Comparison between Standard K-Mean Clustering and Improved K-Mean Clustering

Pooja Pandey
Research Scholar
CSE & IT Department
Baba Banda Singh Bahadur Engineering
Collage Fatehgarh Sahib

Ishpreet Singh
Assistant Professor
CSE & IT Department
Baba Banda Singh Bahadur Engineering
Collage Fatehgarh Sahib

ABSTRACT
Clustering in data mining is very important to discover distribution patterns and this importance tends to increase as the amount of data grows. It is one of the main analytical methods in data mining and its method influences its results directly. K-means is a typical clustering algorithm[3]. It mainly consists of two phases i.e. initializing random clusters and to find the nearest neighbour. Both phases have some shortcomings which are discussed in the paper and two methods are purposed based on that. First one is about how to generate the centroids and the second one will reduce the time while calculating distance from centroid.

Keywords
K-Mean Clustering

1. INTRODUCTION
Clustering is a method that organizes data into different classes of similar characteristics. It is the way of searching hidden patterns. The demand for learning and extracting valuable information from data and organising the data in short time have made the clustering technique so important that it is now used in many areas like artificial intelligence, image processing, customer relationship management, criminal records etc. Clustering is often confused with classification but it is different in many aspects. Clustering is an unsupervised classification of patterns[4]. The grouping is done by minimizing the sum of squared distances between items and the corresponding centroid using Euclidean distance.

This paper purposes a superior k means algorithm which will explains cluster initialization method and then another method which reduces the time for distance calculation of nearest neighbour from centroid. With the increase in size of datasets, this algorithm is losing its importance due to some reasons[6]. These shortcomings are: initialising centroid randomly and Calculating distance between each data object and all data objects in each iteration. For the first one a technique to generate centroid is explained and for the second one it uses an improved k means clustering algorithm which avoids computing the distance of each data object to the cluster centers repeatedly saving running time.

This paper includes five parts. The second part will explain the standard k means algorithm and the shortcomings of it. The third part will explain the method of generating the initial point or the centroid. In the next part an improved k means algorithm is explained which will find the nearest neighbour from centroid in lesser time and the last part will give the conclusion of this paper.

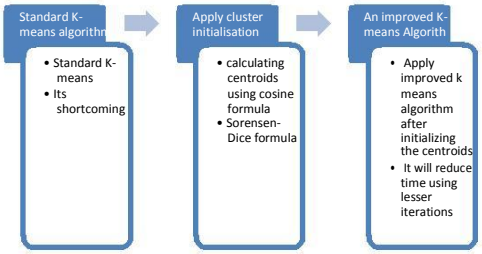

Fig.1.1 Flow of the process

2. STANDARD K MEANS CLUSTERING ALGORITHM

2.1 The Process of K Means Algorithm
K means is simple and widely used technique of clustering. It is completely based partitioning methodology. It partitions n-data items into k groups where k indicates number of clusters specified by the user. Clusters are formed such that each item in the cluster has minimum distance from the centroid. For calculating distance between item and the centroid, k means algorithm uses the Euclidean distance measurement. It aims to minimize the sum of squared distances between all points and the cluster center. This procedure consists of following steps:

Input: K: the number of desired clusters.

Output: A set of k clusters

Algorithm

1. Randomly select k objects as initial centroids naming(m1,m2,m3)
2. Calculate the distance between each object Oi and each centroid, then assign each object to its nearest cluster center, formula for calculating distance as:

   $d(O_i,M_i) = \sqrt{\sum}=1(\ 1-\ 1)^2$, i=l.....N;

   j=l.....k;

   d (O_i, M_j) is the distance between data i and cluster j;
3. Calculate the mean in order to create the new cluster centers

$M_i = \frac{1}{}\Sigma$, i=l.....k; Z_i is the number

of samples of current cluster i;

