УДК 004.021

М.Б. Бардамова, А.Г. Буймов, В.Ф. Тарасенко

Способы адаптации алгоритма прыгающих лягушек к бинарному пространству поиска при решении задачи отбора признаков

Отбор признаков является важном этапом построения любого классификатора. Для проведения отбора часто используются бинарные версии метаэвристических алгоритмов оптимизации. Однако многие метаэвристики изначально создаются для работы в непрерывном пространстве поиска, поэтому их необходимо специально адаптировать к бинарному пространству. В работе предложены пятнадцать способов бинаризации алгоритма прыгающих лягушек на основе трех методов: модифицированных алгебраических операций, операции слияния и функций трансформации. Эффективность бинарного алгоритма прыгающих лягушек проверена в задаче отбора признаков для нечеткого классификатора на наборах данных из репозитория KEEL. Результаты показывают, что все описанные способы бинаризации позволяют сокращать признаки, повышая при этом общую точность классификации.

Ключевые слова: нечеткий классификатор, алгоритм прыгающих лягушек, отбор признаков, бинаризация.
doi: 10.21293/1818-0442-2020-23-4-57-62

Операция отбора признаков в машинном обучении является эффективным инструментом для уменьшения сложности модели. Удаление избыточных признаков помогает уменьшить переобучение, а исключение шумовых атрибутов может позволить улучшить качество прогноза. Кроме того, выделение наиболее информативных переменных позволяет получить новые знания о решаемой задаче, поэтому отбор с применением интеллектуальных алгоритмов часто рассматривается как исследовательский инструмент.

Существуют две основные стратегии отбора признаков: фильтрация [1] и обертка [2]. Фильтры оценивают взаимосвязь между признаками и выходной переменной с помощью информационных критериев, например взаимной информации или коэффициента корреляции. Такие алгоритмы используются на этапе подготовки данных. С одной стороны, фильтры являются стабильным и универсальным средством отбора, так как не зависят от особенностей классификатора, но по этой же причине обладают довольно низкой эффективностью.

Обертки, напротив, функционируют в совокупности с решающим алгоритмом. Они осуществляют подбор наиболее информативных подмножеств признаков на основе качества получаемых моделей. Это позволяет им находить наборы признаков, наиболее важных для конкретного классификатора. В качестве оберток часто используются метаэвристики, в том числе и при построении нечетких систем [3–5]. Однако этот класс алгоритмов зачастую разрабатывается для оптимизации в непрерывном пространстве. В данной работе речь пойдет о способах адаптации алгоритма прыгающих лягушек к бинарному пространству с целью отбора признаков в нечетких классификаторах.

Алгоритм прыгающих лягушек (АПЛ)

Популяция входных векторов в данной метаэвристике имитирует социальное поведение лягушек, осуществляющих поиск пропитания [6, 7]. Популяция разбивается на более мелкие группы, в каждой из которых есть свой локальный лидер – лягушка, сумевшая отыскать наиболее удачное место с пищей. Остальные лягушки групп перемещаются в направлении к своему лидеру. Параллельно происходит обмен информацией между популяцией путем перетасовки групп, выделение среди лидеров самого успешного и постепенное движение всех лягушек к глобальному лидеру.

Метаэвристика имеет небольшое количество параметров. Требуется задать количество групп G, число лягушек в данной группе F, константу для обновления входных векторов c, количество итераций для глобального и локального поиска T_{gl} и T_{lc} соответственно.

Последовательность действий в непрерывном АПЛ следующая. На вход алгоритму поступает популяция векторов $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. Размер популяции устанавливается как $N = F \times G$. Для всех векторов оценивается значение целевой функции $\mathrm{fit}(\mathbf{x}_i)$, $i \in [1, N]$. В качестве целевой функции может выступать, например, точность классификации. В таком случае задача метаэвристики – найти вектор, обеспечивающий максимальное значение точности.

Далее происходит глобальный поиск, в котором популяция сортируется по убыванию целевой функции и разбивается на G групп по F векторов следующим образом: $\mathbf{x}_1$ попадет в первую подгруппу, $\mathbf{x}_2$ – во вторую, вектор $\mathbf{x}_G$ окажется в группе с номером G, $\mathbf{x}_{G+1}$ будет направлен в первую группу и т.д. Внутри каждой группы происходит локальный поиск, заключающийся в итерационном повторении трех этапов. На первом этапе генерируется временный вектор **New**:

$$\mathbf{New} = r \times c \times \left(\mathbf{best}(t) - \mathbf{worst}(t)\right) + \mathbf{worst}(t), \quad (1)$$

где r – случайное число в промежутке от 0 до 1, $\mathbf{best}(t)$ и $\mathbf{worst}(t)$ – векторы с лучшей и худшей фитнесс-функцией в группе на локальной итерации t, $t \in [1, T_{\mathrm{lc}}]$. Если выполняется условие $\mathrm{fit}(\mathbf{New}) \geq$ $\geq \mathrm{fit}(\mathbf{worst}(t))$, то временный вектор записывается в $\mathbf{worst}(t)$ и итерация локального поиска завершается.

