Persistent K-Means: Stable Data Clustering Algorithm Based on K-Means Algorithm

Rasool Azimi[a,*], Hedieh Sajedi[b]

[a] Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran
[b] Department of Computer Science, College of Science, University of Tehran, Tehran, Iran

Abstract

Identifying clusters or clustering is an important aspect of data analysis. It is the task of grouping a set of objects in such a way those objects in the same group/cluster are more similar in some sense or another. It is a main task of exploratory data mining, and a common technique for statistical data analysis This paper proposed an improved version of K-Means algorithm, namely Persistent K-Means, which alters the convergence method of K-Means algorithm to provide more accurate clustering results than the K-means algorithm and its variants by increasing the clusters' coherence. Persistent K-Means uses an iterative approach to discover the best result for consecutive iterations of K-Means algorithm.

Keywords: Data mining, clustering, K-means, Persistent K-Means.

1. Introduction

The main goal of clustering is to generate compact groups of objects or data that share similar patterns within the same cluster, and isolate these groups from those which contain elements with different characteristics [1, 4, 6, 11, 15, and 19].

The K-means algorithm is one of the most commonly used clustering algorithms, which uses the data reassignment method to repeatedly optimize clustering [1, 2, 7 and 9]. Although the K-means algorithm has features such as simplicity and high convergence speed, it is totally dependent on the initial centroids which are randomly selected in the first phase of the algorithm. Due to this random selection, the algorithm does not always converge to the optimized solution [16]. Include taking into account the K value more than its actual value. Or select the initial centroids with a special technical for converging the K-means algorithm to more accurate clustering results. Different variants of K-means algorithm have been proposed to address these limitations. In [18] four initialization approaches are proposed for K-Means algorithm is presented that these approaches include: In [18] four initialization approaches are proposed for K-Means algorithm is presented that these approaches includes: Random, Forgy, MacQueen and Kaufman. Random method divides the database

* Corresponding author. Email: r.azimi@qiau.ac.ir

Archive of SID

www.SID.ir

into a partition of K randomly selected clusters. In Forgy, K instances of the database (seeds) are chosen at random and the rest of the instances are assigned to the cluster represented by the nearest seed.

In Macqueen, K instances of the database (seeds) are chosen at random. Following the instance order, the rest of the instances are assigned to the cluster with the nearest centroid. After each assignment, a recalculation of the centroid has to be carried out. Finally in Kaufman method, the initial clustering is obtained by the successive selection of K representative instances. The first representative is chosen to be the most centrally located instance in the database. The rest of representative instances are selected according to the heuristic rule of choosing the instances that promise to have around them a higher number of the rest of instances and that are located as far as possible from the previously fixed ones.

In [17] a procedure for computing a refined starting condition from a given initial one that is based on an efficient technique for estimating the modes of a distribution. The refined initial starting condition allows the iterative algorithm to converge to a better local minimum. The procedure is applicable to a wide class of clustering algorithms for both discrete and continuous data. Reference [20] proposes a new approach to optimizing the designation of initial centroids for K-means clustering. This approach is inspired by the thought process of determining a set of pillars' locations in order to make a stable house or building. The authers consider the pillars' placement which should be located as far as possible from each other to withstand against the pressure distribution of a roof, as identical to the number of centroids amongst the data distribution. Therefore, the proposed approach in this paper designates positions of initial centroids by using the farthest accumulated distance between them. First, the accumulated distance metric between all data points and their grand mean is created. The first initial centroid which has maximum accumulated distance metric is selected from the data points. The next initial centroids are designated by modifying the accumulated distance metric between each data point and all previous initial centroids, and then, a data point which has the maximum distance is selected as a new initial centroid. This iterative process is needed so that all the initial centroids are designated. This approach also has a mechanism to avoid outlier data being chosen as the initial centroids. In reference [21] the authors propose a new approach to optimize the initial centroids for K-Means which spreads them in the feature space uniformly so that the distance among them is as far as possible.

In spite of the improved performance of the K-Means variants for synthetic datasets with Gaussian distribution, their performance on real datasets is not very promising and different from the original K-Means algorithm. In addition, all K-Means based algorithms lack an efficient method to determine the optimal number of clusters. Either this requires the user to determine the number of clusters arbitrarily or based on practical and experimental estimates, which might not be the optimal values.

In this paper, we propose an improved version of K-Means algorithm, namely Persistent K-Means, which alters the convergence method of K-Means algorithm to provide more accurate clustering results than the K-means algorithm and its variants by increasing the clusters' coherence. Persistent K-Means uses an iterative approach to discover the best result for consecutive iterations of K-Means algorithm.

