Computing, Information Systems, Development Informatics & Allied Research Journal
Vol 7 No 1 March 2016 - www.cisdijournal.net

Swarm-Optimized K-Means Algorithm for Clustering Real World Datasets

Adigun A. A., Egbetola I. F., Oke O. A. & Fenwa O. D.
Department of Computer Science and Engineering,
Ladoke Akintola University of Technology
Ogbomoso, Nigeria.
aaadigun@lautech.edu.ng, funmilolaegbetola@yahoo.com, aooke@lautech.edu.ng and odfenwa@lautech.edu.ng

ABSTRACT

K-means is one of the most widely used classical partitioned clustering algorithms due to its simplicity of implementation, speed of convergence and adaptability to sparse data. However, it does not guarantee global convergence because it only can generate local optimal solution which in turn maximizes its clustering error and as such not suitable for large dataset. Most currently existing improvements on k-means adopt techniques which further incur addition challenges including quick but inaccurate global convergence, inaccurate clustering results, high time and space complexities as well as premature convergence on k-means. Sequel to this, a swarm-optimized k-means clustering algorithm was developed. The developed k-means algorithm combines the global consistency property of Particle Swarm Optimization (PSO) technique with the stable properties of a density-sensitive distance metric to avoid convergence of particles to local minima. The developed algorithm was used to cluster Real World Datasets and its clustering accuracy and time taken to converge were measured. The performance evaluation of the traditional K-means, PCA-based HYBRID (K-PSO), UFT-*K*-means and the developed k-means algorithm were evaluated on the Real World Datasets using the two performance metrics. However, the developed swarm-optimized k-means algorithm had a significant improved performance over the others especially in terms of clustering accuracy. In the same vein, the developed swarm-optimized k-means can identify non-convex clustering structures, thus generalizing the application area of the k-means algorithm.

Keywords – K-Means, Particle Swarm Optimization, Clustering, Large Dataset, Density-Sensitive Distance Metric

1. INTRODUCTION

Data mining is the analysis of datasets that are observational, aimed at finding out unsuspected relationships among datasets and summarizing the data in such a noble fashion that are both understandable and useful to the data users (Neelamadhab, Pragnyaban and Rasmita, 2012). It involves the use of sophisticated data analysis tools to discover previously unknown, valid pattern and relationship in large dataset. It also makes data description possible by means of clustering, visualization, association and sequential analysis. However, clustering is one of the broad fields of data mining. Clustering is the process of partitioning or grouping a given set of patterns into disjoint clusters so that data in each cluster share some common trait (Gursharan and Harpreet, 2014). This is done such that patterns in the same cluster are alike and patterns belonging to two different clusters are different. Clustering aims at finding smaller similar groups from a larger collection of items. Clustering data has been core to many scientific and engineering problems and it has been widely studied in a variety of real time application domains including neural networks, pattern recognition, computer vision, artificial intelligence and statistics among others. As a result, simple, fast and efficient algorithms are desirable (Nasser, Alkhaldi and Vert, 2004).

Generally, clustering algorithms mainly fall into two categories: Hierarchical clustering and partition clustering. In partitioned clustering algorithm, data is partitioned into more than two subgroups in one steps and in hierarchical clustering algorithm, data is divided into two subgroups in each step. Several algorithms have been used for clustering including k-means, Expectation Maximization (EM), ISODATA, CLARA, CLARANS, focusing techniques, P-CLUSTER. DBSCAN, Ejcluster, BIRCH, GRIDCLUS, Single Linkage, Complete Linkage, Fuzzy Clustering, Genetic Algorithms (GA) and Self Organizing Map (SOM) among others (Rauber, 2000). K-means is one of the most widely used classical partitioned clustering algorithm introduced in the field of data mining to partition a dataset into groups of patterns. However, it does not guarantee global convergence as it can only generate local optimal solution, it produces inaccurate results for large dataset and highly sensitive to noisy data (Sharfuddin, Mohammad, Dip and Mashiour, 2015; Adigun, Omidiora, Olabiyisi, Adetunji and Adedeji, 2012).

Numerous improvements to k-means have been done to make its performance better. Approaches based on optimization techniques including unsupervised feature transformation (Min, Tommy and Rosa, 2015), principal component analysis (Chetna and Garima, 2013), genetic algorithms (Jenn-Long, Yu-Tzu and Chih-Lung, 2012), tabu-search (Zhang and Sun, 2002) and simulated annealing (Siedlecki and Sklansky, 1988) produced improved performance but are tested on reasonably small and normalized datasets only. These set of solutions are not directly suitable for large real world datasets. Hence, there is a need to manage these limitations and in a more accurate and computationally efficient manner. Therefore, in this research work, an improved k-means clustering algorithm was developed.

2. RELATED WORKS

Ming-Chuan, Jungpin, Jin-Hua and Don-Lin (2005) modified k-means clustering algorithm using simple partitioning method. The authors highlighted that most *k*-means methods require expensive distance calculations of centroids to achieve convergence. To manage this discrepancy, the original dataset was partitioned into blocks using binary splitting; each block unit, called a unit block (UB), contains at least one pattern. The centroid of a unit block (CUB) was located via a simple calculation. All the computed CUBs form a reduced dataset that represents the original dataset. The reduced dataset was then used to compute the final centroid of the original dataset. Each UB was examined on the boundary of candidate clusters to find the closest final centroid for every pattern in the UB. In this way, the time for calculating final converged centroids was dramatically reduced. It was claimed that the algorithm showed significant improvement in performance in terms of total execution time, the number of distance calculations and the efficiency for clustering than other *k*-means algorithms. However, the modified k-means needed more iterations to achieve the *k* centroids sometimes even spending the maximum number of iterations will not achieve convergence.

Mary and Raja (2009) used the Ant Colony Optimization (ACO) algorithm to improve K-means clustering. The authors improved the cluster quality after grouping. The developed method has two phases. In the first phase, on the basis of statistical modes, initial centroids for K-means clustering are selected. In the second phase, they improve the cluster quality by using ant refinement algorithm. The resultant technique still uses Euclidean distance and is highly sensitive to the changes in the value of the initial k. This makes it less applicable for clustering real world datasets. Qian and Xinjian (2011) developed an improved k-means algorithm in gene expression data analysis based on the Kruskal algorithm. Firstly, the minimum spanning tree (MST) of the clustered objects is obtained by using Kruskal algorithm. Then, K-1 edges are deleted based on weights in a descending order. At last, the average values of the objects contained by the k-connected graphs resulting from last the two steps are regarded as the initial clustering centers to cluster. The results of experiment showed that this method lessened its dependence on initial cluster centers than traditional K-means algorithm and increased the stability and accuracy of clustering results. However, the developed K-means algorithm failed when tested on large, complex, vast and real-time datasets and suffers from high time complexity. In addition, the developed technique has high program difficulty.

