Comparative analysis based on clustering algorithms

Junfei Gu1,*
1School of The University of Sydney, Sydney, NSW 2006 (post), Australia.
*Corresponding author e-mail: jugu0751@uni.sydney.edu.au

Abstract. This article summarizes and evaluates the clustering effects of commonly used clustering algorithms on data sets with different density distributions. In this paper, circled datasets, different sized datasets, and Gaussian mixture datasets were designed as the typical datasets. Then, the K-means, Gaussian mixture clustering, DBSCAN, and Agglomerative clustering were developed to evaluate the clustering performance on these datasets. The results show that the DBSCAN is more stable when the density distributions of the data sets are not clear. Besides, the Agglomerative clustering that calculates the shortest distance can determine the type of data set. Moreover, it is not appropriate to use only a single clustering algorithm to analyze a Gaussian mixture dataset. It is recommended to use multiple clusters to process the dataset after preprocessing.

1. Introduction
In the data science era, almost every field has completed data automation to achieve high-speed operation of daily life and work [1]. To effectively explore and understand the data, unsupervised machine learning analyses the data by knowing only the domain and data without labelling training samples. These unlabeled samples contain the spatial distribution and structural features of unknown data, such as features such as volume, density, shape, and direction [1]. As a form of unsupervised learning, clustering is a grouping method by extracting the "similarity" features of physical or abstract objects [2]. The similarity feature or measurement method between these objects is usually a distance or density function. All in all, the clustering algorithm has a solid theoretical foundation and has become a complex research object in different research communities [3,4]. The most common algorithms are k-means, Gaussian mixture models [5,6], DBSCAN, and the hierarchical cluster method.

The K-means is a partition-based method that attempts to minimize the sum of the squared Euclidean distances of the average of each group as the objective function to pursue homogeneity within clusters and qualitative differences between clusters [7]. This algorithm has been widely used to classify the cloud top pressure optical thickness joint histogram to obtain the cloud system [8]. For example, K-means classification combined with IR information can be applied to grid data. This provides convenience for continuous image analysis of weather and sub-weather processes on a pixel-by-pixel basis [9].

The Gaussian mixture model is a Model-based method. Using a Gaussian mixture model to perform cluster analysis on data is similar to the K-means goal of maximizing the similarity of observations in a shared space [2]. However, the key difference compared with K-means is that the cluster assignment probability in the Gaussian mixture model is definite [10]. Observations with similar probabilities belonging to the same component (or "cluster") are grouped but with a certain degree of uncertainty. Another difference is the existence and use of probabilities, which makes most hybrid models Become
parameterized. In this way, the Expectation-Maximization algorithm is an iterative algorithm, most commonly used to estimate parameters, limiting the clustering process [10].

As one of the most famous Density-based methods, the DBSCAN uses the scan radius (eps) and the minimum number of points (MinPts) to find neighbours or obtain the density of each data point. For specific application scenarios such as user communities linked by friends in social networks, the ecosystem can be displayed as regions with homogeneous feature values in satellite images [11]. Therefore, the DBSCAN is widely utilized in many fields, such as aviation. The DBSCAN could analyze and identify unknown events in conventional flight data without prior knowledge of historical flight data [12]. In the biological field, the DBSCAN plays a role in detecting protein structure [13]. Besides, the early warning method based on DBSCAN clustering can effectively predict oil spill accidents in the energy field [14].

The hierarchical method establishes an attractive tree structure, making the algorithm more explanatory and persuasive for data partitioning [15]. Among them, Agglomerative clustering is a bottom-up hierarchical algorithm. The principle is to search for objects and clusters starting from the initial partition, where each object corresponds to a group [16]. The algorithm merges the two most similar sets in the simplest form. This process continues until the algorithm reaches the partition. Therefore, it is a method often chosen in exploratory analysis. For example, through the integration of diversified investment portfolios, using the agglomeration clustering model to trade stocks has a higher rate of return [17].

Since these four different clustering methods are widely used in different fields, this article will discuss their clustering performance-based three data sets with different density distribution. This article reported what types of data sets are suitable for different clustering algorithms, including the accuracy of these algorithms, cohesion, and the separation of different data sets. It provides a reference for future researchers to choose to cluster in various fields.