В противном случае осуществляется переход ко второму этапу.

На втором этапе вектор **New** генерируется заново, но вместо **best**(t) используется глобально лучший вектор $\mathbf{x}_1$. Если и в этом случае условие fit(**New**) ≥ ≥ fit(**worst**(t)) не выполняется, то на третьем этапе на месте **worst**(t) создается вектор случайным образом.

Когда во всех группах заданное количество локальных итераций T_{lc} будет выполнено, популяция объединяется и весь процесс повторяется заново с сортировки. Алгоритм останавливается при достижении T_{gl}, на выход подается алгоритм с лучшей целевой функцией.

Бинаризация алгоритма прыгающих лягушек

Варианты бинаризации АПЛ в литературе обычно сводятся к его совмещению с наиболее известными метаэвристиками – алгоритмом роящихся частиц и генетическим алгоритмом [8, 9]. Например, в [8] вместо применения оператора (1) для худшей лягушки рассчитывается непрерывный вектор скорости. Формула для расчета является довольно громоздкой и оперирует большим количеством параметров. Далее вектор скорости переводится в бинарный эквивалент с помощью порога и заменяет худшую лягушку в случае улучшения целевой функции. В противном случае к этой же лягушке поэтапно применяются операции мутации и кроссовера. Ввиду объемности вычислений алгоритм теряет свои достоинства – быструю скорость работы и вычислительную простоту.

Мы предлагаем более простые способы бинаризации АПЛ, позволяющие сохранить его достоинства. Все методы принимают на вход популяцию из бинарных векторов. В задаче отбора признаков размерность каждого вектора совпадает с количеством признаков в таблице наблюдения; элемент вектора, соответствующий единице, указывает на наличие признака в обучении, ноль – на отсутствие.

Модифицированные алгебраические операции

Идея метода заключается в том, что для перехода алгоритма в бинарную версию достаточно поставить в соответствие исходным математическим операциям используемых формул их логические аналоги [10, 11]. Например, вместо сложения использовать операцию дизъюнкции (∨), а вместо умножения – конъюнкцию (∧). Таким образом, (1) можно изменить следующим образом:

$$\mathbf{New} = \mathbf{r} \wedge \big(\mathbf{best}(t) \oplus \mathbf{worst}(t)\big) \vee \mathbf{worst}(t), \quad (2)$$

где $\mathbf{r}$ – случайный бинарный вектор, символ ⊕ обозначает операцию строгой дизъюнкции, являющейся заменой вычитанию.

Метод слияния

При использовании данного метода (1) заменяется операцией слияния двух векторов – худшей лягушки в группе и локального или глобального лидера в зависимости от этапа локального поиска. Операция слияния заключается в следующем. Элементы двух входных векторов поэлементно сравниваются. Если два элемента, стоящие на одной и той же позиции, одинаковы, то в результирующем векторе в этой позиции будет записано значение этих элементов. Если элементы различаются, то осуществляется генерация случайного числа. Если оно меньше или равно 0,5, то в соответствующую позицию нового вектора записывается элемент из худшей лягушки. В противном случае на этом месте будет выставлен элемент из лучшего вектора [12].

Комбинация модифицированных арифметических операций и слияния

При совмещении двух ранее описанных способов бинаризации сначала генерируется вектор **New** на основе (2), затем **New** и **worst**(t) подаются на вход операции слияния.

Функции трансформации

Некоторые метаэвристики не способны оперировать бинарными векторами напрямую. Часто для перехода из непрерывного пространства в бинарное в таком случае используют функции трансформации, которые осуществляют отображение вещественного числа на промежуток от нуля до единицы. Если вычисленное значение функции больше случайного числа, то соответствующий элемент вектора либо принимает значение единицы, либо заменяется на противоположное [13, 14].

