mathematics

MDPI

Article

Robust Parameter Optimisation of Noise-Tolerant Clustering for DENCLUE Using Differential Evolution

Omer Ajmal [1,*], Humaira Arshad [1], Muhammad Asad Arshed [2], Saeed Ahmed [2,3] and Shahzad Mumtaz [1,4,*]

[1] Department of Computer Science, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan; humera.arshad@iub.edu.pk
[2] School of Systems and Technology, University of Management and Technology, Lahore 54770, Pakistan; asad.arshed@umt.edu.pk (M.A.A.); saeed.ahmad075@gmail.com (S.A.)
[3] Department of Experimental Medical Science, Biomedical Center (BMC), Lund University, 22184 Lund, Sweden
[4] School of Natural and Computing Sciences, University of Aberdeen, Aberdeen AB24 3UE, UK
* Correspondence: omer.ajmal@iub.edu.pk (O.A.); shahzadmumtaz22@gmail.com (S.M.)

Abstract: Clustering samples based on similarity remains a significant challenge, especially when the goal is to accurately capture the underlying data clusters of complex arbitrary shapes. Existing density-based clustering techniques are known to be best suited for capturing arbitrarily shaped clusters. However, a key limitation of these methods is the difficulty in automatically finding the optimal set of parameters adapted to dataset characteristics, which becomes even more challenging when the data contain inherent noise. In our recent work, we proposed a Differential Evolution-based DENsity CLUstEring (DE-DENCLUE) to optimise DENCLUE parameters. This study evaluates DE-DENCLUE for its robustness in finding accurate clusters in the presence of noise in the data. DE-DENCLUE performance is compared against three other density-based clustering algorithms—DPC based on weighted local density sequence and nearest neighbour assignment (DPCSA), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and Variable Kernel Density Estimation–based DENCLUE (VDENCLUE)—across several datasets (i.e., synthetic and real). The study has consistently shown superior results for DE-DENCLUE compared to other models for most datasets with different noise levels. Clustering quality metrics such as the Silhouette Index (SI), Davies–Bouldin Index (DBI), Adjusted Rand Index (ARI), and Adjusted Mutual Information (AMI) consistently show superior SI, ARI, and AMI values across most datasets at different noise levels. However, in some cases regarding DBI, the DPCSA performed better. In conclusion, the proposed method offers a reliable and noise-resilient clustering solution for complex datasets.

Keywords: DENCLUE algorithm; differential evolution; density-based clustering; parameter optimisation; noise robustness

MSC: 68W50

check for updates

Citation: Ajmal, O.; Arshad, H.; Arshed, M.A.; Ahmed, S.; Mumtaz, S. Robust Parameter Optimisation of Noise-Tolerant Clustering for DENCLUE Using Differential Evolution. *Mathematics* **2024**, *12*, 3367. https://doi.org/10.3390/math12213367

Academic Editors: Ming Yang and Liqun Shan

Received: 29 August 2024
Revised: 18 October 2024
Accepted: 25 October 2024
Published: 27 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Clustering is a major unsupervised method in data analysis and machine learning that identifies and groups data points based on their essential characteristics. It identifies cohesive groups within complex datasets with applications spanning fields like biological analysis, market segmentation, and pattern recognition. From the literature point of view, clustering methods are classified into several categories, i.e., hierarchical-based methods, grid-based methods, partition-based methods, model-based methods, and density-based methods [1]. Partition-based algorithms divide the data into discrete clusters representing specific data points. These techniques divide the dataset into divisions or clusters, and each cluster contains at least one data point. The most well-known example of a partition-based

method is k-means [2], which distributes data points to the closest cluster centre and keeps updating the centres until the clusters stabilise. This approach reduces cluster distance, effectively grouping comparable data points [3]. Second, hierarchical algorithms create a cluster hierarchy by gradually merging or separating data points depending on their commonalities. These techniques consider each data point a cluster, combining the closest pairing (bottom-up approach). Conversely, the complete dataset is considered a cluster and is then partitioned into smaller clusters (top-down approach). The final result is a tree-like structure called a dendrogram, representing the relationship between clusters, which is effective for cluster identification in nested form [4]. The density-based algorithms organise the data points into clusters based on data point density. In the density-based algorithms, data points in the denser regions are effectively distinguished from the sparser portions of the dataset. The main strength of this approach lies in the fact that there is no need to specify the number of clusters in advance, and the clusters are based on density, unlike methods that distinguish clusters based solely on distance. Furthermore, density-based approaches can automatically manage noise by classifying solitary points as outliers to improve cluster building [5].

In grid-based algorithms, the original data space is partitioned into a grid of predetermined cells. Each cell is analysed to capture the local statistical information about data points. After collecting the necessary information from the cell, the algorithm clusters the data in the group according to their statistical properties. This approach is computationally efficient, especially for large datasets [6].

In model-based algorithms, it is an explicit assumption that each cluster attaches to a predefined mathematical or statistical model. The algorithm effectively matches the data to select the best-fit cluster. There are two main types of model-based algorithms: statistical learning and neural network learning. The statistical learning algorithm considers the classical statistical models (such as Gaussian mixture) to define the clusters. At the same time, neural network learning is based on a machine learning approach such as autoencoder and other neural network topologies for detecting and creating clusters. The approach is helpful in the case of complex data structures and can be efficiently represented with existing frameworks [7].

Density-based algorithms such as DENsity CLUstEring (DENCLUE) [8] performed well compared to other partition-based or hierarchical methods for detecting clusters with complex geometries. This method is helpful in case of noisy data and when information about cluster numbers is unknown a priori. DBSCAN has been widely applied to diverse real-world datasets such as geospatial, biological, and big data [9,10].

Jang et al. [11] built a density-based clustering model to identify dialogue intention using word2vec embedding to capture data similarity. To handle the large dimensionality of word vectors, they employed T-SNE to transform the vectors into two dimensions for easier clustering. They chose to use the DENCLUE method over DBSCAN due to the former's efficiency in clustering dialogue intents. Nevertheless, the application of DBSCAN is widespread, but it is sensitive to geographic heterogeneity and fluctuations in density, leading to ineffective grouping [12–14]. Simple DBSCAN can become computationally intensive as data size increases [15]. In such cases, approximation algorithms are particularly valuable because their performance is significantly faster than the original. A trade-off between accuracy and speed is managed efficiently.

KNN-BLOCK DBSCAN [16] is based on a fast KNN approach called Fast Library for Approximate Nearest Neighbors (FLANN). The priority search and FLANN are combined to execute the nearest neighbour search. The core blocks are merged, and non-core points are attached to their respective clusters.

Another algorithm is BLOCK-DBSCAN [17] to enhance clustering. In this algorithm, a well-known cover tree due to the efficiency of the nearest neighbour is used to identify the inner core block, outer core points, and border points. A rapid and approximate approach is used to predict density-based availability between two inner core blocks. Inner-core blocks and outer-core points are combined to form a cluster in this algorithm.

PARDICLE [18] uses dynamic programming to achieve spatial proximity and load distribution. It is a parallel, fast, and approximate algorithm that employs approximate neighbour calculations to identify clusters accurately.

Soft clustering approaches were recently introduced, like E-DBSCAN [19] and Fuzzy DBSCAN [20]. These methods used soft restraints rather than exact values to express predictable values of the input parameters. The crisp DBSCAN methods identify the clusters with clear borders, while soft computing methods use overlapping and fuzzy edges to generate clusters. The strength of the fuzzy approach is that several runs of the original approach are summarised in a single run.

To overcome DBSCAN's limitations due to spatial heterogeneity, Liu, Zhou, and Wu [13] proposed the VDBSCAN (Varied DBSCAN) technique. Unlike DBSCAN, VDBSCAN is adaptive because of the adjustment of epsilon values using a k-dist plot. For each data point, average distances to the k-nearest neighbours are computed and sorted before being plotted. Sharp variations in the figure show the correct epsilon values. Through this technique, VDBSCAN can effectively handle density issues in various locations.

DENCLUE [21] is a notable density-based clustering algorithm that stands out in several aspects. DENCLUE differs from clustering algorithms like OPTICS [22] and DBSCAN by utilising Kernel Density Estimation instead of estimating density through a radius-defined region. It offers superior clustering performance in noisy datasets [23] and has the advantage of mathematically describing arbitrary-shaped clusters in high-dimensional spaces. DENCLUE can be generalised in hierarchical and partitioning-based algorithms [23,24]. By using hypercubes to map data and storing information only for populated cubes, DENCLUE reduces the computational cost typically associated with Kernel Density Estimation. Additionally, implementing a tree-based structure further enhances the efficiency of the search process. Since its introduction, several studies have been conducted for DENCLUE, mostly based on improving performance, estimating parameters, using proximity measures other than Euclidean distance, and hybrid versions [25]. Among the variants, Varying KDE-based DENCLUE (VDENCLUE) is an improved version of DENCLUE that uses variable KDE to yield a much-improved clustering performance, especially when densities vary and are geographically heterogeneous [26].

Another notable density-based technique is density peak clustering. DPCSA (Density Peaks Clustering with Weighted Local Density Sequence and Nearest Neighbor Assignment), introduced by Yu et al. [27], is a notable method. DPCSA enhances the choice of cluster centre by both the fluctuation in density and the relative positions of the data points, thus giving more accurate and reliable clustering results in complex datasets. It extends the classical density peaks clustering by introducing a new concept of weighted local density sequences and the closest neighbour assignment to enhance the clustering performance. Table 1 shows the advantages and disadvantages of existing clustering techniques, namely DBSCAN, DENCLUE, VDENCLUE, and DPCSA.

Table 1. Pros and Cons of DBSCAN, DENCLUE, VDENCLUE, and DPCSA.

Algorithm	Pros	Cons
DBSCAN [28]	- No need to predefine the number of clusters - Robust to noise and outliers - Versatile in finding clusters of various shapes and sizes	- Sensitive to parameter choices (epsilon and minPts) - Struggles with datasets that have varying densities
DENCLUE [21]	- Handles clusters of varying shapes and sizes - Less sensitive to specific distance parameters - Robust to noise	- Computationally intensive - May still struggle with datasets of highly varying densities
VDENCLUE [26]	- Effectively handles varying densities - Flexible, accommodates different cluster shapes and sizes - Improved performance over traditional DENCLUE	- Computationally intensive - Requires careful parameter tuning
DPCSA [27]	- Enhanced accuracy with local density and nearest neighbour consideration - Flexible, handles varying densities and complex shapes - Robust in complex datasets	- Increased computational complexity - Requires careful parameter tuning

Although several clustering methods have been introduced from these methods, each method has some limitations, such as challenges with high dimensional data, the need for heuristic-based or used-defined input parameters for clustering, and the overall performance issues of the methods due to the sensitivity of these user-defined parameters. In a related work, Tam et al. introduced ACDE [29], an evolutionary framework for automatic clustering based on optimising the Cluster Similarity (CS) measure; the technique is designed for centroid-based clustering, which may not be suitable for density-based datasets, especially when the clusters vary significantly in size and densities. Additionally, the CS measure is more focused on compact and well-separated clusters, which does not promote arbitrary-shaped clusters of varying sizes and densities.

Additionally, few studies evaluate the performance of density-based methods like DENCLUE in the presence of noise. Noise, the persistent and inevitable challenge in real-world datasets, can be attributed to various factors, including data gathering faults, anomalies, or external disruptions, such as hardware malfunction or environmental interference in sensor data [29], and biological variability or sample contamination in bioinformatics [30], etc. The presence of high levels of noise can significantly impact the quality of clustering by obscuring genuine patterns, resulting in the inclusion of irrelevant data points inside clusters or the exclusion of actual data points [31–33]. Consequently, the compromised ability of a clustering algorithm to accurately identify the true underlying structure of the data might provide significant challenges in deriving clusters that truly reflect the true grouping within the data.

An efficient method has been proposed to overcome the problem of parameter tuning. This method has been tested with density-based clustering algorithms to show that the proposed method can automatically tune parameters and improve clustering performance. It is essential to use density-based clustering to identify patterns in the datasets, especially the large ones. From various approaches, two of them are rather effective: DPCSA and VDENCLUE. DPCSA enhances accuracy and minimises manual parameter tuning; in contrast, VDENCLUE is best suited to fluctuating density as it uses Kernel Density Estimation to handle this variation, motivating to compare these methods with the proposed method to evaluate the effectiveness of the new approach. While the authors did not release the source code for VDENCLUE, a custom implementation was provided to enable a robust comparison, describing its advancements in managing datasets with varying densities. The major contributions of this study are listed below:

1. A novel clustering technique based on optimising DENCLUE parameters is introduced to address the issue of parameter tuning.
2. The proposed technique is compared with the recent state-of-the-art methods, i.e., VDENCLUE and DPCSA, as well as classical DBSCAN. The comparison involves testing both the clustering quality and stability under noisy conditions. The evaluation involves several key clustering metrics, both internal and external, to present a comprehensive view of the performance.
3. The methods are tested on several datasets, including both real (IRIS, Heart Disease, Seeds, and Wine) and synthetic (Aggregation, Two Moons, Path-based, Shapes, Spiral, and Zahn's Compound) with diverse characteristics to prove the efficacy of the proposed models.

The rest of the article is organised as follows: Section 2 describes the methodology, explaining the noise evaluation framework of the DE-based DENCLUE method, including the noise injection process, the benchmark methods, and the evaluation metrics. Section 3, Results, presents the comparison of the proposed method with other benchmark methods, including the experimental configuration for the proposed method and the benchmark methods and the performance of these methods across various noise levels. Section 4, Discussion, presents the robustness, strengths, and limitations of the proposed DE-based DENCLUE. Section 5 concludes the article by summarising the achievements, limitations, and potential areas for future enhancements.

2. Methodology

This study compares the noise tolerance of three existing density-based methods, i.e., DBSCAN, DPCSA, and VDENCLUE, with the proposed DE-based DENCLUE based on our foundational work [34]. The evaluation framework involves introducing noise systematically into datasets, applying the proposed method and other density-based clustering algorithms, and assessing the robustness of the resulting clusters using various performance metrics. Figure 1 shows the detailed architecture of the evaluation methodology of different methods under noisy conditions.

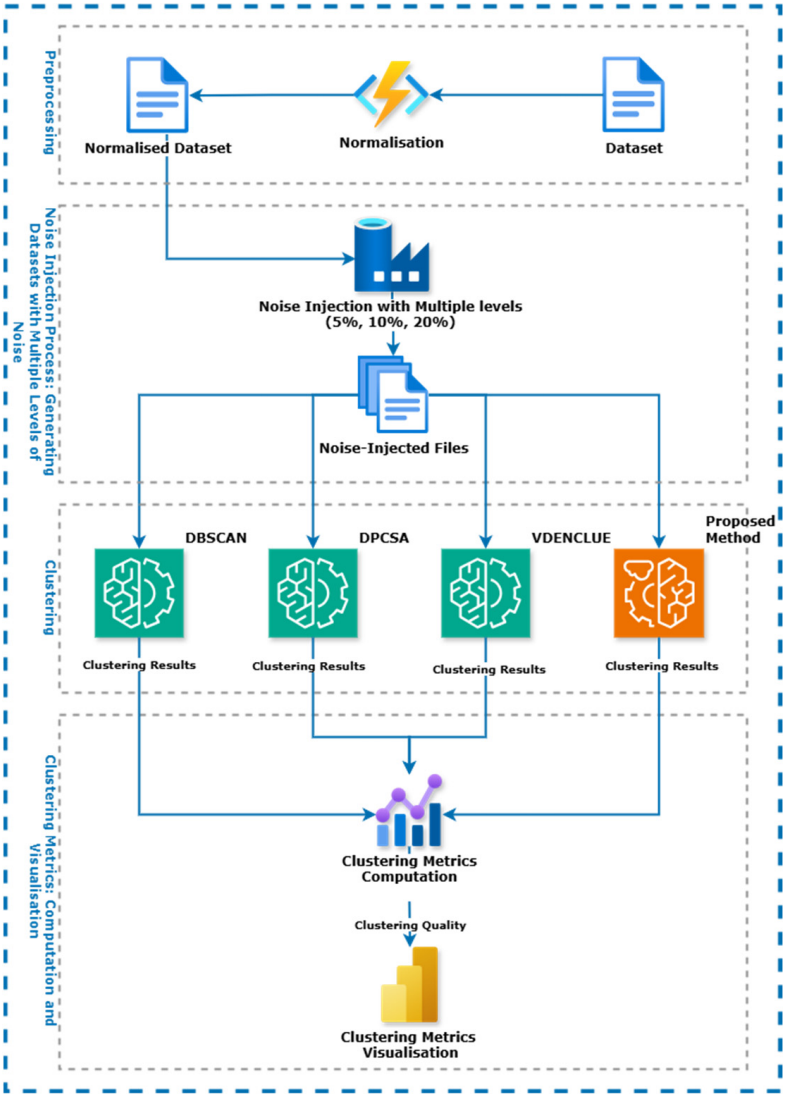


Figure 1. Architecture diagram for evaluating noise-tolerant clustering methods.

2.1. Data Preprocessing

The datasets are preprocessed to ensure consistency and comparability across different noise levels. Preprocessing involves normalising datasets so that all attributes fall within the range [0, 1] and is essential to mitigate the effects of different scales among features. Otherwise, the clustering results could be skewed. After normalisation, a pairwise distance matrix is computed for each dataset used by the DE-based DENCLUE method, as it significantly reduces computational demands by pre-calculating distances that will be frequently used throughout the iterative steps of the algorithm.

2.2. Noise Injection

To evaluate the noise tolerance of the proposed method, controlled levels of random noise are systematically injected into the datasets. The goal is to test the proposed method on low to moderate noise levels. Noise is added at 5%, 10%, and 20%, which are commonly used for evaluating the robustness of clustering methods in low and moderate noisy environments [33,35,36]. The selected noise levels effectively represent real-world data scenarios such as sensor inaccuracies or human-labelling errors, often ranging from 5% to 20% [37,38]. For each level, a set percentage of the dataset is modified by introducing uniformly distributed random data points. The noise points are added proportionately to the cluster size, simulating real-world scenarios where larger clusters may attract more noise. This step creates multiple noise-injected versions of each dataset, which are inputs for subsequent clustering evaluations. Experiments have been designed to test the methods' performance in terms of established clustering quality metrics, including the Silhouette Index (SI), Davies–Bouldin Index (DBI), Adjusted Rand Index (ARI), and Adjusted Mutual Information (AMI), across varying levels of noise. In addition to these traditional clustering metrics, we also evaluated the proposed method through density-based clustering validation (DBCV) metric [39], a metric designed to promote the density-based structure of complex arbitrary-shape clusters of varying densities. DBCV is an internal clustering validation metric, as it does not require labelled data, and it effectively identifies complex clusters in the density landscape based on density sparseness and density separation. It is based on the concepts of density sparseness (the areas within a cluster having the lowest density) and density separation (the highest-density regions between adjacent clusters). Metrics will be obtained for both the presence and absence of noise, i.e., the results of the proposed approach on a dataset with 0% noise. In these experiments, the primary focus is understanding and demonstrating how the proposed approach behaves as the noise level increases. Testing at various noise thresholds allows for pinpointing where performance begins to degrade, providing invaluable insights into the approach's noise tolerance boundaries.

2.3. Evaluation of Clustering Methods

The original and noise-injected datasets were then subjected to clustering using the proposed approach and three recognised density-based algorithms. Detailed descriptions of the proposed method and benchmark algorithms are provided in the following sections.

