they achieved in the supervised settings. This analysis will disclose the possibilities achieved and the underlying factors that led to the highest performances. Table 5 discussed the comparisons between the unsupervised and supervised forms. The unsupervised form has 10,000 attempts, while the supervised form contains 20,625,000 attempts.

Table 5 displays that UDP and SDP can reach the optimal solution. However, the success rate (SR) of the optimal performance is almost impossible for each UDP and SDP. Integrating the supervised settings with the AP model results in performance dropping. Thus, the model is considered to have the lowest performance compared

Table 4: Investigating the impact of distance and features on each other

$\mathcal{D}$		Features											
		F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	AVG
D1	M	0.9867	0.8267	0.9733	0.9667	0.9733	0.9733	0.9867	0.9600	0.9667	0.9800	0.9733	0.9581
	A	0.8735	0.6651	0.8402	0.8163	0.8344	0.8125	0.8850	0.8735	0.8637	0.9076	0.9224	0.8412
D2	M	0.9800	0.8200	0.9733	0.9667	0.9733	0.9733	0.9733	0.9600	0.9667	0.9867	0.9733	0.9575
	A	0.9019	0.6535	0.8578	0.8946	0.8325	0.8782	0.9120	0.8508	0.8870	0.9150	0.9241	0.8638
D3	M	0.9800	0.8333	0.9600	0.9467	0.9600	0.9533	0.9800	0.9533	0.9533	0.9800	0.9733	0.9566
	A	0.8600	0.6865	0.8595	0.7787	0.8536	0.7821	0.8650	0.8851	0.8644	0.8963	0.9006	0.8320
D4	M	0.9800	0.8267	0.9667	0.9667	0.9667	0.9667	0.9800	0.9600	0.9667	0.9867	0.9733	0.9592
	A	0.8695	0.6636	0.8323	0.8149	0.8287	0.8098	0.8736	0.8698	0.8680	0.9059	0.9201	0.8376
D5	M	0.9800	0.3000	0.3000	0.3000	0.9533	0.9200	0.9267	0.6467	0.3000	0.9533	0.3000	0.6655
	A	0.9144	0.3000	0.3000	0.3000	0.9030	0.8328	0.8539	0.6467	0.3000	0.9029	0.3000	0.6063
D6	M	0.9867	0.7667	0.9467	0.9200	0.9600	0.9533	0.9733	0.9000	0.9467	0.9600	0.8267	0.9236
	A	0.9138	0.6571	0.8850	0.8658	0.8812	0.8624	0.9070	0.8225	0.9155	0.8891	0.6831	0.8402
D7	M	0.9867	0.8333	0.9733	0.9667	0.9800	0.9667	0.9867	0.9600	0.9667	0.9800	0.9800	0.9647
	A	0.8732	0.6798	0.8505	0.8029	0.8451	0.8046	0.8770	0.8847	0.8725	0.9085	0.9164	0.8415
D8	M	0.6267	0.6000	0.6067	0.6200	0.6067	0.6133	0.6200	0.5933	0.6400	0.6333	0.6867	0.6225
	A	0.4839	0.3905	0.4124	0.4492	0.4153	0.4590	0.4714	0.4031	0.4457	0.4697	0.4594	0.4418
D9	M	0.6267	0.6000	0.6067	0.6200	0.6067	0.6133	0.6200	0.5933	0.6400	0.6333	0.6867	0.6225
	A	0.4839	0.3905	0.4124	0.4492	0.4153	0.4590	0.4714	0.4031	0.4457	0.4697	0.4594	0.4418
D10	M	0.9800	0.7600	0.9333	0.9200	0.9600	0.9533	0.9733	0.9000	0.9467	0.9600	0.8267	0.9204
	A	0.9120	0.3710	0.4868	0.8110	0.8633	0.8904	0.8925	0.4686	0.2514	0.9004	0.5376	0.6831
D11	M	0.9400	0.8333	0.9733	0.9667	0.9400	0.9600	0.9667	0.9533	0.9667	0.9400	0.9600	0.9554
	A	0.6896	0.6762	0.6995	0.7251	0.6940	0.7240	0.7230	0.7366	0.7626	0.7460	0.7782	0.7214
D12	M	0.6267	0.6000	0.6067	0.6200	0.6067	0.6133	0.6200	0.5933	0.6400	0.6333	0.6867	0.6225
	A	0.4839	0.3905	0.4124	0.4492	0.4153	0.4590	0.4714	0.4031	0.4457	0.4697	0.4594	0.4418
D13	M	0.9867	0.8333	0.9733	0.9667	0.9800	0.9667	0.9867	0.9600	0.9667	0.9800	0.9800	0.9704
	A	0.8732	0.6798	0.8505	0.8029	0.8451	0.8046	0.8770	0.8847	0.8725	0.9085	0.9164	0.8415
D14	M	0.9800	0.8333	0.9600	0.9667	0.9667	0.9667	0.9867	0.9667	0.9667	0.9867	0.9800	0.9673
	A	0.8123	0.6864	0.7745	0.8110	0.7519	0.7846	0.8419	0.7533	0.7853	0.8400	0.9313	0.7950
D15	M	0.9867	0.8333	0.9733	0.9667	0.9800	0.9667	0.9867	0.9600	0.9667	0.9800	0.9800	0.9704
	A	0.8746	0.6789	0.8525	0.8018	0.8461	0.8048	0.8771	0.8857	0.8684	0.9070	0.9157	0.8418
AVG	M	0.8898	0.7296	0.8271	0.8223	0.8745	0.8545	0.8775	0.8071	0.8235	0.8813	0.8292	
	A	0.7845	0.5625	0.6260	0.6663	0.6823	0.6720	0.7144	0.6863	0.6808	0.7426	0.7314	

