Greedy heuristic algorithm for solving series of eee components classification problems*

L A Kazakovtsev[1,2], A N Antamoshkin[1], V V Fedosov[3]
[1]Siberian State Aerospace University named after Academician M.F.Reshetnev
31 “KrasnoyarskiyRabochiy” prospect, Krasnoyarsk, 660037, Russia.
[2]Siberian Federal University,
79, Svobodny Prospect, Krasnoyarsk, Russia
[3]”TTC – NPO PM” JSC
20, Molodezhnaya ul., Zheleznogorsk, Krasnoyarskiy kray, 662970, Russia

E-mail: levk@bk.ru

Abstract. Algorithms based on using the agglomerative greedy heuristics demonstrate precise and stable results for clustering problems based on k-means and p-median models. Such algorithms are successfully implemented in the processes of production of specialized EEE components for using in space systems which include testing each EEE device and detection of homogeneous production batches of the EEE components based on results of the tests using p-median models. In this paper, authors propose a new version of the genetic algorithm with the greedy agglomerative heuristic which allows solving series of problems. Such algorithm is useful for solving the k-means and p-median clustering problems when the number of clusters is unknown. Computational experiments on real data show that the preciseness of the result decreases insignificantly in comparison with the initial genetic algorithm for solving a single problem.

Introduction

Supplying the electronic units of the complex technical systems with the highest quality EEE components increases the whole system reliability. Moreover, for reaching the highest reliability of an electronic unit, the EEE components of the same type must have equal characteristics which assure their coherent operation. The highest homogeneity of the characteristics is reached if the EEE components are produced as a single production batch from a single batch of the raw materials [1]. The critically important units are integrated from EEE components manufactured as a special production lots with special quality requirements [2, 3].

The characteristics of each EEE device in the lot are checked using destructive and nondestructive tests [2, 4]. Resulting data of such tests are used for analyzing the lot homogeneity [4]. For splitting the EEE components into several assumed production batches, the k-means method is used [5, 6, 7].

* Results were obtained in the framework of the state task № 346 of the Ministry of Education and Science of the Russian Federation

Manufacturers of the EEE components in the United States of America and European Union produce components of special quality classes, Military and Space [8, 9]. Manufacturers in the Russian Federation do not form a special class of components for use in space systems [2, 3].

The k-means problem can be classified as a continuous problem of the location theory [10, 11, 12]. The aim is to find *k* points (centers, centroids) in a *d*-dimensional space such that the total squared distance from each of the data vectors (known points, measurement result vectors) to the nearest of *k* chosen centers reaches its minimum:

$$\arg \quad \min \quad F(X_1,\dots,X_k) = \sum_{i=1}^{N} \underline{\min}_{j\in\{\overline{1,k}\}} \left\|X_j - A_i\right\|^2.$$

The aim of a continuous p-median problem (also known as k-median problem) [11] is finding k points (called centers or medians) such that the sum of weighted distances from *N* known points (data vectors) to the nearest of *k* centers reach its minimum.

$$\arg \quad \min \quad F(X_1,\dots,X_k) = \sum_{i=1}^{N} w_i \underline{\min}_{j\in\{\overline{1,k}\}} L(X_j, A_i).$$

Here, wi are the weight coefficients, *L*() is some distance function between two given poins. Usually, *L*() is a metric in the *d*-dimensional space.

Continuous location problems with Euclidean, Manhattan (rectilinear), Tschebychev metrics a well investigated (all such metrics are special cases of metric based on Minkovski l_p-norms [13]), many authors propose algorithms for solving p-median problems with such metrics and their special case, 1-median problem (also called the Weber problem) which is used for solving the p-median problem. In particular, well known Weiszfeld procedure [44] was generalized for metrics based on Minkovski norms.

Traditionally, the problem with the Euclidean metric $L(X_j,A_i)=\sqrt{\sum_{k=1}^{d}(x_{j,k}-a_{i,k})^2}$ is called p-median problem. Here, $X_j=(x_{j,1},\dots,x_{j,k})\ \forall j=\overline{1,p}$, $A_i=(a_{i,1},\dots,a_{i,k})\ \forall i=\overline{1,N}$. In case of the squared Euclidean metric $L(X_j,A_i)=\sum_{k=1}^{d}(x_{j,k}-a_{i,k})^2$ and weight coefficients $w_i=1\forall i=\overline{1,N}$, we have the k-means problem.