Adigun, Omidiora, Olabiyisi, Adetunji and Adedeji (2012) developed a hybrid k-means – Expectation Maximization (KEM) algorithm to manage the limitations of K-means and Expectation Maximization (EM) algorithms. K-means converges only to local minima after large number of trials while EM converges prematurely. The hybrid KEM algorithm was developed via two (2) stages: the initialization stage and the iterative stage. In the initialization stage, the weighted average variation of the K-means algorithm was used to classify the data into the number of clusters desired. At the iterative stage, a large number M, of uniformly distributed random cluster point vectors for the cluster centers are selected. Any cluster point vectors that are too close to other cluster point vectors are eliminated and M is reduced accordingly until the clusters produced equal to K, the number of desired clusters. This was achieved by computing the distances between all the clusters, and eliminating the clusters with distances lesser than ϵ (a value that is selected experimentally). Assigning each of the feature vectors to the nearest random cluster point vector, is the next step, and was achieved by computing the distance between each feature vector and all other cluster point vectors. The feature vector was assigned to the cluster point vector such that the distance between them is the shortest. The hybrid algorithm showed improvements over k-means and EM more accurately in a computationally efficient manner and was tested on real world educational dataset. However, the hybrid KEM still converges to local minima because the K-means component used Euclidean distance metric and as such not suitable for clustering large real world dataset. The hybrid KEM was not developed to handle noise which characterizes the real world datasets.

Shanmugapriya and Punithavalli (2012) developed a modified projected K-Means clustering algorithm with effective distance measure that continuously optimizes a comprehensive objective function. In the objective function of this developed algorithm, an effective distance measure makes use of local and non-local information to provide better clustering results in high dimensional data. In order to avoid the value of the objective function from decreasing as a consequence of the exclusion of dimensions, virtual dimensions are incorporated with the objective function. It only works efficiently in principle as the developed algorithm was not evaluated. Nidhi and Ujjwal (2013) developed an incremental k-means clustering algorithm that assigns any random data object to the first cluster of a given set of data objects. After selecting the next random object, the distance between selected object and centroids of existing clusters was determined. This distance was compared with the threshold limit in order to group the object into existing cluster or form a new cluster with that object. Experimental results revealed that the developed algorithm produced clusters in lesser computation time but only with small and noise-free dataset. It cannot handle large, noisy dataset in a computationally efficient manner due to the rigid nature of the incremental approach used.

Chetna and Garima (2013) developed a linear principal component analysis (PCA) based hybrid K-Means particle swarm optimization (PSO) algorithm for clustering large dataset. The flowchart of the hybrid technique is presented in Figure 1. PCA module was executed to convert high dimensional data to low dimensional one using covariance matrix. Then, the K-means clustering algorithm was made to search for the clusters' centroid locations using the Euclidean distance similarity metric. This information was passed to the PSO module for the generation of the final optimal clustering solution as the result. In general, PSO conducts a global search for the optimal clustering, but more iteration is required. The PSO was complimented by K-means clustering to start with good initial cluster centroids that converge faster thereby giving a more compact result. The result from the K-means module was treated as the initial seed for the PSO module to discover the optimal solution by a globalized search to avoid high computational time complexity. Better clustering result was obtained with PCA-based HYBRID (K-PSO) algorithm when compared with PSO only. The hybrid system is largely complex, converged to local minima, incurred high computational overhead and was not evaluated with other improved K-means variants.

Li, Lei, Bo, Yue and Jin (2015) developed an improved K-means algorithm based on map reduce and grid. The improved method is divided into the same grid in space according to the size of the data point property value and assigns it to the corresponding grid. It counts the number of data points in each grid, selects M (M>K) grids, comprising the maximum number of data points and calculates the central points. These M central points serve as input data to determine the k value based on the clustering results. In the M points, it finds K points farthest from each other and those K center points as the initial cluster center of K-means clustering algorithm. At the same time, the maximum value in M was included in K. If the number of data in the grid is less than the threshold, then these points were considered as noise points and were removed. In order to make the improved algorithm adapt to handle large data, the improved k-mean algorithm was paralleled and combined with the MapReduce framework. Theoretical analysis and experimental results show that the improved algorithm compared to the traditional K-means clustering algorithm has high quality results, less iteration and good stability. Parallelized algorithm has a very high efficiency in data processing with good scalability and speedup.

Sharfuddin, Mohammad, Dip and Mashiour (2015) argued that the current minimum distance in traditional K-means is not always the correct minimum distance because the distance between a cluster center and each data point is measured in every iteration. This makes the algorithm more complex and increases the number of computations. In the modified version of K-means algorithm developed by the authors, one check point value was added to store the center point of the distance of two cluster centers and used to determine the cluster any object is going to be assigned to. This check point value reduced the possibility of error during the clustering process. The authors reported that the modified K-means requires less computation and has enhanced accuracy than the traditional K-means algorithm as well as some modified variant of the traditional K-Means. However, shortage of available resources and time limited the work. The developed k-means algorithm was not tested on large, complex, vast and real world datasets.

Min, Tommy and Rosa (2015) clustered heterogeneous data with *k*-means by mutual information-based unsupervised feature transformation (UFT). The work addressed the computational intensiveness of k-means algorithm for datasets with large sample sizes as well as its sensitive nature to outliers and inability to update the clustering structure while processing data. To address these challenges, the mutual information-based unsupervised feature transformation which can transform non-numerical features into numerical features was integrated with the conventional *k*-means to cluster the heterogeneous data. The results of simulation studies show that, the integrated UFT-*k*-means outperformed other clustering algorithms and provided reasonable clusters for one modified real-world dataset and five real-world benchmark datasets. However, the developed algorithm is parameter dependent and highly computationally inefficient.

