

Метод средних решающих правил для построения нелинейных границ при решении больших двухклассовых задач SVM

А.И. Макарова¹, М.Ю. Курбаков¹, В.В. Сулимова¹

¹Тульский государственный университет, пр. Ленина 92, Тула, Россия, 300012

Аннотация. В рамках предыдущей работы нами был предложен метод средних решающих правил для быстрого приближенного решения двухклассовой задачи SVM в пространстве признаков. Было показано, что предложенный метод позволяет в условиях большого числа объектов достаточно быстро найти решение, не сильно отличающееся от точного. В рамках данной работы мы обобщаем предложенный подход на случай обучения в пространстве, порожденном потенциальной функцией, что позволяет в условиях больших обучающих совокупностей строить нелинейные границы, разделяющие объекты пары классов. Эксперименты показывают, что предложенный вариант метода, как и исходный метод средних решающих правил, является экономичным по памяти и обладает высокой степенью параллелизма по данным, что дает возможность его эффективной реализации с применением технологий параллельных и распределенных вычислений.

1. Введение

Особенность современных прикладных задач, обусловленная глобальной информатизацией, состоит в необходимости анализа большого числа объектов. Не являются исключением и задачи двухклассового распознавания [1], возникающие в таких важных областях, как молекулярная биология, горнодобывающая и нефтяная промышленности, медицинские системы и системы видеонаблюдения, маркетинг и многих других.

Решение задачи двухклассового распознавания состоит из двух этапов: обучение и распознавание. На этапе обучения на основе анализа некоторой доступной совокупности объектов, снабженных метками, определяющими их классовую принадлежность к одному из двух классов, строится решающее правило распознавания - классифицирующая функция, которая для любого поступившего на ее вход объекта (в том числе не участвовавшего в обучении) определяет метку класса. Этап распознавания заключается в применении построенного решающего правила к новым объектам для определения их классов.

Одним из наиболее удобных и хорошо зарекомендовавших себя методов решения задач обучения двухклассовому распознаванию является метод опорных векторов (Support Vector Machines, SVM), предложенный В.Н. Вапником [2], который заключается в построении решающего правила распознавания в виде оптимальной линейной разделяющей гиперплоскости.

В классическом случае обучение производится в линейном пространстве действительных признаков объектов. Однако если сложно сформировать признаковое описание или необходимо

построить нелинейную разделяющую границу, оказывается более целесообразным применять так называемый беспризнаковый вариант метода. Тогда обучение осуществляется в пространстве, порожденном потенциальной функцией (kernel function) - специальной мерой сходства, обладающей свойствами скалярного произведения [3].

Решение небольших задач SVM как в линейном признаковом пространстве, так и в пространстве, порожденном потенциальной функцией, проблемы не представляет, в то время как обучение по большой обучающей совокупности оказывается очень ресурсоемким и трудоемким, в связи с чем появилось множество подходов, направленных на преодоление проблем нехватки оперативной памяти и повышения производительности вычислений.

В частности, для обучения в условиях, когда объектов много, но они помещаются в памяти одного компьютера или имеет место потоковое поступление данных, были предложены методы, позволяющие сократить объем требуемой памяти и ускорить вычисления на одном компьютере, среди которых можно выделить две группы: 1) инкрементные и декрементные методы, детальный обзор которых приведен в [4], в том числе метод стохастического градиентного спуска (SGD) [5] и 2) методы, основанные на декомпозиции: метод образования фрагментов (chunking) [6], метод последовательной оптимизации Sequential Minimal Optimization (SMO) [7] и другие. Методы декомпозиции положены в основу таких популярных библиотек для решения задачи SVM, как LIBSVM [8, 9] и SVMlight [10, 25].

Кроме того, целая серия работ посвящена адаптации методов обучения к условиям распределенной обработки большого числа объектов: в частности, параллельные реализации стохастического градиентного спуска (SGD) с синхронным [11] и асинхронным [12–14] взаимодействием процессов; параллельные реализации покоординатного спуска при решении двойственной задачи обучения [13–15]; методы, осуществляющие аппроксимацию матрицы скалярных произведений [16], кэширование значений матрицы скалярных произведений и "сжатие вычислений" (shrinking) [10] для сокращения требуемого объема памяти и уменьшения сетевого трафика; методы, учитывающие разреженность признаков [17]. Также осуществлялись неоднократные попытки выразить процесс обучения в терминах парадигмы параллельной обработки больших данных MapReduce [18, 19] и т.д.

