International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-10, August 2019

HYBCIM: Hypercube Based Cluster Initialization Method for k-means

Manoj Kumar Gupta, Pravin Chandra

Abstract*: Clustering is a data processing technique that is extensively used to find novel patterns in data in the field of data mining and also in classification techniques. The k-means algorithm is extensively used for clustering due to its ease and reliability. A major effect on the accuracy and performance of the k-means algorithm is by the initial choice of the cluster centroids. Minimizing Sum of Squares of the distance from the centroid of the cluster for cluster points within the cluster (SSW) and maximizing Sum of Square distance between the centroids of different clusters (SSB) are two generally used quality parameters of the clustering technique. To improve the accuracy, performance and quality parameters of the k-means algorithm, a new Hypercube Based Cluster Initialization Method, called HYBCIM, is proposed in this work. In the proposed method, collection of k equi-sized partitions of all dimensions is modeled as a hypercube. The motivation behind the proposed method is that the clusters may spread horizontally, vertically, diagonally or in arc shaped. The proposed method empirically evaluated on four popular data sets. The results show that the proposed method is superior to basic k-means. HYBCIM is applicable for clustering both discrete and continuous data. Though, HYBCIM is proposed for k-means but it can also be applied with other clustering algorithms which are based on initial cluster centroids.

***Index Terms*: Clustering; k-means; Cluster Initialization; Hypercube Based Cluster Initialization Method; Unsupervised Learning.**

I. INTRODUCTION

Data mining is used to discover novel, non-trivial and potentially useful pattern from data [1, 2]. Clustering is an unsupervised learning based function of data mining which partition data objects into subsets called as clusters. In other words, it partition or segment the data in to different groups based on distance or (dis)similarity among the data objects. The objects of cluster are like / near to each other whereas dissimilar / far off with the objects of the other clusters [3, 4, 5]. The clusters are demarcated on the basis of the study of the behavior / characteristics of the data objects by domain experts as well as by the various clustering methods. Cluster analysis is one of the most popular techniques which is not only used in data mining but also extensively used in lots of domains such as statistics, information retrieval, pattern recognition, object recognition, image segmentation, image bioinformatics, etc. [1, 6]. A range of clustering algorithms is proposed by many researchers in the literature [1, 5, 7].

K-means is widely used for identification of clusters in numerous applications. K-means is considered to be the simplest and efficient clustering algorithms [7, 8, 9, 10]. In k-means, *k* represents the number of clusters in which the data objects needs to be grouped. The steps of k-means algorithm are (i) decide the number of clusters (*k*), (ii) choose *k* random data points as initial cluster centroids, (iii) compute distance of each data point with all cluster centroids and assign to the nearest one, (iv) re-compute cluster centroids, (v) repeat step 3 and 4 until cluster membership stabilizes. The psuedocode of basic k-means is presented as Algorithm 1 [1]:

Algorithm 1: Basic k-means

Step 1: *Decide k (no. of clusters)*
Step 2: *Randomly initialize cluster centroids* $C = \{c_1, c_2, \ldots, c_k\}$
Step 3: *Repeat*
- *a. For each data point* (x_i) *in data set (D)*
  - *i. Compute distance* $dis(x_i, C)$ *between* x_i *and all cluster centroids*
  - *ii. Assign* x_i *to the nearest cluster*
- *b. Re-compute cluster centroids as the mean of all cluster members.*

Step 4: *Until cluster membership stabilizes.*

Every run of k-means results in formation of dissimilar sets of clusters with different degree of accuracy and performance because of the arbitrary selection of initial cluster centroids. Hence, accuracy and performance of k-means is majorly depends on initial cluster centroids. Minimization of *SSW* and maximization of *SSB* is the main objective of the clustering algorithms. In view of this, efforts to improve k-means have been made by many researchers. In view of this, a number of cluster initialization methods are proposed in the literature [5, 6, 7].