The rest of the paper is organized as follows. Section II provides a brief description of the Persistent K-means algorithm. Section III demonstrates a case study where the clustering errors are calculated for different algorithms. Finally, Section IV concludes the paper.

2. Persistent K-means

In this section, an improved version of K-Means algorithm, namely Persistent K-Means, is proposed, which alters the convergence method of K-Means algorithm to present deterministic and accurate clustering results. Persistent K-Means uses an iterative

Archive of SID

www.SID.ir

approach to discover the best result for consecutive iterations of K-Means algorithm. Assume a matrix called $Best\_Dist_{K \times N}$, with K rows and N columns. Where, K is the number of clusters, N is the total data count, and each element of the matrix equals the Euclidean distance of each data item from centroids of the discovered clusters. Using this matrix, we can assign each data to the closest cluster centroid. Now, considering another matrix called $M\_Dist_{K \times 3}$, in which each row, from 1 to K, represents a cluster centroid, and the three columns respectively contain the number, total Euclidean distance, and the average distance of the data assigned to the corresponding cluster. Equation (1) presents the third column values of this matrix.

$$M\_Dist_{(K,3)} = (M\_Dist_{(K,2)} / M\_Dist_{(K,1)}) \quad (1)$$

Assuming Ave_Dist is the mean of the second column values in matrix $M\_Dist_{K \times 3}$, for consecutive iterations of K-Means algorithm, a low value of Ave_Dist indicates that the K-Means algorithm is converted to a more accurate result. This condition is the "first constraint" of this dissertation, which is proposed to achieve the best result for different iterations of the algorithm. Generally, in order to achieve the optimal result of K-Means algorithm, we need to run 1 to R iterations of this algorithm, based on the number of iterations defined by the user. The results corresponding to the iteration with the minimum Ave-Dist is returned as the final output of the algorithm. This is illustrated by an example in the following. We must note that all synthetic data used in this section are also applied in [15]. The dataset used in this section includes 25620 2-dimensional data and 10 clusters. In Table 1, results of 10 consecutive iterations are presented and the minimum value of Ave-Dist corresponds to iteration 8. Accuracy evaluation results based on AC accuracy measure, indicate that the clustering accuracy in this iteration is higher that of other iterations.

Table 1

K-Means algorithm results in 10 rounds

Iteration No.	Ave_Dist	AC
1	655401294525.546	80.12
2	338521198628.016	90.00
3	327507683280.353	90.06
4	349949420687.898	90.04
5	827217507568.969	70.24
6	327507683280.353	90.06
7	2117958554025.93	79.48
8	242094107823.816	99.96
9	321337928867.259	90.01
10	349517225397.660	90.05

Table 1, presents the results of matrix $M\_Dist_{K \times 3}$ in iteration 8 and Fig. 1, presents an overview of the clustering results of iteration 8.

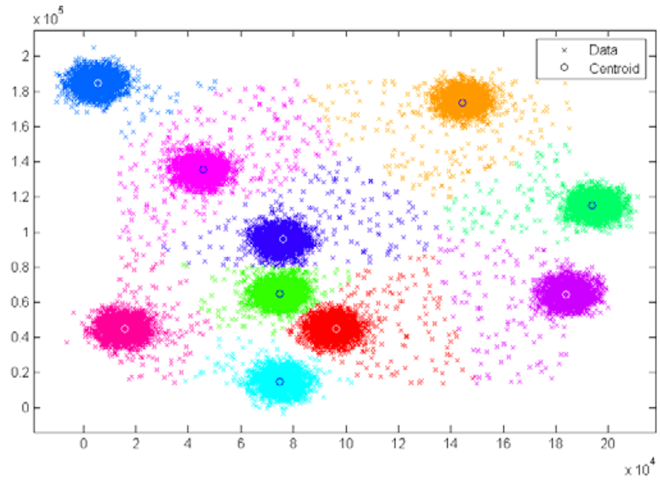

Fig. 1. Eighth iteration results of K-Means algorithm

Table 2 presents more details regarding the results of matrix $M\_Dist_{K \times 3}$ in the eighth iteration of K-Means algorithm using the synthetic data set of the experiments.