Таким образом, в мировом сообществе накопилось большое количество подходов к решению задачи обучения по методу SVM. Однако, несмотря на массовость исследований в данной области, универсального средства для решения задачи SVM до сих пор не найдено. Каждый из подходов и его реализаций имеет свои достоинства и недостатки. В частности, к наиболее существенным недостаткам можно отнести: 1) отсутствие возможности обучения в пространстве, порожденном потенциальной функцией; 2) итерационную организацию вычислений с зависимостями по данным, что существенно снижает эффективность применения технологий параллельных и распределенных вычислений.

В связи с этим в рамках предыдущей работы [20] нами был предложен метод средних решающих правил (Mean Decision Rule method, MDR) для быстрого приближенного решения больших задач SVM в пространстве признаков, а также его высокопроизводительная реализация [21] с применением технологии параллельного программирования MPI.

В рамках данной работы мы обобщаем предложенный подход на случай обучения в пространстве, порожденном потенциальной функцией, что позволяет в условиях больших обучающих совокупностей строить нелинейные границы, разделяющие объекты пары классов. Эксперименты показывают, что предложенный вариант метода, как и исходный метод средних решающих правил, является экономичным по памяти (что позволяет использовать его для обучения даже на одной машине) и обладает высокой степенью параллелизма по данным (что дает возможность его эффективной реализации с применением технологий параллельных и распределенных вычислений).

2. Основная идея метода средних решающих правил

Основная идея метода средних решающих правил (Mean Decision Rule method, MDR), предложенного нами в [20], заключается в формировании множества небольших случайных подвыборок обучающей совокупности, независимом обучении отдельно по каждой из

подвыборки и последующем объединении частных решающих правил распознавания в одно общее путем их усреднения, которое, фактически, сводится к усреднению параметров отдельных решающих правил – направляющего вектора и смещения вдоль него.

В [20] теоретически обосновано и подтверждено экспериментами на модельных и реальных данных в линейном признаковом пространстве, что метод MDR позволяет достаточно быстро найти приближенное, но не сильно отличающееся от точного решение задачи SVM.

В следующем разделе приводится обобщение данного подхода для случая обучения в пространстве, порожденном потенциальной функцией.

3. Метод средних решающих правил (KMDR) для обучения в пространстве, порожденном потенциальной функцией

Пусть Ω^* – множество всех возможных объектов некоторого вида. И пусть $\Omega = \{\omega_j, j = 1, \dots, |\Omega|\} \subset \Omega^*$ – доступное подмножество объектов, снабженных метками $y_j = y(\omega_j) \in \{+1, -1\}$, определяющими классовую принадлежность и образующее вместе с ними обучающую совокупность $[\Omega, Y]$, $\Omega = \{\omega_j, j = 1, \dots, |\Omega|\}$, $Y = \{y_j = y(\omega_j), j = 1, \dots, |\Omega|\}$ для построения решающего правила распознавания, позволяющего оценить скрытую принадлежность любого объекта $\omega \in \Omega^*$ к одному из двух классов $\hat{y}(\omega) : \Omega^* \rightarrow \{-1, 1\}$.

В классической теории распознавания образов [1] предполагается, что на объектах измерены значения признаков, позволяющих рассматривать их как точки в линейном пространстве. Однако в ряде практических задач оказывается сложно заранее указать набор признаков, позволяющих сформировать признаковое пространство, в котором выполнялась бы гипотеза компактности – объекты разных классов образовывали бы компактные, не пересекающиеся (или мало пересекающиеся) множества в этом пространстве.

В таких случаях объекты обычно представляются значениями их парного сходства или несходства с другими объектами. Удобно в качестве способа сравнения использовать меру сходства специального вида, называемую потенциальной функцией $K(\omega', \omega''), \omega', \omega'' \in \Omega^*$ (в англоязычной терминологии kernel function) – действительную симметричную функцию двух аргументов, обладающую свойствами скалярного произведения, матрица значений которой для любой конечной совокупности объектов (в том числе и для обучающего множества) неотрицательно определена [3]. Любая потенциальная функция $K(\omega', \omega''), \omega', \omega'' \in \Omega^*$ погружает множество объектов Ω^* в гипотетическое евклидово линейное пространство $\tilde{\Omega} \supset \Omega^*$ с евклидовой метрикой

$$\rho(\omega', \omega'') = \sqrt{K(\omega', \omega') + K(\omega'', \omega'') - 2 \cdot K(\omega', \omega'')},$$

а сама играет роль скалярного произведения в этом пространстве [22].

