УДК 004.8

Н.П. Корышев, И.А. Ходашинский

Алгоритм формирования базы правил нечёткого классификатора на основе алгоритма кластеризации *K*-средних и метаэвристического алгоритма «китов»

Представлены описание алгоритма генерации нечётких правил для нечёткого классификатора с использованием кластеризации данных, метаэвристики и индекса качества кластеризации, а также результаты проверки работоспособности на реальных наборах данных.

Ключевые слова: нечёткий классификатор, кластеризация, *K*-средние, алгоритм «китов», индексы качества кластеризации.
doi: 10.21293/1818-0442-2021-24-1-42-47

Нечёткий классификатор является одним из актуальных способов решения задачи классификации, пришедшим из области машинного обучения (как и нечёткие системы в целом); он использует нечёткие множества и нечёткую логику в качестве инструмента для представления знаний о решаемой проблеме [1]. В пространстве признаков нечёткая логика позволяет объекту принадлежать к разным классам одновременно с некоторой степенью принадлежности. Главным достоинством нечеткого классификатора является легкая интерпретируемость правил классификации.

Основным компонентом модели нечёткого классификатора является база нечётких правил. Задача формирования базы сводится к ответу на вопросы о том, сколько правил должно быть в базе и как сформировать антецеденты и консеквенты правил. Применяя подход построения на основе таблицы наблюдения, сформировать правила можно с помощью алгоритма по экстремумам классов [2]. Количество генерируемых указанным алгоритмом нечётких правил равно количеству классов в классифицируемом наборе данных. Это свойство является достоинством и одновременно недостатком алгоритма. Минимальное количество правил в нечётком классификаторе способствует лучшему пониманию его работы, уменьшает вычислительные затраты на его обучение. Однако на минимальном количестве правил не всегда удается получить необходимую точность классификации. Второй подход к формированию базы правил основан на применении методов кластеризации, в частности, алгоритме *K*-средних. Алгоритм *K*-средних прост в реализации и эффективен для больших наборов данных с точки зрения времени выполнения (по сравнению с другими алгоритмами). Однако алгоритм обладает рядом недостатков: во-первых, число кластеров должно быть определено заранее; во-вторых, результат зависит от выбора исходных центров кластеров, а их оптимальный выбор неизвестен; в-третьих, не гарантируется достижение глобального минимума целевой функции.

Для преодоления последних двух недостатков алгоритма *K*-средних и усовершенствования самого процесса кластеризации исследователи составляют гибридные алгоритмы: совместно с *K*-средними можно использовать метаэвристические алгоритмы.

Целью работы является разработка гибридного алгоритма формирования базы правил нечёткого классификатора на основе метода *K*-средних и метаэвристического алгоритма «китов».

Задача кластеризации. Алгоритм *K*-средних

Кластеризация (кластерный анализ) – это задача группировки множества объектов на подмножества (кластеры) таким образом, чтобы объекты из одного кластера были более похожи по заданной метрике друг на друга. Пусть X – множество объектов, Y – множество идентификаторов (меток) кластеров. На множестве X задана функция расстояния между объектами $\rho(\mathbf{x}, \mathbf{x}')$. Необходимо разбить множество X на подмножества (кластеры), т.е. каждому объекту $\mathbf{x}_i \in X$ сопоставить метку $y_i \in Y$ таким образом, чтобы объекты внутри каждого кластера были близки относительно метрики ρ, а объекты из разных кластеров значительно различались [2]. Кластеризация отличается от классификации тем, что метки y_i не задаются изначально.

Решение задачи кластеризации объективно неоднозначно: во-первых, не существует однозначного критерия качества кластеризации, во-вторых, результат кластеризации существенно зависит от метрики ρ, и в-третьих, число кластеров заранее неизвестно и выбирается по субъективным критериям [3].

