Abstract. The problem of clustering data arrays described in both vector and matrix forms and based on the optimization of data distribution density functions in these arrays is considered. For the optimization of these functions, the algorithm that is a hybrid of Fish School Search, random search, and evolutionary optimization is proposed. This algorithm does not require calculating the optimized function’s derivatives and, in the general case, is designed to find optima of multiextremal functions of the matrix argument (images). The proposed approach reduces the number of runs of the optimization procedure, finds extrema of complex functions with many extrema, and is simple in numerical implementation.

Keywords: combined optimization, fuzzy clustering, evolutionary algorithms, density functions, Fish School.

INTRODUCTION

The problem of clustering arrays of arbitrary nature observations is an integral part of Data Mining, and more generally Data Science. To solve this problem, it was proposed a lot of approaches that differ as a priori assumptions about the physical nature of data and problems solved by their basis, and the mathematical apparatus that was used [1–4]. From a computational point of view, the simplest are the so-called hierarchical methods and algorithms based on partitions [3], among of that we should mention the \( k \)-means procedure, that has become widespread to solve a variety of problems. It should be noted here that the most adequate mathematical apparatus for solving clustering problems are methods of computational intelligence [5–7] and, above all, artificial neural networks, fuzzy systems, evolutionary optimization and so-called hybrid systems of computational intelligence that connect these three areas. It is interesting to note that one of the most popular neural networks — self-organizing Kohonen maps [8] actually implements the \( k \)-means procedure, presented in recurrent form.

It should be noted that in the general case the solution of the clustering problem is significantly complicated if the original vectors (in the general case matrices) observations have a large variety are, distorted by perturbations and noises, contain outliers and omissions, the original arrays themselves or too large (Big Data) or too short, clusters can have a rather complex shape, and their number is a priori unknown.
In this case, the most effective (but also the most complex) are algorithms based on the analysis of data distribution densities, among which as one of the most “popular” are DENCLUE [9] and its modifications [10–12], which were proposed to solve clustering problems of large arrays of high-dimensional vector data, and the classes formed in the clustering process can have any complex shape. At the heart of these algorithms is the search for extremes — maxima in the data density functions in the analyzed array (multi-extremal optimization), and this function is formed as a superposition of kernel (bell-shaped) functions associated with each observation. In fact, this function is based on Parzen windows [13] and Nadaraya–Watson estimates [14, 15].

From a computational point of view, the clustering problem becomes of finding local extrema of the multiextrema function of the density vector argument using gradient procedures that are repeatedly run from different points in the original data set. It is clear that this takes a long time, because a priori it is not even known how many extremes the formed density function.

The process of finding these extremes can be accelerated by using the ideas of evolutionary optimization, that includes algorithms inspired by nature, swarms algorithms, population algorithms, etc. [16–18]. In this case, the search is conducted simultaneously by a group of agents acting either independently or in interaction, which can significantly speed up the process of finding extremes, each of which “corresponds” to one or another cluster that is being formed.

**FORMATION OF THE DATA DISTRIBUTION DENSITY FUNCTION IN THE CLUSTERING ARRAY**

The initial information for solving the clustering problem is traditionally an array of observation vectors \( X = \{x(1), x(2), \ldots, x(k), \ldots, x(N)\} \), \( x(k) = \{x_i(k)\} \in R^n \), the data are pre-centered on a hypercube so that \( x(k) = \{x_{i,j}(k)\} \in R^{n_1 \times n_2} \). This situation can occur in the case of image array processing. The basic concepts on which DENCLUE is based are the influence function, the density function and the density attractors, which are essentially the local extremes of the density function. In the general case, the influence function for any vector observation \( x(\bullet) \) from the original array \( X \) is a kernel bell shaped function \( f^{x(\bullet)}(x) \), in this case the most popular is the traditional Gaussian one

\[
f^{x(\bullet)}(x) = \exp \left( - \frac{d^2(x,x(\bullet))}{2\sigma^2} \right) = \exp \left( - \frac{\|x - x(\bullet)\|^2}{2\sigma^2} \right), \quad (1)
\]

where \( d^2(x,x(\bullet)) \) — euclidean distance; \( \sigma^2 \) — parameter of the influence function width, due to the simplicity of calculating its derivatives.