В данной работе были опробованы три функции трансформации. Первая функция принадлежит к семейству S-образных и определяется по следующей формуле:

$$F_1(z) = 1\big/\left(1+e^{-z}\right), \quad (3)$$

где z – некоторое входное вещественное число. Вторая рассчитывается с помощью гиперболического тангенса и относится к числу V-образных функций:

$$F_2(z) = \left|\tanh(z)\right|. \quad (4)$$

К тому же семейству относится третья функция:

$$F_3(z) = \left| z\big/\sqrt{1+z^2} \right|. \quad (5)$$

Для обновления элементов в бинарном векторе признаков будем использовать два правила. В первом явно задается бинарное значение элемента:

$$\text{Если } \operatorname{rand}(0,1) < F(z), \text{ то } x^d = 1, \text{ иначе } x^d = 0, \quad (6)$$

где F(z) – одна из трех функций трансформации, d – номер позиции элемента в векторе $\mathbf{x}$. Во втором трансформационном правиле элемент либо меняется на противоположный, либо остается прежним:

$$\text{Если } \operatorname{rand}(0,1) < F(z), \text{ то } x^d = x^d \oplus 1. \quad (7)$$

Для использования описанного метода в SFLA вместо (1) вводится расчет вектора скорости следующим образом:

$$\mathbf{V} = \big(\mathbf{x}_1 - \mathbf{worst}(t)\big) \times \mathbf{r}_1 + \big(\mathbf{best}(t) - \mathbf{worst}(t)\big) \times \mathbf{r}_2, \quad (8)$$

где $\mathbf{r}_1$ и $\mathbf{r}_2$ – непрерывные векторы, заполненные случайными вещественными значениями в диапазоне от 0 до 1. Затем вектор $\mathbf{V}$ подается на вход одной из трех функций трансформации.

Результат применения функции трансформации и одного из двух правил обновления элементов записывается в вектор **worst**(t) в том случае, если происходит улучшение фитнес-функции. В противном случае на месте **worst**(t) генерируется вектор случайным образом.

Комбинация функций трансформации и слияния

К предыдущему методу добавляется второй этап локального поиска. На нем получившийся ранее вектор проходит через операцию слияния с глобальным лидером $\mathbf{x}_1$. Если и он не улучшает худший вектор, то проводится случайная генерация.

Таблица 1

Параметры наборов данных

Данные	Аббр.	Признаки	Экземпляры	Классы
magic	mgc	10	19020	2
wine	wn	13	178	3
heart	hrt	13	270	2
cleveland	clv	13	297	5
vowel	vwl	13	990	11
penbased	pnb	16	10992	10
vehicle	vhc	18	846	4
hepatitis	hpt	19	80	2
segment	sgm	19	2310	7
ring	rng	20	7400	2
twonorm	twn	20	7400	2
thyroid	thr	21	7200	3
wdbc	wdbc	30	569	2
ionosphere	ion	33	351	2
dermatology	drm	34	358	6
satimage	stm	36	6435	7
spectfheart	spc	44	267	2
spambase	spm	57	4597	2
sonar	snr	60	208	2
optdigits	optd	64	5620	10
coil2000	cl2000	85	9822	2

Эксперимент

В эксперименте были использованы наборы данных из открытого репозитория KEEL [15]. Описание наборов приведено в табл. 1. В качестве решающего алгоритма использовался нечеткий классификатор с треугольными термами. Структура классификатора генерировалась алгоритмом на основе экстремальных значений классов [16].

Эксперимент был проведен по схеме десятикратной кроссвалидации: каждый набор был разбит на десять обучающих и тестовых выборок в пропорции 9:1. Параметры алгоритма прыгающих лягушек следующие: количество групп равняется 4, в каждой из них по 10 векторов; локальных итераций 5, глобальных итераций 30. В табл. 2 представлены перечень использованных в АПЛ способов бинаризации и их краткое обозначение.

Таблица 2

Методы бинаризации АПЛ

Обозначение	Метод
AO	Модифицированные алгебр. операции
m	Слияние
AOm	МАО + слияние
F_{11}	ФТ* (3) + правило (6)
F_{21}	ФТ (4) + правило (6)
F_{31}	ФТ (5) + правило (6)
F_{12}	ФТ (3) + правило (7)
F_{22}	ФТ (4) + правило (7)
F_{32}	ФТ (5) + правило (7)
F_{11}m	ФТ (3) + правило (6) + слияние
F_{21}m	ФТ (4) + правило (6) + слияние
F_{31}m	ФТ (5) + правило (6) + слияние
F_{12}m	ФТ (3) + правило (7) + слияние
F_{22}m	ФТ (4) + правило (7) + слияние
F_{32}m	ФТ (5) + правило (7) + слияние

*ФТ – функция трансформации

В табл. 3 приведена результирующая точность при построении классификатора на отобранных подмножествах признаков, усредненная по десяти тестовым выборкам.