2.3.1. Proposed Method—DE-Based DENCLUE Parameter Optimisation

The proposed method optimises the DENCLUE parameters—namely the influence range (σ) and the minimum density threshold (ξ)—using Differential Evolution (DE). The DE process refines these parameters to maximise clustering quality, even under noisy conditions. The method is introduced in our recent work [34], and a detailed mechanism of the method is depicted in Figure 2. The parameters are optimised using an objective function that combines the DBCV metric and Cluster Coverage, aiming to maximise clustering quality while ensuring that a significant portion of the dataset is clustered. The rationale for including Cluster Coverage is that traditional clustering metrics (such as SI, DBI, etc.) do not account for the proportion of data that is clustered and might yield superior metric values even when a small fraction of the dataset is clustered. This limits their applicability in real-world scenarios where visualising the cluster outputs or manual verification by the user is not feasible [35].

The proposed method using DE generates and evolves candidate solutions over generations. A candidate solution for the i-th individual in the population can be represented as a two-dimensional vector, i.e., $[\sigma_i, \xi_i]$. The candidate solutions are passed to DENCLUE, where clustering is performed, and the quality of clustering is evaluated by a scalarised function. This function is formulated as a scalarised value combining the DBCV score and cluster coverage, weighted by Pareto dominance relations to ensure a balance between these objectives. The core of our methodology lies in dynamically assigning these weights

based on Pareto dominance. This allows the optimisation process to maintain a balance between these two objectives, which is a hallmark of multi-objective optimisation. The scalarised value for each candidate solution is calculated using Equation (1), where $DBCV_i$ represents the clustering quality for the current candidate solution, and $ClusterCoverage_i$ is the percentage of the data clustered by this candidate solution. Weights are determined using Pareto Ranking, with higher weights assigned to non-dominated solutions that effectively balance clustering quality and coverage. The objective function of the proposed method guarantees a trade-off where most of the data are clustered, and the underlying density-based structure is detected.

$$ScalarizedValue_i = DBCV_i \times w_i + (1 - w_i) \times ClusterCoverage_i \quad (1)$$

The DE's operators, such as mutation, crossover, and selection, are customised for the parameter estimation of DENCLUE [35]. Bounds are defined for the variable's influence range and minimum density threshold, and a population is generated randomly within the bounds. Once the population is initialised, the mutation is performed using a base vector, and the difference between two randomly selected vectors is used to generate a mutant vector. After mutation, binary crossover is performed to produce a trial vector. The fitness of the trial and the base vector is evaluated by using Equation (1) and the vector with a higher value is retained in the population.

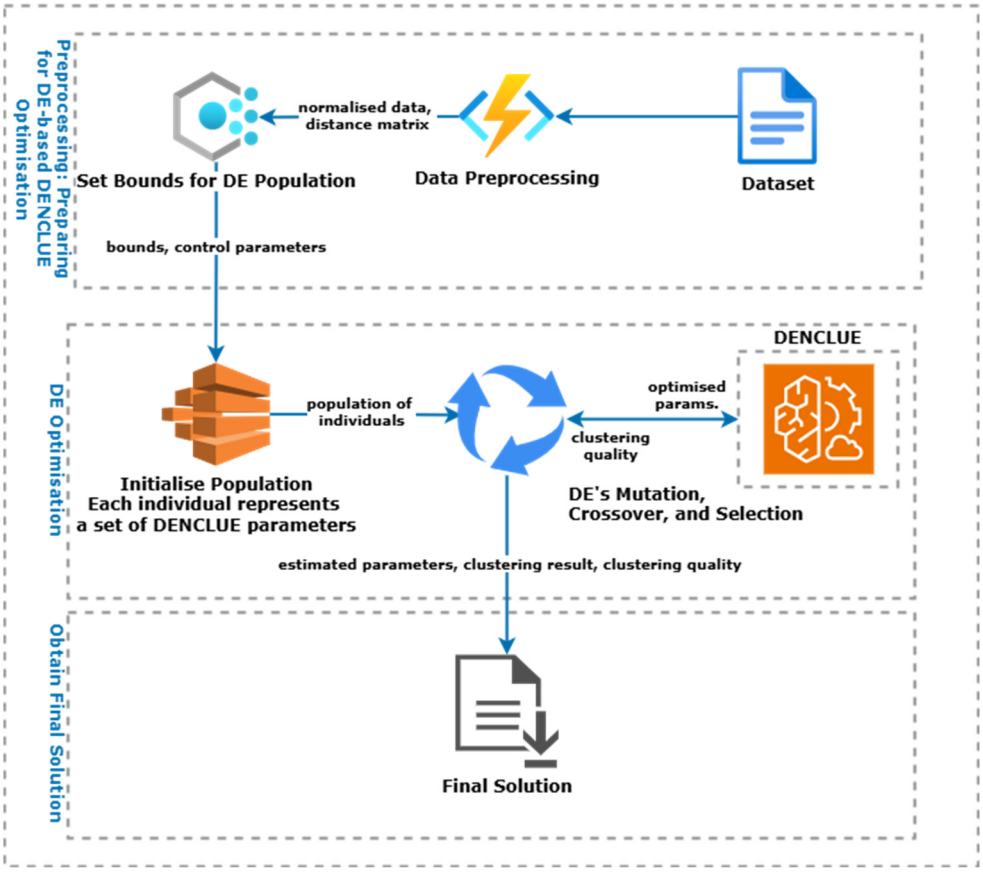


Figure 2. Proposed method (DE-based DENCLUE parameter estimation).

The current study is an extension of our recent work [35] by evaluating its noise tolerance, especially due to the DENCLUE's density-based clustering approach and DBCV's density-focused validation capabilities. These capabilities enable the proposed method to better detect underlying structures even under noisy conditions, ensuring robust performance in real-world scenarios.

The proposed method uses the DE/rand/1/bin scheme, known for its simplicity and robustness in optimisation tasks. The selected DE's control parameters values for scaling factor F and crossover probability CR are based on sensitivity analysis presented in our recent work [35] and conform to the previous empirical studies, further discussed in Section 3.1 Experimental Setup of this article.

Why Differential Evolution (DE) for Parameter Optimisation?

DE has emerged as a robust, efficient, and widely adopted optimisation method, which is particularly suited for problems involving complex, non-convex, discontinuous optimisation tasks—common characteristics of clustering parameter optimisation. Its suitability for optimising continuous parameters (such as DENCLUE's bandwidth/influence range) and effectiveness in searching complex, high-dimensional parameter spaces make it an ideal choice for clustering parameters optimisation. Additionally, DE is superior to other evolutionary algorithms, such as Particle Swarm Optimisation (PSO) [40] or Genetic Algorithm (GA) [41] in certain factors such as the following:

a. DE has a better ability to explore the search space as compared to PSO and GA [42]. While PSO tends to emphasise exploitation (local search) and GA focuses on exploration (introducing new candidates in the search space), DE provides a strong balance between exploitation and exploration. This not only prevents DE from converging prematurely but also ensures that the convergence is carried out in a reasonable time.
b. PSO does not provide a mutation mechanism like DE, which may lead to premature convergence to local optima. Moreover, DE's mutation mechanism is superior to GA's random mutation.
c. PSO, unlike DE and GA, does not provide an explicit crossover mechanism. While GA provides an explicit crossover, DE has a much simpler and more effective crossover mechanism than GA.
d. Both PSO and GA are highly sensitive to control parameters, while DE is relatively robust to the control parameters [43,44]. Also, DE takes fewer control parameters than PSO and GA.
e. In DE, the influence of population size on convergence time is much less than the exponential time of GA due to the sorting and selection process. PSO, although having better convergence speed than DE, premature convergence limits its usage in complex high-dimensional problems. DE is more computationally efficient than GA due to simpler mutation and crossover.
f. DE is a preferred choice in solving continuous optimisation problems especially when the objective function is continuous, multi-modal, and non-differentiable. It has been applied in domains like parameter tuning [45,46], machine learning [47], and signal processing [48].

Computational Complexity of the Proposed Method

The proposed method's time complexity arises from several key components (Figure 2), including reading the dataset, normalising data, computing the pairwise distance matrix, and executing the Differential Evolution (DE) optimisation algorithm. Specifically, the initial steps of reading and normalising the dataset each contribute linearly to the complexity based on the number of data points and their dimensionality. Within the DE optimisation, various operations such as initialisation, mutation, crossover, selection, and fitness evaluation contribute to the overall complexity. Given these factors, the overall time complexity of the proposed method is $\boldsymbol{O}(\boldsymbol{NP}.\boldsymbol{N}^2)$, where NP represents the population size in the DE algorithm, and N is the number of data points. The primary components of the space complexity include the dataset, the pairwise distance matrix, and the population utilised in the DE algorithm. The most significant term among these is the pairwise distance matrix, which leads to an overall space complexity of $\boldsymbol{O}(\boldsymbol{N}^2)$. The detailed computation of the time and space complexity of the proposed method is presented in our recent study [35].

2.3.2. Other Density-Based Methods

In addition to the proposed method, three other well-recognised density-based clustering methods are applied to the datasets, serving as benchmarks to compare the noise tolerance of the proposed approach. The chosen methods show considerable promise in tackling density-based clustering issues. The first benchmark method, DBSCAN, was introduced by Ester et al. [8]; it is a well-recognised density-based algorithm that defines clusters as regions of dense data points separated by sparser areas. DBSCAN is known for its ability to handle noise and discover clusters of arbitrary shapes; significant research has been carried out on its application, evaluation, and modification, making it a valuable candidate for evaluating the proposed method. The computational complexity of DBSCAN is $O(N \cdot logN)$ in optimal cases, primarily due to the efficiency of spatial indexing structures like KD trees. DBSCAN's code is available across various libraries, such as scikit-learn [49], Weka [50], and other open-source implementations. For this work, the scikit-learn implementation is used.

The second benchmark method, DPCSA [27], is an enhanced version of density peak clustering (DPC), which overcomes traditional DPC limitations by employing a weighted local density sequence and a two-stage assignment strategy. The results show that the method can effectively handle datasets with different densities, improving accuracy, efficiency, and automatic cluster centre determination. The method is validated on multiple datasets, and the code is openly available for further research. The third method, VDENCLUE [26], is an improved version of the DENCLUE algorithm designed to handle clusters with varying densities better. The method uses a varying Kernel Density Estimation (KDE) that dynamically adjusts the kernel bandwidth based on local data characteristics. Results show that VDENCLUE is superior to the traditional DENCLUE algorithm across various datasets. The benchmark methods, although designed to handle datasets with clusters of varying densities effectively, their performance on varying noise levels remains unexplored to the best of the authors' knowledge. The computational complexity of VDENCLUE is $O\left(N^2\right)$, as it evaluates the density relationships among all pairs of points. On the other hand, the computational complexity of DPCSA is $O\left(N.\log N\right)$ when optimised, but it can approach $O\left(N^2\right)$ in less efficient implementations.

The original and noise-injected datasets are clustered using all three methods, and then the clustering produced is evaluated using established clustering validation metrics. The analyses help to determine the methods' effectiveness in identifying meaningful clusters within the noisy datasets.

2.4. Performance Evaluation Through Clustering Metrics

Using metrics in cluster analysis is essential to assess and validate the performance of an algorithm objectively. Clustering metrics provide measurable insights into how well the algorithm has grouped the data and help determine the overall effectiveness of the clustering process [51]. External validation metrics are clustering metrics that compare the clustering results to a known ground truth, providing a direct assessment of how closely the predicted clusters match the true labels. On the other hand, internal validation metrics assess the clustering quality without using external labels and focus on cohesion and separation of the clusters. This study employs both types of metrics to evaluate the clustering performance comprehensively. External validation metrics used in the evaluation are the Adjusted Random Index (ARI) [52] and adjusted Mutual Information (AMI) [53]. In contrast, internal validation metrics are the Silhouette Index (SI) [54], Davies–Bouldin Index (DBI) [55], and a density-specific clustering validation DBCV [41]. The ARI measures how similar two clusters are by adjusting for chance and focusing on the agreement between pairs of elements; its range is from -1 to 1, with 1 indicating perfect agreement. In contrast, AMI assesses the shared information between two clustering, adjusted for chance, and ranges from 0 to 1, where 1 indicates perfect agreement. The SI evaluates the cohesion within and separation between clusters, ranging from -1 to 1, where 1 indicates well-separated and cohesive clusters. At the same time, the DBI assesses the separation

between clusters relative to their size, with lower values indicating better clustering and ranges from 0 to infinity [56]. Although these traditional metrics are extensively used in evaluating the performance of clustering algorithms, specifically in the context of noisy datasets [31,57], evaluating the clustering performance and stability in the context of density-based validation metrics provides insights into the strengths and limitations of the methods in identifying complex, non-convex clusters in noisy settings. The DBCV ranges from -1 to $+1$, with higher values indicating superior density-based clustering.

3. Results

The tolerance of the proposed DE-based DENCLUE parameter optimisation method under varying noise levels is evaluated by comparing its performance against well-known density-based clustering algorithms. The ability of each algorithm to maintain clustering accuracy and quality is thoroughly assessed under noise levels of 5%, 10%, and 20%. The analysis is based on external and internal validation metrics to present a comprehensive view of how well each method adapts to noise, helping to provide insights into the strengths and limitations of each method in terms of their robustness to noise and their effectiveness in clustering across real and synthetic datasets.

3.1. Experimental Setup

To obtain a thorough assessment, this section presents the details of the experimental design employed for the performance evaluation of different clustering methods on noisy datasets and their resistance to noise. A collection of ten datasets—four real and six synthetic—are used in the experiments. Table 2 summarises the details of the datasets used. The real datasets are taken from the UCI machine learning repository [58], while the synthetic datasets are benchmarked in clustering evaluation, reported in [59] and are publicly available (http://cs.uef.fi/sipu/datasets/ (accessed on 29 August 2024)). Moreover, for each dataset, experiments were performed using the original dataset (0% noise), and with 5%, 10%, and 20% noise, randomly injected as additional points in proportion to the actual cluster sizes.

Table 2. Datasets for noise robustness—real and synthetic.

Dataset	Samples	Attributes	Clusters	Domain
IRIS	150	4	3	Biological data
Heart Disease	303	13	5	Medical diagnosis
Seeds	210	7	3	Agricultural data
Wine	178	13	3	Chemical analysis
Aggregation	788	2	7	Complex shapes
Two Moons	100	2	2	Non-linear separability
Path-based	300	2	3	Path-shaped clusters
Shapes	400	2	4	Geometric diversity
Spiral	312	2	3	Non-convex clusters
Zahn's Compound	399	2	6	Varying densities

The benchmark methods used for comparison in this study were implemented using widely recognised and reliable libraries to ensure consistency and reproducibility. The DBSCAN algorithm was implemented using the scikit-learn library [49], and the DPCSA method's publicly available implementation (https://github.com/Yu123456/DPCSA (accessed on 29 August 2024)) was used as reported in [27]. A custom implementation of VDENCLUE was utilised due to its relatively straightforward design and ease of implementation, and it has been made available (https://github.com/omerdotajmal/vdenclue (accessed on 29 August 2024)) to ensure reproducibility and fair comparison. Next, the

parameters are set in advance for each counterpart method. The DBSCAN takes two parameters: eps, which specifies the radius of the neighbourhood around a point, and minpts, the minimum number of points required within this neighbourhood to form a cluster. DPCSA's rho (ρ) measures local density, whereas delta (δ) is the distance to denser points for cluster centre identification. VDENCLUE's K adjusts kernel bandwidth while the density threshold sets minimum density for clustering. The features of the datasets are all normalised during the preprocessing to be in the range [0, 1]. Due to this, the small-scaled values of eps and minpts worked and showed the best results. For DPCSA's parameters, five regions of interest in the space [0, 1] are defined. The parameter ranges are defined by the authors of VDENCLUE [26]. The parameter settings for the methods are shown in Table 3. The methods were run using these parameters, and the average metric values were recorded.

Table 3. Parameter settings for methods.

Method	Parameter	Range
DBSCAN	eps minpts	[0.05, 0.1, 0.15, 0.2, 0.25, 0.3] [3, 5, 7, 10, 12]
DPCSA	ρ δ	[0, 1] [0, 1]
VDENCLUE	K density threshold	$[4,\ 12] \bigcap Z$
Proposed method	F CR	0.5 0.8

The proposed parameter optimisation method for DENCLUE based on DE takes two control parameters: the differential evolution's scaling factor F and the crossover probability CR. The choice of F and CR values is based on experimenting with different values and a detailed sensitivity analysis across various datasets reported in our recent work [34]. The selection criteria of a suitable combination are based on the mean generations to reach threshold performance (in terms of the objective function value) for different combinations of F and CR. Figures A1 and A2 represent the top five combinations of F and CR for synthetic and real datasets, respectively. Additionally, the selected combination of F = 0.5 and CR = 0.8 conform to already reported suitable values for similar problems [60–64].

The proposed method was run thrice with selected DE control parameters for consistency, while the other methods were run with values in ranges defined in Table 3. A detailed analysis of the spread of values for each method across multiple runs is presented in Appendix A (Figures A3–A12).

3.2. Clustering Quality and Noise Tolerance

The clustering quality, measured by different metric values produced by the proposed method and three benchmark methods across noise levels, is presented in Table 4 for synthetic datasets and Table 5 for real datasets. The best values for each metric at a particular noise level are boldfaced. The SI and DBI values could not be computed if a single cluster was produced and shown by blanks in the tables. Also, the statistical significance of the performance differences between the proposed method and the benchmark methods was assessed, as reported in Appendix B (Tables A1–A5), to further validate our method's capability in maintaining clustering quality and noise tolerance. The datasets were divided into three groups: the first group containing datasets without significant overlap (all except Path-based, Seeds, and Heart Disease); the second group containing only datasets with significant cluster overlap (Path-based, Seeds, and Heart Disease); and the third group containing all of the datasets.

Table 4. Results for synthetic datasets. For each metric, the best-performing value at each noise level is highlighted in bold to indicate superior clustering quality.

