Recommendation Model Based on K-means Clustering Optimization Neural Network

Lin Jinjian

IBM, GBS, Shenzhen, Guangdong, China

Keywords: k-means clustering; recommendation algorithm; software prediction; diagnosis

Abstract: Unsupervised algorithms, such as clustering algorithm, could be used on the fault tag position for fault prediction of software module. A software fault prediction algorithm based on quadtree k-means clustering algorithm was proposed in the paper. The purpose of adopting quadtree mainly included two aspects: the first was to seek for clustering center required by k-means clustering algorithm using quadtree, and the second was fault prediction of software module using quadtree. In this algorithm, input threshold parameter decided the initial clustering center. Through changing the threshold parameter, users could get the expected center of clustering. The performance of the algorithm was measured using such a new standard as “clustering earnings”. Through simulation and comparison, it was discovered that the algorithm proposed in the paper had highest clustering earnings. Moreover, in most cases, the total error rate of the algorithm proposed in the paper was lower than that of other algorithms, which indicated the effectiveness of the algorithm proposed in the paper in the prediction of software fault.

1. Introduction

K-means clustering algorithm is a widely used clustering algorithm, but it has its own disadvantages. Firstly, the user should initialize the quantity of category, but it is usually difficult to realize. Secondly, such algorithm needs to determine proper initial center of clustering. Thirdly, k-means algorithm is very sensitive to noise. A quadtree method is used for initialization of k-means algorithm in Literature [1]. Quadtree-based method can select proper initial center of clustering and eliminate the abnormal value, so it can overcome the second and the third disadvantages of k-means algorithm. In the paper, we should to focus on one practical problem in the application: the fault data of module is unknown. To solve such problem, researchers adopt one combined clustering method for clustering of modules, then some experienced experts judge whether the fault is contained according to the statistical characteristics of some representative points and data [2]. However, such method needs human participation in the prediction process, and experienced experts usually will not judge each category. In the paper, a software module fault prediction method as quadtree-based K-means algorithm was used. The main contents of the paper include seeking for initial clustering center of k-means algorithm using quadtree method and software module fault prediction method based on quadtree algorithm. Through changing the threshold parameter value, users can produce a group of expected clustering center and take it as the input parameter of k-means algorithm. Through comparing the total error rate of the algorithm proposed in the paper and other algorithms, it is discovered that the algorithm proposed in the paper has better performance in most cases. In the paper, the performance indicators of various prediction algorithms were measured using the clustering earnings. The clustering earnings of best k-means clusterer is similar with that based on quadtree algorithm. Thus, it is proved that the method proposed in the paper is very effective in the fault prediction of software module. To verify the performance of quadtree in the initial k-means algorithm, the initialization methods of quadtree-based k-means algorithm are compared with that of other two k-means algorithms [3-4]. Quadtree algorithm can get very good performance under all the parameters. Global k-means algorithm considers different data in each iteration, so when the data size is large, the complexity of algorithm will become an obstacle and the expandability of algorithm will be restricted.

2. Initialization of K-means Algorithm

2.1 Quadtree

Quadtree of 2D space is a quadtree that use the separation operator paralleling with coordinate axis for recursive decomposition of space. In each step, a square subspace is decomposed into four equal squares. Such data structure is called quadtree. Suppose the set O is defined on n-dimensional space μ, then the quadtree of O is defined as: $\mu=\left[d_{1\mu}:d_{1\mu}^{'}\right]\times\left[d_{2\mu}:d_{2\mu}^{'}\right]\times\cdots\left[d_{n\mu}:d_{n\mu}^{'}\right]$. According to the fact that the data points in any branch is lower than the set threshold, it is known the quadtree is made up of single branch and the set O and μ are saved. In each step, the set is decomposed into 2^n subsets. In the paper, we suppose $n=2$. As is shown in Fig. 1, suppose $\mu_{d_{1L}d_{2R}},\mu_{d_{1R}d_{2R}},\mu_{d_{1L}d_{2L}},\mu_{d_{1R}d_{2L}}$ means four subsets.

