Volume 3, No. 5, Sept-Oct 2012
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

# An Efficient Selection of Initial Cluster in K-Means Using Entropy and Co-efficient of Variation for High Dimensional Data

Dr. P. Krishnakumari
Director, Dept of Computer Application (MCA),
R. V. S. College of Arts and Science,
Coimbatore, India.
kkjagadeesh@yahoo.com

P. Gokila*
M.Phil Research Scholar in Computer Science,
Sri Ramakrishna CAS for women, Coimbatore, India
Coimbatore, India.
goki242@gmail.com

***Abstract:*** K-means clustering is a method of cluster analysis which aims to partition n observations into k-clusters in which each observation belongs to the cluster with the nearest mean. For high dimensional dataset K-means can't give better cluster output, feature selection methods are required to remove irrelevant features. But the proposed algorithm selects primary and secondary axes based on means and variations of individual column. Here entropy value is combined with mean and variation of axes selection. This may find the axes which are more relevant and highly ranked. The integration of these two techniques achieves feature selection and initial centroid selection simultaneously. Real-time high dimensional datasets are used for experiments to show that the proposed algorithm provides better results for high-dimensional dataset.

***Keywords:*** K-means algorithm, initial cluster centers, high dimensional dataset, error percentage, entropy.

## I. INTRODUCTION

Data mining is also called as a data. It is the process of analyzing data from dissimilar prospect and summarizing it into serviceable information. Data mining is used to find correlations or patterns among number of fields in large databases. It utilises methods of artificial intelligence, machine learning, statistics and database systems. It involves some common classes of tasks are anomaly detection, association rule learning, classification, regression and clustering. Clustering is a important analytical method in data mining. A cluster analysis is a method for statistical data analysis.

Clustering is an important unsupervised learning problem, which deals with discover a collection of unlabelled data. Clustering can be the task of organizing data objects whose members in groups are similar. A cluster is a collection of objects that are "similar" in same cluster objects and "dissimilar" to the other cluster objects. The main goal of clustering is to resolve the intrinsic grouping a set of unlabelled data [1]-[4].

## II. K-MEANS CLUSTERING ALGORITHM

The most popular clustering algorithm is a K-means clustering algorithm, which is an exclusive clustering algorithm. K-means clustering algorithm is developed by Mac Queen in 1967.

It is a partition clustering algorithm and it is very effective in smaller datasets [5] & [6]. First select k initial centers based on desired number of clusters. The user can specify k parameter value. Each data point is assigned to nearest centroid and the set of points assigned to the centroid is called a cluster. Each cluster centroid is updated based on the points assigned to the cluster. The process will be repeated until the centroids remain the same or no point changes clusters. The main drawback of K-means algorithm is the quality of the clustering results highly depends on random selection of the initial centroids. For different runs it gives different clusters for the same input data. Another problem in K-means clustering algorithm is it does not work well in high-dimensional data.

## III. RELATED WORK

The problem is traditional K-means algorithm on random selection of the initial centroid points. Several methods have been proposed to find better initial cluster centroids.

Murat Erisoglu et al. [7] proposed a new algorithm for computing initial cluster center in K-means algorithm. Two principal variables are chosen based on maximum coefficient of variation and minimum absolute value of the correlation.

Fang Yuan et al. [8] & K. A. Abdul Nazeer [9] proposed the initial centroids algorithm based on systematic method. It find the distance between each data-points, then points are more similar and build initial centroids depends on these data-points. Different initial cluster centroids provide different cluster results. The accuracy of output is affected in traditional K-means algorithm; the improved K-means clustering algorithm produces high accuracy.

A.M.FAHIM et al. [10] & [11] proposed an efficient enhanced K-means clustering algorithm for large number of clusters in dataset and it makes a K-means more efficient. The traditional K-means algorithm computes distance between each data point and each centroid, it's computationally very expensive. Enhanced algorithm maintains a distance to the nearest cluster for each data point. For each iteration uses a previous nearest cluster distance. So, it reduces the number of distance calculation and also elements into the appropriate clusters. But it use random selection of initial cluster centers and it does not produce the unique clustering results.