		VDENCLUE				DPCSA				DBSCAN				Proposed Method			
		0%	**5%**	**10%**	**20%**	**0%**	**5%**	**10%**	**20%**	**0%**	**5%**	**10%**	**20%**	**0%**	**5%**	**10%**	**20%**
Aggregation	**SI**	0.12	0.09	0.07	0.08	**0.46**	0.40	**0.48**	**0.35**	0.41	**0.44**	0.36	0.34	0.43	**0.43**	0.4	0.31
	DBI	1.03	1.03	1.11	1.08	0.74	**0.57**	**0.72**	**0.90**	**0.74**	0.88	1.25	1.32	0.91	0.89	0.87	0.91
	ARI	0.03	0.03	0.03	0.02	0.56	0.13	0.46	0.17	**0.80**	0.22	0.23	0.11	0.76	**0.75**	**0.73**	**0.69**
	AMI	0.21	0.19	0.17	0.15	0.63	0.14	0.47	0.18	**0.87**	0.24	0.24	0.12	0.86	**0.81**	**0.78**	**0.71**
	DBCV	−0.48	−0.59	−0.46	−0.69	−0.18	−0.39	−0.84	−0.77	−0.26	−0.26	−0.26	−0.34	**−0.1**	**−0.16**	**−0.21**	**−0.22**
Two Moons	**SI**	0.35	0.32	0.29	0.25	0.37	0.34	0.34	0.32	0.34	0.20	0.35	0.33	**0.4**	**0.4**	**0.4**	**0.37**
	DBI	**0.63**	**0.68**	**0.79**	**0.79**	1.01	1.02	0.97	1.14	1.06	1.19	1.16	1.08	1.33	1.33	1.31	1.21
	ARI	0.07	0.07	0.07	0.05	1.00	0.99	0.89	0.68	0.82	0.27	0.26	0.16	**1.00**	**1.00**	**0.97**	**0.85**
	AMI	0.19	0.18	0.18	0.12	1.00	0.99	0.82	0.57	0.85	0.29	0.25	0.16	**1.00**	**1.00**	**0.94**	**0.76**
	DBCV	−0.18	−0.24	−0.25	−0.53	**0.86**	**0.57**	**0.16**	−0.61	0.16	0.1	−0.08	−0.06	0.16	0.16	−0.05	**−0.02**
Path-based	**SI**	0.21	0.18	0.13	0.13	0.34	0.33	0.31	**0.48**	0.09	**0.41**	0.40	0.34	**0.42**	0.27	**0.55**	0.42
	DBI	**0.86**	1.00	1.11	1.02	1.06	1.14	1.14	**0.72**	1.90	1.25	1.56	1.60	1.24	**1.02**	**0.63**	0.78
	ARI	0.07	0.07	0.07	0.05	0.22	0.10	**0.19**	**0.06**	0.37	0.10	0.08	**0.06**	**0.57**	**0.22**	−0.01	0.01
	AMI	0.20	0.19	0.17	**0.14**	0.27	0.12	**0.21**	0.06	0.42	0.11	0.09	0.08	**0.67**	**0.33**	0.01	0.01
	DBCV	−0.38	−0.43	−0.47	−0.56	**0.04**	−0.96	−0.86	−0.91	−0.48	−0.51	−0.59	−0.59	−0.41	**−0.18**	**−0.41**	**−0.56**
Shapes	**SI**	0.27	0.17	0.15	0.14	0.55	0.47	0.43	0.40	0.54	0.46	0.43	0.40	**0.71**	**0.69**	**0.49**	**0.53**
	DBI	0.85	0.97	1.03	1.02	0.64	0.96	0.84	0.87	0.71	1.49	1.52	1.53	**0.58**	**0.58**	**0.58**	**0.67**
	ARI	0.11	0.11	0.10	0.08	0.71	0.53	0.38	0.33	0.73	0.45	0.38	0.36	**1.00**	**0.96**	**0.95**	**0.83**
	AMI	0.27	0.24	0.22	0.19	0.83	0.60	0.48	0.42	0.83	0.48	0.46	0.38	**1.00**	**0.94**	**0.93**	**0.77**
	DBCV	−0.43	−0.47	−0.62	−0.59	**0.87**	−0.15	−0.25	−0.54	0.23	0.09	−0.07	**−0.17**	0.23	**0.09**	**0.08**	−0.26
Spiral	**SI**	**0.29**	0.25	0.23	0.21	0.00	**0.42**	-	0.38	−0.06	0.38	**0.30**	0.31	0.02	0.01	0.1	**0.47**
	DBI	**0.97**	1.09	**1.02**	1.13	5.87	**0.43**	-	**0.53**	4.28	2.24	2.61	2.04	7.73	8.02	4.07	0.82
	ARI	0.03	0.03	0.02	0.02	0.2	−0.000006	0	−0.00002	0.46	0.05	0.05	**0.03**	**1.00**	**0.97**	**0.96**	0.01
	AMI	0.15	0.14	0.13	**0.11**	0.2	0.0008	0	0.001	0.57	0.08	0.08	0.05	**1.00**	**0.94**	**0.94**	0.01
	DBCV	**−0.14**	−0.33	**−0.33**	−0.59	−0.82	−0.82	−1	−0.89	−0.37	−0.34	−0.46	−0.48	−0.23	**−0.24**	−0.41	**−0.43**
Zahn's Compound	**SI**	0.19	0.16	0.13	0.17	0.53	0.54	0.53	**0.51**	0.37	0.40	0.39	0.36	**0.61**	**0.60**	**0.62**	0.41
	DBI	1.01	0.99	1.00	1.03	**0.70**	**0.47**	**0.53**	**0.50**	2.02	1.94	2.17	1.46	0.77	0.79	0.76	0.75
	ARI	0.11	0.10	0.09	0.06	0.29	0.13	0.12	0.09	0.69	0.27	0.21	**0.15**	**0.75**	**0.72**	**0.7**	0.01
	AMI	0.24	0.22	0.20	**0.17**	0.31	0.14	0.12	0.09	0.75	0.28	0.21	0.16	**0.81**	**0.76**	**0.71**	0.01
	DBCV	−0.55	−0.59	−0.69	−0.56	−0.18	−0.68	−0.69	−0.73	**−0.06**	−0.17	−0.24	**−0.2**	−0.1	**−0.12**	**−0.14**	−0.63

Table 5. Results for real datasets. For each metric, the best-performing value at each noise level is highlighted in bold to indicate superior clustering quality.

		VDENCLUE				DPCSA				DBSCAN				Proposed Method			
		0%	**5%**	**10%**	**20%**	**0%**	**5%**	**10%**	**20%**	**0%**	**5%**	**10%**	**20%**	**0%**	**5%**	**10%**	**20%**
IRIS	**SI**	0.2	0.18	0.14	0.06	0.58	0.53	0.43	0.42	0.43	0.42	0.4	0.37	**0.64**	**0.6**	**0.62**	**0.57**
	DBI	1.06	1.14	1.25	1.36	**0.53**	**0.4**	**0.4**	**0.4**	3.14	2.8	2.43	1.75	0.69	0.81	0.71	0.86
	ARI	0.18	0.17	0.16	0.14	**0.63**	0.43	0.26	0.2	0.48	0.45	0.41	0.36	0.57	**0.55**	**0.55**	**0.50**
	AMI	0.27	0.24	0.22	0.19	**0.76**	0.52	0.26	0.2	0.61	0.55	0.5	0.46	0.74	**0.71**	**0.69**	**0.62**
	DBCV	−0.61	−0.79	−0.81	−0.85	−0.53	−0.83	−0.82	−0.88	−0.26	−0.31	0.22	**−0.02**	**0.15**	**−0.2**	**0.25**	−0.29
Heart Disease	**SI**	0.09	0.03	0.02	0.03	0.15	0.21	0.22	0.19	−0.16	−0.17	−0.17	−0.17	**0.33**	**0.30**	**0.31**	**0.39**
	DBI	2.29	2.54	2.25	2.2	**0.43**	**0.54**	**0.49**	**0.45**	1.34	1.33	1.32	1.3	1.67	1.79	1.72	1.45
	ARI	0.1	0.05	0.06	0.02	0.03	0.05	0.05	0.04	−0.05	−0.05	−0.05	−0.05	**0.13**	**0.13**	**0.14**	**0.17**
	AMI	0.08	0.06	0.07	0.05	0.09	0.1	0.1	0.09	0.02	0.02	0.02	0.02	**0.14**	**0.13**	**0.14**	**0.16**
	DBCV	−0.55	−0.66	−0.70	−0.67	−0.87	−0.93	−0.92	−0.89	−0.88	−0.92	−0.91	−0.92	**0.09**	**0.09**	**0.03**	**0.15**
Seeds	**SI**	0.11	0.11	0.09	0.02	**0.38**	**0.4**	0.36	0.3	0.06	0.14	0.13	0.1	0.07	0.08	**0.42**	**0.42**
	DBI	1.38	1.51	1.56	1.76	**0.8**	**0.42**	**0.44**	**0.45**	2.1	2.26	2.27	2.3	1.37	1.59	1.00	1.35
	ARI	0.17	0.16	0.15	0.12	**0.49**	**0.49**	**0.46**	**0.38**	0.16	0.15	0.14	0.11	0.16	0.01	−0.01	−0.01
	AMI	0.22	0.2	0.18	0.15	**0.54**	**0.53**	**0.45**	**0.34**	0.23	0.22	0.2	0.19	0.20	0.01	0.01	0.01
	DBCV	−0.47	−0.68	−0.64	−0.76	−0.58	−0.88	−0.94	−0.94	−0.34	−0.36	−0.41	−0.43	**−0.04**	**−0.24**	**−0.33**	**−0.4**
Wine	**SI**	0.15	0.19	0.15	0.11	0.1	0.16	0.17	**0.16**	−0.27	−0.29	−0.31	−0.32	**0.35**	**0.30**	**0.35**	0.15
	DBI	1.59	1.67	1.82	1.83	**0.87**	**0.55**	**0.44**	**0.39**	1.48	1.56	1.68	1.8	1.64	1.68	1.65	1.81
	ARI	0.2	0.17	0.16	0.12	0.36	0.47	0.37	**0.31**	0.04	0.04	0.04	0.03	**0.50**	**0.49**	**0.50**	−0.01
	AMI	0.24	0.19	0.19	0.15	0.44	0.49	0.41	**0.33**	0.17	0.17	0.16	0.15	**0.61**	**0.60**	**0.61**	0.02
	DBCV	−0.50	−0.69	−0.57	−0.75	**−0.17**	−0.81	−0.76	−0.85	−0.55	−0.68	−0.71	−0.71	−0.19	**−0.19**	**−0.19**	**−0.2**

Additionally, the radar charts—presented in Figure 3 for synthetic datasets and Figure 4 for real datasets—offer a comprehensive view of the performance variability across all ten datasets, with each axis corresponding to a specific noise level (0%, 5%, 10%, and 20%), and for each method, the polygon reflects the methods' performance on different clustering metrics. The radar charts help quickly compare how different methods behave across noise levels. Larger and more regular polygons, for SI, DBCV, ARI and AMI, represent better and more stable performance across noise levels. However, for DBI, smaller polygons are preferred. The polygon's distortion at different noise levels aids in assessing the robustness of each method; a method with minimal changes in the polygon shape across noise levels is more resilient to noise, indicating stable performance. On the other hand, significant shape changes in the polygon's shape across noise levels indicate the method is more affected by noise. Additionally, while radar charts are effective for understanding overall patterns, they are supplemented with heatmaps in Appendix A (Figures A13–A22) for each dataset to enhance further readers' understanding of the specific metric values that contribute to these patterns, especially for overlapping or similar polygon shapes.

The analysis of the SI metric reveals that the proposed method is stable for most datasets across noise levels, except for the Spiral and Seeds dataset, at an increased noise level of 20%. In the Spiral dataset, SI jumps notably at 20% noise; the previously separated spirals tend to have a pathway between them, effectively bridging all three spirals and suggesting improved cluster coherence and separation. However, this is not usually desirable in density-based clustering, where clusters are to be identified based on density, not cohesion, which is usually high for tightly grouped and spherical clusters. For the Seeds dataset, an increase is observed at 10% noise, suggesting the noise points increased the cluster cohesion. Despite the quality fluctuations in these datasets, the proposed method maintains a high SI value for most datasets, achieving the top score in seven out of ten datasets at 5% noise, eight out of ten at 10% noise, and six out of ten at 20% noise. The DPCSA has shown moderate resilience to noise. However, it suffers from performance loss at higher noise levels. VDENCLUE also suffers from a significant loss at higher noise levels, while DBSCAN, although strong at lower noise levels, rapidly loses quality as noise increases.

The adjusted random index is stable in six out of ten datasets. Additionally, the ARI value outperforms other methods in seven of ten datasets, demonstrating high agreement with true clusters and resilience to noise at all levels.

The drop in ARI values for Spiral, Zahn's Compound, and Wine occurs at 20% only, and for the complex Path-based dataset at 10%, the proposed method struggles with the Seeds dataset even without noise due to significant overlap in several dimensions of the dataset. The proposed method maintains higher resilience and strength for the ARI metric than other methods across noise levels. VDENCLUE generally faces challenges with lower performance across noise levels than the proposed method and weaker stability than the proposed method. DPCSA perform reasonably well for low and moderate noise levels; however, it lacks stability at higher noise levels. DBSCAN is most affected by noise as it suffers from a significant performance loss as noise increases with a sharp drop in values, especially for complex synthetic datasets.

A similar pattern for AMI is observed where the proposed method outperforms eight out of ten datasets at all noise levels. Additionally, the method is resilient to noise for six out of ten datasets, only dropping at 20% noise for the complex Path-based, Spiral, Zahn's compound, and Wine datasets. Broadly, the proposed method tends to maintain better performance and stability in low to moderate noise levels, i.e., up to 10%. VDENCLUE performance and stability are consistently lower than the proposed method. DPCSA generally maintains stable performance, however, with lower AMI values as compared to the proposed method. DBSCAN exhibits consistent instability and lower performance than the proposed method with a sharp decline in AMI values in noisy datasets.

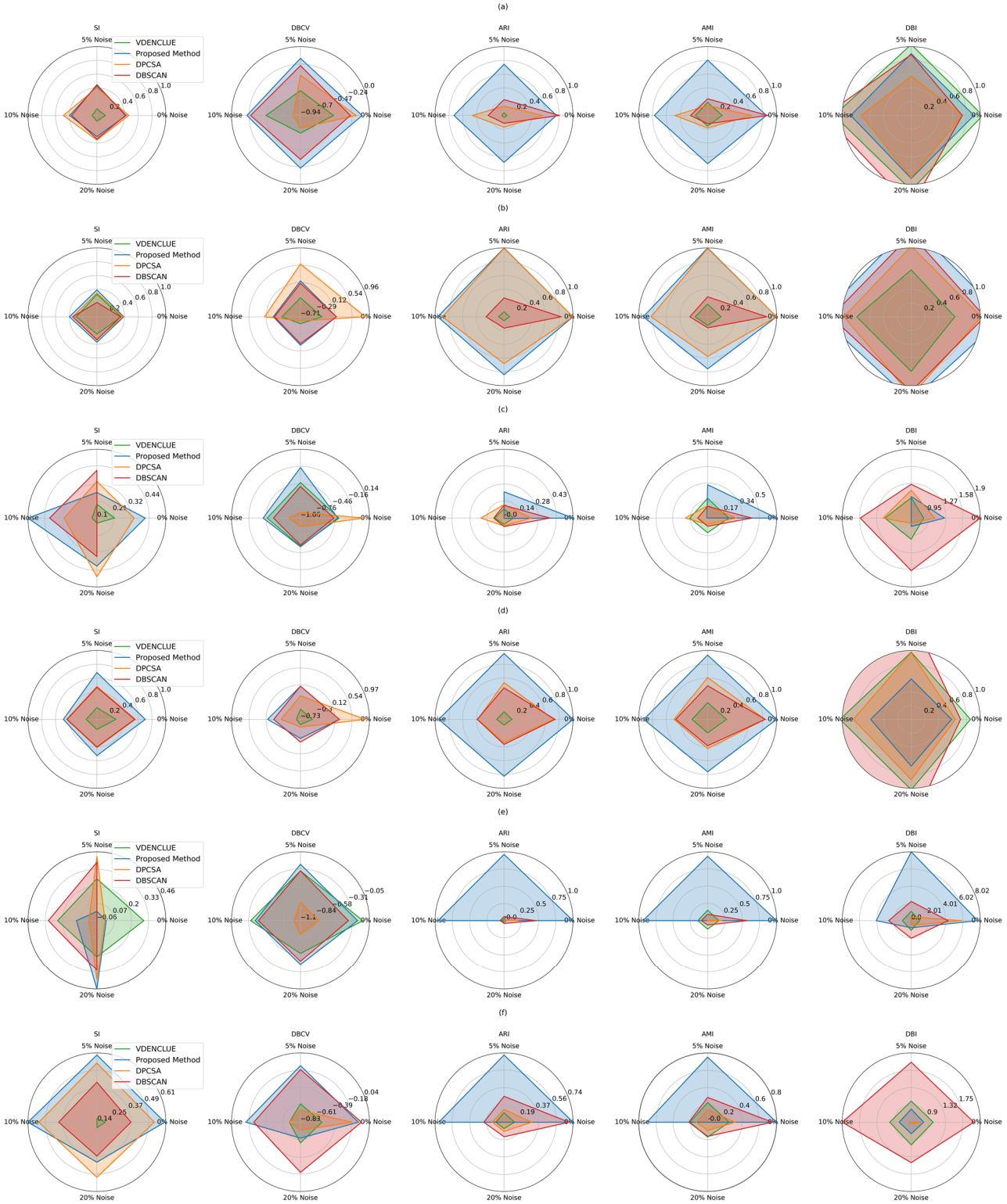


Figure 3. Metric variability across noise levels for synthetic datasets: (**a**) Aggregation; (**b**) Two Moons; (**c**) Path-based; (**d**) Shapes; (**e**) Spiral; and (**f**) Zahn's Compound.

An intriguing phenomenon is observed for the DBI metric. The results demonstrate that DPCSA might prioritise compactness and separation, leading to better (lower) DBI scores but sacrificing accuracy, as seen in ARI and AMI. VDENCLUE and DBSCAN Conversely, the proposed method, based on prioritising density separation and sparseness and less focused on compactness (hence slightly higher DBI values), is better at capturing the true clusters, leading to better scores in ARI and AMI. This indicates a tradeoff between

compactness and accuracy in clustering, with the proposed method maintaining a more balanced performance across noise levels.

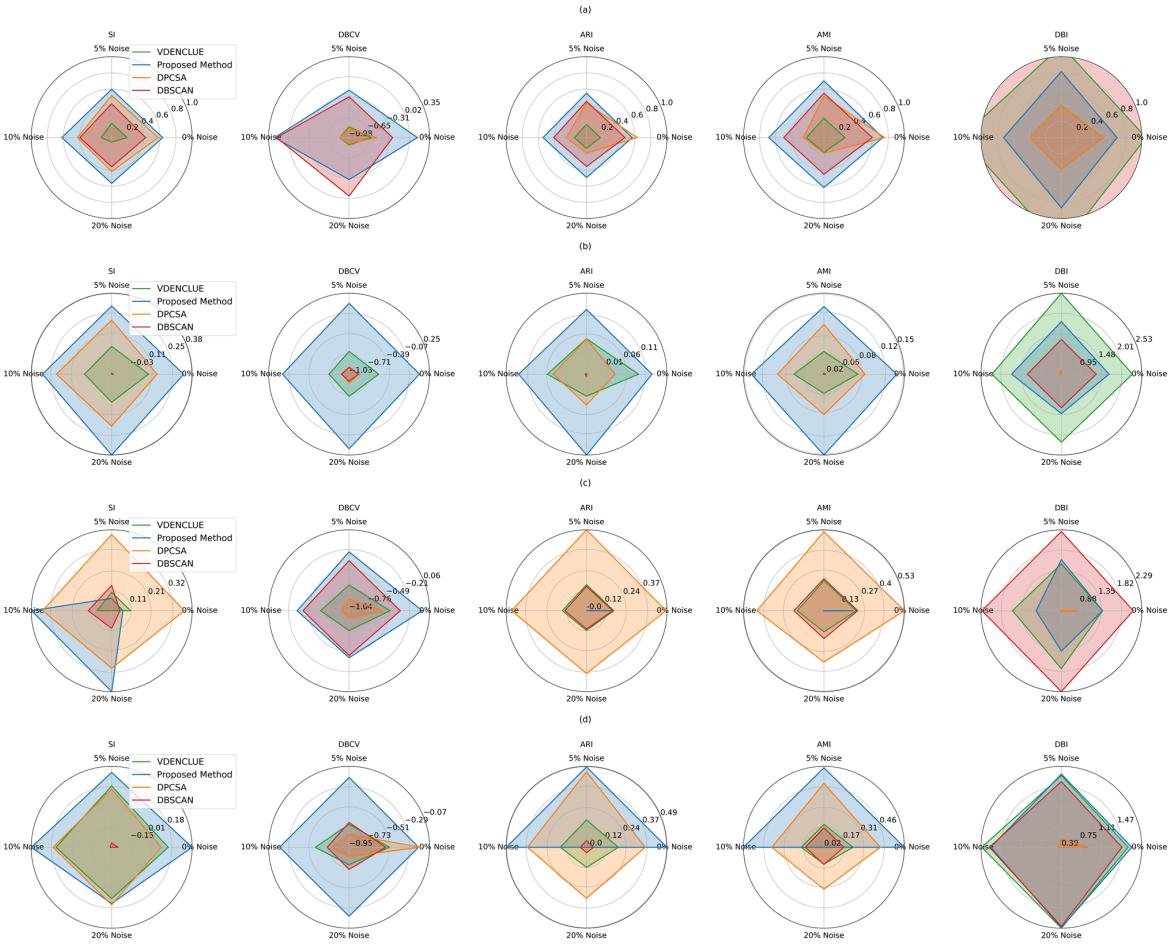


Figure 4. Metric variability across noise levels for real datasets: (**a**) IRIS; (**b**) Heart Disease; (**c**) Seeds; and (**d**) Wine.