Таблица 3

Точность классификации при использовании различных версий бинарного АПЛ для отбора признаков

Данные	ЭК	AO	m	AOm	F_{11}	F_{21}	F_{31}	F_{12}	F_{22}	F_{32}	F_{11}m	F_{21}m	F_{31}m	F_{12}m	F_{22}m	F_{32}m
mgc	71,9	71,8	72,1	73,1	72,7	73,1	72,8	71,9	72,1	72,3	72,4	73,1	73,1	72,1	72,2	71,8
wn	88,1	92,1	93,2	93,8	92,7	93,8	93,2	92,1	92,6	93,2	93,2	92,6	92,6	94,4	92,1	92,1
hrt	71,5	66,3	66,7	66,7	66,3	66,3	66,7	65,6	65,6	66,3	66,3	65,6	65,6	65,6	66,7	65,9
clv	45,2	53,5	54,5	54,5	54,5	54,5	54,5	54,5	54,5	54,2	54,5	54,5	54,5	54,5	53,9	54,2
vwl	43,8	47,8	47,9	47,4	47,5	45,9	47,5	46,7	47,9	47,6	47,9	47,6	47,1	47,6	47,6	47,6
pnb	56,7	57,7	57,7	56,9	57,5	56,8	56,7	56,8	57,7	57,7	57,3	57,2	56,7	57,0	57,6	57,7
vhc	30,4	38,8	46,3	42,2	41,1	43,0	40,0	41,3	45,3	45,3	42,8	44,2	44,7	42,4	45,4	45,9
hpt	83,4	80,3	78,1	77,9	78,1	79,0	79,4	81,0	79,0	80,8	76,1	79,4	79,0	78,1	79,2	82,5
sgm	80,5	84,9	85,5	84,7	85,1	85,1	84,7	84,7	86,0	85,7	85,4	85,6	85,4	85,5	85,6	85,9
rng	64,4	55,9	65,5	65,3	60,5	64,7	65,6	55,4	65,6	65,5	63,2	65,6	65,6	60,1	65,6	65,6
twn	96,1	96,1	96,2	96,1	96,1	96,1	96,1	96,1	96,2	96,1	96,0	96,1	96,1	96,1	96,2	96,2
thr	7,4	88,8	96,8	96,7	96,6	96,7	96,7	93,8	96,8	96,8	96,8	96,8	96,8	96,8	96,8	96,8
wdbc	92,8	94,5	95,6	94,4	95,1	95,3	95,1	94,0	96,0	95,6	95,3	95,1	95,1	95,1	95,1	95,4
ion	79,8	83,2	83,2	83,8	83,5	82,3	82,9	82,9	83,2	83,2	83,2	82,9	83,5	83,2	83,2	83,2
drm	80,7	83,0	87,2	80,7	83,3	78,2	83,2	83,6	88,2	88,6	85,8	83,8	81,5	81,6	88,6	88,0
stm	60,3	63,6	69,1	66,4	65,2	64,7	65,3	63,2	68,9	69,1	66,4	68,6	68,8	64,6	69,3	68,5
spc	78,6	78,6	79,7	80,9	79,0	76,7	80,1	80,2	77,5	79,0	77,1	79,4	80,1	78,6	79,1	78,6
spm	39,4	56,1	76,9	74,7	66,5	72,4	72,1	59,2	77,0	77,4	72,0	78,2	78,4	62,0	75,1	74,2
snr	62,5	54,4	57,1	58,1	52,9	57,2	51,5	53,4	52,9	57,2	54,4	53,5	60,5	54,8	54,8	53,3
optd	13,5	28,1	33,3	27,0	28,1	27,1	26,7	26,9	33,1	33,1	30,1	29,1	28,6	29,5	32,9	32,4
cl2000	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0	94,0
Ср.	**63,9**	**70,0**	**73,2**	**72,2**	**71,3**	**71,6**	**71,7**	**70,3**	**72,9**	**73,3**	**71,9**	**72,5**	**72,7**	**71,1**	**72,9**	**72,9**

Для сравнения приведены результаты построения нечеткого классификатора алгоритмом экстремальных значений классов до отбора признаков (столбец ЭК).

В последней строке таблицы содержатся средние значения точности классификации по всем наборам данных. По сравнению с точностью, полученной без отбора признаков, все методы бинаризации позволили улучшить среднее качество классификации. Лучшие результаты были получены бинарным АПЛ с помощью функции трансформации (5) с правилом (7), а также при использовании метода слияния в отдельности.

