# Harmony Search Optimization in K-Means Clustering

Samina Ahamed K T[#1], Jasila E K[*2]

[#1] MTech student(CSE),MES College of Engineering,Kuttippuram,India
*2 Assistant Professor (CSE),MES College of Engineering,Kuttippuram,India

***Abstract*— Clustering is a data mining technique that classifies a set of observations into clusters based on some similarity measures. The most commonly used partitioning based clustering algorithm is K-means. However, the K-means algorithm has several drawbacks. The algorithm generates a local optimal solution based on the randomly chosen initial centroids. Harmony Search is a recently developed meta-heuristic optimization algorithm which helps to find out near global optimal solutions by searching the entire solution space. The hybrid algorithm that combines harmony search and K-means produce a better solution.**

***Keywords*— Data Mining, Clustering, k-means, Harmony Search Optimization.**

## I. INTRODUCTION

Clustering is one of powerful data mining techniques that can discover intentional structures in data. It groups instances which have similar features and builds a concept hierarchy. As a result, it often extracts new knowledge from a database. Because of the ability to find the intentional descriptions, it is one of the very important techniques in managing databases.

### *A. K-Means Clustering*

The most widely used partitional clustering algorithm is K-means [1]. K-means algorithm clusters the input data into k clusters, where k is given as an input parameter. K-means algorithm finds a partition in such a way that the squared error between the centroid of a cluster and its data points is minimized. The algorithm first takes k random data points as initial centroids and assigns each data point to the nearest cluster until convergence criteria is met. Although K-means algorithm is simple and easy to implement, it suffers from some drawbacks:

1) The number of clusters, k has to be specified as input.
2) The algorithm converges to local optima. The final clusters depend on randomly chosen initial centroids.

To deal with the limitations of K-means, several clustering algorithms have been developed so far. Optimization techniques are used to solve clustering problem. Clustering can be viewed as an optimization problem that tries to maximize the intra-cluster similarity.

### *B. Harmony Search Optimization*

Harmony search (HS) [2][3]is a meta-heuristic algorithm that was conceptualized using the musical process of searching for a perfect state of harmony. The harmony in music is analogous to the optimization solution vector, and the musician's improvisations are analogous to local and global search schemes in optimization techniques. The algorithm searches the entire solution area to find a solution that optimizes the objective function. In each iteration, a new solution is improvised from the existing solutions in harmony memory. If the new solution has better fitness value than the worst solution in the memory, it is replaced. The improvisation process is repeated for several times and the best solution in harmony memory is selected as the final solution.

## II. LITERATURE SURVEY

In this Literature survey several methods to improve k means clustering using harmony search optimization are studied. Harmony search algorithm had been very successful in a wide variety of optimization problems, presenting several advantages with respect to traditional optimization techniques such as the following [2]:

1) HS algorithm imposes fewer mathematical requirements and does not require initial value settings of the decision variables.
2) As the HS algorithm uses stochastic random searches, derivative information is also unnecessary.
3) The HS algorithm generates a new vector, after considering all of the existing vectors.

These features increase the flexibility of the HS algorithm and produce better solutions. The steps in the procedure of harmony search are as follows [2]:

Step 1: Initialize the problem and algorithm parameters.
Step 2: Initialize the harmony memory.
Step 3: Improvise a new harmony.
Step 4: Update the harmony memory.
Step 5: Check the stopping criterion.

First the HS algorithm parameters are specified, these are the harmony memory size (HMS), or the number of solution vectors in the harmony memory, harmony memory considering rate (HMCR), pitch adjusting rate (PAR) and the number of improvisations (NI), or stopping criterion. The

harmony memory (HM) is a memory location where all the solution vectors (sets of decision variables) are stored. Here, HMCR and PAR are parameters that are used to improve the solution vector. In the initialization of HM, the HM matrix is filled with as many randomly generated solution vectors as the HMS. Generating a new harmony is called 'improvisation' [2]. Every component obtained by the memory consideration is examined to determine whether it should be pitch adjusted. This operation uses the PAR parameter, which is the rate of pitch adjustment.

