Evolutionary Approach to Multimodal Clustering on Formal Contexts

Mikhail Bogatyrev[1] [0000−0001−8477−6006], Sergey Dvoenko[1] [0000−0001−8673−7617], Dmitry Orlov[1], and Tatyana Shestaka[1]

[1]Tula State University, 92 Lenin ave., Tula, Russia
okkambo@mail.ru

Abstract. Evolutionary approach to multimodal clustering on multidimensional formal contexts is proposed. Its main advantage is that it allows one to find a well-grounded number of clusters corresponding to the global or close to the global extremum of the function that characterizes the quality of solution of the clustering problem. This approach is effective when the proximity measure for a data being clustered is not Euclidean. Formal context is the data model in Formal Concept Analysis, the area in data analysis where mathematically rigorous methods from lattice theory have been applied for discovering relationships on heterogeneous data. Taking into account the effect of data heterogeneity in cluster analysis can be effectively implemented using multimodal clustering methods. The paper contains main definitions from Formal Concept Analysis, description the principle of evolutionary computation and evolutionary approach to multimodal clustering. Experimental study of proposed approach is performed on the task of phenotyping of disease of myocardial infarction.

Keywords: Evolutionary Computation, Formal Context, Multimodal Clustering.

1 Introduction

One of the features of the data used in the area of data intensive processing is heterogeneity. Accordingly, in the methods of intensive data processing, it is necessary to take into account their heterogeneity, which makes it possible to preserve knowledge presented in data sets obtained on domains of different nature.

There are a lot of works, for example [1-3], devoted to clustering heterogeneous data and cluster interpreting. Taking into account the effect of data heterogeneity in cluster analysis can be effectively implemented using multimodal clustering methods. Classical cluster analysis is based on dividing a single set of objects into disjoint subsets that are clusters. At the same time, despite the wide variety of proximity measures of objects used here, the problem of interpreting the obtained clusters remains urgent in cluster analysis. The clusters interpretation is always performed in the context of the applied proximity measure, and for heterogeneous data from different domains, such an interpretation based on a single proximity measure will obviously not be correct.

Multimodal clustering involves not one, but several sets of objects to be clustered simultaneously. Such sets can be formed by heterogeneous data. A multimodal cluster is a subset in the form of combinations of objects from different sets. The very fact of the combination of certain objects in a multimodal cluster can carry important information and serve as the basis for clusters interpretation.

The simplest variant of multimodal clustering is biclustering, which was first proposed in [4]. Biclustering algorithms work on data in the form of matrices, the rows of which contain instances of objects with attributes (features) located in columns. Biclustering methods have been applied in various fields of data analysis, but especially in bioinformatics, in the task of studying gene expression [5]. Here namely biclusters make it possible to objectively assess the mutual influence of genes on biological processes.

Biclustering data in the form of an object-attribute representation is also used in Formal Concept Analysis (FCA) [6]. FCA is the paradigm of conceptual modeling which studies how objects can be hierarchically grouped together according to their common attributes. Such grouping of objects is really biclustering of them. The output of standard FCA algorithms is conceptual lattice which contains hierarchically linked formal concepts which are biclusters [7]. FCA methods for constructing concept lattices differ from the biclustering methods used, for example, in the analysis of gene expression, and FCA methods expand the possibilities of interpreting clusters. The hierarchy of concepts in the lattice reflects a hierarchy of data that is not obvious in the original view. The use of concept lattices also makes it possible to build association rules and functional dependencies on clustered data.

Triclustering is a generalization of biclustering [8]. However, the appearance of the third set in the data presentation fundamentally changes the situation, and triclustering algorithms are not built by simple scaling of biclustering ones. An overview of triclustering algorithms can be found in [9]. In [10], the analysis of triclustering algorithms implemented using FCA is performed.

The generalization of the approach based on the construction of concept lattices to the n-dimensional case of multimodal clustering is presented in [11]. The most well-known algorithm for n-dimensional multimodal clustering is the DataPeeler algorithm [12], which is often used as a benchmark for comparison with other algorithms. The novelty of this work lies in the development of an approach to multimodal data clustering based on the use of evolutionary calculations. The fact that this approach is used on the data model in the form of formal contexts allows us to naturally take into account the heterogeneity of data.

Evolutionary computation has been applied in clustering [13, 15]. Their main advantage is that the evolutionary algorithms allow one to find a reasonable number of clusters corresponding to the global or close to the global extremum of the function that characterize the quality of solution of the clustering problem.

This approach is also effective when there is no Euclidean proximity measure for clustered data, which is relevant for heterogeneous data.

The rest of the paper is organized as follows. In Section 2, there is brief description of multimodal clustering on formal contexts. It contains main definitions from FCA and explanation why triclustering is not scalable biclustering. Section 3 contains the description of the main contribution of this paper as evolutionary algorithm of multimodal clustering on formal contexts. In the Section 4, the results of experimental study of proposed approach are presented. They are illustrated on the task of phenotyping of disease of myocardial infarction.