Archive of SID

Table 2

Results of $M\_Dist_{K\times3}$ in the eighth iteration of K-Means

K	$M\_Dist_{(K,1)}$	$M\_Dist_{(K,2)}$	$M\_Dist_{(K,3)}$
1	2588	256373187510.78	99062282.654861
2	2643	353934790719.30	133914033.56765
3	2664	357797992679.24	134308555.81052
4	2514	153390625395.99	61014568.574379
5	2524	228026460091.37	90343288.467262
6	2496	147624903712.54	59144592.833550
7	2470	150166385057.56	60796107.310754
8	2607	301044717531.40	115475534.15090
9	2571	284512324697.62	110662125.51444
10	2543	188069690842.32	73955835.958446

Moreover, Fig. 2 presents the clustering results in iteration 10 (which has an accuracy of 90.05%).

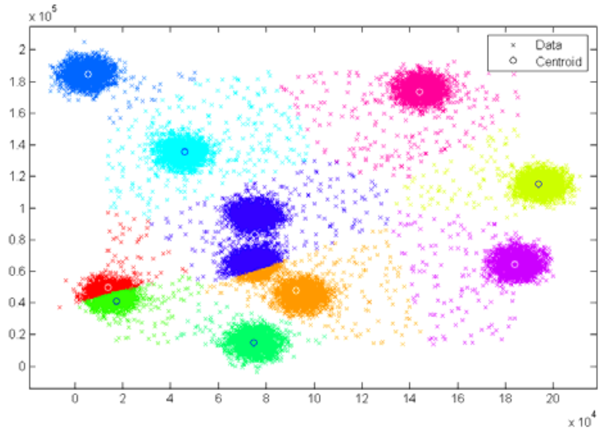

Fig. 2. Tenth iteration results of K-Means algorithm

Satisfying the "first constraint" for consecutive iterations of K-Means algorithm leads to results that are more accurate. However, in some cases, in which some clusters are close and some are far from each other, increases the chance of making mistakes and even by considering the "first constraint", it's not possible to provide the most accurate result. For instance, assume a synthetic dataset consisting of 10752 two-dimensional records and 10 clusters. Table 3 presents the results of 10 consecutive runs of the algorithm on this dataset.

Table 3

K-Means algorithm results in 10 iterations

Iteration No.	Ave_Dist	AC
1	110170616864.106	90.30
2	343677320109.839	79.28
3	204069771177.445	70.36
4	167025068278.392	79.99
5	233767050297.790	79.97
6	86531408864.7107	99.90
7	117312819963.335	80.67
8	199513262433.238	80.16
9	205711134156.617	70.35
10	76159749961.1842	90.33

As can be seen in Table 3, even though the minimum value of Ave_Dist corresponds to iteration 10, the accuracy of iteration 10 is less than that of iteration 6. Considering Ave_Dist, the result of 10 runs of the algorithm must be returned as the final result. However, according to Fig. 3, in iteration 10, the algorithm specifies some cluster centroids by mistake and based on AC accuracy measure results, the best result is obtained by iteration 6. The reason for the wrong answer of iteration 10 is definitely the wrong cluster centroid in low Centralized arrears. In what follows a solution is proposed for this problem.

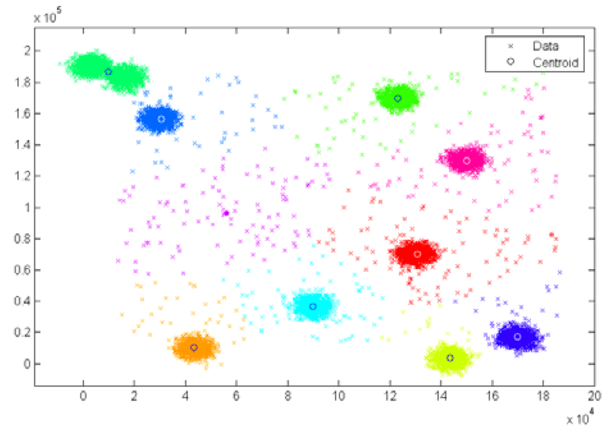

Fig. 3. Tenth round results of K-Means algorithm

www.SID.ir

Table 4 presents the results of $M\_Dist_{K\times3}$ in iteration 10. In an ideal situation, for the relevant datasets, cluster sizes must be almost the same. However, in iteration 10 (Table 4), the data assigned to the centroid of the fifth cluster (K=5) are almost twice the expected values, since one cluster centroid is selected for two adjacent clusters (Fig. 3). Moreover, the number of data records assigned to the centroid of the ninth cluster (K=9) is very low, since this centroid is selected at a point far from the Centralized area of the data space. As it was mentioned, the third column of $M\_Dist_{K\times3}$ representing the average distance of a data record from the corresponding cluster centroid. These values are presented in the fourth column of Table 4.