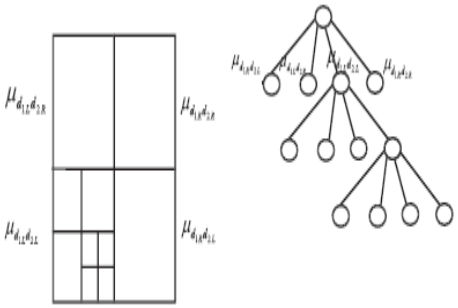

Fig. 1 Quadtree diagram of 2D space

Suppose $d_{1mid}=\left(d_{1\mu}+d_{1\mu}^{'}\right)/2$, $d_{2mid}=\left(d_{2\mu}+d_{2\mu}^{'}\right)/2$ and make the following definition:

$$O_{d_{1R}d_{2R}}=\left\{o\in O:o_{d_1}>d_{1mid},o_{d_2}>d_{2mid}\right\} \quad (1)$$

$$O_{d_{1L}d_{2R}}=\left\{o\in O:o_{d_1}>d_{1mid},o_{d_2}>d_{2mid}\right\} \quad (2)$$

$$O_{d_{1L}d_{2L}}=\left\{o\in O:o_{d_1}\le d_{1mid},o_{d_2}\le d_{2mid}\right\} \quad (3)$$

$$O_{d_{1R}d_{2L}}=\left\{o\in O:o_{d_1}>d_{1mid},o_{d_2}\ge d_{2mid}\right\} \quad (4)$$

Similarly, when $n=3$, it can be divided into eight subsets: $\mu_{d_{1L}d_{2R}d_{3L}},\mu_{d_{1L}d_{2R}d_{3R}},\mu_{d_{1R}d_{2R}d_{3L}},\mu_{d_{1L}d_{2R}d_{3R}}$, $\mu_{d_{1L}d_{2L}d_{3L}},\mu_{d_{1L}d_{2L}d_{3R}},\mu_{d_{1R}d_{2L}d_{3L}},\mu_{d_{1L}d_{2L}d_{3R}}$. For n–dimensional data, the decomposed subset can be expressed as: $\mu_{d_{1\alpha}d_{2\alpha}\ldots d_{n\alpha}},\alpha\in\left\{L,R\right\}$.

2.2 Initialization ethod of k-means algorithm

We firstly provide the expression method and parameter to be used in the next step:

MIN :The points of minimum data in each subtree defined by users

MAX : The points of maximum data in each subtree defined by users

δ :the user-defined nearest distance of seeking for neighbor

White substree: child node of MIN that the data points is fewer than the father nodes

Black subtree: child node of MAX that the data points is fewer than the father nodes

Grey subtree: the nodes between white subtree and black subtree;

R_k : the neighbor node of center c_k in the black subtree;

C : used as the clustering center of initial k-means algorithm.

Algorithm 1 is the initialization method of k-means algorithm. In Line 1 to 8, the original data are decomposed into subtrees. It is known all the subtrees are white subtrees or black subtrees, as is shown in Fig. 2a and 2b. Fig. 2a shows the one decomposition outcome of quadtree and there are three grey subtrees and one white subtree. Fig. 2b shows the further decomposition outcome of grey trees.

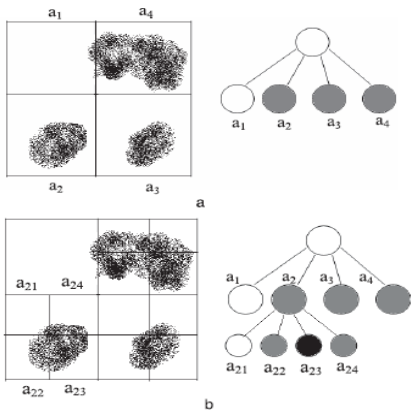

Fig. 2 Quadtree decomposition diagram of sets

Initialization method of k-means algorithm based on quadtree:

```
Input: Max%, Min%, Data set (O), δ
Output: Number of centers |C| and the centers C
1.  initialize the data space as a gray bucket;
2.  while there are gray buckets
3.          {
4.          select a bucket;
5.          divide it into 2^n sub buckets; // n is the dimension
6.          label the sub buckets as white leaf bucket, black leaf
            bucket or gray bucket;
7.          for every black leaf buckets calculate center (c_{i,(1≤i≤m)});
            // m is the number of black leaf buckets;
8.          }
9.  C = Φ;
10. label all centers c_{i,(1≤i≤m)} as unmarked;
11. for i = 1 to m do ℜ_i = c_i;
12. for each neighborhood ℜ_{i(1≤i≤m)}
13.         {
14.         if there exist an unmarked center in ℜ_i then
15.                 {
16.                 while there is an unmarked center c_k in ℜ_i
                    then
17.                         {
18.                         select c_k and label it as marked;
19.                         find δ-nearest unmarked neighbors
                            of c_k and include them in ℜ_i;
20.                         }
21.                 for all c_k ∈ ℜ_i calculate the mean m_i and call it
                    the cluster center;
22.                 C = C ∪ {m_i};
23.                 }
24.         }
25. return C and |C|;
```

