ПРОГРАММИРОВАНИЕ, 2019, № 5, с. 3–9

АНАЛИЗ ДАННЫХ

УДК 004.85:004.056

ОТБОР КЛАССИФИЦИРУЮЩИХ ПРИЗНАКОВ: СРАВНИТЕЛЬНЫЙ АНАЛИЗ БИНАРНЫХ МЕТАЭВРИСТИК И ПОПУЛЯЦИОННОГО АЛГОРИТМА С АДАПТИВНОЙ ПАМЯТЬЮ

© 2019 г. И. А. Ходашинский[a],*, К. С. Сарин[a],**

[a] *Томский государственный университет систем управления и радиоэлектроники*
634050 Томск, пр. Ленина, 40, Россия

**E-mail: hodashn@rambler.ru*

***E-mail: sks@security.tomsk.ru*

Поступила в редакцию 15.05.2018 г.
После доработки 31.10.2018 г.
Принята к публикации 31.10.2018 г.

Рассматривается известная NP-трудная задача отбора признаков. Для решения задачи предложен популяционный алгоритм, основанный на сочетании случайного и эвристического поиска. Решение представляется в виде бинарного вектора, размерность которого определена количеством признаков в наборе данных. Генерация новых решений проводится случайным образом с использованием нормального и равномерного распределения. Эвристика, лежащая в основе предлагаемого подхода, формулируется следующим образом: шанс признака попасть в следующую генерацию пропорционален частоте присутствия этого признака в предыдущих лучших решениях. Эффективность предложенного алгоритма проверена на 18 известных наборах данных. Проведено статистическое сравнение предложенного алгоритма с алгоритмами-аналогами.

DOI: 10.1134/S0132347419050030

1. ВВЕДЕНИЕ

Отбор признаков – важная задача в области интеллектуального анализа данных и машинного обучения, направленная на уменьшения размерности входных данных и повышения эффективности алгоритмов классификации, кластеризации, регрессии и прогнозирования временных рядов [1]. Результатом отбора признаков является синтетическое представление, обычно называемое вектором признаков [2].

Структурированные данные, используемые для анализа, обычно представлены множеством объектов или наблюдений, которые фиксируются в виде строк, а также рядом признаков (переменных, атрибутов или столбцов), которыми собственно и описываются объекты реального мира. Первичные данные могут включать в себя множество нерелевантных или избыточных признаков. Признак считается релевантным, если он существенно влияет на результат классификации, регрессии или прогнозирования. Признак избыточен, если он сильно коррелирует с другими признаками [3].

Принято выделять три технологии отбора признаков: фильтры (filter), обертки (wrapper) и встроенные методы (embedded methods) [4, 5]. Фильтры оценивают релевантность признаков, опираясь только на внутренние свойства данных, и не зависят от алгоритма классификации. В технологии обертки оценка подмножества отобранных признаков выполняется в процессе обучения и/или тестирования конкретного классификатора. Поиск оптимального подмножества признаков во встроенных методах выполняется в процессе построения классификатора и может рассматриваться как поиск в объединенном пространстве подмножеств признаков и параметров классификатора [5].

Алгоритмы отбора признаков могут быть разделены на пакетные методы и потоковые. В первом случае задача отбора признаков выполняется в режиме офф-лайн, когда доступны все экземпляры набора данных. Во втором случае все экземпляры набора данных заранее неизвестны, экземпляры и признаки поступают последовательно [6].

Для решения задачи отбора признаков были применены различные методы поиска, такие как полный перебор, жадные алгоритмы и случайный поиск. Однако большинство существующих методов отбора признаков подвержены попаданию в локальные оптимумы и/или имеют большие вычислительные затраты [2]. Отбор признаков является NP-трудной задачей [4], гарантировано оптимальное решение может быть найдено только

путем полного перебора. Использование метаэвристических методов позволяет получить субоптимальные решения без необходимости исследовать все пространство решения.

Целью данной работы является описание нового популяционного алгоритма с адаптивной памятью для решения задач отбора признаков в пакетном режиме и сравнение эффективности предложенного алгоритма с алгоритмами-аналогами.