The analysis of the DBCV metric reveals interesting patterns. While we observe superior DBCV values compared to the proposed method in the Two Moons dataset for DPCSA and Zahn's Compound for DBSCAN, these methods yield low ARI values, indicating that they are unable to detect the true underlying structure. This is because the DBCV metric alone is insufficient to detect the underlying structure [35]. Therefore, the proposed method's optimisation approach (a combination of DBCV and Cluster Coverage) is better at detecting the underlying density-based structure than the benchmark methods. However, this also opens new avenues to explore other density-based clustering metrics that may prove even more suitable than DBCV alone, offering potential for future investigations.

From a clustering quality perspective, as measured by different metrics, the results largely indicate the proposed method's superiority in SI, ARI, and AMI. It is more accurate in reflecting the true structure of the density-based synthetic data and real-world data. The drop in SI and rise in DBI for some datasets illustrates a common trade-off in clustering: achieving the most compact clusters might not always align with accurately capturing the underlying data structure, as different algorithms vary in how they balance these priorities.

When it comes to the proposed method's resilience to noise, it demonstrates remarkable stability across various noise levels, consistently performing well in terms of the SI, ARI, and AMI across multiple datasets. Compared to other methods, the proposed method either outperforms or matches the best-performing methods across most metrics, making it a reliable choice for clustering under noise. Additionally, the proposed method main-

tains high ARI and AMI values across different noise levels, indicating its robustness in accurately capturing the true clustering structure despite the introduction of noise.

The benchmark methods' inferior results especially under noisy conditions are attributed to the following factors:

a. The clustering performance of benchmark methods' is dependent on user-defined parameters. We have defined appropriate grids for the parameter search of these methods; however, an exhaustive search over the entire parameter space is rarely feasible. Since different parameters produce significantly different clustering outcomes, the criteria for selecting the best results depends on the desired outcome. Typical criteria include clustering validation metric(s) or visual analysis by the user. Our criteria have been to select the results that produced the best ARI (reported in Tables 4 and 5). However, this comes with consequences: the clustering results with the best ARI need not be the best in other validation metrics as well. This is because the parameter values corresponding to the best ARI solution might not be optimal, a phenomenon presented in Figure 5 where the solution with the best ARI suffers from low DBCV and SI, along with deviation from the true cluster structure. In contrast, the proposed method's optimal solution detects the true density-based structure, including density-based clustering validation (DBCV) [34] as part of its objective function. Not only does the proposed approach automatically learn parameters for the optimal clustering solution, but it is also optimised for the density-based structure and does not require the user to select the best solution from multiple options.

b. The proposed method is based on DENCLUE, which has the inherent capability to detect clusters of varying densities even in the presence of noise. DENCLUE, when supplied with optimal parameters estimated through the proposed method, can better detect the underlying structure even in the presence of noise. DBSCAN, on the other hand, struggles with handling clusters of varying densities, which becomes even more challenging in the presence of noise. VDENCLUE, although capable of handling datasets with overlapping clusters due to varying KDE approaches, has inherent challenges when dealing with noisy datasets, as evident from the results shown in Tables 4 and 5. This is due to the method's broader kernel in low-density regions, which may include noise points and incorrectly assign them to clusters. Also, its kernel bandwidth based on the kth nearest neighbour leads to an over-smoothing effect in the noisy regions. Lastly, the DPCSA, although indicating comparable performance to the proposed method, faces challenges with noisy datasets due to two main reasons. DPCSA is sensitive to the cutoff distance; a sub-optimal cutoff value leads to incorrect local density computation, and subsequently misjudgment of noisy points. Moreover, the assignment strategy in DPCSA may lead to increased challenges in noisy datasets; if a point is incorrectly assigned to a cluster, the error can be propagated to other data points.

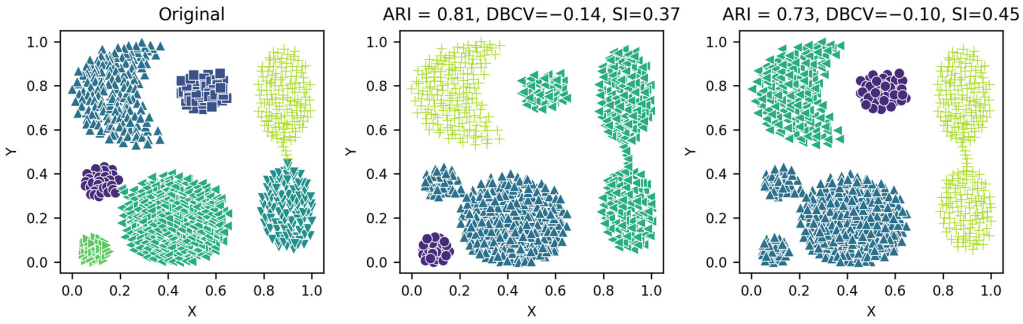


Figure 5. Comparison of clustering results with the original (**left**) and two runs (**middle** and **right**). Each cluster is shown with unique color and symbol. Although the run in the middle has a higher ARI, it compromises in DBCV, Silhouette Index, and True Clusters, demonstrating the trade-offs between different clustering validation criteria.

4. Discussion

The results of the study indicate that the proposed method maintains high clustering quality across varying noise levels, and its performance either surpasses or matches the best-performing method in key metrics such as SI, ARI, and AMI. Traditional clustering algorithms often fail to identify the true underlying structure of the data due to their high sensitivity of input parameters or non-suitability for non-spherical clusters of arbitrary shapes. Even the methods designed to detect the density-based non-spherical clusters often fail under noisy conditions [24,32,33,65,66]. The performance is heavily dependent on selecting appropriate parameters [24,25] and requires domain knowledge or very large grids. Although DBSCAN and DPCSA often perform well for non-noisy datasets when input with appropriate parameters, the classical versions of these algorithms often fail under increased noise levels; some adaptations, such as fuzzy neighbourhoods, potentially improve the stability. In comparison, the proposed method optimised the parameters of DENCLUE without requiring any domain knowledge.

The metrics based on random index and mutual information are impacted by noise because they rely on the accuracy of cluster assignments, which can degrade in the presence of noise [57]. Furthermore, the SI and DBI often degrade at non-Gaussian clusters or clusters of arbitrary shapes, as these metrics are based on the assumption of spherical or compact clusters [26]. The proposed method's optimisation approach is a key aspect of its novelty. While the increased noise levels typically degrade metrics like ARI and AMI [32], the results of the proposed method demonstrate otherwise. By using DBCV as its optimisation objective, which is well-suited for density-based structure, the proposed method successfully identifies the true density-based structure of the data even under noisy conditions, as evident from superior metrics values for metrics such as ARI and AMI across noise levels.

The proposed method was evaluated on low-to-moderate dimensional noisy datasets containing complex clusters of arbitrary shapes, sizes, and densities. The core issue in high-dimensional spaces, a potential area for further study, is the increased sparsity of data which can affect DENCLUE performance in density estimation. However, the proposed method's global search capability due to DE enables it to effectively search the parameter spaces in high-dimensional data. The promising area needs further experiments which will be explored in future extensions of the study.

While the proposed method is generally robust, it performs poorly in complex datasets and at higher noise levels, as seen in Path-based, Seed, and Spiral datasets. Also, the method is based on DE, which introduces additional computational complexity, limiting its use for large-scale datasets or where resources are limited. The method based on DENCLUE limits its applicability to datasets that do not have a clear density-based structure or have a significant cluster overlap, such as Path-based, Zahn's Compound, and heart disease datasets. Future enhancements to the method include improving computational complexity by exploring parallel implementations, thus making it applicable to large-scale, high-dimensional, real-world datasets. Additionally, adaptive variants of DE will be implemented instead of the fixed preset control variables. Finally, more computationally efficient alternatives to DBCV and other density-sensitive clustering metrics will be explored.

5. Conclusions

The proposed DE-based DENCLUE method established its effectiveness in maintaining high-quality clustering in noisy conditions for diverse datasets. The method's optimisation capabilities, particularly through its objective function based on density-based clustering validation (DBCV), enable it to adapt to the dataset characteristics and identify true clusters even in the presence of noise. The method proves to be a robust clustering tool by consistently outperforming other density-based methods such as DBSCAN, DPCSA, and VDENCLUE not only in traditional clustering validation metrics like SI, ARI, and AMI but also in density-focused validation metric DBCV.

While the proposed DE-based DENCLUE method maintains its performance for diverse datasets, its increased computational requirements due to DE may limit its applicability to large-scale, high-dimensional datasets. Additionally, although the DE's control parameters are tuned based on a sensitivity analysis, there may still be some sensitivity to more diverse datasets which can affect the method's efficiency for large-scale high dimensional datasets.

The future enhancements of the proposed method will explore parallel implementations for improved efficiency. The proposed method's challenge with datasets with significant cluster overlap can be addressed by utilising soft clustering approaches, which allow for data points to be assigned to different clusters with various degrees of membership. Furthermore, exploring adaptive approaches in the optimisation process, wherein DE parameters such as F and CR are modified dynamically based on the algorithm's progression may potentially improve the method's efficiency. There also exists an opportunity to include more density-based objectives into the study in addition to DBCV, especially those that are more computationally efficient and capable of performing with datasets having significant cluster overlap.

Author Contributions: Conceptualization, O.A., S.M., H.A. and S.A.; methodology, O.A., S.M. and H.A.; validation, O.A., S.M., H.A., M.A.A. and S.A.; supervision, S.M. and H.A.; investigation, O.A., M.A.A. and S.A.; data curation, O.A.; writing—original draft preparation, O.A.; writing—review and editing, O.A. and M.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: Shahzad Mumtaz was supported by the University of Aberdeen start-up fund (CF10834-23).

Data Availability Statement: The original data presented in this study are openly available at http://cs.uef.fi/sipu/datasets/ (accessed on 29 August 2024) and https://archive.ics.uci.edu (accessed on 29 August 2024). The code is available under CC BY 4.0 licence at https://doi.org/10.6084/m9.figshare.26543863.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

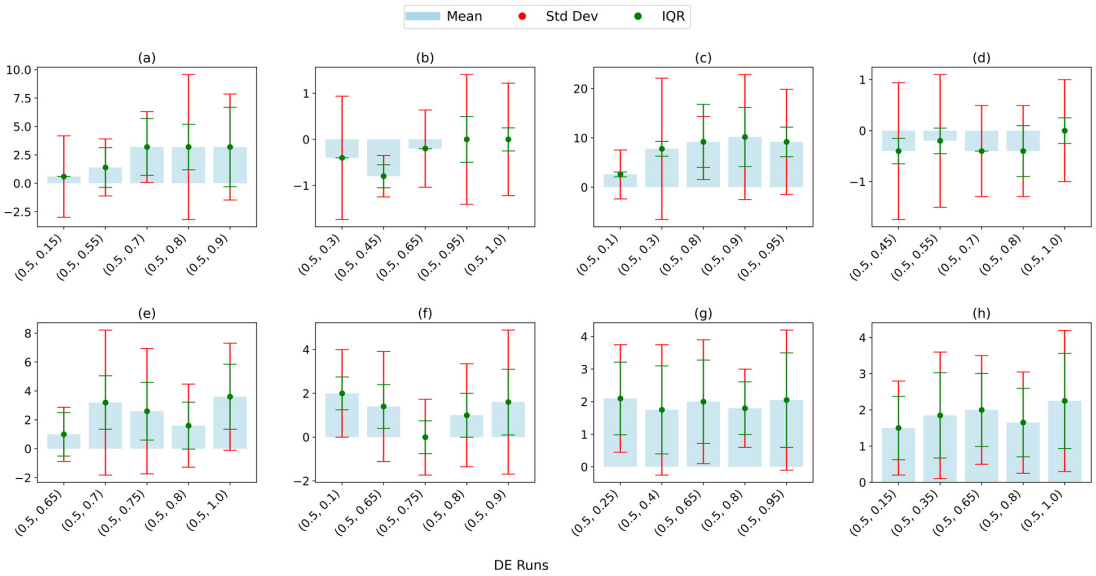


Figure A1. Comparison of different DE control parameters—synthetic datasets. (**a**) Aggregation; (**b**) Two Moons; (**c**) Path-based; (**d**) Shapes; (**e**) Spiral; (**f**) Zahn's Compound; (**g**) S; (**h**) A3. The x-axis in the subplots indicates specific combinations of DE control parameters (F and CR), where the first value in the pair represents F, and the second value represents CR. The y-axis indicates the mean generations to reach the threshold objective value.

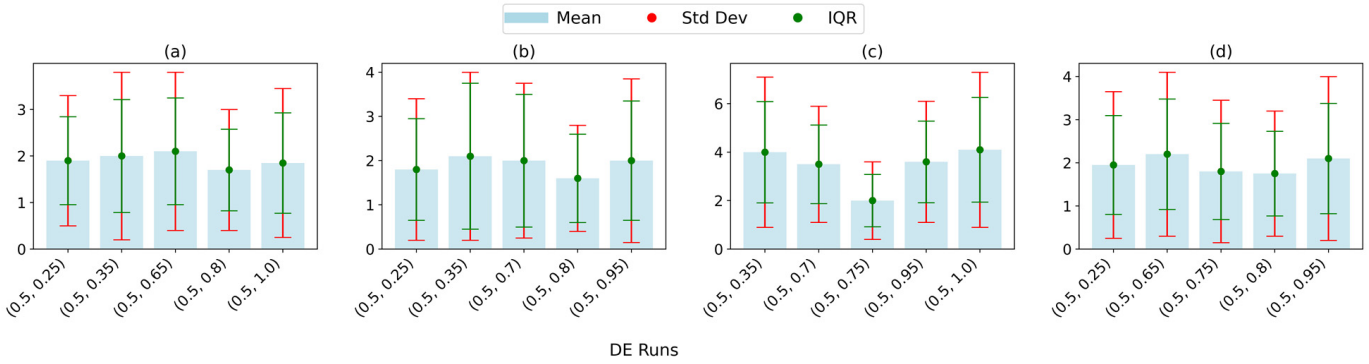


Figure A2. Comparison of different DE control parameters—real datasets. (**a**) IRIS; (**b**) Heart Disease; (**c**) Seeds; (**d**) Wine. The x-axis in the subplots indicates specific combinations of DE control parameters (F and CR), where the first value in the pair represents F, and the second value represents CR. The y-axis indicates the mean generations to reach the threshold objective value.

Distribution of Metric Values

During the experiments, each method was run multiple times for all noise levels; each run had different values from within the parameters' ranges defined for the method. The clustering produced by a method in each run was evaluated for quality using the SI, DBI, ARI, and AMI metrics. Figures A3–A12 present the reader with the distribution of the values produced in different runs. This enhances readers' understanding of how different methods behave for different parameter values and where most of the metric values lie. The X-axis in Figures A3–A12 represents noise levels, while the Y-axis shows metric values. It can be observed from the distribution plots that the spread of metrics' values for DBSCAN in the Aggregation, IRIS, Heart Disease and Seeds datasets vary significantly, highlighting its sensitivity to input parameters. Similarly, DPCSA shows large variations for IRIS, Heart Disease and Wine datasets, while VDENCLUE shows significant variations in Heart Disease and Wine datasets.

Additionally, Figures A13–A22 present heatmaps for readers' enhanced understanding through visual representation of the performance of different clustering methods across different metrics and noise levels. The cells in the heatmap display metrics' values, and the colour intensity indicates the magnitude, allowing for a quick comparison of methods' performance side-by-side across multiple metrics and noise levels.

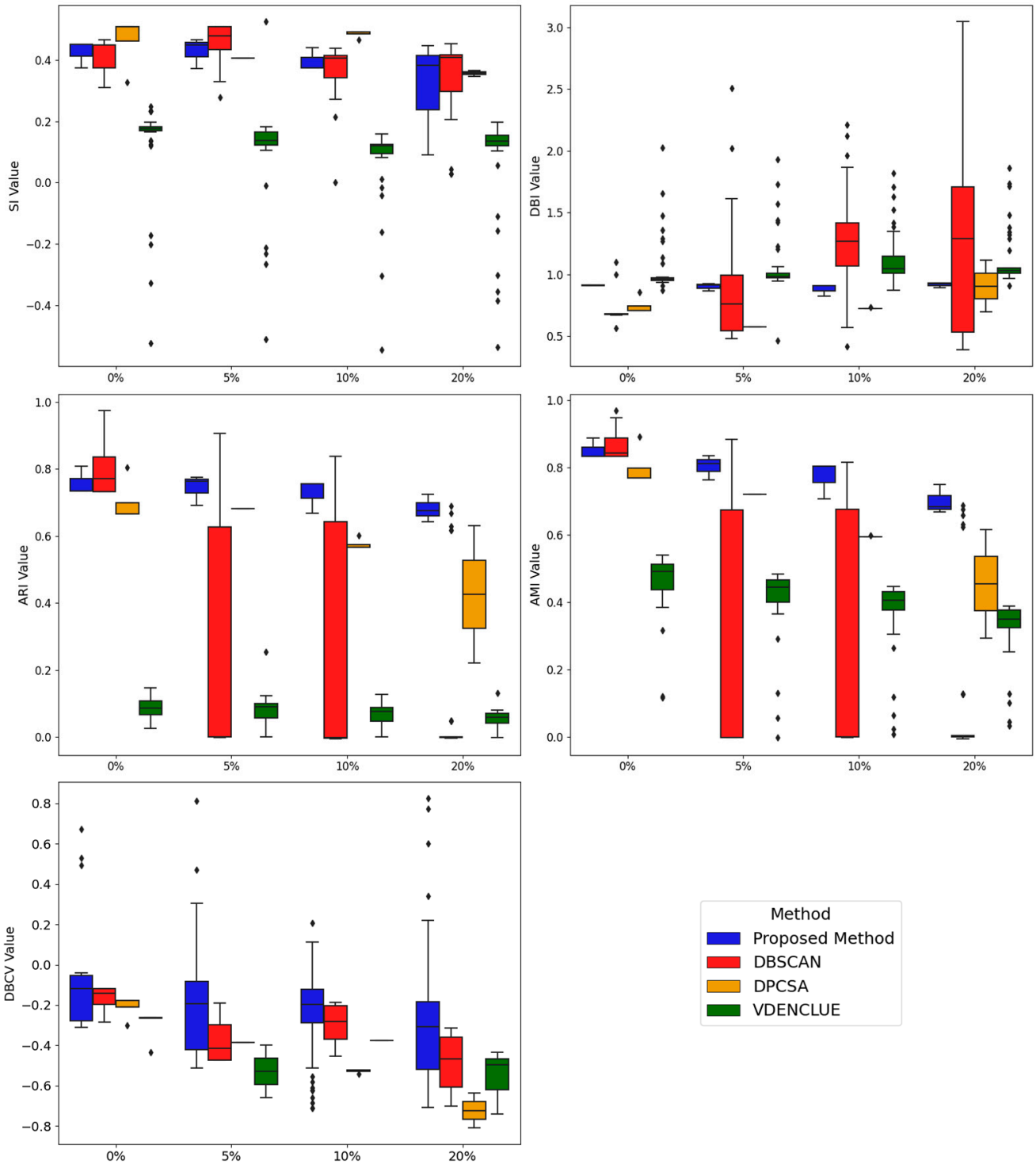


Figure A3. Distribution of metric values for Aggregation dataset across noise levels. The outliers are depicted as isolated points.