Алгоритмы кластеризации классифицируются по трём основным категориям: иерархические, нечёткие и секционированные (чёткие) [4]. Иерархические алгоритмы представляют кластеры в виде древовидной структуры. При нечётком подходе каждый элемент набора данных принадлежит всем кластерам с некоторой степенью принадлежности; он удобен, если кластеры перекрываются друг с другом, но даёт некачественные результаты, если кластеры имеют различную дисперсию по различным размерностям элементов.

Алгоритм *K*-средних (*K*-means) является самым распространённым алгоритмом чёткой кластеризации [5]. Это алгоритм, который группирует объекты на основе значений их признаков в K непересекаю-

щихся кластеров, представленных в алгоритме их центроидами (точками в пространстве признаков, вблизи которых группируются экземпляры данных). Действие алгоритма таково, что он стремится минимизировать суммарную дисперсию точек кластеров относительно центров этих кластеров:

$$\sum_{k=1}^{K}\sum_{\mathbf{x}_i\in C_k}\|\mathbf{x}_i-\mathbf{C}_k\|^2\to\min,\ i=\overline{1,N},\ k=\overline{1,K},\quad (1)$$

где $\mathbf{x}_i=(x_{i1},...,x_{id})$ – i-й экземпляр данных таблицы наблюдения, представленный точкой в d-размерном пространстве признаков, $\mathbf{C}_k$ – координаты центроида k-го кластера, $\|\bullet\|$ – метрика, по которой определяется расстояние между экземпляром и центроидом (обычно в роли метрики выступает Евклидово расстояние), N – количество экземпляров в наборе данных, K – количество кластеров.

Основная идея алгоритма заключается в том, что на каждой итерации экземпляры разбиваются на кластеры в соответствии с тем, какой из новых центроидов оказался ближе по выбранной метрике. Затем заново вычисляется центроид для каждого кластера, состав которого определился на предыдущем шаге. Координаты центроида каждого кластера рассчитываются следующим образом:

$$\mathbf{C}_k=\sum_{\mathbf{x}_i\in C_k}\mathbf{x}_i/|C_k|,\quad (2)$$

где $|C_k|$ – объём k-го кластера. Алгоритм завершается, когда на текущей итерации не происходит изменения расстояния.

Эффективность работы алгоритма K-средних может быть улучшена за счет гибридизации с метаэвристиками, основное назначение которых – решение задач оптимизации [6]. В рамках данной работы исследуется применение метаэвристического алгоритма «китов» (Whale Optimization Algorithm, или WOA) совместно с алгоритмом K-средних.

Метаэвристический алгоритм «китов»

WOA – роевой алгоритм, основанный на поведении горбатых китов. Здесь используются два поведенческих механизма, которые обновляют вектор решения на каждой итерации: сужающегося окружения добычи и построения спиралевидной пузырьковой сети для атаки на добычу. Первый механизм представляет собой выполнение этапа диверсификации, второй – этапа интенсификации, в котором решения стремятся к оптимуму, двигаясь по спиралевидной траектории [7].

Последовательность выполнения WOA состоит из следующих шагов. Вначале случайным образом инициализируется популяция особей, т.е. решений, и определяется наилучшее из них. Затем на каждой итерации все имеющиеся решения корректируются согласно поведенческим механизмам: на начальных итерациях используется преимущественно первый механизм, на последних – второй; в конце итерации определяется лучшее решение. Результат работы метаэвристики – наилучшее решение, найденное за всё время работы алгоритма, и соответствующее ему значение целевой функции.

Применение метаэвристик для кластеризации с помощью *K*-средних

Для того чтобы использовать метаэвристику для кластеризации совместно с K-средними, после выполнения одной итерации метаэвристики необходимо выполнить корректировку всех решений в соответствии с (2). Целевой функцией для метаэвристики, показывающей работоспособность алгоритма кластеризации, будет выступать дисперсия (1).