Свойства потенциальной функции позволяют сформулировать метод опорных векторов (в данном случае его более уместно называть методом опорных объектов), минуя введение признакового описания объектов. Такой подход получил название беспризнакового [23].

Однако и при наличии признакового описания потенциальные функции часто используются для введения нелинейности при построении решающего правила, что в ряде случаев позволяет повысить качество распознавания. Чаще всего в таком случае используется так называемая радиальная потенциальная функция (Radial Basis Function, RBF) [3]:

$$K(\omega', \omega'') = \exp[-\gamma r^2(\omega', \omega'')], \quad (1)$$

где $\gamma > 0$ – параметр потенциальной функции, а $r(\omega', \omega'')$ – Евклидово расстояние между векторами признаков $\mathbf{x}' = \mathbf{x}(\omega')$ и $\mathbf{x}'' = \mathbf{x}(\omega'')$, представляющими соответствующие объекты.

Согласно методу опорных векторов (SVM) решающее правило распознавания строится в виде оптимальной разделяющей гиперплоскости, которая в терминах множителей Лагранжа $\lambda_j, j = 1, \dots, |\Omega|$, являющихся решением соответствующей оптимизационной задачи, в беспризнаковом случае имеет вид [24]:

$$d(\omega | \lambda, b) = \sum_{j=1}^N \lambda_j y_j K(\omega_j, \omega) + b \quad \begin{cases} \geq 0 \Rightarrow \hat{y}(\omega) = +1, \\ < 0 \Rightarrow \hat{y}(\omega) = -1. \end{cases} \quad (2)$$

Метод средних решающих правил при обучении в пространстве, порожденном потенциальной функцией заключается в формировании множества небольших случайных подвыборок $[\Omega^{(i)} \subset \Omega, Y^{(i)} \subset Y] = [\Omega, Y]^{(i)}$, $i = 1, \dots, k$ исходной обучающей совокупности, независимо построению решающего правила

$$d(\omega | \lambda^{(i)}, b^{(i)}) = \sum_{j=1}^{|\Omega^{(i)}|} \tilde{\lambda}_j^{(i)} y_j K(\omega_j, \omega) + b^{(i)}, \quad i = 1, \dots, k \quad (3)$$

по каждой частной подвыборке и последующем нахождении среднего решающего правила

$$d(\omega | \lambda, b) = \frac{1}{k} \sum_{i=1}^k d(\omega | \lambda^{(i)}, b^{(i)}).$$

Каждое решающее правило, построенное по частной подвыборке $[\Omega, Y]^{(i)}$, характеризуется набором своих множителей Лагранжа $\tilde{\lambda}_t^{(i)}$, $t = 1, \dots, |\Omega^{(i)}|$ и смещением $b^{(i)}$.

В силу линейной структуры решающих правил (3) их усреднение возможно путем усреднения их параметров $\lambda^{(i)}$ и $b^{(i)}$, $i = 1, \dots, k$. При этом для обеспечения корректности усреднения мы производим расширение данных векторов до размера всей обучающей совокупности $|\Omega|$, добавляя нули в позиции, соответствующие объектам, которых не было на обучении в частной (i -й) подвыборке:

$$\lambda_j^{(i)} = \begin{cases} \tilde{\lambda}_{t(j)}^{(i)}, & t(j) \in \{1 \dots |\Omega^{(i)}|\} \text{ если } \omega_j \in \Omega^{(i)}, \\ 0, & \text{иначе,} \end{cases} \quad j = 1, \dots, |\Omega|. \quad (4)$$

В результате среднее решающее правило принимает вид

$$d(\omega; \lambda, b) = \sum_{j=1}^{|\Omega|} \lambda_j y_j K(\omega_j, \omega) + b, \quad \lambda_j = \frac{1}{k} \sum_{i=1}^k \lambda_j^{(i)}, \quad b = \frac{1}{k} \sum_{i=1}^k b^{(i)},$$

где множители Лагранжа усредняются с учетом (4).

Учет указанной особенности требует использования большего количества оперативной памяти и большего объема вычислений по сравнению с признаковым случаем, поскольку не позволяет в явном виде вычислить направляющий вектор оптимальной разделяющей гиперплоскости. Однако, несмотря на это, предложенный подход, по-прежнему, не имеет теоретического ограничения на размер обучающей выборки, поскольку не требует одновременного нахождения в памяти всех объектов, что обеспечивает возможность его применения для обучения даже на одной вычислительной машине.

