International Journal of Science and Engineering Applications
Volume 7–Issue 12,479-482, 2018, ISSN:-2319–7560

Efficient K-Means Clustering Algorithm Using Feature Weight and Min-Max Normalization

Ei Ei Phyo
Department of Information Technology
Technological University (Thanlyin),
Myanmar

Ei Ei Myat
Department of Information Technology
Technological University (Thanlyin),
Myanmar

Abstract: Clustering is a process of partitioning a set of data into a set of meaningful sub-classes, called clusters. K-means is an effective clustering technique used to separate similar data into groups based on initial centroids of clusters. In this paper, the proposed algorithm applies normalization prior to clustering on the available data as well as the proposed approach calculates initial centroids based on weights. Experimental results prove the betterment of proposed clustering algorithm over existing K-means clustering algorithm in terms of computational complexity and overall performance.

Keywords: clustering, k-means clustering, min-max normalization, gain ratio, initial centroid

1. INTRODUCTION

Data mining [1] [2] or knowledge discovery is a process of analyzing large amounts of data and extracting useful information. Data mining is widely used in various areas like financial data analysis, retail and telecommunication industry, biological data analysis, fraud detection, spatial data analysis and other scientific applications. Clustering is categorized as one of the data descriptive analysis technique that builds clusters of data objects in such a way that objects in a cluster are closer to each other than the objects of other clusters. K-means uses the concept of Euclidean distance to calculate the centroids of the clusters. This method is less effective when new data sets are added and have no effect on the measured distance between various data objects. The computational complexity of k means algorithm is also very high [1] [3]. K-means is the most popular and best understood traditional clustering algorithm which starts by selecting the random initial centroids and computes the distance between the centroids and the data objects are computed. The objects are then clustered with the centroids at a minimum distance [4]. The algorithm iteratively groups the data objects with minimum distance until there is no change in the centroid or members of the cluster group. Normalization is used to eliminate redundant data and ensures that good quality clusters are generated which can improve the efficiency of clustering algorithms. So it becomes an essential step before clustering as Euclidean distance is very sensitive to the changes in the differences [5]. A feature weight algorithm can be seen as the combination of a search technique for proposing new feature subsets, along with an evaluation measure which scores the different feature subsets. The simplest algorithm is to test each possible subset of features finding the one which minimizes the error rate. This is an exhaustive search of the space, and is computationally intractable for all but the smallest of feature sets [15].

The remaining sections of the paper are organized as follows: Section 2, review of related works, 3 describes the methodology, section 4 denotes the comparison methods, section 5 explicates the experimental results and finally section 6 explains the conclusion of the proposed work.

2. RELATED WORKS

Zhang Chen *et al*. [6] proposed the initial centroids algorithm based on k-means that have avoided alternative randomness of initial center. Fang Yuan [7] proposed the initial centroids algorithm. The standard k-means algorithm selects k-objects randomly from the given data set as the initial centroids. If different initial values are given for the centroids, the accuracy output by the standard k-means algorithm can be affected. In Yuan's method the initial centroids are calculated systematically. To overcome the efficient k-means clustering method reduces computing efficiency and accuracy when the dataset is increased.

3. METHODOLOGY

3.1 Min-Max Normalization

Min-max normalization [14] performs a linear transformation on the original data. Min-max is a technique that helps to normalize a dataset. It will scale the dataset between the 0 and 1. Suppose that min_A and max_A are the minimum and maximum values of an attribute, *A*. Min-max normalization maps a value, *v*, of *A* to *v*' in the range [new_min_A, new_max_A] by computing

$$v' = \frac{v - minA}{maxA - minA} (new\_maxA - new\_minA) + new\_minA$$

Min-max normalization preserves the relationships among the original data values. It will encounter an "out-of-bounds" error if a future input case for normalization falls outside of the original data range for *A*.

