COMPARATIVE ANALYSIS OF CLUSTERING ALGORITHMS ON SYNTHETIC CIRCULAR PATTERS DATA

Hardev Ranglani

EXL Service Inc, USA

ABSTRACT

Clustering algorithms play a pivotal role in discovering hidden patterns in unlabeled data, but their performance varies significantly across datasets with complex geometries. This paper explores the performance of various clustering techniques in identifying distinct circular clusters within the Synthetic Circle Data Set, a benchmark dataset designed to test algorithms on non-linear structures. We evaluate popular clustering methods, including k-means, DBSCAN, Gaussian Mixture Models, hierarchical clustering, and emerging techniques like Self Organizing Maps, Mean Shift Clustering and Spectral Clustering. Using metrics such as Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), and Silhouette Score, along with detailed visualizations, we systematically compare the algorithms' ability to recover the true circle-based clusters without prior labels. Our findings highlight the strengths and limitations of each method, revealing that density- and graph-based algorithms consistently outperform traditional techniques like k-means in handling circular patterns.

KEYWORDS

Clustering, K-Means algorithm, Non-linear patterns, Density-Based Clustering, Hierarchical Clustering, Gaussian Mixture Models, Adjusted Rand Index, Spectral Clustering

1. INTRODUCTION

Clustering, an essential task in unsupervised machine learning, is widely used to discover underlying patterns and structures in unlabeled data. Despite its prevalence, clustering algorithms often face significant challenges when applied to datasets with complex geometries, such as non-linear or concentric patterns. Traditional algorithms, like k-means, work well at clustering linearly separable data but frequently struggle to identify non-linear relationships. This limitation becomes a bigger issue in datasets with overlapping, circular, or nonconvex structures, where clustering boundaries are inherently non-Euclidean. Addressing these challenges requires evaluating advanced clustering algorithms that can adapt to such complexities. To analyze clustering performance on data with non-linear patterns, this study utilizes the Synthetic Circle Data Set from the UCI Machine Learning Repository—a benchmark dataset consisting of two-dimensional points arranged into multiple circular clusters (Synthetic Circle Data Set, 2024) The simplicity of this data set makes it ideal for clustering evaluations, as its two-dimensional structure facilitates easy visualization and interpretation of the results. Each observation is already associated with a ground-truth label identifying the circle it belongs to, enabling rigorous comparisons between predicted clusters and true clusters. The overall goal is to assess whether clustering algorithms can identify the individual circles without access to the true labels during the clustering process.

Clustering algorithms differ significantly in their ability to adapt to such data complexities. Density-based methods like DBSCAN (Ester et al., 1996) are known for their ability to handle irregular cluster shapes and noise, while spectral clustering approaches leverage graph-based representations to uncover nonlinear patterns (Ng et al., 2001). Gaussian Mixture Models Reynolds, 2009) and hierarchical clustering methods (Johnson, 1967; Murtagh & Contreras, 2012) offer flexibility in modeling and structuring clusters, but their performance can depend heavily on parameter tuning. Meanwhile, recent advances such as HDBSCAN (Campello et al., 2015) aim to extend traditional density-based approaches by dynamically determining the number of clusters and addressing varying densities. These methods, along with foundational algorithms like k-means, form a robust foundation for evaluating clustering performance on the Synthetic Circle Data Set.

The Synthetic Circle Data Set provides several advantages for this analysis. It has only 2 features- the X and Y co-ordinates of the data point and the target variable is the "class" which is basically a label for which circle the data point belongs to. So overall, the data has only 3 columns. This allows for easier visualizations that clearly illustrate the success or failure of different clustering methods. The circle label in the dataset facilitate quantitative evaluations using metrics like Adjusted Rand Index (Hubert & Arabie, 1985) and Silhouette Score (Rousseeuw, 1987), enabling objective comparisons of the quality of the clustering. By systematically analyzing and comparing clustering algorithms, this study seeks to identify methods that work well in recovering circular clusters while highlighting the limitations of others. Circular or arbitrarily shaped clusters are commonly encountered in fields such as biology, social networks, and geospatial analysis, and understanding which algorithms are best suited to these structures can enhance the effectiveness of clustering algorithms.