To improve the performance of the HS algorithm and eliminate the drawbacks lie with fixed values of HMCR and PAR, M. Mahdavi et.al [4] proposed an improved harmony search (IHS) algorithm that uses variable PAR. M. Mahdavi et.al [5] modeled the clustering problem as an optimization problem that locates the optimal centroids of the clusters rather than to find an optimal partition and proposed a document clustering algorithm (HClust) based on HS algorithm. They considered each cluster centroid as a decision variable, so each row of harmony memory, which contains k decision variables, represents one possible solution for clustering. Fitness value of each row of HM is determined by average distance of documents to the cluster centroid (ADDC) represented by each solution.

The HClust algorithm performs a globalize searching for solutions, whereas K-means clustering procedure performs a localized searching. HClust is good at finding promising areas of the search space, but not as good as K-means at fine-tuning within those areas. For this reason, a hybrid clustering approach that uses K-means algorithm to replace the refining stage in the HClust algorithm is introduced [5]. Hybrid algorithm combines the power of the HClust with the speed of a K-means algorithm. The HClust finds the region of the optimum, and then the K-means takes over to find the optimum centroid. Two different versions of the hybrid clustering, depending on the stage when the K-means algorithm carries out are proposed in [5].

R.Forsati et.al [6] proposed hybridization of harmony search and K-means three ways. In Sequential Hybridization model, Harmony memory search algorithm is run first and best solution is taken as initial solution for K-means. Then K-means algorithm is executed maximum number of iterations and then the result is taken as final clustered solution. Another hybridization model is Interleaved Hybridization, where initial configuration of k-means is the output of a harmony search algorithm and the result of k-means is again compared with solutions in the harmony memory and replace the worst case solution. This procedure is continued until the fitness value reaches certain threshold value. One step hybridization model is widely used model for many hybrid algorithms in which the generated solution from harmony memory is used as the input to k means algorithm. After refining the solution to local optima, it is compared with solutions in the Harmony search memory clustering with the speed of K-means. These models are less dependent on initial parameters and find the global solution rather than local.

R.Forsati et.al [7] use the sequential hybridization model for web clustering and M.Mahdavi et.al [8] use one step hybridization model for document clustering to improve the accuracy of clusters. B.Amiri et.al [9] also use the sequential hybridization model to get more accuracy in clustering of real and artificial datasets and proved that it is better than using other optimization algorithms.

S.V.Sankaran et.al [10] proposed an improved harmony search algorithm combined with K-means algorithm. Here one step hybridization model is used. This algorithm improves the solutions in harmony memory and also uses different PAR values for improvisation. It improves the ordinary harmony search algorithm by modifying the improvisation step. At first, the harmony memory is initialized with randomly generated feasible solutions. Then new solutions are improvised from the existing solutions. Two improvisation strategies are used. In the first improvisation, a new solution is improvised from the solution set S, in which each solution is refined using a different PAR value which is inversely proportional to the fitness value of the solution.

$PAR=PAR_{min} +\{(PAR_{max} –PAR_{min})*i/k\}$

where $PAR_{min}$ is the minimum PAR, $PAR_{max}$ is the maximum PAR and i is the iteration number. The other is to improvise a new solution from the harmony memory such that the cluster number of each data item is selected from the harmony memory with probability HMCR(Harmony Memory Considering Rate) and randomly selected from the set {1, 2,.....k } with probability(1-HMCR). After selecting a cluster, pitch adjustment process is applied.

L.P.Chandran et.al [11] proposed an improved clustering algorithm which contains 2 phases. Algorithm 1 makes use of the hybrid Harmony search and K-means algorithm proposed in [10]. The improvised solution is used for calculating the initial centroids. This is given to Phase 2 which assigns the data points to appropriate clusters using Algorithm 2 proposed in [12]. In Algorithm 2, initially the distance between all data points and the initial centroids are computed and the data points are assigned to the nearest clusters. For each data point, the cluster number and its distance from the centroid of that cluster is recorded. Then the cluster centroids are recalculated. The next stage is an iterative process in which distance between each data point and its present nearest cluster centroid is computed. If this distance is less than or equal to the previous nearest distance, the data point stays in that cluster itself. Otherwise its distance from all other centroids are computed and the data point is assigned to the cluster with the nearest centroid. This process is repeated until the convergence criterion is met.

