УДК 004.9

АЛГОРИТМ МУРАВЬИНОЙ КОЛОНИИ ПРИ РЕШЕНИИ ЗАДАЧИ КЛАССИФИКАЦИИ И ИСПОЛЬЗОВАНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ ПОДБОРА ЕГО ПАРАМЕТРОВ

магистрант В.М. БАСИНСКИЙ, ст. преп. Ю.Г. СТЕПИН
(Гродненский государственный университет имени Я. Купалы, Беларусь)

Рассматривается задача классификации, которая может быть сформулирована следующим образом. Пусть задано множество объектов X, множество допустимых ответов Y , и существует целевая функция (target function) $y^*: X \to Y$, значения которой y_i = $y^*(x_i)$ известны только на конечном подмножестве объектов$\{x_1, \dots x_l\} \in X$. Пары «объект– ответ» (x_i, y_i) называются прецедентами. Совокупность пар X^l = $(x_i, y_i)_{i=1}^{l}$ называется обучающей выборкой (training sample) [1].

Задача обучения по прецедентам заключается в том, чтобы по выборке X^l восстановить зависимость y^*, то есть построить решающую функцию (decision function) $f: X \to Y$, которая приближала бы целевую функцию $y^*(x)$, причём не только на объектах обучающей выборки, но и на всём множестве X.

Задачу классификации, т.е. задачу поиска и выделения классификационных правил, можно свести к задаче поиска путей на графе, что позволяет использовать для решения любой из алгоритмов случайного поиска на графах, например, алгоритм муравьиной колонии [2].

При решении задачи классификации при помощи муравьиного алгоритма возникает проблема оптимизации его параметров. Параметрами муравьиного алгоритма являются следующие объекты:

- Предельное число поколений муравьев в каждом муравейнике;
- Минимальное число найденных подряд одинаковых правил;
- Максимальное число непокрытых записей в наборе;
- Минимальное число записей, которые должно покрывать каждое правило;
- Весовые коэффициенты при расчете вероятностей переходов муравья;
- Пределы изменения количества феромонов – локальная характеристика вершин;
- Коэффициент кросс-валидации, определяющий величину каждого обучающего набора в виде доли от общего числа записей.
- Используемый метод пересчёта значений оценочной функции и эвристики.

С учетом содержания параметров и сложной структуры данных, становится понятным, что решение задачи поиска зависимости значений параметров от исходной выборки объектов в явном виде становится бесполезным. Однако хотелось бы получить возможность получать значения параметров достаточно близкие к оптимальным.

С решением этой проблемы могут справиться эволюционные алгоритмы. В данной работе использовался генетический алгоритм. Так как часть параметров дискретна, а часть параметров имеет непрерывный набор значений, то и использовать необходимо как операторы классического генетического алгоритма, так и непрерывного [3].

Генетический алгоритм предполагает определение правил трех типов:

1. Скрещивания,
2. Мутации,
3. Отбора.

В работе реализованы следующие варианты:

1. Арифметический и линейный методы скрещивания для непрерывных параметров, равномерное скрещивание для дискретных,
2. Без мутаций,
3. Турнирный отбор с усечением,
4. Элитизм.

Каждому из параметров нужно задать некоторые начальные значения, которые определяются эмпирически в выбранном интервале. После этого первое поколение муравьиных колоний генерируется с такими параметрами, а параметры для последующих поколений подбираются генетическим алгоритмом таким образом, чтобы с увеличением номера поколения качество классификации при тестировании увеличивалось.

На выходе алгоритма можно получить файл следующей структуры, который показан на рисунке 1.

	Euristic Type	Pheromones Type	Divide Type	Pruning Active	Max Ants Generations Number	Max Number For Convergence	Max Uncovered Cases	Min Cases Per Rule	Cross Validation Coefficient	Name	Quality
1	entropy	evaporation	byClass	FALSE	356	59	211	150	0,39	1-1	0,8346
2	density	evaporation	byClass	TRUE	328	67	214	163	0,45	1-2	0,8279
3	entropy	normalization	crossValidation	FALSE	374	45	186	120	0,51	1-3	0,8375
4	entropy	evaporation	crossValidation	FALSE	167	28	221	113	0,34	1-4	0,8215
5	density	evaporation	byClass	TRUE	271	38	194	115	0,21	1-5	0,8362
6	entropy	normalization	crossValidation	TRUE	315	95	200	149	0,31	1-6	0,8164
7	density	normalization	byClass	TRUE	342	84	189	137	0,25	1-7	0,8378
8	entropy	normalization	crossValidation	TRUE	213	31	229	124	0,29	1-8	0,8321

Рисунок 1. – Пример выходных данных о классификаторах

Каждая строка, являющаяся геномом одной колонии муравьев, представляет собой список пар «параметр – значение».

Параметр ***Euristic Type*** определяет, каким образом будет рассчитываться значение эвристической функции – по формуле 1 или 2.

Параметр ***Pheromones Type*** определяет вариант, по которому будет происходить пересчет оценки вершины – количество феромонов в ней – формулы 3 или 4.

Параметр ***Divide Type*** влияет на принцип разбиения тренировочной выборки в процессе обучения: можно использовать кросс-валидацию или обучать каждый из муравейников на определение записей только одного из имеющихся классов.

Флаг ***Pruning Active*** включает или отключает усечение правил после их построения: муравейники для которых параметр принимает значение *false* строят более подробные правила, которые соответствуют меньшему количеству исходных записей, а в другом случае – правила, которые описывают большее множество записей, но обычно обладают меньшей точностью классификации.

