# Accelerating Unique Strategy for Centroid Priming in K-Means Clustering

**Ms. S. Saranya**
*Assistant Professor*
*Department of Computer Science*
*Hindusthan College of Arts & Science, Coimbatore, India*

**Ms. P. Deepika**
*Assistant Professor*
*Department of Computer Science*
*Hindusthan College of Arts & Science, Coimbatore, India*

**Ms. S. Sasikala**
*Assistant Professor*
*Department of Computer Science*
*Hindusthan College of Arts & Science, Coimbatore, India*

**Dr. S. Jansi**
*Assistant Professor*
*Department of Computer Science*
*Hindusthan College of Arts & Science, Coimbatore, India*

**Ms. A. Kiruthika**
*Assistant Professor*
*Department of Computer Science*
*Hindusthan College of Arts & Science, Coimbatore, India*

## Abstract

Clustering is the process of organizing data objects into a set of disjoint classes called clusters. The fundamental data clustering problem may be defined as discovering groups in data or grouping similar objects together. Some of the problems associated with current clustering algorithms are that they do not address all the requirements adequately, and need large number of iterations when dealing with a large number of dimensions. K-Means is one of the algorithms that solve the well-known clustering problem. This algorithm classifies object to a predefined number of clusters, which is given by the user. The idea is to choose random cluster centers, one for each other. The centroid initialization plays an important role in determining the cluster assignment in effective ways. But the performance of K-Means clustering is affected when the dataset used is of high dimension and the accuracy and sum square error is highly affected because of the high dimension data. This paper, proposed a new algorithm of data partitioning based k-means for performing data partitioning along the data axis with the highest variance. This will shows more effective and efficient converge to better clustering results, reduce the number of iterations required clustering also help to reduce the sum square error for all cells than the existing clustering.

**Keywords: Data clustering, k-means algorithm, Data partitioning**

---

## I. INTRODUCTION

Data Mining involves the process of extracting interesting and hidden patterns or characteristics from very huge datasets and using it in decision making and prediction of future behaviour. This improves the need for effective and efficient analysis methods to make use of this information. One of these tasks is clustering.

Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. A loose definition of clustering could be "the process of organizing objects into groups whose Members similar in some way". A cluster is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters. Clustering is the process of partitioning or combination a given set of patterns into displaces clusters. The goal of the clustering is to group data in to cluster such that similarities among data members within the same cluster are maximal while similarities among data members from different are minimal. Many clustering algorithms have been developed. Clustering is categorized into partition method, hierarchical method, density based method, grid based method, and model based methods.

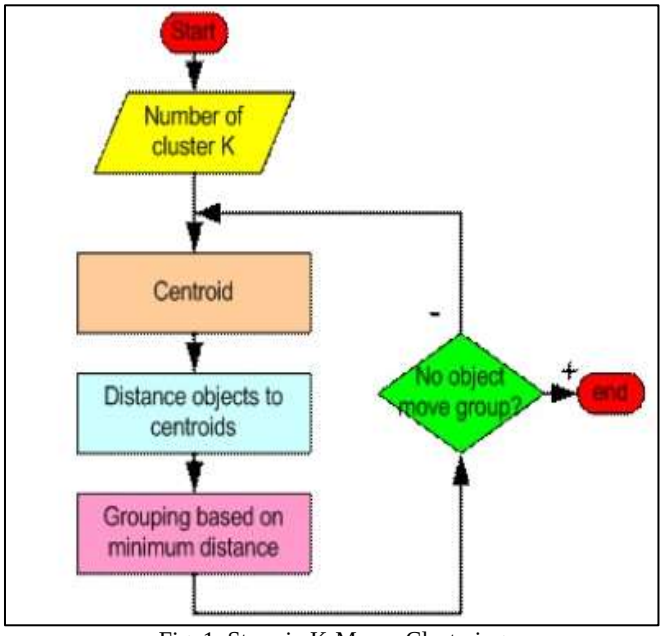

Fig. 1: Steps in K-Means Clustering

Figure 1 show the steps involved in the simple k-means clustering algorithm. The basic Clustering setup is

### *A. Pre-processing and Feature Selection*

Involves choosing an appropriate feature, and doing appropriate pre-processing and feature extraction on data items to measure the values of the chosen feature set. It will often be desirable to choose a subset of all the features available, to reduce the dimensionality of the problem space. This step often requires a good deal of domain knowledge and data analysis.