In line 9, the clustering center is initialized into a null set; in Line 10, the centers of all the black subtrees obtained from Line 7 are marked as “unmarked”. The initialized domain set R_k includes the center $c_i\left(1 \le i \le m\right)$ of black subtree in Line 11. These domain sets will be decomposed into nearest neighbor set δ containing c_i. Such process is realized in Line 14 to 20. After one domain set has been decomposed, the mean of all the centers in R_i are calculated in Line 21 to 22, and they are contained in the clustering center C. Such process is implemented for all the neighbor sets marked with “unmarked”. It is noted, if the initialization center of one domain set is contained in other sets, such set will hardly be decomposed, but such set will be finally marked with the decomposition of other sets. In other words, the center of black subtree is grouped so that each group contains the center of adjacent black subtrees. Then, the means of each group is calculated and taken as the initialization center of k-means algorithm. At the end of iteration, the algorithm will return the quantity of set and center C. The output of quadtree algorithm in Algorithm 1 is the set of clustering center, and these centers are taken as the initial clustering center of k-means algorithm.

3. Algorithm Analysis

The complexity of black subtree is $(b+1)vc$, where, b is the depth of tree, v is the quantity of data points, and c is constant. m black tree domain sets are produced in Line 22 to 24, and it takes $O(m^2)$ time. Thus, the total complexity of algorithm is $O((b+1)vc+m^2)$. Suppose $m \ll v$,then the complexity becomes $O((b+1)v)$.

Standard of selecting parameter δ: to select parameter δ, we suppose $l_{\min}, l_{\max}$ are the lowest hierarchy and highest hierarchy produced by black subtree. Suppose p is the length of initial substree and n is the dimension. Then, $dia_{\max} = \sqrt{n}\,p/2^{l_{\min}}$, $dia_{\min} = \sqrt{n}\,p/2^{l_{\max}}$. δ is usually a number between $dia_{\max}$ and $dia_{\min}$.

4. Experiment and Results

Table 1 means the clustering earnings obtained using single k-means algorithm. The category adopted in the experiment is 12. For each category, six circulations are implemented and the maximum clustering earnings is written down. The initialization clustering center is selected at random. For quadtree-based method, there are four input parameters: MIN , MAX , O和δ. The value of MIN is 5%, and the value of MAX is 95%. In the quadtree algorithm, the value of δ in data AR3, AR4, AR5, Iris, SYD1 and SYD2 is $40,80,0.55,70,120$, and the quantity of clustering center is $3,3,2,3,3,4$. Column 5 in Table 2 shows the clustering earning value of different data using quadtree algorithm. To compre the performances of common k-means algorithm and quadtree algorithm proposed in the paper, the value of δ is adjusted to make the quantity of clustering center in these two algorithms be equal and maximize the clustering earnings in the k-means algorithm. Table 2 shows the comparison of predicted error between quadtree algorithm and other algorithms. KM, CT, DA, CS, NB and QDK are six different attributes of data.

Table 1 Cluster earning value of common k-means algorithm

C#	AR3	AR4	AR5	Iris	SYD1	SYD2
1	0000.000	0.000.000	0.000.000	0.0000	0.0000	0.0000
2	8482.825	7629.863	**5024.245***	266.366	6935.020	15769.25
3	**9487.919***	**10204.855***	4254.536	**270.66***	**11079.62***	18207.11
4	9483.899	9418.109	4197.171	270.614	10988.015	**20228.27***
5	9402.292	9536.031	4066.295	268.483	10939.179	20132.95
6	8679.119	9466.211	3928.565	269.172	10865.840	19909.28
7	8577.743	9408.413	3450.422	267.164	10769.540	19852.80
8	8252.662	9300.438	3501.186	267.982	10672.308	19692.30
9	8213.447	9218.340	3633.281	264.876	10594.040	19628.38
10	8068.498	9268.624	3496.519	261.874	10570.032	19472.81
11	8152.776	9113.923	3367.876	260.001	10564.431	19174.45
12	7818.184	9028.644	3160.925	261.595	10361.688	19096.63

*C#: Number of clusters, * Maximum gain value.*

Table 2 Analysis of prediction error of software fault

Dataset	Parameters	QDK%	KM%	CT%	CS%	NB %	DA%
AR3	FPR	34.54	34.54	44.09	43.63	09.00	3.60
	FNR	25.00	25.00	25.00	25.00	25.00	75.0
	Error	33.33	33.33	41.67	41.27	11.10	12.60
AR4	FPR	4.59	31.03	34.83	35.63	05.70	3.40
	FNR	45.0	20.00	20.00	20.00	55.00	60.00
	Error	12.14	28.97	32.06	32.71	10.20	14.00
AR5	FPR	14.28	14.28	28.92	32.14	14.20	07.14
	FNR	12.50	12.50	12.50	12.50	12.50	37.5
	Error	13.88	13.88	25.28	27.70	13.88	13.80

