CLUSTER ANALYSIS OF SOME CLASSES OF OBJECTS BY APPLICATION OF THE TEST RECOGNITION ALGORITHMS

Tsvetanka Kovacheva

Abstract: *The paper treats the task for cluster analysis of a given assembly of objects on the basis of the information contained in the description table of these objects. Various methods of cluster analysis are briefly considered. Heuristic method and rules for classification of the given assembly of objects are presented for the cases when their division into classes and the number of classes is not known.*
The algorithm is checked by a test example and two program products (PP) – learning systems and software for company management. Analysis of the results is presented.

Key words: *cluster analysis, classification, data analysis, heuristics, heuristic algorithms, heuristic rules.*

1. Cluster Analysis Problem Definition

Let $E=(E_1,E_2,...,E_m)$ be the set of objects, described by a set of characteristics $x_1,x_2,...,x_n$. Objects classification is not known a priori. On the rows of the table

$$T_{m,n,s}=\{a_{ij}\},\ \ i=\overline{1,m},\ \ j=\overline{1,n} \qquad (1)$$

correspond objects, but of the columbs – their characteristics (a_{ij} - the value of the j -th characteristic of the i -th object). The characteristics take values 1 and 0. It is taken, that missing data are already restored from one of methods proposed in [7].

The problem definition is to divide a given group of objects $E=\{E_1,E_2,...,E_m\}$ into s disjoint classes $C=\{C_1,C_2,...,C_s\}$

$$C=\bigcup_{k=1}^{s}C_k \text{ , as } C_u\bigcap C_v=\varnothing \text{ , in the case of } u\neq v\text{ , } u,v=\overline{1,s} \qquad (2)$$

by use of the test recognition algorithms A_R .

The following *quantitative measures* for division into classes are used, characterizing the in formativeness of the features and objects [6], [1]:

informational weights of the features

$$p_j=\frac{\tau_j}{\tau},\ \ j=\overline{1,n}\ ,\quad q_j=\frac{1}{\tau}\sum_{v=1}^{n}\frac{\tau_j^v}{v},\ \ j=\overline{1,n}\ ,\quad r_j=\frac{1}{\theta}\sum_{i=1}^{m}\sum_{v=1}^{n}\frac{\theta_{ij}^v}{v},\ \ j=\overline{1,n} \qquad (3)$$

where - τ_j - number of irreducible tests (IT) containing j – th characteristic;

- τ - number of irreducible tests of matrix $T_{m,n,s}$;
- τ_j^v - number of IT with length v containing j – th characteristic $(\sum_{j=1}^{n}\tau_j^v=\tau_j)$;
- θ - number of irreducible representative set (IRS);
- θ_{ij}^v - number of IRS with length v containing j – th characteristic in the *i*-th object .

informational weights of the objects E_i

$$IP_i=\sum_{j=1}^{n}a_{ij}p_j,\ \ i=\overline{1,m}\ ,\qquad IQ_i=\sum_{j=1}^{n}a_{ij}q_j,\ \ i=\overline{1,m}\ ,\ \ IR_i=\sum_{i=1}^{n}a_{ij}r_j,\ \ i=\overline{1,m} \qquad (4)$$

The following A_R ={A1, A2,…,A8} *test algorithms* are applied for object recognition and for adjustment of the boundaries of the classes [6], [1]:

A1 – A3 algorithms for classification according to the informational weight of the object with use of the weights p_j, q_j *and* r_j of the features x_j respectively;

A4 – A6 algorithms for classification according to the minimal average distance to a class with use of the weights p_j, q_j *and* r_j of the features x_j respectively;

A7 algorithm, based on the principle of voting of the set of irreducible tests (IT);

A8 algorithm, based on the principle of voting for the assembly of the irreducible representative set (IRS), where the voting is effected for each object according to the contained by it IRS, i.e. the repetition factor (multiplicity) of IRS is taken into account.

Solution of A_j *algorithm* for a given object – the number of the class to which the algorithm assigns the given object or a bar ("-") in case of classification rejection.

