Dhaka Univ. J. Sci. 74(2): 271–282, 2026 (July)
ISSN 1022-2502, E-ISSN 2408-8528
DOI: https://doi.org/10.3329/dujs.v74i2.84327

Density-Peak-Based Clustering in Reduced Feature Spaces

Afsana Akter Setu, Juty Singha, and Sohana Jahan*

Department of Mathematics, University of Dhaka, Dhaka-1000, Bangladesh

(Received: 15 September 2025; Accepted: 17 January 2026)

Abstract

Clustering is a fundamental data mining technique that groups data points by similarity. A critical challenge for clustering algorithms is the effective selection of initial cluster centers, often done through inefficient trial-and-error. To address this, a novel Adaptive Cluster Center Initialization using Density Peak for Geodesic Distance-based Clustering (AGDPC) method intelligently identifies optimal centers. It builds upon the Density Peaks Clustering (DPC)and Geodesic-Based Initialization (GDPC) approaches, using weighted Euclideandistances that incorporate the Pearson correlation coefficient. Furthermore, AGDPC adaptively optimizes its threshold parameter using data field density estimation entropy, enhancing its accuracy and automation. It should be noted that, AGDPC currently operates directly on high-dimensional data without dimensionality reduction. While this preserves the original structure of the data, high-dimensional spaces often suffer from the "curse of dimensionality," where distances between points become less meaningful, and computational complexity increases. Introducing Multidimensional Scaling (MDS) as a pre-processing step can address these challenges by projecting data into a lower-dimensional space while preserving pairwise distances or dissimilarities as much as possible. In this work, we propose to use multidimensional scaling that mitigates the impact of irrelevant or redundant features, enabling 2D/3D visualisation of clusters for interpretability. We proposed to use Regularised Multidimensional Scaling using Radial Basis Function (RBF-MDS) for dimension reduction. The idea is to decrease its dimensionality, followed by the application of the adaptive strategy on the dataset in its reduced dimension. Our approaches are tested on various benchmarking datasets, and the outcomes are contrasted with those of DPC, GDPC, and one of the most conventional clustering methods, K-Means clustering. When tested on actual data, the experimental findings show that the proposed methods surpassed current leading approaches based on several clustering validation metrics and reduced computational time significantly.

Keywords: Multidimensional Scaling, Minimum Spanning Tree, Geodesic distance, Pearson Correlation Coefficient, Radial Basis Function

I. Introduction

Clustering is an unsupervised machine learning technique that groups similar data points together based on their inherent patterns or characteristics. It is widely used to discover hidden structures, segment datasets, and identify meaningful categories without prior labelling. Common applications include customer segmentation, anomaly detection, image compression, and biological data classification, making it a fundamental tool for data exploration and insight generation across various domains.

The origin of cluster analysis was related to statistics. In the early 1930s, two renowned statisticians, Robert Fisher and Andre Thiery, were the pioneers of introducing the concept of cluster analysis. Then it was introduced in anthropology in 1932 by Driver and Kroeber[1], later by Joseph Zubin[2] in 1938 and by Robert Tryon[3] in 1939 to psychology. In 1943, Cattell used trait theory classification in personality psychology[4].

Grygorash et al[17]. introduced HEMST, a hierarchical clustering algorithm that iteratively removes the longest edge from a Minimum Spanning Tree (MST)[13,17], merges the points within the resulting connected components, and then rebuilds the spanning tree for the new, merged clusters. Researchers are putting much time in developing new clustering algorithm[21,25,31] now a days.

In 1988, A.K. Jain presented an overview of pattern clustering. Pham DT et al. proposed a cluster number determination method in their paper in 2005[14].

Hung Ming Chaun et al.[15] proposed a fast k-means implementation that achieves results equivalent to standard methods. Their technique involves dividing the dataset into subunits called "unit blocks" (UBs), each containing at least one data point. The centroid of each unit block (CUB) is then computed. These CUBs form a significantly reduced dataset that accurately approximates the original data's structure, allowing the k-means algorithm to run much faster on this smaller representation.

The reduced dataset was then utilized to get the final centroid of the original dataset. They investigated each UB on the edge of possible clusters to get the closest final centroid for each pattern in the UB. In this method, they significantly decreased the time required to calculate the final converged centroids.

* Author for correspondence. e-mail: sjahan.mat@du.ac.bd

In 2018, Xiabo et al. introduced CciMST, a new clustering approach based on Minimum Spanning Trees (MST) that makes use of the cluster centre initialization procedure. Initially, in order to accurately represent the underlying organisation of the data sets, they proposed a method for initialising cluster centres that relies on geodesic distance and point dual densities. Furthermore, they proposed and demonstrated that the inconsistent edge is located on the shortest route connecting the cluster centres, enabling them to identify the inconsistent edge by considering the lengths of the edges and the densities of their endpoints. Similarly, they generated two sets of clustering results. Furthermore, they introduced an innovative approach to distinguish between clusters by calculating the distance between the points where clusters intersect[16].

In 2021, Chunqiong Wu et al[18]. proposed an algorithm. Their study included numerous examples of parallel k-means. They initiated random sampling and then parallelized the distance computation method, which ensures independence between data items and allows them to undertake cluster analysis in parallel. Following MapReduce's parallel processing, they employed many nodes to calculate distance, increasing the algorithm's efficiency.

In 2023, Zubair et al. suggested an efficient method for finding the ideal starting centroids, reducing the number of iterations and execution time in their study. They used several real-world datasets to conduct tests. They started by analysing COVID-19 and patient databases to demonstrate the effectiveness of their suggested strategy. The suggested algorithm's performance was also estimated using a synthetic dataset of 10 million cases with eight dimensions[19].

While conducting their survey in 2023, Wang et al. explored clusters with different densities, optimising parameter values, mitigating cascading effects, clustering extensive datasets, implementing parameter-less DPC, clustering heterogeneous data[24], and clustering unbalanced data. Subsequently, they used 26 artificial and UCI datasets[20] to examine the prevailing and up-to-date DPC expansions.

Till now, a number of significant works have been done on dimensionality reduction in the field of data clustering. Some of them are presented below:

In 2002, Ding et.al published a work that proposed the use of dimensionality reduction techniques to pre-process high-dimensional data before using clustering algorithms[5].

N. Walker et. al[6] employed Singular Value Decomposition (SVD), Principal Component Analysis (PCA), Self-Organising Map (SOM), and FastICA as four methods for reducing dimensions. The results indicated that dimension reductions significantly decrease dimensionality, enhance processing speed, and enhance cluster performance in various health datasets[6].

Arowolo et al. in 2017 used a smaller collection of genes by first selecting genes based on a single principle and then computing the maximum possible mistakes in Bayes to identify and remove unnecessary genes from the pre-selection stage[10]. Their technique was validated by employing K-Nearest Neighbour (KNN) and Support Vector Machine (SVM) classifiers on five datasets[10]. A feature selection-based dimensionality reduction model was proposed[12] for microarray biological data to enhance pre-evaluation information. The model improved the data by selecting the top-ranking pairs of features. KNN and SVM algorithms were employed to test the classification results.

The proposal by E. Amir, F. Joe[7] suggests employing targeted projection pursuit as a dimension reduction strategy to pick combinations of features for classifying a dataset. This is done to address the challenge of searching through a large power set of features and ensuring the appropriate choice of features, particularly in high-dimensional data. The TPP classification process was efficient in terms of time; however, further investigation is needed to examine the impact of output dimensionality.