Forgy [11] proposed the earliest method to initialize k-means in which centroids are selected purely on random basis. Later, McQueen [12] proposed a method similar to the Forgy's Approach but differs in assigning the left over objects to one of the close seed location. Based on the centrally located instance, Kaufman and Rousseeeuw [13] proposed a method for cluster initialization. Katsavounidis et al. [14] proposed method based on the selection of furthest points as initial centroids. Bradley and Fayyad [15] proposed a technique in which the data is randomly broken into the *J* random small sub-subsets and then the initial points are selected.

Manuscript published on 30 August 2019.
***Correspondence Author(s)**
Manoj Kumar Gupta, Research Scholar, USIC&T, Guru Gobind Singh Indraprastha University, Delhi.
Pravin Chandra, Professor, USIC&T, Guru Gobind Singh Indraprastha University, Delhi, India.

A new method named as Cluster Center Initialization Algorithm (CCIA) proposed in [16] for finding initial cluster centroids using the concept of Density-based Multi Scale Data Condensation (DBMSDC). Su and Dy [17] proposed technique based on deterministic divisive hierarchical method for centroid initialization. Arthur and Vassilvitskii [18] proposed a new method named as k-means++ in which initial centroids are chosen one after the other with probability relative to the distance to the nearest centroid. Arai and Barakbah [19] proposed an approach based on transformation of the result by combining with Hierarchical algorithm. Based on finding a large number of local modes, a staged approach is proposed in [20] to specify initial centroids. Naldi et al. [21] suggested the methods based on evolutionary techniques. Initialization method based on iterative selection for k-means is proposed in [22]. For document clustering, Sandhya and Sekar [23] proposed three different approaches for centroid initialization. Automatic Clustering Using Teaching–Learning-based optimization (TLBO) is introduced in [24]. In Section 2 of this paper, a new cluster initialization method named as Hypercube Based Cluster Initialization Method for k-means is proposed to improvise the performance and accuracy of basic k-means. Experiment based on basic k-means and HYBCIM is carried out using four popular datasets in MATLAB. The results of experiment are presented in Section 3. Lastly, conclusion is drawn in Section 4.

II. HYPERCUBE BASED CLUSTER INITIALIZATION METHOD (HYBCIM)

In order to improve the accuracy, performance and objective functions (i.e. *SSW* and *SSB*) of k-means, a new method to initialize the cluster centroids is devised and proposed, named as Hypercube Based Cluster Initialization Method (HYBCIM), in this paper. In the proposed method, the range of each dimension (or attribute), dim_i, of the data set is logically divided in k equi-sized partitions where k is the number of clusters. These collection of k equi-sized partitions of all dimensions is modeled as a hypercube of k^d (i.e. $k_1 \times k_2 \times \ldots \times k_d$, where d is the number of dimensions). E.g. a cube of 3 dimensions with $k=3$ partitions each is presented in *Figure 1*. k unique cells are randomly selected as k centroids for k-means from this hypercube. The motivation behind the proposed method is that the clusters may spread horizontally, vertically, diagonally or in arc shaped. Therefore, to guess the centroids randomly from these cells will be more near to the actual cluster centroids. Hence, the accuracy, performance and objective functions of k-means will be improved. Step 2 of the basic k-means is modified in the proposed method. The psuedocode of the proposed method is presented as Algorithm 2:

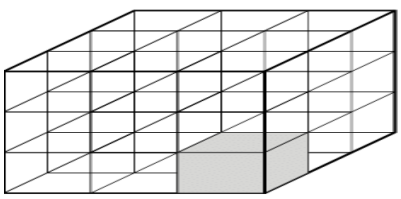

Figure 1. Representation of a Hypercube with 3 dimensions and 3 partitions of each dimension

Algorithm 2: Hypercube Based Cluster Initialization Method (HYBCIM)

Step 1: *Decide k (# of clusters)*

// initialize k cluster centroids as per Steps 2.1 through Step 2.4

Step 2: *Initialize cluster centroids $C = \{c_1, c_2, \ldots, c_k\}$ as:*

// range of values of each dimension (i.e. attribute) is logically divided in k equal sized partitions based on arithmetic average of the respective attributes