2. БЛИЗКИЕ РАБОТЫ

2.1. Метаэвристики для отбора признаков

Сложность задачи отбора признаков обусловлена не только большим пространством поиска, но и проблемой взаимозависимости между признаками. Признак, который сам по себе слабо влияет на точность классификации, в совокупности с другими может существенно увеличить эту точность. Поэтому признаки не должны оцениваться по отдельности, оценка должна быть дана в целом всему подмножеству признаков [1].

Отбор признаков преследует две цели – максимизация точности классификации и минимизация количества признаков. Эти две цели являются противоречивыми, поэтому отбор признаков можно рассматривать как двухкритериальную проблему оптимизации [1], для решения которой могут быть применены метаэвристические методы, такие как эволюционные вычисления, методы роевого интеллекта и их гибриды.

В [7] рассмотрены три метаэвристические стратегии – GRASP, поиск с запретами и меметический алгоритм – для решения проблемы отбора признаков. Эти три стратегии были сопоставлены с генетическим алгоритмом и некоторыми другими методами отбора признаков, включая семейство жадных алгоритмов поиска. Авторы работы [8] для отбора признаков предлагают использовать алгоритм гармонического поиска, а выбор оптимального классификатора на отобранных подмножествах признаков проводить на основе информационного критерия Акаике. Для выбора оптимального подмножества классифицирующих признаков в режиме обертки в [9] предложены бинарные варианты алгоритма китов; здесь основные операторы непрерывного алгоритма китов заменены на бинарные, а также добавлены несколько эволюционных операторов (селекции, кроссовера и мутации). В [10] проведено исследование методов отбора признаков с использованием двух гибридных подходов, основанных не на совместном использовании алгоритмов пчелиной колонии и роящихся частиц, а также пчелиной колонии и генетического алгоритма. В обзорных работах [1] и [3] дан глубокий и всесторонний анализ применения эволюционных вычислений и методов роевого интеллекта для решения задачи отбора признаков. В [1] приведены 45 ссылок на работы, в которых генетические алгоритмы в режиме обертки применялись для отбора признаков, а также 18 работ по использованию генетического программирования для решения указанных задач; 29 и 16 ссылок сделано на работы по отбору признаков с помощью алгоритма роящихся частиц и алгоритма муравьиной колонии, соответственно; алгоритмы пчелиной колонии упомянуты 6 раз, и дифференциальной эволюции – 7 раз; менее популярны меметические алгоритмы, алгоритмы пчелиной колонии, гравитационный поиск, алгоритм искусственной иммунной системы, эволюционная стратегия.

Среди недостатков метаэвристических методов отбора признаков можно отметить их высокие вычислительные затраты, связанные с большим количеством вычислений оценок, и низкую стабильность, проявляющуюся в том, что после каждого прогона можно получить разные подмножества признаков, а это может потребовать разработки методов выбора среди отобранных подмножеств признаков [1].

2.2. Адаптивная память

Использование памяти в метаэвристике было впервые предложено Гловером в поиске с запретами, который по сути является локальным поиском, основанным на понятиях окрестности и адаптивной функции памяти, запрещающей повторный поиск ранее обнаруженных решений [11].

Адаптивная память лежит в основе любого обучения, более того, такие важные процедуры, как интенсификация и диверсификация, чаще всего реализуются на основе адаптивной памяти [12]. Триаду интенсификация–диверсификация–обучение авторы работы [12] исследуют, объединяя поиск с запретами и метод Лагранжевых релаксаций с помощью адаптивной памяти.

В [13] предложен популяционный метод адаптивного симплекса для решения задач стохастической оптимизации. Метод использует операторы отражения и сжатия классического симплексного метода Нелдера–Мида [14], а также стратегию локального поиска и механизмы обнаружения стагнации и удаления дубликатов. Введенный авторами адаптивный порог вероятности позволяет регулировать процесс сходимости алгоритма.