Zena and Duncan in 2015 proposed variants of dimensionality reduction methods for high-dimensional microarray data[11]. Various feature selection and feature extraction methods were developed to remove redundant and unnecessary features in order to improve the accuracy of classification algorithms.

J. Yang, H. Wang[13] devised and executed a technique to reduce the dimensions of microarray datasets by employing k-means clustering algorithms. The clustering accuracy was achieved using the Laplacian eigenmap and isomap approaches. The overall result indicated that the Laplacian eigenmap outperformed the isomap in terms of reducing redundancy.

Antony et al[9]. in 2019 presented a work on the efficiency of dimensionality reduction in clustering[8]. Their study introduced a dimensionality reduction technique using a wrapper approach to enhance the accuracy of classification and clustering.

Motivation

In the DPC technique, the Euclidean distance is used to determine the distance between two points. While this measure works well for datasets with a convex shape, it's not as effective for datasets with non-convex shapes. To tackle this challenge, geodesic distance is adopted in some research work[22]. It is observed that the performance of clustering can be improved if weight is assigned in determining the distances between the points, where the weights depend on the correlation between the data points. Another important key is the parameter defining the cut-off distances to determine the density of each data point in the data field. Though this parameter plays a very important role in improving cluster results, in many clustering algorithms, it is chosen randomly. Higher dimensions of data may sometimes lead to poor clustering results. AGDPC currently operates directly on high-dimensional data without dimensionality

reduction. While this preserves the original structure of the data, high-dimensional spaces often suffer from the "curse of dimensionality," where distances between points become less meaningful, and computational complexity increases. Introducing Multidimensional Scaling (MDS) as a pre-processing step can address these challenges by projecting data into a lower-dimensional space while preserving pairwise distances or dissimilarities as much as possible. In this work, we propose to use multidimensional scaling that mitigates the impact of irrelevant or redundant features, enabling 2D/3D visualisation of clusters for interpretability. We proposed to use Regularised Multidimensional Scaling using Radial Basis Function (RBF-MDS) for dimension reduction.

Our Contribution

- We have used Supervised Regularized Multidimensional Scaling using Radial Basis Function (RBF-MDS)[36] for dimension reduction[39].
- Geodesic distances which uses weighted Euclidean distances for density measure are used where the weight is obtained from the Pearson correlation coefficient,

$$r_{pq} = \frac{\sum(p_i - \overline{p})\sum(q_i - \overline{q})}{\sqrt{\sum(p_i - \overline{p})^2}\sqrt{\sum(q_i - \overline{q})^2}}$$

  Where p and q represents two sample points, where $\overline{p} = \frac{1}{d}\sum_{i=1}^{d} p_i$ refers to the mean of all the d features of$p$ and $\overline{q} = \frac{1}{d}\sum_{i=1}^{d} q_i$ refers to the mean of all the d features of p.

- The parameters are chosen by minimizing the density estimation entropy σ

$$\min_{\sigma} E = \sum_{i=1}^{m} \frac{\rho_i}{\Phi} \log_2\left(\frac{1}{\rho_i} + 1\right)$$

- The performance of the proposed approach is compared with traditional k-means algorithm and other state of art density-based approaches Density Peaks Clustering[32,35] algorithm, Geodesic Distance based Peaks Clustering algorithm**,** and the base method AGDPC algorithm introduced recently, in terms of six clustering validation indices (CVI)AC, PR, RE, F1, ARI and SP.
- The data sets are collected from UCI repository. And One data is collected from different hospitals of Bangladesh.

The rest of the paper is structured as follows

In the next section, we have discussed the basic structure of AGDPC algorithm. The cluster centre is initialized using Density Peaks Clustering algorithm where weighted Geodesic Distance is used for estimating density of the points. The proposed two stage algorithm "Stage 1: dimension reduction using RBF-MDS" and "Stage 2: clustering using AGDPC" is also discussed in this section. The next section provides a short description of the data which are acquired from the (UCI) repository. Several medical datasets are used for comparison of the approaches. Experimental results and comparisons between the algorithms are also demonstrated in this section. Finally, the paper is concluded in Section 4.

II. Methodologies

Geodesic Distance-Based Clustering

Geodesic Distance-Based Clustering[16] is modification of Density Peaks Clustering algorithm where weighted Geodesic Distance is used for estimating density of the points for data sets having nonconvex shapes. The method uses a Minimum spanning tree.

Consider the set of n data points. $X = \{x_1, x_2, \ldots, x_n\}$, where each$x_i \in \mathbb{R}^m$. First, the data set X is represented using an undirected complete graph $G = (V, E)$, where the vertices represent the data points and the edges represent the connection between the data points. Let $T = (V, E_T)$ denote the MST of the graph G, where $|E_T| = n - 1$

For any two vertices $x_i$and x_j let $p = \{p_1, p_2, \ldots, p_j\} \in V$ is the path between them in T , where edge $(p_k, p_{k+1}) \in E_T$, $1 \le k \le l - 1$. GDPC defines the density ρ_i of point x_i in terms of geodesic distances as

$$\rho_i = \sum_j \exp\left(-\frac{D_g(x_i, x_j)^2}{\sigma^2}\right) \quad (1)$$

The variance σ is determined by taking pairwise distances of all sample points. The corresponding distance measure δ_i, is defined as

$$\delta_i = \begin{cases} \min\limits_{j:\rho_i<\rho_j} \left(D_g(x_i, x_j)\right) & \text{if } \rho_i < \rho_j \\ \max\limits_{j}\left(D_g(x_i, x_j)\right) & \text{otherwise} \end{cases} \quad (2)$$

Here $D_g(x_i, x_j) = \sum_{k=1}^{l-1} D(p_k, p_{k+1})$ represents the geodesic distance between two vertices x_i, x_j and $D(p_k, p_{k+1})$denotes the Euclidean distance between two points p_k, p_{k+1}. Using the geodesic distance between point pairs reduces the distance between points within a cluster, while the distance between points from different clusters expands.

AGDPC

AGDPC is a recently proposed[40] method that uses geodesic distance for density of each point adaptively. AGDPC introduces a weighted Euclidean distance function which incorporates the Pearson correlation coefficient[33]. This feature is absent from the original GDPC formulation. The proposed weighted Euclidean distancesis defined by

$$D_w(x_i, x_j)^2 = \frac{1}{|r_{ij}|}\left|x_i - x_j\right|^2 \quad (3)$$

Here r_{st} is the Pearson Correlation coefficient defined by $r_{st} = \frac{\sum(s_i-\bar{s})\sum(t_i-\bar{t})}{\sqrt{\sum(s_i-\bar{s})^2}\sqrt{\sum(t_i-\bar{t})^2}}$, which takes the value between 0 and 1.

$$|r_{st}| = \begin{cases} 1 & \text{if } s \text{ and } t \text{ are relevant or close} \\ 0 & \text{if } s \text{ and } t \text{ are not relevant} \end{cases}$$

Here s and t represent two sample points, $\bar{s} = \frac{1}{d}\sum_{i=1}^{d} s_i$ denotes the mean of all the d feature values of the sample s and $\bar{t} = \frac{1}{d}\sum_{i=1}^{d} t_i$ denotes the mean of all the feature values of the sample t.