3.2 Gain Ratio

Information gain applied to attributes that can take on a large number of distinct values might learn the training set too well. The information gain measure is biased toward tests with many outcomes. That is, it prefers to select attributes having a large number of values. The gain ratio [15] is defined as

$$\text{GainRatio (A)} = \frac{\text{Gain(A)}}{SplitInfo(A)}$$

The attribute with the maximum gain ratio is selected as the splitting attribute. The split information approaches 0, the ratio becomes unstable. A constraint is added to avoid this, whereby

the information gain of the test selected must be large at least as great as the average gain over all tests examined.

3.3 K-Means Clustering Algorithm

The basic idea of K-means algorithm is to classify the dataset D into k different clusters where D is the dataset of n data; k is the number of desired clusters. The algorithm consists of two basic phases [12]. The first phase is to select the initial centroids for each cluster randomly. The second and final phase is to take each point in dataset and assign it to the nearest centroids [12]. To measure the distance between points Euclidean Distance method is used. When a new point is assigned to a cluster the cluster mean is immediately updated by calculating the average of all the points in that cluster [13]. After all the points are included in some clusters the early grouping is done. Now each data object is assigned to a cluster based on closeness with cluster center where closeness is measured by Euclidean distance. This process of assigning a data points to a cluster and updating cluster centroids continues until the convergence criteria is met or the centroids don't differ between two consecutive iterations. Once, a situation is met where centroids don't move any more the algorithm ends. The k-means clustering algorithm is given below.

Step 1: Begin with a decision on the value of k = number of clusters.

Step 2: Put any initial partition that classifies the data into k clusters. You may assign the training samples randomly, or systematically as the following:

1. Take the first k training sample as single-element clusters

2. Assign each of the remaining (N-k) training samples to the cluster with the nearest centroid.

After that each assignment, recomputed the centroid of the gaining cluster.

Step 3: Take each sample in sequence and compute its distance from the centroid of each of the clusters. If a sample is not currently in the cluster with the closest centroid, switch this sample to that cluster and update the centroid of the cluster gaining the new sample and the cluster losing the sample.

Step 4: Repeat step 3 until convergence is achieved, that is until a pass through the training sample causes no new assignments.

3.4 Proposed Methodology

The proposed efficient k-means clustering is upgraded the origin k-means clustering to reduce the computational complexity. In the proposed efficient k-means clustering method, the normalization and feature weight are applied. Firstly, the methodology employs normalized dataset by using min-max normalization to improve the efficiency of clustering algorithm. After that gain ratio method compute feature weights for each attributes of the data to minimize the error rate. It the centroids are then posted to the traditional clustering algorithms for being executed in the way it normally does. The results of the proposed work are validated against number of iterations and accuracy obtained and compared with the randomly selected initial centroids.

Step 1: Accept the dataset to cluster as input values

Step 2: Perform a linear transformation on the original dataset using mix-max normalization

Step 3: Compute the feature weight for each attribute and update the dataset.

Step 4: Initialize the first K cluster

Step 5: Calculate centroid point of each cluster formed in the dataset.

Step 6: Assign each record in the dataset for only one of the initial cluster using a measure Euclidean distance.

Step 7: Repeat step 4 until convergence is achieved, that is until a pass through the training sample causes no new assignments.

4. COMPARISON METHODS

4.1 Normalized Mutual Information (NMI)

The normalized mutual information [11] is a good measure for determining the quality of clustering. Comparing the NMI between different clustering, having different number of clusters the proposed efficient k-means clustering can be measured. The value of NMI is large, the cluster quality is good.

$$\text{NMI (Y, C)} = \frac{2 \times I(Y,C)}{[H(Y)+H(C)]}$$

4.2 Silhouette Coefficient (SC)

The silhouette coefficient [9] is used to compare the quality of clustering for origin k-means and proposed method. The cluster quality is good, when the value of SC is large.

$$SC = \frac{1}{N}\sum_{i=1}^{N} s(x)$$

$$s(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}}$$

4.3 Sum of Square Error (SSE)

The sum of square error [10] is defined to use measure the quality of clustering that is the difference of error between the original k-means and proposed method. The value of SSE is small, the cluster quality is good.

$$\text{SSE} = \sum_{i=1}^{K} \sum_{x \epsilon Ci} dist2\ (mi, x)$$

5. EXPERIMENTAL RESULTS

In this section, we use Iris [8] dataset to validate the proposed algorithm. The performance of our proposed algorithm is examined in the quality of clustering required with different real world dataset and compared with the origin k-means algorithm.