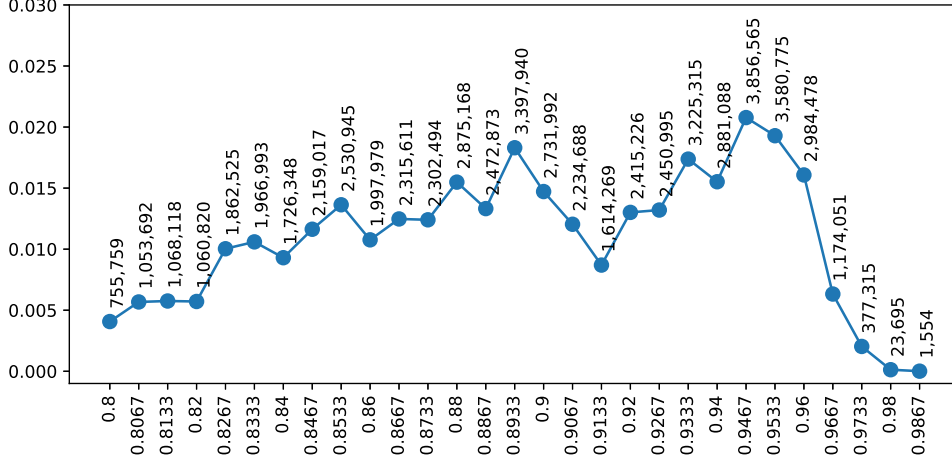


Fig. 7: The repetition ratio of unique accuracies above 0.8 that obtained by all the models with every possibilities on full data only, which are in total 185,625,000 runs

with the other models. Further, other models have not been able to achieve the highest performance in their unsupervised form, especially KFRM and KFM. Finally, the gap values can explain the complexity of reaching the optimal solutions based on the unsupervised form.

Table 5: Compare unsupervised and supervised forms of each model

Methods	Unsupervised		Supervised		Gap
	$\max(U)$ ¹	SR(U) ²	$\max(S)$ ³	SR(S) ⁴	
DP	0.9733	0.0001	0.9733	1.5e-07	= 0
AP	0.6333	1.00	0.42	4.4e-04	↓ -0.2133
MKMDA	0.9733	0.0001	0.9867	2.8e-05	↑ 0.0134
MKMDO	0.9467	0.0004	0.9867	1.8e-05	↑ 0.04
MKM	0.9667	0.0051	0.9867	2.9e-05	↑ 0.02
KFP	0.9067	0.6941	0.98	5.7e-05	↑ 0.0733
KPF1	0.9067	0.4217	0.9667	1.5e-02	↑ 0.6
KFRM	0.7333	0.0818	0.98	5.6e-06	↑ 0.2467
KFM	0.7333	0.4248	0.9667	7.5e-04	↑ 0.2334