The reciprocal of the Pearson correlation coefficient as a weight in Euclidean distance, redefines the distance function: higher relevance results in smaller distances, and lower relevance leads to larger distances. This weighted Euclidean distance ensures that density calculations prioritize nearby or highly relevant points, improving the accuracy of cluster centre selection.

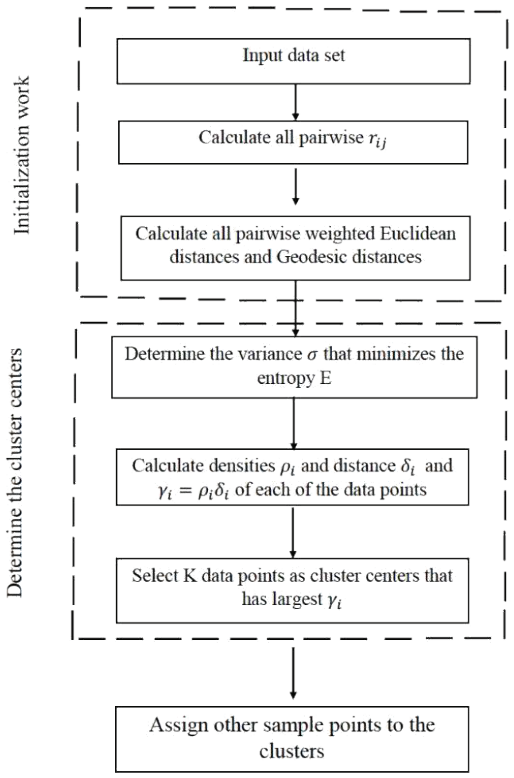


Fig. 1. The basic structure of AGDPC

The density ρ_i of point x_i in terms of geodesic distances is defined as

$$\rho_i = \sum_j \exp\left(-\frac{D_{gw}(x_i,x_j)^2}{\sigma^2}\right) \quad (4)$$

The corresponding distance measure δ_i, is defined as

$$\delta_i = \begin{cases} \min\limits_{j:\rho_i<\rho_j}\left(D_{gw}(x_i,x_j)\right) & if\ \rho_i < \rho_j \\ \max\limits_{j}\left(D_{gw}(x_i,x_j)\right) & otherwise \end{cases} \quad (5)$$

The weighted geodesic distance between two vertices x_i , x_j as

$$D_{gw}(x_i,x_j) = \sum_{k=1}^{l-1} D_w(p_k,p_{k+1})$$

where $D_w(p_k,p_{k+1})$ is weighted Euclidean distance between two points p_k and p_{k+1}.

AGDPC introduced using entropy to determine the optimal variance (σ), which is the cut-off distance in DPC selected in trial-and-error basis. The density estimation entropy[33] can be defined as

$$\mathrm{E} = \sum_{i=1}^{n} \frac{\rho_i}{\Phi}\log_2\left(\frac{1}{\rho_i}+1\right) \quad (6)$$

where, be $\rho_1, \rho_2, \ldots \rho_n$, are the local density of n sample points and $\Phi = \sum_{i=1}^{n}\rho_i$. The value of E ranges from 0 to $\log_2(n)$[33]. Selection of σ in such as way is beneficiary in identifying cluster centers when sample points have nearly identical local densities. Choosing the best value of σ is very important which will lead to the proper selection of the centre using local density. For the case when $\sigma \to 0$ or $\sigma \to \infty$,

the values of $\rho_i$will be the same and therefore the value of E will tend to its highest value $\log_2(n)$. Thus, the optimal value of σ can be adaptively selected depending on the characteristics of the data. The author follows adaptive strategy for cluster centers selection. For each of the data $\nu = \rho\delta$ is calculated. The points with higher ν values indicate the number of clusters as well as the points that will be selected as cluster centers. Fig. (6.b) and Fig. (7.b) represents plot of $\nu = \rho\delta$ for two different data sets. The working procedure of the AGDPC algorithm is given in Fig.1.

Proposed Density-Peak-Based Clustering with Dimension Reduction (RAGDPC): an extension of AGDPC

In real world, we always have to deal with high dimensional data, which often results in poor classification. To understand the hidden pattern of the data, and draw appropriate conclusion from the data, often reduction of the dimensionality of data gives better clustering results. There are many techniques of dimensionality reduction, some most popular algorithm of them are already introduced in the previous chapter. In our proposed algorithm we have at first reduced the dimensionality of the data. We divided our data in training and testing set. Then we used classification and dimensionality reduction. After that finally we have used our Adaptive clustering algorithm. For dimensionality reduction we have used RBF-MDS (MDS using Radial Basis Function)[36, 37, 38].

RBF-MDS

Consider M data points $\{x_i\}_{i=1}^{M}$ as the input where $x_i \in R^m$ and $d_{ij} = \|x_i - x_j\|$be the Euclidean distance associated to the input space. As the real-world data often contains noise, to deal with practical reasons, we consider our data also contains noise. The following methodology of dimensionality reduction was first introduced by Webb. In Webb's method, firstly, the data set is mapped to another space, known as feature space $\mathbb{R}^l$ through nonlinear radial basis function (rbf) $\Phi : \mathbb{R}^m \longrightarrow \mathbb{R}^l$.

Where $\Phi(x) = (\Phi_1(x), \ldots, \Phi_l(x)) \in \mathbb{R}^l$ defined as $\Phi_i(x) = exp\left(-\frac{||x - c_i||^2}{h^2}\right), \quad i = 1,2,\ldots,l.$

Here, h is the bandwidth and c_i is the center of Φ_i.The form of data representation in $\mathbb{R}^p$ is a linear function f of the feature vector Φ,where $f\ is$ of the form, $f(x) = W^T\Phi(x)$, $\forall x \in \mathbb{R}^m,\ W \in \mathbb{R}^{l\times p}$. Basically, the function f is a composite function of a linear function and anrbfΦ. The objective of the method is to yield the best transformation matrix that minimizes the STRESS

$$\sigma^2(W) = \sum_{i,j=1}^{N} \alpha_{ij}(q_{ij}(W) - d_{ij})^2 + \gamma\|W\|_{2,1,}^2$$

where for $i, j = 1, 2, \ldots, N, \alpha_{ij} > 0$ are known weights and $q_{ij}(W) = ||f(x_i) - f(x_j)|| = ||W^T(\Phi(x_i) - \Phi(x_j))||$. Hence, the optimization problem takes the form,

$$\min_{W\in\mathbb{R}^{\ell\times p}} \sigma^2(W) \tag{7}$$

The above problem is solved as a majorization problem. This approach relies heavily on developing the feature vector $\phi(x)$ based on its centers c_i $(i = 1, \ldots, l)$. There are two natural questions here. How to determine how many centers to use? What are the best options among those centres? Webb advises randomly selecting centres and using cross-validation to select the best one. However, the cross-validation approach is typically too expensive to implement. A new perspective on multi-task learning is offered by et.al in 2016[37] to address these problems

The algorithm of Multi-Dimensional Scaling using Radial Basis Function (RBF-MDS) is given below:

RBF-MDS Algorithm

Step 1: Input m-dimensional data set $x_i \in \mathbb{R}$, $\alpha_{ij} = 1$.