In Table 1 and Figure 1 shows the comparison of proposed algorithm and k-means with the quality of clustering for cluster two.

Table 1. Evaluation result of Iris data for the quality of clustering, k = 2

Number of Clusters	Method	Normalized Mutual Information (NMI)	Silhouette Coefficient (SC)	Sum of Square Error (SSE)
K = 2	K-Means	0.6565	0.7813	0.9077

	Propose d	0.6565	0.8149	0.0047

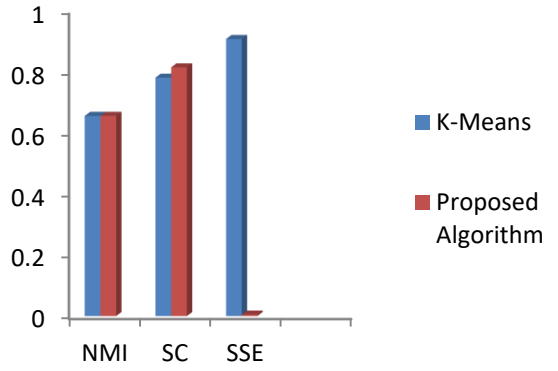

Figure 1. The quality of clustering for cluster 2

The comparison of proposed algorithm and k-means with the quality of clustering for cluster three. The result is shown in Figure 2 and Table 2.

Table 2. Evaluation result of Iris data for the quality of clustering, k = 3

Number of Clusters	Method	Normalized Mutual Information (NMI)	Silhouette Coefficient (SC)	Sum of Square Error (SSE)
K = 3	K-Means	0.7419	0.8149	0.5281
	Proposed	0.8642	0.8515	0.0023

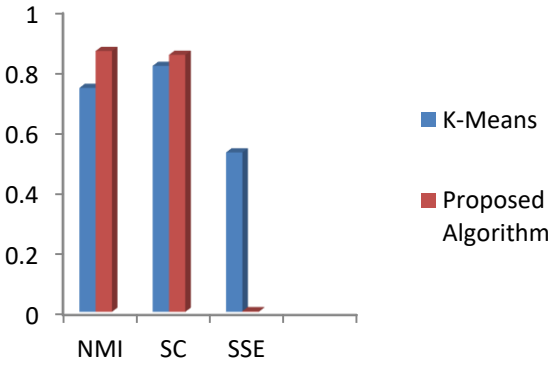

Figure 2. The quality of clustering for cluster 3

In Table 3 and Figure 3 shows the comparison of proposed algorithm and k-means with the quality of clustering for cluster four.

Table 3. Evaluation result of Iris data for the quality of clustering, k = 4

Number of Clusters	Method	Normalized Mutual Information (NMI)	Silhouette Coefficient (SC)	Sum of Square Error (SSE)
K = 4	K-Means	0.7006	0.681	0.409
	Proposed	0.8	0.9141	0.0019

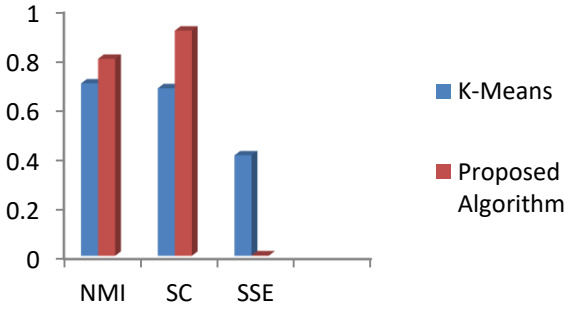

Figure 3. The quality of clustering for cluster 4

The comparison of proposed algorithm and k-means with the quality of clustering for cluster five. The result is shown in Figure 4 and Table 4.