Step 2.1: *Divide the range of values of data of each dimension, dim_i, into k equi-ranged partitions.*

// logically model the partitions of each dimension as hypercube as presented in Fig. 1

Step 2.2: *Consider these partitions as $k_1 \times k_2 \times \ldots \times k_d$ hypercube (or k^d hypercube) where d is the number of Dimensions.*

// randomly select k unique cells from the hypercube and then choose a random value from each chosen k cells as k initial centroids

Step 2.3: *Repeat*

i. *Arbitrarily choose one cell, which was not selected earlier, from the hypercube.*
ii. *Find the randomized value of each cell selected for centroid.*

Step 2.4: *If all centroids are chosen then go to Step 3 else go to Step 2.3*

// find out the cluster membership of each data point iteratively until cluster membership stabilizes

Step 3: *Repeat*

a. *For each data point (x_i) in data set (D)*
   i. *Compute distance dis(x_i, C) between x_i and all cluster centroids*
   ii. *Assign x_i to the nearest cluster*
b. *Re-compute cluster centroids as the mean of all cluster members.*

Step 4: *If cluster membership stabilizes then end else go to Step 3.*

As per step 2(a), the range of values of each dimension dim_i is equally divided into k partitions. These k-partitions of each dimension is modeled as a hypercube with the dimension d and each dimension is partitioned into k parts (Step 2(b)). As per step 2(c), k cells from the hypercube are arbitrarily chosen as k initial centroids such that the same cell will not be repeated. The proposed method ensures that the two or more initial centroids are not chosen from the same cell which is possible in basic k-means.

III. THE EMPIRICAL RESULTS

Basic k-Means and HYBCIM are implemented and executed in MATLAB. Both methods are executed on four popular datasets. The results are computed and compared based on the average of 200 runs of each of the methods on each of the four datasets mentioned in *Table 1*. The ground truth of each dataset is not used during the evaluation. It is used to compute the accuracy by comparing the deviation of the cluster assignment given by each of the methods.

A. Dataset used

For the purpose of empirical evaluation, both methods are evaluated on four different datasets Pen Digit, Iris, Animal Milk and Wine. Animal Milk dataset taken from Hartigan

Retrieval Number: J97740881019/19©BEIESP
DOI: 10.35940/ijitee.J9774.0881019
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

(https://people.sc.fsu.edu/~jburkardt/datasets/hartigan/file02.txt) whereas rest datasets taken from UCI. The detail of the datasets used in the experiment is presented in *Table 1*.

Table 1. Datasets Used

Dataset	# of Instances	# of Attributes	# of Clusters
Pen Digit	7494	16	10
IRIS	150	4	3
Animal Milk	16	4	5
Wine	178	13	3

B. Metric

Both basic k-Means and HYBCIM are implemented and tested using MATLAB. The average of 200 runs of each of these methods on each of the above mentioned four datasets are taken and compared. The implementation is the standard one with no special optimizations.

C. Results and Discussion

The comparative evaluation of basic k-means and HYBCIM for each of above mentioned four datasets are presented in the *Table 2* through *Table 5*. Accuracy of both methods is presented in *Table 2*. *Table 3* presents the performance of both methods. *SSW* and *SSB* are presented in *Table 4* and *Table 5* respectively.

Table 2 shows that the accuracy of HYBCIM is higher than that of basic k-means for all data sets except Animal Milk data set. As compared to basic k-means, HYBCIM converges faster in the case of Pen Digit and IRIS data sets as presented in Table 3. Table 4 shows that *SSW* given by HYBCIM is less than that of basic k-means for all data sets except Wine data set. *SSB* given by HYBCIM is also greater than basic k-means in case of two data sets IRIS and Wine as presented in Table 5.