2 Multimodal Clustering on Formal Contexts

We perceive multimodal clustering on formal context which is the data model in FCA. That is why we briefly consider the main issues of the FCA [6, 10, 12]. Classical FCA deals with two basic notions: *formal context* and *concept lattice.* Formal context is a triple $\mathbf{K} = (G, M, I)$ where G is a set of objects, M – set of their attributes, $I \subseteq G \times M$ – binary relation which represents facts of belonging attributes to objects. Formal context may be represented by [0, 1] - matrix $\mathbf{K} = \{k_{i,j}\}$ in which units mark correspondence between objects $g_i \subseteq G$ and attributes $m_j \subseteq M$. The concepts in the formal context have been determined by the following way. If for subsets of objects $A \subseteq G$ and attributes $B \subseteq M$ there are exist mappings which are realized as prime operators $A' : A \to B$ and $B' : B \to A$ with the following properties of completeness

$$\left.\begin{array}{r} A' := \{m \in M| < g, m > I \in \text{for all } g \in A\} \\ B' := \{g \in G| < g, m > I \in \text{for all } m \in B\} \end{array}\right\} \quad (1)$$

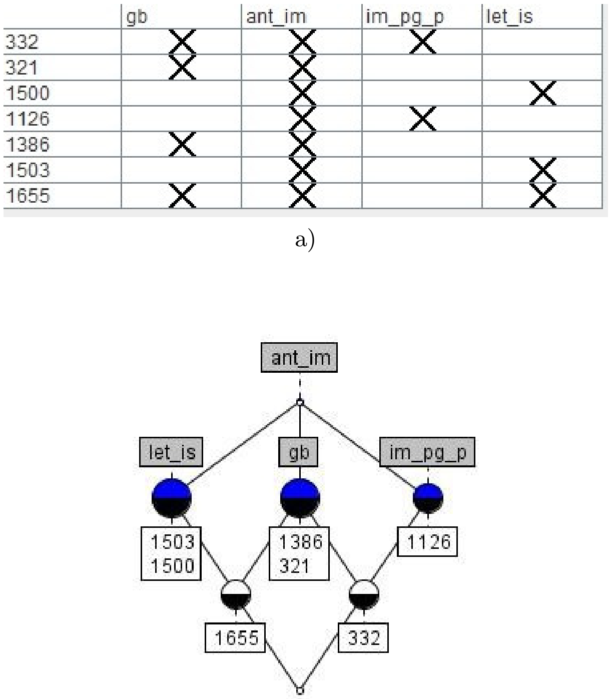
then the pair (A, B) that $A' = B, B' = A$ is named as *formal concept.* The composition of mappings demonstrates following properties of A and B: (A, B) that $A'' = B, B'' = A$; A and B is called the *extent* and the *intent* of a formal context $\mathbf{K} = (G, M, I)$ respectively.

By other words, a formal concept is a pair (A, B) of subsets of objects and attributes which are connected so that every object in A has every attribute in B, for every object in G that is not in A, there is an attribute in B that the object does not have and for every attribute in M that is not in B, there is an object in A that does not have that attribute.

If for formal concepts (A_1, B_1) and (A_2, B_2), $A_1 \sqsubseteq A_2$ and $B_2 \sqsubseteq B_1$ then $(A_1, B_1) \leq (A_2, B_2)$ and formal concept (A_1, B_1) is less general than (A_2, B_2).This order is represented by *concept lattice.* A lattice consists of a partially ordered set in which every two elements have a unique *supremum* (also called a least upper bound or *join*) and a unique *infimum* (also called a greatest lower bound or meet).

In the concept lattice, formal concepts are hierarchically grouped together according to their common attributes. Such grouping of objects is really biclustering of them i.e. clustering on the two sets simultaneously, the set of objects and the set of attributes.

Example 1. Consider an example of a formal context and the concept lattice built on it. In the Fig. 1 a) a fragment of the data contained in the myocardial infarction database is shown in the form of a two-dimensional binary context. The context has 7 objects and 4 binary attributes. The objects are patient IDs, and the meaning of the attributes is as follows:
•*gb* is the presence of an essential hypertension of a patient;
•*ant_im* is the presence of an anterior myocardial infarction (left ventricular);
•*im_pg_p* is the presence of a right ventricular myocardial infarction;
•*let_is* is the lethal outcome of a patient.



	gb	ant_im	im_pg_p	let_is
332	X	X	X	
321	X	X		
1500		X		X
1126		X	X	
1386	X	X		
1503		X		X
1655	X	X		X

a)

b)

Fig. 1: Example of a formal context and the concept lattice.