Known methods

The k-means method uses the ALA procedure (Alternating Location-Allocation) which includes two simple steps:

Algorithm 1. ALA procedure.

Required: data vectors $A_1 \dots A_N$, *k* initial cluster centers $X_1 \dots X_k$.

1. For each center X_i, determine its cluster C_i as a subset of the data vectors for which this center X_i is the closest one.
2. For each cluster C_i, recalculate its center X_i (i.e., solve the Weber problem).
3. Repeat Step 1 unless Steps 1, 2 made no change in any cluster.

Except special cases, the k-means and p-median problems are NP-hard and require global search [15].

The result of the ALA procedure depends on the choice of initial cluster centers. Known k-means++ algorithm [16] has an advantage in comparison with the random choice of the initial centers with uniform distribution. However, this improvement preciseness is insufficient for many practically important problems which need more precise results. For such cases, researchers propose various recombination techniques for initial center sets [10].

The ALA procedure can be optimized with use of many techniques. For example, sampling procedures [17] solve the k-means problem for the randomly selected subset of the data vectors and use the achieved result as an initial set of centers for solving the original problem.

The dependence of the results of the ALA procedure on the initial centers seeding is a serious problem for the reproducibility of the classification algorithm results: depending on the initial centers seeding, different algorithm starts classify the same data vectors as elements of various clusters. For the EEE component production batches classification problem, this means that various EEE component belong to the same or different production batches depending on the initial seeding. Thus, an algorithm for solving k-means problem which returns a stable result is preferred.

Rather precise but extremely slow, the Information Bottleneck Clustering method (IBC) is a deterministic method for solving the cluster analysis and classification problems able to achieve perfect results in many cases.[18]. This algorithm starts from considering each data vector as a separate cluster. Then, clusters are removed one-by-one until the desired clusters quantity remains. Each time, the algorithm eliminates such cluster that its elimination gives the smallest increment of the objective function value. For the k-means and p-median problems, this algorithm eliminates the cluster center which gives the smallest total distance from data vectors to the closest remaining centers.

The genetic algorithms (GAs) with greedy agglomerative heuristic initially designed for the discrete k-median problem on a network [19] are compromise variants by preciseness, stability of the results and time consumption. In [20, 12], author propose an approach for adaptation of these algorithms for the continuous location problems:

Algorithm 2**.** GA with greedy heuristic and floating point alphabet.

Required: Set $V = (A_1, \dots, A_N) \in \mathbb{R}^d$, quantity of clusters p, GA population size N_p.

1: Create N_p sets of coordinates $\chi_1, \dots, \chi_{N_p}$: $\chi_i \subset \mathbb{R}^d$, $|\chi_k| = p \forall k = \overline{1, N_p}$ whica are results of the ALA procedure. Thus, each χ_i is a local minimum of the solved problem. Store the values of the objective functions to an array of variables $\mathcal{F}_1, \dots, \mathcal{F}_{N_p}$.

2: If the stop conditions are reached then go to Step 8.

3: Randomly choose two “parent” sets χ_{k_1} and χ_{k_2}, $k_1, k_2 \in \{\overline{1}, N_p\}$, $k_1 \neq k_2$. Running algorithm 3, obtain the "child" sets of coordinates χ_c which are local minimums of the objective function. Store the value of the objective function $\mathcal{F}_c$.

4: If $\exists k \in \{\overline{1, N_p}\}$: $\chi_k = \chi_c$ then go to Step 2.

5: Choose an index $k_{worst} = \operatorname{argmax}_{k=\overline{1,N_p}} \mathcal{F}_k$. If $\mathcal{F}_{wotst} < \mathcal{F}_c$ then go to 2.

6: Randomly choose two indexes k_1 and k_2, $k_1 \neq k_2$; array $k_{worst} = \operatorname{argmax}_{k \in \{k_1, k_2\}} \mathcal{F}_k$.

7: Swap the values of $\chi_{k_{worst}}$ and χ_c, store $\mathcal{F}_{k_{worst}} = \mathcal{F}_c$ and go to Step 2.

8: STOP. Result is set χ_k^*, $k^* = \operatorname{argmin}_{k=\overline{1,N_p}} \mathcal{F}_k$.

Modification of the greedy agglomerative heuristic procedure for this GA is as follows.

Algorithm 3. Greedy crossingover heuristic for Algorithm 2.