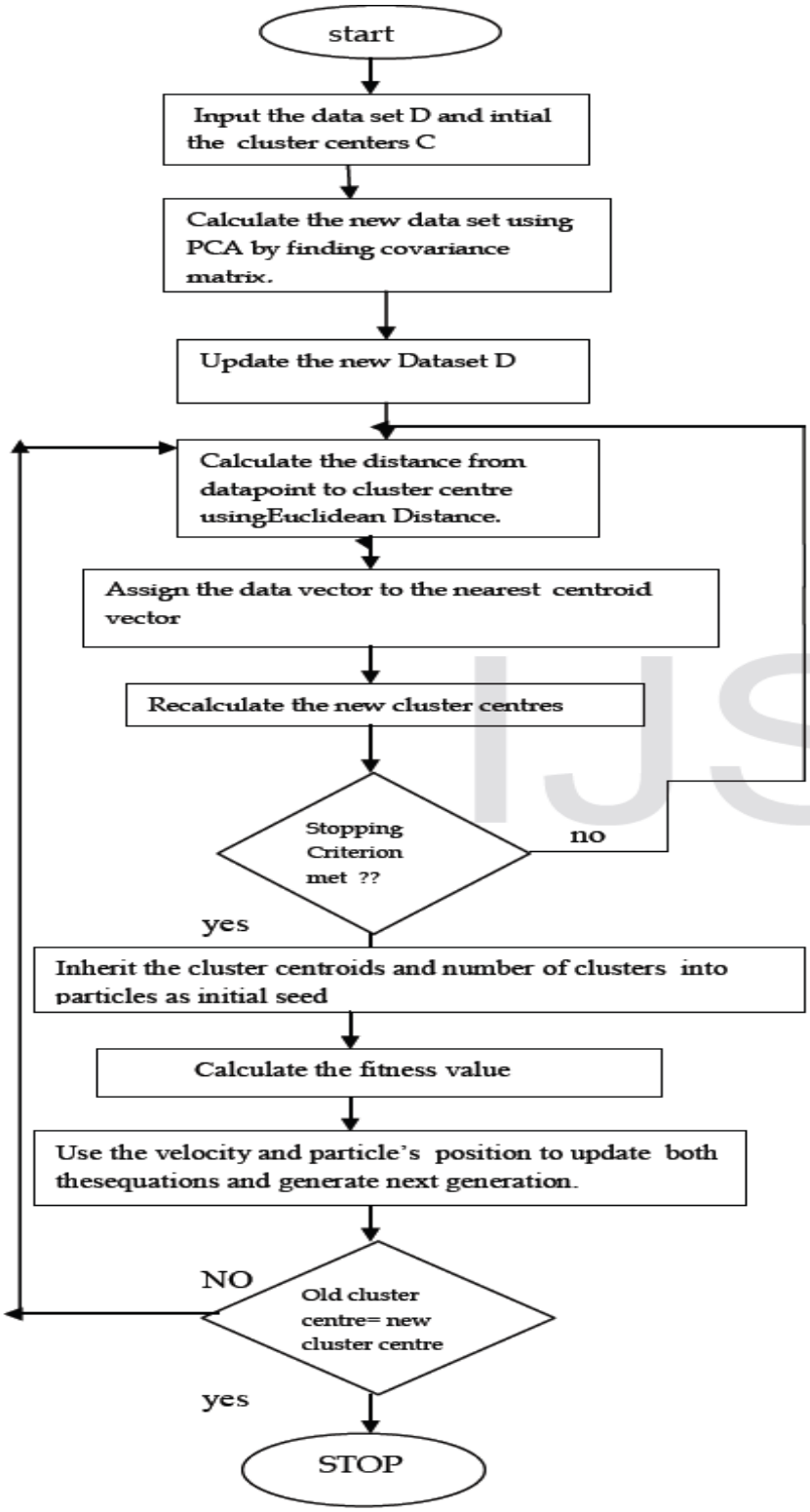


Figure 1: The flowchart of the PCA-based HYBRID (K-PSO) (Chetna and Garima, 2013)

3. MATERIALS AND METHODS

This section presents the statement of the problem, k-means, PSO, the density-sensitive distance measure, the developed swarm optimized k-means, the EPM real world dataset and the performance evaluation metrics.

3.1 Statement of the Problem

K-means does not guarantee global convergence as it can only generate local optimal solution which in turn leads to high clustering error and as such not suitable for large dataset (Ling, Liefeng and Licheng, 2012; Su and Chou, 2001). With certain real world critical datasets emerging from security, medical and finance sectors, errors as a result of limitations arising from K-Means algorithm is highly unacceptable. K-means uses the Euclidean distance dissimilarity measure, however, due to the complex structure of data points with convex distribution, the measure fails in an attempt to obtain correct clusters for all the data points since the objective function of K-Means is not convex (Ling, Liefeng and Licheng, 2012). This indicates that Euclidean distance measure is undesirable when clusters have such random distributions. As a result, K-means is not suitable for large and real world datasets because it fails in an attempt to describe global consistency of data which is more crucial for accurate clustering (Li, Lei, Bo, Yue and Jin, 2015; Amita and Ashwani, 2014). However, most currently existing improvements on K-means adopt techniques such as Unsupervised Feature Transformation (UFT) (Min, Tommy and Rosa, 2015), Principal Component Analysis (PCA) (Chetna and Garima, 2013), genetic algorithm (Jenn-Long, Yu-Tzu and Chih-Lung, 2012), expectation maximization (Adigun, Omidiora, Olabiyisi, Adetunji and Adedeji, 2012), mapreduce and grid (Li, Lei, Bo, Yue and Jin, 2015) and were either tested only on limited dataset size and did not address the limitations inherent in K-means wholly in a timely efficient and accurate manner. As a result, major improvement to K-means still remains largely an open problem. Therefore, this research has developed an improved K-means clustering algorithm to manage these aforementioned drawbacks wholly in a computationally-efficient manner.

3.2 The k-means algorithm

According to Azhar*et al*.(2012), the k-means algorithm works as follows:

INPUT: Number of desired clusters K
Given: Data objects $D = \{d_1, d_2 \ldots, d_n\}$
OUTPUT: A set of K clusters

i. Specify the number of clusters (k in k-means)
ii. Randomly select k cluster centers in the data space
iii. Assign data points to clusters based on the shortest Euclidean distance to the cluster centers
iv. Re-compute new cluster centers by averaging the observations assigned to a cluster
v. Repeat above two steps until convergence criterion is satisfied.

3.3 Particle Swarm Optimization

PSO is a stochastic, population-based evolutionary algorithm for problem solving. The key idea of PSO method is to simulate the shared behavior happening among the birds flocks or fish school (Mohammed *et al*., 2009). It is computationally inexpensive due to its low memory and CPU requirements, it can easily be implemented (Eberhart*et al*., 1996) and has proven to be an efficient method for numerous general optimization problems. However, it does not suffer from problems like overfitting encountered by other evolutionary computation techniques including genetic algorithm characterized by overlapping and mutation calculation (Kennedy, 1995). As presented in Figure 2, the basic PSO algorithm consists of three steps, namely, generation of particles and their information, movements and new information vector (Olaleye, Olabiyisi, Olaniyan and Fagbola,2014).

