Performance Improvement of Clustering Method Based on Random Swap Algorithm

Sunjin Yu¹ and Changyong Yoon²

¹School of Digital Media Engineering, Tongmyong University, Busan, Korea

²Department of Electrical Engineering, Suwon Science College, Hwaseong, Korea

ljfis

Abstract

This paper proposes a method for improving performance of clustering algorithm. Among unsupervised learning methods such as clustering, K-means has the advantage of being widely used and simple to implement, but has problems that is heavily influenced by initial centroids and may be got stuck at local minimum. To minimize these shortcomings, this work uses a clustering method with a random swap algorithm based on K-means++ that calculates the distance between data and the center point as a probability. The swap-based clustering method replaces the existing centroids and the concentrated centroids with each other during clustering steps, and then performing re-partitions, and fine-tuning steps with K-means++. The experimental results show the swap-based K-means++ clustering method proposed in this paper has better performance than other methods, as comparing the method proposed with the method of other clustering under various circumstances. As producing datasets of vehicles on general purpose for measuring performance in various experimental environments, we demonstrate the excellence of the proposed method.

Keywords: Clustering, K-means, K-means++, Random swap algorithm

1. Introduction

Clustering is a method out of unsupervised learnings used in many fields, such as machine learning, document classification, pattern recognition, intelligent robot system and image segmentation [1–4]. Therefore, for a wide variety of fields, it can be used to extract summary information of datasets by grouping unlabeled datasets into groups with the same characteristics and to discover the characteristics of datasets. Clustering has a variety of methods and K-means among these is most widely used. Although it uses small calculations and mount of memory, it has the disadvantage of being sensitive to initial centroids. It generally executes multiple initializations of centroids called repeated K-means. K-means++, that is proposed to solve these problems and keeps good speed and accuracy, is a probability technique based method of selecting initial centroids to avoid outliers as much as possible while keeping the centroids from concentrated [5, 6].

Also, the main technique to be necessary in clustering is to correctly choose global allocation of clusters, because K-means locally perform fine tuning to get exact partition boundaries. The algorithms based on K-means have the disadvantage that it has the risks to converge at local minimum points and result in an unstable result. To address these local optimal problems, swap-based clustering, a method of local search reasoning, is used in this paper. This clustering

Received: Jun. 5, 2019
 Revised : Jun. 13, 2019
 Accepted: Jun. 22, 2019

Correspondence to: Changyong Yoon
 (cyoon@ssc.ac.kr)
 ©The Korean Institute of Intelligent Systems

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

is the technique that finds the optimum centroids by performing the swap between existing centroids and a set of candidate centroids, and followed by fine-tuning [7].

Swap-based clustering includes the random-swap method (Random Swap) that replace the random chosen centroid with the random data of object, and the critical swap method (Deterministic Swap) that choose centroids to replace by increasing the result values of the minimum cost function [8]. In order to improve degraded performance due to the initial centroids and local minimum value of the traditional K-means based method, we propose the swap-based clustering method applying K-means++ algorithm and random swap method.

The experiments in this paper demonstrate that the proposed method recovers the problems of clustering and improves the performance of clustering as applying the proposed method through experiments under various conditions.

The rest of this paper is organized as follows. In the next section, we introduce K-means++ clustering method based on the proposed elementary technique. Section 3 presents the proposed system based on random swap algorithm and Section 4 shows the experimental result to prove the performance improvement of the proposed method through comparison with other methods. Finally, Section 5 summarizes the features and conclusions of this paper and describes future tasks.

2. K-means++ Clustering

In this section, we describes and explains the clustering algorithm called K-means++ with a specific way of selecting centroids. As mentioned in the introduction section, K-means algorithm may not converge to result values depending on how the initial centroids are selected [9]. To improve performance of K-means clustering, K-means++ selects the optimal initial centroids of the clusters from the first to the subsequent by using the probability method. K-means++ can improve speed and accuracy as distributing the centroids of the initial cluster as widely as possible, and it generally uses the Euclidean distance method for the calculation of the minimum distance value [10].

This method is expected to improve performance as yielding less error than K-means clustering. Figure 1 shows the structure of K-means++ clustering algorithm. In step 1-(2), the weighting method called ‘ D^2 weighting’ is used. Although it takes extra time to select the initial centroids, K-means++ converges faster than K-means and the approximation ratio of its method is $O(\log K)$, where k is the number of clusters used.