Likas et al. [12] proposed the global K-means algorithm is an incremental approach to clustering. It adds one cluster center dynamically to determine global search procedure. It provides an optimal solution to the clustering problem and minimizes the clustering error.

D. Napoleon et al. [13] proposed PCA for dimensionality reduction is applied to K-means clustering algorithm. It is used to reduce the sum of total clustering errors in the each cluster and also reduce attributes and dataset is applied to compute K-means cluster centroids.

P. Prabhu et al. [14] proposed a new method Principal Component Analysis is used to reduce the dataset from high dimensional to low dimensional. It produces accurate clustering results compared with original K-means algorithm.

Dudu Lazarov et al. [15] introduce a new cluster method named Smart-Sample and it compared the performance with different data clustering methods. The Smart-Sample cluster method is used to cluster a large high-dimensional datasets successfully. It generates accurate data clustering results.

S. Deelers et al. [16] proposed an algorithm for K-means algorithm to compute cluster centers. It is based on data partitioning algorithm and it is used to reduce the sum of total cluster errors in each clusters. Cutting plane is used to partition the dataset cell into two smaller cells. The plane is perpendicular to the data axis with the highest variance and it reduces the sum of squared errors of two cells. Each cells are partitioned one at a time until the number of cells equals to the specified number of clusters, k. The centers of k cells become the initial cluster centers for K-means.

## IV. EXISTING METHOD

Clustering is the task of segregation of objects into groups of similar or near by similar objects. In feature space clusters are separated groups so it is essential to select initial centers which are well separated. In existing method initial cluster centers for K-means algorithm is computed. In this, two axes are selected one for maximum coefficient of the variations and another for minimum absolute value of the correlation. Then the cluster membership is obtained according to candidate initial cluster and related two axes [7]. After that the algorithm is applied for normalizing the datasets. It is efficient and works well on low-dimensional clusters and doesn't applicable for high-dimensional. So the proposed algorithm is generated to overcome the problem of the existing method.

## V. PROPOSED ALGORITHM

In this section, the ECV (Entropy and Co-efficient of Variation) K-means proposed algorithm generate initial cluster center in K-means for high-dimensional data. In this algorithm two principal variables are selected based on two axes, one is main axis and another is second axis. In main axis calculate the coefficient of variation $CV_j$. Variation coefficient calculate the Standard Deviation and Mean value of variable j. The general formula for the Standard Deviation (σ) is

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} a_i^2}{n} - \left(\frac{\sum_{i=1}^{n} a_i}{n}\right)^2}$$

The Average Mean value is

Mean = Sum of elements /
Number of elements
= a1+a2+a3+.....+an/n

Then the entropy value is determined

$$E = -\sum_{j=1}^{c} P_j \log P_j$$

Where c is number of classes of data, *Pj* is the proportion of data of class *j* in a given cluster[17] & [18]. A weighted sum score is calculated sum of highest variance and lowest entropy. Among weighted sum select highest value as a main axis. Remove the variable having higher entropy value. After determining the main axis, correlation coefficient is used as a second axis.

$$r_{xy} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{(n-1)s_x s_y}$$

Minimum absolute value of correlation among main axis variable and other variable determines a second axis. After finding main axis and second axis mean of data points is calculated as a center of data set. Euclidean distance is calculated for main axis data points and center and also second axis data point and center. Select the highest distance data point as initial cluster center. The same mechanism is applied to select candidate for second initial cluster center.

Respectively, the next candidate of the rest initial cluster calculated between each data point and $C_{r-1}$(where r is a current iteration step). Finally, it is added to the sum of distance (r-1) in rth iteration. The process is repeated until predefined number of cluster equal to initial cluster center.