В табл. 4 представлено усредненное по десяти выборкам количество признаков, оставшихся в обучении после отбора признаков. На разных выборках может получиться различное количество признаков, поэтому числа в таблице могут принимать дробные значения. В столбце ЭК приведено исходное количество признаков в наборе данных до проведения отбора. В последней строке табл. 3 подсчитаны средние значения количества отобранных с помощью АПЛ признаков.

Лучшие способности к отбору признаков алгоритма прыгающих лягушек позволили показать следующие методы: V-образные функции трансформации с правилом (6) и операцией слияния, а также модифицированные алгебраические операции в комбинации с операцией слияния.

На рис. 1 представлен фронт Парето между средними рангами, вычисленными согласно критерию Фридмана, по ошибке классификации и количеству отобранных признаков. В сносках к точкам указаны следующие значения: название метода, средний ранг метода по числу признаков, средний ранг метода по ошибке классификации. Во фронт попали три точки; две из них принадлежат V-образным функциям трансформации с правилом отбора, которое явно задает равенство элемента входного вектора нулю или единице, и операцией слияния. Третья точка принадлежит бинарной версии АПЛ, которая использует функцию слияния без каких-либо других методов.

Таблица 4

Точность классификации при использовании различных версий бинарного АПЛ для отбора признаков

Данные	ЭК	AO	m	AOm	F_{11}	F_{21}	F_{31}	F_{12}	F_{22}	F_{32}	F_{11}m	F_{21}m	F_{31}m	F_{12}m	F_{22}m	F_{32}m
mgc	10	5,7	5,1	1,4	3,3	1,5	1,9	5,1	5,2	4,6	3,4	1,5	1,5	5,2	4,6	4,1
wn	13	7,7	6,0	5,4	5,8	6,4	5,2	7,1	6,3	5,3	6,0	5,3	5,1	6,3	5,6	5,3
hrt	13	6,2	6,3	6,1	6,7	6,5	5,5	6,6	6,8	7,2	6,2	5,9	6,6	6,6	7,3	6,6
clv	13	5,3	3,5	3,1	3,0	3,4	2,8	3,4	3,9	4,3	3,3	3,2	2,5	3,9	3,5	4,3
vwl	13	8,2	7,3	6,8	7,5	6,8	6,6	7,8	7,5	8,1	7,6	7,1	7,2	7,6	7,7	7,9
pnb	16	14,3	14,3	15,6	13,5	15,6	16,0	14,4	14,2	14,3	14,0	13,8	13,6	14,0	14,2	14,3
vhc	18	9,6	6,3	5,2	7,2	5,7	6,2	8,5	7,3	7,2	6,5	6,0	5,6	7,6	7,1	6,1
hpt	19	9,1	9,0	7,5	8,8	7,4	7,7	8,6	8,5	9,3	8,1	7,7	8,3	8,5	8,9	9,2
sgm	19	11,0	9,2	7,9	9,6	8,6	9,0	10,6	9,4	9,4	9,0	8,0	7,6	9,9	8,8	8,3
rng	20	5,2	2,9	2,6	3,5	2,3	2,7	6,0	2,8	2,9	3,6	2,8	2,8	3,8	2,8	2,8
twn	20	18,8	18,7	20,0	20,0	20,0	20,0	20,0	18,7	18,9	19,4	20,0	20,0	19,5	18,7	18,7
thr	21	10,6	7,4	6,0	8,1	7,1	6,0	8,8	7,6	7,7	8,1	6,6	6,7	9,2	7,7	7,8
wdbc	30	13,5	11,1	7,6	10,9	9,8	8,6	13,8	11,3	12,4	11,3	8,4	6,5	14,2	11,3	12,0
ion	33	20,8	19,1	17,6	19,0	17,8	17,6	20,6	19,2	18,9	19,4	16,9	17,1	19,8	18,8	19,7
drm	34	24,8	21,5	16,9	19,5	17,2	16,7	20,6	21,1	19,4	19,2	16,3	14,0	21,7	20,1	18,2
stm	36	16,7	6,0	7,7	12,1	10,5	10,3	15,0	5,9	6,3	10,4	5,1	5,1	13,8	6,3	6,3
spc	44	31,9	24,0	24,3	23,6	22,5	22,3	28,6	24,1	23,8	22,4	20,4	20,4	28,1	24,7	23,2
spm	57	20,5	14,0	8,9	19,5	15,5	15,8	20,2	14,6	14,8	20,4	8,0	7,6	20,5	15,2	15,5
snr	60	44,7	28,4	16,9	33,8	20,6	22,3	37,4	31,5	28,7	27,3	17,1	12,7	33,9	30,8	32,2
optd	64	35,3	32,1	25,1	33,1	29,0	26,0	35,4	32,7	30,9	30,6	24,1	21,3	36,6	31,9	31,7
cl2000	85	76,0	74,7	85,0	74,1	82,1	85,0	68,7	82,2	85,0	78,2	81,6	72,9	74,7	82,3	82,1
Среднее	**30,4**	**18,9**	**15,6**	**14,2**	**16,3**	**15,1**	**15,0**	**17,5**	**16,2**	**16,2**	**15,9**	**13,6**	**12,6**	**17,4**	**16,1**	**16,0**