Required: Set $V = (A_1, \ldots, A_N) \in \mathbb{R}^d$, quantity of clusters p, two "parent" sets of centers χ_{k_1} and χ_{k_2}, values σ_e and L_{min}.

1: Join sets $\chi_c = \chi_{k_1} \cup \chi_{k_2}$. Run the ALA procedure for $|\chi_c|$ clusters starting from the solution χ_c. Store its result in χ_c.

2: If $|\chi_c| = p$, then run the ALA procedure with the initial solution χ_c, then STOP and return result χ_c.

2.1: Calculate the distances from each data vector to the nearest element of χ_c.

$$d_i = \min_{X \in \chi_c} L(X, A_i) \forall i = \overline{1, N}.$$

For each data vector, determine the closest center from χ_c.

$$C_i = \operatorname{argmin}_{X \in \chi_c} L(X, A_i) \forall i = \overline{1, N}.$$

Calculate distances from each data vector to the second nearest element (center) in χ_c.

$$D_i = \min_{Y \in (\chi_c \backslash \{C_i\})} L(Y, A_i).$$

3: For each $X \in \chi_c$, calculate $\delta_X = F(\chi_c \backslash \{X\}) = \sum_{i:C_i]X} (D_i - d_i)$.

4.1: Calculate n_δ.

$$n_\delta = \max\{[(|\chi_c| - p) * \sigma_e], 1\}.$$

Sort values of δ_X and choose a subset $\chi_{elim} = \{X_1, \ldots, X_{n_\delta}\} \subset \chi_c$ from n_δ of coordinates corresponding to minimum values of δ_X.

4.2: For each $j \in \{\overline{2, |\chi_{elim}|}\}$, if $\exists k \in \{\overline{1, j-1}\}: L(X_j, X_k) < L_{min}$ then remove X_j from χ_{elim}.

4.3: Store $\chi_c = \chi_c \backslash \chi_{elim}$.

4.4: Reallocate data vectors between closest centers.

$$C_i^* = \operatorname{argmin}_{X \in \chi_c} L(X, A_i) \forall i = \overline{1, N}.$$

4.5. For each $X \in \chi_c$, if $\exists i \in \{\overline{1, N}\}: C_i = X \; and \; C^*i \neq X$ then recalculate center X^* кластера $C_X^{clust} = \{A_i | C_i^* = X, i = \overline{1, N}\}$. Set $\chi_c = (\chi_c \backslash \{X^*\}) \cup \{X\}$.

5: Go to Step 2.

Experimentally, authors [12] determine optimal values of parameters σ_e=0.25 and $L_{min} = \min_{X \in \chi_c} \{\max\{L(X, X_j), L(X, X_k)\}\}$.

New algorithm

Most algorithms such as ALA procedure or rather efficient genetic algorithm with recombination of fixed length center sets [21] require given value of the clusters quantity *p*. Other algorithms such as *X*-means [22] choose the best value of clusters number *p* with use of a special criterion. The adequacy of such criteria is a separate complex problem. Below, we propose a simple modification of the greedy heuristic: after reaching the quantity of clusters *p*, the process of elimination of the centers from the interim solution does not stop and algorithm fixes the values of the objective function for each value of centers quantity. Thus, we can obtain solutions for series of problems with $p = \overline{2, p_{max}}$.. However, the maximum value of clusters quantity p_{max} must be given.

The algorithm below is a combination of such greedy agglomerative heuristic with the genetic algorithm. However, such heuristic can be used with other global search strategies [23]

```
Algorithm 4. GA with greedy heuristic for solving series of problems with $p=\overline{2,p_{max}}$.
1. Initialization of a population of $N_{pop}$ individuals. Each individual is a set of $p_{max}$ centers (we denote it $X$ and $X_i$ is the $i$th element of this set). Set $F_{new,j}$= +∞ for each $j=\overline{1,N_{pop}}$. Initialize the arrays of the objective function values $F^*_k$ = +∞ and best folutions $X^*_k$={} for each $k=\overline{2,p_{max}}$.
2. Select randomly $j_1, j_2 \in [1,N], j_1 \neq j_2$
3. $X_{new} = X_{j_1} \cup X_{j_2}$
4. While $|X_{new}| > p_{max}$:
    4.1. Select an element $j$ such that its elimination results in minimum increase of the objective function: $j = \arg\min_{j\in\chi_{\mathrm{new}}} F(\chi_{\mathrm{new}}\backslash\{j\})$
    4.2. $X_{new} = X_{new}\backslash\{j\}$. Continue iterations 4.
5. Set $F_{new} = 0$ ; $X^*$=$X_{new}$.
6. While $|X_{new}| > 2$:
    6.1. Set $F_{new} = F_{new} + f(X_{new})$; $k$=$|X_{new}|$ ; $F_k$= $f(X_{new})$; if $F_k$< $F^*_k$ then set $F^*_k$ = $F_k$;
    6.2. Perform Steps 4.1 and 4.2 for $X_{new}$. Continue iterations 6.
7. Choose $j_3$ using tournament selection by value of $F_{new,j}$. Set $F_{j_3} = F_{new}$; $X_{j_3} = X^*$, $F_{new,j_3} = F_{new}$.
8. Check the stop conditions, go to 2.
```

