*ISSN 2500-316X (Online)*

*https://doi.org/10.32362/2500-316X-2019-7-6-134-150*



*УДК 519.248:54.06:678*

# Генетический алгоритм кластеризации

**М.А. Анфёров**

*МИРЭА – Российский технологический университет, Москва 119454, Россия*
*@Автор для переписки, e-mail: anfyorov@inbox.ru*

В рамках гибридного подхода построения информационных интеллектуальных технологий поддержки принятия решений предложен генетический алгоритм кластеризации объектов анализа в различных предметных областях. Алгоритм позволяет учитывать при кластеризации различные предпочтения аналитика, отражаемые в расчетной формуле фитнес-функции. Показано место данного алгоритма среди используемых для кластерного анализа. Алгоритм является простым в его программной реализации, что повышает его надежность в использовании. Используемая технология эволюционного моделирования несколько расширена в рассматриваемом алгоритме. Во-первых, используется десятичная система счисления для кодирования хромосом в отличие от традиционной двоичной. Это вызвано множественным, а не бинарным, состоянием генов хромосомы. С этим связано отсутствие в данном алгоритме генетического оператора инверсии. Во-вторых, введен новый генетический оператор фильтрации, который отсеивает хромосомы, не удовлетворяющие условию требуемого количества кластеров в поставленной задаче. Такие хромосомы могут появляться в стохастическом процессе их эволюции. Представленный алгоритм был исследован в серии вычислительных экспериментов. В результате было выявлено, что стабилизация разбиения на кластеры достигается при числе реализованных поколений эволюции от 200 и более при сравнительно небольшом размере популяции от 150 хромосом, не требующем значительного выделения оперативной памяти компьютера. Проведенные вычисления на реальных данных показали для данного алгоритма хорошее качество кластеризации и вполне приемлемую производительность одного порядка с производительностью алгоритмов SOM и "*k*-means".

***Ключевые слова:*** кластеризация, генетический алгоритм, интеллектуальная технология, принятие решений, вычислительный эксперимент.

***Для цитирования:*** Анфёров М.А. Генетический алгоритм кластеризации. *Российский технологический журнал.* 2019:7(6):134-150. https://doi.org/10.32362/2500-316X-2019-7-6-134-150

# Genetic clustering algorithm

**M.A. Anfyorov**

*MIREA – Russian Technological University, Moscow 119454, Russia*
*@Corresponding author, e-mail: anfyorov@inbox.ru*

The genetic algorithm of clustering of analysis objects in different data domains has been offered within the hybrid concept of intelligent information technologies development aimed to support decision-making. The algorithm makes it possible to account for different preferences of the analyst in clustering reflected in a calculation formula of fitness function. The place of this algorithm among those used for cluster analysis has been shown. The algorithm is simple in its program implementation, which increases its usage reliability. The used technology of evolutionary modeling is rather expanded in the mentioned algorithm. Firstly, the decimal chromosomes coding is used instead of the traditional binary coding. This has resulted from the fact that the chromosome genes condition is multiple and not binary. Moreover, this is due to the absence of the genetic operator of inversion in this algorithm. Secondly, a new genetic operator used for filtering has been implemented. This operator eliminates chromosomes that do not meet the required clusters quantity condition in a task. Such chromosomes can appear in the stochastic process of their evolution. The presented algorithm has been studied in a series of simulation experiments. As a result, it has been found that stabilization of splitting into clusters is reached when the number of completed generations of evolution is 200 and more, and the population size is rather small: from 150 chromosomes (in this case no considerable amount of random-access store is required). The calculations carried out on real data showed for this algorithm the high quality of clustering and the acceptable computing speed of the same order with the computing speed of SOM and "k-means" algorithms.

***Keywords:*** clustering, genetic algorithm, intellectual technology, decision-making, computing experiment.

***For citation:*** Anfyorov M.A. Genetic clustering algorithm *Rossiiskii tekhnologicheskii zhurnal* = Russian Technological Journal. 2019;7(6):134-150 (in Russ.). https://doi.org/10.32362/2500-316X-2019-7-6-134-150

## Введение

Процессы всех этапов жизненного цикла наукоемкой продукции (авиационная техника, электроника и др.) сопровождаются принятием проектных, управленческих, маркетинговых и других решений в условиях неопределенности информации, вызванной незнанием или недостаточностью данных, их неполнотой и расплывчатостью, неточностью (ошибкой) измерения или наблюдения, субъективной нечеткой оценкой. Для раскрытия этих неопределенностей в настоящее время наряду с интеллектом специалистов применяются информационные технологии поддержки искусственного интеллекта, реализующие «мягкие вычисления» с использованием нейронных сетей, генетических алгоритмов, нечетких множеств, кластеризации [1–3]. При этом наибольший эффект достигается при использовании гибридных технологий, использующих сочетание вышеназванных моделей и методов [4–6].

В информационной поддержке принимаемых решений успешно используется технология кластерного анализа в таких областях, как машиностроение [7], экономика (сегментирование рынка [8], стратегическое планирование [9], анализ кредитных рисков [10]), государственное и муниципальное управление (региональная экономическая поли-

тика [11], налоговое администрирование [12], избирательные кампании [13]), медицина [14, 15], образование [16, 17] и др. Еще более широкое распространение получила кластеризация в составе различных информационных технологий, например, распознавания изображений [18], анализа текстов [19], отражения кибератак [20].

В настоящее время использование традиционных алгоритмов кластеризации, например, итеративных ("*k*-means" [21], SOM [22] и т. д.), иерархических дивизимных (BIRCH [23], MST [24] и т. д.) и иерархических агломеративных (CURE [25], ROCK [26] и др.) сменилось гибридным подходом, предполагающим использование при кластеризации вышеназванных технологий «мягких вычислений».

Данное обстоятельство вызвано растущим количеством и сложностью задач, использующих кластеризацию, значительным увеличением объема анализируемых данных, зачастую в режиме онлайн [27]. При этом, рассматривая масштабируемые алгоритмы для работы с большим объемом данных [28, 29], следует отметить их работу с категорийными данными [30].

В настоящее время возможности кластерного анализа расширяются за счет использования наряду с традиционными новых методов (например, метод волновых преобразований в алгоритме WaveCluster [31]), объединения нескольких методов в один [32], выявления и учета при кластеризации взаимозависимости в данных [33, 34], а также за счет гибридного подхода [5, 35].

Предлагаемый алгоритм кластеризации можно также отнести к категории гибридных, так как он использует эволюционный подход, характерный для так называемых генетических алгоритмов. Основанием для его разработки послужила необходимость использования в системах поддержки принятия решений универсального алгоритма по отношению к предпочтениям аналитика в различных ситуациях, то есть формирование кластеров не по одному, а по совокупности нескольких критериев. В представленном алгоритме эта универсальность обеспечивается изменением расчета фитнес-функции (функции приспособленности) используемых в алгоритме хромосом без изменения структуры самого алгоритма. В рассматриваемом случае при кластеризации, кроме близости объектов к центру кластера в метрическом пространстве признаков (критерий формирования более плотных кластеров), учитывается расстояние между кластерами (критерий максимального различия кластеров между собой).

## Описание алгоритма кластеризации

Исходные данные представляют собой набор векторов в пространстве из *n* признаков $\mathbf{x}^p = (x_1^p, x_2^p, ..., x_n^p),\ p = \overline{1,k}$. Каждый *p*-й вектор относится к соответствующему группируемому объекту. В составе исходных данных присутствует также требуемое количество кластеров – *m*.