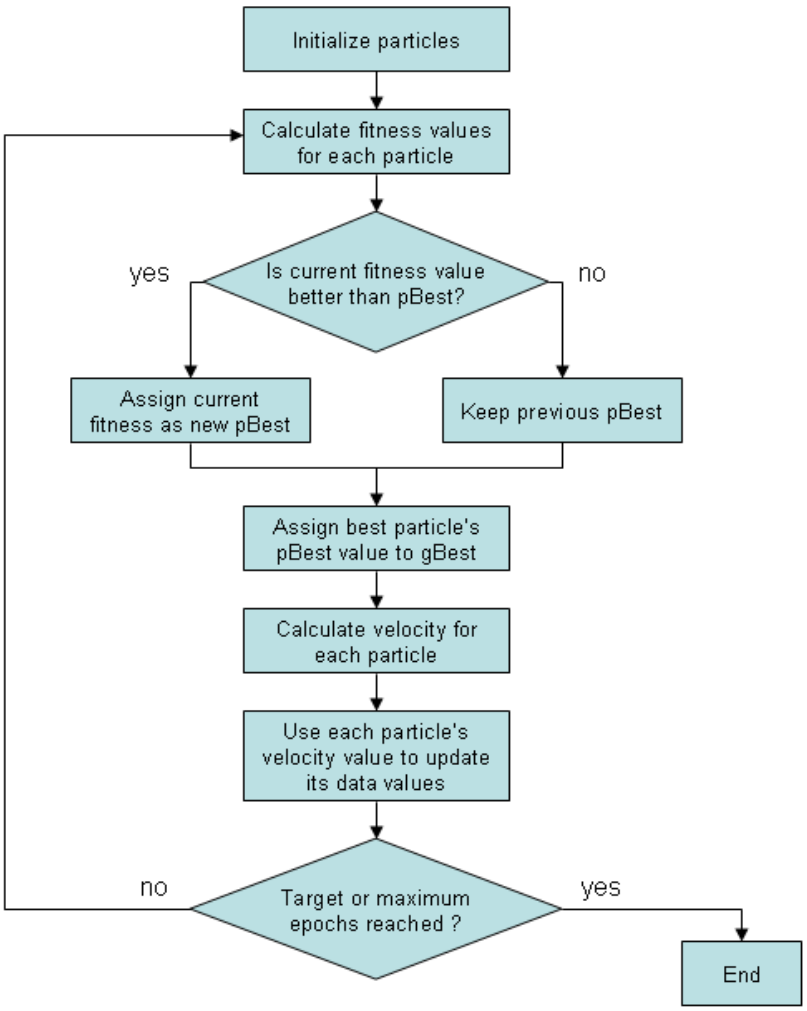


Figure 2: Flow Diagram Illustrating the Particle Swarm Optimization Algorithm (Olaleye *et. al.*, 2014)

The pseudo code of the PSO technique is as follows (Olaleye *et. al.*, 2014):

For each particle
Initialize particle
End

Do
For each particle
Calculate fitness value
If the fitness value is greater than the best fitness value (pBest) in history set current value as the new pBest
End
Choose the particle with the best fitness value of all the particles as the gBest

For each particle
Calculate particle velocity
Update particle position
End

While maximum iterations or minimum error criteria is not attained

3.4 Density-sensitive distance metric

Let data points be the nodes of graph $G = (V, E)$, and $p \in V^i$ be a path of length $i = |p|$ connecting the nodes p_1 and $p_{|p|}$ in which $(p_k, p_{k+1}) \in E$, $1 \leq k < |p|$. Let $P_{i,j}$ denote the set of all paths connecting nodes x_i and x_j. The density-sensitive distance metric between two points is defined to be (Ling, Liefeng and Licheng, 2012):

$$D_{ij} = \min_{p \in P_{i,j}} \sum_{k=1}^{|p|-1} L(P_k, P_{k+1}) \quad (1)$$

such that, D_{ij} satisfies the four conditions for a metric, that is, $Di_j = D_{ji} : D_{ij} \geq 0$; $D_{ij} \leq D_{ik} + D_kj$ for all x_i, x_j, x_k; and $D_{ij} = 0$ *iff* $x_i = x_j$.

With this, the density-sensitive distance metric can measure the geodesic distance along the manifold, which results in any two points in the same region of high density being connected by a lot of shorter edges while any two points in different regions of high density are connected by a longer edge through a region of low density. This achieves the aim of elongating the distance among data points in different regions of high density and simultaneously shortening that in the same region of high density (Ling, Liefeng and Licheng, 2012). Hence, this distance metric is data-dependent, and can help converge complex and unstructured data to global minima.

A density adjusted length of line segment is defined as (Ling *et al.*, 2012):

$$L(x_i, x_j) = \rho^{dist(x_i, x_j)} - 1 \quad (2)$$

Where
$dist(x_i, x_j)$ is the Euclidean distance between $x_i$and $x_j$while $\rho > 1$ is the flexing factor.

3.5 UCI Educational Process Mining (EPM) Real World Dataset

This dataset is one of the most widely used real world datasets in literatures and can be accessed and downloaded from *https://archive.ics.uci.edu.ml/datasets*. It is a learning analytics dataset collected in 2015 and contains the students' time series of activities during six sessions of laboratory sessions of the course of digital electronics. There are 6 folders containing the students' data per session. Each session folder contains up to 99 csv files each dedicated to a specific student log during that session. The number of files in each folder changes due to the number of students present in each session. However, each file contains 13 features:

Number of instances: 230318;
Number of attributes: 13;
Attribute characteristics: Integer

3.6 The developed Swarm-optimized K-means Algorithm

The development of the swarm-optimized k-means algorithm follows four (4) basic methodological stages as presented in Figure 3. These include the dataset acquisition, particle selection using PSO, design and development of a modified k-means with density-sensitive measure. To ascertain the performance of the developed swarm-optimized K-means algorithm, a performance evaluation was conducted.

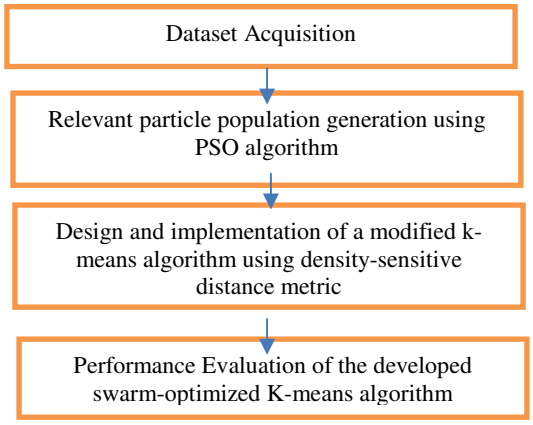


Figure 3: Methodological Structure of the Developed Swarm Optimized K-means Algorithm

3.6.1 Design of a Modified K-Means Algorithm Using Density-Sensitive Distance Metric

The stepwise description of the conventional K-means is presented in Section 3.2. Flowchart showing the integration of the PSO output into modified k-means algorithm is presented in Figure 4. Consequently, the improved K-means is presented as follows:

Input: m *data points* $\{x_i\}_{i=1}^{m}$ *obtained from PSO;* *Cluster number k, maximum iteration number* t_{max}, *stop threshold e.*

Output: Partition of the dataset $C_1,\ldots,C_k$.

i. *Randomly choose k data points using the K best position particles of PSO to initialize k cluster centers;*
ii. *For any two points* x_i, x_j, *compute the density-sensitive distance using equations (1 and 2);*
iii. *Assign each particle to the cluster which the density-sensitive distance of its center to the point is minimum;*
iv. *Recalculate the center of each cluster (cluster centroid) after all particles have been assigned;*
v. *Loop. If centroids no longer move or the number of iterations has reached the maximum number* t_{max}, *then stop. Otherwise, go to step ii.*

3.6.2 Performance Evaluation Metrics

The performance of the developed swarm-optimized K-means algorithm was evaluated using the following metrics:

i. *Clustering time*: This represents the time required to cluster all data points. This parameter depends on the platform where the clustering is implemented and will dictate if real-time functionality is available or not.
ii. *Clustering Accuracy*: This is the main measurement to describe the accuracy of a clustering system. It represents the number of particles that are correctly clustered from the total number of particles clustered (Jeremiah *et al.*, 2012).

$$\text{Clustering Accuracy} = \frac{\text{Number of correctly clustered particles}}{\text{Total number of particles}} \times 100\% \quad (3)$$

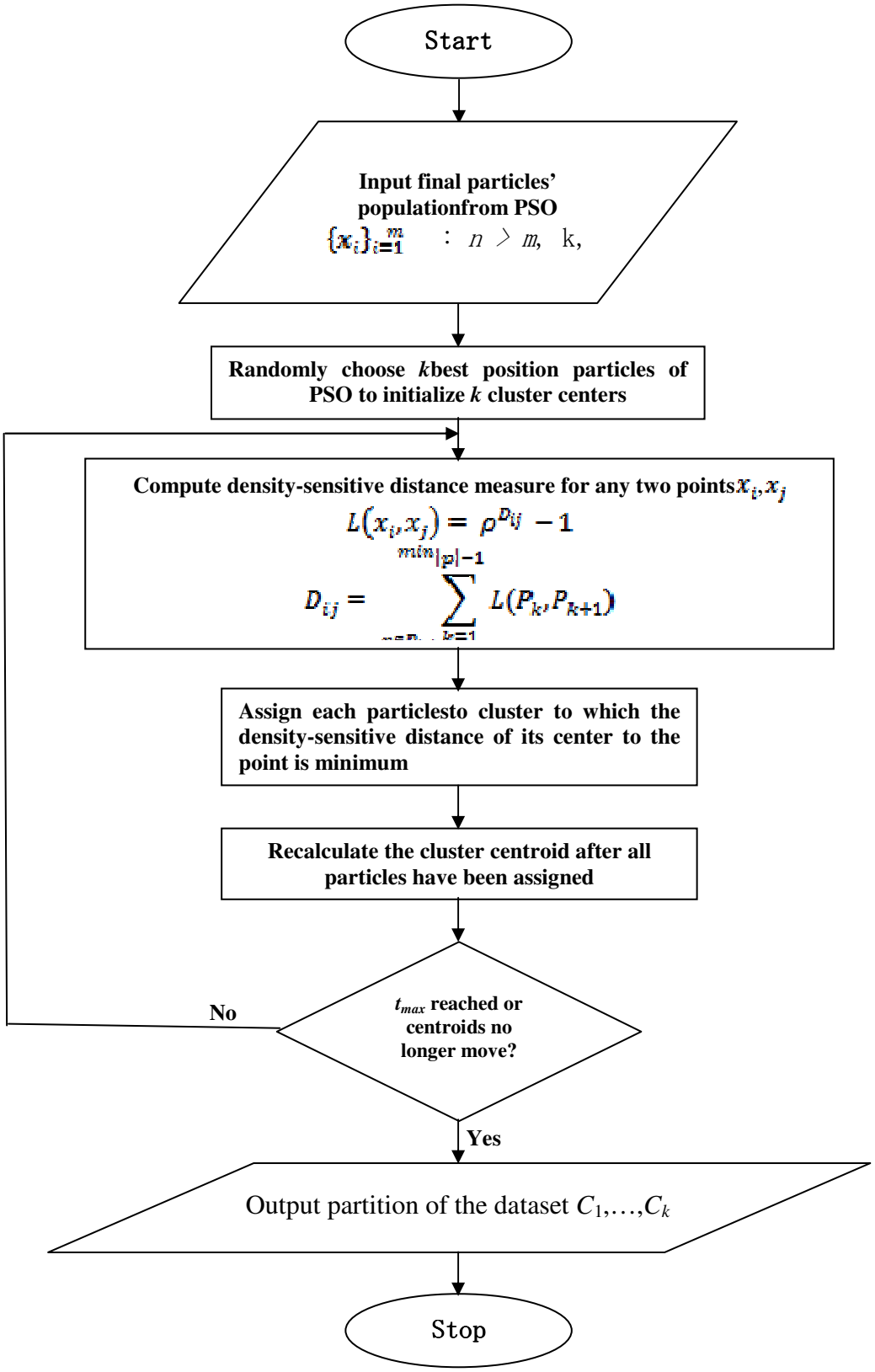

Start
Input final particles' populationfrom PSO
$\{x_i\}_{i=1}^{n}$: $n > m$, k,
Randomly choose kbest position particles of PSO to initialize k cluster centers
Compute density-sensitive distance measure for any two pointsx_i, x_j
$L(x_i, x_j) = \rho^{D_{ij}} - 1$
$D_{ij} = \min_{|p|-1} \sum_{k=1} L(P_k, P_{k+1})$
Assign each particlesto cluster to which the density-sensitive distance of its center to the point is minimum
Recalculate the cluster centroid after all particles have been assigned
t_{max} reached or centroids no longer move?
No
Yes
Output partition of the dataset $C_1,\ldots,C_k$
Stop

Figure 4: Flowchart showing the integration of the PSO output into modified k-means algorithm

4. RESULTS

In this work, a swarm-optimized K-means algorithm was developed and benchmarked with three (3) variants which are k-means, PCA-based HYBRID (K-PSO) and UFT-*k*-means. All the algorithms were implemented using MATLAB 7.7.0 (R2008b) on Windows 7 Ultimate 32-bit operating system, AMD Athlon (tm) X2 DualCore QL-66 central processing unit with a speed of 2.2GHZ, 2GB random access memory and 320GB hard disk drive. However, the results obtained by these clustering algorithms using Educational Process Mining (EPM) dataset is presented in Table 1. See the appendix for sample outputs of the developed swarm-optimized k-means algorithm.