Table 4

Results of $M\_Dist_{K\times3}$ in the tenth iteration of K-Means

K	$M\_Dist_{(K,1)}$	$M\_Dist_{(K,2)}$	$M\_Dist_{(K,3)}$	Ave
1	1124	108927851257.2	96910899.69504	0.66
2	1057	51326998672.24	48559128.35595	0.33
3	1039	33126823778.59	31883372.26043	0.21
4	1080	74642337439.19	69113275.40665	0.47
5	2052	163558171755.2	79706711.38171	0.54
6	1086	55592604154.58	51190243.23626	0.35
7	1067	56332747047.03	52795451.77791	0.36
8	1046	36021771024.52	34437639.60279	0.23
9	105	95117399331.34	905879993.6318	6.24
10	1096	86950795151.80	79334667.10931	0.56

In cases where cluster centroids are falsely selected far from Centralized areas, the average distance of the data assigned to these cluster centroids is much larger than the cluster centroids, which are selected from Centralized areas. For a better representation of the amount of this difference, each element of the fourth column of Table 4 is divided by the mean of its values and results are presented in the fifth column of the Table 4.

According to the results in Table 4 the difference between the element of the ninth cluster (K=9) and the fifth cluster (K=5) is abnormal. The reason behind this is the large difference between the mean and median of the elements in the third column $M\_Dist_{K\times3}$. In order to prevent this type of problems, we have proposed a condition called the second constraint. Accordingly, if equation (2) is satisfied by one of the iterations, the results of that iteration is preferred to the results of iterations which only satisfy the "first constraint". In cases where the second constraint is satisfied, the result of the iteration is selected as the final result of the algorithm. In fact, for determining the final cluster centroids, the "second constraint" precedes the "first constraint".

$$(2\times \text{Mean\_Dist}) > (\text{Med\_Dist}) \tag{2}$$

For clusters 1 to K, the third column of $M\_Dist_{K\times3}$ consists of the average distance of a data record from the corresponding cluster centroid. In equation (3-2), Mean_Dist equals the mean of values in the third column of $M\_Dist_{K\times3}$ and Med_Dist equals the median of the elements in the third column of $M\_Dist_{K\times3}$. Mean_Dist and Med_Dist values of iteration 10 of K-Means in the previous example are:

Med_Dist = 468881682.9461516

Mean_Dist = 144981138.2457970

These results indicate that equation (2) is not satisfied in this iteration of K-Means. Therefore, it's necessary to apply some conditions to select the best result as the result of the algorithm in all cases. The process of Persistent K-Means algorithm is presented in Fig. 4.

According to Fig. 4, after determining dataset (D), the number of clusters (K), and the user-defined number of iterations (R), matrix $M\_Dist_{K\times3}$ are created

Archive of SID

www.SID.ir

based on the results of $Best\_Dist_{K \times N}$ (as one of the outputs of K-Means). Subsequently, Ave_Dist is computed and compared with a default value, i.e. infinite (the initial value of Min_Dist). Clearly, it's the lower value and is recorded as the new value of Min_Dist.