In the matrix case, instead of the Euclidean one, we can use the Frobenius metric, and the influence function takes the form

\[
f^{x(\bullet)}(x) = \exp \left( - \frac{d^2(x,x(\bullet))}{2\sigma^2} \right) = \exp \left( - \frac{Tr(x - x(\bullet))(x - x(\bullet))^T}{2\sigma^2} \right), \quad (2)
\]

where \( Tr(\bullet) \) — matrix trace symbol.

It is easy to see that (2) is a generalization of (1).

Based on the influence functions, formed the data density distribution function in the array \( X \) in the form
Clusterization of vector and matrix data arrays using the combined evolutionary method ...

\[ f^X(x) = \sum_{k=1}^{N} f(x, x(k)), \]  

which is essentially an estimate of Nadaraya–Watson. It’s easy to see what the function \( f^X(x) \) can take values in an interval \( 1 \leq f^X(x) \leq N \). In this case the extrema values from this interval are accepted when the sample contains only one observation or all \( N \) observation observations coincide, i.e. there is only one cluster — a degenerate situation.

To find \( m > 1 \) clusters it’s necessary to introduce some threshold \( \xi > 1 \), that allows to build really significant clusters by excluding anomalous observations and classes that contain too small data.

Actually, the process of cluster formation is associated with finding all extremes of the density function (3) using a gradient procedure

\[ x^l = x^{l-1} + \eta^l \nabla f^X(x^l, x^{l-1}), \quad x_0 = x(k), \quad l = 0,1,2,\ldots; \quad \forall k = 1,2,\ldots,N, \]  
i.e. the number of runs of algorithm (4) is determined by the size of the training sample \( N \). It is clear that with large \( N \) the process of clustering — finding local extrema can take a lot of time. Therefore, the proposed modifications of DENCLUE are associated with speeding up the process of finding local extrema (3) by modifying the gradient procedure (4) \([10–12]\).

In the case, when observations \( x(k) \) in dataset \( X \in \mathbb{R}^{n_1 \times n_2} \) are matrices, it is easy to consider the matrix version of the procedure (4):

\[ x^l = x^{l-1} + \eta^l \Gamma^X(X, x^{l-1})(\text{Tr}\Gamma^X(X, x^{l-1})\Gamma^{-T}(X, x^{l-1}))^{-1/2}, \]

where \( \Gamma^X(X, x^{l-1}) = \left[ \frac{\partial f^X(X, x^{l-1})}{\partial x_{ij}} \right] \in \mathbb{R}^{n_1 \times n_2}. \)

The gradient optimization process ends with a search \( m \) local extrema of function (3), with less value \( \xi \), than more clusters can be formed.

It is possible to speed up the process of finding local extrema by using evolutionary optimization methods instead of gradient search, among which the so-called Fish schools search can be noted as quite efficient, numerically simple and fast \([19–21]\), which should be modified to solve the clustering problem.

**MODIFIED OPTIMIZATION METHOD BASED ON FISH SCHOOL**

When using the methods of evolutionary optimization, which are essentially zero-order optimization methods, i.e. do not use derivatives, it is assumed that when finding the extrema of some function \( f^X(x) \) the population of agents are used, each of them acts either independently or in interaction with others, with the movement of each \( q \)th agent \( (q = 1,2,\ldots,Q) \) on \( l \)th search iteration can be written as:

\[ x_q^l = x_q^{l-1} + \eta_q^l \text{Dir}_q^l, \quad q = 1,2,\ldots,Q, \]

where \( x_q^l = (x_{q1}^l, x_{q2}^l,\ldots,x_{qn}^l)^T; \quad \text{Dir}_q^l \) — vector that specifies the direction of movement \( q \)th agent on \( l \)th search iteration.
In a large family of such methods should be noted the method based on of fish schools, where each agent of the population simulates the movement of an individual fish in the school [19–21].

The main advantage of this method is the sufficient efficiency of finding the global extrema of rather complex functions, which include the density function of data distribution in clustering problems.

The authors of the method introduce iterations related to the movement of the school: feeding and swimming.

The feeding operator is responsible for the weight of each fish as an element of school — the agent. The heavier the fish, the closer it is to the extreme — the maximum. The weight of each fish $w_q$ is tuned according to the expression

$$w_q^j = w_q^{j-1} + \frac{f^x(x_q^j) - f^x(x_q^{j-1})}{\max \{f^x(x_q^j) - f^x(x_q^{j-1})\}} \quad \forall q = 1,2,...,Q,$$

where

$$0 < w_q^j < w_{\text{max}}, \quad w_q^0 = 0.5w_{\text{max}}.$$

The swimming operator describes both the individual movement of each fish and the collective movement of the school as a whole. Three types of movement are considered here: individual, instinctively — collective and collective volitional.