The concept lattice built on the considered formal context is shown on the Fig. 1 b). It has seven formal concepts including top and bottom ones which are unit and zero elements of an abstract lattice correspondingly. Information about patients is contained in five formal concepts, shown as filled circles. It is used so called reduced labeling [21] in order to succinctly represent information about objects and attributes of formal context. If label of attribute A is attached to some concept, that means, that this attribute occurs in objects of all concepts, reachable by descending paths from this concept to zero concept (bottom element) of lattice. If label of object O is attached to some concept, this means, that object O lays in all concepts, reachable by ascending paths in lattice graph from this concept to unit concept (top element) of lattice. If drawing of node contains blue filled upper semicircle, that means, that there is an attribute, attached to this concept. If drawing of node contains black filled lower semicircle, that means, that there is an object, attached to this concept.

In this example, all the patients have anterior myocardial infarction. According with reduced labeling, we can see in the lattice that *ant_im* attribute is presented in all five filled concepts. Among patients, those ones with IDs 1500,1503 and 1655 have the lethal outcome. On the left in the lattice there is a separate path with the nodes as the following formal concepts:

$(\{1500, 1503, 1655\}, \{ant\_im, let\_is\}), (1655, \{ant\_im, let\_is, gb\})$.
The first concept is more general than the second, since it is located above it. The first concept reflects two facts: the presence of a myocardial infarction and a lethal outcome for all of three patients. The second concept reflects the peculiarity of patient 1655: he is the only one of the three who has essential hypertension.

Thus, the formal context and concept lattice are a visual tool for representing knowledge.

2.1 Multidimensional Formal Contexts and Multimodal Clustering

Multidimensional or polyadic formal contexts arise as generalization of usual formal contexts. A multidimensional, n-ary formal context is defined by a relation $R \subseteq D_1 \times D_2 \times \ldots \times D_n$ on data domains $D_1, D_2, \ldots, D_n$. The context is an n+1 set:

$$\mathbb{K} =< K_1, K_2, \ldots, K_n, R >, \tag{2}$$

where $K_i \subseteq D_i$. Every n-ary context begets k-ary contexts, whose number is given by the Stirling formula [12] $S(n,k) = \frac{1}{k!}\sum\limits_{i=0}^{k}(-1)^i\binom{k}{i}(k-i)^n$. As it is shown in [11], multidimensional n-ary context also contains formal concepts which also form a lattice. Clustering on multidimensional formal contexts is called multimodal clustering [10].

According to multimodal clustering, for any dimension of formal context, the purpose of its processing is to find n-sets $H =< X_1, X_2, \ldots, X_n >$ which have the closure property [12]:

$$\forall u = (x_1, x_2, \ldots, x_n) \in X_1, X_2, \ldots, X_n, u \in R, \tag{3}$$

$\forall j = 1, 2, \ldots, n, \forall x_j \in D_j \backslash X_j < X_1, \ldots, X_j \cup \{x_j\}, \ldots, X_n >$ does not satisfy (3). The sets $H =< X_1, X_2, \ldots, X_n >$ constitute *multimodal clusters.*

In the Formal Concept Analysis, biclustering algorithms have been developed in sufficient detail [16]. The same cannot be said about the triclustering algorithms and especially about the multimodal clustering algorithms.

The simplest multidimensional formal context is a triadic context (tricontext) of the form $\mathbb{T} = (G, M, B, I)$, where B is a set specifying the conditions for the belonging of attributes to objects, $I \subseteq G \times M \times B$ is a ternary relation. Accordingly, the ternary concepts (triconcepts) are defined as triplets of the form:

$$(C_1, C_2, C_3), C_1 \subseteq G, C_2 \subseteq M, C_3 \subseteq B, \tag{4}$$

with corresponding closure conditions for prime operators. Prime operators have several other implementations here:

$$\begin{aligned} m' &= \{(g,b)|(g,m,b) \in I\} \\ g' &= \{(m,b)|(g,m,b) \in I\} \\ b' &= \{(g,m)|(g,m,b) \in I\} \end{aligned} \tag{5}$$

as well as the corresponding double prime operators:

$$\begin{aligned} m'' &= \{\widetilde{m}\ |(g,b) \in m' \bigcap (g, \widetilde{m}, b) \in I\} \\ g'' &= \{\widetilde{g}\ |(m,b) \in g' \bigcap (\widetilde{g}, m, b) \in I\} \\ b'' &= \{\widetilde{b}\ |(g,m) \in b' \bigcap (g, m, \widetilde{b}) \in I\} \end{aligned} \tag{6}$$

If formal context $\mathbb{T}$ is represented by a three dimensional tensor, then a triconcept is a 3-dimensional rectangle full of crosses.

Although there are several recognized algorithms for constructing three-dimensional formal concepts, for example, the Data-Peeler algorithm [12], the problem of constructing three-dimensional clusters of a given density, which are insufficiently dense concepts, is of practical interest.