Computational experiments

Computational results are shown in Table 1. For the continuous p-median problems, comparatively precise results are achieved when the quantity of clusters is large. Having decreased this quantity p, we obtain comparatively worse results. Thus, new algorithm is efficient for : $p = \overline{p_{max,}\lceil p_{max}/3\rceil}$. After reaching the minimum number of clusters $\lceil p_{max}/3\rceil$, the algorithm must be restarted for obtaining results with less number of clusters.

For the EEE components classification problems, it is usually enough to solve the problems with p∈{2..10}. Such problems can be successfully solved by a single run of the new algorithm.

Conclusion

New genetic algorithm can be efficiently used for solving k-means and p-median problems which arise during the process of EEE components classification by homogeneous production batches. Computational experiments demonstrate the preciseness of the new algorithm and stability of its results (minimum standard deviation) which is very important for technical problems with high price of an error.

Table 1. Computational results for the p-median and k-means problems

Data set and its parameters	*p* and distance metric	Algorithm (see comments)	Time, sec..	Average result	Std. deviation
Test results of the electronic chip 1526TL1, N=1234, d=120 (in case of new algorithm, p∈{2..20})	*p*=14, l_2^2	ALA multistart Sheng, Liu+ALA New algorithm	15 15 15 (for all problems)	150,124869801 149,954679652 149,78736565*	0,384203928 0,172789313 0,03157532*
	p=10, l_2^2	ALA multistart Sheng, Liu+ALA New algorithm	15 15 15 (for all problems)	198,375350991 198,377650812 198,35974703*	0,018643710 0,024878118 $2\cdot10^{-14}$*
	p=6, l_2^2	ALA multistart Sheng, Liu+ALA New algorithm	15 15 15 (for all problems)	362,70701636* 362,70401636* 362,704051312	0* 0* 0*
UCI Mopsi Joensuu, N=6014, d=2,. (in case of new algorithm , p∈{2..20})	*p*=10, l_2	ALA multistart Sheng, Liu+ALA New algorithm	15 15 15 (for all problems)	359,680203232 359,545250068 359,41046080*	3,964320582 2,526439494 0,177992934*
	p=4, l_2	ALA multistart Sheng, Liu+ALA New algorithm	15 15 15 (for all problems)	596,825210394 596,82520843* 596,825283111	0,000000442 0,000000388 0*
BIRCH-3, N=100000, d=2, (in case of new algorithm, p∈{2..110})	*p*=100, l_2^2	ALA multistart New algorithm	30 30 (for all problems)	$3{,}7513245\cdot10^{13}$ $3{,}740432\cdot10^{13}$*	116786778766 21699776156*
	p=50, l_2^2	ALA multistart New algorithm	30 30 (for all problems)	$9{,}0099578\cdot10^{13}$ $8{,}902789\cdot10^{13}$*	9545892119 0*
	p=20, l_2^2	ALA multistart New algorithm	30 30 (for all problems)	$3{,}303278\cdot10^{14}$* $3{,}3049972\cdot10^{14}$	0* 0*

Comments:” ALA multistart” means multiple starts of the ALA procedure with random initial seeding, “Sheng,Liu+ALA” means running genetic algorithm with recombination of the fixed length subsets powered by the ALA local search [21], “New algorithm” means Algorithm 4. The best results are marked by “*”.

References

[1] Kazakovtsev L A, Antamoshkin A N, Masich I S 2015 Fast Deterministic Algorithm for EEE Components Classification *IOP Conf. Series: Materials Science and Engineering*, Volume 94, article 012015, 10 pages. doi:10.1088/1757-899X/94/1/012015

[2] Fedosov V V, Orlov V I. 2011 Minimal necessary extent of examination of microelectronic products at inspection test stage *Izvestiya Vuzov. Priborostroenie*, Volume 54(4), pp. 62-68.