**Algorithm Steps: Proposed ECV Kmeans**

**Step1:** Choosing two variables that best describe the changes in the dataset according to two axes.

**Step2:** Calculate absolute value of the variation coefficient using "(1)"

$CV_j = |\sigma / Mean|$ (1)

**Step 3:** Calculate entropy value of the columns using "(2)"

$$E = -\sum_{j=1}^{c} P_j \log P_j \quad (2)$$

**Step 4:** Finding the weighted sum score of variables by using variance and entropy.

W(i) =k1(1)*CV(j)+k1(2)*E(j) (3)

The maximum value of the weighted sum variable is selected as main axis.

**Step 5:** The higher weighted sum score value of column variables are removed based on specified threshold value.

**Step6:** Correlation coefficient is calculated between main axis and other variables by using "(4)"

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \quad (4)$$

Minimum absolute value of the correlation between main axis variable and other variables is selected as second axis.

**Step7:** Calculate a center of the dataset according to the selected axis

$m = [x_1\ y_1]$ (5)

Where $x_1$ is mean according to the selected variable main axis. $y_1$ is also defined similarly.

**Step8:** Euclidean distances are calculated between each data point and the center.

$d_{im} = sqrt((x_2 - x_1)^2 + (y_2 - y_1)^2)$ (6)

**Step9:** Again calculate a Euclidean distance for first initial cluster center. The highest distance in $c_1$ is to select the first candidate of the initial cluster center

$d_{ic1} = \text{sqrt}((x_2 - x_m1)^2 + (y_2 - y_m1)^2)$ (7)

**Step10:** To calculate the rest initial cluster center iteration is

$Sd_{i3} = d_{ic1} + d_{ic2}$, i = 1, 2, ……., n (8)

**Step 11:** Repeat the iterative process until the number of initial cluster centers equal to the predefined number of clusters.

## VI. EXPERIMENTAL RESULTS

The proposed ECV algorithm evaluates the number of real world datasets such as Iris data, Wine recognition data, and Spectf Heart data, ECG data, Lymphoma data from UCI machine learning repository.

Table 1: Dataset Description

| Dataset | Data Size | Dimension |
|---|---|---|
| Iris | 150 | 4 |
| Wine | 178 | 14 |
| Spectf Heart | 267 | 45 |
| ECG | 1800 | 8 |
| Lymphoma | 33 | 37 |

The above Table I represents datasets that are used for testing the proposed method. The mean and standard deviation are calculated for the Iris dataset are given in Table II.

Table 2: Mean and SD Value

| | Mean | Standard Deviation |
|---|---|---|
| Sepal length | 5.785 | 0.577 |
| Sepal width | 3.045 | 0.419 |
| Petal length | 3.67 | 1.671 |
| Petal width | 1.21 | 0.767 |

Table 3: Variation Co-efficient and Entropy Value

| | Co-efficient of variation | Entropy |
|---|---|---|
| Sepal length | 0.0997 | 1.0183 |
| Sepal width | 0.1376 | 0.8925 |
| Petal length | 0.4553 | 1.1085 |
| Petal width | 0.6339 | 1.0418 |

The co-efficient of variation and entropy values computed for the Iris dataset are given in above Table III. Calculate a weighted sum score for co-efficient of variation and entropy. Here the threshold value is 0.3. Based on given threshold value select a maximum weighted sum score as a main axis and remove a column attribute less than given threshold value.

Weighted sum score for attribute Sepal length, Sepal width, Petal length and Petal width are given in Table IV.

Table 4: Weighted sum score

| | Weighted Sum W(s) |
|---|---|
| Sepal length | 0.3753 |
| Sepal width | 0.3641 |
| Petal length | 0.7013 |
| Petal width | **0.7562** |

Here each attribute value is greater than or equal to threshold value. So, here the attributes are not removed. When attribute value is less than threshold value removes the attribute from total attribute.