¹The max accuracy from 10,000 unsupervised runs for each model

²The success rate of unsupervised-based max accuracy

³The max accuracy from 20,625,000 unsupervised runs for each model

⁴The success rate of supervised-based max accuracy

Additionally, the shape-based clustering concern needs to be discussed, which is one of the issues that impact performance. Assume that every feature set F1 to F11 is a separate dataset. As the clustering models proposed to solve a certain issue in a certain shape, the initial growing state of the data is nearly separated. In the next

growing state, the data samples are moving closer to each other. This means that if the Iris data contains more samples, it could overlap with every class. Figure A11 proves that Versicolor and Virginica already have an overlapping issue; however, the Versicolor class is growing toward the Setosa class. This demonstrates that clustering should not rely solely on shape-based clustering, as a growing state could provide a different perspective and potentially cause issues with the outcomes. Figure A2 proves this concern: there is no way to cluster the data based on shape because the two classes are overlapped, and they even become overlapped with the Setosa class in its 100% growing state.

We can conclude the outcomes of this study as follows: 1) The “no-free lunch” doesn’t match with the unsupervised development due to the complexity of finding the parameters. 2) A clustering model can be successful with a specific dataset and a specific number of samples. Thus, if a growing state happens, then it would impact the performances, which demand adjustments of $\mathcal{C}$, $\mathcal{D}$, and $\mathcal{F}$. This happens due to the alteration in the state space as well as the possibility of the appearance of overlapped samples. 3) The incorporation partitioning occurred due to the centroids and the existence of overlapped samples. However, various centroids discover various data structures, relations, and interpretations, which proves there is no preference between the EC-based models. 4) The optimal outcomes can be produced if overlapped samples are considered centroids, which is very challenging. 5) Finally, the presented results in this work prove that obtaining these optimal solutions based on unsupervised EC is as challenging as finding a needle in a haystack.

5.5 Open Issue

To prove all the concerns and discussions above, we have suggested an open issue that challenges the clustering models to obtain higher performance in an unsupervised form. We generated a dataset that contains 4,000 RGB images sized 224×224 from ImageNet 10K [26]. The dataset contains three classes; three of them were attacked by various adversarial attack models, and one was on clean images. The adversarial attack models are AdvLB [27], AdvLS [28], and AdvXL [29], respectively.

The reason to create this open issue is because the implemented perturbation is very low and would generate a serious overlapping issue because the same images in the “Clean” class have been attacked by these three adversarial attack models. The dataset challenges clustering models with its complexity and the high ratio of overlapping existence, which easily leads to false confidence and non-sense interpretation in the outcomes of unsupervised clustering models.

We provided a CSV file that contains three columns: labels, image path, and growing state. The growing state contains six rules, which are “initial state” and growing state 1–5. The authors are required to train, fine-tune, or develop their **unsupervised clustering model** based on the “initial state” of the growing state. Thus, validate the model on each of the growing states 1 to 5. The success measure of this open issue depends on getting the same performance in each growing state based on the same parameters and model that developed in the “initial state”. The aim of this open issue

is because we need to prove that our concerns make sense as well as that the “no-free lunch” doesn’t fit the concept of developing an unsupervised model with lots of parameters.

6 Conclusion

In this study, we examined the complex relationships between clustering issues, particularly focusing on the challenges posed by overlapping samples and how growing states influence the model’s performance. Our findings emphasize that the “no-free-lunch” theorem is inapplicable with unsupervised clustering models, as the complexity of parameter selection and settings significantly impacts model outcomes. Our experiments have shown that while some clustering models can achieve optimal performance under certain conditions, their effectiveness diminishes as the data grows and overlaps increase. This requires constant adjustment of parameters such as centroids, distance metrics, and features to keep performance stable. In summary, our research offers an in-depth awareness of the factors that impact the effectiveness of clustering models and emphasizes the need for adaptable and strategic clustering approaches in response to growing and overlapping data. Further studies should focus on the development of models capable of flexibly adjusting to these challenges, thereby ensuring reliable and realistic clustering outcomes. In our future work, we are going to focus on developing a non-parametric clustering model that can solve the open issue.

Acknowledgements

This research work was partly supported by National Natural Science Foundation of China under grant No. 62372085, 62376055, 62276053.

References

- [1] Ran X, Xi Y, Lu Y, Wang X, Lu Z. Comprehensive survey on hierarchical clustering algorithms and the recent developments. *Artificial Intelligence Review*. 2023 August;56(8):8219–8264. <https://doi.org/10.1007/s10462-022-10366-3>.
- [2] Taha K. Semi-supervised and un-supervised clustering: A review and experimental evaluation. *Information Systems*. 2023;114:102178.
- [3] Ikotun AM, Ezugwu AE, 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*. 2023;622:178–210. <https://doi.org/10.1016/j.ins.2022.11.139>.
- [4] Kadhimi MR, Tian L, Zheng X, Kang Z, Shi Y, Wang J. β Equation: Toward clustering the overlap data. Available at SSRN. 2024;.

