

Performance Enhancement in Software applications and Web applications

Amit Behl, Archana Behl
Infosys Technologies Limited, Mysore (India)

Abstract

Most of the application software's and web applications deals with Data storage and Retrieval of huge amount of data.

This paper outlines the need of storing data in way that lead to efficient retrieval of data. As a result software /Web application performance can be improved. If data is stored in clusters then retrieval operations can be much faster as compared to traditional approach. We will analyze some of the clustering techniques and efficiency of these techniques with sample data.

I. Introduction

In clustering, the objects having similar properties can be placed in one group. Single disc access to that group will makes the entire data for that group available. The goal of clustering is grouping of the similar objects based on some criteria. There are no such preminent criteria for making clusters. Depending on problem in hand and customer choice, criteria can be decided.

Partitioning of data objects in to clusters can be done in two ways:

1. Hard partitioning
2. Fuzzy partitioning

A. Hard Partitioning

In hard Partitioning, data is divided in to fixed partitions/clusters. Using classical sets, a hard partition can be defined as a family of subsets $\{A_i | 1 \leq i \leq c \subset P(X)\}$ its properties are as follows:

$$\bigcup_{i=1}^c A_i = X,$$

$$A_i \cap A_j = \phi, 1 \leq i \neq j \leq c,$$

$$\phi \subset A_i \subset X, 1 \leq i \leq c.$$

These conditions mean that the subsets A_i contain all the data in X , they must be disjoint and none of them is empty nor contains all the data in X . Expressed in the terms of membership functions:

$$\bigvee_{i=1}^c \mu_{A_i} = 1,$$

$$\mu_{A_i} \vee \mu_{A_j}, 1 \leq i \neq j \leq c,$$

$$0 < \mu_{A_i} < 1, 1 \leq i \leq c.$$

Here μ_{A_i} is the characteristic function of the subset A_i and its value can be zero or one. To simplify the notations, we use μ_i instead of μ_{A_i} , and denoting $\mu_i(x_k)$ by μ_{ik} , partitions can be represented in a matrix notation.

An $N \times c$ matrix $U = [\mu_{ik}]$ represents the hard partition if and only if its elements satisfy:

$$\mu_{ik} \in \{0, 1\}; 1 < i < N; 1 \leq k \leq c,$$

$$\sum_{k=1}^c \mu_{ik} = 1, 1 \leq i \leq N,$$

$$0 < \sum_{i=1}^N \mu_{ik} < N, 1 \leq k \leq c.$$

B. Fuzzy Partitioning

It's like simplification of hard partition, it allows μ_{ik} attaining real values in $[0, 1]$. An $N \times c$ matrix $U = [\mu_{ik}]$ represents the fuzzy partitions, its conditions are given by:

$$\mu_{ik} \in [0, 1]; 1 < i < N; 1 \leq k \leq c,$$

$$\sum_{k=1}^c \mu_{ik} = 1, 1 \leq i \leq N,$$

$$0 < \sum_{i=1}^N \mu_{ik} < N, 1 \leq k \leq c.$$

II. Validity measures

The clustering algorithms always try to find out the best combination for a fixed number of clusters. We can compare the performance of an algorithm using different validity indexes.

Validity indexes are defined below:

A. Partition Coefficient (PC)

Partition Coefficient (PC) measures the amount of overlapping between clusters. It is described below:

$$PC(c) = \frac{1}{N} \sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^2$$

Where μ_{ij} is the membership of data point j in cluster i . The disadvantage of PC is lack of direct connection to some property of the data themselves. The optimal number of cluster is at the maximum value.

B. Classification Entropy (CE)

It measures the fuzziness of the cluster partition only, which is similar to the Partition Coefficient

$$CE(c) = \frac{1}{N} \sum_{i=1}^c \sum_{j=1}^N \mu_{ij} \log(\mu_{ij})$$

C. Partition Index (SC)

Partition Index is the ratio of the sum of compactness and separation of the clusters. It is a sum of individual cluster validity measures normalized through division by the fuzzy cardinality of each cluster.

$$SC(c) = \sum_{i=1}^c \frac{\sum_{j=1}^n (\mu_{ij})^m \|x_j - v_i\|^2}{N \sum_{k=1}^c \|v_k - v_i\|^2}$$

D. Separation Index (S)

Separation Index on the contrary of partition index (SC), the separation index uses a minimum-distance separation for partition validity.

$$S(c) = \frac{\sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^2 \|x_j - v_i\|^2}{N \min_{i,k} \|v_k - v_i\|^2}$$

E. Xie and Beni's Index (XB)

$$XB(c) = \frac{\sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^m \|x_j - v_i\|^2}{N \min_{ij} \|x_j - v_i\|^2}$$

The optimal number of clusters should minimize the value of the index.