If $\mathbb{C} = (X, Y, Z), X \subseteq K_1, Y \subseteq K_2, Z \subseteq K_3$ is a cluster then its density is defined as

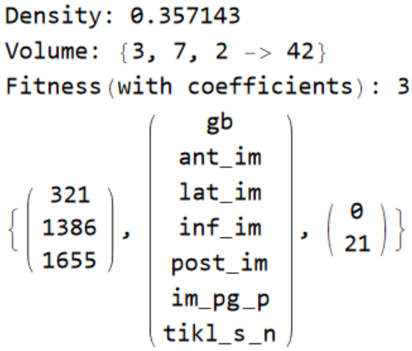
$$d(\mathbb{C}) = \frac{|I \cap (X \times Y \times Z)|}{\mathrm{v}(\mathbb{C})} \tag{7}$$

where cluster volume is

$$\mathrm{v}(\mathbb{C}) = |X| \times |Y| \times |Z| \tag{8}$$

Example 2. A three-dimensional formal context can be built on the data from the infarct database presented in the Example 1. The use of drugs on certain days of hospitality may be a third dimension in the context. The standard maximum treatment time for myocardial infarction is 21 days (at least in Russia), which defines the scale of the third dimension. Fig. 2 shows a regular three-dimensional cluster built on the context from the Example 1 with additional attributes. It is built with our evolutionary modeling framework [20] but may be discovered by other tools. The second subset of the cluster contains attributes *lat_im*, *inf_im*, *post_im* which detail variants of myocardial infarction. Attribute *tikl_s_n* means the use of the drug of *Ticlid* during therapy. It has values on the third dimension. On the Fig. 2 it is seen that *Ticlid* was used immediately (day 0) and on 21 st day of therapy.

The cluster on the Fig. 2 is not dense. This means that not all combinations of elements from the three subsets of the cluster take place. The “informativeness” of multidimensional clusters depends on their density, but not absolutely.



```
Density: 0.357143
Volume: {3, 7, 2 -> 42}
Fitness(with coefficients): 357.143
              gb
              ant_im
   321        lat_im
{ (1386) ,  ( inf_im ) , ( 0  ) }
   1655       post_im      21
              im_pg_p
              tikl_s_n
```

Fig. 2: Three-dimensional cluster.

In this example, we are not sure that the Ticlid was applied to these three patients only on these days, but the fact contained in this cluster may be of interest to cardiologists. “Absolutely reliable” facts have been contained in absolutely dense clusters, which are formal concepts. However, special facts that fall out of the general regularities can also be found in loose clusters. As for this example, this subset of patients may not form absolutely dense clusters and the facts should be searched for in low-density clusters by their additional analysis.

Evolutionary methods make it possible to simulate the clustering process in such a way that they allow to investigate the density distributions across clusters and thereby have a detailed picture of clustering.

3 Evolutionary Approach to Multimodal Clustering

Evolutionary Approach to Multimodal Clustering is based on Evolutionary Computation [15]. Evolutionary computation is a term referring to several methods of global optimization, united by the fact that they all use the concept of the evolution of a set of solutions to an optimization problem, leading to solutions corresponding to the extreme value of some function that sets the optimization quality criterion. Evolutionary computation is effective when working with multimodal functions. If such a function has a global extremum, the evolutionary algorithm finds solutions corresponding to the range of values of the quality function that are sufficiently close to the that global extremum.

3.1 Principle of Evolutionary Computation

Let $\mathbf{X}$ is a set of solutions of a problem. Every solution $x \in \mathbf{X}$ can be characterized by a quality measure named as *fitness function* $f(x)$. This measure, in general, is a *mapping* $f : X \to Y$ where $Y \subseteq R$ is the subset of a set of real numbers. This is basis for existence many variations of fitness functions which for example characterize configuration of clusters.

Let solutions of a problem depend on a set of parameters P. Most problems which have been solved by using Evolutionary Computation can be formulated as the following optimization problem: it is required to find optimal values of parameters p^* which deliver maximum fitness value $y^* \in \mathbf{Y}$, so the following is true:

$$p^* = argmax_{p^* \in P} f(x) \tag{9}$$

Evolutionary approach to solving this problem consists in the following.

Building encoding scheme. *Encoding scheme* is the mapping $\varphi : P \to S$ where set S contains objects which *encode* parameters from P. Genetic algorithms [13] which are widely used in Evolutionary Computation often use binary encoding and every value of $p \in P$ is represented as binary string. Encoding scheme is not necessarily binary (as it is not binary in Nature): every string position contains a symbol (*gene*) from *encoding alphabet*, and there are variants of alphabets applied in encoding schemata [14]. But necessarily there exists an inverse mapping $\varphi^{-1} : S \to P$, so for every $s \in S$ there exists $p \in P$.

Evolutionary algorithm. For given encoding scheme the following algorithm solves the problem (7).