Individual movement is described by the relation

$$x_{qi}^j = \begin{cases} x_{qi}^j + \eta_q^j \text{Rand} \{0,1\}, & \text{if } f^x(x_q^j) > f^x(x_q^{j-1}); \\ x_q^{j-1} & \text{else}, \end{cases}$$

where $\text{Rand} \{0,1\}$ — evenly distributed in the interval $(0,1)$ random number. It should be noted that (6) is essentially a local random search with return, introduced by L. Rastrigin [22]. In fact, this is the procedure of “probing” the function $f^x(x)$ around the point $x_q^{j-1}$, in this case, in addition to (5), any other random search algorithm can be used here.

On the basis of probing the density function with the help of individual movement (5) the instinctive-collective movement in the direction of growth of this function is realized as:

$$x_q^j = x_q^{j-1} + \frac{\left(\sum_{p=1}^{Q} (x_p^j - x_p^{j-1})\right)(f^x(x_q^j) - f^x(x_q^{j-1}))}{\sum_{p=1}^{Q} (f^x(x_p^j) - f^x(x_p^{j-1}))}.$$

At this stage, there is a balanced averaging of individual movements, taking into account the “success” of each of the fish-agents.

And finally, collectively-volitional movement, when all the fishes of the school “pull” to the weighted center of gravity, if the cant goes to the extreme, and “run away” if the population moves in the wrong direction.

Considering the weighted center of gravity of the fish school
Clusterization of vector and matrix data arrays using the combined evolutionary method …

\[ Bar^l = \frac{\sum_{p=1}^{Q} x^l_p w^l_p}{\sum_{p=1}^{Q} w^l_p}, \]  

we can record this movement as

\[
x_q^l = \begin{cases} 
  x_q^l - n_q^l \text{Rand} \{0,1\} \frac{x_q^{l-1} - Bar^{l-1}}{\|x_q^{l-1} - Bar^{l-1}\|}, & \text{if } \sum_{p=1}^{Q} w_p^l > \sum_{p=1}^{Q} w^l_{p-1}; \\
  x_q^l + n_q^l \text{Rand} \{0,1\} \frac{x_q^{l-1} - Bar^{l-1}}{\|x_q^{l-1} - Bar^{l-1}\|}, & \text{if } \sum_{p=1}^{Q} w_p^l < \sum_{p=1}^{Q} w^l_{p-1}.
\end{cases} \]

To increase the efficiency of FSS, an additional breeding operator may be introduced, which allows the creation of new fish-agents that have improved characteristics compared to existing members of the school. To do this, we can use the ideas of evolutionary operations [23], among which from a computational point of view and efficiency — the credibility of finding the extrema can be noted sequential simplex method [24] and its modifications [25].

Let’s form the school that containing \( Q = n + 1 \) fish-agents, but this number remains unchanged in the search process, i.e. the population \( x_1^0, x_2^0, ..., x_Q^0 \) generated randomly. In this population we find the “worst” fish \( x_{q\text{worst}}^0 \), which has the lowest weight \( w_{q\text{min}}^0 \) and the “best” fish \( x_{q\text{best}}^0 \) with the greatest weight \( w_{q\text{max}}^0 \). The main operation of the simplex movement is mapping \( x_{q\text{worst}}^0 \) through the center of gravity \( n \) fishes (without the worst), which can be written in the form

\[ \bar{x}^0 = \frac{1}{n} \sum_{q=1}^{Q} (x_q^0 - x_{q\text{worst}}^0). \]

As a result of this operation, a new fish is created

\[ x_q^{*l} = \bar{x}^0 + \alpha (\bar{x}^0 - x_{q\text{worst}}^0), \]