F. Dunn's Index (DI)

$$DI(c) = \min_{i \in c} \left\{ \min_{j \in c, i \neq j} \left\{ \frac{\min_{x \in C_i, y \in C_j} d(x, y)}{\max_{k \in c} \left\{ \max_{x, y \in C} d(x, y) \right\}} \right\} \right\}$$

G. Alternative Dunn Index (ADI)

$$ADI(c) = \min_{i \in c} \left\{ \min_{j \in c, i \neq j} \left\{ \frac{\min_{x_i \in C_i, x_j \in C_j} |d(y, v_j) - d(x_i, v_j)|}{\max_{k \in c} \left\{ \max_{x, y \in C} d(x, y) \right\}} \right\} \right\}$$

III. Clustering algorithms

This paper discusses the implementation & comparison of two hard clustering algorithms:

1. K-Means
2. K-Mediod and One fuzzy clustering algorithm

1. Fuzzy C-Means (FCM)

A. K-Means and K-Mediod Algorithms

The hard partitioning methods are simple and popular, though its results are not always reliable and these algorithms have numerical problems also. From an $N \times n$ dimensional data set K-means and K-medoid algorithms allocates each data point to one of c clusters to minimize the within-cluster sum of squares:

$$\sum_{i=1}^c \sum_{k \in A_i} \|X_k - V_i\|^2$$

where A_i is a set of objects (data points) in the i -th cluster and v_i is the mean for that points over cluster i denotes actually a distance norm. In K-means clustering v_i is called the cluster prototypes, i.e. *the cluster centers*:

$$v_i = \frac{\sum_{k=1}^{N_i} X_k}{N_i}, X_k \in A_i,$$

Where N_i is number of objects in A_i ,

B. Fuzzy C-Means Algorithm

The Fuzzy C-means clustering algorithm is based on the minimization of an objective function called **C-means functional**. It is defined by Dunn as:

$$J(X; U, V) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m \|X_k - V_i\|_A^2$$

where

$$V = [v_1, v_2, \dots, v_c], v_i \in R^n$$

is a vector of *cluster prototypes* (centers), which have to be determined, and

$$D_{ikA}^2 = \|x_k - v_i\|_A^2 = (x_k - v_i)^T A (x_k - v_i)$$

is a squared inner-product distance norm.

The FCM algorithm computes with the standard Euclidean distance. Hence it can only detect clusters with the same direction and shape, because the common choice of norm inducing matrix is: $A = I$ or it can be chosen as an $n \times n$ diagonal matrix that accounts for different variances in the directions in the directions of the coordinate axes of X :

$$A_D = \begin{bmatrix} (1/\sigma_1)^2 & 0 & \dots & 0 \\ 0 & (1/\sigma_2)^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & (1/\sigma_n)^2 \end{bmatrix}$$

or A can be defined as the inverse of $n \times n$ covariance matrix: $A = F^{-1}$ with

$$F = \frac{1}{N} \sum_{k=1}^N (X_k - \bar{X})(X_k - \bar{X})^T$$

Here, \bar{X} denotes the sample mean of the data.

IV. Validity parameters in k-means

As already discussed in no way one parameter could be regarded as sufficient to approve or perfectly be regarded as a perfect index for the validity of clustering. So we will measures the validity of the K-means algorithm with various indexes.