### *B. Similarity Measure*

Plays an important role in the process of clustering where a set of objects are grouped into several clusters, so that similar objects will be in the same cluster and dissimilar ones in different cluster [2].

### *C. Clustering Algorithm*

Which use particular similarity measures as subroutines. The particular choice of clustering algorithms depends on the desired properties of the final clustering. A clustering algorithm attempts to find natural groups of components (or data) based on some similarity and also finds the centroid of a group of data sets. To determine cluster membership, most algorithms evaluate the distance between a point and the cluster centroids. The output from a clustering algorithm is basically a statistical description of the cluster centroids with the number of components in each cluster.

### *D. Result Validation*

Iterate back to some prior stage. It may also be useful to do a test of clustering tendency, to try to guess if clusters are present at all; note that any clustering algorithm will produce some clusters regardless of whether or not natural clusters exist.

### *E. Result Interpretation and Application*

Typical applications of clustering include data compression (via representing data samples by their cluster representative), hypothesis generation (looking for patterns in the clustering of data), hypothesis testing (e.g. verifying feature correlation or other data properties through a high degree of cluster formation), and prediction. Among the various clustering algorithms, K-Means (KM) is one of the most popular methods used in data analysis due to its good computational performance. K-means clustering is a method of classifying/grouping items into k groups (where k is the number of pre-chosen groups). The grouping is done by minimizing the sum of squared distances (Euclidean distances) between items and the corresponding centroid [3]. In K-Means choosing the proper initial centroids is the key step of the basic K-means procedure. It is easy and efficient to choose initial centroids randomly, but the results are often poor [4].

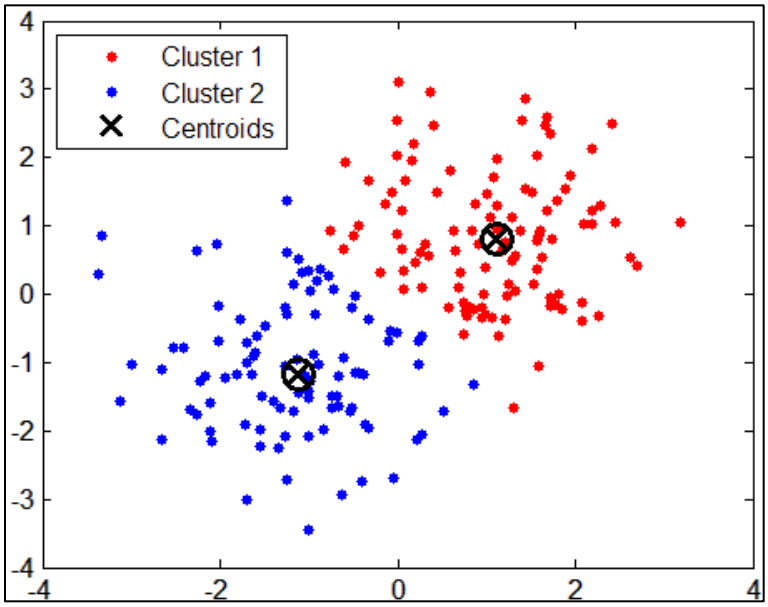

Fig. 2: Centroid Priming in K-Means Clustering

Figure 2 shows the simple centroid priming in a K-Means clustering. Hence the Modified Centroid selection method is introduced Instead of updating the centroid of a cluster after all points have been assigned to clusters, the centroids can be updated as each point is assigned to a cluster. In addition, the relative weight of the point being added may be adjusted. The goal of these modifications is to achieve better accuracy and faster convergence.

## II. RELATED WORK

Likas, N. Vlassis and J.J. Verbeek,(2003)[1] proposed the global k-means algorithm has presented which is an incremental approach to clustering that dynamically adds one cluster center at a time through a deterministic global search procedure consisting of N (with N being the size of the data set) executions of the k-means algorithm from suitable initial positions. The propose method will reduce the computational load without significantly affecting solution quality. The proposed clustering methods are tested on well-known data sets and they compare favourably to the k-means algorithm with random restarts.