Table 4. Evaluation result of Iris data for the quality of clustering, k = 5

Number of Clusters	Method	Normalized Mutual Information (NMI)	Silhouette Coefficient (SC)	Sum of Square Error (SSE)
K = 5	K-Means	0.6939	0.6591	0.3167
	Proposed	0.7036	0.5604	0.0015

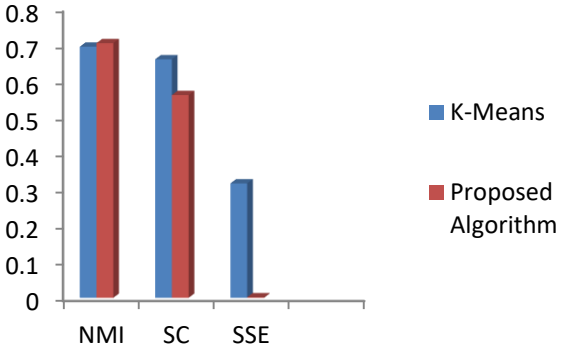

Figure 4. The quality of clustering for cluster 5

6. CONCLUSIONS

K-means is good clustering algorithm but k-means algorithm does not always generate good quality results as automatic initialization of centroids affects final clusters. The proposed algorithm is found to be more accurate and efficient compared to the original k-means algorithm. This proposed method finding the better initial centroids and provides an efficient way of assigning the data to the suitable clusters. The computational complexity of the standard k-means algorithm is high than this proposed k-means algorithm. This system is done by assigning weights to each attribute value to achieve standardization. This algorithm has proved to be better than standard k-means algorithm in terms of cluster quality.

7. REFERENCE

[1] Vaishali R. Patel, Rupa G. Mehta, "Impact of Outlier Removal and Normalization Approach in Modified k-Means Clustering Algorithm", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 2, 2011, pp. 331-336.

[2] R. Agrawal, T. Imielinksi and A. Swami, "Mining association rules between sets of items in large database", The ACM SIGMOD Conference, Washington DC, USA, 1993, pp. 207-216.

[3] David Arthur & Sergei Vassilvitskii , "How Slow is the k means Method?", Proceedings of the 22nd Symposium on Computational Geometry (SoCG), 2006, pp. 144-153.

[4] A. Joy Christy, S. Hari Ganesh, "Building Numerical Clusters Using Multidimensional Spherical Equation", International Journal of Applied Engineering Research, ISSN 0973-4562, Volume 10, Issue No.82, pp:629-634, 2015.

[5] Md Sohrab Mahmud, Md. Mostafizer Rahman, Md. Nasim Akhtar, "Improvement of K-means Clustering algorithm with better initial centroids based on weighted average", 7th International Conference on Electrical and Computer Engineering, 2012, pp. 647-650.

[6] Chen Zhang and Shixiong Xia, K-means 2009. Clustering Algorithm with Improved Initial center, in Second International Workshop on Knowledge Discovery and Data Mining (WKDD), pp: 790-792.

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

[8] 2010. The UCI Repository website [Online]. Available: http://archive.ics.edu/

[9] Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics. Vol.20, pp.53-65.

[10] Kwedlo, W. (2011). A clustering method combining differential evolution with the k-means algorithm. Pattern Recognition Letters. Vol.32, pp.1613–1621.

[11] F. Knops, Zeger & B. Antoine Maintz, J & A. Viergever, Max & P. W. Pluim, Josien. (2003). Normalized Mutual Information Based PET-MR Registration Using K-Means Clustering and Shading Correction. 2717. 31-39.

[12] J. Han and M. Kamber, Data Mining Concepts and Techniques, Morgan Kaufmann Publishers, San Diego, 2001.

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

[14] Navdeep Kaur and Krishan Kumar, "Normalization Based K-means Data Analysis Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, Volume 6, Issue, June 2016, pp. 455-457

[15] R. Praveena Priyadarsini, M.L.Valarmathi and S. Sivakumari, "Gain Ratio Based Feature Selection Method For Privacy", ICTACT Journal on Soft Computing, April 2011, Volume: 01, Issue: 04, pp. 201-205