Параметры ***Max Ants Generations Number***, ***Max Number For Convergence***, ***Max Uncovered Cases***, ***Min Cases Per Rule*** соответствуют следующим параметрам:

- Предельное число поколений муравьев в каждом муравейнике;
- Минимальное число найденных подряд одинаковых правил;

• Максимальное число непокрытых записей в наборе;

• Минимальное число записей, которые должно покрывать каждое правило;

Name используется для идентификации муравейника и состоит из пары чисел: номер поколения – номер муравейника в этом поколении.

Quality – это и есть качество полученного муравейником набора классификационных правил, протестированного на выборке.

Параметр Cross Validation Coefficient имеет влияние только в случае, если значение параметра ***Divide Type*** установлено в *crossValidation*. Он определяет насколько сильно будет дробиться исходная тренировочная выборка при обучении.

Рассмотрим результаты работы алгоритма на примере набора данных, взятого с сайта OpenML: German Credit Data [4].

Качество алгоритма будем определять долей ошибок при тестировании полученного набора классификационных правил. Рассмотрим, как в ходе построения муравейниками правил изменяется точность классификации благодаря использованию генетического алгоритма. Отобразим на графике три величины:

• Точность, которой достиг лучший муравейник в поколении,

• Среднюю точность всех муравейников в текущем поколении,

• Среднюю точность всех муравейников, если при генерации следующего поколения использовать одну из известных идей генетического алгоритма – принцип элитизма.

По оси X откладываем номер поколения, а по оси Y – точность, полученных классификаторов

График изменения описанных величин приведен на рисунке 2.

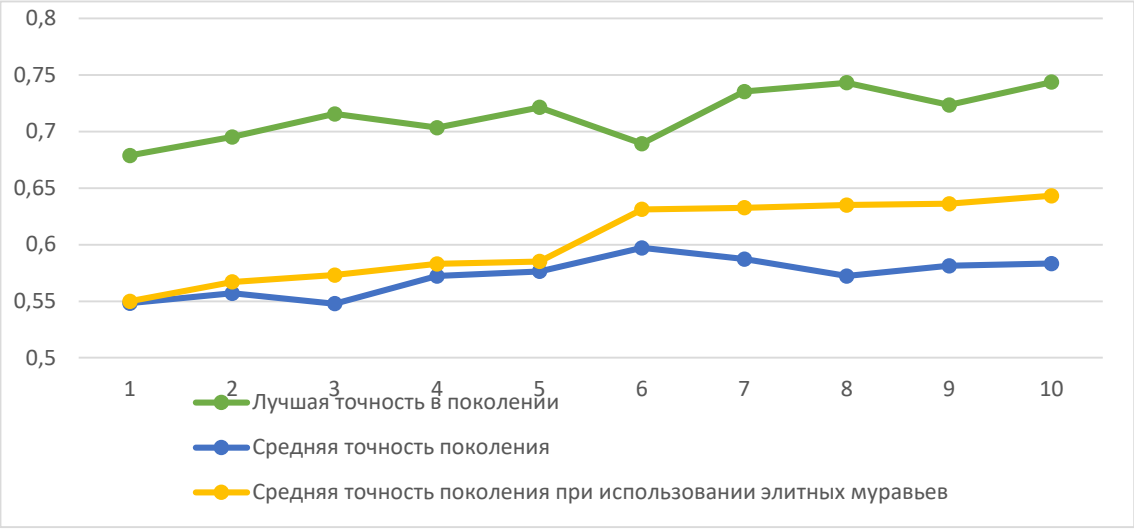

Рисунок 2. – Изменение точности классификации алгоритма в процессе работы

Как видно из графика в общем наблюдается положительная тенденция изменения точности в каждом следующем поколении, как по лучшему значению, так и в сред-

нем по поколениям. Интерес представляет дальнейшее выявление возможностей генетического алгоритма для улучшения качества классификации, так как даже использование одной из базовых идей позволило повысить среднюю точность классификации в процессе построения и эволюции муравейников.

Также одним из перспективных направлений дальнейшего исследования является анализ изменения значений каждого из параметров в процессе работы генетического алгоритма так как это позволит определить, к чему в процессе сходятся его значения. Это должно позволить более точно определять границы изменения значений параметра и осуществлять поиск оптимального значения каждого из параметров с меньшим разбросом и более направленно.

Можно утверждать, что использование генетического алгоритма для подбора и оптимизации параметров алгоритма муравьиной колонии при решении задачи классификации дало положительный эффект. Немаловажным является также и тот факт, что сам алгоритм не уступает в качестве точности предсказания самым популярным алгоритмам классификации, а некоторые даже превосходит [5].

Литература

1. Воронцов, К.В. Машинное обучение : курс лекций [Электронный ресурс] / К.В. Воронцов // Машинное обучение. – Режим доступа: https://bit.ly/1bCmE3Z. – Дата доступа: 10.05.2018.
2. Freitas, A.A. Ant Colony Algorithms for Data Classification / A.A. Freitas, R.S. Parpinelli, H.S. Lopes // Encyclopedia of Information Science and Technology. – Second Edition – Information Resources Management Association. – USA, 2008. – P. 154–159.
3. Непрерывные генетические алгоритмы – математический аппарат [Электронный ресурс] // BaseGroup Labs – Режим доступа: https://basegroup.ru/community/articles/real-coded-ga. – Дата доступа: 10.05.2018.
4. OpenML credit-g [Электронный ресурс] // OpenML – Режим доступа: https://www.openml.org/d/31. – Дата доступа: 10.05.2018.
5. Supervised Classification on credit-g [Электронный ресурс] // OpenML – Режим доступа: https://www.openml.org/t/31. – Дата доступа: 10.05.2018.