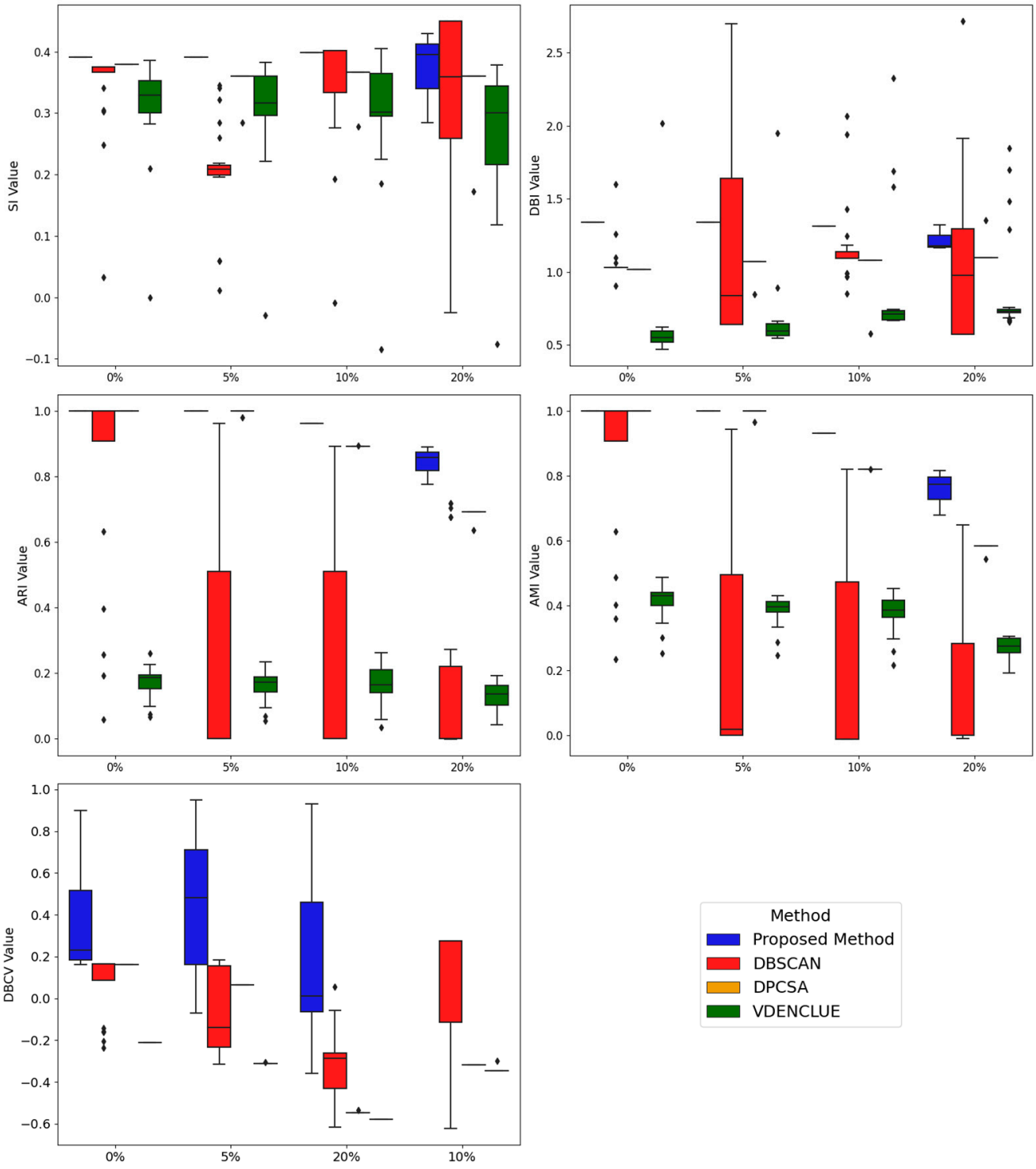


Figure A4. Distribution of metric values for Two Moons dataset across noise levels. The outliers are depicted as isolated points.

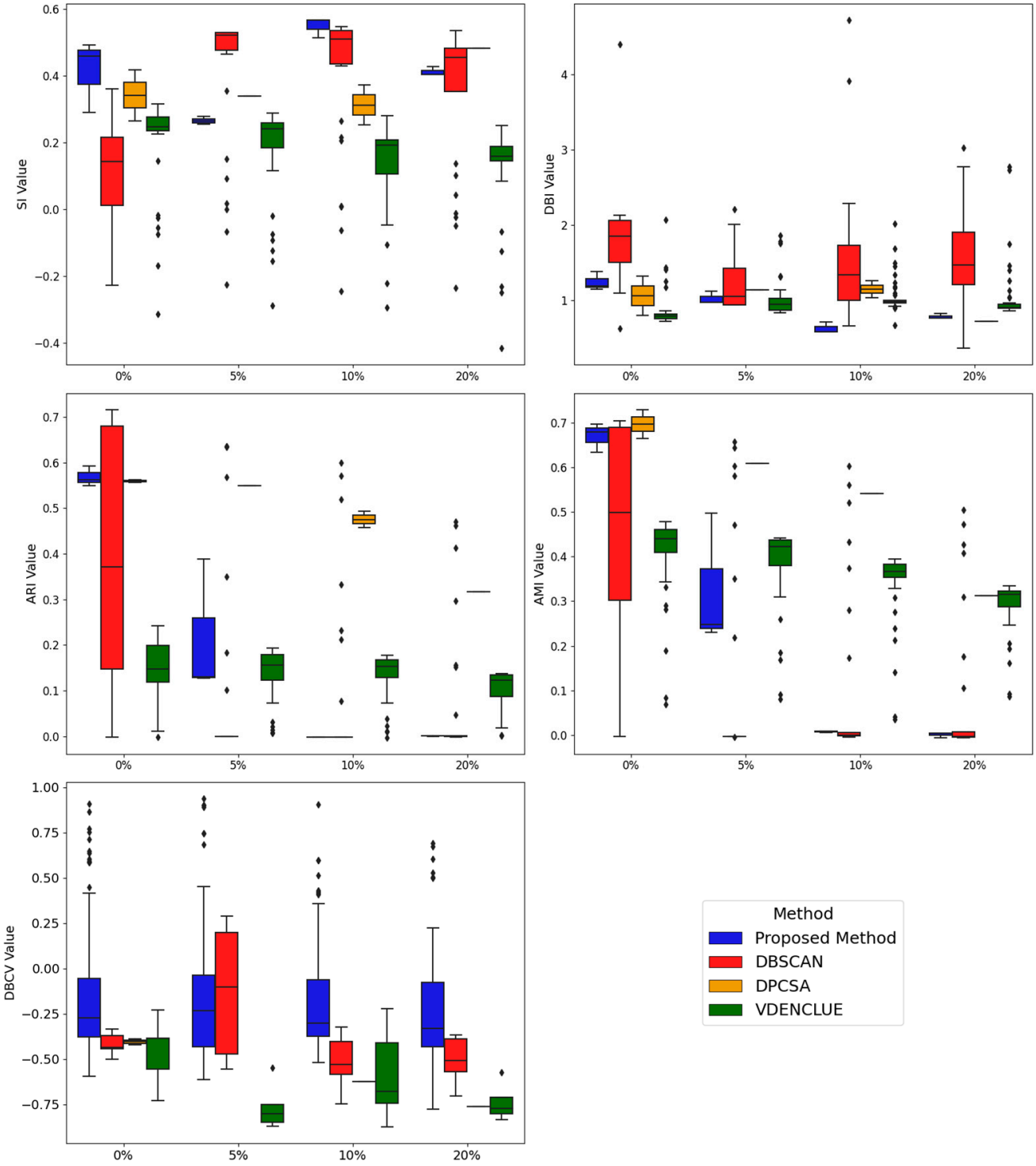


Figure A5. Distribution of metric values for Path-based dataset across noise levels. The outliers are depicted as isolated points.

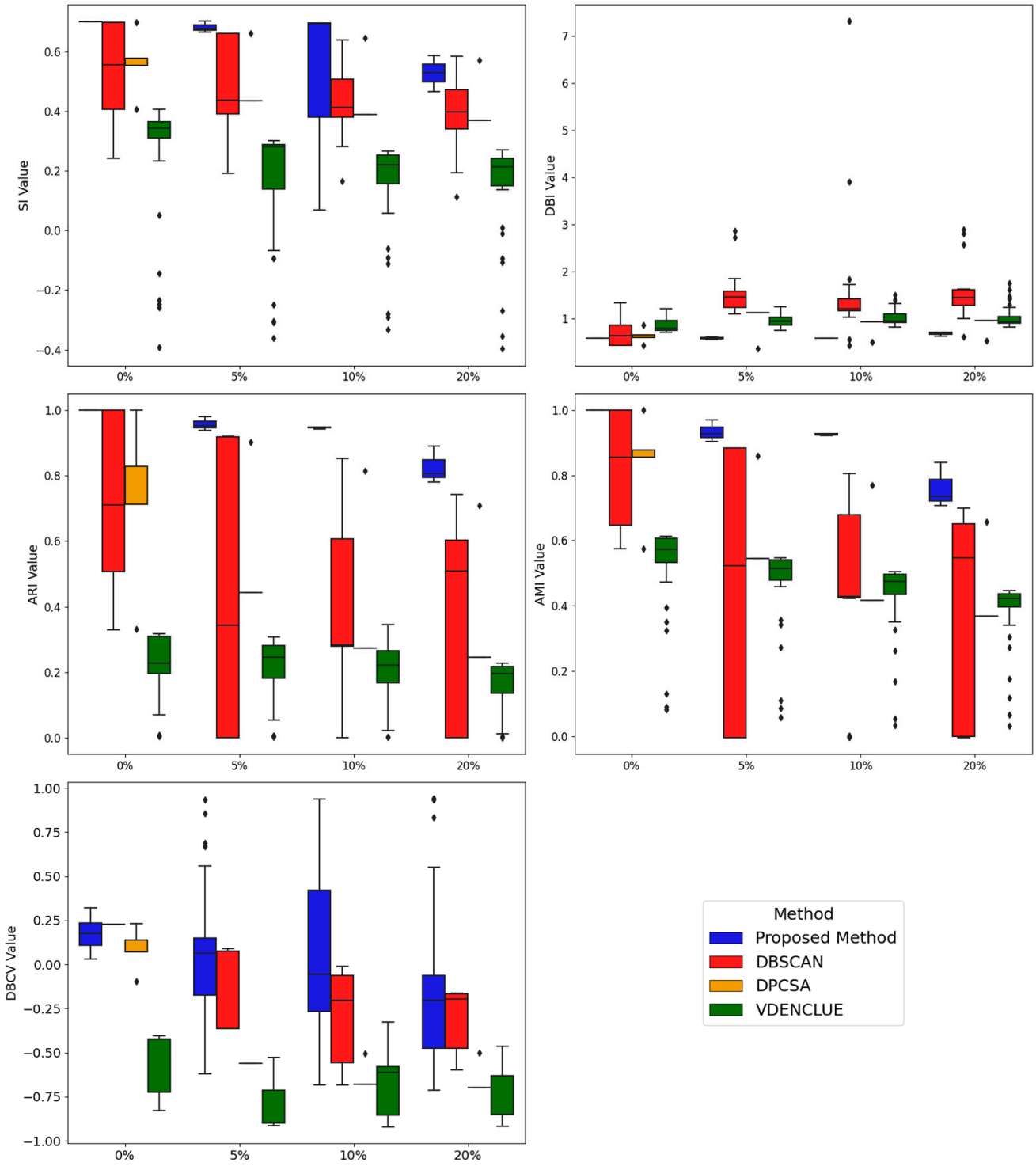


Figure A6. Distribution of metric values for Shapes dataset across noise levels. The outliers are depicted as isolated points.

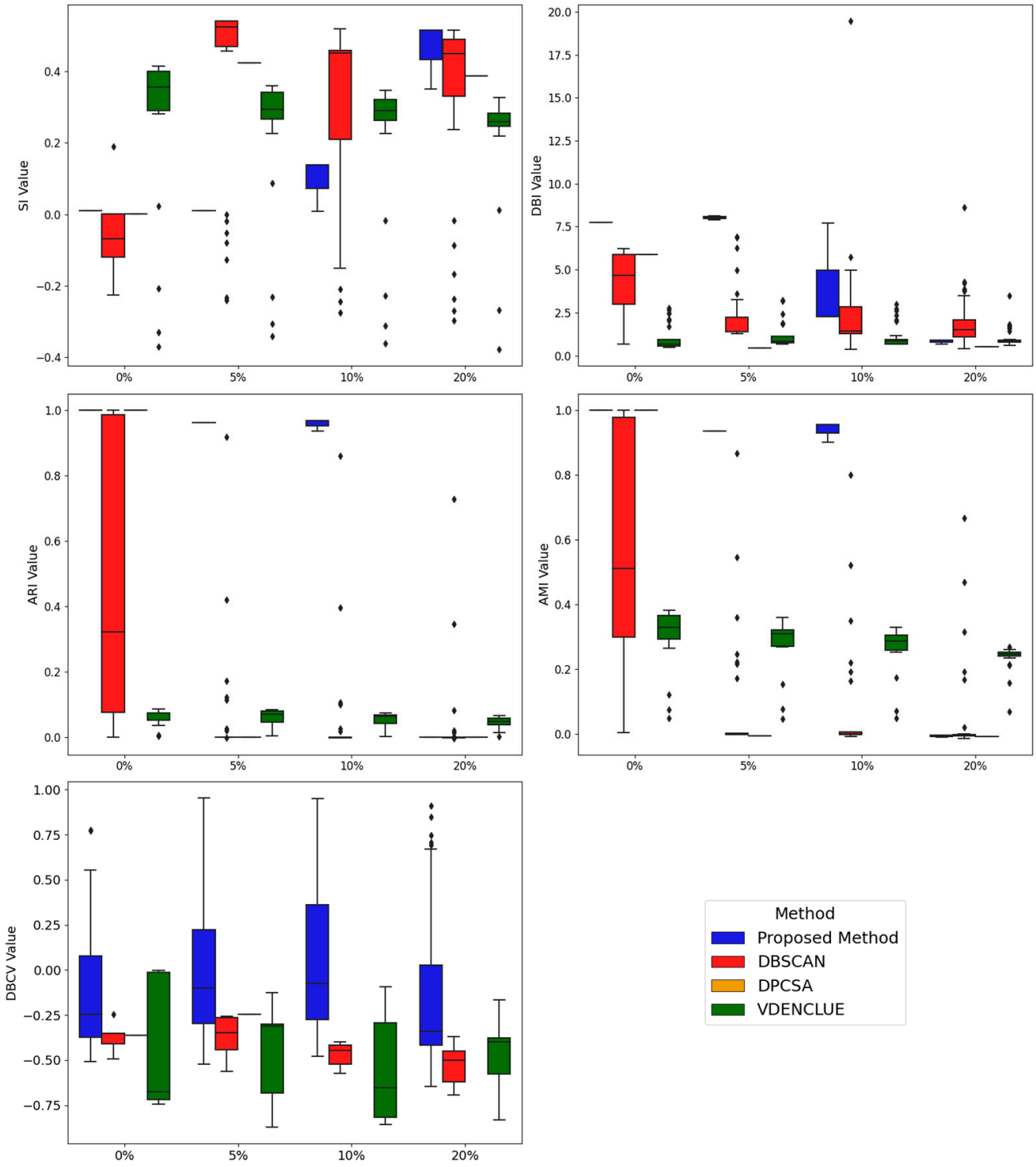


Figure A7. Distribution of metric values for Spiral dataset across noise levels. The outliers are depicted as isolated points.

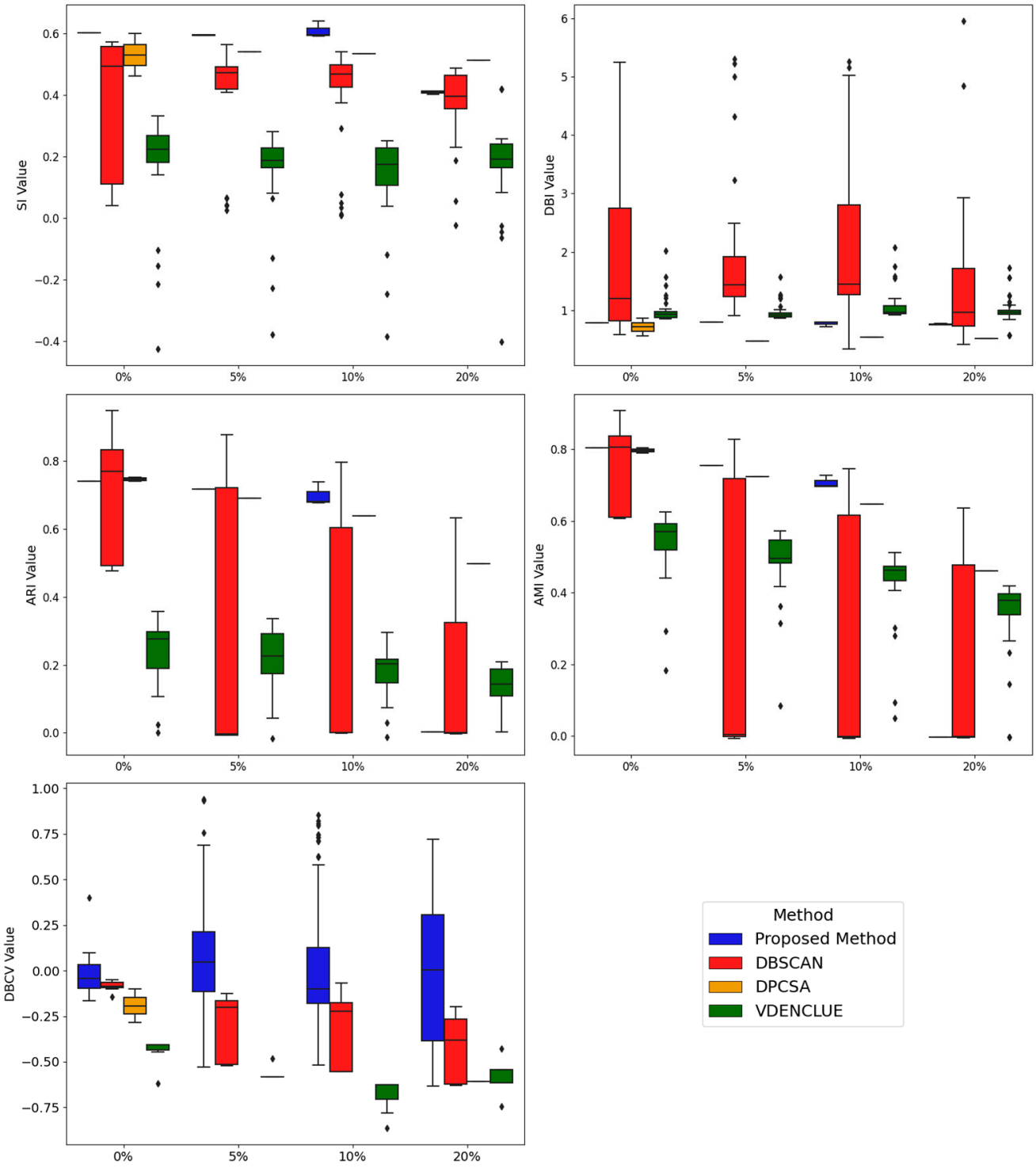


Figure A8. Distribution of metric values for Zahn's Compound dataset across noise levels. The outliers are depicted as isolated points.