1.1. Overall Research Goal and Novelty of this Work

The overall objective of this study is to evaluate and compare the performance of various clustering algorithms on the Synthetic Circle Data Set, which is a dataset with non-linear, circular cluster structures. The goal here is to determine how well algorithms can recover true clusters (circles) without access to ground-truth labels, using metrics like ARI, NMI, and Silhouette Score. This study addresses a gap in clustering algorithms research by focusing on datasets with circular geometries, unlike traditional convex datasets like the Iris dataset. The Synthetic Circle Data Set from the UCI ML repository serves as a novel benchmark, enabling precise evaluation of clustering methods on non-linear patterns. The results highlight the effectiveness of density-based and hierarchical methods for circular clusters and provide a practical framework for evaluating algorithms on non-linear geometries. This study offers actionable insights for real-world applications in biology, geospatial analysis, and social networks, addressing clustering challenges often overlooked in traditional benchmarks. Thus, the overall question answered by this analysis can be summarized as- How effectively can clustering algorithms identify true clusters in data with non-linear, circular geometries without access to ground-truth labels?

This paper contributes to the literature by providing a performance evaluation of clustering algorithms on data with non-linear structures, emphasizing their suitability for separating circular patterns. The insights gained from this analysis can guide practitioners in selecting the most effective methods for clustering on complex datasets, such as those encountered in biological, geographical, and social network analyses. Furthermore, the findings highlight the importance of aligning algorithm choice with the inherent geometry of the data, a consideration often overlooked in clustering applications. The rest of the paper is structured as follows: The Literature review section discusses the previous related work on the subject, the methodology section describes the dataset briefly describes each clustering algorithm, along with the evaluation metrics used to assess the performance of clustering algorithms, the results section

describes in detail the performance of each of the algorithms, and the Conclusion and Future work section mentions how the findings mentioned in this paper can be used for future analysis.

2. LITERATURE REVIEW

Clustering, as a fundamental unsupervised learning task, has been extensively studied, with significant progress seen in developing algorithms tailored to various data structures and application domains. However, challenges still exist in effectively clustering data with complex geometries, such as circular or concentric patterns. This section reviews key advancements in clustering algorithms, their application to non-linear and geometrically complex datasets, and how the Synthetic Circle Data Set provides a unique benchmark for comparing these methods.

2.1. K-Means and its Limitations

K-means clustering (MacQueen, 1967) remains one of the most popular clustering algorithms due to its simplicity and computational efficiency. However, it relies on Euclidean distance that makes it often unsuitable for non-convex or non-linear cluster shapes (Kanungo et al., 2002). Various extensions, such as kernel k-means (Schölkopf et al., 1998), try to overcome this limitation by mapping the data into a higher-dimensional feature space so that clusters may become linearly separable. Despite these advances, k-means is sensitive to initialization and the need to specify the number of clusters remain significant challenges.

2.2. Density-Based Clustering

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) (Ester et al., 1996) performs clustering by identifying clusters as dense regions of data points. Its ability to handle noise and detect arbitrarily shaped clusters makes it particularly effective for non-linear patterns. However, DBSCAN's performance is highly sensitive to parameters like eps(the maximum distance between two samples for one to be considered as in the neighborhood of the other) and minsamples (the number of points, including the core point itself that must exist within an eps neighborhood for a point to be considered a core point) , and it struggles with datasets featuring varying densities. HDBSCAN (Campello et al., 2015) tries to addresses these limitations by dynamically adjusting density thresholds, making it well-suited for datasets with varying cluster densities.

2.3. Hierarchical Clustering

Hierarchical clustering techniques, including agglomerative (Johnson, 1967) and divisive approaches (Murtagh & Contreras, 2012), build tree-like structures (dendrograms) to represent data groupings at multiple levels of granularity. While these methods provide flexibility in cluster formation, they often rely on distance metrics that do not work well with non-linear patterns. Advances such as dynamic dendrogram cutting (Langfelder et al., 2008) aim to improve their utility for complex data.

2.4. Gaussian Mixture Models

Gaussian Mixture Models (GMMs) (Reynolds, 2009) offer a probabilistic approach to clustering, modeling data as a mixture of Gaussian distributions. GMMs excel at handling overlapping clusters and capturing soft memberships, but they assume that clusters follow Gaussian shapes, which may not hold for non-linear geometries like circles. Extensions, such as variational

Bayesian GMMs (Bishop, 2006), attempt to address these limitations by introducing more flexible priors.