Fahim A.M, Salem A. M, Torkey A and Ramadan M. A, "An Efficient enhanced k-means clustering algorithm" has proposed idea that makes k-means more efficient since, in each iteration, the k-means algorithm computes the distances between data point and all centers, this is computationally very expensive especially for huge datasets. For each data point, it keep the distance to the nearest cluster. At the next iteration, we compute the distance to the previous nearest cluster. If the new distance is less than or equal to the previous distance, the point stays in its cluster, and there is no need to compute its distances to the other cluster centers. This saves the time required to compute distances to k−1 cluster center.

In this paper, new algorithm called an efficient k-means clustering based on influence factors, which is divided into two stages and can automatically achieve the actual value of k and select the right initial points based on the datasets characters. Propose influence factor to measure similarity of two clusters, using it to determine whether the two clusters should be merged into one. In order to obtain a faster algorithms theorem is proposed and proofed, using it to accelerate the algorithm

## III. EXISTING APPROACH

### *A. The k-means clustering algorithm*

The segment describes the original kmeans clustering algorithm. The k-means cluster is the method of cluster analysis which aims to partition n observation in to k centroid in which each observation belongs to the centroid with the nearest mean. Euclidean distance is generally considered to determine the distance between data points and the centroid once we find k new centroids a new binding is to created between the same Data points and the nearest new centroid as a results, the k- centroid may change their position in a step by step manner. This process will continue until convergence criteria for clustering.

*1) Algorthim1: The Standard K-means clustering algorithm*

Input
D= {d1, d2, d3……dn} // set of elements
K // number of desired cluster
Output:
K // set of clusters
Steps:

Assign initial centroid for means in k-data items
Repeat
Assign each item di to the cluster which has the closes
Calculate new means for each cluster
Until convergences criteria is met;

### *B. The cluster used Calculating the initial Centroid Algorithm k-means*

*2)Algorithm 2: Modified k-means Algorithm*
Input
D= {d1, d2, d3……dn} // set of elements
K // number of desired cluster
Output:
K // set of clusters
Steps:
Phase 1: Determine the initial centroids of the clusters by using algorithm 3
Phase 2: Assign each data point to the appropriate clusters by algorithm 4

In the first phase, the initial centroids are determined systematically so as to produce clusters with better accuracy. The second phase data points are assigned to appropriate clusters. It starts by forming the initial clusters based on the relative distance of each data-point from the initial centroids. These clusters are subsequently fine-tuned by using a heuristic approach, thereby improving the efficiency. The two phases of the enhanced method are described below as Algorithm 3 and Algorithm 4.

*3)Algorithm 3: Finding the initial centroids*
*Input*
D={ d1,d2,d3……dn} // set of elements
K // number of desired cluster
Output: A set of *k* initial centroids
*Steps:*

1) Set m = 1;
2) Compute the distance between each data point and all other data- points in the set D;
3) Find the closest pair of data points from the set D and form a data-point set Am (1<= m <= k) which contains these two data- points, Delete these two data points from the set D;
4) Find the data point in D that is closest to the data point set Am, add it to Am and delete it from D;
5) Repeat step 4 until the number of data points in Am reaches 0.75*(n/k);
6) If m<k, then m = m+1, find another pair of data points from D between which the distance is the shortest, form another data-point set Am and delete them from D, Go to Step 4;
7) for each data-point set Am (1<=m<=k) find the arithmetic mean of the vectors of data points in Am, these means will be the initial centroids.

Algorithm 3 describes the method for finding initial centroids of the clusters. Initially, compute the distances between each data point and all other data points in the set of data Points. Then find out the closest pair of data points and form a set A1 consisting of these two data points, and delete them from the data point set D. Then determine the data point which is closest to the set A1, add it to A1 and delete it from D. Repeat this procedure until the number of elements in the set A1 reaches a threshold. At that point go back to the second step and form another data-point set A2. Repeat this till 'k' such sets of data points are obtained. Finally the initial centroids are obtained by averaging all the vectors in each data-point set. The Euclidean distance is used for determining the closeness of each data point to the cluster centroids. The distance between one vector X = ($x_1$,$x_2$ ...$x_n$) and another vector Y=($y_1$,$y_2$..yn) is obtained as

$$d(X,Y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots (x_n - y_n)^2}\,.$$

The distance between a data point X and a data point set D is defined as d(X, D) = min (d (X, Y), where Y ¸D). The initial centroids of the clusters are given as input to the second phase, for assigning data-points to appropriate clusters. The steps involved in this phase are outlined as Algorithm 4.