Table 2. Accuracy of the Cluster Assignment

Dataset	Basic k-means	HYBCIM
Pen Digit	75.89%	75.90%
IRIS	88.81%	89.07%
Animal Milk	96.96%	96.64%
Wine	71.70%	72.05%

Table 3. Average # of Iterations taken to Converge (Performance)

Dataset	Basic k-means	HYBCIM
Pen Digit	28.65	27.67
IRIS	9.22	8.91
Animal Milk	38.42	38.72
Wine	11.87	12.89

Table 4. Sum of Squares within Clusters (SSW)

Dataset	Basic k-means	HYBCIM
Pen Digit	3510918.33	3508978.86
IRIS	27.14	26.71
Animal Milk	7.67	7.38
Wine	841436.80	851846.41

Table 5. Sum of Squares between Clusters (SSB)

Dataset	Basic k-means	HYBCIM
Pen Digit	109851.99	108853.51
IRIS	13.05	13.14
Animal Milk	682.73	680.39
Wine	296495.06	299972.64

IV. SUMMARY AND CONCLUSION

Basic k-means is widely used due to its simplicity. There is no complexity is involved in initializing the cluster centroids randomly. But, the accuracy and performance of k-means is sometimes extremely affected due to the initial cluster centroids. Hence, careful selection of initial cluster centroids is desired. A new method of initialization of the cluster centroids is proposed in this paper called Hypercube Based Cluster Initialization Method (HYBCIM). In HYBCIM, the dimensions of the data are partitioned in such a manner that if '*d*' is the dimensionality of data, then k^d (where *k* is the number of desired clusters) hypercube are created. Out of these k^d cubes, the centroids for initialization of the k-means algorithm are chosen in a random manner. This ensures that, two initial cluster centroids at least differ by the hypercube, that is, no two centroids of the two clusters are in the hypercube. The empirical results show that there is improvement in accuracy and performance of the clustering generated using HYBCIM as compared to basic k-means. The objective functions i.e. *SSW* and *SSB* of clustering generated through HYBCIM are also better as compared to basic k-means. The proposed HYBCIM is applicable for clustering both discrete and continuous data. Though, HYBCIM is proposed for k-means but it can also be applied with other clustering algorithms which are based on initial cluster centroids. HYBCIM can be further optimized using nature inspired algorithms.

REFERENCES

1. Han J, Kamber M, Pei J (2012) 'Data mining concepts and techniques', Elsevier, 3rd Edition.
2. Arora, R. and Gupta, M. (2017) "e-Governance using Data Warehousing and Data Mining", International Journal of Computer Applications, Volume 169 - No.8, July 2017.
3. Jain, A. K. and Dubes, R. C. (1988) 'Algorithms for Clustering Data'. Prentice Hall, Englewood Cliffs, NJ.
4. Xu, J., Xu, B., Zhang, W., Zhang, W., Hou, J. (2009) "Stable Initialization Scheme for K-Means Clustering", Wuhan University Journal of Natural Science, Vol. 14, No. 1, pp 24-28
5. Gupta, M.K. and Chandra, P (2019), A Comparative Study of Clustering Algorithms, In Proc. of the 13th INDIACom-2019; IEEE Conference ID: 461816; 6th International Conference on "Computing for Sustainable Global Development".
6. Jain, A.K., Murty, M.N. and Flynn, P.J. (1999) 'Data clustering: a review' ACM Comput. Surv. 31, 3, 60 pages.
7. Jain, A.K. (2010), 'Data clustering: 50 years beyond K-means', Pattern Recognition Letters, Elsevier, vol. 31, pp. 651-666.
8. Aldahdooh, R.T. and Ashour, W. (2013), 'DIMK-means "Distance-based Initialization Methods for K-means Clustering Algorithms", I.J. Intelligent Systems and Applications, Vol. 2. PP 41-51.