A. Randomly generate an initial set (population) S_0 of objects from S.

B. Start *evolution* of the populations by applying a set of operators A to population S_0 and further iteratively so that for every $\mathbf{S}_{k+1} = \mathbf{A}(\mathbf{S_k})$ exists at least one

$$f[\varphi^{-1}(S_{k+1})] \geq f[\varphi^{-1}(S_k)], \tag{10}$$

where $s_k \in \mathbf{S}_k$ and $s_{k+1} \in \mathbf{S}_{k+1}$.

C. Finish the evolution of the population in accordance with the stopping criterion. Most often, the criterion for stopping is the immutability of the fitness function values over several steps of evolution.

If the set of operators $\mathbf{A}$ consists of genetic operators of *selection, mutation and recombination* (crossover) then evolutionary algorithm is named as *genetic algorithm* [14].

Selection works so that condition (10) is supported by the following "biological" principle: good parents produce good offspring (that is not true in Nature). So the higher fitness chromosomes have more opportunity to be selected than the lower ones and good solution is always alive in the next generation.

Crossover is the genetic operator that mixes two chromosomes together to form a new offspring. It does mixing by replacing fragments of chromosome's code divided in certain one or several randomly selected points.

*Mutation*Mutation involves modification of the gene values by randomly selecting new value from the alphabet at random point in the strings of genes.

Being realized, the algorithm (A. – C.) provides fairly accurate solution of the problem (9).

Fairly accurate means that evolutionary algorithm stops in a neighbourhood of global extreme of fitness function *f*. The size of a neighbourhood around extreme depends on the fitness function and parameters of genetic operators. When evolutionary algorithm works too fast it may stop at local extreme. This feature is traditionally considered as the lack of the algorithm but it may be useful for clustering since local extreme of quality measure may be “semantically better” than global extreme because it may correspond to the cluster containing an interesting fact. In our experiments we have observed just that situations.

3.2 Evolutionary Computation in Multimodal Clustering

Here we outline our solution to applying evolutionary computation to clustering in multidimensional contexts.

There are two crucial parameters of clustering problem: a measure of similarity of clustering objects (proximity measure) and number of clusters – is it given or not before clustering. Evolutionary algorithms have advantage over traditional clustering methods when:

1. measure of similarity of clustering objects is not traditional (Euclidian norm) [22];
2. number of clusters is not given and
3. number of clusters is great.

Consider this in more detail.

1. Many clustering algorithms use proximity measures of objects based on calculating the distances between them. Evolutionary and genetic algorithms work on the black box principle. The black box inputs are the values of the parameters P (see previous Section) and at the outputs we get the corresponding solutions, for which we calculate the values of the fitness function. A dependence between input and output may be very complex and not being expressed analytically as traditional proximity measures. For our experiments, there is no analytical dependence between cluster configurations specified by chromosomes and, for example, cluster densities. Therefore, the fitness function used in this case does not use the traditional proximity measure.
2. In evolutionary and genetic algorithms for clustering [15], the number of clusters which will be obtained depends on chromosome encodings. After analyzing the existing variants of chromosome encodings, we settled on two of them. The first variant is our chained integer encoding scheme [20]. The second encoding scheme is a binary scheme organized according to the principle of ”one chromosome – one cluster”. In this scheme, the chromosome gene is the object number. In this encoding, the maximal number of clusters is equal to the number of chromosomes. The number of clusters is not set initially. As a result of the evolution of n chromosomes, really k different chromosomes from the population should remain.
3. Models in the form of formal contexts give rise to a large number of formal concepts and an even greater number of clusters. When applying n-dimensional contexts, the upper bound of the number of clusters is estimated

as $2^{|K_1|} \times \ldots \times 2^{|K_n|}$ [11]. In the study of gene expression with evolutionary algorithms the number of genes in experiments may be over dozens of thousands and the length of chromosomes which represent clusters may be giant. Nevertheless, the computational problem of processing very long chromosomes (usually binary) is solved now [9, 18]. We performed evolutionary clustering with genetic algorithm realizing evolutionary algorithm (A. – C.) and having various parameters

Chromosome encoding. After analyzing the existing variants of chromosome encoding [15], we settled on two of them. The first variant is our chained integerencoding scheme [20]. The second encoding scheme is a binary scheme organized according to the principle of "one chromosome – one cluster". It has one, two or three sections in chromosomes according with the variant of encoding (see Section 3.1) and dimension of a context. Chromosomes for three-dimensional contexts have sections "patients", "attrbutes" and "days". In the sections, a number of gene is the number of patient, number of attribute from the context and number of a day according with objects order in the corresponding subsets in formal tricontext. Different chromosomes form different clusters. Because of the evolution of many such chromosomes, really k different chromosomes from n members of the population should remain.