Заключение

В данной работе была проведена проверка эффективности пятнадцати методов бинаризации алгоритма прыгающих лягушек. Все методы являются вычислительно простыми, поэтому скорость алгоритма довольно высокая. Самую высокую скорость, 5,3 с, алгоритм прыгающих лягушек продемонстрировал при использовании функции трансформации (4) с правилом (7). Дольше всего метаэвристика работала при комбинации функции трансформации (4) с правилом (6) и операцией слияния; этот метод в среднем отрабатывал за 9 с.

Несмотря на то, что разные способы бинаризации лидировали по двум измеряемым характеристикам, можно отметить следующее. Метод алгебраических операций продемонстрировал худшие результаты, поэтому использовать его как единственный метод бинаризации не стоит.

Операция слияния попала во фронт Парето как метод с наименьшим средним рангом по ошибке классификации. Кроме того, добавление слияния к другим способам бинаризации практически во всех случаях позволило улучшить точность классификации этих методов при уменьшении количества отобранных признаков.

S-образная функция трансформации показала более скромные результаты, чем функции из V-образного семейства. Первое трансформационное правило позволило отобрать меньшее количество признаков, но и этот результат удалось сократить при добавлении операции слияния. Второе правило помогло получить большую среднюю точность.

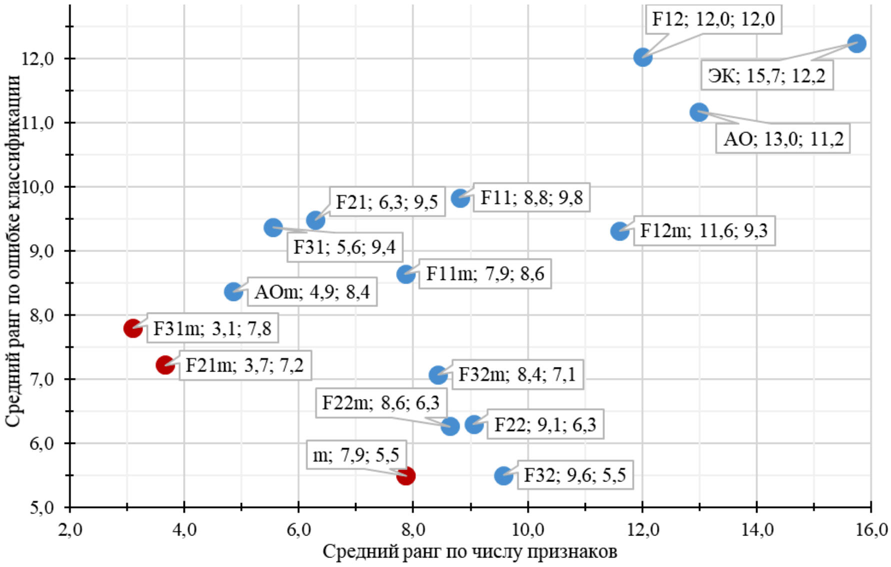

Рис. 1. Парето фронт по рангам критерия Фридмана по соотношению «ошибка классификации»–«число признаков»

Выделить один метод, который можно рекомендовать как универсальный для любых данных и задач, невозможно. При необходимости получения наибольшей точности классификации стоит воспользоваться методом слияния. Алгоритм прыгающих лягушек, использующий в качестве способа бинаризации V-образную функцию, вычисляемую по выражению (5), правило обновления векторов (6) и операцию слияния, может позволить получить наименьшее подмножество информативных признаков.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-37-90064.