2.5. Spectral Clustering

Spectral clustering (Ng et al., 2001) uses graph-based representations of data, using eigenvectors of the graph Laplacian to partition data into clusters. Its ability to handle non-linear and non-convex patterns makes it a strong candidate for datasets like the Synthetic Circle Data Set. Despite its strengths, spectral clustering requires careful selection of similarity measures and parameters.

2.6. Self-Organizing Maps

Self-Organizing Maps (SOMs), introduced by Kohonen (1982), are unsupervised neural networks that project high-dimensional data onto a lower-dimensional grid while preserving topological relationships. SOMs have been widely used in clustering and visualization tasks across fields such as biology and healthcare (Vesanto & Alhoniemi, 2000).However, SOMs struggle with capturing highly non-linear or complex patterns due to their fixed grid topology, which may oversimplify relationships in intricate datasets.

2.7. MeanShift Clustering

MeanShift, a density-based clustering algorithm, identifies clusters by shifting data points toward regions of higher density (Fukunaga & Hostetler, 1975). Unlike Kmeans, it does not require the number of clusters to be predefined. However, despite its flexibility, MeanShift may perform poorly with non-linear or overlapping patterns, as it relies on the kernel bandwidth, which can fail to adapt dynamically to complex density distributions.

2.8. Evaluation of Clustering Algorithms

Several benchmark datasets, such as the Iris dataset (Fisher, 1936) and synthetic datasets (Blobs, Moons), have been used to evaluate clustering algorithms. However, these datasets often do not represent the geometric complexity of real-world data. The Synthetic Circle Data Set, by contrast, introduces a controlled environment where the true cluster shapes are circular, making it ideal for evaluating the ability of clustering algorithms to handle non-linear geometries.

Metrics such as Adjusted Rand Index (ARI) (Hubert & Arabie, 1985), Normalized Mutual Information (NMI) (Vinh et al., 2010), and Silhouette Score (Rousseeuw, 1987) are widely used to quantify clustering performance. These metrics allow researchers to compare algorithms objectively, even across datasets with varying complexities. Existing benchmarks often fail to capture the intricacies of such patterns, leaving a gap in the evaluation of clustering methods tailored for non-linear data. This study addresses this gap by:

1. Using the Synthetic Circle Data Set as a benchmark to evaluate clustering algorithms on non-linear geometries.
2. Systematically comparing algorithms across multiple dimensions, including computational efficiency, clustering accuracy (ARI, NMI), and cluster separability (Silhouette Score), along with detailed visaulizations

3. Providing actionable insights into the strengths and limitations of each method, helping practitioners choose appropriate algorithms for real-world tasks involving complex data structures.

2.9. Differences from current State of the Art

The differences for the current analysis as compared to the existing literature can be summarized as:

1. This analysis uses a non-linear dataset with predefined circular clusters, which are found to be rarely addressed in clustering evaluations.
2. This analysis systematically compares a diverse range of algorithms (densitybased, hierarchical, probabilistic, graph-based, and neural-inspired) in a single framework
3. It also emphasizes actionable insights for practitioners dealing with similar non-linear structures.

3. METHODOLOGY

This section outlines the methodology adopted for evaluating clustering algorithms on the Synthetic Circle Data Set. It includes details on the data set, the application of clustering algorithms, the evaluation metrics used, and the overall experimental setup.

3.1. The Synthetic Circle Dataset

This dataset comprises 10000 two-dimensional points arranged into 100 circles, each containing 100 points, and it is available on the UCI Machine Learning Repository. It was designed to valuate clustering algorithms, by providing a clear and structured clustering challenge. Figure 1 shows a sample of 5 records of the data, which contain only 3 features- the x-coordinate and the y-coordinate of the data point and the ’class’ label indicating which circle the data point belongs to, which can be between 0 to 99. Figure 2 shows a scatter plot of the data in 2 dimensions, clearly indicating the label of which data point belongs to which circle. These labels are used solely for evaluation purposes and are not provided as input to the clustering algorithms. The challenge for the algorithms is to identify each of these 100 circles as 100 separate clusters based purely on the x and y coordinates of the points.

	x	y	class
0	3.15676	116.12252	6
1	16.14436	16.81660	11
2	100.31212	64.99025	53
3	-1.33773	84.81772	4
4	104.37328	62.42373	53

Figure 1: Sample of 5 records of the Synthetic Circle Dataset with 3 features

3.2. Clustering Algorithms

A variety of clustering algorithms are applied to the dataset, chosen for their different approaches to handling non-linear and geometrically complex data:

3.2.1. K-Means

A centroid-based algorithm that partitions data into k clusters using Euclidean distance. It partitions data into k clusters by iteratively minimizing the within-cluster sum of squares. The algorithm lternates between assigning each data point to the nearest centroid and updating the centroids based on the mean of the assigned points. The optimization goal is to minimize:

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||^2,$$

where Ci is the ith cluster and μi is its centroid.