To compare the performances of initialized k-means algorithm using quadtree method, we make initialization experiment using quadtree, GM and DD. In DD algorithm, the distance parameter d is acquired through multiple circulations when the clustering center reaches the expected value. These distance parameters are given in Table 3. When the parameters needing evaluation is the iterations, total MSE, clustering earning and predicted error rate required when k-means algorithm reaches the convergence standard and these are given in Table 5.

Table 3 Outcome obtained using different initialization methods

Data	Techniques	N OI	SSE	Gain	Error
AR3	QDK (δ =40)	4	4110.337	9487.919	33.33
	KM*	6	4110.337	9487.919	33.33
	GM	592	4110.337	9487.919	33.33
	DD (d=100)	6	4110.337	9487.919	33.33
AR4	QDK (δ =80)	4	6596.423	8432.077	12.14
	KM*	8	5639.824	10204.85	28.97
	GM	1213	5639.824	10204.85	28.97
	DD (d=100)	7	5639.824	10204.85	23.36
AR5	QDK (δ =40)	5	2512.944	5024.245	13.88
	KM*	4	2512.944	5024.245	13.88
	GM	113	2512.944	5024.245	13.88
	DD (d=100)	3	2512.944	5024.245	13.88
Iris	QDK (δ =0.55)	10	97.3462	264.090	11.33
	KM*	11	97.3462	264.090	11.33
	GM	1726	97.3462	264.090	11.33
	DD (d=10)	13	97.3462	264.090	11.33
SYD 1	QDK (δ =70)	2	1513.813	110.79.62	03.11
	KM*	3	1513.813	110.79.62	03.11
	GM	793	1513.813	110.79.62	03.11
	DD (d=150)	4	1513.813	110.79.62	03.11
SYD 2	QDK (δ =120)	2	2158.433	20239.39	09.11
	KM*	4	2161.433	20228.27	10.56
	GM	1755	2161.433	20228.27	10.56
	DD (d=190)	3	2159.227	20230.11	10.44

*Values taken from the best of six runs.

5. Conclusion

In the paper, the effectiveness of quadtree-based K-means algorithm in the software fault prediction is evaluated and the outcome obtained using such method is compared with that using common k-means algorithm. The initial clustering center of k-means algorithm is sought by using quadtree method. In the case where the users expect to get K clustering centers of k-means algorithm, K initial clustering center can be obtained using quadtree method and further be taken as input parameters of k-means, which can be realized through changing the threshold parameter. The total error rate of software fault prediction using quadtree method is similar with that using other existing methods. In the paper, we compare the outcome of various algorithms.

References

[1] Arunkumar, N., Ram Kumar, K., Venkataraman, V. Automatic detection of epileptic seizures using permutation entropy, Tsallis entropy and Kolmogorov complexity [J]. Journal of Medical Imaging and Health Informatics, 2016, 6 (2):526-531.

[2] Fernandes, S.L., Gurupur, V.P., Sunder, N.R., Arunkumar, N., Kadry, S. A novel nonintrusive decision support approach for heart rate measurement [J]. Pattern Recognition Letters, 2017.

[3] Arunkumar, N., Mohamed Sirajudeen, K.M. Approximate Entropy based ayurvedic pulse diagnosis for diabetics - A case study [C]// Proceedings of the 3rd International Conference on Trendz in Information Sciences and Computing. 2011, 6169099:133-135.

[4] Yingyue Zhang, Ammar Algburi, Ning Wang, Vladyslav Kholodovych, Drym O. Oh, Michael Chikindas, and Kathryn E. Uhrich. Self-assembled Cationic Amphiphiles as Antimicrobial Peptides Mimics: Role of Hydrophobicity, Linkage Type, and Assembly State, Nanomedicine: Nanotechnology [J]. Biology and Medicine, 2017, 13(2):343-352.

[5] Du X, Shi Z, Peng Z, Zhao C, Zhang Y, Zhe W, Li X, Liu G, Li X. Acetoacetate induces hepatocytes apoptosis by the ROS-mediated MAPKs pathway in ketotic cows [J]. Journal of Cellular Physiology, 2017, 232(12):3296-3308.