Генетический алгоритм традиционно работает с популяцией хромосом, модифицируемых в итерационном процессе эволюции. В нашем случае в популяции размером w каждая хромосома отображает один из возможных вариантов кластеризации, в качестве которой используется вектор $\mathbf{g}_l^t = (g_{l1}^t, g_{l2}^t, ..., g_{lk}^t),\ t = \overline{1,w}$ ($t$ – номер хромосомы внутри популяции). Количество компонентов вектора хромосомы является фиксированным и равным количеству кластеризуемых объектов, то есть *k*. В отличие от традиционного подхо-

да, при кодировании значений компонентов хромосомы (значений «генов») используется десятичная система счисления. Целое значение компонента $g_{lr}^{t}$ (*r*-го «гена» в хромосоме) кодирует номер кластера, к которому следует отнести *r*-й объект, и находится в интервале [1, *m*].

Критерий качества кластеризации определяется с учетом среднего размера $\overline{R}_t$ формируемых кластеров

$$\overline{R}_t = \frac{1}{m}\sum_{j=1}^{m}\frac{1}{k_j}\sum_{p=1}^{k_j} d^2\left(\mathbf{x}_j^p;\mathbf{x}_{0_j}\right) \tag{1}$$

и оценки удаленности кластеров друг от друга

$$\overline{S}_t = \frac{1}{C_m^2}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} d^2\left(\mathbf{x}_{0_i};\mathbf{x}_{0_j}\right), \tag{2}$$

где $k_j$ – число входных образов (объектов), попавших в *j*-й кластер; $\mathbf{x}_{0_j}$ – точка центра *j*-го кластера в пространстве признаков; $d\left(\mathbf{x}_j^p;\vec{\mathbf{x}}_{0_j}\right)$ – евклидово расстояние от входного образа $\mathbf{x}_j^p$ до центра своего *j*-го кластера; $C_m^2$ – число сочетаний из *m* по 2; *m* – количество кластеров; $d^2\left(\mathbf{x}_{0_i};\mathbf{x}_{0_j}\right)$ – квадрат расстояния между центрами *i*-го и *j*-го кластеров.

Данный критерий качества исполняет в генетическом алгоритме роль фитнес-функции хромосом и рассчитывается по формуле[1]:

$$\mu(\mathbf{g}_l^t) = \overline{S}_t / \overline{R}_t. \tag{3}$$

Кластеризация с бóльшим значением фитнес-функции характеризуется более плотной группировкой образов кластеризуемых объектов вокруг центра своего кластера в пространстве признаков и более удаленным расположением этих центров друг от друга.

В алгоритме используются генетические операторы селекции, кроссовера, мутации, а также новый оператор фильтрации, исключающий из рассмотрения хромосомы, отображающие недопустимую кластеризацию, то есть не соответствующую требуемому числу кластеров. Вероятность такого события достаточно высока ввиду случайного характера выбора вышеназванных значений компонентов при формировании начальной популяции хромосом, а также вследствие работы других генетических операторов. Истинность такого события определяется значением функции $\phi(\mathbf{g}_l^t)$:

$$\phi(\mathbf{g}_l^t) = \begin{cases} 1 \text{ если } \exists\, r \in [1,k] \text{ такое, что } g_{lr}^t = j \text{ для } \forall j \in [1,m] \\ 0 \text{ если } \exists\, j \in [1,m] \text{ такое, что } \nexists\, r \in [1,k] \text{ , для которого } g_{ir}^t = j \end{cases} \tag{4}$$

В результате выполнения данного оператора рассматриваемая хромосома либо продолжает участвовать в дальнейшей работе алгоритма при $\phi(\mathbf{g}_l^t) = 1$, либо она удаляется

[1] Для других метрических пространств или в случае использования категориальных данных могут использоваться другие формулы и подходы к оценке качества.

при $\phi(\mathbf{g}_l^t) = 0$. Другими словами, оператор фильтрации – Fil ставит в соответствие хромосоме, над которой он выполняется, либо ее саму, либо пустое множество:

$$\mathrm{Fil}(\mathbf{g}_l^t) = \begin{cases} \mathbf{g}_l^t \text{ если } \phi(\mathbf{g}_l^t) = 1 \\ \emptyset \text{ если } \phi(\mathbf{g}_l^t) = 0 \end{cases} \quad (5)$$

Основное описание предлагаемого алгоритма представлено блок-схемой, которая показана на рис. 1. Ниже приводятся пояснения выполняемых шагов алгоритма, за исключением шагов, описанных непосредственно в блок-схеме.

*Шаг 1.* Исходные данные для работы алгоритма включают множество векторов $\{\mathbf{x}^p\}$, $p = \overline{1,k}$ (см. выше), количество искомых кластеров – $m$, количество группируемых объектов – $k$, количество хромосом в популяции – $w$, количество поколений в процессе эволюции – $L$, вероятность выполнения генетического оператора кроссовера – $P_c$, вероятность выполнения генетического оператора мутации – $P_m$.

*Шаг 4.* Необходимую последовательность равномерно распределенных целых чисел из диапазона [1, $m$] можно получить, воспользовавшись соответствующей вычислимой функцией R($a$, $b$), использующей в свою очередь вычислимую функцию Rnd – генератор