3.2.2. DBSCAN

A density-based algorithm that clusters points by identifying dense regions based on two parameters: ϵ(the radius of the neighborhood) and minPts (minimum number of points required for a dense region). Points are classified as core, border, or noise. The algorithmgrows clusters from core points by including points within ϵ that are directly or indirectly density- reachable. Mathematically,

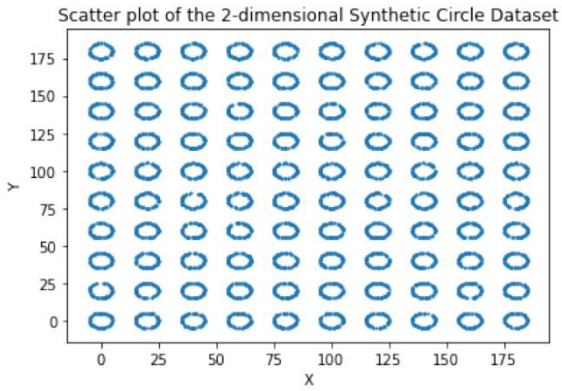

Figure 2: Scatter plot of the Synthetic Circle Dataset, representing 100 circles

for a point p, the neighborhood is defined as

$$N(p) = \{q : dist(p, q) \leq \epsilon\}$$

and p becomes a core point if $|N(p)| \geq$ minPts.

3.2.3. Hierarchical Clustering

It organizes data into a dendrogram that represents nested groupings based on their similarity. It can be agglomerative, starting with each data point as its luster, or divisive, starting with one large cluster. Clusters are merged or divided based on linkage criteria such as single-linkage (minimum distance between clusters), complete-linkage (maximum distance), or average-linkage

(mean distance). UsingWard's method, the distance between clusters u and v after merging with cluster s is updated as:

$$d(u,v) = \sqrt{\frac{|v|+|s|}{T}d(v,s)^2 + \frac{|u|+|s|}{T}d(u,s)^2 - \frac{|s|}{T}d(u,v)^2}$$

Where

$$T = |u| + |v| + |s|$$

3.2.4. Gaussian Mixture Model (GMM)

It is a clustering algorithm based on the assumption that data is generated from a mixture of several Gaussian distributions with unknown parameters. The Expectation-Maximization (EM) algorithm estimates the parameters iteratively. The probability density function of the data is:

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$$

where πk is the weight, μk is the mean, and Σk is the covariance matrix. GMM assigns points to clusters probabilistically, making it more flexible than hard clustering methods like k-means.

3.2.5. Self-Organizing Maps (SOM)

are a type of neural network used for clustering and dimensionality reduction. They project high-dimensional data onto a low dimensional (usually 2D) grid, preserving topological relationships. During training, data points adjust the weights of the winning neuron and its neighbors using:

$$w_i(t+1) = w_i(t) + \eta(t)h_{ci}(t)[x(t) - w_i(t)]$$

where hci(t) is the neighborhood function, and $\eta(t)$ is the learning rate.

3.2.6. Spectral Clustering algorithm

uses the eigen values of a graph Laplacian matrix derived from the data to form clusters. It embeds the data into a lower dimensional space, capturing the structure of the data graph, and applies a standard clustering algorithm like k-means. The normalized graph Laplacian is computed as:

$$L_{sym} = D^{-1/2}LD^{-1/2}$$

, where D is the degree matrix, and L = D −W is the unnormalized Laplacian with W as the adjacency matrix. This method is effective for capturing non-linear cluster structures.

3.2.7. Mean Shift Clustering algorithm

It identifies clusters by locating areas of high density in the feature space. Starting with random initial points, it iteratively shifts them toward the mean of their neighborhood defined by a kernel function, such as Gaussian. The update step for each point x_i is given by:

$$x_i^{t+1} = \frac{\sum_j K(x_i^t - x_j)x_j}{\sum_j K(x_i^t - x_j)}$$

Where K is the kernel function.