- [5] Golalipour K, Akbari E, Hamidi SS, Lee M, Enayatifar R. From clustering to clustering ensemble selection: A review. *Engineering Applications of Artificial Intelligence*. 2021;104:104388.
- [6] Liang B, JiYe L, Fuyuan C. Semi-supervised clustering with constraints of different types from multiple information sources. *IEEE Transactions on pattern analysis and machine intelligence*. 2020;43(9):3247–3258.
- [7] Mustafa K, Qingyuan L, Jianbo W, Kui W, Xu Z, Zhao K, et al. Gravity Sub-centroids for Optimal Clustering. *Authorea Preprints*. 2023;.
- [8] Mustafa K, Guangyao Z, Wenhong T. A novel self-directed learning framework for cluster ensemble. *Journal of King Saud University - Computer and Information Sciences*. 2022;34:7841–7855.
- [9] He Q, Chen Z, Ji K, Wang L, Ma K, Zhao C, et al. Cluster Center Initialization and Outlier Detection Based on Distance and Density for the K-Means Algorithm. In: Abraham A, Cherukuri AK, Melin P, Gandhi N, editors. *Intelligent Systems Design and Applications*. Cham: Springer International Publishing; 2020. p. 530–539.
- [10] Zang W, Liu X, Ma L, Che J, Sun M, Zhao Y, et al. Density peaks clustering based on superior nodes and fuzzy correlation. *Information Sciences*. 2024;672:120685. <https://doi.org/10.1016/j.ins.2024.120685>.
- [11] An overview on density peaks clustering. *Neurocomputing*. 2023;554:126633. <https://doi.org/10.1016/j.neucom.2023.126633>.
- [12] Rashki M, Faes MGR. No-Free-Lunch Theorems for Reliability Analysis. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*. 2023;9(3):04023019. <https://doi.org/10.1061/AJRUA6.RUENG-1015>.
- [13] Schurz G, Thorn P. Escaping the No Free Lunch Theorem: A Priori Advantages of Regret-Based Meta-Induction. *Journal of Experimental & Theoretical Artificial Intelligence*. 2024;36(1):87–119. <https://doi.org/10.1080/0952813X.2022.2080278>.
- [14] Frey BJ, Dueck D. Clustering by Passing Messages Between Data Points. *Science*. 2007;315(5814):972–976. <https://doi.org/10.1126/science.1136800>.
- [15] Tran V, Kammermann M, Baumann P. The MPFCC Algorithm: A Model-Based Approach for Fair-Capacitated Clustering. In: *2023 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. IEEE; 2023. p. 0677–0681.

- [16] Schubert E, Rousseeuw PJ. Fast and eager k-medoids clustering: $O(k)$ runtime improvement of the PAM, CLARA, and CLARANS algorithms. *Information Systems*. 2021;101:101804. <https://doi.org/10.1016/j.is.2021.101804>.
- [17] Schubert E, Rousseeuw PJ. Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms. In: Amato G, Gennaro C, Oria V, Radovanović M, editors. *Similarity Search and Applications*. Cham: Springer International Publishing; 2019. p. 171–187.
- [18] Lenssen L, Schubert E. Clustering by Direct Optimization of the Medoid Silhouette. In: Skopal T, Falchi F, Lokoč J, Sapino ML, Bartolini I, Patella M, editors. *Similarity Search and Applications*. Cham: Springer International Publishing; 2022. p. 190–204.
- [19] Bhattacharjee P, Mitra P. A survey of density based clustering algorithms. *Frontiers of Computer Science*. 2021;15:1–27.
- [20] Solorio-Fernández S, Carrasco-Ochoa JA, Martínez-Trinidad JF. A review of unsupervised feature selection methods. *Artificial Intelligence Review*. 2020;53(2):907–948.
- [21] Jeon H, Oh S. Hybrid-recursive feature elimination for efficient feature selection. *Applied Sciences*. 2020;10(9):3211.
- [22] Aggrawal R, Pal S. Sequential feature selection and machine learning algorithm-based patient’s death events prediction and diagnosis in heart disease. *SN Computer Science*. 2020;1(6):344.
- [23] García-Moya M, Polo-Blanco I, Blanco R, Goñi-Cervera J. Teaching Cartesian product problem solving to students with autism spectrum disorder using a conceptual model-based approach. *Focus on Autism and Other Developmental Disabilities*. 2023;38(4):245–257.
- [24] Adhikari D, Jiang W, Zhan J, He Z, Rawat DB, Aickelin U, et al. A Comprehensive Survey on Imputation of Missing Data in Internet of Things. *ACM Comput Surv*. 2022 dec;55(7). <https://doi.org/10.1145/3533381>.
- [25] Adhikari D, Jiang W, Zhan J, Assefa M, Khorshidi HA, Aickelin U, et al. A Lightweight-Window-Portion-Based Multiple Imputation for Extreme Missing Gaps in IoT Systems. *IEEE Internet of Things Journal*. 2024;11(3):3676–3689. <https://doi.org/10.1109/JIOT.2023.3315137>.
- [26] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*. 2015;115:211–252.