Fitness function. As in FCA, we control cluster density (7), its volume (8) and special kind of interestingness. There is the trade-off problem between the density and the volume of triclusters [10, 17]. Depending on the data, density and volume may be contradictory characteristics of clusters. The data that we use are sparse, and if we collect enough units in a cluster, it will be simultaneously voluminous. Therefore, we do not use the volume of clusters in the fitness function, but only use their density. Nevertheless we calculate cluster volumes during evolution.

For the binary encoding scheme, fitness function has the form:

$$f(d) = \frac{1}{N} \sum_{i=1}^{N} \alpha_i d(C_i), \qquad (11)$$

where α_i is user defined coefficient, which in general depends on cluster density, N is the number of chromosomes in population which is equal to the maximal number of clusters.

For the chained integer-encoding scheme fitness function is the following:

$$f(d) = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{K_j} \sum_{i=1}^{K_j} \alpha_i d(C_i), \qquad (12)$$

where K_j is the number of clusters in the j-th chromosome.

Interestingness of a clusters. In a genetic algorithm, the whole fitness of population hides the features of individual chromosomes. But if selection leaves chromosomes with maximum fitness, then there is a chance that they will lead evolution to good solutions. Patient ID values found in clusters, other attributes corresponding to them from the “treatment” and “treatment outcomes” domains are evaluated for the presence of information in them that can be treated as facts. The formal criteria for selecting such “interesting” clusters are the following.

A single cluster. The presence of a single cluster at the end of evolution means that the algorithm most likely stopped at the global extremum of a fitness function [14, 22]. That cluster may be interesting and it probably may be dense. But as it was mentioned above, the diversity of clustering results is important and the presence of a single cluster is a reason to change parameters of the algorithm to have several extremes of a fitness function.

Dense clusters. The densest clusters among the received are very important since the information they contain may be certainly treated as facts.

Clusters of the maximum volume are interesting if they are dense enough. A large number of objects from different domains in the cluster is a sign that some pattern occurs on the data.

Clusters with given values of density and volume are interesting when density or volume values are specified in relation to some other characteristics of the clustering task.

4 Experiments

Experimental studies of the proposed approach were carried out in order to test the performance of the genetic algorithm under various parameters and for checking the possibility of interpreting clustering results as facts.

4.1 Data Set

We use Myocardial Infarction Complications Data Set [19] for experiments. It contains information about 1700 patients having disease of myocardial infarction. This data set has 1700 objects and 124 attributes collected in the multivalued formal context. Among attributes, there are ones about patients (ID only), their anamnesis, their treatment methods, and complications after treatment. An attribute may be binary or has a value as natural number.

4.2 Clustering Task

Based on the data of myocardial infarction, the task of phenotyping this disease was solved. Phenotyping refers to the determination of the form of the disease based on the clinical profile. A clinical profile is a cluster that can include various data describing both the disease itself and the methods of its treatment, as well as the conditions of patients. Therefore, we were interested in various triples of attribute sets from the domains ”patient”, ”treatment”, ”treatment results”.

4.3 Investigated Properties of the Algorithm

In the experiments, we investigated, first of all, the effectiveness of various options for implementing the evolutionary clustering algorithm, as well as its performance.

Diversity in clusters. To ensure diversity in clusters, we used large values of mutation probability. The graph in Fig. 3 shows that when a certain threshold value of the mutation probability is reached, the number of clusters increases sharply and remains unchanged with an increase in the mutation probability. In this case, the algorithm found 30 local extrema of the fitness function.

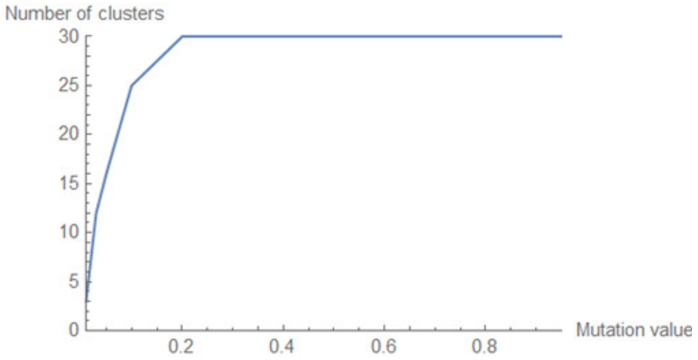


Fig. 3: Effect of the mutation probability value on the number of clusters.

Scaling algorithm performance. The algorithm processes very long three-section chromosomes of about 2000 genes fairly quickly. To investigate algorithm performance we have constructed seven formal contexts acquired from the whole set which number of objects and attributes are shown in Table 1 where ECG is electrocardiogram.

Table 1: Number of objects and attributes of formal contexts.

Context	**Objects**	**Attributes**
Anamnesis	1700	33
Therapy	1700	24
Analyzes	1700	19
Infarct	1700	6
ECG	1700	27
Therapy results	1700	14
Full data	1700	123

Fig. 4 shows clustering execution time for each of the seven contexts. The graph on this figure shows that the cluster construction time depends quasilinearly on the amount of data.