Step 2: Select the centersc_i.

Step 3: Calculate Euclidean distance $d_{ij} = ||x_i - x_j||$.

Step 4: Determine the Radial basis function

$$\Phi_i(x) = exp\left(-\frac{||x - c_i||^2}{h^2}\right), i = 1,2,\ldots,l$$

Step 5: Calculate raw STRESS

$$\sigma^2(W) = \sum_{i,j=1}^{N} \alpha_{ij}(q_{ij}(W) - d_{ij})^2 + \gamma\|W\|_{2,1,}^2$$

where for $i, j = 1, \ldots\ldots, N$ and find W such that $\min_{W\in\mathbb{R}^{\ell\times p}} \sigma^2(W)$

Step 6: Define the f as $f(x) = W^T\Phi(x),\ \forall x \in \mathbb{R}^m$

RBF-MDS is a great and effective way of Dimensionality Reduction technique. The workflow of this method is given in Fig2.

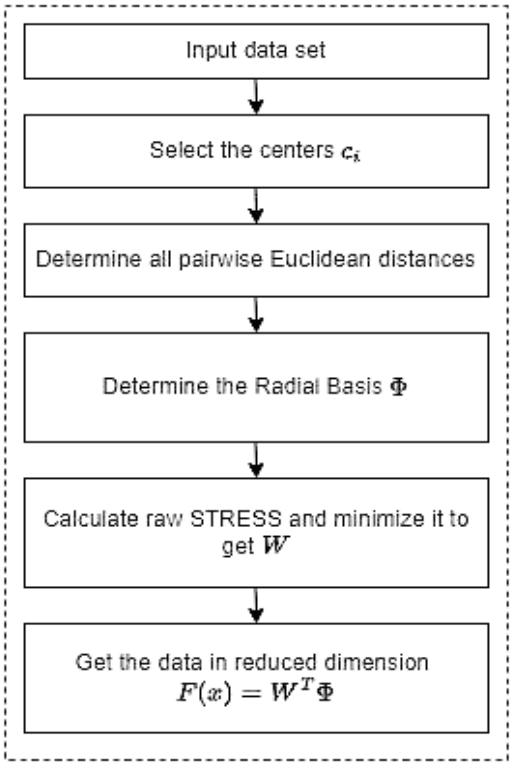


Fig. 2. RBF-MDS method

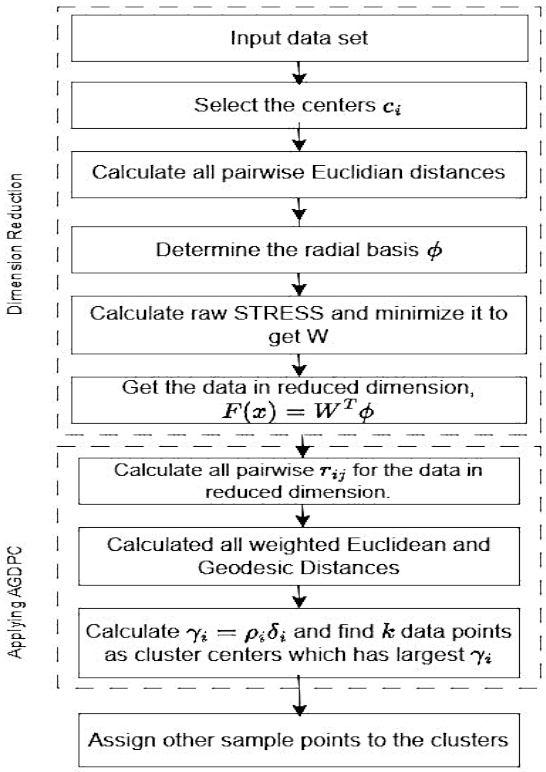


Fig. 3. Structure of RAGDPC

Proposed RAGDPC Algorithm

Step 1: Input data
Step 2: Calculate all pairwise Euclidean distances of the data
Step 3: Determine the radial basis functional values with selected centres c_i.
Step 4: Calculate and minimize the raw STRESS to get W
Step 5: The data in reduced dimension is $Y = W^T \phi$
Step 6: Calculate pairwise r_{ij} of the data in reduced dimension.
Step 7: Calculate all weighted Euclidean and Geodesic Distances
Step 8: Calculate $\gamma_i = \rho_i\delta_i$ and find k data points as cluster centres which has the largest γ_i
Step 9: Assign other sample points to the clusters
Step 10: Output: Clustered data set.

The above algorithm demonstrates the working process of RAGDPC. From the algorithm we can see that, firstly we have to transform the data set from higher dimensional space to a lower dimensional space. In the proposed RAGDPC algorithm we have used RBF-MDS as the dimension reduction method. After getting the reduced data, then the previous AGDPC approach is applied on the reduced data. The experimental result of the two approaches on different higher dimensional real-world data is given in the next section.

III. Numerical Experiment

Data set Description

In this study, five benchmarking data Cervical Cancer Behaviour Risk Data Set[22], Acute Inflammations Data Set[28] and three large-scale data, Seismic Bump[27], Estimation of Obesity Levels Based on Eating Habits and Physical Condition[29], and Mice Protein Expression[30] are used which are collected from UCI repository[22]. The Gestational diabetes mellitus (GDM) data (collected from different hospitals of Bangladesh. The details information of each of these data are given in the table. All tests of this work are carried out using the 64-bit version of MATLAB R2020a on a macOS Catalina Dual-Core Intel i5 Core CPU of 2.7GHz and 8GB of RAM

Table 1. List of data sets used in our research

Data set	no. of instance	no. of attributes	no. of classes
Acute Inflammation (AI)	120	8	2
Estimation of Obesity Levels (EOL)	2111	16	7
Seismic	2584	18	2
Cervical Cancer Risk Factor (CCBRF)	72	19	2
GDM data	117	19	2
Mice Protein Expression (MPE)	1008	77	8

Experimental Results on Real Data Sets

We have applied the four methods K-means, DPC, GDPC, AGPDC and RAGDPC to the above-mentioned data. In RAGDPC we first reduced the dimension of the data set by projecting them into lower dimensional space. Figure 4 (a, b) and Figure 5(a, b) gives scatter plot of CCBRF and seismic bump data in 2-dimensional space. Figure 4(a) and Figure 5(a) gives scatter plot of the datasets along two dimensions chosen randomly. Whereas, Figure 4(b) and Figure 5(b) gives scatter plot of the datasets projected in 2-dimensional space using RBF-MDS algorithm. Though CCBRF data projected well in 2D, figure 5(b) shows that 2D projection may not always a good choice for better clustering. So, we projected different dataset into different dimensional space that gives better clustering result. The reduced dimension of each data is documented in table 2.