Также необходимо определиться с тем, как особь популяции (поисковый агент) будет представлять решение задачи кластеризации. Первый способ называется «ассоциацией объект-кластер» [8]. Каждое решение кластеризации показано через матрицу $N\times K$, элементы которой содержат целочисленные или бинарные значения; что представляют целочисленные значения, зависит от метаэвристики и от критерия оптимизации. Такой подход использовался при кластеризации, основанной на эволюционных алгоритмах (в частности, генетическом). Однако он приводит к двум серьёзным недостаткам: высокой стоимости хранения и вычислений (особенно для кластеризации больших наборов данных) и избыточности информации, которые несёт в себе решение.

Второй способ является самым распространённым и лучше первого в отношении вычислительной сложности: решение представлено в виде вещественного вектора, который содержит координаты центроидов кластеров [9]. Решение $\mathbf{P}_j$, $j=\overline{1,J}$, где J – размер популяции, представляется как

$$\mathbf{P}_j=\left(\mathbf{C}_1^j,...,\mathbf{C}_K^j\right),\ \mathbf{C}_k^j=\left(C_{k1}^j,...,C_{kd}^j\right),\quad (3)$$

где C_{kl}^j – l-я координата k-го центроида в d-мерном пространстве признаков ($l=\overline{1,d}$) в составе j-го вектора позиции особи.

Третий способ также использует центроиды. Однако здесь решение задачи кластеризации состоит из всех агентов (а не из одного агента), и поэтому размер популяции агентов равен количеству кластеров [10]. Данный способ требует меньшего объема памяти и меньшего времени вычислений по сравнению с первым. Его потенциальная проблема заключается в том, что он не может в полной мере использовать преимущества, предоставляемые вторым способом, а именно – параллельный поиск решения несколькими агентами. Поэтому в данной работе в разработанном гибридном алгоритме кластеризации было принято представление решения в виде (3).

В оригинальном алгоритме K-средних не допускается работа с пустыми кластерами [11]. Для того чтобы работа алгоритма кластеризации, использующего метаэвристику, не прерывалась, можно применить различные способы искусственного определения кластера; например, можно заново сгенерировать случайным образом соответствующий центроид или заменить его на точку в пространстве признаков, соответствующую экземпляру данных.

Алгоритм формирования базы правил

Первый недостаток алгоритма *K*-средних, заключающийся в том, что число кластеров должно быть определено заранее, не позволяет ему в общем случае разбить данные на оптимальное количество кластеров: перед применением алгоритма *K*-средних параметр *K* произвольно задаётся экспериментатором. Здесь под оптимальным подразумевается количество кластеров, равное количеству классов, на которые в действительности делятся экземпляры данных; в случае когда *K* оптимально, сформированные кластеры соответствуют классам. Недостаток приобретает большую важность в реальных случаях применения алгоритмов кластеризации – когда требуется разбить на кластеры экземпляры данных, классы которых неизвестны (неизвестно их количество). В таких случаях не всегда ясно, какое значение *K* необходимо выбрать, чтобы затем получившиеся кластеры действительно позволили бы адекватно классифицировать объекты с высокой точностью.

Решение задачи формирования базы правил нечёткого классфикатора сводится к решению задачи построения (суб)оптимального количества кластеров. Для того чтобы это сделать с помощью алгоритма *K*-средних, предлагается следующий алгоритм.

Шаг 1. Выполняется предобработка данных: значения признаков экземпляров данных нормируются.

Шаг 2. Задаётся параметр $K = 2$ – количество кластеров, на которые необходимо разбить данные.

Шаг 3. Выполняется алгоритм кластеризации данных (в данной работе им выступает алгоритм, разработанный на основе алгоритмов *K*-средних и WOA).

Шаг 4. На основе данных о сформированных кластерах (таких, как состав кластеров, координаты центроидов кластеров) определяется значение критерия валидности для оценки качества проведённой кластеризации.

Шаг 5. После расчёта критерия проверяется, равно ли *K* максимальному значению. Максимальное значение параметра *K* выбирают, исходя из характеристик классифицируемого набора данных. Если да, идёт переход на шаг 6. В ином случае старое значение *K* увеличивается на 1, после чего идёт переход на шаг 3.