Table 1: Validity parameters for K-means

Clusters	PC	SC	S	XB	DI	ADI
2	1	1.0739	0.002	3968	0.0534	0.002
3	1	2.100	0.0014	∞	0.084	0.0012
4	1	1.410	0.002	∞	0.021	0.0101
5	1	1.123	0.01	∞	0.0403	0.0021
6	1	1.030	0.002	∞	0.0203	0.008
7	1	0.80	0.0013	∞	0.028	0
8	1	0.629	0.00175	∞	0.029	0
9	1	0.541	0.0015	∞	0.037	0
10	1	0.657	0.0010	∞	0.029	0

Table 1, shows the numeric validity measures of the all the seven indexes. Values of the indexes are for 2 clusters to 10 clusters. Each column shows the values of a particular Index, for all the clusters (from 2 to 10). Example column number 2 show the values of Partition Coefficient (PC) for numbers of clusters 2 to 10 and similarly column number 3 show the values of Partition Index (SC) for numbers of clusters 2 to 10.

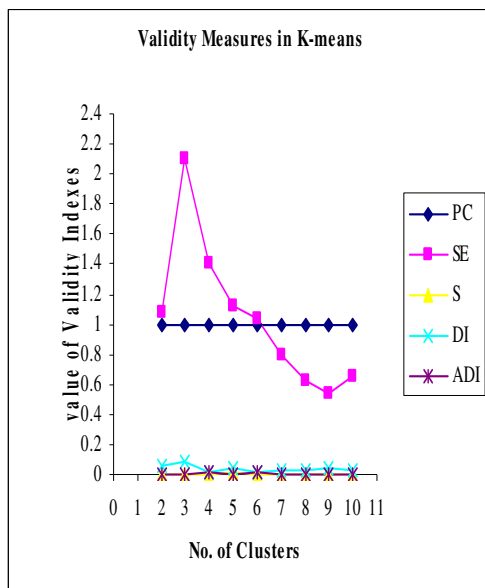


Figure 1: Variation of validity indexes with respect to number of clusters for K-means

Figure 1, shows the variation of validity indexes for K-means with respect to number of clusters, X-axis

represents number of clusters and Y-axis represents the values of the validity indexes.

V. Validity parameters in k-mediod

We will measures the validity of the K-mediod algorithm with seven most popular indexes. As already discussed in no way one parameter could be regarded as sufficient to approve or perfectly be regarded as a perfect index for the validity of clustering.

Comparing the efficiency of the algorithm on different indexes will be a better for the judgment of the effectiveness of the algorithm.

Table 2: Validity parameters for K-Mediod clustering

Clusters	PC	SC	S	XB	DI	ADI
2	1	2.0867	0.0035	Inf	0.0335	0.0672
3	1	2.1182	0.0054	Inf	0.0383	0.0124
4	1	1.5116	0.0038	Inf	0.0191	0.0131
5	1	1.2565	0.0031	Inf	0.0203	0.0021
6	1	1.0310	0.0029	Inf	0.0203	0.0038
7	1	0.8330	0.0023	Inf	0.0268	0
8	1	0.7929	0.0020	Inf	0.0294	0
9	1	0.5180	0.0015	Inf	0.0311	0
10	1	0.6588	0.0019	Inf	0.0248	0

Table 2 shows the numeric validity measures of the all the seven indexes for K-Mediod. Values of the indexes are for 2 clusters to 10 clusters.

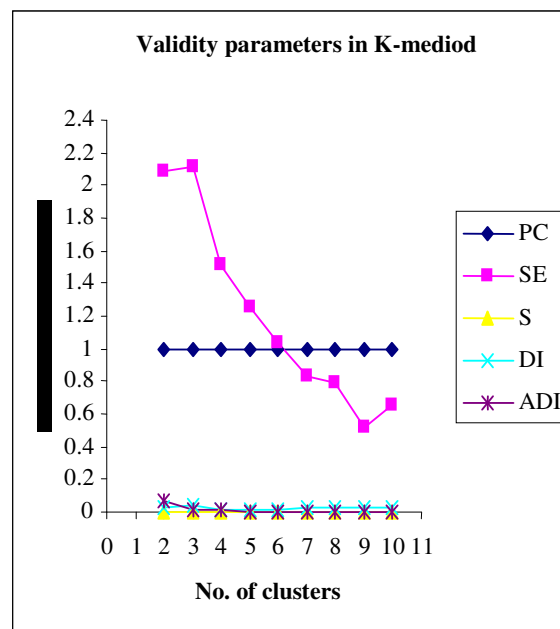


Figure 2: Variation of partition index with respect to number of clusters for K-Mediod

Figure 2, shows the variation of validity indexes for K-Mediod with respect to number of clusters, X-axis represents number of clusters and Y-axis represents the values of the validity indexes

VI. Validity parameters in fcm

Comparing the efficiency of the algorithm on different indexes will provide a better judgment of the effectiveness of the algorithm.