Measures for Comparing Clustering Algorithm

To assess clustering performance, this study employs six validation metrics: Accuracy (AC), Precision (PR), Recall (RE), F1-score (F1), Specificity (SP), and the Adjusted Rand Index (ARI). Since each metric's value is directly proportional to clustering quality, an algorithm with higher scores is deemed superior. The metrics are computed based on a dataset of n points grouped into K true clusters $C_1, C_2, C_3, \ldots, C_k$. The core components for these calculations are defined for each cluster C_i as:

The CV indices AC, PR, RE, and F1 are defined by:

$$AC = \frac{\sum_{i=1}^{k} p_i}{n}, \quad PR = \frac{\sum_{i=1}^{k}\left(\frac{p_i}{p_i+q_i}\right)}{k}, \quad RE = \frac{\sum_{i=1}^{k}\left(\frac{p_i}{p_i+r_i}\right)}{k},$$

$$F1 = \frac{2\times PR\times RE}{PR+RE}$$

Here for a cluster C_i, p_i is the count of correctly assigned points, $q_i$is the count of incorrectly assigned pointsand $r_i$is the count of incorrectly rejected points. Also, if $U = \{U_1, U_2, \ldots, U_K\}$ be the true division of the data set in K clusters. $U_i \cap U_j = \emptyset$, $\sum_{i=1}^{k} U_i = S$ and $V = \{V_1, V_2, \ldots, V_R\}$ be the division of the dataset after implementing the clustering algorithm. Define

N_{11}: the count of true positives (TP); pairs agreeing on co-clustering in U and V;

N_{00} : the count of true negatives (TN); pairs agreeing on not co-clustering in U and V;

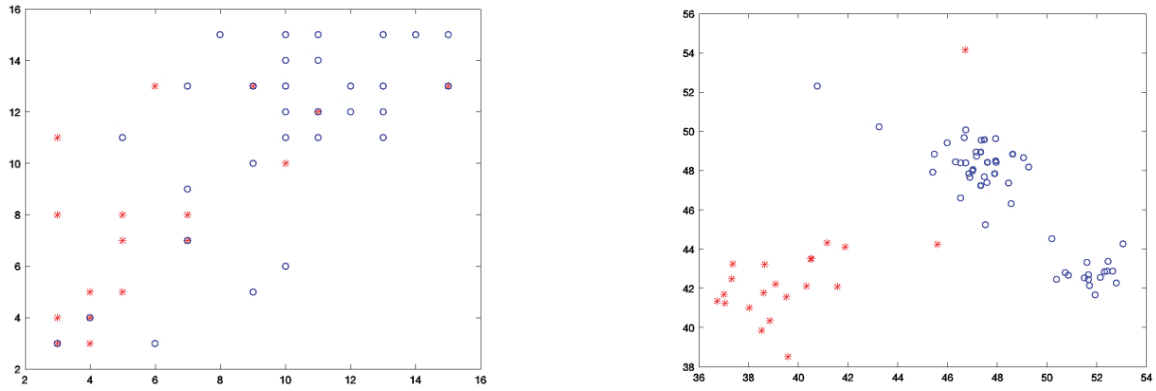
N_{10} : the count of false negatives (FN); pairs co-clusters in U but not in V;

N_{01} : the count of false positives (FP); pairs not co-clusters in U butco-clusters in V;

Then the Adjusted Rand index (ARI) [26], and Specificity (SP) are given by:

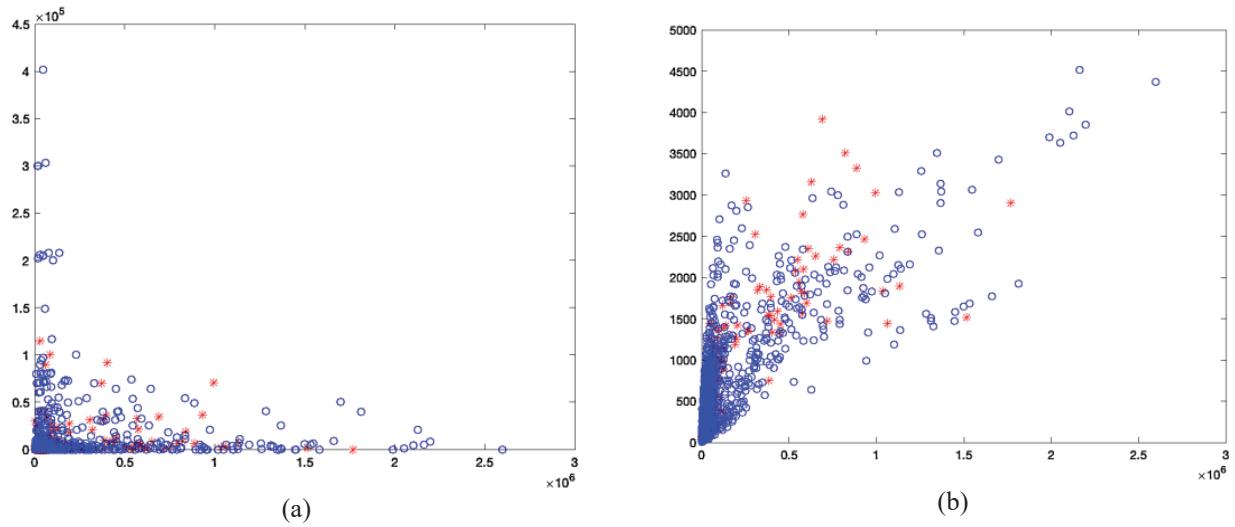
$$ARI = \frac{2(N_{11}N_{00}+N_{10}N_{01})}{(N_{11}+N_{10})(N_{10}+N_{00})+(N_{11}+N_{01})(N_{01}+N_{00})},$$

$$SP = \frac{N_{00}}{N_{10}+N_{00}}$$



(a) (b)

Fig. 4. (a) 2D scatter plot of CCBRF data before dimension reduction. (b) 2D scatter plot of projection of CCBRF data after dimension reduction.



(a) (b)

Fig. 5. (a) 2D projection of Seismic Bump data before dimension reduction. (b) 2D projection of Seismic Bump data after dimension reduction. The data is not projected well in 2D. So before using clustering algorithm we have projected it in 4D.

Table 2. Dimension of projected space for each of the datasets.

Data set	Original dimension	Reduced dimension
Acute Inflammation (AI)	8	2
Estimation of Obesity Levels (EOL)	16	4
Seismic	18	4
Cervical Cancer Risk Factor (CCBRF)	19	2
GDM data	19	6
Mice Protein Expression (MPE)	77	10

Comparison of Algorithms in terms of Clustering Validation Indices

The decision graphs in Figures 6 and 7 depict the potential cluster centres identified by the RAGDPC algorithm for the GDM data and Seismic Bump datasets, respectively. The decision graphs for other datasets can be generated following an identical process. RAGDPC algorithm generates a decision graph is a two-step process: first, reducing the data to a suitable lower dimension (e.g., 2D), and second, applying the AGDPC approach to this reduced dataset. All optimal parameters for these methods were determined through extensive experimentation. The parameter σ is selected by minimizing the density estimation entropy equation 6.DPC considers this density estimation parameter as 2% of distances of all sample points. For GDPC also considers the pairwise distances of the points in selecting σ. To ensure statistical robustness, all experiments were repeated 10 times, and the average outcomes are reported in Table 3. This table compiles the results of external (CVI)including AC, PR, RE, F1, SP, and ARI for each algorithm across every dataset. The data shows that the proposed RAGDPC algorithm achieves superior performance on all metrics compared to AGDPC, GDPC, DPC, and K-means. This dominance is also clearly illustrated in the comparative bar charts (Fig 8.a, 8.b). It is also noteworthy that GDPC itself delivers greater clustering accuracy than both DPC and K-means in the majority of cases.