9. Gan, G., Ma, C., and Wu, J., (2007) 'Data Clustering: Theory, Algorithms, and Applications', American Statistical Association and the Society for Industrial and Applied Mathematics, SIAM.
10. Gupta, M.K. and Chandra, P (2019), An Empirical Evaluation of k-means Clustering Algorithm using Different Distance/Similarity Metrics, In Proc. of the International Conference on Emerging Trends in Information Technology (ICETIT-2019), Springer.
11. Forgy E. (1965) "Cluster Analysis of Multivariate Data: Efficiency vs. Interpretability of Classifications" [J]. Biometrics, 1965, 21(3): 768.
12. McQueen, J.B. (1967), 'Some methods for classification and analysis of multi-variate observation' Symposium on Mathematical Statistics and Probability, University of California Press.
13. Kaufman, L. and Rousseeuw, P.J. (1990), 'Finding Groups in Data. An Introduction to Cluster Analysis' Wiley, Canada.
14. Katsavounidis, I, Kuo, C. Zhang, Z. (1994), 'A new initialization technique for generalized Lloyd iteration', IEEE, 1(10), 144-146.
15. Bradley, P.S. and Fayyad (1998), 'Refining initial points for K-Means clustering', Proc. 15th Intl. Conf. on Machine Learning, San Francisco, CA, pp 91-99
16. Khan, S.S. and Ahmad, A. (2004), 'Cluster Centre Initialization Algorithm for k-means clustering', Pattern Recognition Letters 25(11), pp 1293-1302.
17. Su, T. and Dy, J. (2004) "A Deterministic Method for Initializing K-means Clustering" Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference, pp. 784 - 786, Nov 2004.
18. Arthur, D. and Vassilvitskii, S. (2007) "k-means++: The advantages of careful seeding," ACM-SIAM Symposium on Discrete Algorithms (SODA 2007) Astor Crowne Plaza, New Orleans, Louisiana, pp. 1-11.
19. Arai, K. and Barakbah, A.R. (2007) "Hierarchical K-means: an algorithm for centroids initialization for K-means" Rep. Fac. Sci. Engrg, Saga Univ. , vol. 36.
20. Maitra, R. (2009) "Initializing partition-optimization algorithms," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 6, pp. 144-157.
21. Naldi, M.C., Campello, R.J.G.B., Hruschka, E.R. and Carvalho, A.C.P.L.F. (2011) "Efficiency issues of evolutionary k-means," Applied Soft Computing, vol. 11 , pp. 1938-1952.
22. Poomagal, S., Saranya, P., Karthik, S. (2016), A novel method for selecting initial centroids in K-means clustering algorithm, International Journal of Intelligent Systems Technologies and Applications, Volume 15, Issue 3, https://doi.org/10.1504/IJISTA.2016.078347
23. Sandhya N., Sekar M.R. (2018) Analysis of Variant Approaches for Initial Centroid Selection in K-Means Clustering Algorithm. In: Satapathy S., Bhateja V., Das S. (eds) Smart Computing and Informatics. Smart Innovation, Systems and Technologies, vol 78. Springer, Singapore
24. Kurada R.R., Kanadam K.P. (2019) A Novel Evolutionary Automatic Clustering Technique by Unifying Initial Seed Selection Algorithms into Teaching-Learning-Based Optimization. In: Soft Computing and Medical Bioinformatics. Springer Briefs in Applied Sciences and Technology. Springer, Singapore

AUTHORS PROFILE

MANOJ KUMAR GUPTA is research scholar of University School of Information, Communication & Technology, GGSIPU, Delhi, India. He worked as Professor at Rukmini Devi Institute of Advanced Studies (Aff. to Guru Gobind Singh Indraprastha University), Delhi, India. He was also Dean Examination, Admission and Administration in the Institute as additional charge. He has more than 20 years of experience in teaching and administration. His interest areas are Database Systems, Data Warehousing and Data Mining. He has 4 books and 20+ international / national research papers to his credit.

PRAVIN CHANDRA received the M.Sc. degree in Physics from the University of Delhi, India, in 1993, the M.Tech. degree from the Indian School of Mines, Dhanbad, India, in 1998, and the Ph.D. degree from Guru Gobind Singh Indraprastha University in 2004. He is currently a Professor at the University School of Information, Communication and Technology, Guru Gobind Singh Indraprastha University, Delhi, India where he was the Controller of Examinations also as the additional charge. His research interests are in the areas of artificial neural networks, soft computing, finger print analysis, ad-hoc networks, and software engineering.