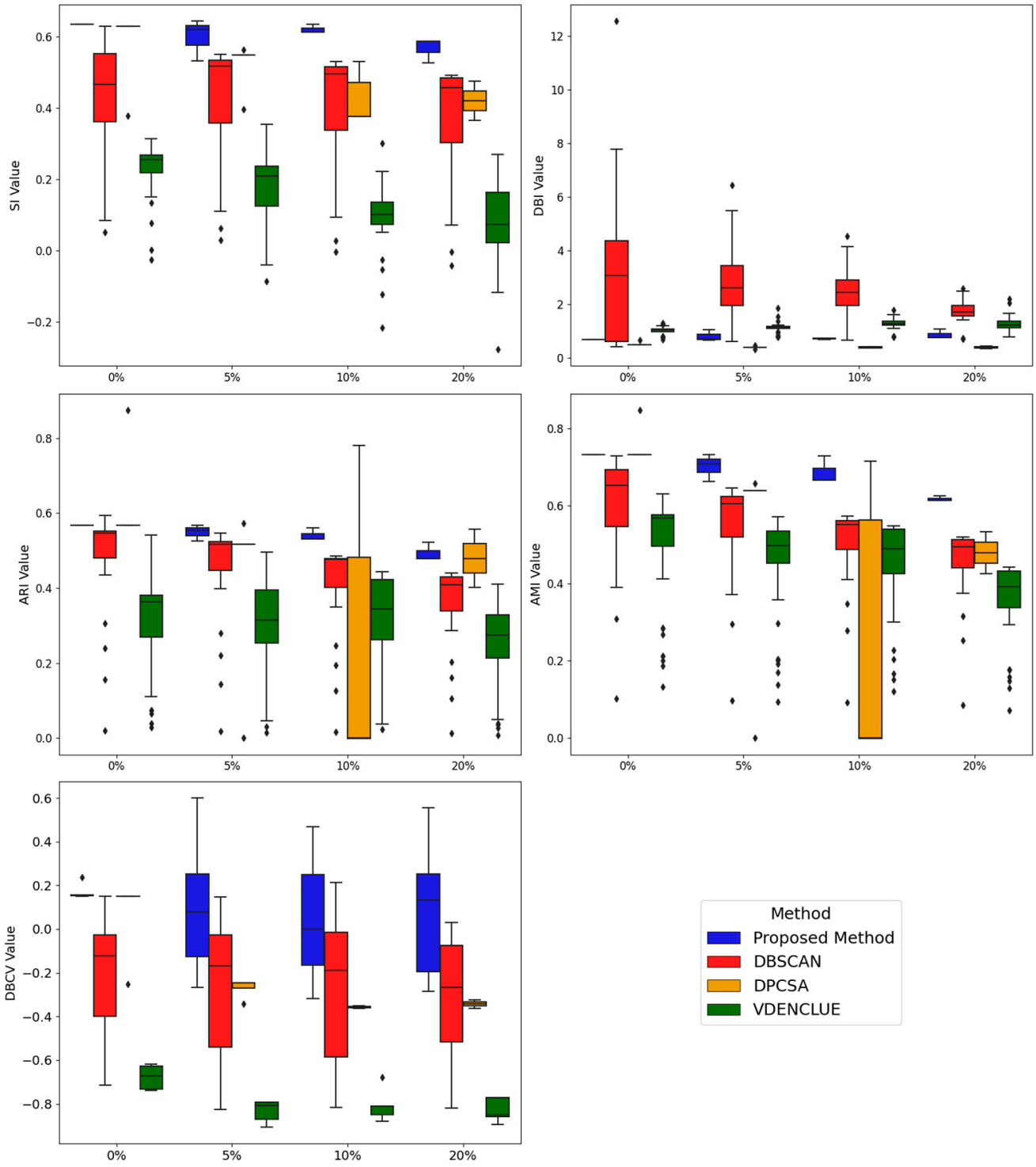


Figure A9. Distribution of metric values—IRIS across noise levels. The outliers are depicted as isolated points.

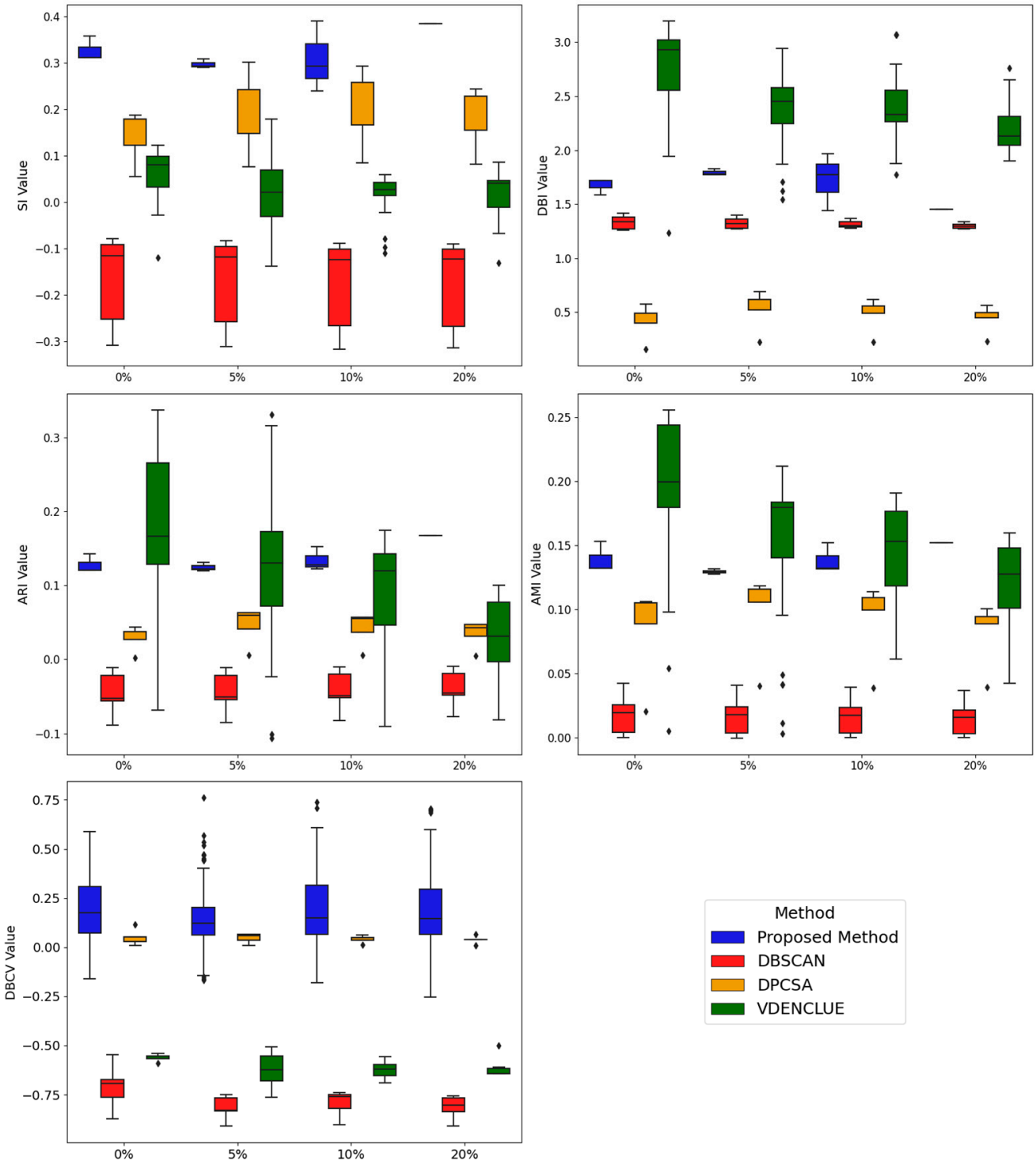


Figure A10. Distribution of metric values—Heart Disease across noise levels. The outliers are depicted as isolated points.

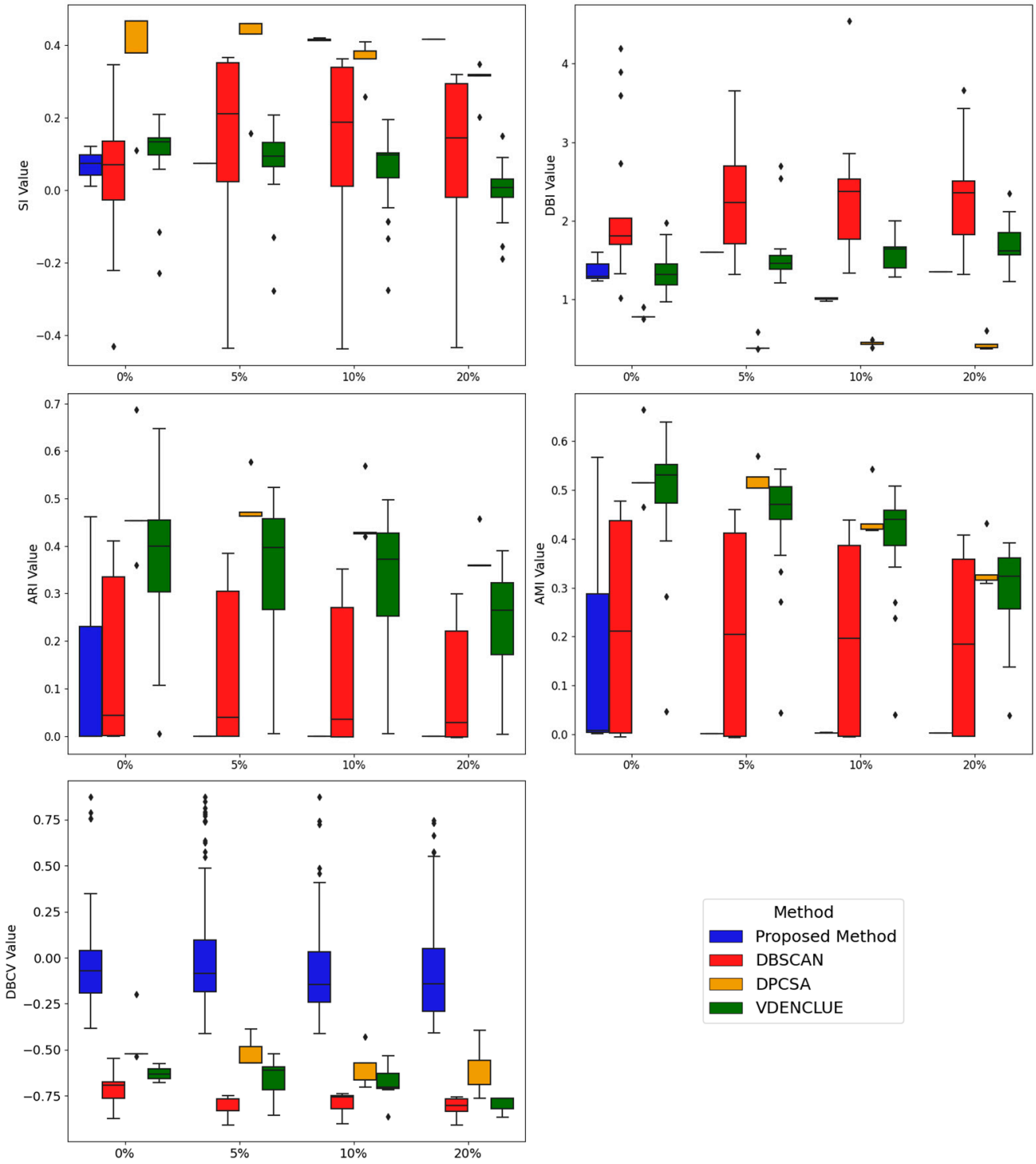


Figure A11. Distribution of metric values—Seeds across noise levels. The outliers are depicted as isolated points.

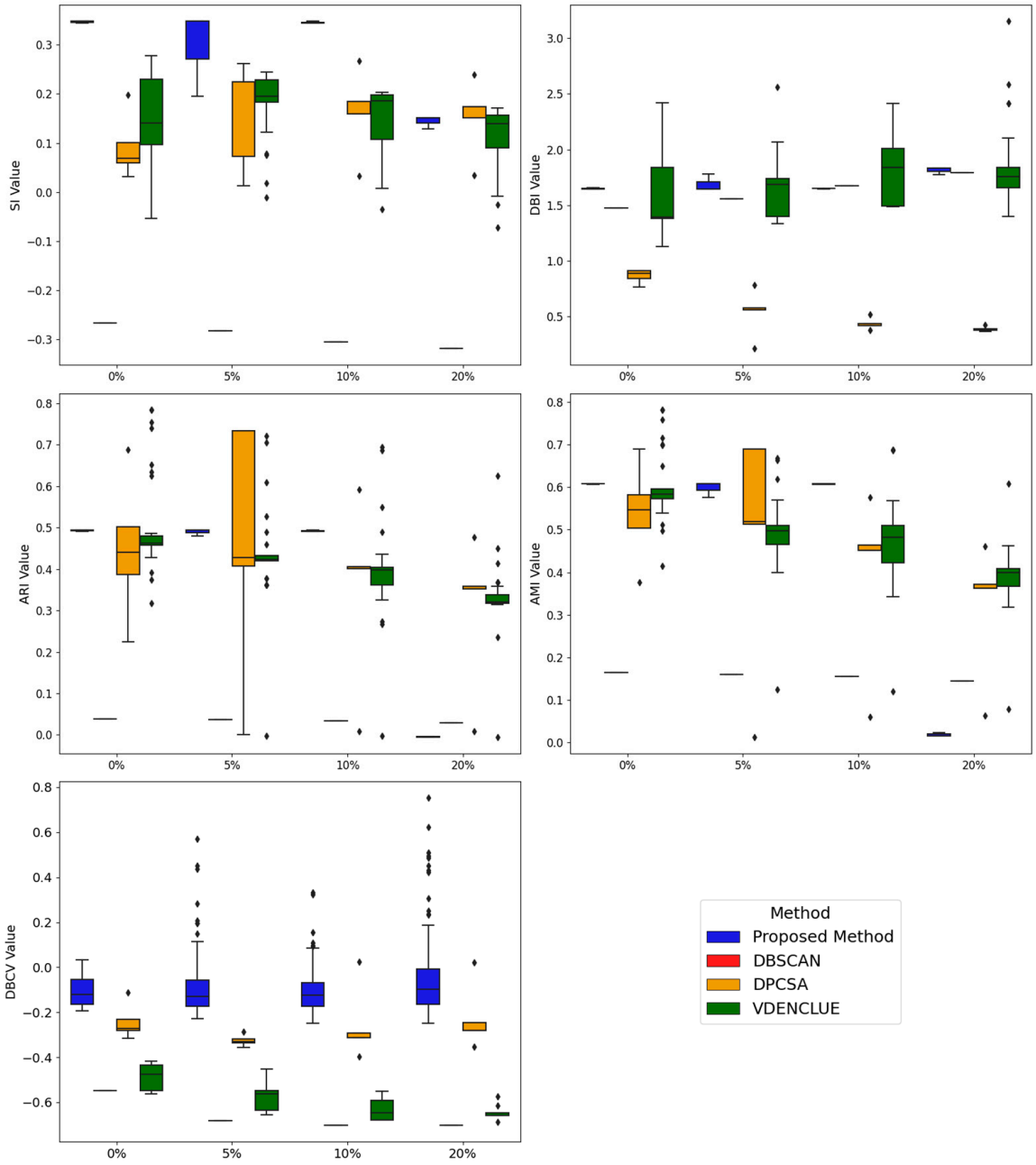


Figure A12. Distribution of metric values—Wine across noise levels. The outliers are depicted as isolated points.

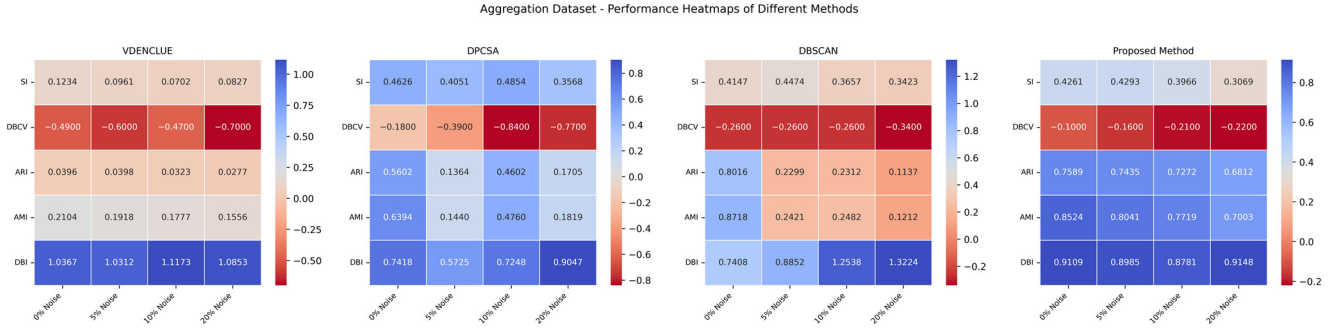


Figure A13. Benchmark vs. proposed method clustering heatmaps for Aggregation dataset.