The RAGDPC algorithm demonstrated superior performance on the GDM dataset, achieving the highest scores across all of the validation metrics. It recorded the best Accuracy (AC: 0.9421), Precision (PR: 0.9268), Recall (RE: 0.9587), F1-Score (F1: 0.9595), Specificity (SP: 0.9045), and Adjusted Rand Index (ARI: 0.9665). AGDPC and GDPC formed a clear second and third tier, respectively, both significantly outperforming the baseline DPC and K-means algorithms. This result indicates that RAGDPC provides the most accurate and reliable clustering for the GDM data.

AI data and CCBRF data show a similar performance for each of the validation indices. For Acute Inflammations Data, DPC and GDPC show similar results in terms of AC. All the indices for K-means show the smallest value which indicates the worst performance. Since the results are an average of 10 runs, the error bar in each of the bar charts indicates that RAGDPC is more stable than the other methods for each of the data sets. GDPC shows better stability than DPC and K-means.

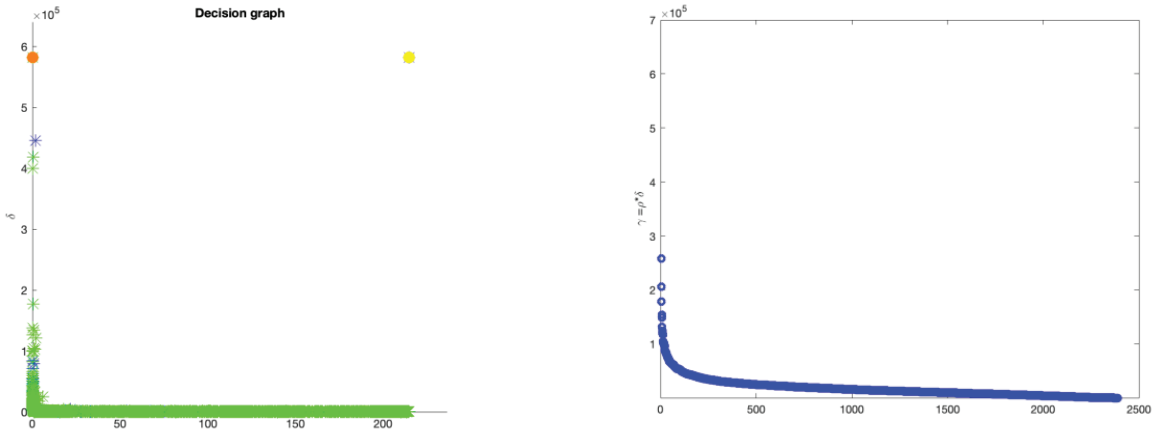
For CCBRF data in terms of stability, DPC works the same as GDPC. Also, the performance of K-means is the worst here compare to other algorithms, it yields smallest CVI values comparing to other algorithms. AGDPC and

RAGDPC has almost similar efficiency here. The CVI values of RAGDPC is slightly larger than AGDPC. So, data in reduced dimension yields more accurate clustering.

The Seismic Data results showcase a significant strength of the RAGDPC algorithm, particularly in the Adjusted Rand Index (ARI), where it achieved a score of 0.9424, vastly outperforming all other methods. It also registered the highest Precision (PR: 0.8344). The smallest error bar for several runs in comparison to other methods is an indicator of its exceptional clustering validity for this complex dataset, solidifying its position as the most effective algorithm. Similar Performance is observed for other large-scale data sets as shown in table 3.

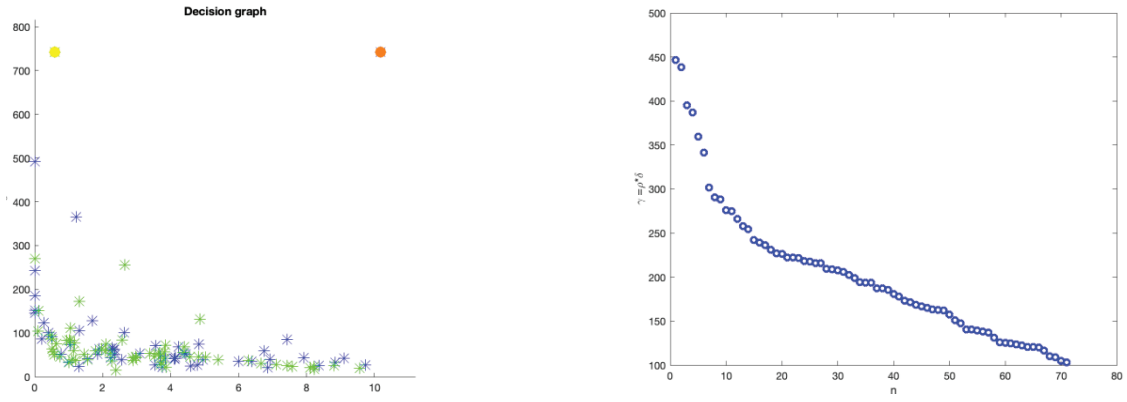
CPU Time

High-dimensional data is computationally expensive to process. Therefore, reducing the dimension of the data first should reduce the computational cost of the subsequent clustering step (AGDPC). Across all six diverse datasets, RAGDPC consistently achieved the fastest execution times, significantly outperforming all other methods, including AGDPC, GDPC, DPC, and K-means. For instance, on the larger and more complex datasets (seismic bump, EOL data and MP data) RAGDPC completed clustering in just 189.45, 179.27, and 328.94 seconds respectively, which is less than half the time required by the next fastest algorithm. This dramatic reduction in runtime up to a 55%decrease compared to AGDPC directly validates the core design principle of RAGDPC, by first applying a strategic dimensionality reduction, the computational burden of the subsequent clustering step is drastically alleviated. Therefore, RAGDPC successfully delivers a dual advantage, it maintains the high clustering accuracy of advanced density-based methods while achieving a level of computational speed that makes it highly practical for analysing large-scale[21] data which is clearly visible in Table 4 as well as in Figure 9.



(a) (b)

Fig. 6. (a) Decision graph for Seismic data, (b) $\gamma = \rho\delta$: representing γ values of the points of Seismic data. Higher value of γ more potential of being cluster center.



(a) (b)

Fig. 7. (a) Decision graph GDM data: Red and yellow marked points are the selected centers. (b) $\gamma = \rho\delta$: representing the γ of the points of GDM Data.