The membership of the boundary objects to the respective class is studied during the clarification of the boundaries.

Boundary objects – the first and the last object in the arrangement of the objects within the framework of the class.

The following coefficient is introduced in order to evaluate the quality of the applied algorithms for determination of the class of each object:

$$KR_j = \frac{n_j}{m}, \quad j = \overline{1,8} \tag{5}$$

where - n_j - number of the correctly classified to their class objects by the A_j algorithm;

- m - total number of objects.

The objective is: to establish algorithms and heuristic rules, which allow, without assistance of experts, to:

divide the given set t optimal number of classes;

define the best algorithm for classification of objects.

2. Methods for Cluster Analysis

The task for Cluster Analysis (CA) includes two stages:

first stage – formation of relatively remote from each other groups of adjacent objects, according to the information about the distances or the links (proximity measures) among them;

second stage – independent classification not only of already known objects (from a given sequence) but also of new objects.

[2], [9] and [8] consider various CA methods, their specifics, problems and development. Their variety is generated by the large number of possible methods for calculation of the distances among the individual features and clusters, as well as by the optimum assessment manifold of the final cluster structure. Most of the classification procedures are heuristic and have no strict statistical justification.

The following *two groups of CA methods* have found widest application:

➢ *agglomerative hierarchical algorithms:*

- *merging algorithms* – the objects are considered initially as individual clusters.

The distances among them are calculated according to a given metrics. Then the process of consecutive agglomeration (adjunction) is initiated step by step. The last step merges all objects in a cluster. A dendogram is built and the number of the clusters is defined in accordance with the maximal jumps.

- *divisive algorithms* – the whole sample is considered as one cluster in the initial stage and then begins the component sectionalization process. The division continues until each object becomes a separate cluster. The divisive algorithms are of two types: *monothetic* (classification on the basis of maximal in formativeness feature) and *polythetic* (all features accounted).

- *iteration methods* of aggregation - the K-means method is mostly used. The required number of final clusters should be set.

3. CA algorithms with application of test algorithms

3.1. Agglomeration-hierarchical joining method [1]

The algorithm of this method is the following:

Step 1. It is assumed that each $E_i,\ i=\overline{1,m}$ object of the basic table $T_{m,n,s}$ is a class.

m = *s* different $\{C_i\},\ i=\overline{1,m}$ classes are derived, each with one object.

Step 2. Sequential subtraction of each E_i object from $T_{m,n,s}$.

Step 3. Classification of the E_i object according to the remaining after the subtraction table $T_{m-1,n,s}$, made by the objects $\{E_1,E_2,...,E_{i-1},E_{i+1},...,E_m\}$, with the help of the respective recognition algorithm A_R, applying the following rules:

- if more than half of the algorithms indicate one and the same class for the studied object, then it is associated to this class;
- in case the answers are less than or equal (to the half), then the studied object is not classified to any of the considered classes and remains alone.

Step 4. Joining of objects, which mutually associate to the largest extent, while the rule is either by majority or by examination of the table with the numbers of the classes, as a second stage of recognition.

Step 5. Repetition of Step 2 – Step 4, until the equality of the number of classes is derived for two sequential cycles.

The application of this method in case of large amount of features and objects leads to the following difficulty: the computer time and the process of merging objects into clusters has slow convergence. The following divisive algorithm is suggested in order to accelerate the clustering process of a given set of objects.

3.2. Divisive method for separation into classes by sequential bipartition of the objects in two classes

The method is based on creation of a sequence of embedded divisions. It includes *two stages*:

- *first stage* – arrangement of the objects in descending order according to the informational weights calculated in agreement with formulae (4), as the objects with larger informational weights are of higher rank and vice versa. Sequential division of the objects into groups and examination of the boundary objects for membership to the given set.
- *second stage* – adjustment of the membership of all objects to the defined classes.

There are various criteria for division of objects into groups – division by two of each group, division according to a given threshold – for example maximal difference of the informational weights of the objects, etc.