Table 3: Validity parameters for FCM

Clusters	PC	SC	S	XB	DI	ADI
2	0.8147	1.7875	0.0030	4.2471	0.0168	0.0645
3	0.7053	2.0172	0.0054	5.1938	0.0203	0.0050
4	0.6390	1.8297	0.0047	3.3675	0.0190	0.0037
5	0.5778	1.3992	0.0041	3.1252	0.0260	6.7162e-004
6	0.5795	1.5659	0.0035	3.7140	0.0248	0.0011
	0.5690	1.1813	0.0029	3.2661	0.0278	0
8	0.5883	1.1010	0.0028	3.0105	0.0275	0
9	0.5812	1.1302	0.0031	3.2557	0.0275	0

Table 3, shows the numeric validity measures of the all the seven indexes for FCM. Values of the indexes are from 2 clusters to 9 clusters. Each column shows the values of a particular Index, for all the clusters (from 2 to 9). Example column number 2 shows the values of Partition Coefficient (PC) for numbers of clusters 2 to 9.

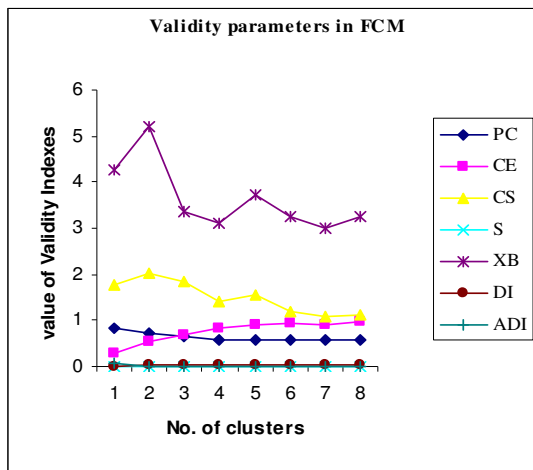


Figure 3: Variation of partition index with respect to number of clusters for FCM

Figure 3, a plot showing the variation of validity indexes for FCM with respect to number of clusters, X-axis represents number of clusters and Y-axis represents the values of the validity indexes. Each validity index is shown with a different color to differentiate it from others indexes in the figure.

Table 4: Comparison of various parameters of K-Mean,

Clus Algo	PC	SC	S	XB	DI	ADI
K-means	1	1.411	0.002	Inf	0.021	0.0101
K-medoid	1	1.5116	0.0038	Inf	0.0191	0.0131
FCM	0.6390	1.8297	0.0047	3.3675	0.0190	0.0037

K-Mediod and FCM for 4 clusters

In this indexes are represented in columns and clustering type is represented in rows.

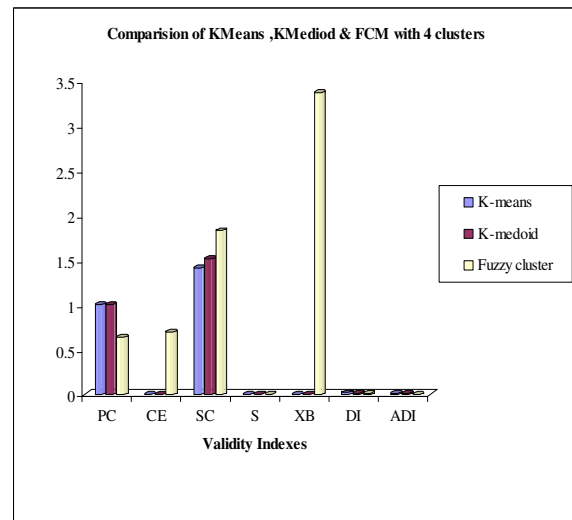


Figure 4, Shows the comparison of KMeans, KMediod & FCM with 4 clusters.