Литература

1. Vergara J.R. A review of feature selection methods based on mutual information / J.R. Vergara, P.A. Estevez // Neural Computing and Applications. – 2014. – Vol. 24. – P. 175–186.
2. Kohavi R. Wrappers for feature subset selection / R. Kohavi, G. John // Artificial Intelligence. – 1997. – Vol. 97, No. 1. – P. 273–324.
3. Hodashinsky I.A. Feature selection for fuzzy classifier using the spider monkey algorithm / I.A. Hodashinsky, M.M. Nemirovich-Danchenko, S.S. Samsonov // Business Informatics. – 2019. – Vol. 13, No. 2. – P. 29–42.
4. Метаэвристические методы отбора информативных классифицирующих признаков / А.Е. Анфилофьев, М.Б. Бардамова, И.А. Ходашинский, К.С. Сарин // Информационные и математические технологии в науке и управлении. – 2017. – № 2(6). – С. 11–20.
5. Improving the OVO performance in Fuzzy Rule-Based Classification Systems by the genetic learning of the granularity level / P. Villar, A. Fernández, R. Montes, A.M. Sánchez, F. Herrera // Proc. of 2015 IEEE International Conference on Fuzzy Systems. – Istanbul, 2015. – P. 1–7.
6. Eusuff M. Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization / M. Eusuff, K. Lansey, F. Pasha // Engineering Optimization. – 2006. – Vol. 38, No. 2. – P. 129–154.
7. Elbeltagi E. A modified shuffled frog-leaping optimization algorithm: applications to project management / E. Elbeltagi, T. Hegazy, D. Grierson // Structure and Infrastructure Engineering. – 2007. – Vol. 3, No. 1. – P. 53–60.
8. Vakil Baghmisheh M.T. A discrete shuffled frog optimization algorithm / M.T. Vakil Baghmisheh, K. Madani, A. Navarbaf // Artificial Intelligence Review. – 2011. – Vol. 36. – P. 267–284.
9. A novel hybrid optimization method of shuffled frog leaping algorithm and particle swarm optimization / M.J. Lin, F. Luo, Y.G. Xu, L. Luo // Advanced Materials Research. – 2013. – Vol. 717. – P. 433–438.
10. Jia D. Binary artificial bee colony optimization using bitwise operation / D. Jia, X. Duan, M.K. Khan // Computers and Industrial Engineering. – 2014. – Vol. 76. – P. 360–365.
11. Afshinmanesh F. A novel binary particle swarm optimization method using artificial immune system / F. Afshinmanesh, A. Marandi, A. Rahimi-Kian // Proc. of the International Conference on Computer as a Tool. – IEEE, 2005. – Vol. 1. – P. 217–220.
12. Feature selection based on swallow swarm optimization for fuzzy classification / I. Hodashinsky, K. Sarin, A. Shelupanov, A. Slezkin // Symmetry. – 2019. – Vol. 11, No. 11. – P. 1423.
13. Mirjalili S. S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization / S. Mirjalili, A. Lewis // Swarm and Evolutionary Computation. – 2013. – Vol. 9. – P. 1–14.
14. Putting continuous metaheuristics to work in binary search spaces / B. Crawford, R. Soto, G. Astorga, J. García, C. Castro, F. Paredes // Complexity. – 2017. – Vol. 2017. – P. 1–19.
15. Knowledge Extraction Based on Evolutionary Learning [Электронный ресурс]. – Режим доступа: http://www.keel.es, свободный (дата обращения: 24.08.2020).
16. Алгоритмы структурной идентификации компактных и точных нечетких систем / И.А. Ходашинский, И.В. Горбунов, К.С. Сарин, С.Р. Субханкулова // Информационные и математических технологии в науке и управлении. – 2016. – № 1. – С. 82–93.

Бардамова Марина Борисовна
Преп. каф. комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС)
Томского государственного ун-та
систем управления и радиоэлектроники (ТУСУР)
Ленина пр-т, д. 40, г. Томск, Россия, 634050
ORCID: 0000-0002-0567-590X
Тел.: +7-903-951-61-13
Эл. почта: 722bmb@gmail.com

Буймов Аркадий Георгиевич
Д-р техн. наук, профессор каф. экономики ТУСУРа
Ленина пр-т, д. 40, г. Томск, Россия, 634050
ORCID: 0000-0003-1335-6922
Эл. почта: arkadii.g.buimov@tusur.ru

Тарасенко Владимир Феликсович
Д-р техн. наук, профессор каф. государственного
и муниципального управления НИ ТГУ
Тел.: 8 (382-2) 70-15-91
Эл. почта: vtara54@mail.ru

Bardamova M.B., Buymov A.G., Tarasenko V.F.

Methods for adapting the leaping frog algorithm to the binary search space when solving the feature selection problem

The feature selection is an important step in constructing any classifier. Binary versions of metaheuristic optimization algorithms are often used for selection. However, many metaheuristics are originally created to work in the continuous search space, so they need to be specially adapted to the binary space. In this paper, the authors propose fifteen ways to binarize the Shuffled frog leaping algorithm based on the following methods: modified algebraic operations, merge operation, and transformation functions. The efficiency of the binary algorithm was tested in the problem of feature selection for fuzzy classifiers on data sets from the KEEL repository. The results show that all the described methods of binarization allow reducing the features, while increasing the overall accuracy of classification.