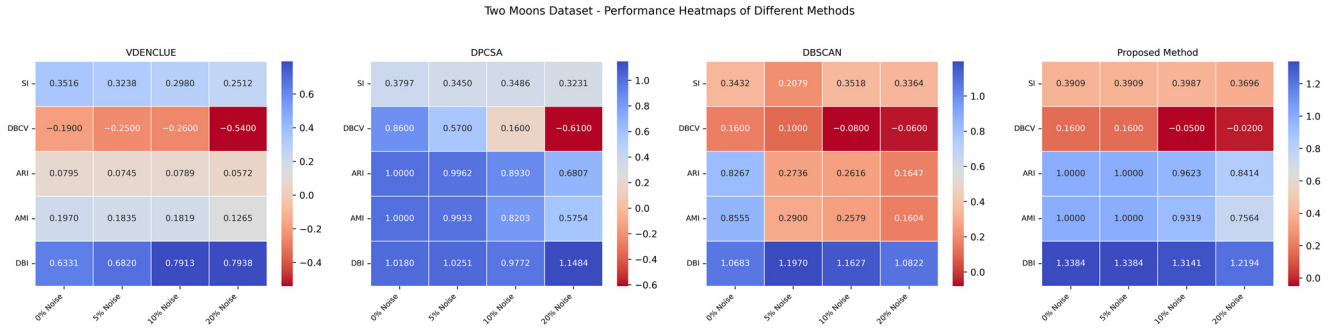


Figure A14. Benchmark vs. proposed method clustering heatmaps for Two Moons dataset.

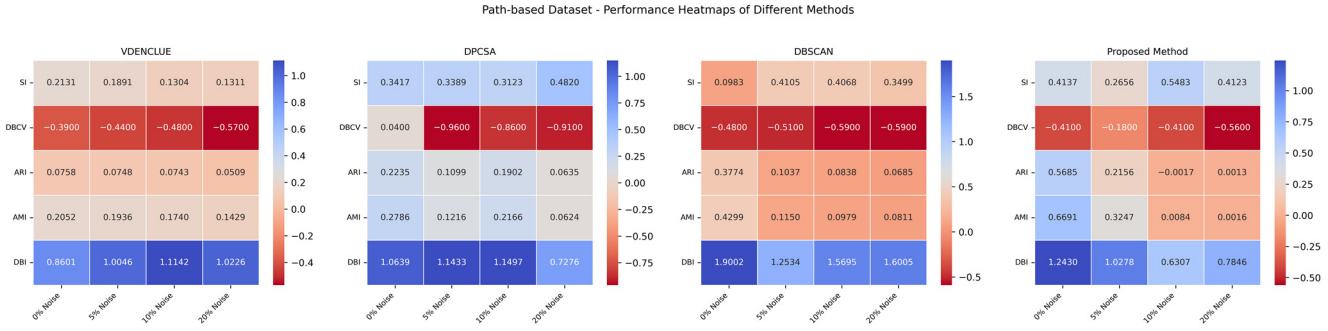


Figure A15. Benchmark vs. proposed method clustering heatmaps for Path-based dataset.

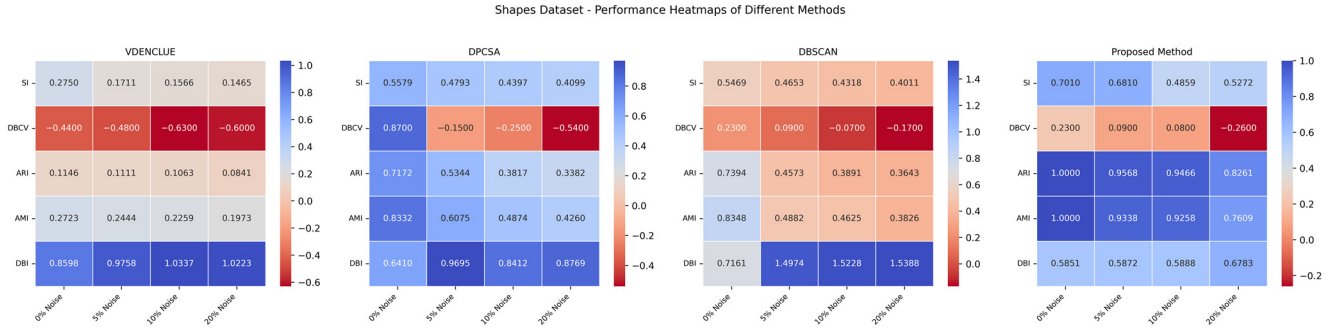


Figure A16. Benchmark vs. proposed method clustering heatmaps for Shapes dataset.

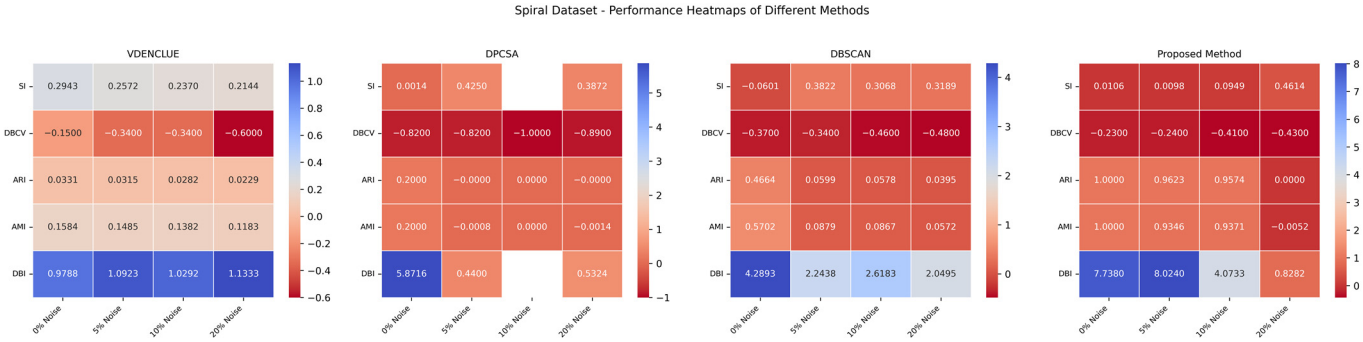


Figure A17. Benchmark vs. proposed method clustering heatmaps for Spiral dataset.

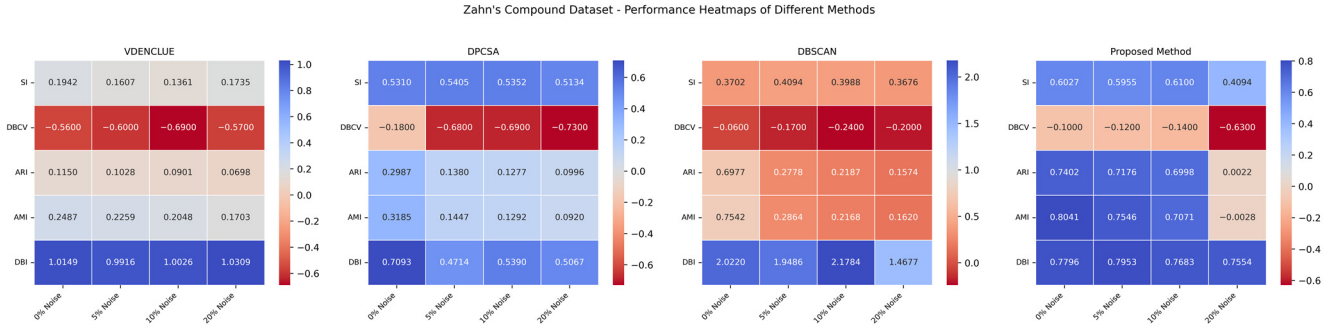


Figure A18. Benchmark vs. proposed method clustering heatmaps for Zahn's Compound dataset.

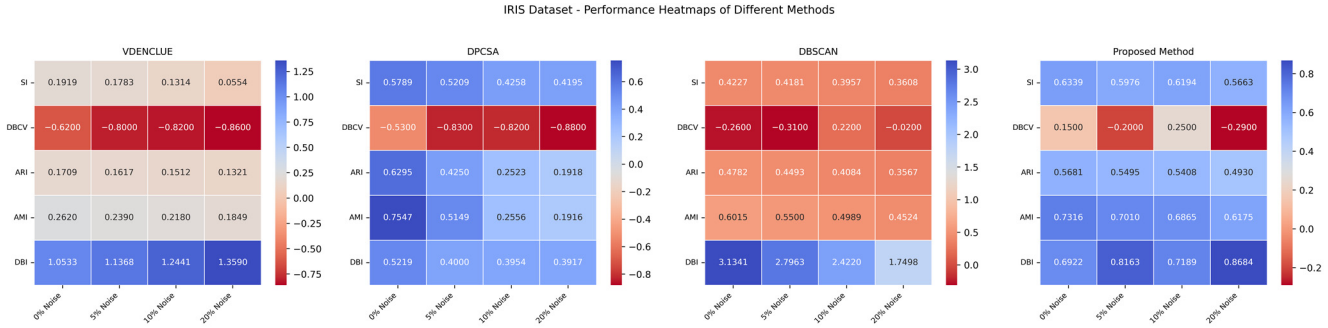


Figure A19. Benchmark vs. proposed method clustering heatmaps for Iris dataset.

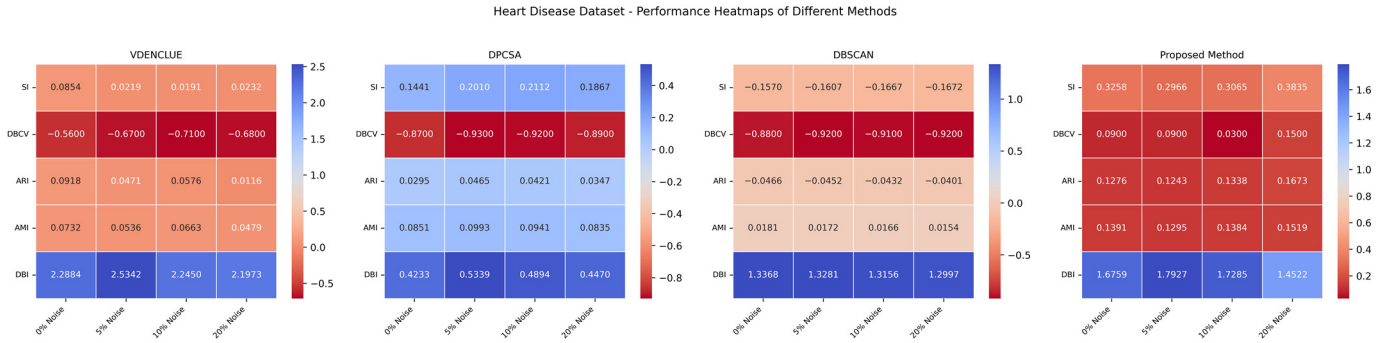


Figure A20. Benchmark vs. proposed method clustering heatmaps for Heart Disease dataset.

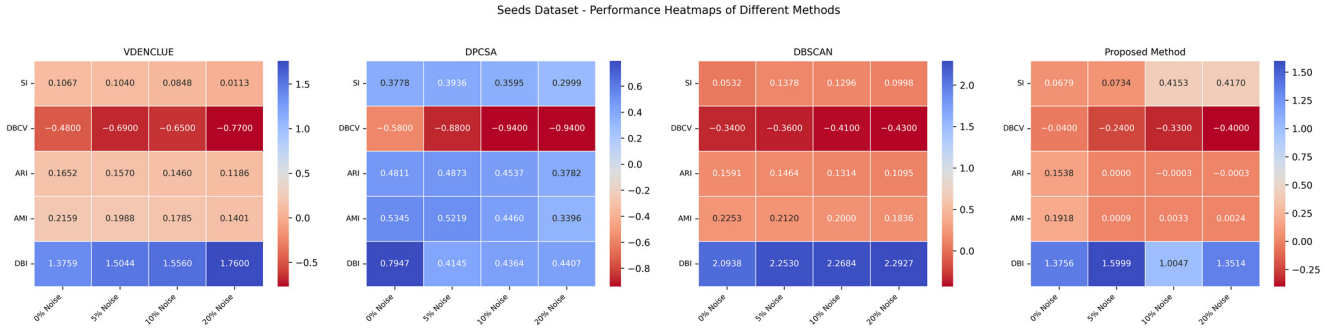


Figure A21. Benchmark vs. proposed method clustering heatmaps for Seeds dataset.

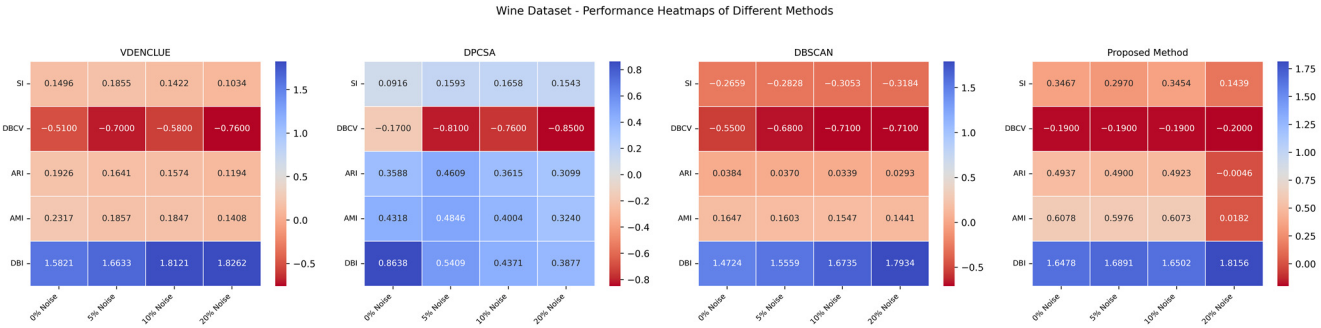


Figure A22. Benchmark vs. proposed method clustering heatmaps for Wine dataset.

Appendix B

Statistical Analysis of Method Comparisons

To analyse the statistical significance of the differences in performance between the proposed method and the benchmark clustering algorithms (VDENCLUE, DPCSA, and DBSCAN), the Wilcoxon signed-rank test and the Friedman test were considered. Both are non-parametric tests suitable for comparing algorithm performance across datasets [67,68]. However, the Wilcoxon signed-rank test was chosen because it is more sensitive to small differences in performance and better suited for cases with a smaller number of datasets. On the other hand, the Friedman test primarily focuses on ranking algorithms and is less sensitive to detecting subtle differences in pairwise comparisons.

The analysis was performed across noise levels and dataset groups using clustering validation metrics. To control for Type I error due to multiple comparisons, a Bonferroni correction was applied, adjusting the significance threshold to $p < 0.0167$ (0.05/3). This correction is made as three pairwise comparisons were conducted for each dataset group and noise level. Tables A1–A5 present the p-values for each noise level and dataset group, indicating whether the differences in performance are statistically significant. However, it is important to distinguish between statistical significance (shown by p-values) and practical significance (real-world impact of the differences). For instance, although p-values slightly above the threshold (e.g., $p = 0.02$ or $p = 0.03$) do not meet the strict Bonferroni-corrected threshold for statistical significance, the differences are small and likely practically insignificant. While there may be minor variations in performance, they may not always lead to a meaningful difference in clustering results in practice.

The results indicate that the proposed method outperforms algorithms like VDENCLUE and DBSCAN in clustering metrics such as ARI and AMI, across various noise levels, indicating better noise resilience, especially for synthetic datasets with complex structures.

However, the p-values for the comparison with DPCSA for the DBI metric are high. This is due to the fact that DPCSA focuses on compactness, resulting in lower DBI values. In contrast, the proposed method prioritises identifying true cluster structures (as reflected by metrics such as ARI) over achieving the compact, well-separated clusters measured by DBI.

Table A1. Wilcoxon signed-rank test p-values for each noise level and dataset group (SI metric). A Bonferroni correction was applied to adjust for Type I errors due to multiple comparisons. The significance threshold was set at $p < 0.0167$ (0.05/3) to account for three pairwise comparisons in each group. Comparisons are considered significant if the p-value is below this threshold.

		Proposed vs.			
Noise Level	**Dataset Group**	**VDENCLUE**	**DPCSA**	**DBSCAN**	**Significance ($p < 0.0167$)**
0	Group 1 (Excluding datasets with significant overlap)	0.0651	0.0651	0.0156	Yes (DBSCAN)
0	Group 2 (Only datasets with significant overlap)	0.5	1.0	0.25	No
0	Group 3 (All datasets)	0.0371	0.1601	0.0019	Yes (DBSCAN)

Table A1. *Cont.*

		Proposed vs.			
Noise Level	**Dataset Group**	**VDENCLUE**	**DPCSA**	**DBSCAN**	**Significance ($p < 0.0167$)**
5	Group 1 (Excluding datasets with significant overlap)	0.0651	0.2968	0.2968	No
5	Group 2 (Only datasets with significant overlap)	0.5	0.75	1.0	No
5	Group 3 (All datasets)	0.0163	0.6953	0.1933	Yes (VDENCLUE)
10	Group 1 (Excluding datasets with significant overlap)	0.0468	0.2187	0.1562	No
10	Group 2 (Only datasets with significant overlap)	0.25	0.25	0.25	No
10	Group 3 (All datasets)	0.0058	0.0390	0.0273	Yes (VDENCLUE)
20	Group 1 (Excluding datasets with significant overlap)	0.0156	0.4687	0.0468	Yes (VDENCLUE)
20	Group 2 (Only datasets with significant overlap)	0.25	0.5	0.25	No
20	Group 3 (All datasets)	0.0019	0.1933	0.0058	Yes (Except DPCSA)

Table A2. Wilcoxon signed-rank test *p*-values for each noise level and dataset group (DBI metric). A Bonferroni correction was applied to adjust for Type I errors due to multiple comparisons. The significance threshold was set at $p < 0.0167$ (0.05/3) to account for three pairwise comparisons in each group. Comparisons are considered significant if the *p*-value is below this threshold.

		Proposed vs.			
Noise Level	**Dataset Group**	**VDENCLUE**	**DPCSA**	**DBSCAN**	**Significance ($p < 0.0167$)**
0	Group 1 (Excluding datasets with significant overlap)	1.0	0.0312	0.8125	No
0	Group 2 (Only datasets with significant overlap)	0.75	0.25	0.5	No
0	Group 3 (All datasets)	1.0	0.0039	0.7695	Yes (DPCSA)
5	Group 1 (Excluding datasets with significant overlap)	1.0	0.1093	0.9375	No
5	Group 2 (Only datasets with significant overlap)	1.0	0.5	0.75	No
5	Group 3 (All datasets)	0.7695	0.0273	0.5566	No
10	Group 1 (Excluding datasets with significant overlap)	0.8125	0.1562	0.375	No
10	Group 2 (Only datasets with significant overlap)	0.25	0.5	0.5	No
10	Group 3 (All datasets)	0.3222	0.1289	0.2324	No
20	Group 1 (Excluding datasets with significant overlap)	0.2187	0.0651	0.0651	No
20	Group 2 (Only datasets with significant overlap)	0.25	0.25	0.5	No
20	Group 3 (All datasets)	0.04882	0.0136	0.0273	Yes (DPCSA)

Table A3. Wilcoxon signed-rank test *p*-values for each noise level and dataset group (ARI metric). A Bonferroni correction was applied to adjust for Type I errors due to multiple comparisons. The significance threshold was set at $p < 0.0167$ (0.05/3) to account for three pairwise comparisons in each group. Comparisons are considered significant if the *p*-value is below this threshold.