*4)Algorithm 4: Assigning data-points to clusters*
*Input:*
D = { d1, d2, d3……dn } // set of *n* data-points.
C = {c1,c2,c3….cn} // set of *k* centroids
*Output:*
A set of *k* clusters
*Steps:*

1) Compute the distance of each data-point *di* (1<=i<=n) to all the centroids *cj* (1<=j<=k) as *d (di ,cj)*;
2) For each data-point *di*, find the closest centroid *cj* and assign *di* to cluster *j*.
3) Set ClusterId[i] =j; // j: Id of the closest cluster

4) Set Nearest_Dist[i]S = *d(di, cj)*;
5) For each cluster *j* (1<=j<=k), recalculate the centroids;
6) Repeat
7) For each data-point *di*,
– 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;
– Else
    1) 7.2.1 For every centroid *cj* (1<=j<=k) Compute the distance *d (di, cj)*;
    2) End for;
    3) 7.2.2 Assign the data-point *di* to the cluster with the nearest centroid *cj*
    4) 7.2.3 Set ClusterId[i] =j;
    5) 7.2.4 Set Nearest_Dist[i] = *d (di, cj)*;
– End for;
8) For each cluster *j* (1<=j<=k), recalculate the centroids;

Until the convergence criteria is met

## IV. PROPOSED APPROACH

New algorithm data Partitioning Based K-means:

*5) Algorithm 5: Cluster using new algorithm data partitioning based K-Means*

Input
D= {d1, d2, d3……dn} // set of elements
K // number of desired cluster
Output:
K // set of clusters
Steps:

1) Phase 1: Determine the cluster using new algorithm data partitioning based k-means by algorthim6
2) Phase 2: Assign each data point to the appropriate clusters by algorithm 4

The proposed new algorithm data partitioning based k-means to find the cluster center initialization based on considering values for each attributes of the given data set this provides the some information leading to a good initial cluster center, the algorithm are describe below.

*6) Algorithm 6: Finding initial centroid using new algorithm data partitioning based k-means*

Input:
D = { d1, d2, d3……dn } // set of n data-points.
C = { c1,c2,c3….cn } // set of k centroids
Output:

1) Set m = 1;
2) Sort all data in the cell c in ascending order on each attribute value and links data by a linked list for each attribute.
3) Compute variance of each attribute of cell c. Choose an attribute axis with the highest variance as the principal axis for partitioning.
4) Compute squared Euclidean distances between adjacent data along the data axis with the highest variance $D_j=d(c_j,c_{j+1})$ and compute the $dsum_{l=} \sum^{l}_{j=1} D_j$
5) Compute centroid distance of cell c:
6) $CentroidDist = \sum^{n}_{i=1} dsum_i \, n$
7) where dsumi is the summation of distances between the adjacent data.
8) Divide cell c into two smaller cells. The partition boundary is the plane perpendicular to the principal axis and passes through a point m whose dsumi approximately equals to CentroidDist. The sorted linked lists of cell c are scanned and divided into two for the two smaller cells accordingly
9) Compute Delta clustering error for c as the total clustering error before partition minus total clustering error of its two sub cells and insert the cell into an empty Max heap with Delta clustering error as a key.
10) For each data-point set Am (1<=m<=k) find the arithmetic mean of the vectors of data points in Am, these means will be the initial centroids.