Шаг 6. После того как была проведена оценка качества кластеризации при максимальном *K*, сравниваются значения критерия, полученные при различных значениях параметра *K*. Наилучшее значение использованного критерия качества кластеризации соответствует тому значению параметра *K*, которое показывает оптимальное по данному критерию количество кластеров.

Шаг 7. Формируется база нечётких правил в количестве, равном оптимальному количеству построенных кластеров, найденному на предыдущем шаге. Каждое правило соответствует одному из кластеров.

Критерии качества кластеризации

Вышеуказанный алгоритм использует расчёт значения индекса (или критерия) кластерной валидности (или индекса правильности) на основе результатов кластеризации. Индекс является оценкой качества кластеризации, проведённой при заданном *K*: индексы используются не только для осуществления процесса кластеризации, но и для определения оптимального количества кластеров по результатам проведённой кластеризации; дисперсия (1) также является одним из индексов валидности.

Поскольку не существует наилучшего критерия качества кластеризации, построение оптимального количества кластеров и как следствие работоспособность всего алгоритма формирования базы правил будут зависеть от использованного критерия. Следовательно, перед реализацией и проверкой работы алгоритма формирования базы правил для нечёткого классификатора необходимо определиться с тем, какой индекс (или индексы) будет в нём использоваться. Для этого потребуется экспериментальным путём проверить работоспособность алгоритма построения оптимального количества кластеров отдельно для каждого из этих критериев.

Критерии валидности разделяются на внешние и внутренние. Внешние индексы для оценки результата используют предварительную информацию о наборе данных: о количестве классов, о том, к какому классу экземпляры данных в действительности относятся; они используются для выбора наилучших результатов кластеризации для конкретного набора данных [12]. Внутренние используют только информацию о центроидах и составе найденных кластеров; на её основе индексы определяют компактность кластеров и разделимость кластеров друг от друга [13].

В данной работе реализованы только внутренние критерии кластерной валидности. Далее приводятся наиболее распространённые внутренние критерии качества кластеризации, использующиеся для определения оптимального количества кластеров. Эти критерии были проверены и в рамках данной работы.

Индекс Дэвиса–Болдуина (DB) представляет собой меру среднего сходства каждого сформированного кластера с кластером, наиболее близким к данному [14]. Разделимость между кластерами характеризуется значением знаменателя – разницей между центрами кластеров, а компактность кластеров относительно друг друга характеризуется значением числителя – отклонением экземпляров данных от центра кластеров. Кластеры должны быть как можно более удалены друг от друга, при этом отклонение экземпляров от центров кластеров должно оставаться минимальным. Величина индекса неотрицательна. Минимальное значение индекса DB соответствует оптимальному количеству кластеров.

Индекс Данна (Dunn) также работает на предположении, что кластеры компактны и хорошо разделены, если расстояние между кластерами большое, а диаметр кластеров мал [15]. Величина индекса неотрицательна. Максимальное значение индекса указывает на оптимальное количество кластеров. Однако он относительно сложен для вычисления и

чувствителен к присутствию шумовых значений в наборах данных: на значение индекса сильно влияют выбросы значений, поскольку они могут увеличить значения диаметров кластера [16].

Индекс «силуэта» (Silhouette) предполагает расчёт для каждого экземпляра данных меры того, насколько данный экземпляр схож с другими экземплярами своего кластера [17]. В отношении каждого экземпляра значение «силуэта» принимает значение из диапазона [–1; 1]. Высокое значение указывает на то, что экземпляр хорошо соответствует своему собственному кластеру и плохо соответствует соседним кластерам. Усреднённое по всем экземплярам значение индекса будет являться мерой того, насколько плотно сгруппированы экземпляры данных.

В индексе Calinski–Harabasz (CH) величина компактности основана на расстоянии от точек кластера до их центроидов, а величина разделимости – на расстоянии от центроидов кластеров до глобального центроида. Максимальное значение индекса указывает на оптимальное количество кластеров [17].