The following *heuristic rules* are observed in the application of the method for division by two:

Rule 1 Division of the considered group of objects into two subgroups (classes):

- at *m* = 2*k* - both classes consist of equal number of objects;
- at $m=2k+1$ – it is assumed that the first class has one object more.

Rule 2 A given class is not divided in case one of its objects remains alone in a class.

Rule 3 The studied boundary object is associated to the class that is recognized by no less than half of the A_R algorithms.

Rule 4 In case a boundary object of a given class is recognized to a not neighboring class, then these classes are merged together with the intermediate classes.

Rule 5 If the boundary can not be fixed (the object is recognized at times to one class, at others to another class), then this boundary is eliminated.

Rule 6 If a boundary object is alone in a class, then it is not examined.

The algorithm of this method is the following:

Step 1. Initial ranking of the objects $E_i,\ i=\overline{1,m}$ in descending order according to the number of ones in the rows of Table $T_{m,n,s}$.

Step 2. Division of the considered objects in accordance with *Rule 1 and Rule 2* according to the order of the entered classes.

Step 3. Calculation of the informational weights of the features and objects in accordance with formulae (3) and (4).

Step 4. Objects $E_i,\ i=\overline{1,m}$ ranking in descending order according to their informational weights $IP_i,\ IQ_i$ or IR_i within the framework of each class.

Step 5. Adjustment of the boundaries among the obtained classes, by examination of the boundary objects with test algorithms and defining their class membership:

- sequential separation of each of the boundary objects E_i from the table;
- determination by A_R of the class of E_i from the remaining table after the separation and application of *Rule 3*.

In case a given boundary object passes to a neighbor class, the boundary shifts and *Step 3* is applied.

The process continues with application of *Rule 4 – Rule 6* until fixing, boundary removal or merging of classes.

Step 6. Inspection of the stability of the remaining borders among the classes by application of *Rule 4 – Rule 6.*

Step 7. Return to Step 2 until obtaining division with maximal possible number of classes for the considered objects, or approaching the maximal number of classes of the program with the test algorithms.

Step 8. Clarification of the membership of all objects to the defined classes. If some object is recognized belonging to another class, then it is moved to it in the table and it is proceeded to *Step 3*.

The process continues until all objects are recognized by their class. If this could not be achieved, the division goes a step back (with less number of classes). The program which uses test algorithms requires object separation into classes in advance and then calculation of their informational weights. Therefore, the initial arrangement of the objects is made according to the number of the ones for each object, as the best objects have the greatest number of ones, while the worse objects have the smallest number of the same. It is possible, in this arrangement for an object with smaller numbers of ones but with higher informational weight, to fall in a neighbor class. When the objects are divided into two groups, their information weights are calculated with the test algorithms and they are ranked in the framework of the class, the object is moved to the border. In the process of verification of the boundary objects it goes in its class and the boundary is shifted. The original arrangement of the objects decreases the number of transpositions at the initial division in two classes.

3.3 Methods based on calculation of distances [5]

The following methods were used for comparison with the suggested method:

➢ nearest neighbor – smallest distance method that uses the smallest distance among objects in two groups u и v

$$d(r,s)=\min(dist(a_{ui},a_{vj})),\ \ i=(\overline{1,n_u}),\ \ j=(\overline{1,n_v}) \tag{6}$$

➢ furthest neighbor - largest distance method that uses the farthest distance among objects in two groups u и v

$$d(r,s)=\max(dist(a_{ui},a_{vj})),\ \ i=(\overline{1,n_u}),\ \ j=(\overline{1,n_v}) \tag{7}$$

➢ average distance method – uses the average distance between all couples of objects in group u and group v

$$d(r,s)=\frac{1}{n_u.n_v}\sum_{i=1}^{n_u}\sum_{j=1}^{n_v}dist(a_{ui},a_{vj}) \tag{8}$$

where - n_u, n_v - are the numbers of the objects in groups u and v accordingly.