Согласно теореме о “бесплатных завтраках” (No-free-lunch) не существует метаэвристического алгоритма, который одинаково успешно решал все задачи оптимизации. Если конкретный метаэвристический алгоритм превосходит другие для определенного класса задач оптимизации, то нет уверенности в том, что он будет эффективен для другого класса задач оптимизации. Это побуждает исследователей предлагать новые метаэври-

Таблица 1. Описание наборов данных

Наборы данных	Число признаков	Число образцов
Breastcancer	9	699
BreastEW	30	569
CongressEW	16	435
Exactly	13	1000
Exactly2	13	1000
HeartEW	13	270
IonosphereEW	34	351
KrvskpEW	36	3196
Lymphography	18	148
M-of-n	13	1000
PenglungEW	325	73
SonarEW	60	208
SpectEW	22	267
Tic-tac-toe	9	958
Vote	16	300
WaveformEW	40	5000
WineEW	13	178
Zoo	16	101

стические алгоритмы, а также улучшать работу существующих алгоритмов [15].

3. ПОСТАНОВКА ЗАДАЧИ

Введем следующие обозначения:

$X = \{x_1, x_2, ..., x_n\}$ – множество входных признаков;

$\mathbf{S} = (s_1, s_2, \ldots, s_n)^{\mathrm{T}}$ – бинарный вектор-решение.

Переменные задачи:

$$s_i = \begin{cases} 1, & \text{если } i\text{-й признак} \\ & \text{используется классификатором.} \\ 0, & \text{в противном случае} \end{cases}$$

Точность решения *acc*, полученная классификатором на таблице наблюдений $\{(\mathbf{x}_i, c_i),\ i = 1, 2, ..., z\}$ и использующим признаки вектора $\mathbf{S}$, вычисляется следующим образом

$$acc(\mathbf{S}) = \frac{\sum_{i=1}^{z} \begin{cases} 1, & \text{если} \quad f(\mathbf{x}_i; \mathbf{S}) = c_i \\ 0, & \text{иначе} \end{cases}}{z},$$

здесь $f(\mathbf{x}_i; \mathbf{S})$ выход классификатора для экземпляра входных данных $\mathbf{x}_i$ с используемыми признаками $\mathbf{S}$.

Каждое решение оценивается в соответствии с предлагаемой целевой функцией, которая зависит от точности классификации, и количества выбранных признаков в решении:

$$F(\mathbf{S}) = \alpha(1 - acc(\mathbf{S})) + \beta(r/n),$$
$$\alpha + \beta = 1, \quad \alpha, \beta \in [0,1],$$
$$\min F(\mathbf{S}),$$

при ограничениях

$$s_i \in \{0,1\}, \quad i = 1, \ldots, n,$$

r – число признаков, используемых классификатором.

Для решения указанной задачи предлагается использовать разработанный авторами популяционный алгоритм с адаптивной памятью.

4. ПОПУЛЯЦИОННЫЙ АЛГОРИТМ С АДАПТИВНОЙ ПАМЯТЬЮ

Алгоритм основан на следующей эвристике: текущее значение элемента вектора-решения зависит от его значений в лучших решениях на предыдущих итерациях. Поскольку вектор бинарный, то для каждого i-го элемента достаточно хранить в памяти число итераций b_i, на которых i-й элемент принимал значение 1, тогда адаптивный параметр алгоритма p, вычисляемый как относительная частота появления 1, равен

$$p = b_i / t,$$

где t – число выполненных итераций.

Работа алгоритма начинается с формирования популяции – набора сгенерированных случайным или иным образом векторов $\mathbf{S}$. Число векторов в популяции – это наперед заданное целое число, называемое размером популяции. Для каждого вектора вычисляется значение целевой функции F.

На каждой итерации определяется вектор с минимальным значением F – лучшее решение на текущей итерации. Другим важным элементом алгоритма является вектор $\mathbf{B}$, в котором каждый элемент b_i хранит число появлений i-го признака в лучших решениях на предыдущих итерациях, размерность этого вектора совпадает с размерностью $\mathbf{S}$. Вектор $\mathbf{B}$, выполняющий роль адаптивной памяти, служит для реализации следующей эвристики: шанс признака попасть в следующую популяцию пропорционален частоте присутствия этого признака в предыдущих лучших решениях. Новая популяция формируется на основе указанной эвристики и случайного поиска. Нормально распределенная случайная величина $u \sim N(0, \sigma_g)$ определяет механизм сокращения или добавления признаков, а также количество сокращенных/добавленных признаков. Если значение u больше нуля, то происходит добавление новых