Эксперимент

Разработанные гибридный алгоритм кластеризации данных и алгоритм нахождения оптимального количества кластеров с помощью этого гибридного алгоритма были реализованы на языке программирования C#.

Эксперимент состоит из двух частей. Первая часть эксперимента проводилась для тестирования работы непосредственно алгоритма кластеризации на основе WOA в сравнении с алгоритмом роящихся частиц (PSO) (так же совмещённого с алгоритмом *K*-средних) и с оригинальным алгоритмом *K*-средних. Тестирование проводилось на 15 помеченных наборах данных из репозитория KEEL. Размер популяции для WOA и PSO был равен 50, количество итераций – 200. Все алгоритмы запускались на каждом наборе данных по 50 раз. Количество кластеров задавалось равным числу классов в соответствующем наборе данных.

Также была проверена значимость теста знаковых рангов Уилкоксона для оценки статистической значимости результатов работы алгоритмов. Значимость теста проводилась на нормированных значениях результатов работы алгоритмов. Алгоритмы были настроены так, что в случае обнаружения пустого кластера алгоритм *K*-средних прекращал свою работу, а метаэвристики WOA и PSO заменяли соответствующий центроид на точку, координаты которой соответствуют случайно выбранному экземпляру данных.

Усреднённые значения результатов работы алгоритмов кластеризации приведены в табл. 1 (полужирным шрифтом выделены наименьшие значения). Сформулирована нулевая гипотеза: медиана разностей между сравниваемыми выборками равна нулю. На уровне значимости 0,05 критерий знаковых рангов Уилкоксона для связных выборок указывает на значимое отличие между значениями медиан дисперсии алгоритмов кластеризации WOA и *K*-средних (p-value < 0,001) и WOA и PSO (p-value = 0,026), следовательно, нулевая гипотеза отклоняется.

Таблица 1

Значения дисперсии алгоритмов кластеризации

Набор данных	WOA	*K*-средних	PSO
Iris	**97,363**	111,547	97,367
Wine	22796,088	23512,313	**22545,198**
Glass	**217,647**	264,204	229,942
Contraceptive	**7799,101**	11141,315	7805,321
Vowel	**1877,590**	4455,837	1970,139
Hepatitis	**5462,169**	6055,845	5471,713
Balance	**1423,851**	1425,811	1426,093
Heart	13303,286	14052,665	**13208,503**
Cleveland	**13903,251**	14812,181	13904,009
Pima	**54690,055**	73193,074	55700,059
Ecoil	**9266,317**	9567,224	10227,896
Bupa	**10165,352**	10981,388	10204,181
Ionosphere	**796,093**	796,560	796,357
Newthyroid	**1983,277**	2086,499	1986,451
Wisconsin	**2984,068**	2987,137	2984,171
Vehicle	**51565,498**	58058,472	53179,808

Вторая часть эксперимента посвящена проведению тестирования работы алгоритма построения оптимального количества кластеров, формируемых алгоритмом кластеризации на основе WOA и *K*-средних, и определению наиболее подходящего индекса валидности. Здесь были использованы те же наборы. Размер популяции для WOA был равен 50, количество итераций – 200. Алгоритм запускался на каждом наборе данных по 30 раз. По результатам каждого запуска осуществлялся расчёт значений критериев кластерной валидности. Полученные значения соответствующих индексов усреднялись по количеству запусков; решение о том, сколько же кластеров является оптимальным по этим критериям, принималось на основе средних значений этих критериев. Максимальное проверяемое количество кластеров равнялось удвоенному действительному количеству кластеров: например, для набора Iris с тремя классами максимальное количество проверяемых кластеров равнялось 6.

Также была проверена значимость теста знаковых рангов Уилкоксона для оценки статистической значимости результатов работы алгоритмов построения оптимального количества кластеров с различными критериями.

