НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
июль–август 2017 Том 17 № 4 ISSN 2226-1494 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
July–August 2017 Vol. 17 No 4 ISSN 2226-1494 http://ntv.ifmo.ru/en

УДК 004.023:004.852:004.832.23

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ ВЫБОРА ПРИЗНАКОВ НА ОСНОВЕ ОЧЕРЕДИ С ПРИОРИТЕТОМ

И.Б. Сметанников[a]

[a] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация
Адрес для переписки: smeivan@mail.ru

Информация о статье
Поступила в редакцию 20.05.17, принята к печати 30.06.17
doi: 10.17586/2226-1494-2017-17-4-664-669
Язык статьи – русский

Ссылка для цитирования: Сметанников И.Б. Параллельный алгоритм выбора признаков на основе очереди с приоритетом // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17. № 4. С. 664–669. doi: 10.17586/2226-1494-2017-17-4-664-669

Аннотация

Предмет исследования. Исследованы методы и алгоритмы выбора признаков в задачах классификации, применяемые в машинном обучении. Предложен метод ускоренного выбора признаков, сводящийся к задаче оптимизации линейной комбинации (ансамбля) других алгоритмов выбора признаков. **Метод.** Суть предлагаемого алгоритма состоит в выборе признаков на основе очереди с приоритетом. Предложенное решение является развитием алгоритма выбора признаков measure linear form (MeLiF). Предложенный вариант алгоритма реализует очередь с приоритетом для эффективного распараллеливания вычислений и, по сути, является многопоточной версией алгоритма MeLiF. **Основные результаты.** Оценивание качества алгоритма и сравнение его с исходным алгоритмом проведено по критериям времени, затрачиваемого на оптимизацию, и итогового качества классификации. Исследования выполнены на 36 наборах данных ДНК-микрочипов из различных открытых баз данных. Показано, что при одинаковом качестве классификации время, затрачиваемое предложенным алгоритмом, сокращается от 4,2 до 22 раз на 24-ядерном процессоре в 50 потоках. **Практическая значимость.** Предложенный алгоритм может быть использован для выбора значимых признаков в наборах данных с большим числом признаков. Алгоритм может быть применен для предобработки данных в задачах машинного обучения и использоваться в широком спектре задач классификации на достаточно больших наборах данных.

Ключевые слова
машинное обучение, выбор признаков, объединение метрик, ранжирующие фильтры, агрегация метрик, MeLiF, параллельные вычисления

Благодарности
Работа выполнена при финансовой поддержке Правительства Российской Федерации, грант 074-U01 и РФФИ, грант 16-37-60115-мол_а_дк.

FEATURE SELECTION PARALLELIZATION BASED ON PRIORITY QUEUE

I.B. Smetannikov[a]

[a] ITMO University, Saint Petersburg, 197101, Russian Federation
Corresponding author: smeivan@mail.ru

Article info
Received 20.05.17, accepted 30.06.17
doi: 10.17586/2226-1494-2017-17-4-664-669
Article in Russian

For citation: Smetannikov I.B. Feature selection parallelization based on priority queue. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2017, vol. 17, no. 4, pp. 664–669 (in Russian). doi: 10.17586/2226-1494-2017-17-4-664-669

Abstract

Subject of Research. The paper deals with feature selection algorithms in machine learning and, particularly, in classification. A method for fast feature selection is proposed. This method combines several other feature selection methods into one linear combination (ensemble) and then optimizes their coefficients. **Method.** Proposed method is a priority queue based method for feature selection. It is an improvement of measure linear form (MeLiF) algorithm. This method uses priority queue for parallelization, and basically is a parallel version of the MeLiF algorithm. **Main Results.** Proposed and original algorithms were compared by classification quality and computation time. Comparison was performed on 36 open

DNA-microarrays. It was shown that both methods had approximately the same classification quality but computation time of the new method is 4.2 to 22 times lower on a 24-core processor with 50 threads. **Practical Relevance.** Proposed algorithm could be used as one of the main steps in data preprocessing for high dimensional data in machine learning. Therefore, it could be used in a wide specter of classification problems on high-dimensional datasets.