## III. Problem Definition

K-means algorithm is the most commonly used partitioning based clustering algorithm. But it generates a local optimal solution. The final clusters depend on the initial centroids which are randomly chosen. The existing improved harmony search based K-means algorithm uses harmony search optimization to find a global optimal solution and thereby improves the accuracy of the result. This algorithm

has high execution time compared to K-means algorithm. The problem identified here is to find out an algorithm for K-means clustering using harmony search optimization with less computational time and more accuracy than the existing work.

## IV. PROPOSED METHOD

The proposed method is named as Global Best Harmony Search Based K-Means Clustering (GBHS K-means) and it uses an important improvement in Harmony Search, Global Best Harmony Search (GHS) introduced by Omran et.al [13].GHS modifies the pitch-adjustment step of the HS such that the new harmony can mimic the best harmony in the HM. Thus, replacing the bw parameter altogether and adding a social dimension to the HS. Intuitively, this modification allows the GHS to work efficiently on both continuous and discrete problems. GHS modifies the pitch-adjustment step of the HS such that a new harmony is affected by the best harmony in the harmony memory. This modification alleviates the problem of tuning the bw parameter of HS which is difficult to specify a priori. The proposed method GBHS K-Means works as follows:

The GBHS K - Means Clustering Algorithm:

Input:
D = {d1, d2, ….. dn} // set of n data items.
K // Number of desired clusters.
HMS // Harmony memory size
HMCR // Harmony Memory Considering Rate
PARmin // Minimum Pitch Adjusting Rate
PARmax // Maximum Pitch Adjusting Rate
MI // Maximum Number of Improvisation

Output:
A set of K clusters.

Steps:
1. Determine the initial centroids of the clusters by using Algorithm 1.
2. Assign the data points to the clusters by using Algorithm 2.

Algorithm 1 : The GBHS KMeans Clustering Algorithm Phase1

Input:
D = {d1, d2, ….. dn} // set of n data items.
K // Number of desired clusters.
HMS // Harmony memory size
HMCR // Harmony Memory Considering Rate
PARmin // Minimum Pitch Adjusting Rate
PARmax // Maximum Pitch Adjusting Rate
MI // Maximum Number of Improvisation

Output:
A set of K initial centroids

Steps:
1. Initialize the Harmony memory with HMS random solutions.
2. Evaluate the fitness of all solutions in Harmony memory.
3. Improvise new Harmony (NHM) as follows:

```
for each iε[1,N] do
    if U(0, 1) _ HMCR then
    begin
        x'_i = x^j_i where j ~ U (1,…, HMS).
        if U(0, 1)≤ PAR(t) then
        begin
            x'_i = x^best_k where best is
            the index of the best harmony in
            the HM
            end if
    else random selection
    end if
end
```

4. Evaluate the fitness of all new solutions in NHM
5. Update Harmony memory with new solutions until maximum number of improvisation is reached.
6. Select a new solution from the harmony memory with best fitness.
7. Calculate the cluster Centroids for the new solution.
8. Done.

GHS is inspired by the concept of swarm intelligence. Particle Swarm Optimization (PSO) [14] is a population-based stochastic optimization method. It is motivated by social behavior of organisms such as bird flocking and fish schooling. In the PSO algorithm, the potential solutions called particles are flown in the problem hyperspace. Change of position of a particle is called velocity. The particle changes their position with time. During flight, particles velocity is stochastically accelerated toward its previous best position and toward a neighborhood best solution. POS has been successfully applied to solve various optimization problems, artificial neural network training, fuzzy system control and others. In a global best PSO system, a swarm of individuals (called particles) fly through the search space. Each particle represents a candidate solution to the optimization problem. The position of a particle is influenced by the best position visited by itself (i.e. its own experience) and the position of the best particle in the swarm (i.e. the experience of swarm).

For assigning the data points to the clusters this method make use of second phase of the enhanced method in [11].