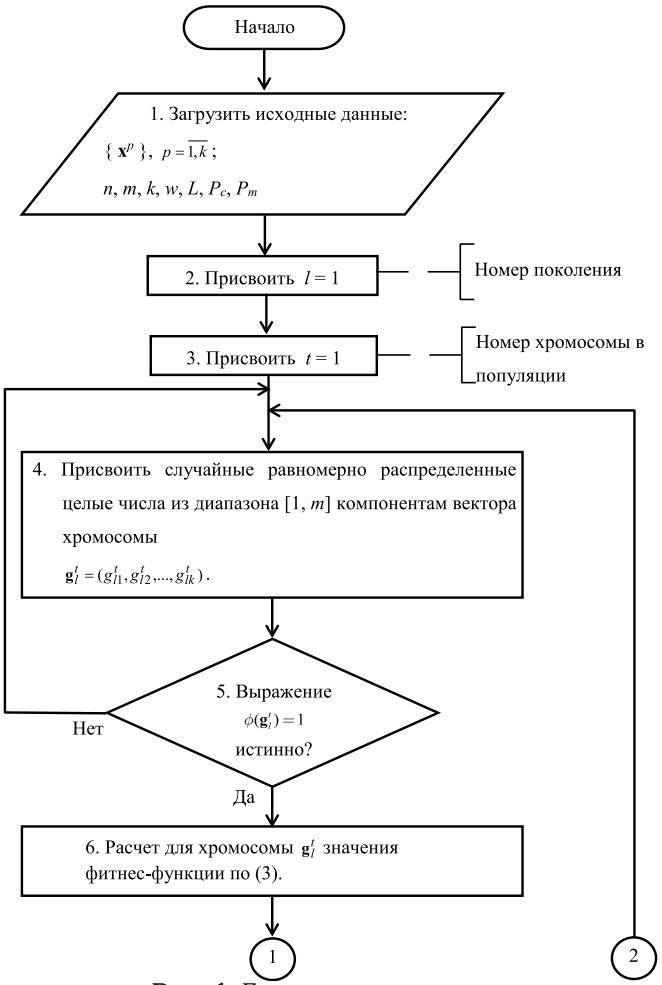


**Рис. 1.** Блок-схема алгоритма.

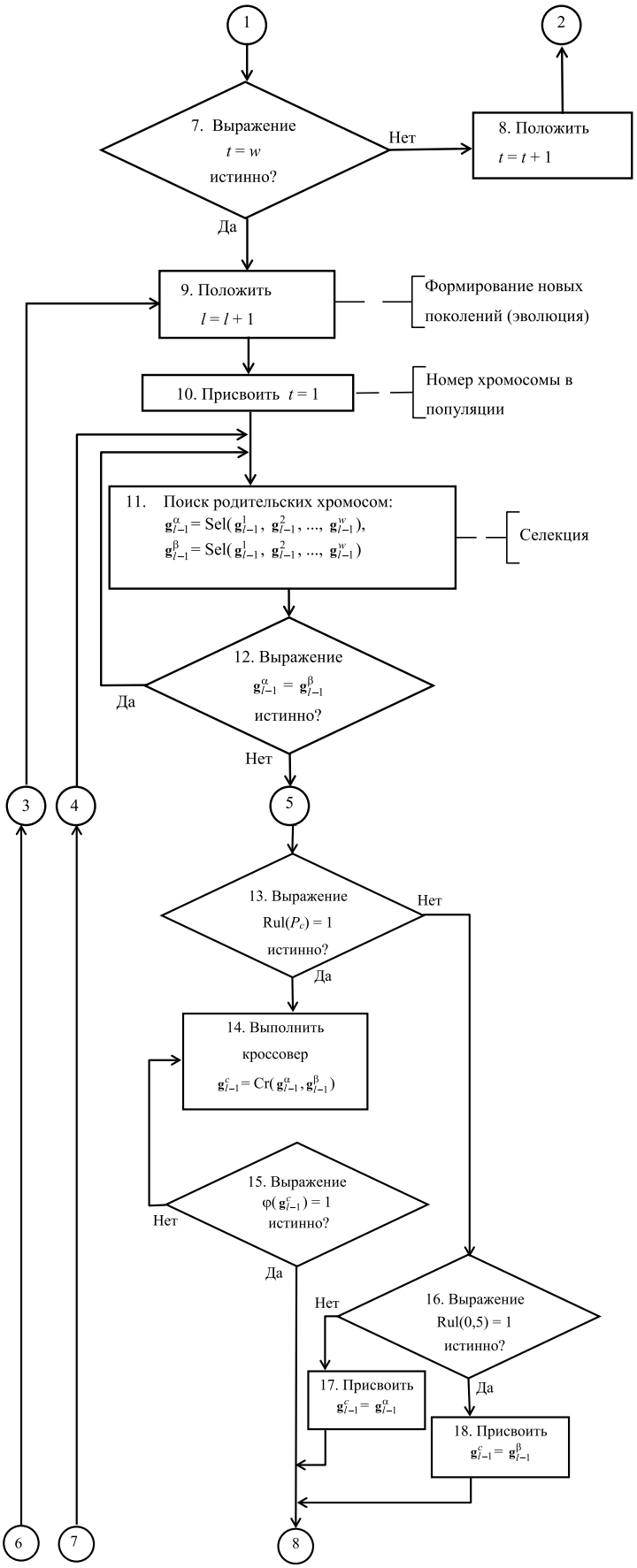


**Рис. 1.** Продолжение

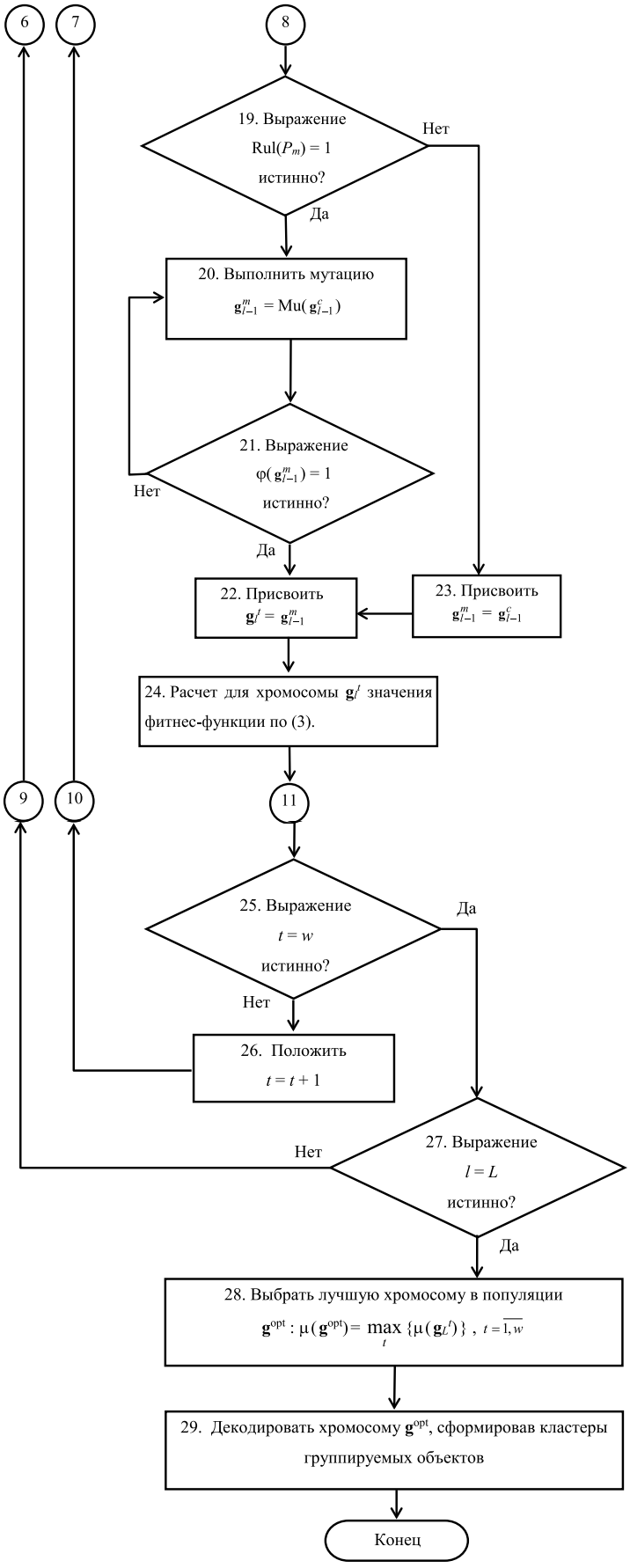


**Рис. 1.** Окончание

равномерно распределенных случайных действительных чисел из диапазона [0, 1]. Для генерации равномерно распределенных случайных чисел можно воспользоваться одним их известных методов, например, линейным конгруэнтным методом.

Один из возможных вариантов построения алгоритма функции R(*a, b*) приведен ниже, где *a* и *b* – границы диапазона [*a, b*] генерируемых целых чисел.

*Алгоритм А.* Вычислить значение функции R(a, b):

А1. Задать исходные данные – целые числа *a* и *b* (*a* < *b*).

А2. Рассчитать приращение $\Delta = 1/(b - a + 1)$.

А3. Получить равномерно распределенное в интервале [0, 1] вещественное случайное число $\lambda$ = Rnd.

А4. Положить $i = 1$.

А5. Проверить: логическое выражение $(i-1)\cdot\Delta \le \lambda \le i \cdot\Delta$ истинно? Если да, то закончить. Присвоить функции R(*a, b*) искомое значение: $a + (i-1)$.

А6. Положить $i = i + 1$ и перейти к шагу А5.

*Шаг 5.* Функция $\phi(\mathbf{g}_l^t)$ вычисляется согласно (4).

*Шаг 11.* На данном шаге используется вычислимый оператор селекции Sel, отбирающий хромосому для дальнейшего воспроизводства по принципу «выживает лучший». При отборе участвует процедура «виртуальная рулетка», позволяющая хромосомам с бóльшим значением фитнес-функции быть отобранными с бóльшей вероятностью. Ниже приведен алгоритм оператора селекции.

*Алгоритм В.* Выполнить селекцию – отобрать хромосому[2]
$\mathbf{g}_{l-1}^{\alpha}$ = Sel( $\mathbf{g}_{l-1}^{1}$, $\mathbf{g}_{l-1}^{2}$, ..., $\mathbf{g}_{l-1}^{w}$ ):

В1. Задать исходные данные – множество хромосом $\{\mathbf{g}_{l-1}^{t}\}$ и множество значений соответствующих им фитнес-функций $\{\mu(\mathbf{g}_{l-1}^{t})\}$, $t = \overline{1,w}$.

В2. Рассчитать суммарное значение фитнес-функции по всей популяции $S = \sum_{t=1}^{w} \mu(\mathbf{g}_{l-1}^{t})$.