2. Method

2.1. Clustering Algorithms

2.1.1. K-means

Given a sample set \( D = \{x_1, x_2, \ldots, x_m\} \), the K-means algorithm divides the clusters \( C = \{C_1, C_2, \ldots, C_m\} \) to minimize the square error:

\[
E = \sum_{i=1}^{k} \sum_{x \in C_i} \| x - \mu_i \|^2,
\]

\( \mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x \) is the mean vector of the cluster \( C_i \). (1) shows how closely the samples in the cluster surround the mean vector of the cluster. The similarity of the samples in the cluster increases as the \( E \) value decreases.

However, minimizing the value of \( E \) is not easy. Finding its optimal solution needs to examine all possible cluster divisions of the sample set \( D \). Therefore, K-means solves the \( E \) value through iterative optimization. Its algorithm flow is shown in figure 1. First, the mean vector is initialized, and then the current cluster division and mean vector are iteratively updated. If the nodes in the cluster are unchanged, the recent cluster division result will return.
2.1.2. Gaussian mixture clustering

Unlike K-means using a mean vector, the GMM uses a probability model to construct a cluster structure. The first is to define the (multivariate) Gaussian distribution. For a random vector \( x \) in the \( n \)-dimensional sample space \( X \), if \( x \) is suitable for the Gaussian distribution, the probability density function is:

\[
p(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}, \tag{2}
\]

\( \mu \) is the \( n \)-dimensional mean vector, and \( \Sigma \) is the covariance matrix of \( x \times x \). According to (2), the mean vector \( \mu \) and the covariance matrix \( \Sigma \) can ultimately determine the Gaussian distribution. From this, we define the Gaussian mixture distribution:

\[
p_M(x) = \sum_{i=1}^{k} \alpha_i \cdot p(x \mid \mu_i, \Sigma_i). \tag{3}
\]

The distribution consists of \( k \) mixed components. Each mixture component corresponds to a Gaussian distribution, where \( \mu_i \) and \( \Sigma_i \) are the Gaussian mixture component parameters. \( p(x \mid \mu, \Sigma) \) is the probability density function, and \( \alpha_i > 0 \) is the corresponding mixing coefficient, \( \sum_{i=1}^{k} \alpha_i = 1 \).

Assuming that the Gaussian mixture distribution gives the sample generation process. First, select Gaussian mixture components according to the prior distribution defined by \( \alpha_1, \alpha_2, \ldots, \alpha_k \), where \( \alpha_i \) is the probability of the mixture components; then, according to the selected mixture components' probability density function, sampling generates the corresponding samples.

If the above process generates the training set \( D = \{x_1, x_2, \ldots, x_m\} \), let the random variable \( z_j \in \{1, 2, \ldots, k\} \) represents the Gaussian mixture component of the unknown value of the generated sample \( x_j \). The prior probability \( P(z_j = i) \) of \( z_j \) corresponds to \( \alpha_i (i = 1, 2, \ldots, k) \). According to Bayes' theorem, the posterior distribution of \( z_j \) corresponds to:

\[
p_M(z_j = i \mid x_j) = \frac{P(x_j = i) \cdot p_M(x_j \mid z_j = i)}{p_M(x_j)} = \frac{\alpha_i \cdot p(x_j \mid \mu_i, \Sigma_i)}{\sum_{i=1}^{k} \alpha_i \cdot p(x_j \mid \mu_i, \Sigma_i)}. \tag{4}
\]
When the Gaussian distribution (formula 3) is known, the Gaussian mixture clustering will divide the sample set $D$ into $k$ clusters $C = \{C_1, C_2, \ldots, C_k\}$, and the cluster label $\lambda_j$ of each sample $x_j$ is as follows:

$$\lambda_j = \arg \max_{i \in \{1, 2, \ldots, k\}} p_M(z_j = i \mid x_j).$$

(5)

The model parameter $\{(\alpha_i, \mu_i, \Sigma_i) \mid 1 \leq i \leq k\}$ in (3) will be solved using the Expectation-Maximization method, and the formula is:

$$LL(D) = \ln \left( \prod_{j=1}^{m} p_M(x_j) \right) = \sum_{j=1}^{m} \ln \left( \sum_{i=1}^{k} \alpha_i \cdot p(x_j \mid \mu_i, \Sigma_i) \right).$$

(6)

Its iterative optimization method is as follows:

If the parameter $\{(\alpha_i, \mu_i, \Sigma_i) \mid 1 \leq i \leq k\}$ can maximize the (6), then $\frac{\partial LL(D)}{\partial \mu_i} = 0$ has:

$$\sum_{j=1}^{m} \alpha_i \cdot p(x_j \mid \mu_i, \Sigma_i) (x_j - \mu_i) = 0.$$  

(7)

From (4) and $\gamma_{ji} = p_M(z_j = i \mid x_j)$, there are:

$$\mu_i = \frac{\sum_{j=1}^{m} \gamma_{ji} x_j}{\sum_{j=1}^{m} \gamma_{ji}}.$$  

(8)

The weighted average of the samples can estimate the mean value of each composite component, and the sample weight is the posterior probability of each piece belonging to the element. From $\frac{\partial LL(D)}{\partial \Sigma_i} = 0$, we can get:

$$\Sigma_i = \frac{\sum_{j=1}^{m} \gamma_{ji} (x_j - \mu_i)(x_j - \mu_i)^T}{\sum_{j=1}^{m} \gamma_{ji}}.$$  

(9)

For the mixing coefficient $\alpha_i$, in addition to maximizing $LL(D)$, it also needs to satisfy $\alpha_i \geq 0$, $\sum_{i=1}^{k} \alpha_i = 1$. Considering the Lagrange form of $LL(D)$ is:

$$LL(D) + \lambda \left( \sum_{i=1}^{k} \alpha_i - 1 \right),$$

(10)

$\lambda$ is the Lagrange multiplier, since the derivative of (10) to $a$ is 0, then:

$$\sum_{j=1}^{m} \frac{p(x_j \mid \mu_i, \Sigma_i)}{\sum_{i=1}^{k} \alpha_i \cdot p(x_j \mid \mu_i, \Sigma_i)} + \lambda = 0.$$  

(11)

Multiplying both sides by $\alpha_i$, and sum all samples to know $\lambda = -m$, which has:

$$\alpha_i = \frac{1}{m} \sum_{j=1}^{m} \gamma_{ji}.$$  

(12)

The mixing coefficient of each Gaussian component is determined by the average posterior probability of the sample belonging to the element.

The EM algorithm of the GMM can be obtained from the above derivation. In each iteration, the posterior probability $\gamma_{ji}$ of each sample belonging to each Gaussian component is calculated according to the current parameters. Then the model parameter $\{(\alpha_i, \mu_i, \Sigma_i) \mid 1 \leq i \leq k\}$ is updated according to (8), (9) and (12).

2.1.3. **DBSCAN**

Density clustering algorithms usually examine the connectivity between samples from the perspective of sample density and continuously expand clusters based on connectable pieces to obtain the final clustering results. The DBSCAN is based on a set of neighborhood parameters $(\epsilon, MinPts)$ to describe
the density of the sample distribution. Then, we define the concept using the given data set \( D = \{ x_1, x_2, \ldots, x_m \} \):

- **Neighborhood parameter:** For \( x_j \in D \), the neighborhood parameter includes the samples whose distance from \( x_j \) in sample set \( D \) (the default is even Euclidean distance) is not greater than \( \epsilon \), that is, \( N_{\epsilon}(x_j) = \{ x_i \in D \mid \text{dist}(x_i, x_j) \leq \epsilon \} \).

- **Core object:** If the neighborhood of \( x_j \) contains at least \( \text{MinPts} \) samples, that is, \( |N_{\epsilon}(x_j)| \geq \text{MinPts} \), then \( x_j \) is a core object.

- **Directly density-reachable:** If \( x_j \) is in the neighborhood of \( x_i \), and \( x_i \) is the core object, then the point \( x_j \) is directly density-reachable from the point \( x_i \).

- **Density-reachable:** For \( x_i \) and \( x_j \), if there are sample sequences \( p_1, p_2, \ldots, p_n \), where \( p_1 = x_i, p_n = x_j \) and the point \( p_{i+1} \) is directly density-reachable from a point \( p_i \), then the point \( x_j \) is density-reachable from the point \( x_i \).

- **Density-connected:** For \( x_i \) and \( x_j \), if \( x_k \) exists so that both \( x_i \) and \( x_j \) are reachable by the density of \( x_k \), then the point \( x_i \) is density-connected from the point \( x_j \).

Based on these concepts, the DBSCAN will give the domain parameters \( (\epsilon, \text{MinPts}) \), cluster \( C \subseteq D \) satisfies the non-empty sample subset:

- **connectivity:** \( x_i \in C, x_j \in C \Rightarrow x_i \) is density connected from \( x_j \),
- **maximality:** \( x_i \in C, x_j \) is density reachable from \( x_i \Rightarrow x_j \in C \).