According to Table IV, to select a petal width as a main axis which has maximum value of weighted sum score. The minimum absolute value of the correlation coefficient is used to determine the second axis. The correlation coefficient among the petal width and other attributes are given in Table V.

Table 5: The correlation coefficient between the petal width and the other variables

| | Petal width |
|---|---|
| Sepal length | 0.8023 |
| Sepal width | **-0.6349** |
| Petal length | 0.9581 |

The Sepal width is selected as a second axis which has minimum absolute value of correlation coefficient according to the Table V by proposed algorithm. Based on main axis and second axis to determined the data center as m = [1.21, 3.045].

Select the highest distance data point as a initial cluster center of the Iris data is calculated using Euclidean distance for k-means algorithm are obtained as $m_1$ = [5.8000 2.8000 5.1000 2.4000], $m_2$ = [5.7000 3.8000 1.7000 0.3000]. Repeat the process until desired cluster center is obtained.

## VII. COMPARISON CRITERIA

**"**Fig. 1, Fig. 2 and Fig. 3" have been compared with Wine dataset to prove the unique cluster formation.

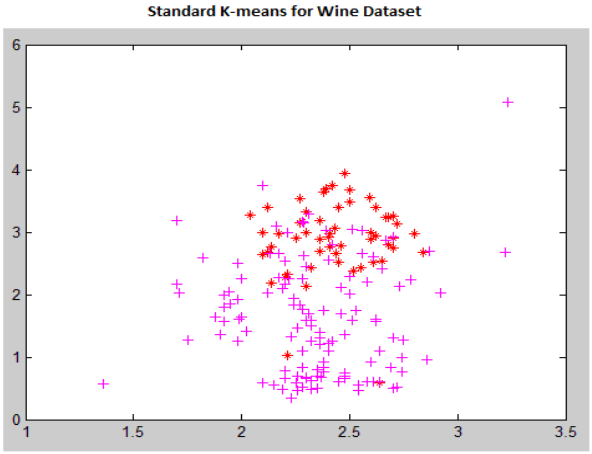


Figure 1. Standard K-means clustering result for Wine Dataset

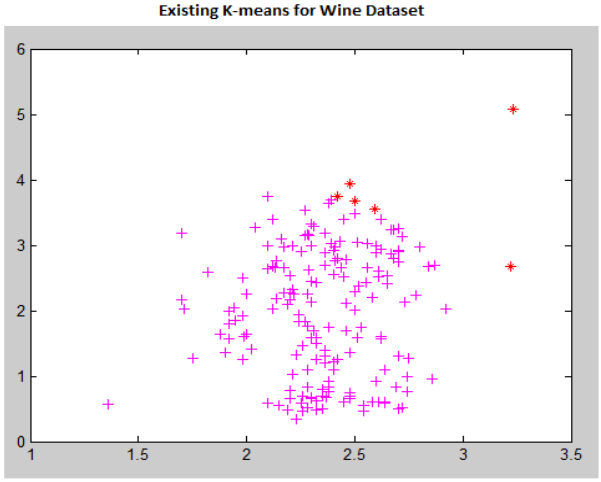


Figure 2. Existing K-means clustering result for Wine Dataset

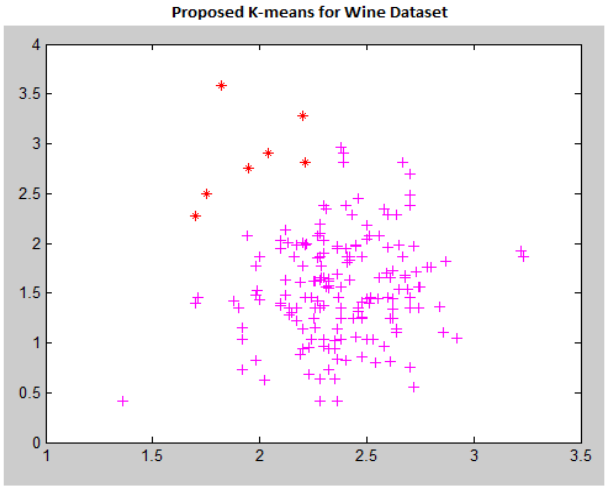


Figure 3. Proposed K-means clustering result for Wine Dataset

To compare the clustering results with the Selected and Total attributes are given in Table VI.