Определённое с помощью рассмотренного критерия качество кластеризации оптимальное количество кластеров, построенных с помощью гибридного алгоритма кластеризации – WOA и *K*-средних, приведено в табл. 2; результаты проверки статистической значимости по критерию Уилкоксона (уровень значимости 0,05) – в табл. 3. Значимое отличие между действительным и предсказанным количеством кластеров показал индекс CH. Для трех остальных индексов отличие незначимо. Лучший результат показал индекс Dunn.

Таблица 2

Количество кластеров

Набор данных	Кол-во кластеров действительное	Количество кластеров предсказанное			
		Silhouette	CH	DB	Dunn
Iris	3	2	2	2	2
Pima	2	3	**2**	3	**2**
Wine	3	**3**	**3**	**3**	5
Contraceptive	3	2	2	2	2
Glass	7	2	2	2	2
Ecoli	8	2	2	2	2
Vehicle	4	2	2	2	2
Ionosphere	2	**2**	**2**	**2**	**2**
Balance	3	6	2	6	6
Newthyroid	3	2	**3**	2	**3**
Hepatitis	2	**2**	**2**	**2**	**2**
Wisconsin	2	**2**	**2**	**2**	**2**
Heart	2	**2**	**2**	**2**	**2**
Bupa	2	**2**	**2**	**2**	**2**
Vowel	11	2	2	2	2

Таблица 3

Тест Уилкоксона. Значения *p*-value

	Количество кластеров предсказанное			
	Silhouette	CH	DB	Dunn
Кол-во кластеров действительное	0,094	0,017	0,094	0,182

Заключение

Усреднённые результаты кластеризации свидетельствуют о том, что алгоритм кластеризации на основе WOA превосходит алгоритмы PSO и *K*-средних (кроме набора данных Heart и Wine). Алгоритм *K*-средних по полученным результатам значительно уступает метаэвристикам. Это связано с одним из недостатков данного алгоритма: результат зависит от выбора исходных центров кластеров, что часто ведёт к сходимости к локальному оптимуму. Результаты теста Уилкоксона подтверждают превосходство алгоритма WOA, следовательно, разработанный алгоритм может использоваться для кластеризации и разработки алгоритма формирования базы правил нечёткого классификатора.

Результаты нахождения оптимального количества кластеров с помощью алгоритма кластеризации WOA-*K*-средние показывают, что применённые критерии качества кластеризации обеспечивают формирование действительного количества кластеров только в том случае, если это действительное количество не более трёх (например, для Wisconsin, Heart и Bupa). В остальных случаях, особенно при высоком действительном количестве кластеров, индексы позволяют сформировать количество кластеров, которое далеко от действительного. Яркими примерами являются проверки на наборах данных Glass, Ecoli, Vowel: согласно большинству критериев, для этих наборов два кластера являются оптимальным количеством, тогда как в действительности в них определено 7, 8 и 11 классов соответственно.

Проверка работоспособности гибридного алгоритма кластеризации WOA-*K*-средние и алгоритма формирования оптимального количества кластеров с использованием критерия Dunn показывает, что они могут быть применены для реализации предлагаемого алгоритма генерации базы нечётких правил. В дальнейшем предполагается проверка работоспособности алгоритма формирования базы правил с учётом этого критерия при построении нечёткого классификатора на реальных наборах данных.

Литература

1. Kaja N. An intelligent intrusion detection system / N. Kaja, A. Shaout, D. Ma // Applied Intelligence. – 2019. – No. 49. – P. 3235–3247.

2. Мех М.А. Сравнительный анализ применения методов дифференциальной эволюции для оптимизации параметров нечетких классификаторов / М.А. Мех, И.А. Ходашинский // Изв. Рос. академии наук. Теория и системы управления. – 2017. – № 4. – С. 65–75.

3. Abraham A. Swarm intelligence algorithms for data clustering / A. Abraham, S. Das, S. Roy // Soft Computing for Knowledge Discovery and Data Mining, Part IV. – 2007. – P. 279–313.

4. Shafiq A. Analysis of particle swarm optimization based hierarchical data clustering approaches / A. Shafiq, D. Gillian, U.R. Saeed // Swarm and Evolutionary Computation. – 2015. – P. 1–16.