4.1 Clustering Accuracy

As presented in Figure 5, the clustering accuracies produced by the original K-means, PCA-based HYBRID (K-PSO), UFT-*K*-means and the developed swarm-optimized K-means for 2 clusters (k = 2) are 64.8%, 72.1%, 77.7% and 80.2% respectively. For 3 clusters (k = 3), the accuracies obtained by the original K-means, PCA-based HYBRID (K-PSO), UFT-*K*-means and the developed swarm-optimized K-means are 67.3%, 76.4%, 79.1% and 83.6% respectively. When cluster number was increased to 4 (k = 4), the original K-means, PCA-based HYBRID (K-PSO), UFT-*K*-means and the developed swarm-optimized K-means yielded accuracies of 69.2%, 83.9%, 87.3% and 92.4% respectively.

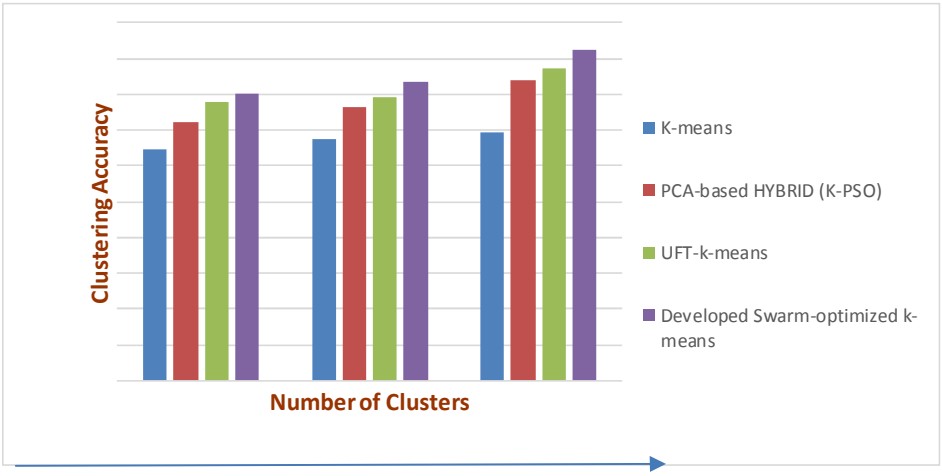


Figure 5: Accuracy of the Clustering Algorithms

Table 1: Evaluation results of the clustering algorithms using the EPM dataset

Number of Clusters	Algorithm	Clustering Accuracy (%)	Clustering Time (s)
2	K-means	64.8	83.2
2	PCA-based HYBRID (K-PSO)	72.1	99.4
2	UFT-*k*-means	77.7	87.2
2	Developed Swarm-Optimized K-means	80.2	85.7
3	K-means	67.3	78.7
3	PCA-based HYBRID (K-PSO)	76.4	91.8
3	UFT-*k*-means	79.1	84.7
3	Developed Swarm-Optimized K-means	83.6	80.5
4	K-means	69.2	74.4
4	PCA-based HYBRID (K-PSO)	83.9	87.2
4	UFT-*k*-means	87.3	80.4
4	Developed Swarm-Optimized K-means	92.4	74.8

4.2 Clustering Time

The execution times of the clustering algorithms obtained are presented in Figure 6. The traditional K-means, PCA-based HYBRID (K-PSO), UFT-*K*-means and the developed swarm-optimized K-means converged approximately in 83.2s, 99.4s, 87.2s and 85.7s respectively when the number of clusters was 2,78.7s, 91.8s, 84.7s and 80.5s respectively for 3 clusters and 74.4s, 87.2s, 80.4s and 74.8s respectively when the cluster number was 4.

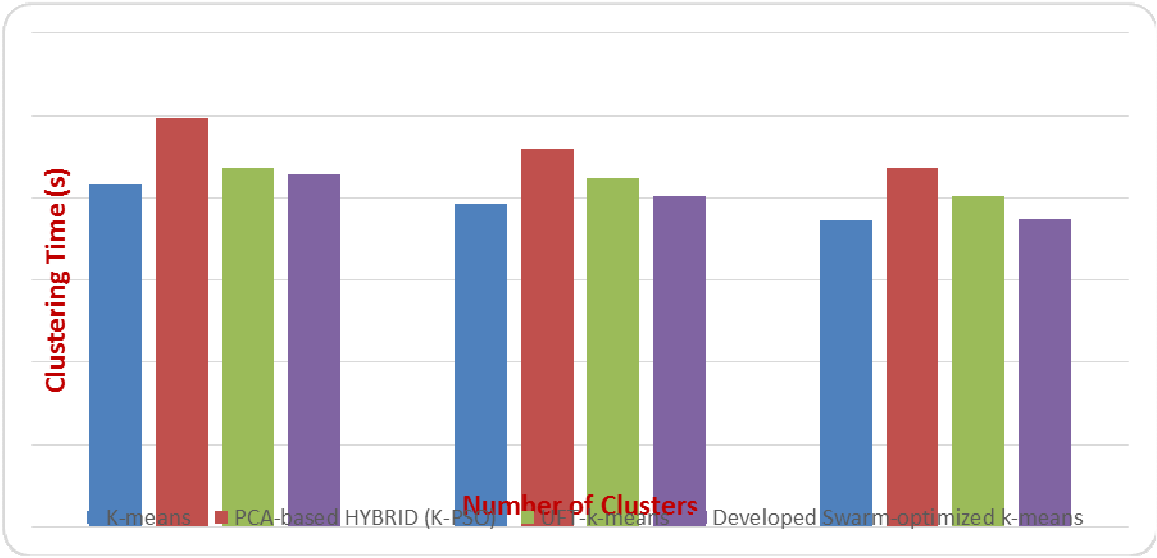


Figure 6: Execution time of the Clustering Algorithms

5. CONCLUSION AND FUTURE WORKS

This research presents a swarm-optimized K-Means clustering based on relevant particle selection using PSO and density-sensitive distance measure. The results obtained reveal that the developed swarm-optimized k-means algorithm has a more dominant performance over the conventional K-means, UFT-*k*-means and PCA-based HYBRID (K-PSO) clustering algorithms especially in terms of clustering accuracy. This challenging performance by the developed swarm-optimized K-means algorithm is due to the fact that relevant particle selection procedure as well as the globally converging density-sensitive distance measure were incorporated into the developed swarm-optimized K-means algorithm. Olaleye *et al.* (2014) and Fagbola *et al.* (2012) stated that improvements obtained for efficient and effective feature selection procedures invariably impact on the effectiveness of clustering algorithms which justifies the results obtained for the swarm-optimized K-means algorithm.