В3. Получить равномерно распределенное в интервале [0, 1] вещественное случайное число $\lambda$ = Rnd.

В4. Положить $\gamma = 1$ и $t = w$.

В5. Проверить: логическое выражение $(\gamma - \frac{\mu(\mathbf{g}_{l-1}^{t})}{S}) \le \lambda \le \gamma$ истинно? Если да, то закончить: $\mathbf{g}_{l-1}^{\alpha} = \mathbf{g}_{l-1}^{t}$.

В6. Положить $\gamma = \gamma - \frac{\mu(\mathbf{g}_{l-1}^{t})}{S}$ и $t = t - 1$. Перейти к шагу В5.

*Шаг 13.* Используется вычислимая функция Rul($P_c$), которая возвращает значение либо 1 в случае наступления случайного события с вероятностью $P_c$, либо 0 в противном случае. Алгоритм вычисления приведен ниже.

*Алгоритм С.* Вычислить значение функции Rul($P_c$):

С1. Задать исходные данные – значение вероятности $P_c$.

С2. Получить равномерно распределенное в интервале [0, 1] вещественное случайное число $\gamma$ = Rnd.

С3. Проверить: логическое выражение $\gamma \le P_c$ истинно? Если да, то присвоить функции Rul($P_c$) искомое значение 1, если нет – значение 0.

[2] Для хромосомы $\mathbf{g}_{\beta}$ алгоритм аналогичен.

С4. Закончить.

*Шаг 14.* На данном шаге используется вычислимый оператор кроссовера Cr, скрещивающий две хромосомы $\mathbf{g}_{l-1}^{\alpha}$ и $\mathbf{g}_{l-1}^{\beta}$ и конструирующий в результате новую хромосому для помещения ее в новую популяцию. Для этого равновероятным случайным образом определяется точка кроссовера $r_c \in [2, k–1]$, т. е. номер «гена», с которого будет производиться обмен значениями рассматриваемых хромосом. Для этого можно воспользоваться рассмотренной выше вычислимой функцией R(2, $k$–1). Ниже приведен алгоритм оператора кроссовера.

*Алгоритм D.* Выполнить кроссовер – отобрать хромосому $\mathbf{g}_{l-1}^{c} = \mathrm{Cr}(\mathbf{g}_{l-1}^{\alpha}, \mathbf{g}_{l-1}^{\beta})$:

D1. Задать исходные данные – хромосомы $\mathbf{g}_{l-1}^{\alpha}$, $\mathbf{g}_{l-1}^{\beta}$.

D2. Определить точку кроссовера $r_c = \mathrm{R}(2, k–1)$.

D3. Построить хромосомы $\tilde{\mathbf{g}}_{l-1}^{\alpha}$, $\tilde{\mathbf{g}}_{l-1}^{\beta}$ :

$\tilde{\mathbf{g}}_{l-1}^{\alpha} = (g_{(l-1)1}^{\alpha}, g_{(l-1)2}^{\alpha}, ..., g_{(l-1)(r_c-1)}^{\alpha}, g_{(l-1)r_c}^{\beta}, ..., g_{(l-1)k}^{\beta})$,

$\tilde{\mathbf{g}}_{l-1}^{\beta} = (g_{(l-1)1}^{\beta}, g_{(l-1)2}^{\beta}, ..., g_{(l-1)(r_c-1)}^{\beta}, g_{(l-1)r_c}^{\alpha}, ..., g_{(l-1)k}^{\alpha})$.

D4. Проверить: логическое выражение $\mathrm{Rul}(P_c) = 1$ истинно? Если да, то $\mathbf{g}_{l-1}^{c} = \tilde{\mathbf{g}}_{l-1}^{\alpha}$, если нет – $\mathbf{g}_{l-1}^{c} = \tilde{\mathbf{g}}_{l-1}^{\beta}$.

D4. Закончить.

*Шаг 20.* На данном шаге используется вычислимый оператор мутации Mu, заменяющий значение случайно определенного «гена» с номером $r_m \in [1, k]$ на новое случайно определенное значение с равной вероятностью из интервала [1, $m$]. Поиск значения $r_m$ осуществляется также с равной вероятностью, для чего можно воспользоваться рассмотренной выше вычислимой функцией R(1, $k$). Ниже приведен алгоритм оператора мутации.

*Алгоритм E.* Выполнить мутацию хромосомы $\mathbf{g}_{l-1}^{c}$ :

E1. Задать исходные данные – хромосому $\mathbf{g}_{l-1}^{c}$ .

E2. Определить номер мутирующего «гена»: $r_m = \mathrm{R}(1, k)$.

E3. Присвоить новое значение мутирующему «гену»:

$g_{(l-1)r_m}^{c} = \mathrm{R}(1, m)$.

E4. Закончить.

*Шаг 28.* Поиск лучшей хромосомы $\mathbf{g}^{\mathrm{opt}}$ в популяции последнего поколения ($L$ – номер последнего поколения) осуществляется обычным перебором значений фитнес-функций всех хромосом с поиском хромосомы, имеющей максимальное значение данной функции.

*Шаг 29.* Декодирование найденной хромосомы предполагает сортировку ее компонентов («генов») по их значению, поскольку каждый компонент вектора хромосомы соответствует группируемому объекту, а его значение – номеру кластера.

## Исследование алгоритма кластеризации

Поскольку работа алгоритма представляет собой стохастический процесс, то его сходимость является также категорией вероятностной, требующей своей оценки. Для выполнения данного анализа были проведены исследования эффективности работы алгоритма в несколько этапов.

На первом этапе были подготовлены исходные данные в виде множества многомерных векторов значений признаков абстрактных кластеризуемых объектов. Значения

подбирались случайным образом с помощью несложной программы. При этом разброс координат объектов по измерениям внутри кластеров поддерживался в рамках необходимой дисперсии. Данные готовились для двух серий вычислительных экспериментов – с сильно и слабо выраженным удалением кластеров. Во втором блоке данных расстояние между кластерами, оцениваемое методом «ближнего соседа», было соизмеримо с дисперсией значений координат внутри кластеров.

На втором этапе были проведены серии вычислительных экспериментов с использованием специально разработанной для этой цели программы. Работа алгоритма в рамках данной программы выполнялась в различных режимах, отличающихся варьированием размера популяции хромосом w и количества отработанных алгоритмом поколений эволюции $L$. Значения параметров вероятности для генетических операторов во всех экспериментах задавались следующими: $P_c = 0.9$ и $P_m = 0.1$.

Результаты экспериментов на обоих блоках данных показывают (см. рис. 2, 3), что процесс эволюции хромосом стабилизируется уже при небольших размерах их популяции ($w > 150$) и числе реализованных поколений эволюции ($L > 200$).

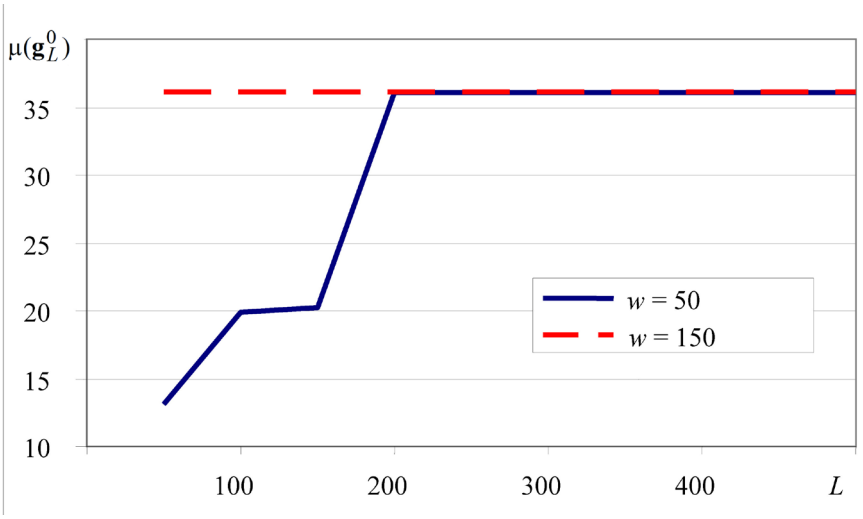


**Рис. 2.** К определению параметров стабилизации работы алгоритма при сильно выраженной кластеризации.

Значение параметра качества кластеризации Θ выбиралось равным значению фитнес-функции $\mu(\mathbf{g}_L^0)$ ($\mathbf{g}_L^0$ – лучшая хромосома, отображающая результат кластеризации). При сильно выраженной кластеризации, как и предполагалось, значение Θ оказалось выше, чем во втором случае (36.1079 > 3.1977) вследствие большей удаленности кластеров друг от друга.