Table 3. Comparison of performance of different methods in terms of CVI

Dataset	Measure	K-means	DPC	GDPC	AGDPC	RAGDPC
GDM Data	AC	0.7112	0.8423	0.8741	0.9321	**0.9421**
	PR	0.7032	0.8232	0.8424	0.9173	**0.9268**
	RE	0.6921	0.8589	0.8636	0.9465	**0.9587**
	F1	0.6841	0.8386	0.8435	0.9452	**0.9595**
	SP	0.6175	0.8136	0.8272	0.8913	**0.9045**
	ARI	0.8234	0.8795	0.9434	0.9542	**0.9665**
AI Data	AC	0.6023	0.8383	0.90428	0.94321	**0.94431**
	PR	0.5812	0.83337	0.85513	0.9012	**0.91321**
	RE	0.6724	0.8505	0..87939	0..92452	**0.92667**
	F1	0.5876	0.8417	0.85224	0.91212	**0.9411**
	SP	0.6527	0.8388	0.90428	0.94321	**0.9533**
	ARI	0.7219	0.8976	0.9582	0.9823	**0.9855**
CCBRF Data	AC	0.7305	0.8215	0.85278	0.90124	**0.91431**
	PR	0.6397	0.7507	0.79888	0.85024	**0.8912**
	RE	0.5875	0.7123	0.76486	0.83562	**0.85221**
	F1	0.6125	0.7309	0.78145	0.87123	**0.87551**
	SP	0.7305	0.8215	0.85278	0.90124	**0.9115**
	ARI	0.8134	0.8251	0.8972	0.9634	**0.9645**
Seismic Data	AC	0.7305	0.8215	0.85278	0.90124	**0.9124**
	PR	0.6723	0.7453	0.7828	0.8245	**0.8344**
	RE	0.6017	0.7219	0.77368	0.8317	**0.8429**
	F1	0.6345	0.8023	0.8172	0.8923	**0.9078**
	SP	0.8145	0.8426	0.8952	0.9235	**0.9186**
	ARI	0.7305	0.8215	0.85278	0.90124	**0.9424**
EOL Data	AC	0.7217	0.8124	0.8348	0.9524	**0.9323**
	PR	0.7032	0.8236	0.8528	0.9376	**0.9161**
	RE	0.6943	0.8385	0.8739	0.9567	**0.9482**
	F1	0.6824	0.8582	0.8832	0.9251	**0.9293**
	SP	0.6175	0.8333	0.8473	0.8114	**0.9346**
	ARI	0.8277	0.8296	0.9231	0.9345	**0.9462**
MPE Data	AC	0.7201	0.8314	0.81277	0.9123	**0.9425**
	PR	0.6322	0.7555	0.7325	0.8446	**0.8646**
	RE	0.5244	0.7122	0.7234	0.8225	**0.8346**
	F1	0.6315	0.7415	0.7463	0.8513	**0.8628**
	SP	0.6542	0.8221	0.8272	0.8627	**0.9277**
	ARI	0.8245	0.8227	0.8351	0.9232	**0.9488**

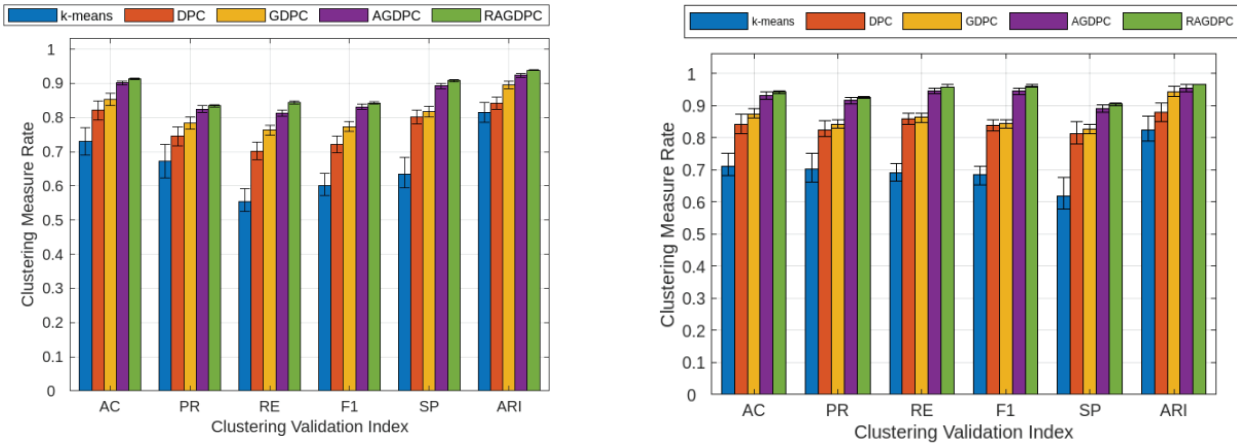


Fig. 8. CVI measure for (a) Seismic Data (b) GDM data

Table 4. CPU time in seconds for different methods

Dataset	K-means	DPC	GDPC	AGDPC	RAGDPC
GDM Data	170.34	185.54	186.14	189.54	**89.54**
AI Data	111.50	127.52	127.51	128.71	**78.76**
CCBRF Data	128.72	146.75	148.25	150.32	**70.32**
Seismic Data	583.35	640.67	590.21	599.45	**189.45**
EOL Data	620.34	610.47	614.28	619.27	**179.27**
MPE Data	700.54	735.72	735.87	738.94	**328.94**

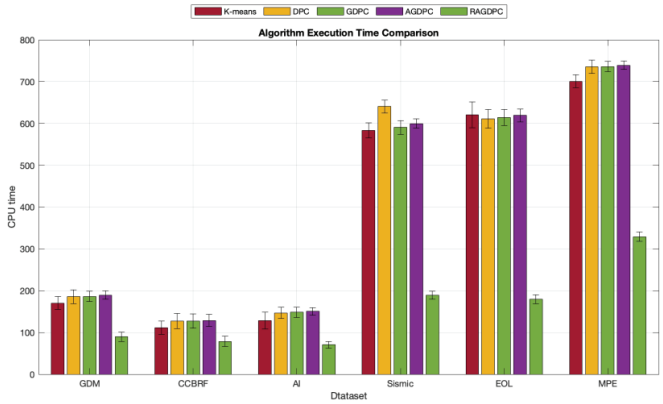


Fig. 9. CPU time for each of the Data.

IV. Conclusion

DPC algorithm is one of the most traditional algorithms that select cluster centers using the density of the data points. But DPC does not take into consideration the intrinsic structure of the data field and therefore works poorly for data set with non-convex shape. To avoid this issue a cluster center initialization algorithm based on geodesic distances is proposed. In AGDPC, the geodesic distances are determined by incorporating Pearson correlation Coefficient as weight to Euclidean distances. Also, a density estimation entropy is minimized to adaptively optimize the threshold parameter for the algorithm and therefore an adaptive strategy is followed to select the cluster centers. However, higher dimension of data may lead to poor clustering and computationally expensive. So, projection of data into lower dimensional space is expected to provide better clustering in less computational time. In this work we have proposed to apply dimension reduction algorithm RMDS to project the data into lower dimensional space. For visualization purpose some of the data projected into 2-dimensional space but for better clustering, the dimension of the projected spaces varies. The efficiency of the proposed approach RAGDPC is compared with the traditional method k-means and other density-based methods DPC, GDPC and AGDPC. All of these algorithms are applied to several benchmarking data such CCBRF data, GDM data, AI data, Seismic Bump data, MPE data and EOL Data, collected from various sources. Six clustering validation indices (CVI) are calculated for the clustering results. Numerical experiments show that RAGDPC outperform all other algorithms in terms of the CVI. Moreover, due to the reduced dimension the method also uses almost 55% less computational time that other methods.

Acknowledgment

This research work is funded by special allocation of Government of the People's Republic of Bangladesh Ministry of Science and Technology 2023-24.