Therefore, if \( x \) is the core object and the set of all sample combinations whose density-reachable is counted as \( X = \{ x' \in D \mid x' \) is density reachable from \( x \} \), it is not difficult to prove that \( X \) is a cluster that satisfies the continuity and the maximum value.

Therefore, the DBSCAN algorithm first selects a core object in the data set as the seed and then determines the corresponding cluster. The algorithm flow is as follows. First, the algorithm finds all core objects according to the given neighborhood parameters \( (\epsilon, \text{MinPts}) \). It then takes any core object as a starting point to find out the samples with reachable density to generate clusters until all core objects are visited.

### 2.1.4. Agglomerative clustering

The hierarchical clustering attempts to divide the data set at different levels to form a tree-shaped cluster structure. Agglomerative clustering is a bottom-up hierarchical clustering algorithm. It first regards each sample in the data set as an initial cluster and then finds the two closest groups in each step of the algorithm and merges them until the preset number of groups is reached. The formula for calculating the distance between clusters \( C_i \) and \( C_j \) is as follows:

\[
\begin{align*}
    d_{\text{min}}(C_i, C_j) &= \min_{x \in C_i, z \in C_j} \text{dist}(x, z), \\
    d_{\text{max}}(C_i, C_j) &= \max_{x \in C_i, z \in C_j} \text{dist}(x, z), \\
    d_{\text{avg}}(C_i, C_j) &= \frac{1}{|C_i||C_j|} \sum_{x \in C_i} \sum_{z \in C_j} \text{dist}(x, z).
\end{align*}
\]

The closest sample of two clusters determines the minimum distance. The farthest selection of the two clusters determines the maximum length and the same piece of the two clusters determines the average distance. When the cluster distance \( d_{\text{min}}, d_{\text{max}} \) or \( d_{\text{avg}} \) is calculated, the algorithm is called single-linkage, complete-linkage and average-linkage algorithm accordingly.
2.2. Cluster evaluation method

2.2.1. Adjusted Rand index
The Adjusted Rand index is evolved from the Rand index and requires true label. For the Rand index, given \( n \) object sets \( S = \{O_1, O_2, \ldots, O_n\} \), suppose \( U = \{u_1, \ldots, u_k\} \) and \( V = \{v_1, \ldots, v_c\} \) represent two different divisions of \( S \) and satisfy \( \bigcup_{i=1}^{k} u_i = S = \bigcup_{j=1}^{c} v_j \), \( u_i \cap u_i^* = \emptyset = v_j \cap v_j^* \), where \( 1 \leq i \neq i^* \leq k, 1 \leq j \neq j^* \leq c \).

Suppose \( U \) is the external evaluation standard, namely true label, and \( V \) is the clustering result. The four statistics are defined as:
- \( a \) is the number of data points of the same category in \( U \) and the same category in \( V \).
- \( b \) is the number of pairs of data points that belong to the same class in \( U \) but belong to different types in \( V \).
- \( c \) is the number of data points that are not in the same category in \( U \) but are in the same category in \( V \).
- \( d \) is the number of data points that are not in the same category in \( U \) and do not belong to the same type in \( V \).

At this time, the Rand index is:
\[
RI = \frac{a + d}{a + b + c + d}.
\]

Adjust the Rand coefficient is:
\[
ARI = \frac{RI - E(RI)}{\max(RI) - E(RI)}.
\]

Its value range is \((-1, 1)\), the value of 1 indicates the best clustering effect, and the value of -1 indicates the worst clustering effect.

2.2.2. V-measure
V-Measure is the harmonic average of cluster homogeneity and completeness, which is to find the degree of uncertainty in the division of a certain category after another. Its value is \((0, 1)\). The value of 1 indicates the best clustering effect, and a value of 0 indicates the worst clustering effect.

Among them, the formula for homogeneity is:
\[
h = 1 - \frac{H(C|K)}{H(C)}.
\]

Among them, \( H(C|K) \) is the entropy of a given cluster classification, \( H(C) \) is the classification entropy.