Keywords
machine learning, feature selection, ensemble feature selection, ranking filters, metric aggregation, MeLiF, parallel computing

Acknowledgements
This work was financially supported by the Government of the Russian Federation, Grant 074-U01, and the Russian Foundation for Basic Research, Grant 16-37-60115 mol_a_dk.

Введение

В современном мире алгоритмы машинного обучения используются повсеместно. Особенно эффективно их применение в сферах с плохо формализуемыми либо достаточно большими данными, где ручное построение моделей малоэффективно или даже невозможно. Примерами таких областей являются анализ социальных сетей, таргетирование рекламы, автоматическая фильтрация спама, обработка информации, считываемой системами датчиков и др. [1, 2]. В подобных задачах число признаков, описывающих объекты, может достигать десятков или сотен тысяч, что приводит к значительному увеличению времени на построение модели и, как правило, к ухудшению качества предсказаний построенной модели.

Одним из основных семейств алгоритмов машинного обучения, которое позволяет решить обозначенную проблему, являются алгоритмы выбора признаков [3]. Их принято делить на три группы: алгоритмы-обертки, алгоритмы фильтрации и встраиваемые алгоритмы [4].

У каждой из приведенных групп алгоритмов имеются свои достоинства и недостатки:

- алгоритмы-обертки позволяют находить подмножество признаков, близкое к оптимальному, однако время их работы растет экспоненциально с ростом числа признаков в наборе данных, поэтому, как правило, они не применяются для наборов данных с большим числом признаков;
- встраиваемые алгоритмы используют особенности заранее выбранного алгоритма машинного обучения и, исходя из этих особенностей, выбирают множество значимых признаков. Ключевые проблемы данного подхода заключаются в том, что выбранное множество не всегда близко к оптимальному, а сами алгоритмы жестко привязаны к применяемому впоследствии алгоритму машинного обучения;
- алгоритмы фильтрации выбирают итоговое множество признаков исходя из некоторой меры качества признаков и отсекающего правила, определяющего те признаки, которые необходимо отфильтровать. Данные алгоритмы позволяют достаточно быстро выбрать некоторое подмножество признаков, так как во время отбора не производится обучение модели, однако выбранный набор признаков, скорее всего, достаточно далек от оптимального.

Зачастую возникают задачи, в которых тяжело определить, какой именно подход использовать и какой из алгоритмов подойдет лучше всего. В связи с этим, чтобы не тратить дополнительные ресурсы на выбор хорошей модели для выбора признаков, используют различные агрегации нескольких моделей. Всего принято рассматривать три подхода к агрегации алгоритмов выбора признаков. В первом подходе агрегируют несколько моделей, уже обученных на нескольких наборах данных, отобранных из исходного агрегируемыми алгоритмами выбора признаков [5, 6]. Во втором подходе каждому признаку сопоставляется некоторый ранг, исходя из заданного метода ранжирования, а затем несколько ранжированных списков объединяются между собой в один общий, и с его помощью осуществляется итоговый выбор признаков [7–9]. В последнем подходе признаки оцениваются исходя из некоторой метрики, и затем эти метрики объединяются между собой [10, 11].

В работе [10] был предложен метод выбора признаков, в котором задача выбора признаков сводится к построению линейной композиции мер качества признаков. По результатам экспериментов, описанных в работе [10], видно, что предложенный метод значительно медленнее, чем фильтрующие методы, так как подразумевает некоторое число циклов обучения–тестирования. Однако число таких циклов значительно меньше, чем в алгоритмах-обертках, что позволяет эффективно использовать его на достаточно больших наборах данных. Что же касается качества классификации, то предложенный в [10] метод превосходит агрегируемые методы фильтрации по качеству выбора признаков, но, скорее всего, находит субоптимальное решение, в отличие от методов обертки. Таким образом, данный подход сочетает в себе достаточно высокую скорость работы и хорошее качество работы итогового алгоритма машинного обучения. Но, как показали исследования, предложенный алгоритм не допускает эффективной наивной параллелизации [11], которая позволила бы решить часть указанных проблем алгоритма.