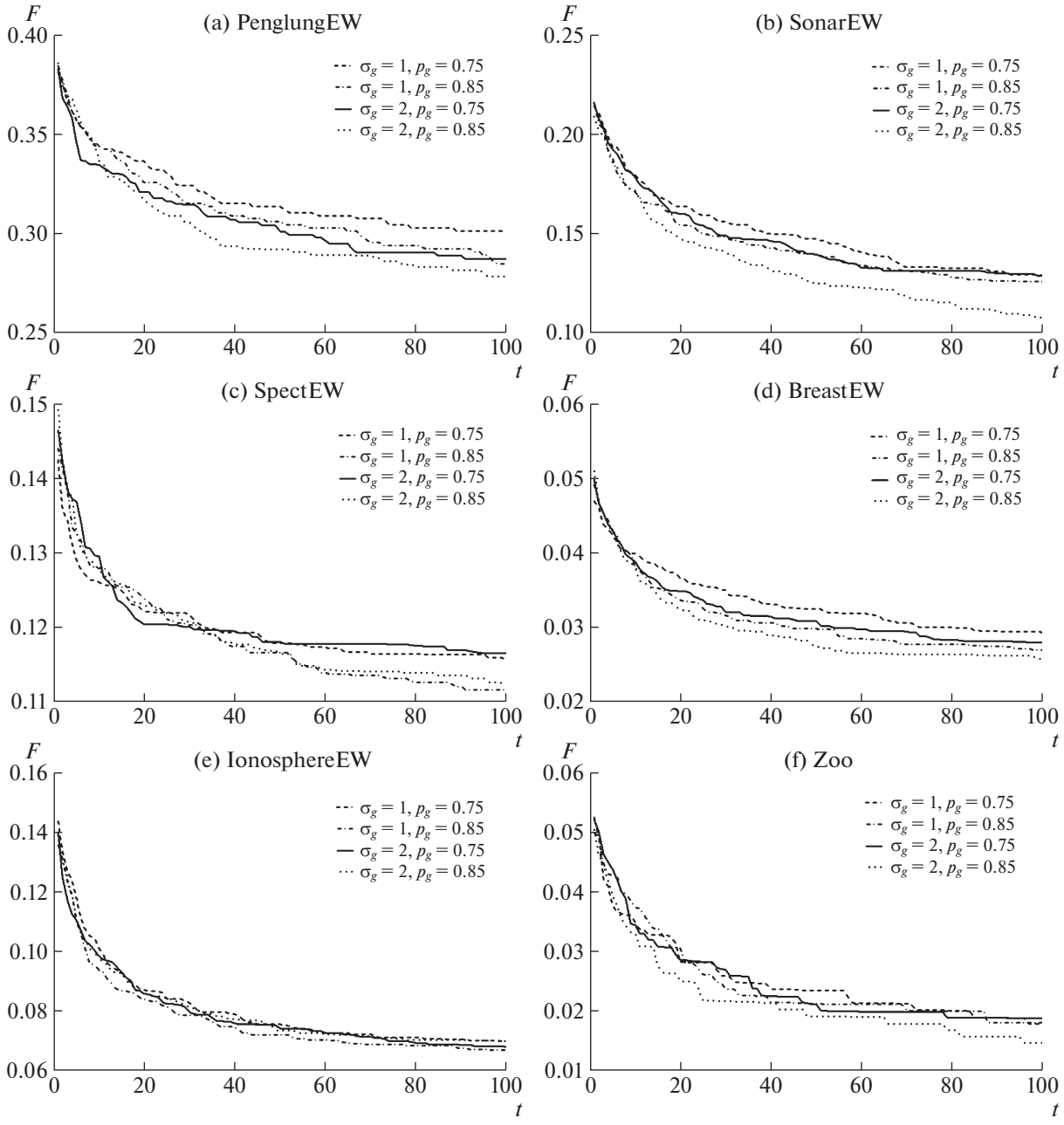

Рис. 1. Зависимость среднего значения целевой функции от числа итераций.