The distances in formulae (6) – (8) are determined among the very informational weights of the objects.

The algorithm of this method is the following:

Step 1. Calculation of the informational weights of the features and of the objects according to formulae (3) and (4).

Step 2. Division of the group of objects to a preliminary set number of classes, by the three methods.

4. Example

The suggested algorithm is applied for division of classes of objects in one test example and two specific types of program systems.

Example 1: test example.

The table with the descriptions of the objects is modeled so that to obtain more clear discrimination of the informational weights of the objects, for three classes at least. The set of objects $E = \{E_1, E_2, ..., E_{10}\}$ is described by 14 features $x_1, x_2, ..., x_{14}$ (Table 1.1).

Example 2: Author's system [4].

The set of objects $E = \{E_1, E_2, ..., E_9\}$ is described by 14 features $x_1, x_2, ..., x_{14}$ (Table 1.2).

Example 3 : Software for company management [3]

The set of objects $E = \{E_1, E_2, ..., E_9\}$ is described by 14 features $x_1, x_2, ..., x_{15}$ (Table 1.3).

Example 1												Table 1.1		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
E1	1	1	0	1	0	1	1	1	1	1	1	1	1	1
E2	1	1	1	0	1	0	1	1	1	1	1	1	1	1
E3	0	1	1	1	1	1	1	1	1	0	0	1	1	1
E4	1	0	0	0	0	1	1	1	1	1	1	1	1	1
E5	1	1	0	0	0	0	1	1	1	1	1	1	1	1
E6	0	1	1	0	1	1	1	0	0	0	0	1	1	1
E7	1	0	0	0	1	1	1	1	0	1	0	1	0	0
E8	0	0	0	0	1	0	1	1	0	1	0	0	0	0
E9	0	0	0	0	0	0	0	0	1	1	0	1	0	1
E10	0	0	0	0	0	0	0	1	0	0	0	1	0	0

Author's system												Table 1.2		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
E1	1	1	0	1	0	1	1	1	1	1	1	1	1	1
E2	0	1	1	1	1	1	1	1	1	0	0	1	1	1
E3	1	0	0	0	0	1	1	1	1	1	1	1	1	1
E4	1	0	0	1	1	1	1	1	0	1	0	1	0	0
E5	1	0	1	0	1	1	0	0	0	1	0	1	0	1
E6	0	0	0	1	0	0	0	1	0	1	0	1	1	0
E7	0	0	0	0	1	0	1	1	0	1	0	0	0	0
E8	0	1	0	0	0	0	0	0	0	0	0	0	0	0
E9	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Software for company management													Table 1.3		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
E1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0
E5	1	1	1	1	0	0	1	0	1	1	1	1	0	1	0
E6	0	0	1	0	1	1	1	1	1	1	1	1	0	0	0
E9	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1
E8	1	0	1	1	0	1	0	1	1	1	1	1	0	0	0
E4	1	0	1	0	1	1	1	1	1	0	1	0	1	0	0
E3	1	0	1	1	1	1	1	1	0	1	0	0	0	0	0
E7	1	0	1	1	1	1	1	1	0	0	0	0	0	0	0
E2	1	0	0	1	1	0	1	1	0	1	0	0	0	0	0

5. Calculation and analysis of the results.

The algorithm in 3.2 is applied for the three examples. Tables 2.1 – 2.3 give the weights of the features calculated in accordance with formulae (3) and Tables 3.1 – 3.3 give the informational weights of the objects calculated according to formulae (4).