Целью данной работы является разработка эффективной параллелизации для алгоритма Measure Linear Form (MeLiF).

MeLiF: метод выбора признаков на основе агрегации метрик ранжирующих фильтров

В данном подходе задача выбора признаков была сведена к задаче оптимизации. Общая схема алгоритма приведена на рисунке. Алгоритм требует спецификации меры оценки качества работы алгоритма выбора признаков. Зачастую в качестве этой меры выбирается мера оценки качества работы заранее выбранного алгоритма кластеризации на обработанном наборе данных.

Исходными, помимо набора данных, для алгоритма являются множества мер оценки значимости признаков, на основе которых будет строиться одна специфическая мера, наиболее подходящая для решаемой задачи.

Алгоритм состоит из следующих шагов.

Шаг 1. Для исходного набора данных и каждой меры рассчитывается значимость каждого признака.

Шаг 2. Выбираются некоторые стартовые коэффициенты α_i для агрегации мер.

Шаг 3. Из взвешенной суммы метрик старых списков форируется новый список значений для каждого признака.

Шаг 4. Новый список заново сотритуется в порядке убывания новой метрики, к нему применяется выбранное отсекающее правило.

Шаг 5. На оставшихся признаках проводится цикл обучения и оценки результата согласно выбранной мере качества. В соответствии с выбранным алгоритмом оптимизации изменяются коэффициенты α_i и запускается процесс с шага 2, либо программа останавливается.

В методе MeLiF, описанном в работе [10], в качестве алгоритма для шага 5 выбрана жадная модификация метода спуска. Жадность подхода позволяет значительно сократить число циклов алгоритма и ускорить его работу. О том, какие метрики использовались для агрегации, какое отсекающее правило использовалось и каковы были результаты исходного метода MeLiF, будет сказано в следующем разделе. Сама жадная модификация устроена следующим образом: алгоритм поиска поочередно пытается изменить каждый из коэффициентов агрегации мер α_i на некоторую $+\delta$ и $-\delta$. Если для данного α_i удалось получить некоторое улучшение, то переходим к шагу 2, если не удалось получить улучшения ни для какого α_i, то алгоритм останавливается. Таким образом, жадность алгоритма заключается в том, что он сразу движется в сторону первого обнаруженного улучшения, а не ищет наиболее сильное улучшение среди всех вариантов вокруг текущей точки.