Table 6: Comparison between Selected and Total attributes

| Dataset | Selected Attributes | Total Attributes |
|---|---|---|
| Iris | 4 | 4 |
| Wine | 12 | 14 |
| Spectf Heart | 34 | 45 |
| ECG | 8 | 8 |
| Lymphoma | 11 | 37 |

To compare the results in terms of the Selected and Total attributes are given in "Fig. 4".

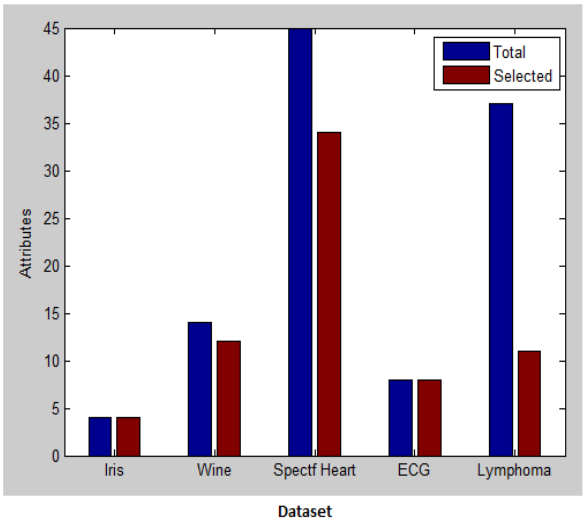


Figure 4. Comparison between selected and Total attributes

To compare the clustering results with the Error percentage, the Rand index and Wilk's lambda test. To represent the Error percentage is calculated from number of misclassified patterns and the total number of patterns in the data sets. The Error percentage is defined "(9)" as follows,

$$\text{Error} = \frac{\text{Number of misclassified}}{\text{Number of Patterns}} \times 100\% \quad (9)$$

The comparison of initial cluster centers computed using proposed algorithm and existing algorithm for the data sets, is shown in Table VII.

To compare the results in terms of the classification error (%), given in "Fig. 5" for Standard K-means, proposed ECV K-means and existing New K-means algorithm.

Table 7: Comparison results between proposed and existing algorithm according to error percentage

| Dataset | Method | Error Percentage |
|---|---|---|
| Iris | ECV K-means | **96.6667** |
| | New K-means | 106.2383 |
| | Standard K-means | 117.1976 |
| Wine | ECV K-means | **70.2247** |
| | New K-means | 74.9931 |
| | Standard K-means | 85.5403 |
| Spectf Heart | ECV K-means | **80.0000** |
| | New K-means | 99.5022 |
| | Standard K-means | 109.6408 |
| ECG | ECV K-means | **67.4712** |
| | New K-means | 71.9101 |
| | Standard K-means | 82.0594 |
| Lymphoma | ECV K-means | **15.1515** |
| | New K-means | 21.0302 |
| | Standard K-means | 31.2878 |

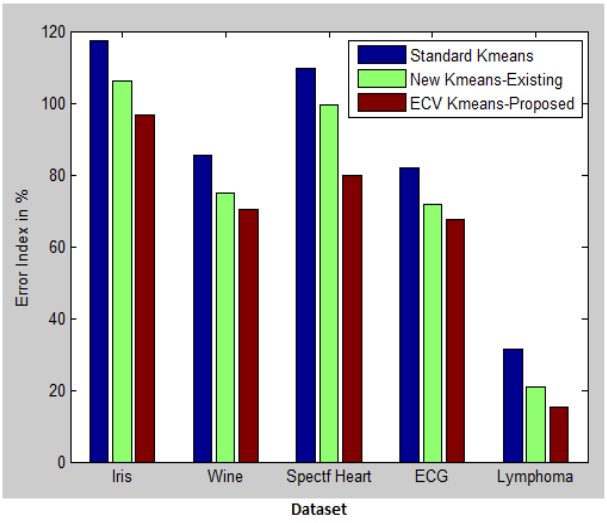


Figure 5. Error comparison between Standard K-means, proposed ECV K-means and existing New K-means