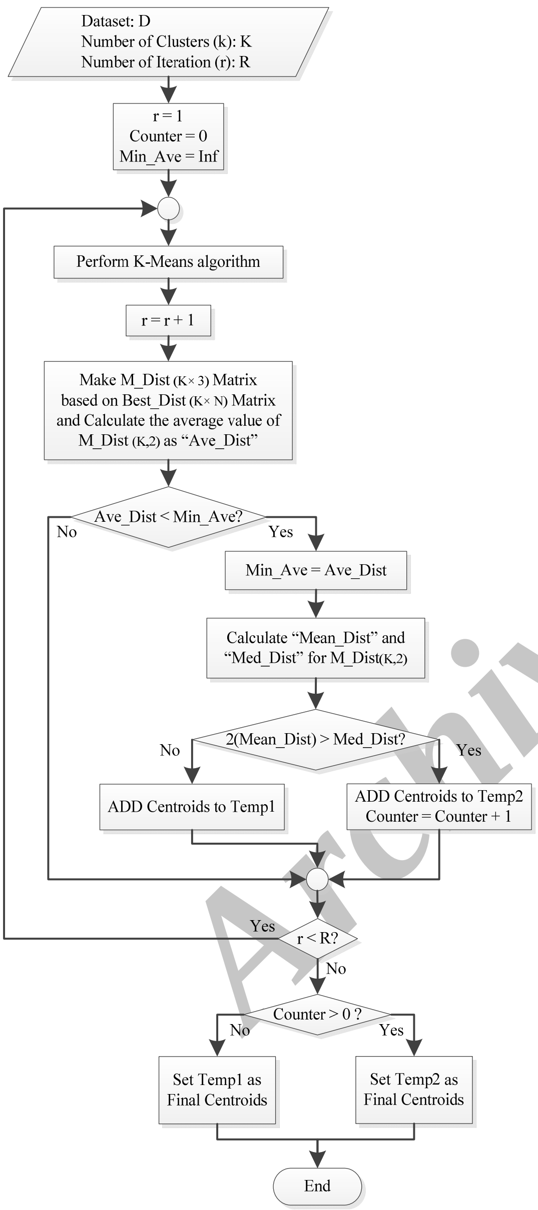

Fig. 4. Persistent K-Means flowchart

We must note that in the next iterations, only if the value of Ave_Dist in the corresponding iteration is less than Min_Dist, its value is recorded as a new minimum value in the Min_Dist variable. Otherwise, the results of that iteration is ignored and a new iteration of the algorithm is executed. After updating Min_dist, the second constraint is checked for the corresponding iteration. If it's satisfied, its result is stored in Temp2 and the value of the Counter variable is incremented; otherwise, the clustering result of that iteration is stored in Temp1. Subsequently, the algorithm checks whether there are more iterations to run? if Yes, it runs the next iteration. otherwise, it checks the Counter variable and if its value is larger than 0, it means that the second constraint has been satisfied with one of the iterations. Then the result of that iteration, stored in Temp2, is selected as the final output of the algorithm. If its value is not larger than 0, the last result stored in Temp1 is selected as the final result and the algorithm is stopped. In what follows, the results of Persistent K-Means clustering in the previous example are investigated.

Table 5 presents the results of Persistent K-Means (assuming r=10) on the corresponding dataset. The minimum value of Ave_Dist corresponds to iteration 10 of the algorithm. However, since the second constraint is not satisfied, the fifth iteration is returned as the final result of the algorithm. The reason is that in iteration 5, according to table 5, in addition to minimizing the Ave_Dist value, the second constraint is also satisfied. Since this constraint precedes the first condition, the result of this iteration is selected as the final output. The evaluation results based on accuracy (AC) measure [20] for 10 consecutive iterations of the algorithm shows the clustering precision of this iteration of Persistent K-Means algorithm.

Table 5

Persistent K-Means algorithm results in 10 rounds

Iteration No.	Ave_Dist	AC
1	317873556933.486	79.50
2	238935794873.245	70.50
3	256760408954.824	79.51
4	255168615089.361	79.82
5	86531408864.7107	99.90
6	163652149307.042	79.98
7	428409853900.782	78.98
8	259409970278.423	80.03
9	278407541804.449	79.37
10	76159749961.1842	90.33

Archive of SID

Fig. 5, and Fig. 6, respectively presents an overview of the results of Persistent K-Means for iterations 5 and 10. According to Table 5, the clustering accuracy for iteration 5, based on AC accuracy measure equals 99.9%, which is the maximum possible accuracy for 10 consecutive iterations of K-Means algorithm.

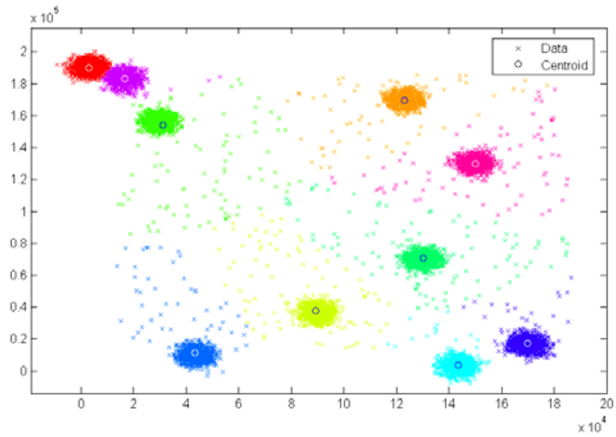

Fig. 5. Persistent K-Means algorithm results in (r=5)

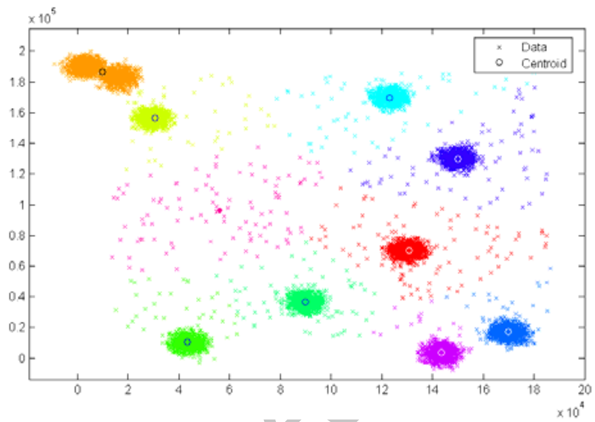

Fig. 6. Persistent K-Means algorithm results in (r=10)