4. Repeat step 2 and 3 until the criterion function E converged, return (m1, m2.....mk). Algorithm terminates.

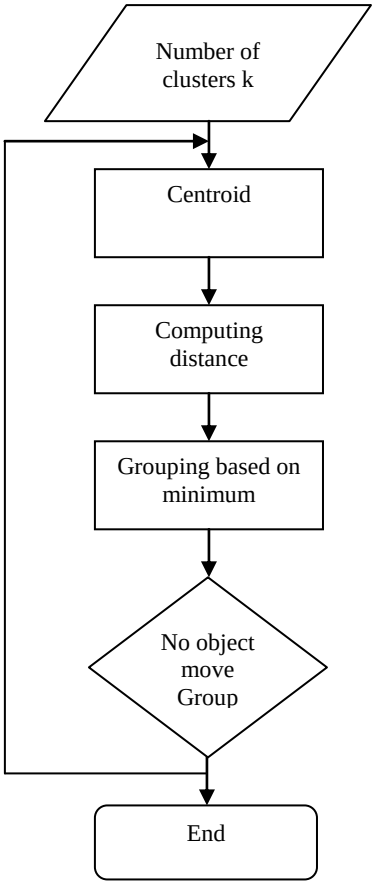

Fig. 2.1 The K- Means Algorithm Process

2.2 The Shortcomings of K Means Algorithm

- Calculating distance from each data object to every cluster center in each iteration which will take a lot of time.
- Random generation of centroid
- In huge dataset it will consume more time as it uses whole data at the same time.
- Problem in choosing initializing centers.
- K means has problem when the data contain outliers
- K means has problem when clusters are of different size, densities and non regular shapes.

The focus of this paper is on the first two limitations and also to purpose the alternative to these limitations.

3. INITIALISING THE CENTROID

The main part in the k means is to provide appropriate number of clusters. Selecting random clusters without applying any algorithm is impractical and it also require deep knowledge of clustering. This part purposes an algorithm that improves the process of selecting centroid.

The complexity of standard k-means algorithm:

Time complexity is O(n*k*t)

Where n = total no. Of elements

k = no. of cluster iteration

t = no. of iterations

3.1 Algorithm

1. Select one lowest and one highest centroid point.
2. 2) After selecting these two cluster points, create two clusters with members which wre not similar to each other.

Input: data set with N number of attributes

A1,A2.....An ,

Where n = No. of attributes

Attributes should be in numeric form

Output: Appropriate no. of clusters with n data points[1].

Fig.3.1 Flow of initializing the clusters

4. AN IMPROVED K MEANS CLUSTERING ALGORITHM

An improved k means algorithm is based on weights. This partitioning algorithm can handle the data of numerical attribute. Data of symbol attribute can also be handled by it. This method also reduces noise and the impact of the isolated point. By doing this it enhances the efficiency of clustering. However this method has no improvement on complexity of time. This method gives a systematic way to find initial cluster centers. 0Hence this method can produce more accurate clustering results than standard k means algorithm.

The standard k means algorithm needs to calculate the distance from each data object to all the centers of k clusters, with the execution of each iteration which takes a lot of time specially in large data set. The main idea of algorithm is to set two simple data structures to retain the labels of cluster and the distance of all the data objects to the nearest cluster during the each iteration, that can be used in next iteration, first calculate the distance between the current data object and the new cluster center, if the calculated distance is smaller than or equal to the distance to the old center, the data object stays in its cluster that was assigned to in previous iteration.

Therefore, there is no need to calculate the distance from this data object to the other k-1clustering centers, saving the calculative time to the k-1 cluster centers. Otherwise, we must calculate the distance from the current data object to all k cluster centers, and find the nearest cluster center and assign this point to the nearest cluster center and then we can separately record the label of nearest cluster center and the distance to its center[2].

The process of the improved algorithm is described as follows:

Input

The number of desired clusters k, and a database

D={d1,d2,d3,d4 .....dn} containing n data objects.

Output

A set of k clusters

Steps:

1. Randomly select k objects from dataset D as initial cluster centers.
2. Calculate the distance between each data object di (1 < = i < = n) and all k cluster centers cj (1 < = j < = k) as Euclidean distance d(di , cj) and assign data object di to the nearest cluster.
3. For each data object di, find the closest center cj and assign di to cluster center j;
4. Store the label of cluster center in which data object di is and the distance of data object di to the nearest cluster and store them in array Cluster[ ] and the Dist[ ] separately.

   Set Cluster[i] =j, j is the label of nearest cluster.

   Set Dist[i] =d(di, cj), d(di, cj) is the nearest Euclidean distance to the closest center.
5. For each cluster j (1< = j < = k), recalculate the cluster center;
6. Repeat
7. For each data object di

   Compute its distance to the center of the present nearest cluster;

a. If this distance is less than or equal to Dist[i], the data object stays in the initial cluster;

b. Else

   For every cluster center cj(1< = j < = k), compute the distance $d(d_i, c_j)$ of each data object to all the center, assign the data object di to the nearest center c_j.

   Set Cluster[i]=j;

   Set Dist[i]= d(di, cj);

8. For each cluster center j(1< = j< = k), recalculate the centers;
9. Until the convergence criteria is met.
10. Output the clustering results[2];

This process of not calculating the distance of calculated ones makes this algorithm faster than the standard k means algorithm

5. ACKNOWLEDGEMENT

The authors are highly grateful and thankful to theDr. Baljit Singh Khehra(Head Of CSE & IT Department) of the Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib and to the college also.