Each algorithm is configured with parameters optimized for the data set, ensuring a fair comparison. The primary objective of this study is to evaluate how effectively clustering algorithms can recover the true circular clusters. Specifically, the algorithms are evaluated on the basis of grouping observations into clusters that correspond to the underlying circles in the dataset, and achieving this clustering without access to the ground-truth labels (*circle_id*), which are used only for evaluation.

3.3. Evaluation Metrics

To objectively compare the performance of the clustering algorithms, the following metrics are used:

1. Adjusted Rand Index (ARI): Measures the similarity between the predicted clusters and the true labels, adjusted for chance. Values range from -1 (poor agreement) to 1 (perfect agreement).
2. Normalized Mutual Information (NMI): Captures the shared information between the predicted and true clusters. NMI values range from 0 (no shared information) to 1 (perfect match).
3. Silhouette Score: Evaluates cohesion within clusters and separation between clusters. Values range from -1 (poorly defined clusters) to 1(well-separated clusters).
4. Visual Assessment: Scatter plots of the clustered data, with each data point colored according to its cluster, along with centroids of the cluster, are compared to see if the circles are correctly identified by each cluster. Additionally, a Voronoi diagram is also overlaid to visualize the partitioning of the feature space, illustrating the boundaries between clusters. This visualization allows for a direct comparison of the algorithm's clustering results with the expected structure of the data, particularly highlighting its ability (or inability) to separate the circles.

4. RESULTS

The results demonstrate that density-based methods (DBSCAN, MeanShift) and Hierarchical Clustering are highly effective at identifying non-linear, circular clusters, outperforming traditional methods like k-means and Gaussian Mixture Models. This highlights the importance of choosing algorithms tailored to the data's geometric complexity. The research also underscores the limitations of Self-Organizing Maps and Spectral Clustering for such tasks, offering valuable insights into their applicability. The detailed results for each of the clustering algorithms are highlighted in the next subsections.

4.1. K-Means Algorithm

As seen in the figure 3, the k-means algorithm does a decent job of separating each circle into its own cluster, but some of the circles are not clearly separated into distinct clusters. The Adjusted Rand Index (ARI) for the k-Means algorithm is 0.9688, the Normalized Mutual Information (NMI) is 0.99166 and Silhouette Score is 0.59042, highlighting the performance of the k-Means algorithm.

4.2. DBSCAN Algorithm

As seen in the figure 4, the DBSCAN algorithm does a much better job of separating each circle into its own cluster, as all of the circles are clearly separated into distinct clusters. The Adjusted Rand Index (ARI) for the k-Means algorithm is 1.0, the Normalized Mutual Information (NMI) is 1.0 and Silhouette Score is 0.6085, highlighting the performance of the DBSCAN algorithm.

4.3.Agglomerative Clustering Algorithm

As seen in the figure 5, the Agglomerative Clustering algorithm also does a good job of separating each circle into its own cluster, as all of the circles are clearly separated into distinct clusters. The Adjusted Rand Index (ARI) for the k-Means algorithm is 1.0, the Normalized Mutual Information (NMI) is 1.0 and Silhouette Score is 0.6085, highlighting the performance of the DBSCAN algorithm.

4.4. Gaussian Mixture Models algorithm

As seen in the figure 6, the Gaussian Mixture models algorithm is able to separate most of the circles into its own cluster, while some of the circles are overlapping.

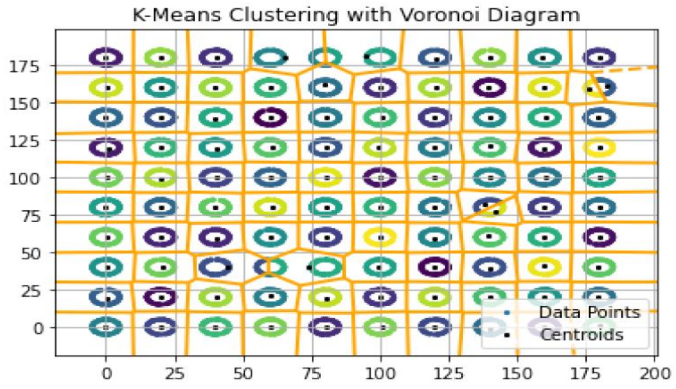

Figure 3: Scatter plot of the kmeans algorithm. Most of the circles are separated into a different cluster, while come circles have overlapping clusters