The proposed new algorithm will partition the given data into k cells, we start with a cell containing all given data and partition the cell into two cells. Later on we select the next cell to be partitioned that yields the largest reduction of total clustering errors (or Delta clustering error). This can be defined as Total clustering error of the original cell – the sum of Total clustering errors of the two sub cells of the original cell. This is done so that every time we perform a partition on a cell, the partition will help reduce the sum of total clustering errors for all cells, as much as possible. We can now use the partitioning algorithm to partition a given set of data into k cells. The centers of the cells can then be used as good initial cluster centers for the K-means

## V. RESULTS AND DISCUSSION

The proposed algorithm on Wine, leukemia dataset taken from the UCL repository of machine learning databases. The performance of the proposed clustering algorithm is evaluated using the following parameters,

- Number of Iterations
- Sum of the Squared Error
- Accuracy

### *A. Wine Dataset:*

This dataset is the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. There are three classes with number of instances as 59, 71 and 48 respectively.

### *B. Leukemia Dataset:*

The leukemia data set contains expression levels of 7129 genes taken over 72 samples. Labels indicate which of two variants of leukemia is present in the sample (AML, 25 samples, or ALL, 47 samples). This dataset is of the same type as the colon cancer dataset and can therefore be used for the same kind of experiments.

First, the number of iterations, accuracy and Sum of the Squared required for various techniques are compared. Table 1 represents the comparison of number of iterations required for various techniques with different dataset. From the table, it can be observed that the proposed clustering results in lesser number of iteration when compared to K-Means and modified K-Means techniques.

Table – 1
Comparison of Number of Iterations Required for the Proposed and Existing Technique for Various Datasets

| *Dataset* | *Iterations* | | |
|---|---|---|---|
| | *K-means algorithm* | *Modified k-means algorithm* | *New algorithm* |
| *Wine* | *7* | *5* | *3* |
| *Leukemia* | *9* | *5* | *2* |

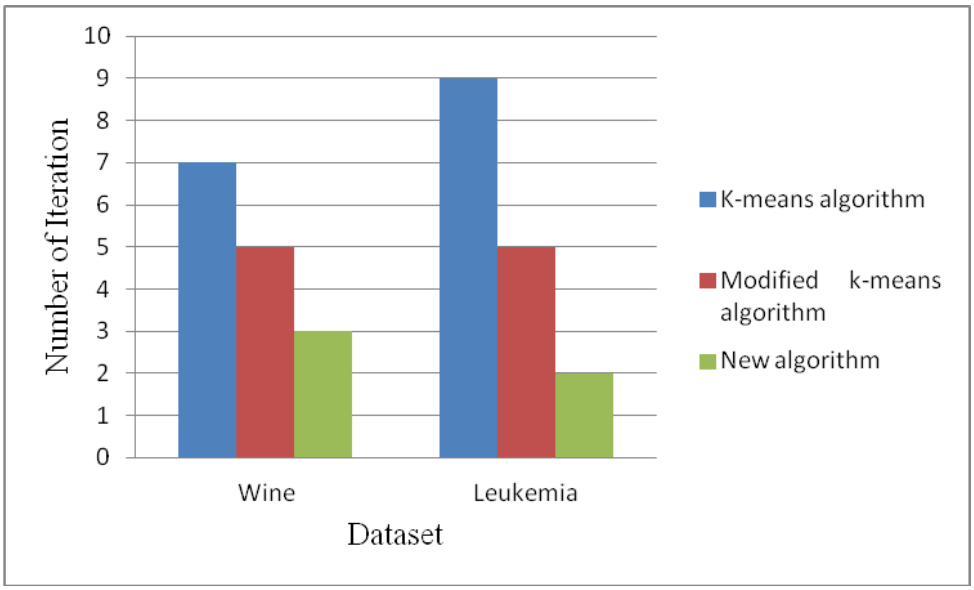


Fig. 2: No of Iteration for various Dataset

Figure 2 represents the resulted number of iteration for using various clustering algorithms. From the Figure 1, it is clear that the proposed clustering technique requires only lesser number of iteration when compared to other existing clustering techniques

Next, the Sum of the Squared Error resulted for various techniques are compared. Table 2 represents the comparison of resulted Sum of the Squared Error various techniques with different dataset. From the table, it can be observed that the proposed clustering results have minimum Sum of the Squared Error when compared to K-Means and modified K-Means techniques.