The Rand index given a set of n elements R = $\{i_1,\ldots\ldots, i_n\}$ and two partitions of R represent S and P. S = $\{s_1,\ldots\ldots,s_k\}$ and P = $\{p_1,\ldots\ldots,p_k\}$ a partitions of R into k subsets.

Four principles are

a) The number of pairs of elements in R that are in the same set in S and in the same set in P.
b) The number of pairs of elements in R that are in different sets in S and in different sets in P.
c) The number of pairs of elements in R that are in the same set in S and in different sets in P.
d) The number of pairs of elements in R that are in different sets in S and in the same set in P.

The Rand index is given by "(10)"

$$\text{Rand index} = \frac{a+b}{a+b+c+d} \quad (10)$$

a + b can be considered as no. of agreements between S and P and c + d as no. of disagreements between S and P. The Rand index has a value between 0 and 1, with 0 represents that the two partition data points are different and 1 represents that the two partition data points are same.
The Wilk's lambda is given by "(11)"

$$\lambda = \frac{|W|}{|W+B|} \qquad (11)$$

Where W is sum of squares and products matrix and W + B is total number of sum of squares and products matrix.

Table 8: Comparison results between proposed and existing algorithm according to Rand index and Wilk's lambda

| Dataset | Method | Rand index | Wilk's lambda |
|---|---|---|---|
| Iris | ECV K-means | **0.9351** | **0.1387** |
| | New K-means | 0.9351 | 0.1387 |
| | Standard K-means | 0.9001 | 0.2228 |
| Wine | ECV K-means | **0.5794** | **0.9791** |
| | New K-means | 0.5582 | 0.9999 |
| | Standard K-means | 0.5385 | 1.0253 |
| Spectf Heart | ECV K-means | **0.8177** | **0.6135** |
| | New K-means | 0.6759 | 0.9499 |
| | Standard K-means | 0.6508 | 1.0313 |
| ECG | ECV K-means | **0.5937** | **0.9536** |
| | New K-means | 0.5201 | 0.9921 |
| | Standard K-means | 0.4585 | 1.0165 |
| Lymphoma | ECV K-means | **0.7803** | **0.8142** |
| | New K-means | 0.7348 | 0.9935 |
| | Standard K-means | 0.6875 | 1.0865 |

The results are compared in terms of the Rand index and Wilk's lambda test, given in "Fig. 6" and "Fig. 7" for Standard K-means, proposed ECV and existing New K-means algorithm.

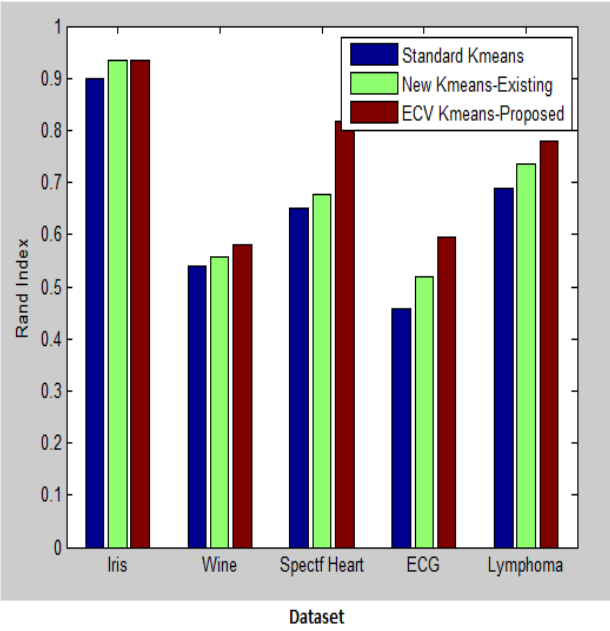


Figure 6. Rand index between Standard K-means, proposed ECV K-means and existing New K-means