6. CONCLUSION

K-means is an algorithm typically used for clustering large data sets. This paper elaborates k-means algorithm, analyse its shortcomings and also purposes the alternatives to these shortcomings. This algorithm creates the centroids on the random basis which is impractical and also it requires the deep knowledge of clustering. Secondly reassigning the data points a number of times makes the efficiency of this algorithm low. This paper explains the simple and efficient way of assigning centroids to clusters using cosine, Sorensen dice and Manhatten distance formula and an improved version of k-means which ensures the entire process in O(nk) time without altering the accuracy of clusters.

7. REFERENCES

[1] Rajeswari, K., Acharya, O., Sharma, M., Kopnar, M., & Karandikar, K." Improvement in k-Means Clustering Algorithm Using Data Clustering", In Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on,vol.3, no.15, pp. 367-369, IEEE.

[2] Research on k-means Clustering Algorithm An Improved k-means Clustering Algorithm Shi Na College of Information Engineering, Capital Normal

[3] Lima, M. F., Zarpelao, B. B., Sampaio, L. D., Rodrigues, J. J., Abrao, T., & Proença Jr, M. L." Detection using baseline and K-means clustering", In Software, Telecommunications and Computer Networks (softcom), 2010 International Conference on, vol.3, no.5 pp. 305-309, IEEE.

[4] Ren, Q., & Zhuo, X. " Application of an improved k-means algorithm in gene expression data analysis" In Systems Biology (ISB), 2011 International Conference on, pp. 87-91, IEEE.

[5] Wang, H., Qi, J., Zheng, W., & Wang, M. "Balance K-means algorithm. In Computational Intelligence and Software Engineering," Cise 2009 International Conference on, pp. 1-3, IEEE.

[6] Esteves, R. M., Hacker, T., & Rong, C. "Competitive k-means, a new accurate and distributed k-means algorithm for large datasets" In Cloud Computing Technology and Science (cloudcom), 2013 IEEE 5th International Conference on ,Vol. 1, pp. 17-24, IEEE.

[7] Tian, L., & Jianwen, W. "Research on network intrusion detection system based on improved k-means clustering algorithm", In computer Science-Technology and Applications, 2009. IFCSTA'09. International Forum on Vol. 1, pp. 76-79, IEEE.

[8] Singh, G., Antony, D. A., & Leavline, E. J" Data mining in network security-techniques & tools: a research perspective", Journal of theoretical & applied information technology, vol.2, no.57

[9] Yang, Q., & Wu, X. "10 challenging problems in data mining research.International Journal of Information Technology & Decision Making", vol.5, no.4,pp.597-604.

[10] Chen, C. H., Tseng, V. S., & Hong, T. P. ,"Cluster-based evaluation in fuzzy-genetic data mining. Fuzzy Systems", IEEE Transactions on, vol. 1, no.16,pp. 249-262.

[11] Liao, S. H., Chu, P. H., & Hsiao, P. Y," Data mining techniques and applications–A decade review from 2000 to 2011", Expert Systems with Applications, vol.12, no.39, pp.11303-11311.

[12] Balabantaray, R. C., Sarma, C., & Jha, M. (2015). Document Clustering using K-Means and K- Medoids. Arxiv preprint arxiv:1502.07938.

[13] Sujatha, M. S., & Sona, M. A. S.,"New fast k-means clustering algorithm using modified centroid selection method", Ininternational Journal of Engineering Research and Technology ,Vol. 2, No. 2 ,February-2013.

[14] Brar, R., & Sharma, N., "A Novel Density Based K-Means Clustering Algorithm for Intrusion Detection", Journal of Network Communications and Emerging Technologies (JNCET) www. Jncet. Org, vol.3, no.7

[15] W. Zhao, H. Ma, and Q. He, "Parallel K-Means Clustering Based on MapReduce," vol. 5931, Springer Berlin / Heidelberg, 2009, pp. 674– 679.

[16] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable k-means++," Proc. VLDB Endow., vol. 5, no. 7, pp. 622–633, 2012.

[17] M.V.B.T.Santhi,V.R.N.S.S.V.SaiLeela,P.U.Anitha,D.Na gamalleswari" Enhancing K-Means Clustering Algorithm" International Journal on Computer Science And Technology(IJCST) Vol 2,Issue 4,Oct-Dec 2011