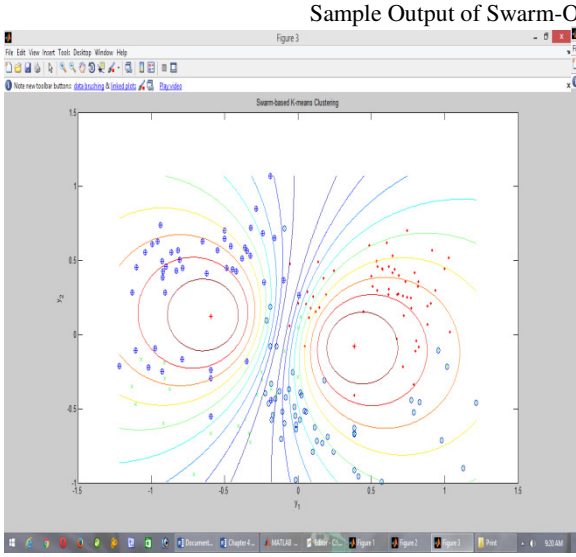
On the other hand, the least accuracies produced by K-means in all the evaluations corroborated with the assertion of Li *et al.* (2015) that K-means is not a good candidate for clustering large real world datasets. The developed swarm-optimized K-means can identify non-convex clustering structures, thus generalizing the application area of the conventional K-Means algorithm. The experimental results on EPM world dataset which contains 230318 instances validate the effectiveness of the developed algorithm. The developed swarm-optimized K-means algorithm can be applied in situations where the distributions of data points are not compact super-spheres. However, the near-optimal clustering time produced by the developed swarm-optimized K-means can be further investigated for possible improvements.

REFERENCES

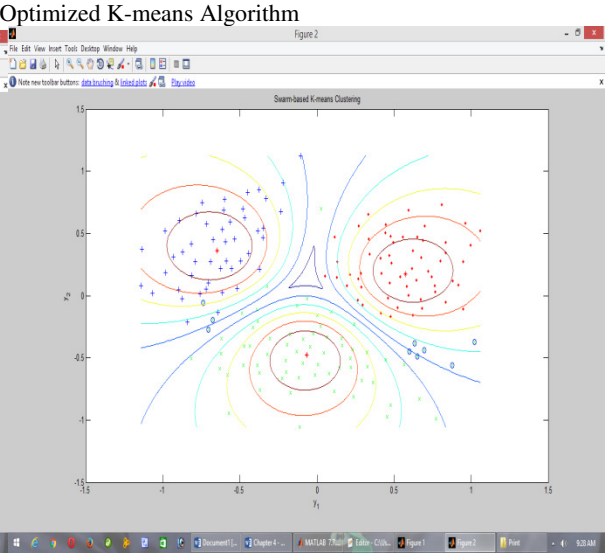
1. Adigun A.A, Omidiora E.O, Olabiyisi S.O, Adetunji A.B, Adedeji O.T (2012): "Development of a Hybrid K-means-expectation maximization clustering algorithm", Journal of computations & modeling, 2(4):55-65
2. Amita V. and Ashwani K., (2014): "Performance Enhancement of K-means Clustering Algorithms for High Dimensional Data sets", International Journal of Advanced Research in Computer Science and Software Engineering, 4(1), ISSN: 2277 128.
3. Azhar T., Arthur D. and S. Vassilvitskii (2012): "K-Means++: The advantages of careful seeding", Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms (DA '07), PA, USA, 1027-1035.
4. ChetnaSethi and Garima Mishra (2013): "A Linear PCA based hybrid K-Means PSO algorithm for clustering large dataset", International Journal of Scientific & Engineering Research, 4(6), 1559-1566.
5. Eberhart R.C., Simpson P. and Dobbins R. (1996): "Computational Intelligence PC Tools", A Book of Intelligent Systems, Academic Press.
6. FagbolaTemitayo, Olabiyisi Stephen and Adigun Abimbola (2012): "Hybrid GA-SVM for Efficient Feature Selection in E-mail Classification", Computer Engineering and Intelligent Systems, 3(3): 17-28.
7. GursharanSaini and HarpreetKaur (2014): "A Novel Approach towards K-Mean Clustering Algorithm with PSO", International Journal of Computer Science and Information Technologies, 5 (4), 5978-5986.
8. Jenn-Long, Yu-Tzu H. and Chih-Lung, G. (2012): Mining Student Behavior Models in Learning-by-Teaching Environments. In *Proceedings of the 1st International Conference on Educational Data Mining;* 127-136.
9. Jeremiah R. Barr, Kevin W. Bowyer, Patrick J. Flynn and Soma Biswas (2012): "Face Recognition from Video: A Review", International Journal of Pattern Recognition and Artificial Intelligence, World Scientific Publishing Company 16(2): 1-56.
10. Kennedy J. and Eberhart R.C. (1995): "Particle Swarm Optimization", Proceedings IEEE International Conference on Neural Networks, IV, p. 1942-1948.
11. Li Zheng, Lei Tao, Yue Yin and Jin Ding (2015): "A Framework for Hierarchical Ensemble Clustering"**,ACM Transactions on Knowledge Discovery from Data (ACM TKDD)**, **9**(2): 9-16.
12. Ling Wang, Liefeng Bo, Licheng Jiao (2012): "A Modified K-Means Clustering with a Density-Sensitive Distance Metric", Technical report, University of California, Department of Information and Computer Science, Ir-vine, CA.
13. Mary C. Immaculate and Raja Kasmir (2009): "A Modified Ant-based Clustering for Medical Data", International Journal on Computer Science and Engineering, 2(7), 2253-2257.
14. Min Wei, Tommy W. S. Chow and Rosa H. M. Chan (2015): "Clustering Heterogeneous Data with *k*-Means by Mutual Information-Based Unsupervised Feature Transformation", **Entropy**2015, **17**(3), 1535-1548; doi:10.3390/e17031535.
15. Ming-Chuan Hung, Jungpin Wu, Jin-Hua Chang and Don-Lin Yang (2005): "An Efficient k-Means Clustering Algorithm Using Simple Partitioning", Journal of Information Science and Engineering, 21, 1157-1177.
16. Mohammed Tiri, Pavlik, P., Cen, H., Wu, L. and Koedinger, K. (2009): Using Item-type Performance Covariance to Improve the Skill Model of an Existing Tutor. In Proceedings of the 1st International Conference on Educational Data Mining: 77-86.
17. Nasser S., Alkhaldi R. and Vert G. (2004): Semi-supervised learning literature survey, University of Wisconsin-Madison.
18. NeelamadhabPadhy, Pragnyaban Mishra and RasmitaPanigrahi (2012): The Survey of Data Mining Applications and Feature Scope.CoRR abs/1211.5723.
19. Nidhi Gupta and Ujjwal R. L. (2013), "An Efficient Incremental Clustering Algorithm" in World of Computer Science and Information Technology Journal (WCSIT), 3(5),97-99.
20. OlaleyeOludare, Olabiyisi Stephen, OlaniyanAyodele and FagbolaTemitayo (2014): "An Optimized Feature Selection Technique for Email Classification", International Journal of Scientific and Technology Research, 3(10): 286-293.
21. QiangNiu and Xinjian Huang (2011): "An improved fuzzy C-means clustering algorithm based on PSO", Journal of Software. 6(5), 873-879.
22. Rauber (2000): Educational Data Mining: A Survey from 1995 to 1999. *Expert Systems with Applications; 33*; 125-146.
23. Shanmugapriya B. and Punithavalli M. (2012): "A Modified Projected K-Means Clustering Algorithm with Effective Distance Measure", *International Journal of Computer Applications* 44(8):32-36.
24. ***Sharfuddin***Mahmood, ***Mohammad***SaiedurRahaman, **Dip** Nandi, ***Mashiour***Rahmann (2015): "A Proposed Modification of K-Means Algorithm", IJMECS, 7(6), 37-42.
25. Siedlecki W. and Sklansky J. (1988). "On Automatic Feature Selection", Int. J. Patt. Recog. Art. Intell.2(2): 197-220.
26. Su, M.C., Chou, C.H., 2001. A modified version of the K-means algorithm with a distance based on cluster symmetry.IEEE Trans. Pattern Anal.Machine Intel. 23 (6), 674–680
27. Zhang H. and Sun G. (2002): "Feature Selection Using Tabu Search Method," *Pattern Recognition*, 35(3): 701-711, 2006.