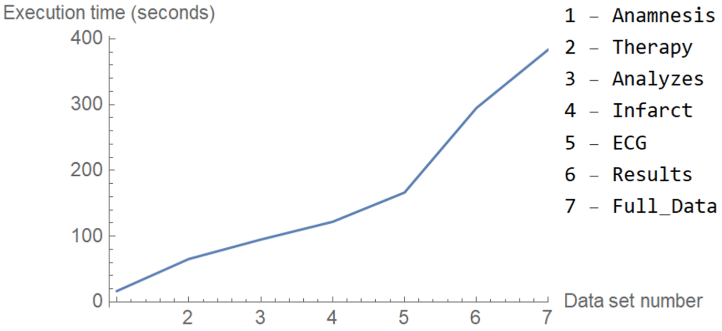
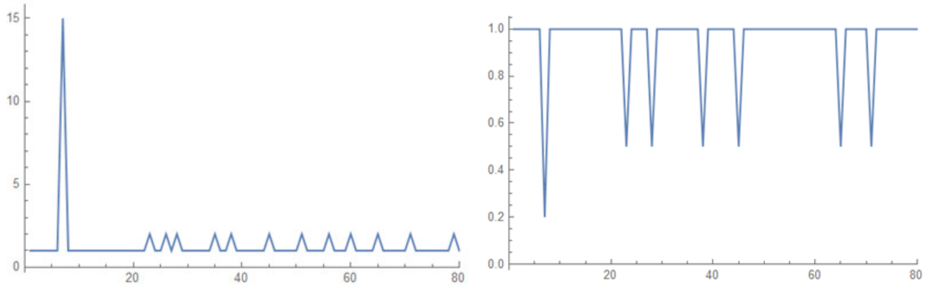


Fig. 4: Clustering execution time for several formal contexts.

Conflicting criteria. As we expected the criteria for maximizing the volume and density of clusters contradict each other. In the graphs on Figure 5 it is shown that the volume (graph a) and density (graph b) of clusters synchronously change during evolution.



a)

b)

Fig. 5: Effect of the mutation probability value on the number of clusters.

The combination of cluster density and volume in a single fitness function masks certain relationships between attributes in clustering results. Therefore, here we make a principal conclusion that it is necessary to apply two optimization criteria in this clustering task with genetic algorithm.

Comparison with Data-Peeler. We were also interested in absolutely dense clusters, the formal concepts. As expected, there were few such clusters, which follows from the sparsity of myocardial data. To compare our results with well known another algorithm, we selected Data-Peeler [12] and modernized its code [23] by adding graphical user interface. Comparison of the results is shown in Table 2.

Table 2: Clustering results compared with Data-Peeler.

Formal context	Number of clusters	Number of dense clusters	Number of Data-Peeler concepts
Anamnesis	30	14	449639
Therapy	30	19	28599
Analyzes	30	17	162
Infarct	30	30	65
ECG	30	10	689011
Therapy results	30	12	7798
Full data	30	4	12564890

The results in the last row of Table 2 can be explained by the high sparsity of data in this formal context. Accordingly, the Data-Peeler algorithm has built a lot of small concepts.

4.4 Facts Extracted with Clustering

We were interested in special clusters. First of all, these are clusters with large groups of patients characterized by certain combinations of attributes from the domains ”patient”, ”treatment”, ”treatment results”. Several such groups were obtained.

1. We have found that the lethal outcome of myocardial infarction is inherent in elderly patients over 60 years of age. This fact is consistent with the known data of cardiology.
2. In more detail, cases of heart attack in the anamnesis correlate with a fatal outcome, which also looks natural.

For both this groups of patients, we found absolutely dense clusters built on tensors with age and anamnesis attributes.

Unexpected result. We have found one unexpected result, which is as follows. On the data of myocardial infarction, there are stable (not changing according with different parameters of the genetic algorithm) and rather dense clusters in which a subgroup of patients with a lethal outcome have not got certain drugs. At the same time, patients with a non-lethal outcome had these drugs.

5 Conclusion

This paper proposes an approach to multimodal clustering on multidimensional formal contexts using evolutionary computation. This approach has been shown to be effective in experiments on clustering three-dimensional formal contexts based on data of patients with myocardial infarction.

The presented experimental results reflect the initial stage of research in this area. In the future, it is planned to do the following.

1. Evaluate the informativeness of the obtained clusters not manually, but using a user interface focused on doctors.
2. Experiments have confirmed that the criteria of cluster density and volume contradict each other. Therefore, it is necessary to apply multi-objective evolutionary clustering with appropriate algorithms.
3. Transition to the dimension of formal contexts greater than three. Separate groups of parameters can be represented as dimensions. Then their combinations obtained in clusters will reflect in more detail the relationships in heterogeneous data.