The formula for completeness is:
\[
c = 1 - \frac{H(K|C)}{H(C)}.
\]

The formula of V-Measure is:
\[
v = \frac{2 \times h \times c}{h + c}.
\]

3. Results and Discussion

3.1. Circle dataset
The original data set is two concentric circles generated from 1500 samples, as shown in figure 2. The color in the figure reflects the correct clustering results.
The results of K-means clustering, Gaussian mixture clustering (GM), and DBSCAN clustering are as follows:

A preliminary conclusion can be drawn from figure 3: the clustering effect of DBSCAN on the circle dataset is better than the effect of K-means and Gaussian mixture model.

For hierarchical clustering, this clustering has different effects in different distance selections. Three different distances are assessed here, namely the minimum distance (AC-single), the maximum distance (AC-complete), and the average distance (AC-average). Their clustering effects are as follows:

A preliminary conclusion can be drawn from figure 4: the hierarchical clustering effect using the minimum distance is better than the hierarchical clustering effect of the maximum distance and the average distance.

Through the evaluation results of Adjust rand index and V-measure, the effect of each cluster can be seen more clearly.

| Table 1. The evaluation results of the circle dataset |
|-----------------------------------------------|
| K-means | GM       | DBSCAN | AC-single | AC-complete | AC-average |
| Adjust rand index | -6.6*10^-4 | -6.39*10^-4 | 1 | 1 | 3.82*10^-3 | -6.32*10^-4 |
| V-measure | 5.13*10^-6 | 2.05*10^-5 | 1 | 1 | 3.47*10^-5 | 2.08*10^-5 |
From Table 1, the clustering effect of DBSCAN and AC-single is completely consistent with the correct clustering effect. The subsequent effects from good to bad are AC-complete, AC-average, GM, and K-means.

3.2. Different sized dataset
The original data set is three clusters of different sizes generated from 1500 samples, as shown in figure 5. The color in the figure reflects the correct clustering results.

The results of K-means clustering, Gaussian mixture clustering, and DBSCAN clustering are as follows:

A preliminary conclusion can be drawn from figure 6: the clustering effect of the Gaussian mixture model on different sized datasets is better than that of K-means and DBSCAN.

For hierarchical clustering, the clustering effects of the minimum distance (AC-single), the maximum distance (AC-complete), and the average distance (AC-average) are as follows:

A preliminary conclusion can be drawn from figure 7 is the clustering effect of hierarchical clustering on this data set is not ideal.

Six clustering models are evaluated by Adjust rand index and V-measure:

|                | K-means | GM   | DBSCAN | AC-single | AC-complete | AC-average |
|----------------|---------|------|--------|-----------|-------------|------------|
| Adjust rand index | 0.83    | 0.97 | 0.94   | 1.43*10^-5 | 0.51        | 0.72       |
| V-measure       | 0.81    | 0.94 | 0.91   | 5.25*10^-3 | 0.57        | 0.74       |
From Table 2, the accuracy of clustering results of these six models on different sized datasets from high to low is: GM, DBSCAN, K-means, AC-average, AC-complete, AC-single.

3.3. Gaussian mixture dataset
Ten Gaussian clusters are generated from 1500 samples in the original data set, and the center of each cluster is randomly selected. As shown in the figure below, the color in the figure reflects the correct clustering result.

The results of K-means clustering, Gaussian mixture clustering, and DBSCAN clustering are as shown in figure 9:

For hierarchical clustering, the clustering effects of the minimum distance (AC-single), the maximum distance (AC-complete), and the average distance (AC-average) are as figure 10:

Six clustering models are evaluated by Rand coefficient and V-measure:

|             | K-means | GM  | DBSCAN | AC-single | AC-complete | AC-average |
|-------------|---------|-----|--------|-----------|-------------|------------|
| Adjust rand index | 0.17    | 0.14| 0.12   | 0.10      | 0.13        | 0.17       |
| V-measure   | 0.32    | 0.31| 0.32   | 0.29      | 0.28        | 0.34       |

It can be seen from Table 3 that the evaluation scores of these six clustering methods on the clustering effect of the Gaussian mixture dataset are very low.
4. Conclusion
By evaluating the effects of six clustering models on three different types of data sets (circle dataset, data sets of different sizes, and Gaussian mixture data sets), we can draw the following conclusions:

- Since DBSCAN has a better classification effect on small data sets, when the data set is uncertain, the researchers can give priority to DBSCAN.
- Agglomerative clustering (AC-single), which calculates the shortest distance, only has an excellent clustering effect on circle datasets. When the data set is complex, AC-single can determine whether the data set is a circle dataset.
- It is not appropriate to use only conventional clustering to analyze Gaussian mixture data sets. It would help if used multiple clusters to process the data set or preprocess the data set before using clustering.