признаков путем увеличения количества единиц в векторе **S**, в противном случае происходит сокращение количества признаков. Количество потенциально возможных изменяемых признаков определяется размерностью вектора **S**, равной n, и числом единиц (позиций, где $s_i \neq 0$) в векторе **S**, положим равное r. Если значение u меньше нуля, то среди r элементов вектора **S**, содержащих единицы, случайным образом выбираются l признаков-кандидатов на сокращение по формуле

$$l = \text{round}(r|\text{th}(u)|).$$

Определение числа добавляемых признаков выполняется по формуле

$$l = \text{round}((n - r)|\text{th}(u)|).$$

Однако собственно изменение значения элемента s_i определяется значением относительной частоты p, новое значение s_i определяется следующим образом:

$$s_i = \begin{cases} 1, & \text{если } rand(0,1) \le p \\ 0, & \text{иначе} \end{cases}.$$

Таблица 2. Значение целевой функции

Набор данных	ALO	GA	PSO	WOA	PAM
Breastcancer	**0.021**	0.028	0.03	0.035	0.024
BreastEW	0.033	0.036	0.03	0.034	**0.021**
CongressEW	0.046	0.043	0.04	0.047	**0.023**
Exactly	0.289	0.281	0.28	**0.005**	0.036
Exactly2	0.24	0.25	0.25	0.259	**0.238**
HeartEW	**0.122**	0.138	0.15	0.193	0.142
IonosphereEW	0.108	0.125	0.14	**0.076**	0.082
KrvskpEW	0.05	0.068	0.05	**0.027**	0.034
Lymphography	0.136	0.171	0.19	0.148	**0.107**
M-of-n	0.107	0.075	0.11	**0.005**	0.009
PenglungEW	**0.139**	0.22	0.22	0.203	0.215
SonarEW	0.179	0.13	0.13	**0.079**	0.102
SpectEW	0.124	0.137	0.13	0.135	**0.123**
Tic-tac-toe	0.222	0.242	0.24	0.220	**0.08**
Vote	**0.037**	0.054	0.05	0.050	0.038
WaveformEW	0.206	0.203	0.22	0.250	**0.182**
WineEW	0.017	0.014	0.02	0.045	**0.013**
Zoo	0.073	0.082	0.1	0.023	**0.013**

Чтобы избежать ранней сходимости алгоритма, необходимо добавить ограничение на значение относительной частоты:

$$1 - p_g \le p \le p_g,$$

где p_g – пороговое значение.

Алгоритм выполняется итерационно, после выполнения заданного числа итераций производится декодирование лучшего вектора в полученное решение.

Ниже представлен псевдокод алгоритма; здесь *popul* – размер популяции, T – максимальное число итераций, p_g – пороговое значение, $\mathbf{S}^j$ – j-й вектор-решение, **Sbest** – вектор лучшего решения и *Fbest* соответствующее значение целевой функции.

Инициализация: $t = 1$, $\mathbf{S}^j = \text{rand}\{0, 1\}^n$ $j = 1, ..., popul$, $\mathbf{B} = \{0.5\}^n$;

Sbest = $\mathbf{S}^1$, *Fbest* = $F(\mathbf{S}^1)$;

Цикл по итерациям $t = 1, 2, ..., T$

Цикл по популяции $j = 1, 2, ..., popul$

Если $F(\mathbf{S}^j) < Fbest$ *то* $Fbest = F(\mathbf{S}^j)$, **Sbest** = $\mathbf{S}^j$;

Присвоить r число элементов $\mathbf{S}^j$, равных 1.

$u \sim N(0, \sigma_g)$.

Если $u \ge 0$, то случайным образом среди r элементов выбрать элементы в количестве round(r|th(u)|). Каждому выбранному элементу присвоить значение 1, если rand $< b_k/t$, и 0 в противном случае, здесь k – номер элемента в векторе $\mathbf{S}^j$.

Если $u < 0$, то случайным образом выбрать среди элементов $\mathbf{S}^j$, равных 0, элементы в количестве round(($n - r$)|th(u)|). Каждому выбранному элементу присвоить значение 0, если rand $< b_k/t$, и 0 в противном случае, здесь k – номер элемента в векторе $\mathbf{S}^j$.

Конец цикла по популяции j;

B = **B** + **Sbest**;

Если $b_k/(t + 1) > p_g$, *то* $b_k = p_g(t + 1)$, $k = 1,2, ..., n$;

Если $b_k/(t + 1) < (1 - p_g)$, *то* $b_k = (1 - p_g)(t + 1)$, $k = 1, 2, ..., n$;

Конец цикла по итерациям t;

Вывод **Sbest**, *Fbest*.