Algorithm 2 : The GBHS KMeans Clustering Algorithm Phase2

Input:
D = {d1, d2,….,dn} // set of n data items.

C = {c1; c2,…., ck} // set of K centroids.

Output:

A set of K clusters.

Steps:
1. Compute the distance of each data point to all the centroids.
2. For each data point $d_i$, find the closest centroid $c_j$ and assign $d_i$ to cluster j;
3. Set ClusterId[i] = j; // j:Id of the closest cluster;
4. Set NearestDist[i] = $d(d_i, c_j)$;
5. For each cluster recalculate the centroids
6. Repeat
7. For each data point $d_i$
7.1 Compute its distance from the centroid of the present nearest cluster;
7.2 If this distance is less than or equal to the present nearest distance, the data point stays in the cluster;
7.3 Else
7.3.1 For every centroid Compute the distance $d(d_i, c_j)$;
7.3.2 Assign the data point $d_i$ to the cluster with the nearest centroid $c_j$ ;
7.3.3 Set ClusterId(i) = j;
7.3.4 Set NearestDist[i] = $d(d_i, c_j)$;
8. End for
9. For each cluster recalculate the centroids;
10. Until the clusters do not change.

In the second phase, initially the distance between all data points and the initial centroids are computed. Then the data points are assigned to the nearest clusters. For each data point, the cluster number and its distance from the centroid of that cluster is recorded. The cluster centroids are recalculated. The next stage is an iterative process in which distance between each data point and its present nearest cluster centroid is computed. If this distance is less than or equal to the previous nearest distance, the data point stays in that cluster itself. Otherwise its distance from all other centroids are computed and the data point is assigned to the cluster with the nearest centroid. This process is repeated until the convergence criteria is met.

## IV. Implementation Details

Implementation was done in Java under Windows platform for multidimensional datasets. The original K-Means clustering algorithm was implemented first. The Improved Harmony Search based K-means algorithms presented in [10] which is denoted as IHSK-1 and the Improved Harmony Search based K-means algorithms presented in [11] which is denoted as IHSK-2 were also implemented. Proposed method was implemented and compared with above three existing methods. Datasets used for experiment are shown in Table I.

TABLE I
Datasets Used For Testing

| Datasets | Number of Classes | Number of Data Points | Number of Attributes |
|---|---|---|---|
| Iris | 3 | 150 | 4 |
| New Thyroid | 3 | 215 | 5 |
| E-coli | 8 | 336 | 7 |

## VI. Experimental Results

The accuracy of the clusters is measured using Cluster Purity Metric [11] and the results are shown in Table II.Purity of a cluster is a measure of correctly classified data points. Let (|Cj|class=i) denote number of items of class i assigned to cluster j. Purity of this cluster is given by

$$Purity(Cj) = \max(|Cj|_{class=i}) \ / |Cj|$$

The overall purity of a clustering solution could be expressed as a weighted sum of individual cluster purities.

$$Purity = \sum_{j=1}^{k} |Cj|/n(Purity(Cj))$$

where n is the total number of data points in the dataset.Fig.1shows the graphical representation of comparison of these methods.

TABLE III
Comparison of Accuracy

| Data set | K-means | IHSK[10] | IHSK[11] | Proposed Method |
|---|---|---|---|---|
| IRIS | 84.4 | 89.2 | 90.7 | 91.6 |
| New Thyroid | 77.7 | 79.3 | 79.54 | 80.7 |
| E-coli | 77.6 | 79.9 | 80.13 | 81.5 |

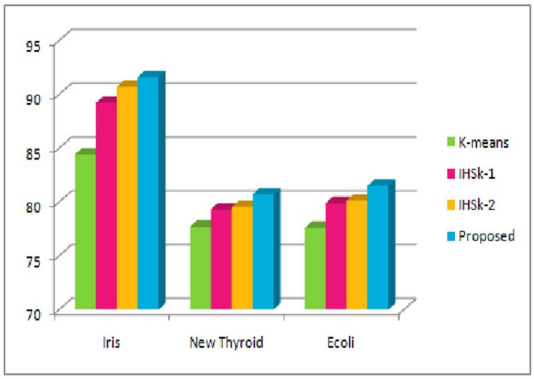

Fig.1 Comparison of Accuracy

## VII. Conclusions

The most commonly used partitioning based clustering algorithm, K-means algorithm suffers from some major drawbacks. It depends on initialization and finds local optimal solution. Other major drawback is high computational complexity. To deal with the limitations that exist in K-means, recently, new concepts and techniques have been entered. The k-means algorithm using harmony search optimization finds a global optimal solution. In this paper different methods in this area are analyzed. Improved Harmony Search Optimization based K-means clustering improves the accuracy of the result but it has high execution time compared to K-means