5. Xu R. Clustering / R. Xu, D.C. Wunsch // New Jersey. – Hoboken: John Wiley & Sons, Inc., 2009. – 357 p.

6. Elephant search algorithm applied to data clustering / S. Deb, Z. Tian, S. Fong, R. Wong, R. Millham, K.K.L. Wong // Soft Computing. – 2018. – No. 22. – P. 6035–6046.

7. Mirjalili S. The Whale Optimization Algorithm / S. Mirjalili, A. Lewis // Advances in Engineering Software. – 2016. – No. 95. – P. 51–67.

8. Das S. Automatic clustering using an improved differential evolution algorithm / S. Das, A. Abraham, A. Konar // IEEE Transactions on Systems, Man, and Cybernetics. – Part A: Systems and Humans. – 2008. – Vol. 38, No. 1. – P. 218–237.

9. Chen C.Y. Particle swarm optimization algorithm and its application to clustering analysis / C.Y. Chen, Y. Fun // Proceedings of 17th conference on electrical power distribution networks (EPDC). – 2012. – P. 789–794.

10. Merwe D.W. Data clustering using particle swarm optimization / D.W. Merwe, A.P. Engelbrecht // 2003 Congress on evolutionary computation (CEC 2003). – 2003. – Vol. 1. – P. 215–220.

11. MacQueen J. Some methods for classification and analysis of multivariate observations // Proceedings of the fifth Berkeley symposium on mathematics statistics and probability. – 1967. – Vol. 1. – P. 281–296.

12. Alok A.K. Development of An External Cluster Validity Index using Probabilistic Approach and Min-max Distance / A.K. Alok, S. Saha, A. Ekbal // International Journal of Computer Information Systems and Industrial Management Applications. – 2014. – Vol. 6. – P. 494–504.

13. NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set / M. Charrad, N. Ghazzali, V. Boiteau, A. Niknafs // Journal of Statistical Software. – 2014. – Vol. 61, No. 6. – P. 1–36.

14. Starczewski A. A new validity index for crisp clusters // Pattern analysis and applications. – 2017. – No. 20. – P. 687–700.

15. Mittal M. Validation of k-means and Threshold based Clustering Method / M. Mittal, R.K. Sharma, V.P. Singh //

International Journal of Advancements in Technology. – 2014. – Vol. 5, No. 2. – P. 153–160.

16. Muca M. A Proposed Algorithm for Determining The Optimal Number of Clusters / M. Muca, G. Kutrolli, M. Kutrolli // European Scientific Journal. – 2015. – Vol. 11, No. 36. – P. 112–120.

17. An extensive comparative study of cluster validity indices / O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J.M. Perez, I. Perona // Pattern recognition. – 2013. – No. 46. – P. 243–256.

Корышев Николай Павлович
Студент каф. безопасности информационных систем (БИС) ТУСУРа
Ленина пр-т, 40, г. Томск, Россия, 634050
Тел.: +7-923-522-99-02
Эл. почта: koryshev1997@gmail.com

Ходашинский Илья Александрович
Д-р техн. наук, профессор каф. комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС) Томского государственного ун-та систем управления и радиоэлектроники (ТУСУР)
Ленина пр-т, 40, г. Томск, Россия, 634050
ORCID: https://orcid.org/0000-0002-9355-7638
Тел.: +7 (382-2) 70-15-29
Эл. почта: hodashn@gmail.com

Koryshev N.P., Hodashinsky I.A.

Algorithm to forming a rule base for a fuzzy classifier designed on the basis of the *K*-means clustering algorithm and the whale optimization algorithm

The article presents a description of the algorithm for generating fuzzy rules for a fuzzy classifier using data clustering, metaheuristic, and the clustering quality index, as well as the results of performance testing on real data sets.