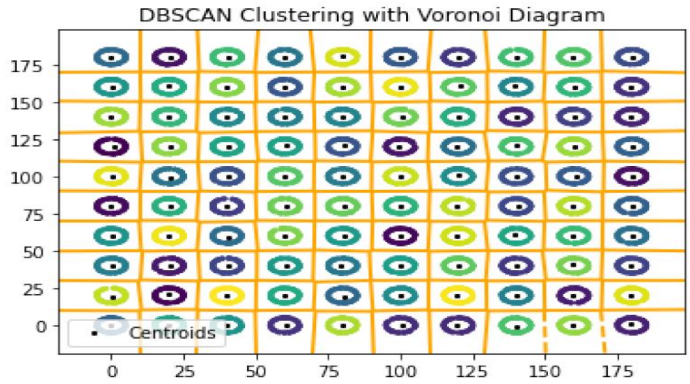

Figure 4: Scatter plot of the DBSCAN algorithm. All the circles are separated into a different cluster

The Adjusted Rand Index (ARI) for the GMM algorithm is 0.94630, the Normalized Mutual Information (NMI) is 0.98971 and Silhouette Score is 0.5688, highlighting the performance of the GMM algorithm.

4.5. Spectral Clustering Algorithm

As seen in the figure 4, the Spectral Clustering algorithm is unable to separate most of the circles into its own cluster. The Adjusted Rand Index (ARI) for the GMM algorithm is 0.2337, the Normalized Mutual Information (NMI) is 0.8195 and Silhouette Score is -0.14351, highlighting the performance of the Spectral Clustering algorithm.

4.6. Self-Organizing Maps algorithm

As seen in the figure 4, the Self-Organizing Maps algorithmis unable to separate most of the circles into its own cluster. The Adjusted Rand Index (ARI) for the SOM algorithm is 0.597084, the Normalized Mutual Information (NMI) is 0.8788 and Silhouette Score is 0.3216, highlighting the performance of the Spectral Clustering algorithm

4.7. Mean Shift Clustering algorithm

As seen in the figure 4, the Means Shift algorithm is able to perfectly separate all of the circles into its own cluster. The Adjusted Rand Index (ARI) for the SOM algorithm is 1.0, the Normalized Mutual Information (NMI) is 1.0 and Silhouette Score is 0.6085, highlighting the performance of the Spectral Clustering algorithm.

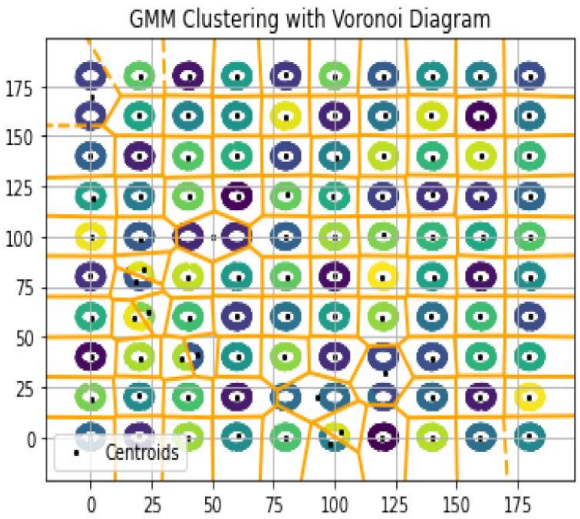

Figure 6: Scatter plot of the GMM algorithm. Most of the circles are separated into a different cluster

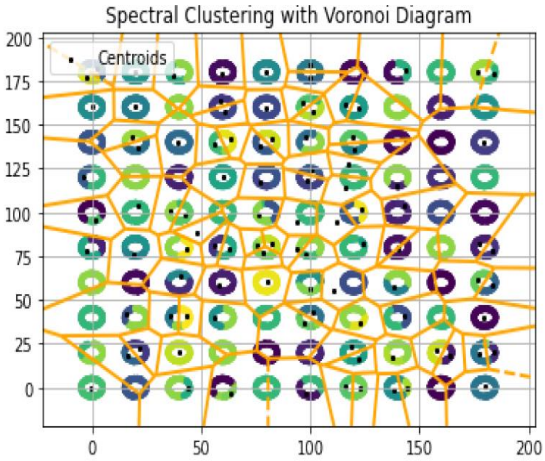

Figure 7: Scatter plot of the Spectral clustering algorithm. Most of the circles are not separated into a different cluster