According to Fig. 6, despite the fact that the minimum value of Ave_Dist is obtained through iteration 10, the algorithm does not have an acceptable performance in detecting clusters centroids. More specifically, selecting a cluster centroid from the less Centralized areas, as well as a centroid for two distinct clusters, its accuracy is 90.3% based on AC, which is less than that of iteration 5. The reason for this wrong selection is ignoring the second constraint.

Table 6

Results of $M\_Dist_{K\times3}$ matrix in (r=5)

K	$M\_Dist_{(K,1)}$	$M\_Dist_{(K,2)}$	$M\_Dist_{(K,3)}$	Ave
1	1019	21799480091.22	21393012.84712	0.27
2	1083	82748709007.44	76406933.52488	0.96
3	1113	123911945535.1	111331487.4529	1.40
4	1110	201863624709.5	181859121.3599	2.30
5	1134	144924755005.2	127799607.5883	1.61
6	1039	33126823778.59	31883372.26043	0.40
7	1070	101072219006.8	94460017.76343	1.19
8	1047	37120242601.35	35453908.88381	0.44
9	1039	25739118546.10	24772972.61415	0.31
10	1098	93007170365.61	84705983.93953	1.07

According to Table 6, Mean_Dist and Med_Dist of iteration 5 are:

Med_Dist :101626067.103523

Mean_Dist : 79006641.8234663

These results indicate the satisfaction of the second constraint in iteration 5. Accuracy evaluation results of Persistent K-Means clustering algorithm using a real dataset indicate the absolute superiority of the proposed algorithm in comparison to the classic version of K-Means algorithm and other improved versions of this algorithm.

In the third phase of running GBDC-P2P algorithm, each peer separately runs Persistent K-Means algorithm on its external data to obtain the final cluster centroids. Corresponding peer using the centroids obtained from this algorithm assigns its internal data to the closest cluster centroid. Eventually, the peer clusters its data independently from other peers and thus its local clusters are formed.

www.SID.ir

3. Case Study

This section focuses on evaluating the performance of Persistent K-Means centralized clustering algorithm. For this purpose, the result of Persistent K-Means algorithm is compared with the results of other algorithms on the Ruspini [26], IRIS and Wine and New Thyroid [27] real data sets.

It should be noted that in order to evaluate the Persistent K-Means centralized clustering algorithm, accuracy (AC) assessment [20] criterion is used that will be discussed briefly in the following. Also, more information about the real data set examined in this section is provided in Table 6.

Table 6

The real datasets used in the evaluation of Persistent K-Means.

Ref	Dataset Name	Number of Data	Number of Dimensi ons	Number of Cluster s
[20]	Ruspini	75	2	4
[13]	IRIS	150	4	3
[13]	NewThyroid	215	5	3
[13]	Wine	178	13	3

4-1.Assessment criteria of clustering accuracy "AC"

If C shows the results of real integer clustering, at the end of each round of the centralized clustering algorithm, K clusters of $C^P = \{C_1^P, C_2^P, C_3^P, \ldots, C_K^P\}$ ore extracted called calculated clusters (C^P). According to Equation (3), the AC criterion gives a number between zero and one. If the number is closer to one, it reflects the high accuracy of the concerned clustering algorithm (Xu, 2003). In equation (3), |D| is the total amount of network data. the map (c) function is used for mapping a calculated cluster C^P to the real peer cluster C. The $\delta(x, y)$ function is "1" if (x = y), otherwise, it returns the value of zero [5, 11].

$$AC = \frac{\sum_{d \in D} \delta(C(d), map(C^p(d)))}{|D|} \quad (3)$$

4-2.Persistent K-Means clustering algorithm evaluation results over real datasets

In this section, based on AC criterion, we compare the clustering accuracy of Persistent K-Means algorithm with, K-Means with random init, K-Medoids [12], Fuzzy C-Means [3]. We also compare the Persistent K-Means result with the several improved versions of K-Means algorithm, including Forgy, MacQueen, Kaufman [18], Refinement [17], MDC [20] and Improved Pillar K-Means [21]. Comparisons of results on real data sets of Ruspini in Fig. 7, IRIS in Fig. 8, Wine in Fig. 9, and New Thyroid in Fig. 10 are shown.