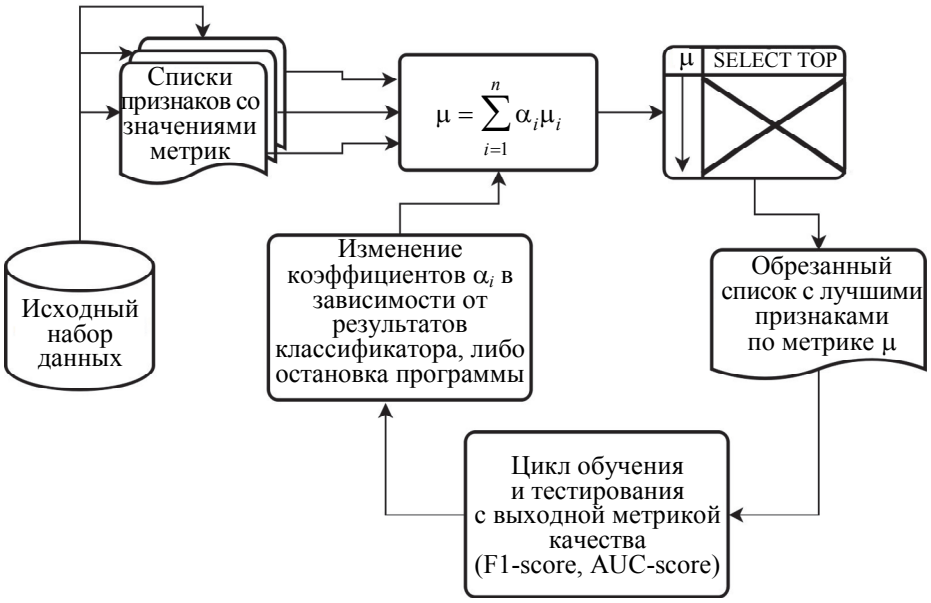

Рисунок. Общая схема алгоритма выбора признаков, основанного на агрегации ранжирующих метрик

Как видно из схемы алгоритма, шаги сразу несколько циклов оптимизации могут быть произведены параллельно в несколько потоков выполнения. Конечно, это не обязательно приведет к прямому улучшению результата, однако, позволит за меньший временной отрезок собрать больше информации о пространстве поиска, и, как следствие, завершить поиск быстрее.

Описание параллельной реализации метода MeLiF

Прежде всего следует упомянуть некоторые особенности исходного метода. Во-первых, исходный метод искал агрегацию четырех метрик, взятых из ранжирующих алгоритмов выбора признаков, описанных в следующем разделе. Во-вторых, исходный алгоритм на тестовых наборах данных всегда сходился к локальному оптимуму быстрее, чем за 100 циклов. Данный факт позволяет внести дополнительные огра-

ничения на параллельную реализацию. В качестве отсекающего правила используется правило top100, которое оставляет ровно 100 лучших признаков из исходного набора по полученной агрегированной метрике.

Для удобства будем называть набор коэффициентов агрегации мер значимости «точкой», так как он, по сути, является точкой в пространстве поиска. Исходный алгоритм запускает поиск оптимальной агрегации с нескольких стартовых точек. В ходе эксперимента было установлено, что из большинства выбранных стартовых точек алгоритм делает всего несколько шагов и затем останавливается [10]. На основе проведения эксперимента с наивной реализацией параллельного выполнения, при которой старт каждой новой стартовой точки обсчитывается в отдельном потоке выполнения, был получен вывод, что она не является оптимальной, так как большая часть потоков быстро завершает выполнение и процесс продолжается уже в одном–двух потоках.

Для решения обозначенных проблем в настоящей работе предложен подход, основанный на очереди с приоритетом. Все стартовые точки складываются в очередь с приоритетом 1,0. Далее свободные потоки берут точку из очереди и начинают ее обработку, либо ждут появления новых точек, если очередь пуста. После обработки потоком в очередь складываются точки-соседи только что посчитанной точки с приоритетом, равным значению оптимизируемой величины в данной точке. После этого поток берет следующую точку на обработку. Для того чтобы все точки имели возможность рано или поздно быть обсчитанными после каждых нескольких итераций, приоритет всех точек, уже положенных в очередь, немного увеличивается.

В отличие от оригинального метода, у данного метода нет явного критерия остановки, поэтому в качестве такого критерия были опробованы различные ограничения, описанные далее. Приведенный алгоритм устраняет недостатки наивной реализации, позволяя использовать потоки выполнения на полную мощность без простаивания. Исключением является самое начало работы, в ситуации, когда число потоков выполнения больше, чем число стартовых точек, однако сразу же после появления дополнительных точек для обсчета проблема устраняется.

Описание экспериментов

Алгоритм на основе очереди с приоритетом строился аналогично оригинальному методу MeLiF, описанному ранее. Сначала выбирались стартовые коэффициенты α_i для агрегации мер. Такими стартовыми значениями послужили точки следующего вида: (1, 0, …, 0), (0, 1, …, 0), …, (0, 0, …, 1), а также точка (1, 1, …, 1). По сути, эти точки изначально являются достаточно хорошими приближениями искомой агрегации, так как соответствуют исходным фильтрам. В качестве агрегируемых мер, как и для оригинального метода, послужили четыре метрики:

– коэффициент корреляции Спирмана (Spearman correlation);
– симметричная неопределенность (symmetrical uncertainty);
– метрика различия значений (value difference metric);
– критерий приспособленности (fit criterion) [10].