Table – 2
Comparison of SSE Resulted for the Proposed and Existing Technique for Various Datasets

| *Dataset* | *Sum of the Squared Error* | | |
|---|---|---|---|
| | *K-means algorithm* | *Modified k-means algorithm* | *New algorithm* |
| *Wine* | *0.0236* | *0.0119* | *0.0117* |
| *Leukemia* | *0.0983* | *0.0695* | *0.0640* |

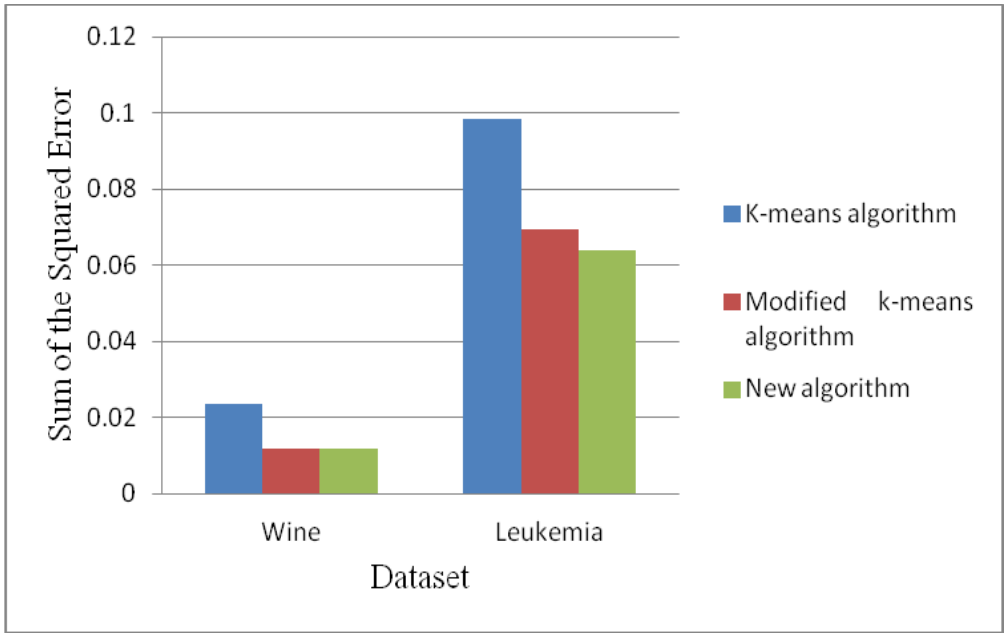


Fig. 3: Resulted Sum of the Squared Error for Wine Dataset

Figure 3 represents the result of comparison the Sum of Squared Error for wine & leukemia dataset. From the results, it can be observed that minimum Sum of the Squared Error resulted for the proposed clustering technique when compared to the existing clustering techniques.

Next, the clustering accuracy resulted for using various clustering algorithm is compared. Table 3 represents the comparison of resulted cluster accuracy for various techniques with different dataset. . From the table, it can be observed that the proposed clustering results have maximum accuracy when compared to K-Means and modified K-Means techniques.

Table – 3
Comparison of Accuracy Resulted for the Proposed and Existing Technique for Various Datasets

| *Dataset* | *Accuracy (%)* | | |
|---|---|---|---|
| | *K-means algorithm* | *Modified k-means algorithm* | *New algorithm* |
| *Wine* | *76.87%* | *78.30%* | *79.50%* |
| *Leukemia* | *90.95%* | *93.55%* | *95.24%* |