- [27] Ranjie D, Xiaofeng M, K QA, Yuefeng C, Shaokai Y, Yuan H, et al. Adversarial Laser Beam: Effective Physical-World Attack to DNNs in a Blink. In: Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021. p. 16057–16066.
- [28] Chengyin H, Yilong W, Kalibinuer T, Wen L. Adversarial laser spot: Robust and covert physical-world attack to dnns. In: Proceedings of Asian Conference on Machine Learning. PMLR; 2023. p. 483–498.
- [29] Hanyu Wu PXBH Ke Yan, Tian L. Adversarial Cross-laser Attack: Stable and Effective Attack to DNNs in the Real World. In: Proceedings of the 29th Australasian Conference on Information Security and Privacy; 2024. p. accepted.

Appendix A Supplementary Information

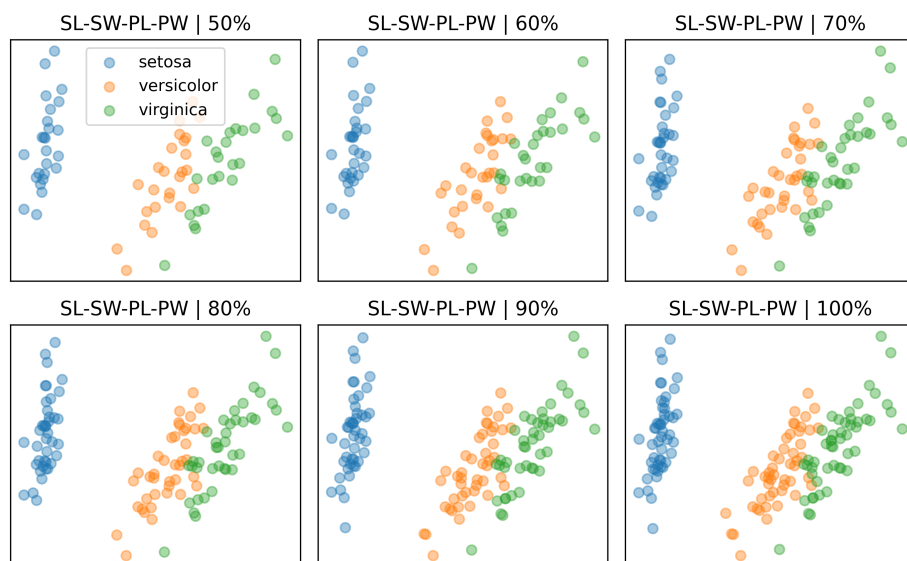


Fig. A1: Visualize F1 features set in different growing status

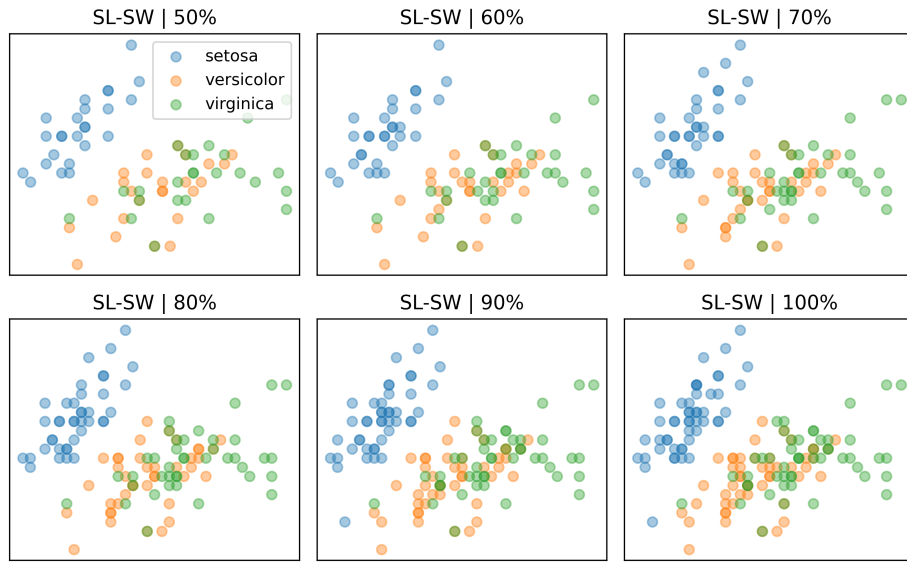


Fig. A2: Visualize F2 features set in different growing status

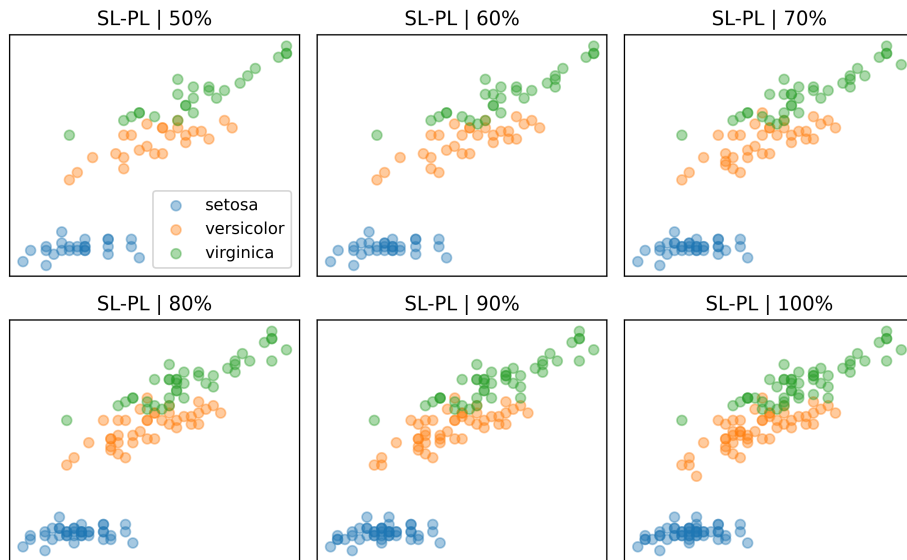