X-axis represents Validity Indexes and Y-axis represents the values of the validity indexes. Each Clustering algorithm is shown with a different color.

- Value of Partition coefficient (PC) for KMeans is 1 , KMediod is 1 & for FCM is 0.6390
- Value of Partition Index (SC) for KMeans is 1.141 , KMediod is 1.5116 & for FCM is 1.8297
- Value of Dunn Index (DI) for KMeans is 0.021 , KMediod is 0.0191 & for FCM is 0.0190
- Value of Separation Index (S) for KMeans is 0.002, KMediod is 0.0038 & for FCM is 0.0047

- Value of Alternative Dunn Index (ADI) for KMeans is 0.101 , KMediod is 0.0131 & for FCM is 0.0037
 - Value of Classification Entropy (CE) for KMeans is - , KMediod is - & for FCM is 0.6965
 - Value of Xie and Beni's Index (XB)for KMeans & KMediod is infinity, 3.3675 for FCM.
- On the score of the values of the two "most popular and used" indexes for fuzzy clustering (Partition Coefficient and Xie and Beni's Index) the fuzzy clustering has the very best results for this data set.
 - Some pattern which cannot be separated by hard clustering algorithms can be separated by fuzzy clustering algorithms.
 - Hybrid approach can be used to exploit the features of both hard as well as fuzzy clustering and to increase the efficiency of decision making.

VII. References

- [1] Rob Short, Rod Gamache, John Vert and Mike Massa "Windows NT Clusters for Availability and Scalability", Microsoft Online Research Papers, Microsoft Corporation, 1998.
- [2] M. Aladjem, I. Dinstein and B. Lerner, "On patterns classification with sammon's nonlinear mapping - an experimental study", Pattern Recognition Letters, vol. 4, pp.371-381, 1998.
- [3] Miyamoto, S, "An overview and new methods in fuzzy clustering", Proceedings KES '98, 1998 Second International Conference, vol. 1, pp. 33-40, 1998.
- [4] Kreinovich, V. Nguyen, H.T. Starks, S.A. Yeung Yam, " Decision making based on satellite images: optimal fuzzy clustering approach", Proceedings of the 37th IEEE Conference, vol. 4, pp. 4246-4251. 1998.
- [5] Tai Wai Cheng Dmitry B. Goldgof and Lawrence O. Hall, "Fast fuzzy clustering, Fuzzy Sets and Systems" , Vol. 93 , pp.1145-1147, January 1998.
- [6] Altman, D., "Efficient fuzzy clustering of multi-spectral images", IGARSS '99 Proceedings, IEEE 1999 International, vol. 3, pp. 1594-1596,1999.
- [7] M. E. Tipping and C. M. Bishop, " Mixtures of probabilistic principal components analysis", Neural Computation, vol. 11, pp.443-482, 1999.
- [8] Russell, S. and Lodwick, W., " Fuzzy clustering in data mining for telco database marketing campaigns", NAFIPS. 18th International Conference of the North American, pp. 720-726. 1999.
- [9] Steven Schockaert, Martine De Cock, and Etienne E. Kerre,"Automatic Acquisition of Fuzzy Footprints", FUZZ IEEE 2000. The Ninth IEEE International Conference, vol. 1, pp. 176-180, 2000.
- [10] Kraft, D.H., and Chen, J. Mikulcic, A. , "Combining fuzzy clustering and fuzzy inferencing in information retrieval", FUZZ IEEE 2000. The Ninth IEEE International Conference, vol. 1, pp. 375-380, 2000.
- [11] Yiping Liu, Yi Shen , Zhiyan Liu , "An approach to fault diagnosis for non-linear system based on fuzzy cluster analysis", Instrumentation and Measurement Technology Conference, IMTC 2000, vol.3, pp. 1469-1473, 2000.
- [12] Nefti, S. and Oussalah, "Probabilistic-fuzzy clustering algorithm Systems, Man and Cybernetics", 2004 IEEE International Conference, vol. 5, pp. 4786-4791, 2004
- [13]. Bandyopadhyay, S, "Simulated annealing using a reversible jump Markov chain Monte Carlo algorithm for fuzzy clustering", IEEE Transactions, vol. 5, pp. 479-490, 2005