which replaces the worst individual in the school \( x_{q\text{worst}}^0 \). Thus creates a new population \( x_1^1, x_2^1, ..., x_Q^1 \). Here \( 0.5 \leq \alpha \leq 2 \) — parameter that controls the shape of the school-simplex in the optimization process. In the case when \( \alpha = 1 \) the mapping of the simplex through the center of gravity is realized \( \bar{x}^0 \) in the case if \( f^x(x_q^{*l}) > f^x(x_{q\text{best}}^0) \) accepted \( \alpha = 2 \), that is, the school is “stretched” in a favorable direction, if \( f^x(x_q^{*l}) < f^x(x_{q\text{worst}}^0) \) \( \alpha = 0.5 \) is accepted, that is, the simplex faces a relatively unfortunate direction. Thus the motion of the school-simplex can be described by relations.
\[
\begin{aligned}
\lambda^l &= \frac{1}{n} \sum_{q=1}^{Q} (x_q^{l-1} - x_{\text{worst}}^{l-1}); \\
\lambda_q^l &= x_q^{l-1} + \alpha (\bar{x}^{l-1} - x_{\text{worst}}^{l-1}),
\end{aligned}
\]
then in the general case it is essentially an Nelder–Mead optimization algorithm [25]. Thus, in the process of finding the extreme, the worst fishes with the lowest weight are removed from the cant and new agents with higher weight are created.

Thus, the process of combined optimization of the density function (5)–(10) is essentially a combination of FSS, random search and evolutionary operating based on the Nelder–Mead method.

Since the problem under consideration is essentially a problem of multiextrema optimization, it is necessary to find a set of extremums, each of which is a centroid of a cluster. Therefore, the optimization problem must be solved repeatedly at different values \( \sigma^2 \) and \( \xi \). When finding any of the extremes from the original sample \( X \) observations located directly in its vicinity are excluded. After this removal, the proposed procedure of combined evolutionary optimization is repeated until all extrema centroids are found.

**EXPERIMENTAL RESEARCH**

The experimental research was conducted on two databases, such as Page blocks and Spam base and two test multieextrema functions. The description of datasets shown in Table 1 and test multieextrema functions in Table 2.

**Table 1.** Data set description

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Instances</th>
<th>Attributes</th>
<th>Clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page blocks</td>
<td>5472</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Spam base</td>
<td>4601</td>
<td>57</td>
<td>2</td>
</tr>
</tbody>
</table>

**Table 2.** Test multieextrema functions

<table>
<thead>
<tr>
<th>Name of function</th>
<th>Formulas</th>
<th>Domain</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rastrigin</td>
<td>( f(x) = 20 + x^2 + y^2 - 10 \cos(2\pi x) + \cos(2\pi y) )</td>
<td>([-5.12; 5.12])</td>
<td>0.01</td>
</tr>
<tr>
<td>Griewangk’s</td>
<td>( f(x) = \frac{1}{4000} x + \frac{1}{4000} y - \cos \left( \frac{x}{\sqrt{1}} \right) \cos \left( \frac{x}{\sqrt{2}} \right) + 1 )</td>
<td>([-30; 30])</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Due the fact, that Rastrigin’s and Griewangk’s functions has a lot if local extreme points in its search area, as shown on Fig. 1, \( a \) and Fig. 2, \( a \), we add 514 agents.

**Fig. 1.** Rastrigin’s function, that has a lot of extreme points (a); modified optimization method based on fish School on Rastrigin’s function (b)
Clusterization of vector and matrix data arrays using the combined evolutionary method …

In Page blocks dataset was presents classified blocks of the page layout in a document that has been detected by a segmentation process. Spam base dataset also extracted from the UCI Machine Learning Repository and describes e-mail classified as spam or not spam.

The accuracy comparison of the well known optimization algorithms such as Fish School (FSS) and Cat Swarm (CSO) and proposed Modified Optimization Method Based on Fish School (OMFS).

<table>
<thead>
<tr>
<th>Data</th>
<th>OMFS</th>
<th>FSS</th>
<th>CSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rastrigin</td>
<td>Mean</td>
<td>190.46</td>
<td>189.65</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>195.83</td>
<td>195.59</td>
</tr>
<tr>
<td>Griewangk’s</td>
<td>Mean</td>
<td>3.65</td>
<td>4.12</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>4.82</td>
<td>4.12</td>
</tr>
<tr>
<td>Page blocks</td>
<td>Mean</td>
<td>951.47</td>
<td>951.01</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>959.64</td>
<td>959.43</td>
</tr>
<tr>
<td>Spam base</td>
<td>Mean</td>
<td>291.77</td>
<td>291.17</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>299.84</td>
<td>299.48</td>
</tr>
</tbody>
</table>

From obtained result, that shown in Table 3 we see, that Modified Optimization Method Based on Fish School generally perform better than the original algorithm. The convergence process of hybrid algorithm demonstrate in Fig. 3.