Fig. A3: Visualize F3 features set in different growing status

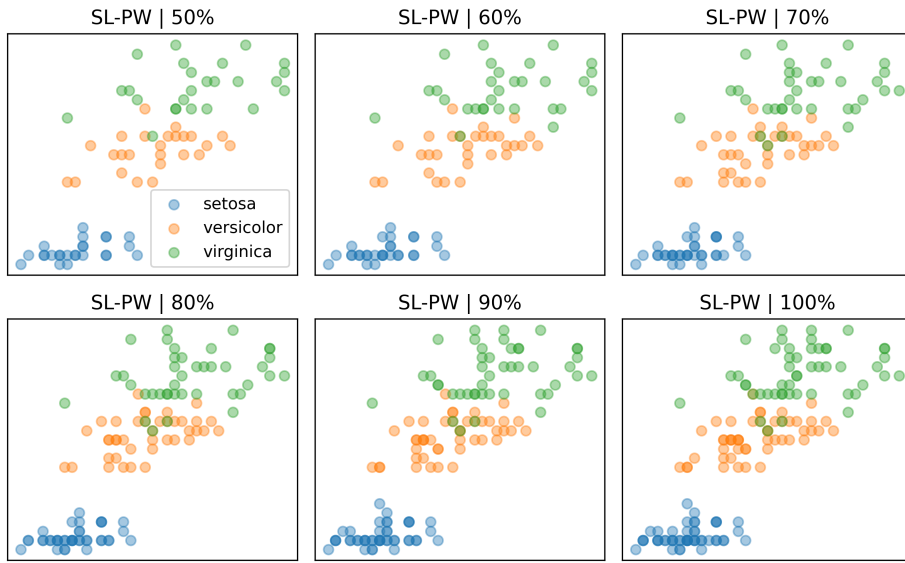


Fig. A4: Visualize F4 features set in different growing status

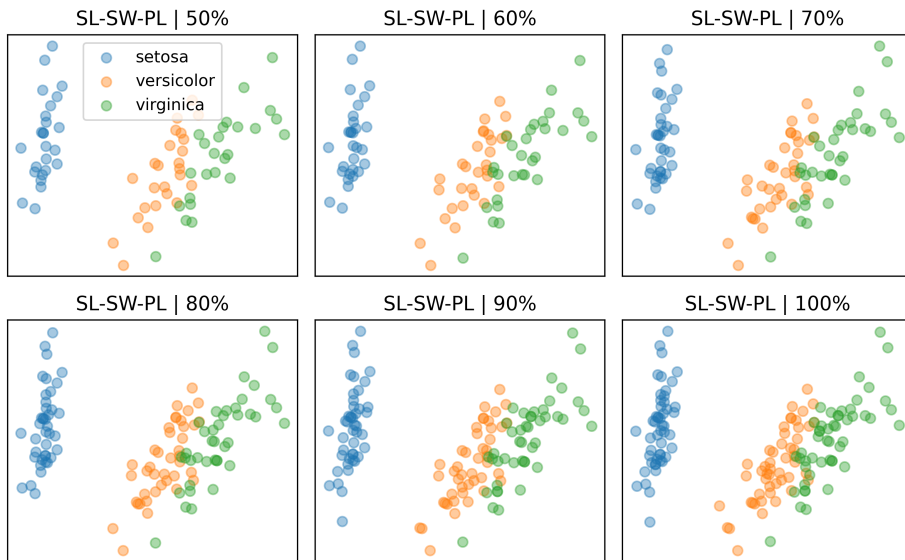


Fig. A5: Visualize F5 features set in different growing status

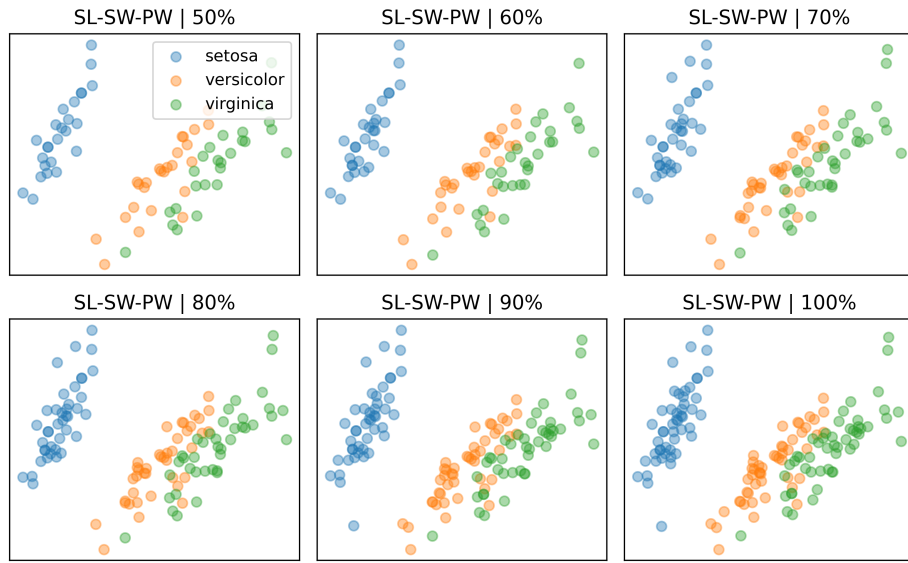


Fig. A6: Visualize F6 features set in different growing status

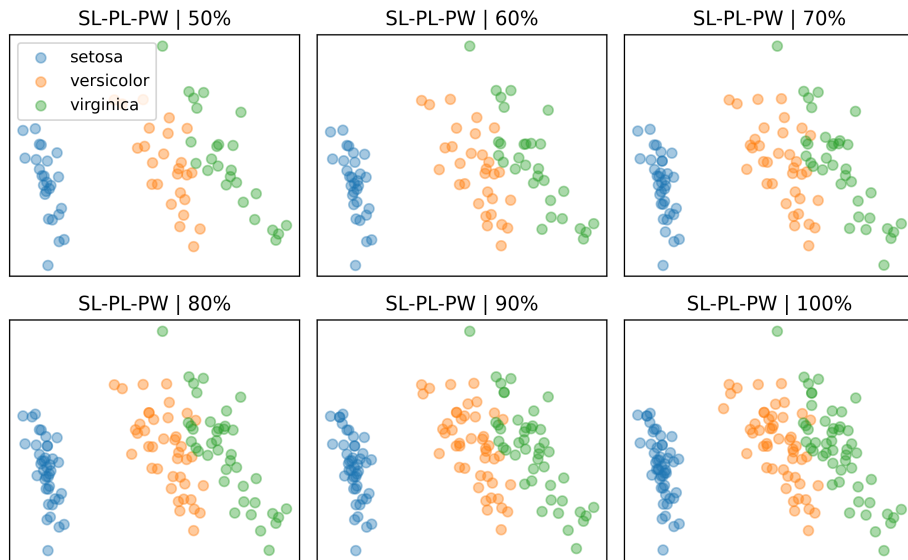