References

1. Driver, H. A. Kroeber, 1932, Quantitative Expression of Cultural Relationships, University of California Publications in American Archaeology and Ethnology, University of California Press, 211–256
2. Zubin, J. 1938, A Technique for Measuring Like-Mindedness, 33, 508–516, https://doi.org/10.1037/h0055441
3. Tryon, R. C. 1939, Cluster Analysis: Correlation Profile and Orthometric (Factor) Analysis for the Isolation of Unities in Mind and Personality, Edwards Brothers
4. Cattell, R. B. 1943, The Description of Personality: Basic Traits Resolved into Clusters, **38**, 476–506, https://doi.org/10.1037/h0054116
5. Ding, C. X. He, H. Zha, H. Simon, 2002, Unsupervised Learning: Self-Aggregation in Scaled Principal Component Space, 112–124, Springer Berlin Heidelberg
6. Walker, N. 2015, International Journal of New Computer Architectures and Their Applications (IJNCAA), 5
7. Amir, E. F. Joe, 2015, Feature Selection with Targeted Projection Pursuit, 5, 34–39, International Journal of Information Technology and Computer Science
8. Liu, L. S. Ma, L. Rui, J. Lu, 2017, Locality Constrained Dictionary Learning for Non-Linear Dimensionality Reduction and Classification, **11**, 60–67, IET Computer Vision
9. Asir Antony Gnana Singh, D. E. Jebamalar Leavline, 2019, Dimensionality Reduction for Classification and Clustering,

International Journal of Intelligent Systems and Applications (IJISA), 11, 61–68, https://doi.org/10.5815/ijisa.2019.04.06

10. Arowolo, M. O. S. O. Abdulsalam, R. M. Isiaka, K. A. Gbolagade, 2017, A Hybrid Dimensionality Reduction Model for Classification of Microarray Dataset, International Journal of Information Technology and Computer Science, 9, 57–63
11. Zena, M. F. G. Duncan, 2015, A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data
12. Songlu, L. O. Sejong, 2016, Improving Feature Selection Performance Using Pairwise Pre-evaluation, BMC Bioinformatics, **17**, 1–13
13. Yang, J. H. Wang, H. Ding, A. Ning, A. Gil, 2017, Nonlinear Dimensionality Reduction for Synthetic Biology Biobrick Visualization, BMC Bioinformatics, **17**, 4, 1–10
14. Pham, D. T. S. S. Dimov, C. D. Nguyen, 2005, Selection of K in K-means Clustering, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 219, 103–119, https://doi.org/10.1243/095 4 4 0605X8298
15. Hung, M. J. Wu, J. Chang, D. Yang, 2005, An Efficient k Means Clustering Algorithm Using Simple Partitioning, Journal of Information Science and Engineering, 21, 1157–1177
16. X. Lv, Y. Ma, X. He, H. Hui, J. Yang, 2018, CciMST: A Clustering Algorithm Based on Minimum Spanning Tree and Cluster Centers, Mathematical Problems in Engineering, 1–14, https://doi.org/10.1155/2018/8451796
17. Grygorash, O. Z. Yan, Z. Jorgensen, 2006, Minimum Spanning Tree-Based Clustering Algorithms, Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence, 73–81
18. Wu, C. B. Yan, R. Yu, B. Yu, X. Zhou, Y. Yu, N. Chen, 2021, k-Means Clustering Algorithm and Its Simulation Based on Distributed Computing Platform, Complexity, 1–10, https://doi.org/10.1155/2021/9446653
19. Zubair, M. M. A. Iqbal, A. Shil, 2022, An Improved K-means Clustering Algorithm Towards an Efficient Data-Driven Modelling, Annals of Data Science
20. Wang, Y. J. Qian, M. Hassan, X. Zhang, T. Yang, C. Zhou, X. Jia, F. Zhang, 2023, Density Peak Clustering Algorithms: A Review on the Decade 2014–2023, Expert Systems with Applications, 238, 121860, https://doi.org/10.1016/j.eswa.2023.121860
21. Shao, H. P. Zhang, X. Chen, F. Li, G. Du, 2019, A Hybrid and Parameter-Free Clustering Algorithm for Large Data Sets, IEEE Access, **7**, 24806–24818
22. UCI Machine Learning Repository, 2011, http://www.ics.uci.edu/ mlearn/MLRepository.html
23. Sokolov, A. A. Ben-Hur, 2010, Hierarchical Classification of Gene Ontology Terms Using the GOstruct Method, Journal of Bioinformatics and Computational Biology, **8**, 357–376, https://doi.org/10.1142/S0219720010004744
24. Yizhang Wang, Jiaxin Qian, Muhammad Hassan, Xinyu Zhang, Tao Zhang, Chao Yang, Xingxing Zhou, Fengjin Jia, 2024, Density Peak Clustering Algorithms: A Review on the Decade 2014–2023, Expert Systems with Applications, 238, 121860, https://doi.org/10.1016/j.eswa.2023.121860
25. Huang, Y. J. R. Powers, G. T. Montelione, 2005, Protein NMR Recall, Precision, and F-Measure Scores (RPF Scores): Structure Quality Assessment Measures Based on Information Retrieval Statistics, Journal of the American Chemical Society, **127**, 1665–1674
26. Hubert, L. P. Arabie, 1985, Comparing Partitions, Journal of Classification, **2**, 193–218
27. Sikora, M. L. Wrobel, 2010, Seismic-Bumps Dataset, UCI Machine Learning Repository, https://doi.org/10.24432/C5W902.
28. Czerniak, J. 2009, Acute Inflammations, UCI Machine Learning Repository.
29. Palecho, F. M. A. D. Manotas, 2019, Dataset for Estimation of Obesity Levels Based on Eating Habits and Physical Condition in Individuals from Colombia, Peru and Mexico, Data in Brief.
30. Higuera, C. K. Gardiner, K. Cios, 2015, Mice Protein Expression, UCI Machine Learning Repository.
31. Theodoridis, S. K. Koutroumbas, 2010, An Introduction to Pattern Recognition: A MATLAB Approach, Elsevier Inc.
32. Rodriguez, A. A. Laio, 2014, Clustering by Fast Search and Find of Density Peaks, Science, **344**, 1492–1496.
33. Sun, L. R. Liu, J. Xu, S. Zhang, 2019, An Adaptive Density Peaks Clustering Method with Fisher Linear Discriminant, IEEE Access, **7**, 72936–72955.
34. Luo, T. C. Zhong, 2010, A Neighbourhood Density Estimation Clustering Algorithm Based on Minimum Spanning Tree, Lecture Notes in Computer Science, 6401, 557–565.
35. Li, C. G. Ding, D. Wang, Y. Li, Y. Yan, S. Wang, 2016, Clustering by Fast Search and Find of Density Peaks with Data Field, Chinese Journal of Electronics, **25**, 397–402.
36. Islam, M. T. M. A. Islam Bhuiyan, S. Jahan, 2022, Supervised Regularized Multidimensional Scaling Using Weighted Stress Measure, Universal Journal of Applied Mathematics, **10**, 20–32.
37. Jahan, S. H. Qi, 2016, Regularized Multidimensional Scaling with Radial Basis Functions, Journal of Industrial and Management Optimization, **12**, 543–563.
38. Jahan, S. 2021, Discriminant Analysis of Regularized Multidimensional Scaling, Numerical Algebra, Control & Optimization, **11**, 255–267.
39. Jahan, S. A. A. Setu, S. Muntaha, T. Biswas, 2025, Adaptive Cluster Center Initialization Using Density Peaks for Geodesic Distance-Based Clustering, Numerical Algebra, Control and Optimization, 15, 601–618, https://doi.org/10.3934/naco.2024016.