Значение параметра качества на режимах работы алгоритма ниже стабилизационных описывается эмпирическими функциональными зависимостями, полученными по результатам тех же экспериментов для каждого блока данных (см. рис. 4, 5). Что касается сильно выраженной кластеризации, то эта зависимость имеет вид

$$\Theta = -9{,}759 + 0{,}569w + 0{,}084L - 0{,}001836w^2 - 0{,}000328L^2 . \quad (6)$$

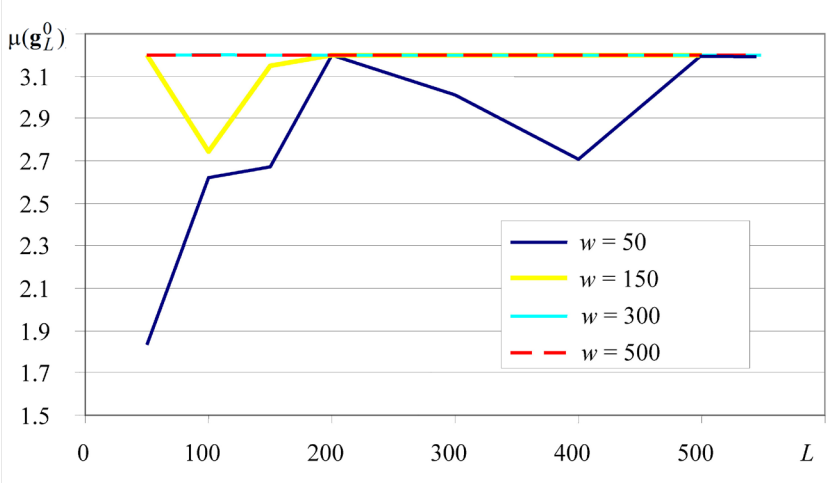


**Рис. 3.** К определению параметров стабилизации алгоритма при слабо выраженной кластеризации.

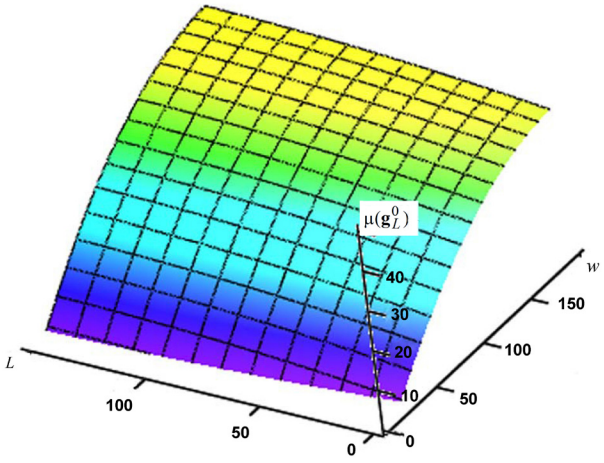


**Рис. 4.** Работа алгоритма на переходных режимах: сильно выраженная кластеризация.

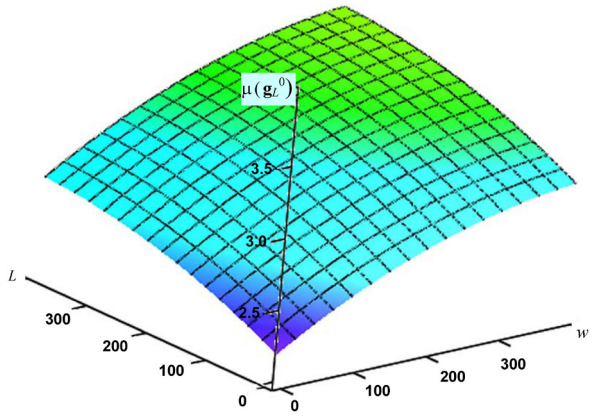


**Рис. 5.** Работа алгоритма на переходных режимах: слабо выраженная кластеризация.

Для случая слабо выраженной кластеризации мы имеем функцию

$$\Theta = 2{,}1549 + 0{,}0053w + 0{,}0035L - 0{,}00000915w^2 - 0{,}00000646L^2. \quad (7)$$

Обе эмпирические зависимости (6) и (7) были проверены на адекватность по критерию Фишера.

На третьем этапе ставился вычислительный сравнительный эксперимент на реальных данных, использованных автором ранее при выявлении потенциальных объектов выездных налоговых проверок (предприятий) в рамках налогового администрирования [12]. Пространство признаков предприятий, используемых в данной задаче, состоит из 16 финансовых показателей, входящих в состав методических положений по оценке финансового состояния предприятий и установлению неудовлетворительной структуры баланса, утвержденной Федеральным управлением по делам несостоятельности (банкротства) РФ. Показатели образуют четыре группы и характеризуют рентабельность предприятия,

ликвидность и его платежеспособность, деловую активность и финансовую устойчивость (подробно см. [12]).

Для сравнения предлагаемого алгоритма с существующими в первую очередь был выбран SOM-алгоритм [22], который по характеру является также стохастическим и гибридным, а также один из популярных алгоритмов – "*k*-means" [21]. Вычисления с использованием нейронных сетей Кохонена и алгоритма "*k*-means" выполнялись в программе Deductor Studio Basegroup Labs. В пространстве вышеназванных 16 признаков выполнялась кластеризация 24 предприятий с формированием трех кластеров, как и в задаче поддержки налогового управления [36]. Расчет среднего значения параметра качества кластеризации $\Theta'' = \overline{S}^0 / \overline{R}^0$ [3] ($\overline{S}^0$ и $\overline{R}^0$ рассчитаны для лучшей хромосомы популяции по (1) и (2)) по результатам повторных прогонов SOM-алгоритма составил 3.97, а алгоритма "*k*-means" – 2.04. Использование же предлагаемого алгоритма в данной задаче при размерах популяции хромосом ($w$) и числе реализованных поколений эволюции ($L$), равных 200, показало среднее значение $\Theta'' = \mu(\mathbf{g}_L^0) = 7.29$. Результат оказался вполне ожидаемым, так как алгоритмы SOM и "*k*-means" учитывают только один критерий кластеризации, тогда как в задаче [36] лучшее решение оценивается двумя критериями, такими же, как и в предлагаемом алгоритме.

Чтобы поставить сравниваемые алгоритмы в равное положение, не связанное с приоритетами принятия решений в [36], в предлагаемом алгоритме была изменена фитнес-функция (см. (3)) на значение $\mu(\mathbf{g}_l^t) = 1/\overline{R}_t$, учитывающее только один критерий кластеризации – близость образов кластеризуемых объектов к центру кластера в пространстве измерений. Именно этот критерий используется в алгоритмах SOM и "*k*-means". По результатам новой серии экспериментов были получены следующие значения критерия качества $\Theta' = 1/\overline{R}^0$: для предлагаемого алгоритма – 0.147, алгоритма SOM – 0.089, алгоритма "*k*-means" – 0.072.