1. Selecting the initial centroids of clusters
 - (1) Selecting the centroid c_1 of first cluster from data X
 - (2) Selecting the centroid c_i of next cluster using $D(x)^2 / \sum_{x \in X} D(x)^2$
 $D(x)$ is the minimum distance from x to the centroid of the closest cluster we have already chosen.
 - (3) Repeating step (2) until the centroid of the k clusters are completely calculated
2. Calculating the distance from the centroid of each cluster to each point in the dataset
3. Assigning a point in the dataset to the closest centroid of cluster
4. Calculating the centroids of each cluster by the average value of each point using $c_j = \sum_{i=1}^{N_j} X_i / N_j$
5. Repeating Step 2-Step 4 until no change happens

Figure 1. The structure of K-means++ clustering algorithm.

3. The Proposed Random Swap based Clustering Algorithm

In this section, we propose random swap based clustering method for recovering performance deterioration due to strong dependency on initial centroid and the high possibility to get stuck in local minimum value.

This algorithm selects initial values with the probability method used in K-means++ and performs partitions on data in a cluster with the swapped centroids that removed from clusters. Also, It performs partitions for new cluster to which the swapped centroid is reallocated [11]. Since the swapped centroids will only affect the partitions around them, and not affect the other partitions, the performance of the local repartition is meaningful like that. The local partition is performed in two steps. The first step is to perform an optimal partition only for the data object of the swapped cluster to remove from the clusters, as comparing the distances to all other prototypes and selecting the closest one.

$$\text{optimal_partition} = \arg \min_{1 \leq k \leq M} d(x_i, c_k)^2 \quad \forall i \mid p_i = j. \quad (1)$$

The second step is to perform the creation of a partition for the new cluster in the area where the swapped centroid is relocated, as getting data from nearby clusters by computing the distance of all data to the new prototype.

$$\text{new_partition} = \arg \min_{k=j \vee k=p_i} d(x_i, c_k)^2 \quad \forall i \in [1, N], \quad (2)$$

where M is the number of clusters, C is the centroid set of cluster. And, N is the number of data objects, P is partition set. X is dataset with N data objects.

Thus, it can reduce the probability of occurrence on results falling into local minimum value. After the local repartition step [12], fine tuning is performed by repeated use of two K-means++ for both cluster areas.

To guarantee the improvement of clustering quality, the number of iterations for random swap clustering should be set much enough to perform successful swaps. For analyzing exactly, the algorithm has the characteristics of linearly depending on the number of data. The swap is performed if the quality of clustering result improves with trial-and-error. This approach is very simple to implement and very effective. This algorithm consists of five steps; initialization step, partition step, swap step, local repartition step, and fine-tuning step. As mentioned above, the probabilistic method is used to be less affected by the initial value in the initialization step, and the K-means++ method was used to reduce the phenomenon of falling into the local minimum during the fine-tuning step.

Figure 2 shows the structure of the proposed algorithm in this paper.

4. Experimental Results

This section shows that the performance of the proposed method is better than that of the other methods through various experiments. In the experimental circumstance of this paper, we use CPU 2.50 GHz, RAM 8 G as hardware specification, and MATLAB is used as development software.

In the first experiment of this paper, performance comparison was performed for K-means, K-means++, Random Swap with K-means (RSK), and the Random Swap-based algorithm (Proposed Random Swap with K-means++) applied the proposed method. K-means and K-means++ are the standard clustering methods that have the difference on the selecting of the initial centroids. RSK is the clustering method applying K-means at initialization step and fine-tuning step, the proposed method is that applying K-means++ at the same steps.

In this experiment, the dataset uses Iris and Wine, which they are famous in the UCI Repository. Iris dataset has 3 instances and 4 attributes, and Wine dataset has 178 instances and 13 attributes. Through this experiment, we will show and compare the performance differences depending on data's dimensions.

[Initialization Step]

1. Selecting the initial centroids of clusters
 - (1) Selecting the centroid c_1 of first cluster from at random data X
 - (2) Selecting the centroid c_i of next cluster using $D(x)^2 / \sum_{x \in X} D(x)^2$
 $D(x)$ is the minimum distance from x to the centroid of the closest cluster we have already chosen.
 - (3) Repeating step (2) until the centroid of the k clusters are completely calculated

[Partition Step]

2. Calculating the distance from the centroid of each cluster to each point in the data set
3. Getting a clustering partition by assigning a point in the dataset to the closest centroid of cluster

[Swap Step]

4. Randomly selecting the centroid to replace and reassigning to the location of the randomly selected data.

[Local Repartition Step]

5. Performing partition only for the data in the swapped cluster that will be removed from the cluster
6. Creating partitions for new clusters in areas where the swapped centroid are reassigned

[Fine-tuning Step]

7. Tuning as applying K-means++ algorithm with

$$\begin{aligned} &[\text{new_cluster}, \text{new_partition}] \\ &= \text{Kmeans++}(X, \text{new_cluster}, \text{new_partition}) \end{aligned}$$

8. If the new partition which created from this calculation provides a lower distortion than the previous partition, this swap is allowed.
9. Repeating Step 4-Step 8 until no change happens in clusters

Figure 2. The proposed random swap based algorithm.