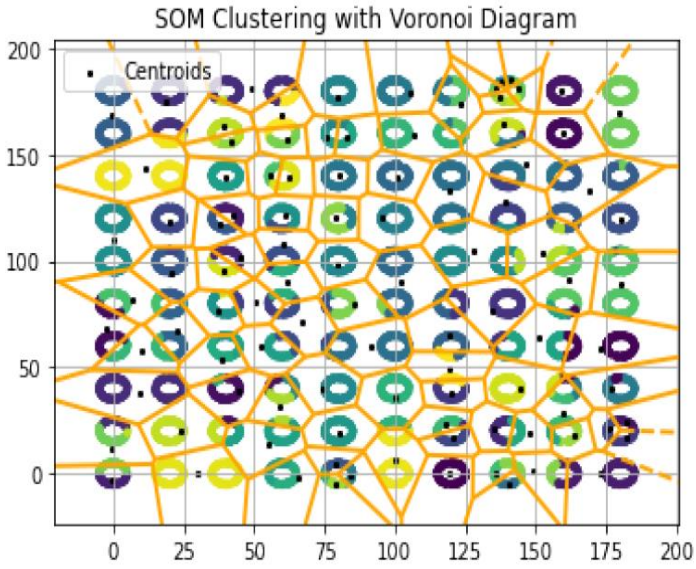

Figure 8: Scatter plot of the SOM algorithm. Most of the circles are not separated into a different cluster

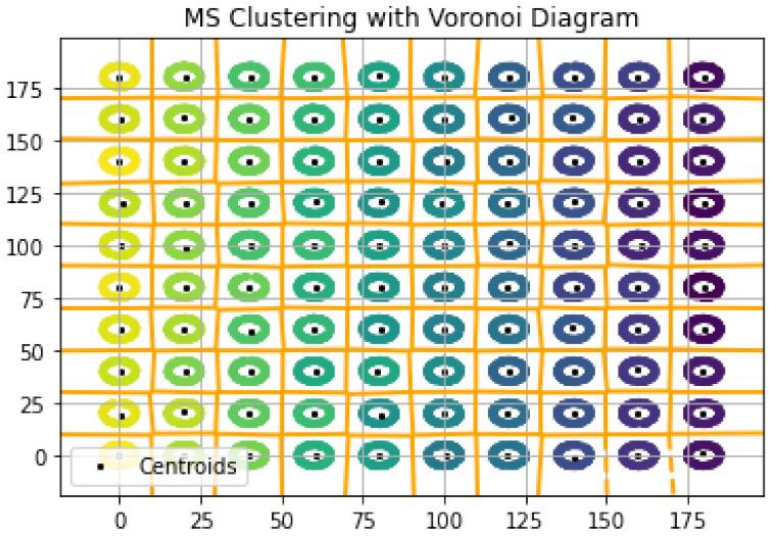

Figure 9: Scatter plot of the SOM algorithm. Most of the circles are not separated into a different cluster

4.8. Overall Analysis

Below is the table comparing the results of all algorithms –

Table 1: Clustering Algorithm Performance Metrics

Clustering Algorithm	Adjusted Rand Index (ARI)	Normalized Mutual Information (NMI)	Silhouette Score
DBSCAN	1.00	1.00	0.61
Hierarchical	1.00	1.00	0.61
MeanShift	1.00	1.00	0.61
kMeans	0.97	0.99	0.59
Gaussian Mixture Models	0.95	0.99	0.57
Self Organizing Maps	0.60	0.88	0.32
Spectral	0.23	0.82	-0.14

The results in table 1 show that DBSCAN, Hierarchical Clustering, and MeanShift are the most effective algorithms for this dataset, primarily due to their ability to handle non-linear and circular patterns robustly. In contrast, algorithms like k-Means, GMM, SOMs, and Spectral Clustering are less suited to the dataset's non-linear structure, requiring careful parameter tuning or fundamental modifications to achieve comparable performance.

DBSCAN performs well likely because it excels at detecting arbitrarily shaped clusters, such as circles, and is robust to noise. Hierarchical Clustering (likely agglomerative) performs equally well because its bottom-up approach effectively captures the nested and non-linear structure of the data. Mean Shift also demonstrates strong performance due to its density-based nature, which aligns well with the clustered circular geometry of the dataset.

k-Means and Gaussian Mixture Models (GMM) perform slightly worse, with ARI values of 0.97 and 0.95, respectively. While they capture most of the clusters correctly, they are not able to perfectly separating overlapping or noisy clusters as they rely on Euclidean distance and Gaussian assumptions.