Keywords: fuzzy classifier, clustering, *K*-means, Whale Optimization Algorithm, clustering quality indices.
doi: 10.21293/1818-0442-2021-24-1-42-47

References

1. Kaja N., Shaout A., Ma D. An intelligent intrusion detection system. *Applied Intelligence*, 2019, no. 49, pp. 3235–3247.

2. Mekh M.A. Comparative analysis of differential evolution methods to optimize parameters of fuzzy classifiers / M.A. Mekh, I.A. Hodashinsky // Journal of Computer and Systems Sciences International, 2017, vol. 56, no. 4, pp. 616–626.

3. Abraham A., Das S., Roy S. Swarm intelligence algorithms for data clustering. *Soft Computing for Knowledge Discovery and Data Mining, Part IV*, 2007, pp. 279–313.

4. Shafiq A., Gillian D., Saeed U.R. Analysis of particle swarm optimization based hierarchical data clustering approaches. *Swarm and Evolutionary Computation*, 2015, pp. 1–16.

5. Xu R., Wunsch D.C. Clustering. John Wiley & Sons, Inc, Hoboken, New Jersey, 2009, 357 p.

6. Deb S., Tian Z., Fong S., Wong R., Millham R., Wong K.K.L. Elephant search algorithm applied to data clustering. *Soft Computing*, 2018, no. 22, pp. 6035–6046.

7. Mirjalili S., Lewis A. The Whale Optimization Algorithm. *Advances in Engineering Software*, 2016, no. 95, pp. 51–67.

8. Das S., Abraham A., Konar A. Automatic clustering using an improved differential evolution algorithm. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 2008, vol. 38, no. 1, pp. 218–237.

9. Chen C.Y., Fun Y. Particle swarm optimization algorithm and its application to clustering analysis. Proceedings of 17th conference on electrical power distribution networks (EPDC), 2012, pp. 789–794.

10. Merwe D.W., Engelbrecht A.P. Data clustering using particle swarm optimization. 2003 Congress on evolutionary computation (CEC 2003), 2003, vol. 1, pp. 215–220.

11. MacQueen J. Some methods for classification and analysis of multivariate observations. Proceedings of the fifth Berkeley symposium on mathematics statistics and probability, 1967, vol. 1, pp. 281–296.

12. Alok A.K., Saha S., Ekbal A. Development of An External Cluster Validity Index using Probabilistic Approach and Min-max Distance. *International Journal of Computer Information Systems and Industrial Management Applications*, 2014, vol. 6, pp. 494–504.

13. Charrad M., Ghazzali N., Boiteau V., Niknafs A. NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set. *Journal of Statistical Software*., 2014, vol. 61, no. 6, pp. 1–36.

14. Starczewski A. A new validity index for crisp clusters. *Pattern analysis and applications*, 2017, no. 20, pp. 687–700.

15. Mittal M., Sharma R.K., Singh V.P. Validation of k-means and Threshold based Clustering Method. *International Journal of Advancements in Technology*, 2014, vol. 5, no. 2, pp. 153–160.

16. Muca M., Kutrolli G., Kutrolli M. A Proposed Algorithm for Determining The Optimal Number of Clusters. *European Scientific Journal*, 2015, vol. 11, no. 36, pp. 112–120.

17. Arbelaitz O., Gurrutxaga I., Muguerza J., Perez J.M., Perona I. An extensive comparative study of cluster validity indices. *Pattern recognition*, 2013, no. 46, pp. 243–256.

Nikolay P. Koryshev
Student, Department of Information Systems Security TUSUR
40, Lenin pr., Tomsk, Russia, 634050
Phone: +7-923-522-99-02
Email: koryshev1997@gmail.com

Ilya A. Hodashinsky
Doctor of Science in Engineering, Professor, Department of Complex Information Security of Computer Systems, Tomsk State University of Control Systems and Radioelectronics (TUSUR)
40, Lenin pr., Tomsk, Russia, 634050
ORCID: https://orcid.org/0000-0002-9355-7638
Phone: +7 (382-2) 70-15-29
Email: hodashn@gmail.com