Таким образом, в обоих случаях качество кластеризации оказалось лучше в предлагаемом алгоритме. Это объясняется тем, что в данном алгоритме пространство поиска покрывает все допустимое множество решений (генетический оператор мутации обеспечивает появление в рассматриваемом множестве таких решений, которых не было заложено в генетическом материале начальной стартовой популяции), в то время как в сравниваемых алгоритмах пространство решений зависит от их инициализации (стартовых значений координат центров кластеров в "*k*-means" и координат нейронов в SOM).

Сравнительная оценка алгоритмов по производительности учитывала то обстоятельство, что время работы алгоритма SOM напрямую зависит от количества эпох обучения нейронной сети входными данными, а исследуемого алгоритма – от размеров популяции хромосом и числа реализованных поколений эволюции. В процессе повторных вычислений было выявлено, что стабилизация алгоритма SOM по точности кластеризации наступает в диапазоне от 1500 до 2000 эпох. Для предпочтительной для данного алгоритма оценки было определено время его работы при реализации 1500 эпох обучения нейронной сети на 32-разрядном микропроцессоре с тактовой частотой 1.7 ГГц. Среднее значение времени его работы составило 4.5 с. Среднее время работы предлагаемого алгоритма при $w = 200$ и $L = 200$ на том же микропроцессоре составило 0.89 с, а алгоритма "*k*-means" – 0.64 с. Это говорит о соразмерности производительности сравниваемых алгоритмов.

[3] Сравните с (3).

Для более детального исследования производительности предлагаемого алгоритма была проведена серия вычислений при различных значениях $w$ и $L$. По полученным результатам (см. таблицу) выполнена оценка воспроизводимости эксперимента с использованием G критерия Кохрена. Расчетное значение критерия (0.299) не превышает табличного (0.478) при уровне значимости 0.05, что не просто подтверждает случайный характер расхождения дисперсий значений времени в повторных вычислениях, но и свидетельствует о стабильной надежной работе алгоритма, результат которого не зависит от его инициализации.

К оценке производительности алгоритма

| Размер популяции ($w$) | Число поколений ($L$) | Время работы алгоритма при повторных вычислениях, с | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| 200 | 200 | 0.859 | 0.907 | 0.911 |
| 200 | 300 | 1.359 | 1.361 | 1.375 |
| 200 | 400 | 1.781 | 1.797 | 1.812 |
| 300 | 200 | 1.765 | 1.735 | 1.750 |
| 300 | 300 | 2.562 | 2.547 | 2.625 |
| 300 | 400 | 3.469 | 3.391 | 3.531 |
| 400 | 200 | 2.687 | 2.828 | 2.718 |
| 400 | 300 | 4.125 | 4.109 | 4.281 |
| 400 | 400 | 5.421 | 5.532 | 5.359 |

С использованием полученной статистики (см. таблицу) построена регрессионная модель зависимости времени работы алгоритма ($\tau$) в данном эксперименте от параметров $w$ и $L$ с высоким значением множественного коэффициента детерминации ($R^2 = 0.97$):

$$\tau = -4{,}11639 + 0{,}01383w + 0{,}00885L, \quad (8)$$

позволяющая более полно оценивать производительность предлагаемого алгоритма при $w > 150$ и $L > 200$.

## Заключение

Представленный генетический алгоритм кластеризации является гибким по отношению к предпочтениям аналитика в процессе принятия решений, так как позволяет выполнять кластеризацию, исходя из различных критериев, например, максимального взаимного удаления кластеров, близости геометрических образов группируемых объектов к центру кластера, других критериев. Это достигается изменением расчетной формулы фитнес-функции, учитывающей необходимое сочетание критериев кластеризации без изменения структуры алгоритма.

Как и любой генетический алгоритм, данный алгоритм является простым в его программной реализации, что делает его надежным в использовании. Высокая надежность алгоритма была подтверждена проведенными вычислительными экспериментами на данных, отражающих различные кластерные структуры. Эксперименты показали быструю сходимость алгоритма, оцениваемую по небольшому числу поколений эволюции в про-

цессе нахождения искомого решения. При этом необходимы небольшие вычислительные ресурсы компьютера в виде его оперативной памяти, так как надежная работа алгоритма не требует большой популяции генетических хромосом.

Алгоритм отличает нечувствительность к инициализации, так как в процессе эволюции хромосом за счет использования генетических операторов алгоритм полностью покрывает все множество допустимых решений, что, в итоге, обеспечивает высокое качество кластеризации. Проведенный вычислительный эксперимент на реальных данных показал не только хорошее качество кластеризации, но и вполне приемлемую производительность представленного алгоритма, соразмерную с производительностью общепризнанных алгоритмов SOM и "*k*-means".

## Литература:

1. Нечеткие системы, мягкие вычисления и интеллектуальные технологии: Труды VII Всероссийской научно-практической конференции. Санкт-Петербург, 03–07 июля 2017 г. Т. 2. СПб.: Политехника-сервис, 2017. 210 с.

2. Юдин В.Н., Карпов Л.Е. Неполностью описанные объекты в системах поддержки принятия решений // Программирование. 2017. № 5. С. 24–31.

3. Анфёров М.А. Системная оптимизация наукоемких технологий // Известия вузов. Авиационная техника. 2002. № 2. С. 57–60.

4. Батыршин И.З., Недосекин А.А., Стецко А.А., Тарасов В.Б., Язенин А.В., Ярушкина Г.Г. Нечеткие гибридные системы: теория и практика / Под ред. Н.Г. Ярушкиной. М.: Физматлит, 2007. 207 с.

5. Аджемов С.С., Кленов Н.В., Терешонок М.В., Чиров Д.С. Использование искусственных нейронных сетей для классификации источников сигналов в системах когнитивного радио // Программирование. 2016. № 3. С. 3–11.

6. Hu Z., Bodyanskiy Y., Tyshchenko O.K. Self-learning procedures for a kernel fuzzy clustering system // Advances in Computing Science for Engineering and Education. 2019. V. 754. P. 487–497. http://dx.doi.org/10.1007/978-3-319-91008-6_49

7. Анфёров М.А., Ханнанов М.Г. Кластерный подход к проектированию в САПР ТП // Проведение научных исследований в области обработки, хранения, передачи и защиты информации. Сб. науч. тр. в 4 т. Т. 3. Ульяновск: УлГТУ, 2009. С. 60–65.

8. Бороздина Н.А. Применение иерархического кластерного анализа для сегментации потребителей рынка сотовой связи // Молодой ученый. 2016. № 29. С. 365–367. URL: https://moluch.ru/archive/133/37358/ (дата обращения: 13.11.2019).

9. Дударин П.В., Ярушкина Н.Г. Подходы к нечеткой и иерархической кластеризации и классифиации ключевых показателей эффективности системы стратегического планирования российской федерации // Труды VII Всероссийской научно-практической конференции «Нечеткие системы, мягкие вычисления и интеллектуальные технологии». Санкт-Петербург, 03–07 июля, 2017г. Т. 2. СПб.: Политехника-сервис, 2017. С. 65–73.

10. Петухова. М.В. Кластеризация заемщиков – физических лиц по уровню дефолтов: рейтинговый подход (на примере регионов Сибирского федерального округа) // Журнал Новой экономической ассоциации. 2012. № 4 (16). С. 71–102.

11. Анфёров М.А. Сети Кохонена в задаче выявления экономически нестабильных региональных структур // Труды XV Всероссийской научно-технической конференции «Нейроинформатика 13». Москва, 21–25 января, 2013 г. Т. 3. М: НИЯУ МИФИ, 2013. С. 177–184.

12. Анфёров М.А., Рашитова О.Б. SADT моделирование системы налогообложения в Российской Федерации // Экономика и управление: научно-практический журнал. 2015. № 2(124). С. 94–101.

13. González del Pozo R., García-Lapresta J.L., Pérez-Román D. Clustering U.S. 2016 presidential candidates through linguistic appraisals // Advances in Intelligent Systems and Computing. 2018. V. 642. P. 143–153. https://doi.org/10.1007/978-3-319-66824-6_13

14. Хвостиков А.В., Крылов А.С., Камалов Ю.Р. Текстурный анализ ультразвуковых изображений для диагностирования фиброза печени // Программирование. 2015. № 5. С. 39–46.