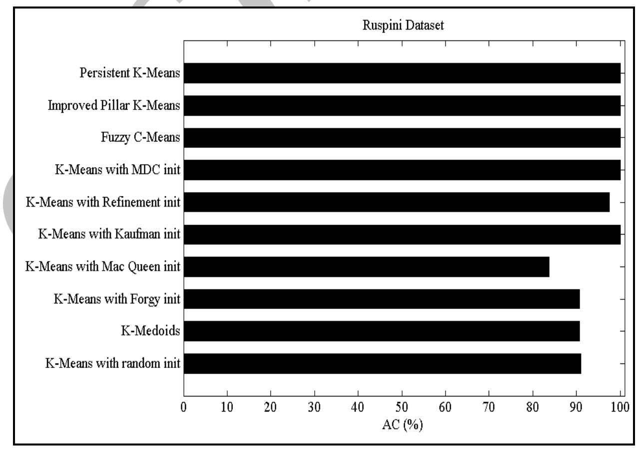

Fig. 7. Comparing the clustering results on Ruspini dataset

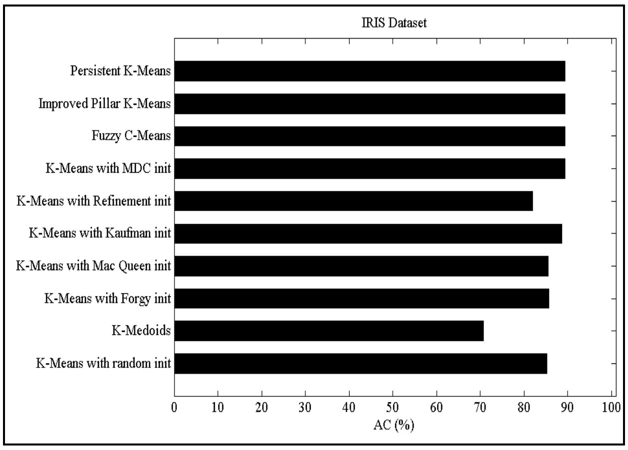

Fig. 8. Comparing the clustering results on IRIS dataset

Archive of SID

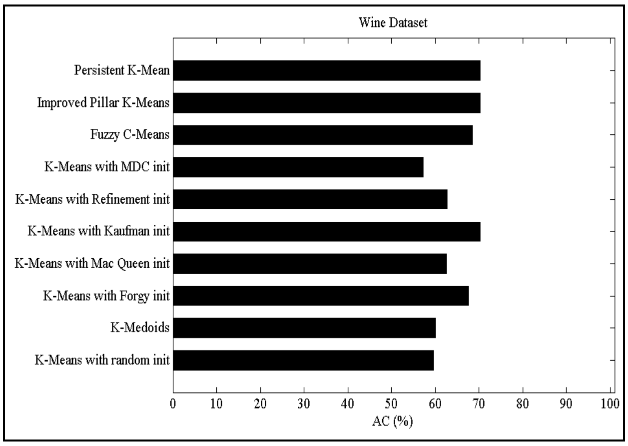

Fig. 9. Comparing the clustering results on Wine dataset

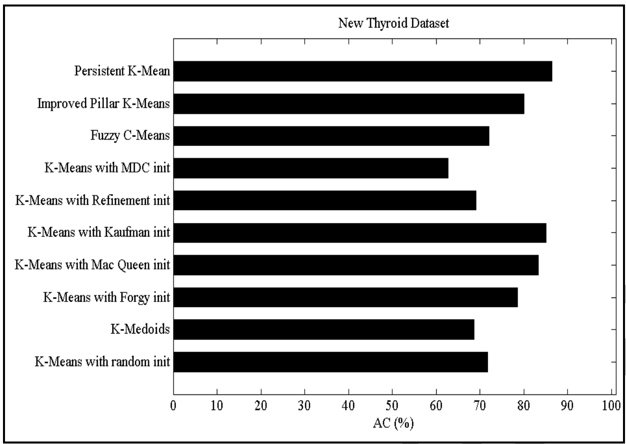

Fig. 10. Comparing the clustering results on New Thyroid dataset

The evaluation results indicate the absolute superiority of the results of Persistent K-Means algorithm compared to the K-Means classical version and its relative advantages in comparison with improved versions of K-Means algorithm. It should be noted that the results of evaluation of the proposed algorithm is provided for the default condition (r= 10).