Keywords: fuzzy classifier, shuffled frog leaping algorithm, fuzzy selection, binarization.
doi: 10.21293/1818-0442-2020-23-4-57-62

References

1. Vergara J.R., Estevez P.A. A review of feature selection methods based on mutual information. *Neural Computing and Applications*, 2014, vol. 24, pp. 175–186.

2. Kohavi R., John G. Wrappers for feature subset selection. Artificial Intelligence, 1997, vol. 97, no. 1, pp. 273–324.

3. Hodashinsky I.A., Nemirovich-Danchenko M.M., Samsonov S.S. Feature selection for fuzzy classifier using the spider monkey algorithm. *Business Informatics*, 2019, vol. 13, no. 2, pp. 29–42.

4. Anfilofiev A.E., Bardamova M.B., Hodashinsky I.A., Sarin K.S. [Metaheuristic methods for selecting informative classifying features]. *Information and Mathematical Technologies in Science and Management*, 2017, no. 2(6), pp. 11–20 (in Russ.).

5. Villar P., Fernández A., Montes R., Sánchez A. M., Herrera F. Improving the OVO performance in Fuzzy Rule-Based Classification Systems by the genetic learning of the granularity level. Proc. of 2015 IEEE International Conference on Fuzzy Systems. Istanbul, IEEE, 2015, pp. 1–7.

6. Eusuff M., Lansey K., Pasha F. Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization. *Engineering Optimization*, 2006, vol. 38, no. 2, pp. 129–154.

7. Elbeltagi E., Hegazy T., Grierson D. A modified shuffled frog-leaping optimization algorithm: applications to project management. *Structure and Infrastructure Engineering*, 2007, vol. 3, no. 1, pp. 53–60.

8. Vakil Baghmisheh M.T., Madani K., Navarbaf A. A discrete shuffled frog optimization algorithm. *Artificial Intelligence Review*, 2011, vol. 36, pp. 267–284.

9. Lin M.J., Luo F., Xu Y.G., Luo L. A novel hybrid optimization method of shuffled frog leaping algorithm and particle swarm optimization. *Advanced Materials Research*, 2013, vol. 717, pp. 433–438.

10. Jia D., Duan X., Khan M. K. Binary artificial bee colony optimization using bitwise operation. *Computers and Industrial Engineering*, 2014, vol. 76, pp. 360–365.

11. Afshinmanesh F., Marandi A., Rahimi-Kian A. A novel binary particle swarm optimization method using artificial immune system. Proc. of the International Conference on Computer as a Tool. IEEE, 2005. Vol. 1, pp. 217–220.

12. Hodashinsky I., Sarin K., Shelupanov A., Slezkin A. Feature selection based on swallow swarm optimization for fuzzy classification. *Symmetry*, 2019, vol. 11, no. 11, pp. 1423.

13. Mirjalili S., Lewis A. S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization. *Swarm and Evolutionary Computation*, 2013, vol. 9, pp. 1–14.

14. Crawford B., Soto R., Astorga G., García J., Castro C., Paredes F. Putting continuous metaheuristics to work in binary search spaces. *Complexity*, 2017, vol. 2017, pp. 1–19.

15. Knowledge Extraction Based on Evolutionary Learning. Available at: http://www.keel.es/ (Accessed: August 24, 2020).

16. Hodashinsky I.A., Gorbunov I.V., Sarin K.S., Subhankulova S.R. [Algorithms for structural identification of compact and precise fuzzy systems]. *Information and Mathematical Technologies in Science and Management*, 2016, no. 1, pp. 82–93 (in Russ.).

Marina B. Bardamova
Lecturer, Department of Complex Information Security
of Computer Systems, Tomsk State University
of Control Systems and Radioelectronics (TUSUR)
40, Lenin pr., Tomsk, Russia, 634050
ORCID: 0000-0002-0567-590X
Phone: +7-903-951-61-13
Email: 722bmb@gmail.com

Arkadiy G. Buymov
Doctor of Engineering Sciences, Professor,
Department of Economics, TUSUR
40, Lenin pr., Tomsk, Russia, 634050
ORCID: 0000-0003-1335-6922
Email: arkadii.g.buimov@tusur.ru

Vladimir F. Tarasenko
Dr. Tech. Sciences, Professor of the Department state
and municipal administration, NR TSU
36, Lenin pr., Tomsk, Russia, 634050
Phone: 8 3822 70 15 91
Email: vtara54@mail.ru