5. ЭКСПЕРИМЕНТ И ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

5.1. Описание эксперимента

Проведение эксперимента и наборы данных, участвовавшие в нем, соответствовали работе [9]. В качестве классификатора для оценки работоспособности алгоритма использовался алгоритм k-ближайших соседей с $k = 5$. Описание наборов данных представлено в таблице 1. Эксперимент проводился по схеме 10-кратной кросс-валидации: 9 из 10 частей экземпляров данных использовались для обучения и валидации, а оставшаяся часть для тестирования. Согласно используемому подходу число образцов обучающих и валидационных данных одинаково, число векторов в популяции – 15, число итераций – 100.

Результаты работы популяционного алгоритма с памятью сравнились с результатами других бинарных алгоритмов, применяемыми для отбора признаков при классификации и описанных в работе [9].

5.2. Подбор рабочих параметров алгоритма

Основной фазе эксперимента предшествовала фаза эмпирической оценки параметров алгоритма. Параметру σ_g задавались значения 1 и 2; параметру p_g – 0.75 и 0.85. На рисунке 1 представлены зависимости усредненного значения целевой функции по 20 запускам алгоритма от числа итераций. Алгоритм с параметрами $\sigma_g = 2$ и $p_g = 0.85$ показал некоторое увеличение скорости к достижению оптимума. Данные значения параметров были взяты для проведения эксперимента.

5.3. Результаты эксперимента

Для оценки эффективности работы популяционного алгоритма с адаптивной памятью прово-

Таблица 3. Статистики доверительных интервалов парных разностей целевой функции

Пара	Среднее	95% доверительный интервал для разности		Значимость
		Нижняя	Верхняя	
PAM–WOA	–0.019556	–0.039032	–0.000079	0.049
PAM–ALO	–0.037056	–0.072797	–0.001314	0.043
PAM–GA	–0.045278	–0.076750	–0.013806	0.007
PAM–PSO	–0.049889	–0.082190	–0.017588	0.005

дились сравнения с результатами работы алгоритмов поведения муравьиного льва (ALO), роящихся частиц (PSO), генетического алгоритма (GA) и алгоритма поведения китов (WOA) [9]. Коэффициенты целевой функции α, β в экспериментах устанавливались в значения 0.99 и 0.01, соответственно. В таблице 2 приведены усредненные значения целевой функции *F*. Популяционный алгоритм с адаптивной памятью в таблице представлен аббревиатурой PAM.

Для оценки статистической значимости различий в значениях целевой функции классификаторов, сформированных популяционным алгоритмом с адаптивной памятью, и классификаторов-аналогов, использованы доверительные интервалы для разности средних, двухфакторный ранговый дисперсионный анализ Фридмана для связных выборок и критерий парных сравнений Уилкоксона.

Проверка гипотез с помощью доверительных интервалов основана на следующем правиле [16]:

"Если $100(1-\alpha)$-процентный доверительный интервал разности средних не содержит нуля, то различия статистически значимы ($P < \alpha$); напротив, если этот интервал содержит нуль, то различия статистически не значимы ($P > \alpha$)".

Статистики 95% доверительных интервалов разности значений целевой функции приведены в таблице 3.

Статистики теста знаковых рангов Уилкоксона для медиан разностей значений целевой функции приведены в таблице 4. Нулевая гипотеза H0 сформулирована следующим образом: медиана разностей между парами равна нулю, уровень значимости $\alpha = 0.05$.

Сравнительный анализ позволил сделать следующие выводы:

1) 95% доверительные интервалы для разности средних значений целевой функции не содержат нуля, значит с заданной уверенностью можно считать статистически значимыми различия между средними значениями целевой функции;

2) двухфакторный ранговый дисперсионный анализ Фридмана для связных выборок указывает на значимое отличие в распределениях пяти сравниваемых значениях целевой функции (p-value < 0.001);

3) критерий знаковых рангов Уилкоксона для связных выборок указывает на значимое отличие между значениями целевой функции (p-value <0.036).