4. Conclusion

In this paper inverse kinematic formulation of a radial symmetric (hexagonal) hexapod has been verified and demonstrated by an experimental hexapod robot. SiWaReL hexapod robot prototype and its design is discussed and the implementation process is studied. It is shown that a modular view for solving inverse kinematic problem and gait analysis for this kind of robot works well.

References

[1] S.M. Song and K. J. Waldron, Machines that walk: the adaptive suspension vehicle. The MIT Press, 1989.

[2] B. Siciliano and O. Khatib, Springer handbook of robotics. Springer, 2008.

[3] P. G. de Santos, E. Garcia, and J. Estremera, “Quadrupedal locomotion,” 2006.

[4] K. Waldron and R. McGhee, “The adaptive suspension vehicle,” Control Systems Magazine, IEEE, vol. 6, no. 6, pp. 7–12, 1986.

[5] K. Lilly and D. Orin, “Efficient dynamic simulation for multiple chain robotics mechanisms,” in Proceedings of 3rd annual conference aerospace computational control, Pasadena, 1989, pp. 73–87.

[6] Z. Wang, X. Ding, A. Rovetta, and A. Giusti, “Mobility analysis of the typical gait of a radial symmetrical six-legged robot,” Mechatronics, vol. 21, no. 7, pp. 1133–1146, 2011.

[7] S. Shah, S. Saha, and J. Dutt, “Modular framework for dynamic modeling and analyses of legged robots,” Mechanism and Machine Theory, vol. 49, pp. 234–255, 2012.

[8] M. Shahriari, K. G. Osguie, and A. A. A. Khayyat, “Modular framework kinematic and fuzzy reward reinforcement learning analysis of a radially symmetric six-legged robot,” Life Science Journal, vol. 10, no. 8s, 2012.

[9] M. Shahriari, “Design, implementation and control of a hexapod robot using reinforcement learning approach,” Master’s thesis, Sharif University of Technology, Int. Campus, 2012.

[10] M. Shahriari and A. A. Khayyat, “Gait analysis of a six-legged walking robot using fuzzy reward reinforcement learning,” in Fuzzy Systems (IFSC), 2013 13th Iranian Conference on. IEEE, 2013, pp. 1–4.

[11] T.-T. Lee, C.-M. Liao, and T.-K. Chen, “On the stability properties of hexapod tripod gait,” Robotics and Automation, IEEE Journal of, vol. 4, no. 4, pp. 427–434, 1988.

[12] M. Garc´ıa-Lopez, E. Gorrostieta-Hurtado, E. Vargas-Soto, J. Ramos-´Arregu´ın, A. Sotomayor-Olmedo, and J. Morales, “Kinematic analysis for trajectory generation in one leg of a hexapod robot,” Procedia Technology, vol. 3, pp. 342–350, 2012.

[13] G. Figliolini, S. Stan, and P. Rea, “Motion analysis of the leg tip of a six-legged walking robot,” in Proceedings of the 12th IFToMM World Congress, Besançon, France. Citeseer, 2007.

[14] F. Hardarson, “Locomotion for difficult terrain,” Dept. Mach. Des., Royal Inst. Technol., Stockholm, Sweden, Tech. Rep. TRITA-MMK, vol. 3, pp. 1400–1179, 1998.

[15] A. Mahajan and F. Figueroa, “Four-legged intelligent mobile autonomous robot,” Robotics and Computer-Integrated Manufacturing, vol. 13, no. 1, pp. 51–61, 1997.

Archive of SID

www.SID.ir

[16] Wikipedia, six-legged walking robot. [Online]. Available: http://en.wikipedia.org/wiki/Hexapod_(robotics)

[17] P. Gonzalez de Santos, E. Garc´ ´ıa Armada, J. A. Cobano, T. Guardabrazo et al., "Using walking robots for humanitarian de-mining tasks," 2004.

[18] J. E. Clark, J. G. Cham, S. A. Bailey, E. M. Froehlich, P. K. Nahata, R. J. Full, and M. R. Cutkosky, "Biomimetic design and fabrication of a hexapedal running robot," in Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on, vol. 4. IEEE, 2001, pp. 3643–3649.

[19] J. E. Clark, "Design, simulation, and stability of a hexapedal running robot," Ph.D. dissertation, stanford university, 2004.

[20] C. Prahacs, A. Saudners, M. K. Smith, D. McMordie, and M. Buehler, "Towards legged amphibious mobile robotics," Proceedings of the Canadian Engineering Education Association, 2011.

Archive of SID

www.SID.ir