Для каждой из агрегируемых мер формируется упорядоченный список признаков с вычисленными значениями мер. Далее алгоритм действует согласно схеме, описанной в предыдущем разделе. Для проверки качества отобранных признаков, так же как и в оригинальном MeLiF, как классификатор используется SVM из библиотеки Weka[1] с параметром C = 1 и скользящим контролем по пяти стратам. Отсечение производилось по правилу top100 – выбор ста наиболее значимых признаков. Оценка качества работы построенного алгоритма выбора признаков выполнялась по F_1-мере качества работы итогового классификатора на заданном наборе данных.

Алгоритмы сравнивались между собой по времени, затраченному на оптимизацию и итоговому качеству классификации на 36 наборах данных ДНК-микрочипов из различных открытых баз данных[2,3,4,5,6]. Эксперименты проводились на компьютере со следующими характеристиками: 32 ядерным процессором AMD Opteron 6272, 2.1 GHz, 128 GB RAM. Для вычислений было использовано 50 потоков.

Экспериментальное исследование параллельной реализации метода MeLiF на основе очереди с приоритетом

При проведении экспериментов было важно выбрать критерий остановки алгоритма, использующего очередь с приоритетом. В качестве критерия остановки были опробованы несколько подходов: остановка после 75 циклов обучения–тестирования, остановка после 100 циклов обучения–тестирований,

[1] Weka 3: Data Mining Software in Java. http://www.cs.waikato.ac.nz/ml/weka/
[2] Набор данных GEO, http://www.ncbi.nlm.nih.gov/geo
[3] Наборы данных раковых заболеваний Broad institute, http://www.broadinstitute.org/cgi-bin/cancer/datasets.cgi
[4] Медицинский набор данных Kent Ridge, http://datam.i2r.a-star.edu.sg/datasets/krbd
[5] Наборы данных для выбора признаков государственного университета Аризоны, http://featureselection.asu.edu/datasets.php
[6] Конкурс по извлечению информации из данных RSCTC'2010 Discovery Challenge, http://tunedit.org/repo/RSCTC/2010/B/public

остановка после 125 циклов обучения–тестирования и остановка после того, как последние 32 шага не улучшали текущий известный оптимум. По итогу экспериментов было выявлено, что наилучшие результаты выдает алгоритм, который останавливается после 75 циклов обучения–тестирования.

Результаты экспериментального сравнения алгоритма на основе очереди с приоритетом указаны в таблице. Столбцы MeLiF и Priority в общем столбце «Время» содержат временные затраты соответствующих алгоритмов в секундах на компьютере, описанном в предыдущем разделе. Столбцы MeLiF и Priority в общем столбце F_1 содержат F_1-меру соответствующих методов на указанных наборах данных. F_1-мера вычисляется как среднее гармоническое между точностью и полнотой построенной модели. Это позволяет получить более репрезентативную оценку в случае, когда набор данных не сбалансирован по классам. F_1-мера принимает значение в интервале от нуля до единицы: чем ближе значение к единице, тем лучше результат.