APPENDIX

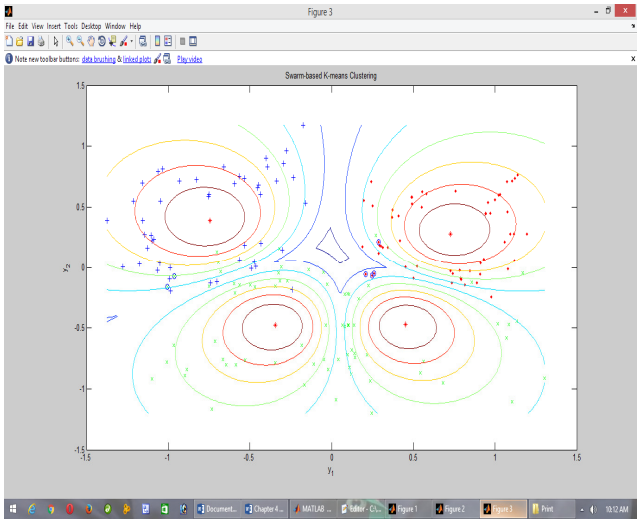
Sample Output of Swarm-Optimized K-means Algorithm



(k = 2)



(k = 3)



(k = 4)

Sample Final Clustered Output: PCA-based HYBRID (K-PSO) with EPM Dataset

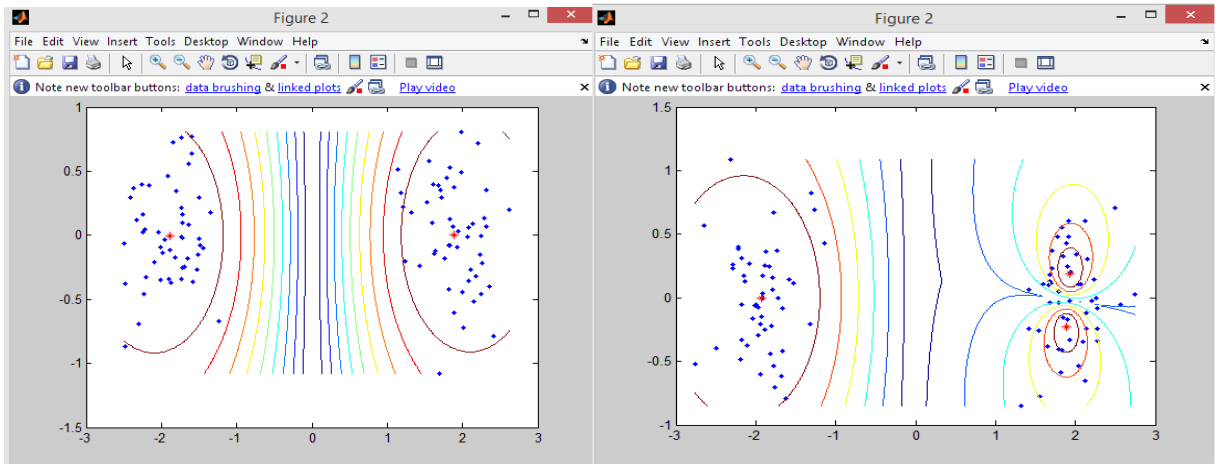


Figure 2
File Edit View Insert Tools Desktop Window Help
Note new toolbar buttons: data brushing & linked plots Play video
Figure 2
File Edit View Insert Tools Desktop Window Help
Note new toolbar buttons: data brushing & linked plots Play video

(K = 2)(K = 3)

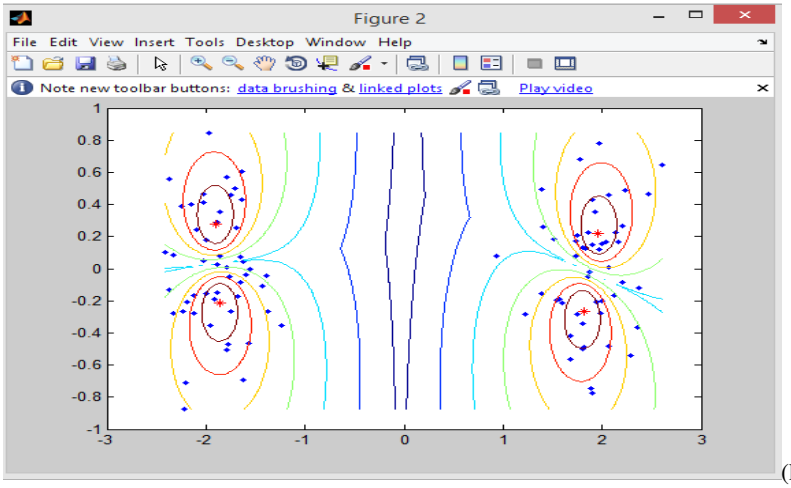


Figure 2
File Edit View Insert Tools Desktop Window Help
Note new toolbar buttons: data brushing & linked plots Play video

(k = 4)