6. ЗАКЛЮЧЕНИЕ

В статье описан новый популяционный алгоритм с адаптивной памятью для бинарной оптимизации с практическим применением для решения задачи отбора информативных признаков. В качестве классификатора использовался алгоритм *k*-ближайших соседей. Восемнадцать известных наборов данных из хранилища UCI использовались для оценки эффективности предложенного алгоритма. Сравнительный статистический анализ предложенного алгоритма с алгоритмами–аналогами позволил сделать следующий вывод: при отборе признаков с точки зрения выбранной целевой функции популяционный алгоритм с адаптивной памятью является предпочтительным по сравнению с алгоритмами–аналогами.

В дальнейшем предполагается исследовать эффективность популяционного алгоритма с адаптивной памятью для отбора признаков на несбалансированных наборах данных и на классификаторах других типов.

7. БЛАГОДАРНОСТИ

Работа выполнена при финансовой поддержке Министерства образования и науки РФ в рамках базовой

Таблица 4. Тест Уилкоксона

Пара	Значимость	Решение
PAM–WOA	0.035	H0 отклонить
PAM–ALO	0.012	H0 отклонить
PAM–GA	<0.001	H0 отклонить
PAM–PSO	<0.001	H0 отклонить

части государственного задания в сфере научной деятельности (проект 2.3583.2017/4.6).

СПИСОК ЛИТЕРАТУРЫ

1. *Xue B., Zhang M., Browne W. N., Yao X.* A survey on evolutionary computation approaches to feature selection // IEEE Transactions on Evolutionary Computation. 2016. V. 20. P. 606–626.
2. *Labati R.D., Genovese A., Munoz E., Piuri V., Scotti F.* Applications of Computational Intelligence in Industrial and Environmental Scenarios // Studies in Computational Intelligence. 2018. V. 756. P. 29–46.
3. *de la Iglesia B.* Evolutionary computation for feature selection in classification problems // WIREs Data Mining and Knowledge Discovery. 2013. V. 3. P. 381–407.
4. *Kohavi R., John G.H.* Wrappers for feature subset selection // Artificial Intelligence. 1997. V. 97. P. 273–324.
5. *Saeys Y., Inza I., Larranaga P.* A review of feature selection techniques in bioinformatics // Bioinformatics. 2007. V. 23. P. 2507–2517.
6. *Armanfard N., Reilly J.P., Komeili M.* Logistic Localized Modeling of the Sample Space for Feature Selection and Classification // IEEE Transactions on Neural Networks and Learning Systems. 2018. V. 29. P. 1396–1413.
7. *Yusta S. C.* Different Metaheuristic Strategies to Solve The Feature Selection Problem. Pattern Recognition Letters. 2009. V. 30. P. 525–534.
8. *Ходашинский И.А., Мех М.А.* Построение нечеткого классификатора на основе методов гармонического поиска // Программирование. 2017. № 1. С. 54–65.
9. *Mafarja M., Mirjalili S.* Whale optimization approaches for wrapper feature selection // Applied Soft Computing. 2018. V. 62. P. 441–453.
10. *Djellali H., Djebbar A., Zine N.G., Azizi N.* Hybrid Artificial Bees Colony and Particle Swarm on Feature Selection // Computational Intelligence and Its Applications. CIIA 2018. IFIP Advances in Information and Communication Technology. 2018. V. 522. P. 93–105.
11. *Glover F., Hanafi S.* Tabu search and finite convergence // Discrete Applied Mathematics. 2002. V. 119. P. 3–36.
12. *Riley R.C.L., Rego C.* Intensification, diversification, and learning via relaxation adaptive memory programming: a case study on resource constrained project scheduling // Journal of Heuristics. 2018. P. 1–15.
13. *Omran M.G.H., Clerc M.* APS 9: an improved adaptive population-based simplex method for real-world engineering optimization problems // Applied Intelligence. 2018. V. 48. P. 1596–1608.
14. *Nelder J., Mead R.* A simplex method for function minimization // Computer Journal. 1965. V. 7. P. 308–313.
15. *Saha S., Mukherjee V.* A novel chaos-integrated symbiotic organisms search algorithm for global optimization // Soft Computing. 2018. V. 22. P. 3797–3816.
16. *Гланц С.* Медико-биологическая статистика. Пер. с англ. М.: Практика, 1998. 459 с.