		Proposed vs.			
Noise Level	**Dataset Group**	**VDENCLUE**	**DPCSA**	**DBSCAN**	**Significance ($p < 0.0167$)**
0	Group 1 (Excluding datasets with significant overlap)	0.0156	0.0463	0.0468	Yes (VDENCLUE)
0	Group 2 (Only datasets with significant overlap)	0.5	0.75	0.5	No
0	Group 3 (All datasets)	0.0039	0.0663	0.0136	Yes (Except DPCSA)
5	Group 1 (Excluding datasets with significant overlap)	0.0156	0.01562	0.0156	Yes
5	Group 2 (Only datasets with significant overlap)	1.0	1.0	0.75	No
5	Group 3 (All datasets)	0.0097	0.0371	0.0097	Yes (Except DPCSA)
10	Group 1 (Excluding datasets with significant overlap)	0.0156	0.0156	0.0156	Yes
10	Group 2 (Only datasets with significant overlap)	0.75	0.5	1.0	No
10	Group 3 (All datasets)	0.0136	0.1054	0.0097	Yes (Except DPCSA)
20	Group 1 (Excluding datasets with significant overlap)	0.2187	0.296	0.2968	No
20	Group 2 (Only datasets with significant overlap)	1.0	0.75	1.0	No
20	Group 3 (All datasets)	0.2324	0.4921	0.2753	No

Table A4. Wilcoxon signed-rank test *p*-values for each noise level and dataset group (AMI metric). A Bonferroni correction was applied to adjust for Type I errors due to multiple comparisons. The significance threshold was set at $p < 0.0167$ (0.05/3) to account for three pairwise comparisons in each group. Comparisons are considered significant if the *p*-value is below this threshold.

		Proposed vs.			
Noise Level	**Dataset Group**	**VDENCLUE**	**DPCSA**	**DBSCAN**	**Significance ($p < 0.0167$)**
0	Group 1 (Excluding datasets with significant overlap)	0.0156	0.0463	0.0312	Yes (VDENCLUE)
0	Group 2 (Only datasets with significant overlap)	0.5	0.75	0.5	No
0	Group 3 (All datasets)	0.0039	0.0663	0.0097	Yes (Except DPCSA)
5	Group 1 (Excluding datasets with significant overlap)	0.0156	0.0156	0.0156	Yes
5	Group 2 (Only datasets with significant overlap)	1.0	1.0	1.0	No
5	Group 3 (All datasets)	0.0097	0.0371	0.0136	Yes (Except DPCSA)
10	Group 1 (Excluding datasets with significant overlap)	0.0156	0.0156	0.0156	Yes
10	Group 2 (Only datasets with significant overlap)	0.5	0.5	0.75	No
10	Group 3 (All datasets)	0.0165	0.1308	0.0165	Yes (Except DPCSA)
20	Group 1 (Excluding datasets with significant overlap)	0.2187	0.2968	0.2187	No
20	Group 2 (Only datasets with significant overlap)	0.5	0.75	0.75	No
20	Group 3 (All datasets)	0.4921	0.5566	0.375	No

Table A5. Wilcoxon signed-rank test *p*-values for each noise level and dataset group (DBCV metric). A Bonferroni correction was applied to adjust for Type I errors due to multiple comparisons. The significance threshold was set at $p < 0.0167$ (0.05/3) to account for three pairwise comparisons in each group. Comparisons are considered significant if the *p*-value is below this threshold.

		Proposed vs.			
Noise Level	**Dataset Group**	**VDENCLUE**	**DPCSA**	**DBSCAN**	**Significance ($p < 0.0167$)**
0	Group 1 (Excluding datasets with significant overlap)	0.0312	0.6875	0.0312	No
0	Group 2 (Only datasets with significant overlap)	0.25	0.25	0.5	No
0	Group 3 (All datasets)	0.0039	0.1933	0.1308	Yes (VDENCLUE)
5	Group 1 (Excluding datasets with significant overlap)	0.0156	0.6875	0.0468	Yes (VDENCLUE)
5	Group 2 (Only datasets with significant overlap)	0.5	0.5	1.0	No
5	Group 3 (All datasets)	0.0039	0.25	0.1054	Yes (VDENCLUE)
10	Group 1 (Excluding datasets with significant overlap)	0.0312	0.625	0.0312	No
10	Group 2 (Only datasets with significant overlap)	0.25	0.5	0.5	No
10	Group 3 (All datasets)	0.0039	1.0	0.1308	Yes (VDENCLUE)
20	Group 1 (Excluding datasets with significant overlap)	0.01562	0.4375	0.1093	Yes (VDENCLUE)
20	Group 2 (Only datasets with significant overlap)	0.25	0.75	1.0	No
20	Group 3 (All datasets)	0.0019	0.8203	0.1601	Yes (VDENCLUE)

References

1. Nayyar, A.; Puri, V. Comprehensive analysis & performance comparison of clustering algorithms for big data. *Rev. Comput. Eng. Res.* **2017**, *4*, 54–80. [CrossRef]
2. Premkumar, M.; Sinha, G.; Ramasamy, M.D.; Sahu, S.; Subramanyam, C.B.; Sowmya, R.; Abualigah, L.; Derebew, B. Augmented weighted K-means grey wolf optimizer: An enhanced metaheuristic algorithm for data clustering problems. *Sci. Rep.* **2024**, *14*, 5434. [CrossRef]
3. Mahdi, M.A.; Hosny, K.M.; Elhenawy, I. Scalable Clustering Algorithms for Big Data: A Review. *IEEE Access* **2021**, *9*, 80015–80027. [CrossRef]
4. Murtagh, F.; Contreras, P. Algorithms for hierarchical clustering: An overview. *Wiley Interdiscip Rev. Data Min. Knowl. Discov.* **2012**, *2*, 86–97. [CrossRef]
5. Kriegel, H.P.; Kröger, P.; Kröger, K.; Sander, O.; Zimek, A. Density-based clustering. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2011**, *1*, 231–240. [CrossRef]
6. Cheng, W.; Wang, W.; Batista, S. Grid-Based Clustering. In *Data Clust*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2018; pp. 128–148. [CrossRef]
7. Mcnicholas, P.D. Model-Based Clustering. *J. Classif.* **2016**, *33*, 331–373. [CrossRef]
8. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd* **1996**, *96*, 226–231.
9. Gholizadeh, N.; Saadatfar, H.; Hanafi, N. K-DBSCAN: An improved DBSCAN algorithm for big data. *J. Supercomput.* **2021**, *77*, 6214–6235. [CrossRef]

10. Wang, Y.; Gu, Y.; Shun, J. Theoretically-efficient and practical parallel DBSCAN. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of 2020, Portland, OR, USA, 14–19 June 2020; p. 17. [CrossRef]
11. Jang, J.; Lee, Y.; Lee, S.; Shin, D.; Kim, D.; Rim, H. A novel density-based clustering method using word embedding features for dialogue intention recognition. *Clust. Comput.* **2016**, *19*, 2315–2326. [CrossRef]
12. Idrissi, A.; Rehioui, H.; Laghrissi, A.; Retal, S. An improvement of DENCLUE algorithm for the data clustering. In Proceedings of the 2015 5th International Conference on Information & Communication, Paris, France, 12–13 October 2015.
13. Liu, P.; Zhou, D.; Wu, N. VDBSCAN: Varied density based spatial clustering of applications with noise. In Proceedings of the 2007 International Conference on Service Systems and Service, Vienna, Austria, 17–20 September 2007.
14. Fahim, A.M.; Saake, G.; Salem AB, M.; Torkey, F.A.; Ramadan, M.A. An enhanced density based spatial clustering of applications with noise. In Proceedings of the 2009 IEEE International Advance Computing Conference, Patiala, India, 6–7 March 2009.
15. Bryant, A.; Cios, K. Rnn-dbscan: A density-based clustering algorithm using reverse nearest neighbor density estimates. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 1109–1121. [CrossRef]
16. Saeed, M.M.; Al Aghbari, Z.; Alsharidah, M. Big data clustering techniques based on Spark: A literature review. *PeerJ Comput. Sci.* **2020**, *6*, e321. [CrossRef] [PubMed]
17. Chen, Y.; Zhou, L.; Pei, S.; Yu, Z.; Chen, Y.; Liu, X.; Du, J.; Xiong, N. KNN-BLOCK DBSCAN: Fast Clustering for Large-Scale Data. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 3939–3953. [CrossRef]
18. Patwary, M.A.; Satish, N.; Sundaram, N.; Manne, F.; Habib, S.; Dubey, P. Pardicle: Parallel Approximate Density-Based Clustering. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC, New Orleans, LA, USA, 16–21 November 2014; Volume 2015, pp. 560–571. [CrossRef]
19. Bessrour, M.; Elouedi, Z.; Lefevre, E. E-DBSCAN: An evidential version of the DBSCAN method. In Proceedings of the 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, 1–4 December 2020; pp. 3073–3080. [CrossRef]
20. Ienco, D.; Bordogna, G. Fuzzy extensions of the DBScan clustering algorithm. *Soft Comput.* **2018**, *22*, 1719–1730. [CrossRef]
21. Yu, X.G.; Jian, Y. A new clustering algorithm based on KNN and denclue. In Proceedings of the 2005 International Conference on Machine Learning and Cybernetics, ICMLC 2005, Guangzhou, China, 18–21 August 2005; pp. 2033–2038. [CrossRef]
22. Ankerst, M.; Breunig, M.M.; Kriegel, H.P.; Sander, J. OPTICS: Ordering Points to Identify the Clustering Structure. *SIGMOD Rec. (ACM Spec. Interest Group Manag. Data)* **1999**, *28*, 49–60. [CrossRef]
23. Khader, M.; Al-Naymat, G. Discovery of arbitrary-shapes clusters using DENCLUE algorithm. *Int. Arab J. Inf. Technol.* **2020**, *17*, 629–634. [CrossRef]
24. Kazemi, U.; Boostani, R. FEM-DBSCAN: An Efficient Density-Based Clustering Approach. *Iran. J. Sci. Technol. Trans. Electr. Eng.* **2021**, *45*, 979–992. [CrossRef]
25. Khader, M.; Al-Naymat, G. An overview of various enhancements of DENCLUE algorithm. In Proceedings of the Second International Conference on Data Science, E, New York, NY, USA, 2 December 2019. [CrossRef]
26. Khader, M.S.; Al-Naymat, G. VDENCLUE: An Enhanced Variant of DENCLUE Algorithm. *Adv. Intell. Syst. Comput.* **2021**, *1251*, 425–436. [CrossRef]
27. Yu, D.; Liu, G.; Guo, M.; Liu, X.; Yao, S. Density Peaks Clustering Based on Weighted Local Density Sequence and Nearest Neighbor Assignment. *IEEE Access* **2019**, *7*, 34301–34317. [CrossRef]
28. Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications. *Data Min. Knowl. Discov.* **1998**, *2*, 169–194. [CrossRef]
29. Xibilia, M.G.; Latino, M.; Marinkovic, Z.; Atanaskovic, A.; Donato, N. Soft Sensors Based on Deep Neural Networks for Applications in Security and Safety. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 7869–7876. [CrossRef]
30. Burnham, P.; Gomez-Lopez, N.; Heyang, M.; Cheng, A.P.; Lenz, J.S.; Dadhania, D.M.; Lee, J.R.; Suthanthiran, M.; Romero, R.; De Vlaminck, I. Separating the signal from the noise in metagenomic cell-free DNA sequencing. *Microbiome* **2020**, *8*, 1–9. [CrossRef] [PubMed]
31. Askari, S. Fuzzy C-Means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development. *Expert Syst. Appl.* **2021**, *165*, 113856. [CrossRef]
32. Du, M.; Ding, S.; Xue, Y. A robust density peaks clustering algorithm using fuzzy neighborhood. *Int. J. Mach. Learn. Cybern.* **2018**, *9*, 1131–1140. [CrossRef]
33. Iam-On, N. Clustering data with the presence of attribute noise: A study of noise completely at random and ensemble of multiple k-means clusterings. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 491–509. [CrossRef]
34. Ajmal, O.; Mumtaz, S.; Arshad, H.; Soomro, A.; Hussain, T.; Attar, R.W.; Alhomoud, A. Enhanced Parameter Estimation of DENsity CLUstEring (DENCLUE) Using Differential Evolution. *Mathematics* **2024**, *12*, 2790. [CrossRef]
35. Zhang, T.T.; Yuan, B. Density-Based Multiscale Analysis for Clustering in Strong Noise Settings with Varying Densities. *IEEE Access* **2018**, *6*, 25861–25873. [CrossRef]
36. Li, Z.; Liu, J.; Chen, S.; Tang, X. Noise robust spectral clustering. In Proceedings of the 2007 IEEE 11th International Conference on Computer Vision, Rio De Janeiro, Brazil, 14–21 October 2007.
37. Tatjer, A.; Nagarajan, B.; Marques, R.; Radeva, P. CCLM: Class-Conditional Label Noise Modelling. *Lect. Notes Comput. Sci.* **2023**, *14062*, 3–14. [CrossRef]
38. Rączkowska, A.; Osowska-Kurczab, A.; Szczerbiński, J.; Jasinska-Kobus, K.; Nazarko, K. AlleNoise—large-scale text classification benchmark dataset with real-world label noise. *arXiv* **2024**, arXiv:2407.10992.

39. Moulavi, D.; Jaskowiak, P.A.; Campello, R.J.G.B.; Zimek, A.; Sander, J. Density-based clustering validation. In Proceedings of the SIAM International Conference on Data Mining 2014, SDM 2014, Philadelphia, PA, USA, 24–26 April 2014; Volume 2, pp. 839–847. [CrossRef]
40. Clerc, M. Particle Swarm Optimization. In *Part. Swarm Optim*; Wiley Online Library: Minneapolis, MN, USA, 2010. [CrossRef]
41. Bakirtzis, A.; Kazarlis, S. Genetic Algorithms. In *Advanced Solutions in Power Systems: HVDC, FACTS, and Artificial Intelligence*; Wiley Online Library: Minneapolis, MN, USA, 2016; pp. 845–902. [CrossRef]
42. Kachitvichyanukul, V. Comparison of three evolutionary algorithms: GA, PSO, and DE. *Ind. Eng. Manag. Syst.* **2012**, *11*, 215–223. [CrossRef]
43. Srinivas, C.; Reddy, B.R.; Ramji, K.; Naveen, R. Sensitivity analysis to determine the parameters of genetic algorithm for machine layout. *Procedia Mater. Sci.* **2014**, *6*, 866–876. [CrossRef]
44. Isiet, M.; Gadala, M. Sensitivity Analysis of Control Parameters in Particle Swarm Optimization. *J. Comput. Sci.* **2020**, *41*, 101086. [CrossRef]
45. Karami, A.; Johansson, R. Choosing DBSCAN parameters automatically using differential evolution. *Int. J. Comput. Appl.* **2014**, *91*, 1–11. [CrossRef]
46. Santosh, T.; Ramesh, D. DENCLUE-DE: Differential Evolution Based DENCLUE for Scalable Clustering in Big Data Analysis. *Lect. Notes Data Eng. Commun. Technol.* **2020**, *44*, 436–445. [CrossRef]
47. Juwono, F.H.; Wong, W.K.; Pek, H.T.; Sivakumar, S.; Acula, D.D. Ovarian cancer detection using optimized machine learning models with adaptive differential evolution. *Biomed. Signal Process. Control.* **2022**, *77*, 103785. [CrossRef]
48. Golilarz, N.A.; Mirmozaffari, M.; Gashteroodkhani, T.A.; Ali, L.; Dolatsara, H.A.; Boskabadi, A.; Yazdi, M. Optimized wavelet-based satellite image de-noising with multi-population differential evolution-assisted harris hawks optimization algorithm. *IEEE Access* **2020**, *8*, 133076–133085. [CrossRef]
49. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
50. Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. The WEKA data mining software. *ACM SIGKDD Explor. Newsl.* **2009**, *11*, 10–18. [CrossRef]
51. Hassan, B.A.; Tayfor, N.B.; Hassan, A.A.; Ahmed, A.M.; Rashid, T.A.; Abdalla, N.N. From A-to-Z review of clustering validation indices. *Neurocomputing* **2024**, *601*, 128198. [CrossRef]
52. Hubert, L.; Arabie, P. Comparing partitions. *J. Classif.* **1985**, *2*, 193–218. [CrossRef]
53. Vinh, N.X.; Epps, J.; Bailey, J. Information theoretic measures for clusterings comparison: Is a correction for chance necessary? In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009. [CrossRef]
54. Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [CrossRef]
55. Davies, D.L.; Bouldin, D.W. A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.* **1979**, *2*, 224–227. [CrossRef]
56. Amrulloh, K. Comparison Between Davies-Bouldin Index and Silhouette Coefficient Evaluation Methods in Retail Store Sales Transaction Data Clusterization Using K-Medoids Algorithm. In Proceedings of the 3rd South American International Industrial Engineering and Operations Management Conference, Asuncion, Paraguay, 19–21 July 2022; pp. 1952–1961.
57. Panwong, P.; Boongoen, T.; Iam-On, N. Improving consensus clustering with noise-induced ensemble generation. *Expert Syst. Appl.* **2020**, *146*, 113138. [CrossRef]
58. Kelly, M.; Longjohn, R.; Nottingham, K. The UCI Machine Learning Repository. Available online: https://archive.ics.uci.edu/ (accessed on 29 August 2024).
59. Fränti, P.; Sieranoja, S. K-means properties on six clustering benchmark datasets. *Appl. Intell.* **2018**, *48*, 4743–4759. [CrossRef]
60. Zhou, Y.; Xiang, Y.; He, X. Constrained multiobjective optimization: Test problem construction and performance evaluations. *IEEE Trans. Evol. Comput.* **2020**, *25*, 172–186. [CrossRef]
61. Hou, Y.; Wu, Y.; Liu, Z.; Han, H.; Wang, P. Dynamic multi-objective differential evolution algorithm based on the information of evolution progress. *Sci. China Technol. Sci.* **2021**, *64*, 1676–1689. [CrossRef]
62. Ahmad, M.F.; Isa, N.A.M.; Lim, W.H.; Ang, K.M. Differential evolution: A recent review based on state-of-the-art works. *Alex. Eng. J.* **2022**, *61*, 3831–3872. [CrossRef]
63. Ronkkonen, J.; Kukkonen, S.; Price, K.V. Real-parameter optimization with differential evolution. In Proceedings of the 2005 IEEE Congress on Evolutionary Computation, Edinburgh, Scotland, UK, 2–5 September 2005; pp. 506–513.
64. Pant, M.; Zaheer, H.; Garcia-Hernandez, L.; Abraham, A. Differential Evolution: A review of more than two decades of research. *Eng. Appl. Artif. Intell.* **2020**, *90*, 103479.
65. Xie, Y.; Jia, X.; Shekhar, S.; Bao, H.; Zhou, X. Significant DBSCAN+: Statistically Robust Density-based Clustering. *ACM Trans. Intell. Syst. Technol.* **2021**, *12*, 1–26. [CrossRef]
66. Marques, J.C.; Orger, M.B. Clusterdv: A simple density-based clustering method that is robust, general and automatic. *Bioinformatics* **2019**, *35*, 2125–2132. [CrossRef]

67. Biju, V.G.; Prashanth, C.M. Friedman and Wilcoxon evaluations comparing SVM, bagging, boosting, K-NN and decision tree classifiers. *J. Appl. Comput. Sci. Methods* **2017**, *9*, 23–47. [CrossRef]
68. Zimmerman, D.W.; Zumbo, B.D. Relative power of the wilcoxon test, the friedman test, and repeated-measures ANOVA on ranks. *J. Exp. Educ.* **1993**, *62*, 75–86. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.