Weights of the features %(x10)														Table 2.1
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
p	89	68	57	104	120	94	63	104	78	63	73	21	36	31
q	91	67	54	100	105	85	55	109	104	58	80	15	46	33
r	101	88	64	43	78	87	52	78	96	74	74	15	86	65

Weights of the features %(x10)														Table 2.2
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
p	39	39	85	190	128	93	58	66	58	54	31	93	31	35
q	44	42	87	192	130	100	55	64	57	47	32	86	32	33
r	88	75	73	80	98	100	46	91	76	65	58	42	57	50

Weights of the features %(x10)															Table 2.3
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
p	65	71	108	39	58	51	68	37	86	158	61	62	55	37	44
q	63	71	104	43	59	56	66	35	85	160	63	64	53	35	43
r	75	83	17	78	61	73	86	37	57	101	101	77	41	37	75

Informational weights of the objects										Table 3.1
objects	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
IP(E)	0,824	0,803	0,776	0,652	0,626	0,554	0,49	0,35	0,193	0,125
IQ(E)	0,843	0,817	0,773	0,676	0,658	0,518	0,46	0,327	0,21	0,124
IR(E)	0,859	0,871	0,752	0,728	0,729	0,485	0,535	0,282	0,25	0,093

Informational weights of the objects									Table 3.2
objects	E1	E2	E3	E4	E5	E6	E7	E8	E9
IP(E)	0,832	0,752	0,581	0,528	0,443	0,343	0,263	0,194	0
IQ(E)	0,831	0,754	0,574	0,531	0,438	0,344	0,265	0,196	0
IR(E)	0,758	0,733	0,548	0,583	0,438	0,328	0,285	0,091	0

Informational weights of the objects									Table 3.3
objects	E1	E2	E3	E4	E5	E6	E7	E8	E9
IP(E)	0,803	0,43	0,584	0,59	0,755	0,69	0,43	0,67	0,68
IQ(E)	0,806	0,43	0,586	0,58	0,754	0,69	0,43	0,67	0,68
IR(E)	0,745	0,44	0,528	0,55	0,787	0,61	0,43	0,62	0,74

The sequence of division of the objects into classes is given in Tables 4.1 – 4.3 for the three examples, while the changes in the informational weights of the objects are in Tables 5.1 – 5.3 and Tables 6.1 – 6.3 present the results of the recognition of all objects by the A_R algorithms. Table 4.1 shows that the optimal division of the objects of *Example 1* into 3 classes is found for 11 steps:

$$I\ class = \{E2, E1, E4, E5\},\ II\ class = \{E3, E6\},\ III\ class = \{E7, E8, E9, E10\}$$

This division is achieved when after testing of each object for class membership, the second class object E7 has moved to third class. Each object is recognized to its class by the test algorithms, during the division (Table 6.1).

Table 4.1																
1	2		3	4	5	6	7		8		9		10		11	
introducing of a new limit (2 cl.)	introducing of a new limit (3 cl.)	arrangement	moving of the limit	moving of the limit	introducing of a new limit (4 cl.)	moving of the limit	merger (return to 2 cl.)	arrangement	introducing of a new limit (3 cl.)	arrangement	introducing of a new limit (4 cl.)	arrangement	moving of the limit	arrangement	division of 3 cl.	division of 3 cl. - transposition
E1	E1	E3	E3	E3	E3	E3	E3	E1	E1	E2	E2	E1	E1	E1	E2	E2
E2	E2	E1	E1	E1	E1	E1	E1	E2	E2	E1	E1	E2	E2	E2	E1	E1
E3	E3	E2	E2	E2	E2	E2	E2	E4	E4	E4	E4	E4	E4	E4	E4	E4
E4	E4	E4	E4	E4	E4	E4	E4	E5	E5	E5	E5	E5	E5	E5	E5	E5
E5	E5	E5	E5	E5	E5	E5	E5	E3	E3	E3	E3	E3	E3	E3	E3	E3
E7	E7	E6	E6	E7	E7	E7	E7	E7	E7	E7	E7	E6	E6	E7	E7	E6
E6	E6	E7	E7	E6	E6	E6	E6	E6	E6	E6	E6	E7	E7	E6	E6	E7
E8	E8	E8	E8	E8	E8	E8	E8	E9	E9	E8	E8	E8	E8	E9	E8	E8
E9	E9	E9	E9	E9	E9	E9	E9	E8	E8	E9	E9	E9	E9	E8	E9	E9
E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10	E10