Figures 3 and 4 show the average MSE change trends between the centroid and the data points within the cluster with Eq. (3), according to the clustering method and the number of clusters. As can be seen in Figures 3 and 4, even in situations where the dimensions are large and performance is deteriorated, the proposed method shows better performance than other methods. Also, as the number of clusters increases, we see that the proposed method improves the value of MSE [13]. Tables 1 and 2 show the average MSE value precisely computed according to the clustering method and the number of clusters in a different

Figure 3. MSE change graph according to cluster # on Iris dataset.

Figure 4. MSE change according to cluster # on Wine dataset.

Table 1. MSE value according to cluster # on Iris dataset

Methods	Cluster#			
	2	3	4	5
K-means	1.2000	0.5233	0.3990	0.3214
K-means++	1.1000	0.5220	0.3812	0.3190
RSK	1.1000	0.5215	0.3800	0.3050
Proposed method	1.1000	0.5210	0.3715	0.3001

Table 2. MSE value according to cluster # on Wine dataset

Methods	Cluster#			
	9	11	13	15
K-means	3.8000	3.5000	3.3000	3.1200
K-means++	3.5000	3.5000	3.2500	3.0200
RSK	3.5000	2.9000	2.7500	2.7000
Proposed method	3.4000	2.8000	2.6500	2.5100

representation of Figures 3 and 4.

$$cf(c) = \frac{1}{N} \sum_{i=1}^N d(x_i, c_i)^2. \quad (3)$$

Figures 5 and 6 show the comparison result to check the

Figure 5. Time distortion efficiency according to clustering method on Iris dataset.

Figure 6. Time distortion efficiency according to clustering method on Wine dataset.

clustering quality. At the beginning of time in Figure 5, the proposed method is not as good as the others. But, as swapping a centroid that is likely to fall into the local minimum, the performance of the proposed method is superior to other methods over time and it has faster convergence than other method, even though it is tiny. Figure 6 shows that the proposed method has generally better performance and faster convergence than other methods over the entire time.

In the following experiment, we collect information on various vehicles, and then create datasets [14]. This experiment performs various clustering method with these datasets, and comparative analysis. The information of these datasets is shown in Table 3.

Figure 7 and Table 4 show the results of comparing accuracy according to clustering method using above data, after setting the repeat count (TC) of K-means-based algorithm module to 10,000 and the number of clusters (C) to 4. We can see that the proposed method has small times to converge and higher

Table 3. Vehicle dataset

Number of instances	150
Area	Road
Characteristics of attribute	Real
Number of attributes	4
Kinds of attributes	The ratio (height/width) of Front, The ratio (height/width) of Rear, The ratio (height/width) of Side, The circumference of rear

Figure 7. Accuracy graph according to clustering method.

Table 4. Accuracy value according to clustering methods

Method	Accuracy (%)
K-means	98.0
K-means++	98.1
RSK	99.3
Proposed method	99.5

accuracy than other methods.

5. Conclusions

This paper proposed a random swap-based algorithm with K-means++ to solve the problems in which the performance of K-means clustering is greatly affected by the initial value and may falls into local minimum. By not selecting the initial centroids of clusters randomly and selecting the most distributed points using a probabilistic method as a whole, we can see that this method can be converged with the improvements of performance and even fewer iterations because they were chosen as the these centroids. Additionally, it was confirmed through

the experimental results that the performance of clustering was improved even as the number of clusters changed, as the state of local minimum were reduced by continuing to perform partitions until the optimal centroid was selected.

Also, in addition to the given data, we create datasets for the vehicle and perform the experiments. Over time, it was possible to verify that the proposed method was higher in accuracy than other methods.

Conflict of Interest

No potential conflict of interest relevant to this article has been reported.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2017R1C1B5017751).