Набор данных	Время, с		F_1	
	MeLiF	Priority	MeLiF	Priority
Arizona1	558	85	0,833	0,833
Arizona5	219	37	0,768	**0,786**
Breast	161	27	**0,844**	0,812
CNS	33	7	0,742	**0,899**
Data_train0	172	28	**0,853**	0,849
Data_train1	180	32	0,866	**0,877**
Data4_train	513	73	**0,823**	0,775
Data5_train	370	59	0,847	**0,901**
Data6_train	381	65	0,835	**0,859**
DLBCL	65	12	0,799	0,800
GDS2771	299	42	0,798	0,801
GDS2819_1	303	15	1,000	1,000
GDS2819_2	436	60	0,948	**0,957**
GDS2901	88	4	1,000	1,000
GDS2960	33	5	**0,990**	0,977
GDS2961	49	8	**0,860**	0,829
GDS2962	45	8	0,877	**0,924**
GDS3116	142	30	0,852	**0,868**
GDS3257	131	8	1,000	1,000
GDS3929	376	45	0,809	0,810
GDS4103	265	54	**0,933**	0,923
GDS4109	142	24	**0,936**	0,924
GDS4222	454	73	0,974	0,970
GDS4318	275	40	0,923	**0,970**
GDS4336	200	30	**0,928**	0,916
GDS4431	537	85	**0,827**	0,817
GDS4600	472	94	0,983	0,979
GDS4837_1	413	57	**0,916**	0,828
GDS4837_3	316	54	0,960	**0,969**
GDS4901	220	44	**0,931**	0,919
GDS4968_0	226	37	0,905	**0,913**
GDS4968_1	224	40	0,923	**0,939**
GDS5037_0	243	49	0,825	**0,867**
GDS5037_2	293	47	0,756	**0,789**
GDS5047	185	9	1,000	1,000
GDS5083	195	29	0,862	**0,872**
Leukemia_3c_0	34	7	0,989	0,986
Leukemia_3c_1	33	8	0,981	0,98
Ovarian	192	9	1,000	1,000
plySRBCT	17	1	1,000	1,000
prostate_tumor	93	16	0,919	**0,927**

Таблица. Время работы и F_1-мера приведенных алгоритмов на различных наборах данных. Жирным выделен лучший результат для заданного набора данных

Как видно из таблицы, во всех экспериментах произошло многократное ускорение работы исходного метода. Минимальное ускорение составило 4,125 раза, максимальное – 22 раза, среднее ускорение – 8 раз. В 16 случаях применение нового метода позволило улучшить итоговый результат выбора признаков, в 11 случаях – ухудшить, а в остальных 9 случаях привело к похожему или такому же результату.

Заключение

Проведено экспериментальное сравнение метода выбора признаков MeLiF и его оптимизации на основе очереди с приоритетом. В качестве алгоритма машинного обучения для настройки параметров использовался SVM из библиотеки машинного обучения Weka. Сравнение было произведено на 36 наборах данных. В результате исследования можно сделать следующие выводы:

- предложенный метод значительно превосходит оригинальный по производительности, что позволяет ставить его в один ряд с фильтрующими методами машинного обучения по времени работы;
- благодаря использованию очереди с приоритетом удалось сбалансировать нагрузку на потоки, что позволило ускорить сходимость метода и, как следствие, снизить время работы, необходимое для оптимизации. Кроме того, благодаря подобной балансировке можно добиться практически линейного роста производительности при увеличении числа вычислительных узлов;
- полученные в результате работы значения оптимизируемой величины показывают, что предложенный метод сходится примерно также, как и исходный. Примерно в трети случаев метод сходится к более хорошему результату, чем исходный, примерно в трети – к худшему, и примерно в трети – к такому же. При этом в ситуации, когда метод сходится к худшему результату, итоговый результат оптимизации все равно превосходит результаты агрегируемых фильтрующих методов.