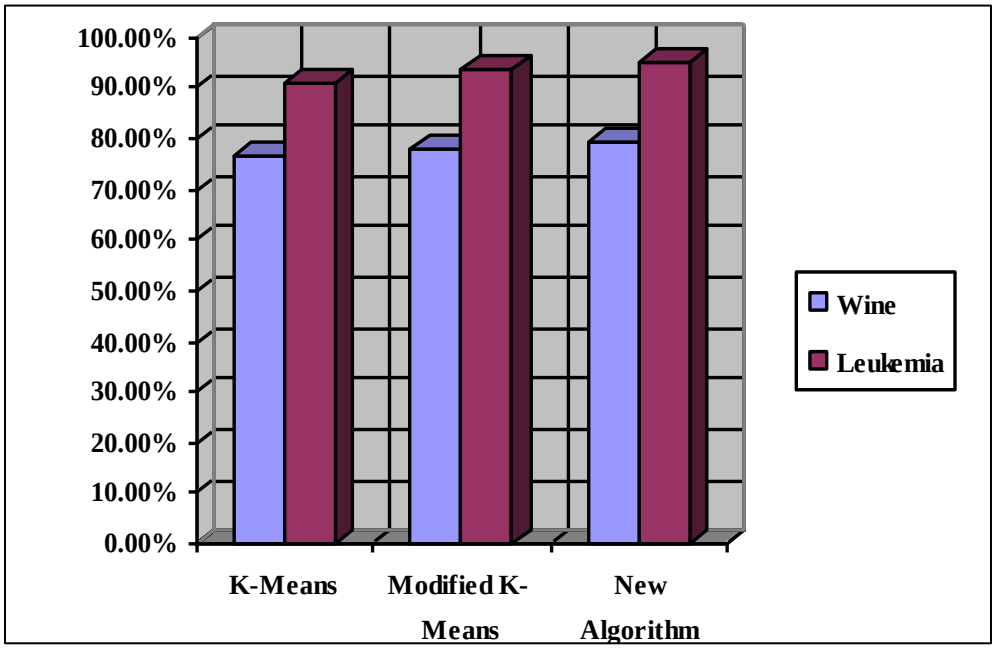


Fig. 4: Accuracy Comparison

## VI. CONCLUSION

Clustering is classifying of data into groups of similar objects. Representing the data by smaller amount of clusters necessarily loses certain fine details, but achieves simplification. It models data by its clusters. The most commonly used efficient clustering technique is k-means clustering as it is simple and effective. Generally, the cluster centroid is chosen in random before clustering starts. Sometime, this may leads to reduction in clustering accuracy and also increases the time for clustering. Therefore, initial starting points that are generated randomly by K-means algorithm plays an important role in better clustering result. So to overcome this disadvantage a new technique is proposed. The proposed clustering technique is evaluated using different dataset, namely Wine, Iris. The parameters used for comparison are number of iterations, Sum of the Squared Error and accuracy of clustering.

From the results, it can be observed that the proposed technique results in lesser number of iteration. When Sum of the Squared Error is considered, the proposed clustering technique results in lesser Sum of the Squared Error which indicates that the proposed technique will produce better accuracy for clustering. Considering all these results, the proposed clustering algorithm results in better clustering result when compared to the other existing techniques. This is satisfied for all the considered dataset.

## REFERENCES

[1] A. Likas, N. Vlassis and J.J. Verbeek, "The Global k-means Clustering algorithm", Pattern Recognition , Volume 36, Issue 2, 2003, pp. 451- 461.
[2] Weijiang Jiang; Jun Ye; "Decision-making method based on an improved similarity measure between vague sets", IEEE 10th International Conference on Computer-Aided Industrial Design & Conceptual Design (CAID & CD), Pp. 2086 – 2090, 2009.
[3] de Souza, R.M.C.; de Carvalho, F.A.T.; "A Clustering Method for Mixed Feature-Type Symbolic Data using Adaptive Squared Euclidean Distances", 7th International Conference on Hybrid Intelligent Systems (HIS), Pp. 168 – 173, 2007.
[4] Chen, B.; Tai, P.C.; Harrison, R.; Yi Pan; "Novel hybrid hierarchical-K-means clustering method (H-K-means) for microarray analysis", IEEE Computational Systems Bioinformatics Conference, Pp. 105 – 108, 2005.
[5] Mingwei Leng; Haitao Tang; Xiaoyun Chen; "An Efficient K-means Clustering Algorithm Based on Influence Factors", Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD), Vol. 2, Pp. 815 – 820, 2007.
[6] Merz C and Murphy P, UCI Repository of Machine Learning Databases, Available: ftp://ftp.ics.uci.edu/pub/machine-learningdatabases
[7] S. S. Khan and A. Ahmad, "Cluster Center Initialization for K-mean Clustering", Pattern Recognition Letters, Volume 25, Issue 11, 2004, pp. 1293-1302