algorithm. The future work of this paper will include modified method that can improve the accuracy and complexity.

## ACKNOWLEDGMENT

We take this opportunity to convey our deep and sincere thanks to the Principal Prof. V.H .Abdul Salam and Head of the Department Dr. P. M. S. Nambisan of MES College of Engineering, Kuttippuram, India. We express our sincere gratitude to all the staff members of Computer Science and Engineering Department and beloved family members who helped us with their timely suggestions and support. We also express our sincere thanks to all friends who helped in all the conditions. All glory and honor be to the Almighty God, who showered His abundant grace on me to make this work a success.

## REFERENCES

[1] MacQueen and James, "Some methods for classification and analysis of multivariate observations", in *Proc.the fifth Berkeley symposium on mathematical statistics an d probability*, vol. 1, no. 281-297. California, USA,1967, p. 14.

[2] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice", *Computer methods in applied mechanics and engineering*, vol. 194, no. 36, pp. 3902- 3933, 2005.

[3] Z. W. Geem, "Music-inspired harmony search algorithm: theory and applications",*Springer*, 2009, vol. 191.

[4] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems", *Applied Mathematics and Computation*, vol. 188, no. 2, pp. 1567- 1579, 2007.

[5] M. Mahdavi, M. H. Chehreghani, H. Abolhassani, and R. Forsati, "Novel meta-heuristic algorithms for clustering web documents", *Applied Mathematics and Computation*, vol. 201, no. 1, pp. 441- 451, 2008.

[6] R. Forsati, M. Meybodi, M. Mahdavi, and A. Neiat, "Hybridization of k-means and harmony search methods for web page clustering", in *Proc.Web Intelligence and Intelligent Agent Technology, WI-IAT'08*. IEEE/WIC/ACM International Conference on, vol. 1. IEEE, 2008, pp. 329-335.

[7] R. Forsati, M. Mahdavi, M. Kangavari, and B. Safarkhani, "Web page clustering using harmony search optimization", in *Proc.Electrical and Computer Engineering, CCECE 2008*. Canadian Conference on. IEEE, pp. 001 601- 001 604.

[8] M. Mahdavi and H. Abolhassani, "Harmony k-means algorithm for document clustering", *Data Mining and Knowledge Discovery*, vol. 18, no. 3, pp. 370-391, 2009

[9] B. Amiri, L. Hossain, and S. E. Mosavi, "Application of harmony search algorithm on clustering", in *Proc. World Congress on Engineering and Computer Science*, vol. 1, 2010, pp. 20-22.

[10] S. V. Sankaran and K. A. Nazeer, "Improving harmony search based k-means clustering algorithm", in MTech Thesis, Dept of CSE, NIT Calicut,India, 2009.

[11] L. P. Chandran and K. A. Nazeer, "An improved clustering algorithm based on k-means and harmony search optimization", in *Proc. Recent Advances in Intelligent Computational Systems (RAICS)*, 2011 IEEE,pp. 447- 450.

[12] K. A. Nazeer and M. Sebastian, "Improving the accuracy and efficiency of the k-means clustering algorithm", in *Proc. the world congress on Engineering*, vol. 1, 2009, pp. 1-3.

[13]M. G. Omran and M. Mahdavi, "Global-best harmony search," *Applied Mathematics and Computation*, vol. 198, no. 2, pp. 643 - 656, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0096300307009320

[14] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Micro Machine and Human Science. MHS'95., Proc. the Sixth International Symposium on. IEEE, 1995, pp. 39-43.*

.