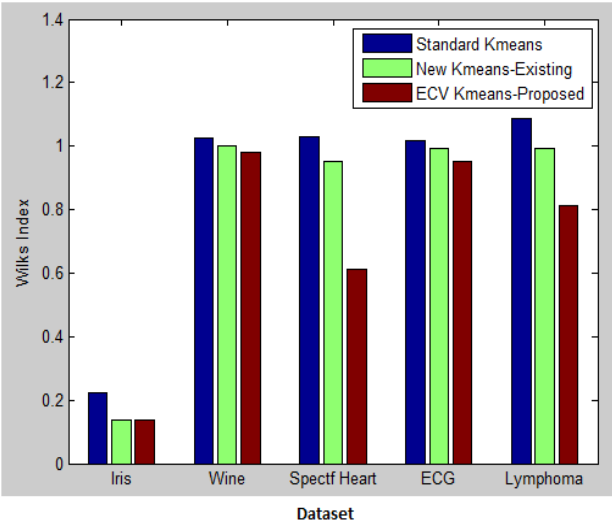


Figure 7. Wilk's lambda between Standard K-means, proposed ECV K-means and existing New K-means

Table IX shows to compare the CPU time of the proposed ECV algorithm with Standard K-means and New K-means method.

Table IX: Performance Comparison of Standard K-means, proposed ECV and existing New K-means

| Dataset | Method | Time taken |
|---|---|---|
| Iris | ECV K-means | **0.1465** |
| | New K-means | 0.3336 |
| | Standard K-means | 0.3595 |
| Wine | ECV K-means | **0.0921** |
| | New K-means | 0.1131 |
| | Standard K-means | 0.1456 |
| Spectf Heart | ECV K-means | **0.0993** |
| | New K-means | 0.1242 |
| | Standard K-means | 0.1538 |
| ECG | ECV K-means | **0.1024** |
| | New K-means | 0.1300 |
| | Standard K-means | 0.1903 |
| Lymphoma | ECV K-means | **0.0995** |
| | New K-means | 0.1395 |
| | Standard K-means | 0.1809 |

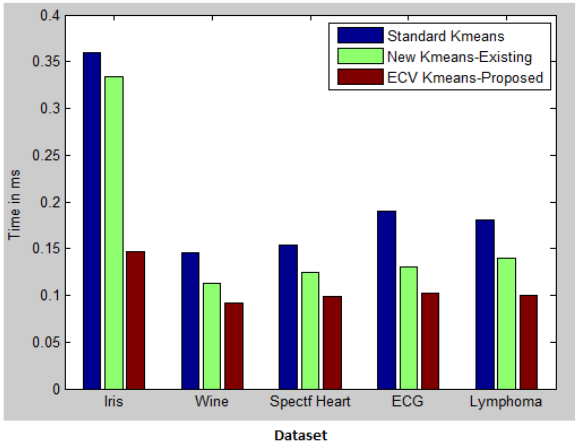


Figure 8. Time Taken between Standard K-means, proposed ECV K-means and existing New K-means

## VIII. CONCLUSION

The k-means algorithm is broadly used for clustering huge sets of data. But the classical k-means algorithm do not always provides good results. Because the output clusters is formed based on the selection of initial centroid. For improving performance of k-means cluster two principal variables are selected according to maximum coefficient of the variation and minimum absolute value of the correlation. This method also cannot give good results for high dimensional data. Even though best cluster centroids are selected by this methods clusters are formed based on all variables. For supporting the same methods for high dimensional data, the maximum coefficient variation and entropy values are combined for selecting the centroids. The experimental results show that proposed algorithm provides better results for various datasets. The constraint of the proposed algorithm is number of cluster. The number of clusters is still being given as an input. In future, an efficient method for determining number of cluster can be investigates by the algorithm itself.