These findings may be a good reference for those who intend to contribute to learning research and machine learning based on basic clustering models in small sample data sets in the future. Besides, the analysis results may help people in other fields who need to use unsupervised learning to support their results.

References

[1] U. Kokate, A. Deshpande, P. Mahalle, and P. Patil, “Data Stream Clustering Techniques, Applications, and Models: Comparative Analysis and Discussion,” Big Data and Cognitive Computing, vol. 2, no. 4, p. 32, 2018.

[2] M. Lavielle and P. D. Waggoner, “Exploring and Comparing Unsupervised Clustering Algorithms,” Journal of Open Research Software, vol. 8, 2020.

[3] A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, and C.-T. Lin, “A review of clustering techniques and developments,” Neurocomputing, vol. 267, pp. 664–681, 2017.

[4] Y. Sun, K. Tang, L. L. Minku, S. Wang, and X. Yao, “Online Ensemble Learning of Data Streams with Gradually Evolved Classes,” IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 6, pp. 1532–1545, 2016.

[5] K. Mahesh Kumar and A. Rama Mohan Reddy, “A fast DBSCAN clustering algorithm by accelerating neighbor searching using Groups method,” Pattern Recognition, vol. 58, pp. 39–48, 2016.

[6] F. Ros and S. Guillaume, “DENDIS: A new density-based sampling for clustering algorithm,” Expert Systems with Applications, vol. 56, pp. 349–359, 2016.

[7] H. Ismkhan, “I-k-means→+: An iterative clustering algorithm based on an enhanced version of the k-means,” Pattern Recognition, vol. 79, pp. 402–413, 2018.

[8] L. Oreopoulos, N. Cho, D. Lee, and S. Kato, “Radiative effects of global MODIS cloud regimes,” Journal of Geophysical Research: Atmospheres, vol. 121, no. 5, pp. 2299–2317, 2016.

[9] A. E. Yuchechen, S. G. Lakkis, A. Caferri, P. O. Canziani, and J. P. Muszkats, “A Cluster Approach to Cloud Cover Classification over South America and Adjacent Oceans Using a k-means/k-means++ Unsupervised Algorithm on GOES IR Imagery,” Remote Sensing, vol. 12, no. 18, p. 2991, 2020.

[10] J. Griss, Y. Perez-Riverol, M. The, L. Käll, and J. A. Vizcaíno, “Response to ‘Comparison and Evaluation of Clustering Algorithms for Tandem Mass Spectra,’” Journal of Proteome Research, vol. 17, no. 5, pp. 1993–1996, 2018.

[11] D. Ienco and G. Bordogna, “Fuzzy extensions of the DBScan clustering algorithm,” Soft Computing, vol. 22, no. 5, pp. 1719–1730, 2018.

[12] I. Alhussein and A. H. Ali, “Application of DBSCAN to Anomaly Detection in Airport Terminals,” 2020 3rd International Conference on Engineering Technology and its Applications (IICETA), 2020.

[13] A. A. Lashkov, S. V. Rubinsky, and P. A. Eistrikh-Heller, “Application of the DBSCAN Algorithm to Detect Hydrophobic Clusters in Protein Structures,” Crystallography Reports, vol. 64, no.
[14] L. Haibo and W. Zhi, “Application of an intelligent early-warning method based on DBSCAN clustering for drilling overflow accident,” *Cluster Computing*, vol. 22, no. S5, pp. 12599–12608, 2018.

[15] A. Hulot, J. Chiquet, F. Jaffrézic, and G. Rigaill, “Fast tree aggregation for consensus hierarchical clustering,” *BMC Bioinformatics*, vol. 21, no. 1, pp. 120–120, 2020.

[16] L. Ramos Emmendorfer and A. M. de Paula Canuto, “A generalized average linkage criterion for Hierarchical Agglomerative Clustering,” *Applied Soft Computing*, vol. 100, p. 106990, 2021.

[17] R. Yusuf, B. D. Handari, and G. F. Hertono, “Implementation of agglomerative clustering and genetic algorithm on stock portfolio optimization with possibilistic constraints,” *In AIP Conference Proceedings*, vol. 2168, 2019.