15. Kumar S., Mishra S., Asthana P. Automated detection of acute leukemia using k-mean clustering algorithm // Advances in Intelligent Systems and Computing. 2018. V. 554. P. 655–670. https://doi.org/10.1007/978-981-10-3773-3_64

16. Abadi S., Sari T.I., Maseleno A., Muslihudin M., Mat The K.S., Nasir B.M., Huda M., Ivanova N.L., Satria F. Application model of k-means clustering: insights into promotion strategy of vocational high school // International Journal of Engineering and Technology. 2018. V. 7. № 2.27. P. 182–187. http://dx.doi.org/10.14419/ijet.v7i2.11491

17. Hussain S., Atallah R., Kamsin A., Hazarika J. Classification, clustering and association rule mining in educational datasets using data mining tools: A case study // Advances in Intelligent Systems and Computing. 2019. V. 765. P. 196–211. https://doi.org/10.1007/978-3-319-91192-2_21

18. Харинов М.В. Кластеризация пикселей для сегментации цветового изображения // Программирование. 2015. № 5. С. 20–30.

19. Астраханцев Н.А., Федоренко Д.Г. Турдаков Д.Ю. Методы автоматического извлечения терминов из коллекции текстов предметной области // Программирование. 2015. № 6. С. 33–52.

20. Lakhno V., Zaitsev S., Tkach Y., Petrenko T. Adaptive expert systems development for cyber attacks recognition in information educational systems on the basis of signs' clustering // Advances in Intelligent Systems and Computing. 2019. V. 754. P. 673–682. https://doi.org/10.1007/978-3-319-91008-6_66

21. Hartigan, J.A., Wong, M.A. Algorithm AS 136: A k-means clustering algorithm // Journal of the Royal Statistical Society, Series C (Applied Statistics). 1979. V. 28. № 1. P. 100–108.

22. Kohonen T. Self-Organizing Maps: 3rd edition. Berlin - New York: Springer-Verlag, 2001. 521 p.

23. Zhang T., Ramakrishnan R., Livny M. BIRCH: an efficient data clustering method for very large databases // Proceedings of the ACM SIGMOD international conference on Management of data (SIGMOD '96). 1996. P. 103–114. https://doi.org/10.1145/235968.233324

24. Päivinen N. Clustering with a minimum spanning tree of scale-free-like structure // Pattern Recognition Letters. 2005. V. 26. Iss. 7. P. 921–930. https://doi.org/10.1016/j.patrec.2004.09.039

25. Sudipto Guha, Rajeev Rastogi, Kyuseok Shim. CURE: An Efficient Clustering Algorithm for Large Databases // Information Systems. 1998. V. 26. № 1. P. 35–58. https://doi.org/10.1016/S0306-4379(01)00008-4

26. Sudipto Guha, Rajeev Rastogi, Kyuseok Shim. ROCK: a robust clustering algorithm for categorical attributes // Information Systems. 2000. V. 25. № 5. P. 345–366. https://doi.org/10.1016/S0306-4379(00)00022-3

27. Bodyanskiy Y., Didyk O. On-line robust fuzzy clustering for anomalies detection // Advances in Intelligent Systems and Computing. 2019. V. 754. P. 402–409. https://doi.org/10.1007/978-3-319-91008-6_40

28. Иванова Е.В., Соколинский Л.Б. Методы параллельной обработки сверхбольших баз данных с использованием распределенных колоночных индексов // Программирование. 2017. № 3. С. 3–21.

29. Shao J., Yang Q., Schmidt B., Dang H-V., Kramer S. Scalable Clustering by Iterative Partitioning and Point Attractor Representation // ACM Transactions on Knowledge Discovery from Data. 2016. V. 11. № 1. P. 5:1–5:23. https://doi.org/10.1145/2934688

30. Songlei J.https://orcid.org/0000-0001-5760-6431, Guansong P., Longbing C., Kai L., Hang G. CURE: Flexible Categorical Data Representation by Hierarchical Coupling Learning // IEEE Transactions on Knowledge and Data Engineering. 2019. V. 31. Iss. 5. P. 853–866. https://doi.org/10.1109/TKDE.2018.2848902

31. Sheikholeslami G., Chatterjee S., Zhang A. WaveCluster: A Wavelet-Based Clustering Approach for Spatial Data // VLDB Journal. 2000. № 8(3-4). P. 289–304. http://dx.doi.org/10.1007/s007780050009

32. Gionis A., Mannila H., Tsaparas P. Clustering Aggregation // ACM Transactions on Knowledge Discovery from Data. 2007. V. 1. № 1. Article 4. 30 p. https://doi.org/10.1145/1217299.1217303

33. Wang C., She Z., Stantic B., Chi C.H., Cao L. Coupled Clustering Ensemble by Exploring Data Interdependence // ACM Transactions on Knowledge Discovery from Data. 2018. V. 12. № 6. P. 63:1-63:38. http://dx.doi.org/10.1145/3230967

34. Zhang X., Zhang X., Liu H. Smart Multitask Bregman Clustering and Multitask Kernel Clustering // ACM Transactions on Knowledge Discovery from Data. 2015. V. 10. № 1. P. 8:1-8: p. https://doi.org/10.1145/2747879

35. Abasi A., Sajedi H. Fuzzy-clustering based data gathering in wireless sensor network // International Journal on Soft Computing (IJSC). 2016. V.7. № 1. P.1–15. https://doi.org/10.5121/ijsc.2016.7101

36. Горбатков С.А., Рашитова О.Б. Моделирование налоговых управленческих решений на основе нейронных сетей Кохонена // Информационные технологии. 2013. № 5. С. 60–65.

## References:

1. Fuzzy systems, soft calculations and intellectual technologies: Proceedings of the VII All-Russian Scientific and Practical Conference. St. Petersburg, July 03-07, 2017. V. 2. St. Petersburg: Politekhnika-servis Publ., 2017. 210 p. (in Russ.).

2. Yudin V.N., Karpov L.E. Incompletely described objects in decision support. *Programming and Computer Software.* 2017;43(5):294-299.

3. Anfyorov M.A. System optimization of high technologies. *Izv. VUZ. Aviatsionnaya tekhnika* = Russian Aeronautics. 2002;2:57-60 (in Russ.).

4. Batyrshin I.Z., Nedosekin A.A., Stetsko A.A., Tarasov V.B., Yazenin A.V., Yarushkina N.G. Fuzzy hybrid systems: theory and practice. Ed. N.G. Yarushkina. Moscow: Fizmatlit Publ., 2007. 207 p. (in Russ.).

5. Adzhemov S.S., Klenov N.V., Tereshonok M.V., Chirov D.S. The use of artificial neural networks for classification of signal sources in cognitive radio systems. *Programming and Computer Software.* 2016;42(3):121-128. 10.1134/S0361768816030026

6. Hu Z., Bodyanskiy Y., Tyshchenko O.K. Self-learning procedures for a kernel fuzzy clustering system. *Advances in Computing Science for Engineering and Education.* 2019;754:487-497. http://dx.doi.org/10.1007/978-3-319-91008-6_49

7. Anfyorov M.A., Khannanov M.G. Cluster approach to design in CAM. In: Provedeniye nauchnykh issledovaniy v oblasti obrabotki, khraneniya, peredachi i zashchity informatsii = Conducting scientific research in the field of

processing, storage, transmission and protection of information. Collection of scientific papers in 4 v. V. 3. Ul'yanovsk: UlGTU Publ., 2009; pp. 60-65 (in Russ.).

8. Borozdina N.A. The use of hierarchical cluster analysis for segmentation of consumers of the market of cellular communication. *Molodoi uchenyi* = Young scientist. 2016;29:365-367. (in Russ.). URL: https://moluch.ru/archive/133/37358/ (accessed November 13, 2019).