Self-Organizing Maps (SOMs) and Spectral Clustering perform poorly compared to the other methods. SOMs clearly fail to adapt to the exact circular structure, with an ARI of 0.60 and a

relatively low Silhouette Score of 0.32. This shows that while some clusters are correctly identified, others overlap or are misclassified. Spectral Clustering exhibits the weakest performance (ARI: 0.23, MI: 0.82, Silhouette: -0.14), likely because of challenges in configuring the graph similarity matrix or eigenvalue based partitioning for this dataset.

4.9. Significance of the Results

The significance of this analysis can highlighted by the fact that the results demonstrate that density-based methods (DBSCAN, Mean Shift) and Hierarchical Clustering are highly effective at identifying non-linear, circular clusters, outperforming traditional methods like k-means and Gaussian Mixture Models. This highlights the importance of choosing algorithms tailored to the data's geometric complexity. The research also underscores the limitations of Self-Organizing Maps and Spectral Clustering for such tasks, offering valuable insights into their applicability.

This analysis thus demonstrates the importance of selecting the right clustering algorithm for datasets with circular or non-linear geometries. These results can have practical implications for domains like biology (e.g., detecting circular patterns in molecular structures), social networks (e.g., circular communities), and geospatial analysis (e.g., clustering geographic regions with circular features). This analysis shows that practitioners working with non-linear data should prioritize density- or hierarchical-based clustering approaches. The code used to perform the analysis and get all the resutls can be found on this github repository

5. CONCLUSION & FUTURE WORK

This study evaluated the performance of several clustering algorithms on the Synthetic Circle Data Set, focusing on their ability to identify circular clusters without prior knowledge of the true labels. future work could extend this analysis to higher dimensional or noisier datasets, where overlapping clusters and real-world complexities present additional challenges. Automated parameter tuning and enhancements to existing methods, such as custom distance metrics or graph representations, could further improve their adaptability. Applying these findings to real-world problems in biology, geospatial analysis, and social networks would validate their practical utility. This study provides a foundation for understanding and improving clustering performance on geometrically intricate datasets.

REFERENCES

[1] Synthetic Circle Data Set [Dataset]. (2024). UCI Machine Learning Repository. https://doi.org/10.24432/C51909.

[2] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 281–297.

[3] Kanungo, T., et al. (2002). An efficient k-means clustering algorithm: Analysis and implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), 881–892.

[4] Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation, 10(5), 1299–1319.

[5] Ester, M., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD), 226–231.

[6] Campello, R. J. G. B., et al. (2015). Density-based clustering based on hierarchical density estimates. ACM Transactions on Knowledge Discovery from Data, 10(1), 1–51.

[7] Johnson, S. C. (1967). Hierarchical clustering schemes. Psychometrika, 32(3), 241–254.

[8] Murtagh, F., & Contreras, P. (2012). Algorithms for hierarchical clustering: An overview. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2(1), 86–97.
[9] Reynolds, D. A. (2009). Gaussian Mixture Models. Encyclopedia of Biometrics, 659–663.
[10] Ng, A. Y., et al. (2001). On spectral clustering: Analysis and an algorithm. Advances in Neural Information Processing Systems (NIPS), 849–856.
[11] Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Maps. Biological Cybernetics, 43(1), 59–69.
[12] Xie, J., Girshick, R., & Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. Proceedings of the International Conference on Machine Learning (ICML), 478–487.
[13] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7(2), 179–188.
[14] Hubert, L., & Arabie, P. (1985). Comparing partitions. Journal of Classification, 2(1), 193–218.
[15] Fukunaga, K., Hostetler, L. (1975). The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. IEEE Transactions on Information Theory, 21(1), 32–40.
[16] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65.
[17] Vinh, N. X., Epps, J., & Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, and validity. Journal of Machine Learning Research, 11, 2837–2854.
[18] Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Maps. Biological Cybernetics, 43(1), 59–69.
[19] Vesanto, J., & Alhoniemi, E. (2000). Clustering of the Self-Organizing Map. IEEE Transactions on Neural Networks, 11(3), 586–600.