References

- [1] J. Chen, "Swap-based clustering for location-based services," Master's thesis, University of Eastern Finland, Kuopio, Finland, 2011.
- [2] M. C. Su and C. H. Chou, "A modified version of the K-means algorithm with a distance based on cluster symmetry," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 23, no. 6, pp. 674-680, 2001. <http://doi.org/10.1109/34.927466>
- [3] J. Qin, W. Fu, H. Gao, and W. X. Zheng, "Distributed k-means algorithm and fuzzy c-means algorithm for sensor networks based on multiagent consensus theory," *IEEE Transactions on Cybernetics*, vol. 47, no. 3, pp. 772-783, 2017. <http://doi.org/10.1109/TCYB.2016.2526683>
- [4] M. Kim, "Simultaneous learning of sentence clustering and class prediction for improved document classification," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 17, no. 1, pp. 35-42, 2017. <https://doi.org/10.5391/IJFIS.2017.17.1.35>
- [5] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, New Orleans, LA, 2007, pp. 1027-1035.

- [6] F. Dzikrullah, N. A. Setiawan, and S. Sulisty, "Implementation of scalable k-means++ clustering for passengers temporal pattern analysis in public transportation system (BRT Trans Jogja case study)," in *Proceedings of 2016 6th International Annual Engineering Seminar (InAES)*, Yogyakarta, Indonesia, 2016, pp. 78-83. <http://doi.org/10.1109/INAES.2016.7821911>
- [7] P. Franti, O. Virtajoki, and V. Hautamaki, "Probabilistic clustering by random swap algorithm," in *Proceedings of 2008 19th International Conference on Pattern Recognition*, Tampa, FL, 2008, pp. 1-4. <http://doi.org/10.1109/ICPR.2008.4761798>
- [8] Q. Zhao and P. Franti, "Centroid ratio for a pairwise random swap clustering algorithm," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 5, pp. 1090-1101, 2014. <http://doi.org/10.1109/TKDE.2013.113>
- [9] D. G. Ji and D. M. Huang, "A research based on k-means clustering and artificial fish-swarm algorithm for the vehicle routing optimization," in *Proceedings of 2012 8th International Conference on Natural Computation*, Chongqing, China, 2012, pp. 1141-1145. <http://doi.org/10.1109/ICNC.2012.6234729>
- [10] B. Aubaidan, M. Mohd, and M. Albared, "Comparative study of k-means and k-means++ clustering algorithms on crime domain," *Journal of Computer Science*, vol. 10, no. 7, pp. 1197-1206, 2014. <http://doi.org/10.3844/jcssp.2014.1197.1206>
- [11] P. Franti, "Efficiency of random swap clustering," *Journal of Big Data*, vol. 5, article no. 13, 2018. <https://doi.org/10.1186/s40537-018-0122-y>
- [12] M. G. C. Resende and R. F. Werneck, "On the implementation of a swap-based local search procedure for the p-median problem," in *Proceedings of the 5th Workshop on Algorithm Engineering and Experiments*, Baltimore, MD, 2003, p. 119-127.
- [13] S. Krinidis and V. Chatzis, "A robust fuzzy local information C-means clustering algorithm," *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp. 1328-1337, 2010. <https://doi.org/10.1109/TIP.2010.2040763>
- [14] S. B. Changalasetty, A. S. Badawy, L. S. Thota, and W. Ghribi, "Classification of moving vehicles using k-means clustering," in *Proceedings of 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Coimbatore, India, 2015, pp. 1-6. <https://doi.org/10.1109/ICECCT.2015.7226041>

Sunjin Yu received his B.S. degree from the Department of Electronics and Information Engineering of Korea University in 2003 and received his M.S. degree in Graduate Program in Biometrics and Ph.D. degree in Electrical and Electronic Engineering from Yonsei University in 2003 and 2011, respectively. He was a senior research engineer in LGE Advanced Research Institute from 2011 to 2012. From 2012 to 2013, he was a research professor in the Department of Electrical Engineering in Yonsei University and from 2013 to 2016, he was a professor in the Department of Broadcasting and Image, Cheju Halla University. He is currently a professor in the School of Digital Media Engineering, Tongmyong University. His research interests include 3D computer vision, human computer interaction, and augmented/virtual reality.
E-mail: sjyu@tu.ac.kr

Changyong Yoon received his B.S., M.S., and Ph.D. degrees in Electrical and Electronic Engineering from Yonsei University, Seoul, Korea, in 1997, 1999, and 2010, respectively. He was a senior research engineer in LG Electronics Inc., and LG-Nortel, and he developed system software for the DVR and WCDMA from 1999 to 2006. From 2010 to February 2012, he was a chief research engineer in LG Display and developed the circuit and algorithms in touch systems. Since 2012, he has been a Professor in the Department of Electrical Engineering, Suwon Science College, Gyeonggi-do, Korea. His main research interests include intelligent transportation systems, pattern recognition, robot vision, and fuzzy application systems.
E-mail: cyyoon@ssc.ac.kr