Table 4.2			
1		2	
introducing of a new limit (2 cl.)	arrengement	introducing of a new limit (3 кл.)	arrengement
E1	E1	E1	E1
E2	E3	E3	E3
E3	E2	E2	E2
E4	E5	E5	E4
E5	E4	E4	E5
E6	E6	E6	E6
E7	E7	E7	E7
E8	E8	E8	E8
E9	E9	E9	E9

Table 4.3	
1	
introducing of a new limit (2 cl.)	arrengement
E1	E5
E5	E1
E6	E9
E9	E8
E8	E6
E4	E4
E3	E3
E7	E2
E2	E7

The optimal division of the objects into 3 classes is found within two steps for *Example 2* (Table 4.2 and Table 5.2):

$$I\ class = \{E1, E3, E2\}\ ,\ II\ class = \{E4, E5\}\ ,\ III\ class = \{E6, E7, E8, E9\}$$

The results of the recognition of each object to the obtained classes are presented in Table 6.2.

The division of the objects into 2 classes is found yet during the first step for *Example 3* (Table 4.3 and Table 5.3):

$$I\ class = \{E1, E5, E6, E9, E8\}\ ,\ II\ class = \{E4, E3, E7, E2\}$$

It is evident from Tables 4.1–4.3 and Tables 5.1–5.3, after calculation and analysis of the differences in the informational weights of adjacent objects that the maximal informational difference is not a factor determining the division into classes.

Due to this reason, *Rule 1* for object division is suggested. The tests of the algorithm with application of the distances among the objects on the basis of the informational weights as a criterion (threshold) for their division into classes, show a larger number of iterations.

Table 5.1

1	2		3	4	5	6	7		8		9		10		11	
introducing of a new limit (2 cl.)	introducing of a new limit (3 cl.)	arrangement	moving of the limit	moving of the limit	introducing of a new limit (4 cl.)	moving of the limit	merger (return to 2 cl.)	arrangement	introducing of a new limit (3 cl.)	arrangement	introducing of a new limit (4 cl.)	arrangement	moving of the limit	arrangement	division of 3 cl.	division of 3 cl. - transposition
0,824	0,745	0,895	0,892	0,814	0,786	0,768	0,73	0,886	0,844	0,871	0,746	0,781	0,872	0,872	0,871	0,838
0,803	0,72	0,745	0,835	0,816	0,852	0,863	0,886	0,886	0,871	0,844	0,781	0,746	0,731	0,731	0,844	0,816
0,776	0,895	0,72	0,671	0,805	0,825	0,833	0,886	0,823	0,757	0,757	0,569	0,569	0,639	0,639	0,757	0,697
0,652	0,485	0,485	0,535	0,672	0,72	0,738	0,823	0,774	0,714	0,714	0,527	0,527	0,602	0,602	0,714	0,654
0,626	0,465	0,465	0,507	0,623	0,678	0,696	0,774	0,73	0,771	0,771	0,858	0,858	0,81	0,81	0,771	0,783
0,554	0,455	0,57	0,556	0,519	0,561	0,585	0,69	0,69	0,625	0,625	0,509	0,557	0,495	0,521	0,625	0,537
0,49	0,57	0,455	0,492	0,488	0,519	0,504	0,512	0,512	0,565	0,565	0,557	0,509	0,521	0,495	0,565	0,538
0,35	0,32	0,32	0,321	0,344	0,325	0,336	0,373	0,394	0,351	0,355	0,297	0,297	0,283	0,285	0,355	0,347
0,193	0,155	0,155	0,15	0,218	0,255	0,289	0,394	0,373	0,355	0,351	0,256	0,256	0,285	0,283	0,351	0,267
0,125	0,085	0,085	0,1	0,132	0,182	0,198	0,254	0,254	0,188	0,188	0,164	0,164	0,144	0,144	0,188	0,112