Литература

1. Fan J., Samworth R., Wu Y. Ultrahigh dimensional feature selection: beyond the linear model // Journal of Machine Learning Research. 2009. V. 10. P. 2013–2038.
2. Bolon-Canedo V., Sanchez-Marono N., Alonso-Betanzos A. et.al. A review of microarray datasets and applied feature selection methods // Information Sciences. 2014. V. 282. P. 111–135. doi: 10.1016/j.ins.2014.05.042
3. Saeys Y., Inza I., Larranaga P. A review of feature selection techniques in bioinformatics // Bioinformatics. 2007. V. 23. N 19. P. 2507–2517. doi: 10.1093/bioinformatics/btm344
4. Jiliang T., Salem A., Huan L. Feature Selection for Classification: A Review. CRC Press, 2014. 37 p.
5. Dietterich G. Ensemble methods in machine learning // Lecture Notes in Computer Science. 2000. V. 1857. P. 1–15.
6. Bolon-Canedo V., Sanchez-Marono N., Alonso-Betanzos A. An ensemble of filters and classifiers for microarray data classification // Pattern Recognition. 2012. V. 45. N 1. P. 531–539. doi: 10.1016/j.patcog.2011.06.006
7. DeConde R.P., Hawley S., Falcon S. et.al. Combining results of microarray experiments: a rank aggregation approach // Statistical Applications in Genetics and Molecular Biology. 2006. V. 5. P. i-23.
8. Dwork C. et.al. Rank aggregation methods for the web // Proc. 10th Int. Conf. on World Wide Web. 2001. P. 613–622.
9. Filchenkov A. et. al. PCA-based algorithm for constructing ensembles of feature ranking filters // Proc. ESANN. Bruges, Belgium, 2015. P. 201–206.
10. Smetannikov I., Filchenkov A. MeLiF: filter ensemble learning algorithm for gene selection // Advanced Science Letters. 2016. V. 22. N 10. P. 2982–2986. doi: 10.1166/asl.2016.7078
11. Isaev I., Smetannikov I. MeLiF+: Optimization of filter ensemble algorithm with parallel computing // IFIP Advances in Information and Communication Technology. 2016. V. 475. P. 341–347. doi: 10.1007/978-3-319-44944-9_29

References

1. Fan J., Samworth R., Wu Y. Ultrahigh dimensional feature selection: beyond the linear model. Journal of Machine Learning Research, 2009, vol. 10, pp. 2013–2038.
2. Bolon-Canedo V., Sanchez-Marono N., Alonso-Betanzos A. et.al. A review of microarray datasets and applied feature selection methods. *Information Sciences*, 2014, vol. 282, pp. 111–135. doi: 10.1016/j.ins.2014.05.042
3. Saeys Y., Inza I., Larranaga P. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 2007, vol. 23, no. 19, pp. 2507–2517. doi: 10.1093/bioinformatics/btm344
4. Jiliang T., Salem A., Huan L. *Feature Selection for Classification: A Review*. CRC Press, 2014, 37 p.
5. Dietterich G. Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 2000, vol. 1857, pp. 1–15.
6. Bolon-Canedo V., Sanchez-Marono N., Alonso-Betanzos A. An ensemble of filters and classifiers for microarray data classification. *Pattern Recognition*, 2012, vol. 45, no. 1, pp. 531–539. doi: 10.1016/j.patcog.2011.06.006
7. DeConde R.P., Hawley S., Falcon S. et.al. Combining results of microarray experiments: a rank aggregation approach. *Statistical Applications in Genetics and Molecular Biology*, 2006, vol. 5, pp. i-23.
8. Dwork C. et.al. Rank aggregation methods for the web. *Proc. 10th Int. Conf. on World Wide Web*, 2001, pp. 613–622.
9. Filchenkov A. et. al. PCA-based algorithm for constructing ensembles of feature ranking filters. *Proc. ESANN*. Bruges, Belgium, 2015, pp. 201–206.
10. Smetannikov I., Filchenkov A. MeLiF: filter ensemble learning algorithm for gene selection. *Advanced Science Letters*, 2016, vol. 22, no. 10, pp. 2982–2986. doi: 10.1166/asl.2016.7078
11. Isaev I., Smetannikov I. MeLiF+: Optimization of filter ensemble algorithm with parallel computing. *IFIP Advances in Information and Communication Technology*, 2016, vol. 475, pp. 341–347. doi: 10.1007/978-3-319-44944-9_29

Авторы

Сметанников Иван Борисович – аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, smeivan@mail.ru

Authors

Ivan B. Smetannikov – postgraduate, ITMO University, Saint Petersburg, 197101, Russian Federation, smeivan@mail.ru