Acknowledgments. The reported study was funded by Russian Foundation of Basic Research, the research projects № 19-07-01178, № 20-07-00055 and RFBR and Tula Region according to research project № 19-47-710007.

References

1. Data Clustering: Algorithms and Applications. Ed. Charu Aggarwal, Chandan Reddy. CRC Press, London, (2013).
2. S. Theodoridis and K. Koutroumbas: Pattern Recognition. 3rd edn. Academic Press, (2006).
3. Sergey D. Dvoenko, Jan W. Owsinski: The Permutable k-means for the Bi-partial Crite-rion. Informatica 43(4), 253–262 (2019).
4. Hartigan J A. Direct clustering of a data matrix. Journal of the American statistical association, 67(337): 123—129 (1972)
5. Madeira SC, Oliveira AL. Biclustering algorithms for biological data analysis: a survey. IEEE/ACM Trans. Comput. Biol. Bioinform. Jan-Mar;1(1):24-45. (2004) DOI: 10.1109/TCBB.2004.2.
6. Ganter, Bernhard; Stumme, Gerd; Wille, Rudolf, eds., Formal Concept Analysis: Foundations and Applications, Lecture Notes in Artificial Intelligence, No. 3626, Springer-Verlag. Berlin (2005) DOI:10.1007/978-3-540-31881-1
7. Kaytoue, Mehdi, Kuznetsov, Sergei, Napoli, Amedeo. Biclustering Numerical Data in Formal Concept Analysis. 135-150. (2011). DOI: 10.1007/978-3-642-20514-9_12.
8. Ignatov D. I., Kuznetsov S. O., Zhukov L. E., Poelmans J., Can triconcepts become triclusters? // International Journal of General Systems, Vol. 42. No. 6 (2013)
9. Henriques R., Madeira S. Triclustering Algorithms for Three-Dimensional Data Analy-sis: A Comprehensive Survey. ACM Comput. Surv. V. 51. № 5. P. 1–43. (2019) DOI: 10.1145/3195833.
10. Dmitry V. Gnatyshak, Dmitry I. Ignatov, Sergei O. Kuznetsov: From Triadic FCA to Triclustering: Experimental Comparison of Some Triclustering Algorithms. In: Proceedings of the Tenth International Conference on Concept Lattices and Their Applications (CLA'2013), La Rochelle: Laboratory L3i, University of La Rochelle, pp. 249-260, (2013)
11. Voutsadakis, G. Polyadic concept analysis. – Order. Vol. 19 (3). Pp. 295–304 (2002)

12. Cerf, L., Besson, J., Robardet, C., Boulicaut, J.F.: Closed Patterns Meet N-ary Relations. In: ACM Trans. Knowl. Discov. Data. 3, 1, Article 3, 36 p. (2009)
13. R. M. Cole: Clustering with Genetic Algorithms, MSc Thesis, University of Western Australia, Australia (1998)
14. Goldberg D.E. Genetic Algorithms in Search Optimization and Machine Learning. Addison-Wesley, Reading, MA, USA (1989)
15. Hruschka E., Campello R., Freitas A., de Carballo A. A Survey of Evolutionary Algorithms for Clustering. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on Evolutionary Computation. V. 39. P. 133–155. (2009) DOI: 10.1109/TSMCC.2008.2007252.
16. S.O. Kuznetsov and S.A. Obiedkov: Comparing Performance of Algorithms for Generating Concept Lattices. Journal of Experimental and Theoretical Artificial Intelligence, Vol. 14, no. 2-3, pp. 189-216, 2002.
17. Ignatov D. I., Gnatyshak D. V., Sergei O. Kuznetsov, Boris G. Mirkin: Triadic Formal Concept Analysis and triclustering: searching for optimal patterns. In: Machine Learning, April, 2015, pp. 1-32.
18. Ma P., Chan K., Yao X., Chiu D.: An evolutionary clustering algorithm for gene expression microarray data analysis. IEEE Transactions on Evolutionary Computation. V. 10. P. 296– 314 (2006) doi: 10.1109/TEVC.2005.859371.
19. Myocardial infarction complications Data Set. http://archive.ics.uci.edu/ml/machine-learning-databases/00579/
20. M. Y. Bogatyrev, A. P. Terekhov: Framework for Evolutionary Modelling in Text Mining. - Proceedings of the SENSE'09 - Conceptual Structures for Extracting Natural Language Semantics. Workshop at 17th International Conference on Conceptual Structures (ICCS'09), p.p. 26-37 (2009)
21. Serhiy A. Yevtushenko: System of data analysis "Concept Explorer". (In Russian). In: Proceedings of the 7th National Conference on Artificial Intelligence KII-2000, p. 127-134, Moscow (2000)
22. Dan Simon: Evolutionary Optimization Algorithms: Biologically-Inspired and Population-Based Approaches to Computer Intelligence. John Wiley & Sons, New Jersey (2013)
23. https://github.com/ibrahim85/d-peeler