Table 5.2

1		2	
introducing of a new limit (2 cl.)	arrengement	introducing of a new limit (3 кл.)	arrengement
0,79	0,79	0,86	0,86
0,67	0,68	0,75	0,75
0,68	0,67	0,61	0,61
0,54	0,54	0,51	0,61
0,54	0,54	0,61	0,51
0,35	0,35	0,44	0,44
0,32	0,32	0,37	0,37
0,07	0,07	0,05	0,05
0	0	0	0

Table 5.3

1	
introducing of a new limit (2 cl.)	arrengement
0,68	0,72
0,72	0,68
0,45	0,68
0,68	0,53
0,53	0,45
0,51	0,51
0,44	0,44
0,36	0,42
0,42	0,36

In the course of division of the objects from *Example 1* into 3 classes by weights IP_i, IQ_i and IR_i (Table 7.1), with the help of the distances methods of 3.3, five objects display the following division:

$$I\ class = \{E1, E2, E3, E4, E5\},\ II\ class = \{E6, E7\},\ III\ class = \{E8, E9, E10\}$$

It does not coincide with the division obtained by the method described in 3.2 (Table 4.1).

The division of the objects into 3 classes by the distances methods (Table 7.2) does not give an univocal division for *Example 2*. The three weight algorithms IR_i give the following division:

$$I\ class = \{E1, E2\},\ II\ class = \{E3, E4, E5, E6, E7\},\ III\ class = \{E8, E9\}.$$

It also does not coincide with the division obtained by the method described in 3.2 (Table 4.2).

The algorithms A_R are used for recognition of each object in the obtained division of the objects into 3 or 4 classes according to the three methods of distances. The A_R algorithms do not substantiate the membership of all objects to their classes. However, after relevant consecutive transpositions until each object is recognized in its class, the objects are divided into classes, as it is achieved with the proposed divisive algorithm.

Table 6.1									
N	A1	A2	A3	A4	A5	A6	A7	A8	cl.
E2	1	1	1	1	1	1	1	1	1
E1	1	1	1	1	1	1	1	1	1
E4	1	1	2	1	1	1	1	1	1
E5	1	1	1	1	1	1	1	1	1
E3	1	1	1	2	2	2	2	2	2
E6	3	3	3	2	2	2	2	2	2
E7	2	2	2	3	3	3	3	3	3
E8	3	3	3	3	3	3	3	3	3
E9	3	3	3	3	3	3	3	3	3
E10	3	3	3	3	3	3	3	3	3

Table 6.2									
N	A1	A2	A3	A4	A5	A6	A7	A8	cl.
E1	1	1	1	1	1	1	1	1	1
E3	1	1	1	1	1	1	1	1	1
E2	1	1	1	1	1	1	1	1	1
E4	1	1	1	2	2	2	2	2	2
E5	2	2	2	2	2	2	2	2	2
E6	3	3	3	3	3	3	3	3	3
E7	3	3	3	3	3	3	3	3	3
E8	3	3	3	3	3	3	3	3	3
E9	3	3	3	3	3	3	3	3	3

Table 6.3									
N	A1	A2	A3	A4	A5	A6	A7	A8	cl.
E5	1	1	1	1	1	1	1	1	1
E1	1	1	1	2	2	1	2	2	1
E9	2	1	1	2	1	1	2	2	1
E8	1	1	1	1	1	1	1	1	1
E6	1	1	1	1	1	1	-	2	1
E4	1	2	2	2	2	2	2	1	2
E3	2	2	2	2	2	2	2	2	2
E2	2	2	2	2	2	2	2	2	2
E7	2	2	2	2	2	2	2	2	2

Table 7.1										
3 cl. - weights IP	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
Shortest distance	1	1	1	1	1	1	1	2	3	3
Largest distance	1	1	1	2	2	2	2	3	3	3
Average distance	1	1	1	2	2	2	2	3	3	3
3 cl .- weights IQ	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
Shortest distance	2	2	2	2	2	1	1	3	3	3
Largest distance	3	3	3	3	3	1	1	1	2	2
Average distance	1	1	1	1	1	2	2	3	3	3
3 cl. - weights IR	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
Shortest distance	1	1	1	1	1	2	2	3	3	3
Largest distance	1	1	1	1	1	2	2	3	3	3
Average distance	1	1	1	1	1	2	2	3	3	3