9. Dudarin P.V., Yarushkina N.G. Approaches to fuzzy and hierarchical clustering and classification of key process indicators of the strategic planning system of the Russian Federation. In: Proceedings of the VII All-Russian Scientific and Practical Conference "Fuzzy Systems, Soft Computing and Intelligent Technology". St. Petersburg, July 03–07, 2017. V. 2. SPb.: Polytekhnik servis Publ., 2017; pp. 65-73 (in Russ.).

10. Petukhova M.V. Clustering of borrowers at the level of defaults: Rating approach (regions of Siberian Federal District). *Zhurnal Novoi ekonomicheskoi assotsiatsii* = J. New Economic Association. 2012;4(16):71-102 (in Russ.).

11. Anfyorov M.A. Kohonen networks in a problem of identification of economically unstable regional structures. Proceedings of the XV All-Russian Scientific and Practical Conference "Nejroinformatika 13" [Neuroinformatics-2013]. Moscow, 21–25 January, 2013. V. 3. M.: NIYaU MIFI, 2013; pp. 177-184 (in Russ.).

12. Anfyorov M.A., Rashitova O.B. SADT modeling of the Russian Federation tax system. *Ekonomika i upravlenie: nauchno-prakticheskii zhurnal* = Economics and Management: Research and Practice Journal. 2015;2(124):94-101 (in Russ.).

13. González del Pozo R., García-Lapresta J.L., Pérez-Román D. Clustering U.S. 2016 presidential candidates through linguistic appraisals. *Advances in Intelligent Systems and Computing.* 2018;642:143-153. https://doi.org/10.1007/978-3-319-66824-6_13

14. Kvostikov A.V., Krylov A.S., Kamalov U.R. Ultrasound image texture analysis for liver fibrosis stage diagnostics. *Programming and Computer Software.* 2015;41(5):273-278. https://doi.org/10.1134/S0361768815050059

15. Kumar S., Mishra S., Asthana P. Automated detection of acute leukemia using k-mean clustering algorithm. *Advances in Intelligent Systems and Computing.* 2018;554:655-670. https://doi.org/10.1007/978-981-10-3773-3_64

16. Abadi S., Sari T.I., Maseleno A., Muslihudin M., Mat The K.S., Nasir B.M., Huda M., Ivanova N.L., Satria F. Application model of k-means clustering: insights into promotion strategy of vocational high school. *International Journal of Engineering and Technology.* 2018;7(2.27):182-187. http://dx.doi.org/10.14419/ijet.v7i2.11491

17. Hussain S., Atallah R., Kamsin A., Hazarika J. Classification, Clustering and Association Rule Mining in Educational Datasets Using Data Mining Tools: A Case Study. *Advances in Intelligent Systems and Computing.* 2019;765:196-211. https://doi.org/10.1007/978-3-319-91192-2_21

18. Kharinov M.V. Pixel clustering for color image segmentation. *Programming and Computer Software.* 2015:41(5):258-266. https://doi.org/10.1134/S0361768815050047

19. Astrakhantsev N.A., Fedorenko D.G. Turdakov D.YU. Methods for automatic term recognition in domain-specific text collections: A survey. *Programming and Computer Software.* 2015;41(6):336-349. https://doi.org/10.1134/S036176881506002X

20. Lakhno V., Zaitsev S., Tkach Y., Petrenko T. Adaptive expert systems development for cyber attacks recognition in information educational systems on the basis of signs' clustering. *Advances in Intelligent Systems and Computing.* 2019;754:673-682. https://doi.org/10.1007/978-3-319-91008-6_66

21. Hartigan, J.A., Wong, M. A. Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society, Series C (Applied Statistics).* 1979;28(1):100-108.

22. Kohonen T. Self-Organizing Maps: 3rd edition. Berlin - New York: Springer-Verlag, 2001. 521 p.

23. Zhang T., Ramakrishnan R., Livny M. BIRCH: an efficient data clustering method for very large databases. In: Proceedings of the ACM SIGMOD international conference on Management of data (SIGMOD '96). 1996; pp. 103-114. https://doi.org/10.1145/235968.233324

24. Päivinen N. Clustering with a minimum spanning tree of scale-free-like structure. *Pattern Recognition Letters.* 2005;26(7):921-930. https://doi.org/10.1016/j.patrec.2004.09.039

25. Sudipto Guha, Rajeev Rastogi, Kyuseok Shim. CURE: An Efficient Clustering Algorithm for Large Databases. *Information Systems.* 1998;26(1):35-58. https://doi.org/10.1016/S0306-4379(01)00008-4

26. Sudipto Guha, Rajeev Rastogi, Kyuseok Shim. ROCK: a robust clustering algorithm for categorical attributes. *Information Systems.* 2000;25(5):345-366. https://doi.org/10.1016/S0306-4379(00)00022-3

27. Bodyanskiy Y., Didyk O. On-line robust fuzzy clustering for anomalies detection. *Advances in Intelligent Systems and Computing.* 2019;754:402-409. https://doi.org/10.1007/978-3-319-91008-6_40

28. Ivanova E.V., Sokolinsky L.B. Parallel processing of very large databases with using distributed columnar indexes. *Programming and Computer Software.* 2017;43(3):131-144. https://doi.org/10.1134/S0361768817030069

29. Shao J., Yang Q., Schmidt B., Dang H-V., Kramer S. Scalable Clustering by Iterative Partitioning and Point Attractor Representation. *ACM Transactions on Knowledge Discovery from Data.* 2016;11(1):5:1-5:23. https://doi.org/10.1145/2934688

30. Songlei J.https://orcid.org/0000-0001-5760-6431, Guansong P., Longbing C., Kai L., Hang G. CURE: Flexible Categorical Data Representation by Hierarchical Coupling Learning. *IEEE Transactions on Knowledge and Data Engineering.* 2019;31(5):853-866. https://doi.org/10.1109/TKDE.2018.2848902

31. Sheikholeslami G., Chatterjee S., Zhang A. WaveCluster: A Wavelet-Based Clustering Approach for Spatial Data. *VLDB Journal.* 2000;8(2-4):289-304. http://dx.doi.org/10.1007/s007780050009

32. Gionis A., Mannila H., Tsaparas P. Clustering Aggregation. *ACM Transactions on Knowledge Discovery from Data.* 2007;1(1):Article 4. 30 p. https://doi.org/10.1145/1217299.1217303

33. Wang C., She Z., Stantic B., Chi C.H., Cao L. Coupled Clustering Ensemble by Exploring Data Interdependence. *ACM Transactions on Knowledge Discovery from Data.* 2018;12(6):63:1-63:38. https://doi.org/10.1145/3230967

34. Zhang X., Zhang X., Liu H. Smart Multitask Bregman Clustering and Multitask Kernel Clustering. *ACM Transactions on Knowledge Discovery from Data.* 2015;10(1):8:1-8:29. https://doi.org/10.1145/2747879

35. Abasi A., Sajedi H. Fuzzy-clustering based data gathering in wireless sensor network. *International Journal on Soft Computing (IJSC).* 2016:7(1):1-15. https://doi.org/10.5121/ijsc.2016.7101

36. Gorbatkov S.A., Rashitova O.B. Modeling of tax administrative decisions on the basis of Kohonen's neural networks. *Informatsionnye tekhnologii* = Information Technology. 2013;5:60-65 (in Russ.).

***Об авторе:***

***Анфёров Михаил Анисимович***, доктор технических наук, профессор, профессор кафедры «Прикладная и бизнес-информатика» Института комплексной безопасности и специального приборостроения ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78).

***About the author:***

***Mikhail A. Anfyorov,*** Dr. of Sci. (Engineering), Professor of the Chair "Applied and Business Informatics," Institute of Complex Security and Special Instrumentation, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow 119454, Russia).