[3] Kharchenko V S, Yurchenko Yu B 2003 Rating of fault-tolerant onboard complexes frames at usage electronic components industry *Tekhnologiya I konstruirovanie v elektronnoy apparature*, 2, pp. 3-10.

[4] Kazakovtsev L A, Orlov V I, Stupina A A, Masich I S 2014 Problem of electronic components classifying *Vestnik SibGAU*, issue 4(56), pp. 55-61.

[5] Ackermann M R et al. 2012 StreamKM: A Clustering Algorithm for Data Streams *J. Exp. Algorithmics*, Volume 17, 2.4:2.1-2.30.

[6] Kanungo T et al. 1999 Computing nearest neighbors for moving points and applications to clustering *Proc.of the tenth annual ACM-SIAM symp. on Discrete algorithms (Society for Industrial and Applied Mathematics)*, pp 931-932.

[7] Kazakovtsev L A, Stupina A A, Orlov V I 2014 Modification of the genetic algorithm with greedy heuristic for continuous location and classification problems *Sistemy upravleniya i informatsionnye tekhnologii*, No. 2(56), pp. 31–34.

[8] Hamiter L 1991 The History of Space Quality EEE Parts in the United States *ESA Electronic Components Conf., ESTEC (Noordwijk, The Netherlands, 12–16 Nov 1990 ESA SP-313)*

[9] Kirkconnell C S et al. 2014 High Efficiency Digital Cooler Electronics for Aerospace Applications *Proc. SPIE 9070, Infrared Technology and Applications XL 90702Q (June 24, 2014)*

[10] Farahani R Z, Hekmatfar M (editors) 2009 *Facility Location: Concepts, Models, Algorithms and Case Studies* (Berlin Heidelberg: Springer-Verlag)

[11] Kazakovtsev L A, Stupina A A 2014 Fast Genetic Algorithm with Greedy Heuristic for p-Median and k-Means Problems *IEEE 2014 6th Int. Congress on Ultra Modern Telecommunications and Control Systems and Workshops ICUMT (St.-Petersburg)*, pp. 702-706.

[12] Kazakovtsev L A, Antamoshkin A N 2014 Genetic Algorithm wish Fast Greedy Heuristic for Clustering and Location Problems *Informatica*, Volume 38(3), pp. 229-240.

[13] Deza M M 2013 Metrics on Normed Structures *Encyclopedia of Distances* (Berlin Heidelberg:Springer), pp. 89-99. doi: 10.1007/978-3-642-30958-85.

[14] Weiszfeld E 1937 Sur le point sur lequel la somme des distances de n points donnes est minimum *Tohoku Mathematical Journal*, Volume 43(1), pp.335-386.

[15] Cooper L 1963 Location-allocation problem *Oper. Res.*, Volume 11, pp. 331-343.

[16] Arthur D, Vassilvitskii S 2007 k-Means++: the Advantages of Careful Seeding *Proc. of the eighteenth annual ACM-SIAM symp. on Discrete algorithms*, pp. 1027-1035.

[17] Mishra N, Oblinger D, Pitt L 2001 Sublinear time approximate clustering *12th SODA*, pp. 439-447.

[18] Sun Zh et al. 2014 A parallel clustering method combined information bottleneck theory and centroid-based clustering *The Journal of Supercomputing*, Volume 69(1), pp.452-467.

[19] Alp O, Erkut E, Drezner Z 2003 An Efficient Genetic Algorithm for the p-Median Problem *Annals of Operations Research*, Volume 122, pp.21-42.

[20] Neema M N, Maniruzzaman K M, Ohgai A 2011 New Genetic Algorithms Based Approaches to Continuous p-Median Problem *Netw. Spat. Econ.*, Volume 11, pp. 83-99.

[21] Sheng W A, Liu X 2004 Genetic k-medoids clustering algorithm *Journal of Heuristics*, Volume 12(6), pp. 447-466.

[22] Pelleg D, Moore A 2000 X-means: Extending k-means with efficient estimation of the number of clusters *Seventeenth Internat. Conf. on Machine Learning*, pp. 727–734.

[23] Kazakovtsev L A, Antamoshkin A N 2015 Greedy heuristic method for location problems *Vestnik SibGAU*, Volume 16(2), pp. 317-325.