Table 7.2								
E1	E2	E3	E4	E5	E6	E7	E8	E9
1	1	2	2	2	2	2	2	3
1	1	2	2	2	2	3	3	3
1	1	2	2	2	2	3	3	3
E1	E2	E3	E4	E5	E6	E7	E8	E9
1	1	2	2	2	2	2	2	3
1	1	2	2	2	2	3	3	3
3	3	3	3	1	1	1	1	2
E1	E2	E3	E4	E5	E6	E7	E8	E9
1	1	2	2	2	2	2	3	3
1	1	2	2	2	2	2	3	3
1	1	2	2	2	2	2	3	3

On the basis of the results of the classification of the objects in their ultimate grouping in classes (Tables 6.1-6.3), the KR_j coefficient for each A_R algorithm is calculated according to formula (5).

The following algorithms have the highest value of the coefficient for the considered examples (Table 8): *Example 1*, *Example 2* - A4-A8, *Example 3* - A2, A3, A6, A8.

The algorithm with the highest number of votes (coefficient) from the group of algorithms is selected for operation with a specific PP.

							Table 8
	A1	A2	A3	A4	A5	A6	A7	A8
Example 1	0.70	0.70	0.60	1	1	1	1	1
Example 2	0.89	0.89	0.89	1	1	1	1	1
Example 3	0.70	1	1	0.714	0.857	1	0.571	0.428

Conclusion.

The proposed divisive method is convergent to the sought solution. In the course of examination of the company management software and the learning systems the division is achieved after the first and second steps, respectively.

The use of the test algorithms allows self-checking of the division and particularization of the membership of the objects to the respective class. These algorithms account at each division step the changes of the weights of the features and objects.

The method has a simple way of division of the objects into classes. On the basis of the operations – arrangement of the objects according to the number of the ones in their descriptions conforming to their quality in the initial stage, sequential division of the objects into two classes, examination of the obtained stability limits, merging of classes and ranking of the objects in the obtained classes according their informational weights and of the introduced heuristic rules to effect the operations, the method allows division of a given set of objects into classes without experts.

By contrast with the methods using distances (3.3) the proposed method does not require preliminary definition of the number of classes. The method determines the maximum possible number of classes for division of the objects as well as the limits for division into less number of classes.

Bibliography

[1] Angelova V. “Evaluation quality of programme products through classification methods”, Autoreferat, Sofia, 1987.

[2] Cluster Analysis – main notions and terms, introduction and synopsis of methods, http://www.doctor.ru/biometr/lib/cluster/

[3] “Company management software”, PC WORLD Bulgaria, VII, 2001, 18-23.

[4] Eskenasi A., R. Radev “Evaluation of the quality of authoring systems”, Education&Computing 5 (1989), 43-47.

[5] Jambu M. “Classification automatique pour l’analyse des données – méthodes et algorithmes”, M., “Finance and Statistics”, 1988 (in Russian).

[6] Juravlev J. I., M. M. Kamilov, Sh. Tuljganov “ Algorithms of calculation of evaluations and their using”, “FAN”, Russian, 1974.

[7] Kovacheva Ts. “Heuristic algorithms for object evaluation by incomplete data”, Comptes rendus de l'academie des sciences, v. 56,2. Sofia, 2003.

[8] Schulcloper J. R. “Logical Combinatorial Pattern Recognition”, IRIS Laboratory, http://iristown.engr.utk.edu

[9] Zagorujko N. G. “About restoring of missing data”, Laboratory of Data Analysis, http://lda.tsu.tula.ru/cluster/demo1_3e.htm

Author information

Tsvetanka Kovacheva - Department Mathematics, Technical University – Varna, e-mail: ts_kovacheva@hotmail.com