Fig. A7: Visualize F7 features set in different growing status



Fig. A8: Visualize F8 features set in different growing status

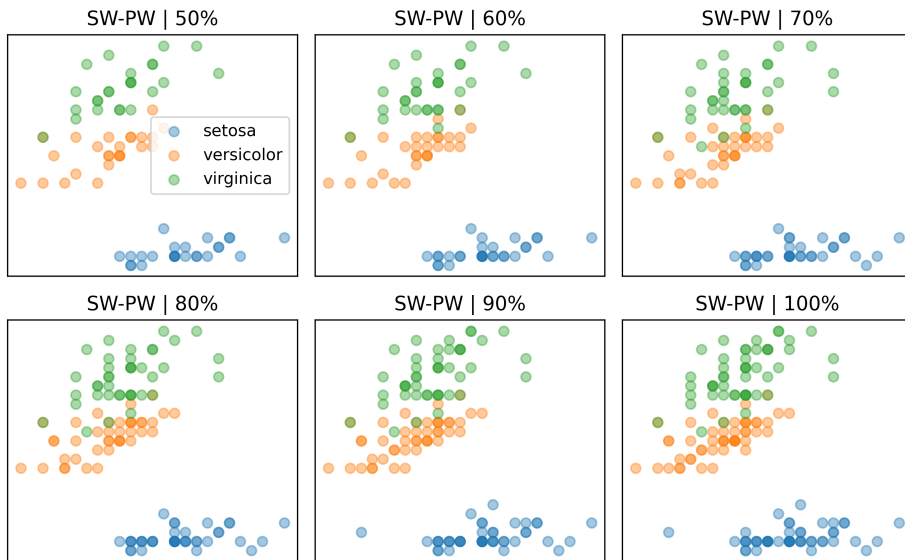


Fig. A9: Visualize F9 features set in different growing status



Fig. A10: Visualize F10 features set in different growing status

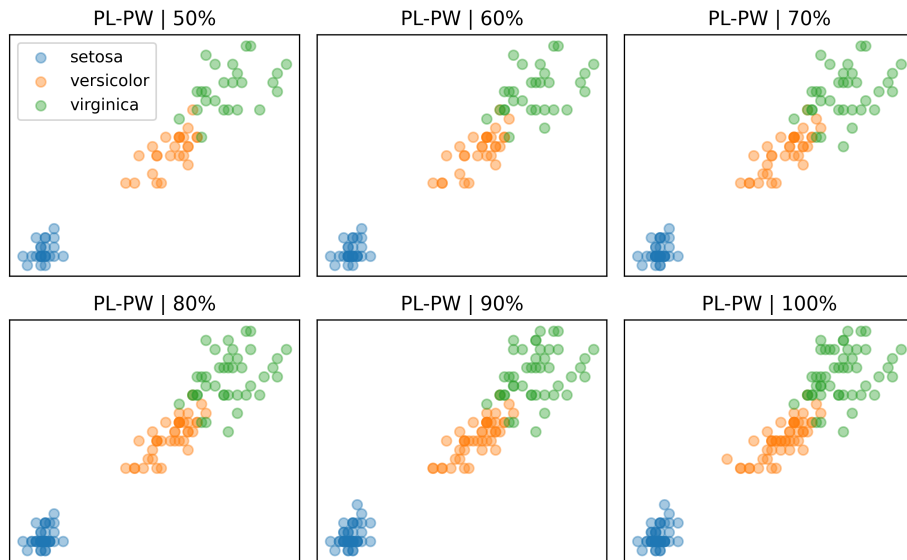


Fig. A11: Visualize F11 features set in different growing status

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Supplementarymaterial.zip](#)