## IX. REFERENCES

[1] A.K.Jain and R.C.Dubes, "Algorithms for Clustering Data". Englewood Cliffs, NJ: Prentice-Hall, 1988.

[2] M.R.Anderberg, "Cluster Analysis for Application". Academic Press, 1973.

[3] Margaret H Dunham, "Data Mining-Introductory and Advanced Concepts", Pearson Education, 2006.

[4] H. Tsai, S. Horng, S. Tsai, S. Lee, T. Kao, and C. Chen. "Parallel clustering algorithms on a reconfigurable array of processors with wider bus networks", in Proc. IEEE International Conference on Parallel and Distributed Systems, 1997.

[5] Moth'd Belal. Al-Daoud, "A New algorithm for cluster Initialization", World Academy of Science, Engineering and Technology 4 2005.

[6] Kohei Arai and Ali Ridho Barakbah, "Hierarchical K-means: An Algorithm for centroids initialization for K-means", Reports of the faculty of Science and Engineering, Saga University, Vol.36, No.1, 2007.

[7] Murat Erisoglu, Nazif Calis, Sadullah Sakallioglu, "A new algorithm for initial cluster centers in k-means algorithm", in Pattern Recognition Letters 32 (2011) 1701-1705.

[8] F. Yuan, Z. H. Meng, H. X. Zhangz, C. R. Dong, "A New Algorithm to Get the Initial Centroids", proceedings of the 3rd International Conference on Machine Learning and Cybernetics, pp. 26-29, August 2004.

[9] K. A. Abdul Nazeer, M. P. Sebastian, "Improving the accuracy and efficiency of the K-means clustering algorithm", Proceedings of the World congress on Engineering 2009 vol I.

[10] A. M. Fahim, A. M. Salem, F. A. Torkey and M. A. Ramadan, "An Efficient enhanced k-means clustering algorithm", journal of Zhejiang University, 10(7): 16261633, 2006.

[11] A. M. Fahim, G. Saake, A. M. Salem, F. A. Torkey and M. A. Ramadan, "K-means for Spherical clusters with Large Variance in Sizes", World Academy of Science, Engineering and Technology 45 2008.

[12] Likas, Vlassis and J. J. Verbeek, "The global K-Means clustering algorithm", in Pattern Recognition , vol. 36, no. 2, pp. 451-461, 2003.

[13] D. Napoleon and S. Pavalakodi, "A New method for dimensionality reduction using k-means clustering algorithm for high dimensional data set", International journal of computer applications, (0975-8887) vol.13-No.7, Jan 2011.

[14] P. Prabhu and N. Anbazhagan, "Improving the performance of k-means clustering for high dimensional data set", International journal on computer science and engineering.

[15] Dudu Lazarov, Gil David, Amir Averbuch, "Smart-Sample: An Efficient Algorithm for clustering Large High-Dimensional Datasets", Tel-Aviv University, Tel-Aviv 69978, Israel.

[16] S. Auwatanamongkol and S. Deelers, "Enhancing K-Means Algorithm with Initial Cluster Centers Derived from Data Partitioning along the Data Axis with the Highest Variance", International Journal of Computer Science, Vol. 2, Number 4.

[17] Robert Jenssen, Kenneth E. Hild II, Deniz Erdogmus, Jase C. Principe and Torbjorn Eltoft, "Clustering using Renyi's Entropy", 2003.

[18] V. V. Jaya Ramakrishnaiah, Dr.K. Ramchand H Rao, Dr. R.Satya Prasad, "Entropy Based Mean Clustering: A Enhanced clustering approach", The International Journal of computer science and applications, ISSN -2278-1080, Vol 1, No.3, May 2012.

 