Fig. 2. Griewangk’s function, that has a lot of extreme points (a); modified optimization method based on fish School on Griewangk’s function (b)

Fig. 3. Modified Optimization Method Based on Fish School on test function: Rastrigin’s function (a) and Griewangk’s function (b)

To evaluate the performance of clustering method used the several validity metrics: Dunn Index (DI) — high value indicates a better clustering; Davies-
Bouldin Index (DBI) — the smallest value indicates the better clustering; Cluster Accuracy (CA) — the high value indicates the best clustering quality.

For comparison proposed method classification of vector and matrix data sets based on combined optimization of distribution functions (CODF) against classical DENCLUE algorithm and DENCLUE-IM for big data clustering (Table 4).

**Table 4.** The comparison algorithms according to their validity metric

<table>
<thead>
<tr>
<th>Data</th>
<th>Measures</th>
<th>Spam base</th>
<th>Page blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODF</td>
<td>DI</td>
<td>0.835</td>
<td>0.721</td>
</tr>
<tr>
<td>DENCLUE</td>
<td></td>
<td>0.789</td>
<td>0.721</td>
</tr>
<tr>
<td>DENCLUE-IM</td>
<td></td>
<td>0.831</td>
<td>0.693</td>
</tr>
<tr>
<td>CODF</td>
<td>DBI</td>
<td>0.768</td>
<td>0.764</td>
</tr>
<tr>
<td>DENCLUE</td>
<td></td>
<td>0.867</td>
<td>0.864</td>
</tr>
<tr>
<td>DENCLUE-IM</td>
<td></td>
<td>1.041</td>
<td>1.041</td>
</tr>
<tr>
<td>CODF</td>
<td>CA</td>
<td>0.718</td>
<td>0.920</td>
</tr>
<tr>
<td>DENCLUE</td>
<td></td>
<td>0.805</td>
<td>0.920</td>
</tr>
<tr>
<td>DENCLUE-IM</td>
<td></td>
<td>0.701</td>
<td>0.911</td>
</tr>
</tbody>
</table>

All these results conclude that proposed method classification of vector and matrix data sets based on combined optimization of distribution functions has an acceptable clustering performance.

**CONCLUSION**

The problem of clustering data arrays that are described in both vector and matrix forms based on the optimization of data distribution density functions in these arrays is considered. For optimization of these functions — local extrema search we have proposed the hybrid of Fish School Search algorithm, random search and evolutionary optimization. This algorithm does not require the calculation of derivatives of the function, which is optimized and in the general case is designed to find the maxima of multitemrema functions of the matrix argument (images).

The proposed approach allows to reduce the number of the optimization procedure runs, allows to find the extremes of complex shape functions and is easy in numerical implementation.

**REFERENCES**

Clusterization of vector and matrix data arrays using the combined evolutionary method


INFORMATION ON THE ARTICLE

Yevgeniy V. Bodianskiy, ORCID: 0000-0001-5418-2143, Kharkiv National University of Radio Electronics, Ukraine, e-mail: yevgeniy.bodianskiy@nure.ua

Alina Yu. Shafronenko, ORCID: 0000-0002-8040-0279, Kharkiv National University of Radio Electronics, Ukraine, e-mail: alina.shafronenko@nure.ua

Iryna P. Pliss, ORCID: 0000-0001-7918-7362, Kharkiv National University of Radio Electronics, Ukraine, e-mail: iryna.pliss@nure.ua

КЛАСТЕРИЗАЦІЯ ВЕКТОРНИХ ТА МАТРИЧНИХ МАСИВІВ ДАНИХ ІЗ ВИКИРИСТАННЯМ КОМБІНОВАНОГО ЕВОЛЮЦІЙНОГО МЕТОДУ РИБНИХ ЗГРАЙ / Є.В. Бодянський, А.Ю. Шафроненко, І.П. Пліс

Анотація. Розглянуто задачу кластеризації масивів даних, що описано як у векторній, так і матричній формі на основі оптимізації функцій щільності розподілу даних у цих масивах. Для оптимізації цих функцій - пошуку локальних екстремумів запропоновано алгоритм, що є гібридом Fish School Search, випадкового пошуку та еволюційної оптимізації. Цей алгоритм не потребує обчислення похідних функції, що оптимізується, і у загальному випадку прийнятий для відшукування максимальних багатоекстремальних функцій матричного аргумента (зображення). Запропонований підхід дозволяє скоротити кількість запусків процедур оптимізації, знаходити екстремуми функцій складної форми та є простим у числової реалізації.

Ключові слова: комбінована оптимізація, нечетка кластеризація, еволюційні алгоритми, функція щільності, Fish School.