Еще одним важным достоинством метода KMDR является то, что он имеет простое алгоритмическое выражение и при этом обладает высокой степенью параллелизма по данным, в результате чего удобен для реализации с использованием технологий параллельной и распределенной обработки данных.

Кроме того, следует отметить, что для обучения по частным подвыборкам может использоваться любая реализация SVM, в том числе для GPU. Мы в данном случае используем библиотеку LibSVM [8,9].

4. Экспериментальное исследование

4.1. Модель генерации данных

В рамках данной работы экспериментальное исследование метода KMDR проводилось на нескольких наборах модельных данных, отличающихся числом объектов и числом признаков.

Генерация обучающих и тестовых множеств для всех экспериментов осуществлялась одинаково, внутри гиперкуба со стороной $a = 300$, согласно следующей модели:

- объекты двух классов образуют пересекающиеся множества,
- объекты положительного класса распределены равномерно внутри гиперсферы радиуса $r = 100$, а вне ее их количество убывает экспоненциально с увеличением расстояния от гиперсферы (равномерно по всем направлениям),

- объекты отрицательного класса распределены равномерно вне гиперсферы, а внутри нее их количество убывает экспоненциально с увеличением расстояния от гиперсферы.

4.2. Характеристики вычислительной системы

Экспериментальное исследование проводилось на персональном компьютере со следующими основными характеристиками: процессор Intel® Core™ i5-8250U CPU 1.6 GHz, ОЗУ 8 Гб, ОС 64-разрядная.

4.3. Визуализация решающих правил в двухмерном пространстве

Для модельных данных в виде точек в двухмерном признаковом пространстве решающие правила можно увидеть визуально. На рисунке 1 представлены решающие правила, построенные по методу KMDR и при помощи библиотеки LibSVM [8,9] для набора данных, состоящего из 1000 объектов. При этом при обучении были использованы следующие параметры: параметр SVM $c = 10$, потенциальная функция – радиальная (1) с параметром $g = 0.75$, число подвыборок в методе KMDR: 200, число объектов в подвыборке в методе KMDR: 50.

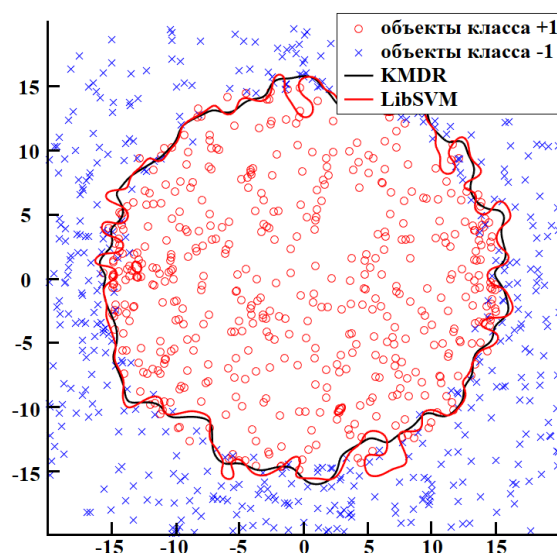


Рисунок 1. Решающие правила, построенные по методу KMDR и при помощи библиотеки LibSVM для набора данных из 1000 объектов.

Из рисунка 1 видно, что построенные решающие правила в целом визуально достаточно близки. При этом решающее правило, построенное при помощи KMDR является более гладким, чем решающее правило, построенное при помощи LibSVM.

4.4. Влияние параметров KMDR на качество и скорость обучения

Метод средних решающих правил имеет два параметра – число подвыборок и число объектов в подвыборке, которые влияют на качество и скорость решения задачи SVM. Примеры соответствующих зависимостей, приведенные в данном разделе, получены на вычислительной системе с характеристиками, указанными в разделе 4.2, для модельной выборки из 10000 объектов и 100 признаков, сгенерированной согласно модели, описанной в разделе 4.1.

На рисунке 2 (слева) представлены зависимости AUC (площади под ROC-кривой) от числа подвыборок для нескольких значений числа объектов в подвыборке. Здесь (и на рисунке 2 справа) сплошными линиями показаны средние значения, полученные по 10 запускам, а пунктирными – соответствующие среднеквадратические отклонения.

Из графиков видно, что при увеличении размера случайной подвыборки качество решения повышается (конечно, это происходит до некоторого предела), а при увеличении числа подвыборок имеется тенденция к уменьшению дисперсии результатов.

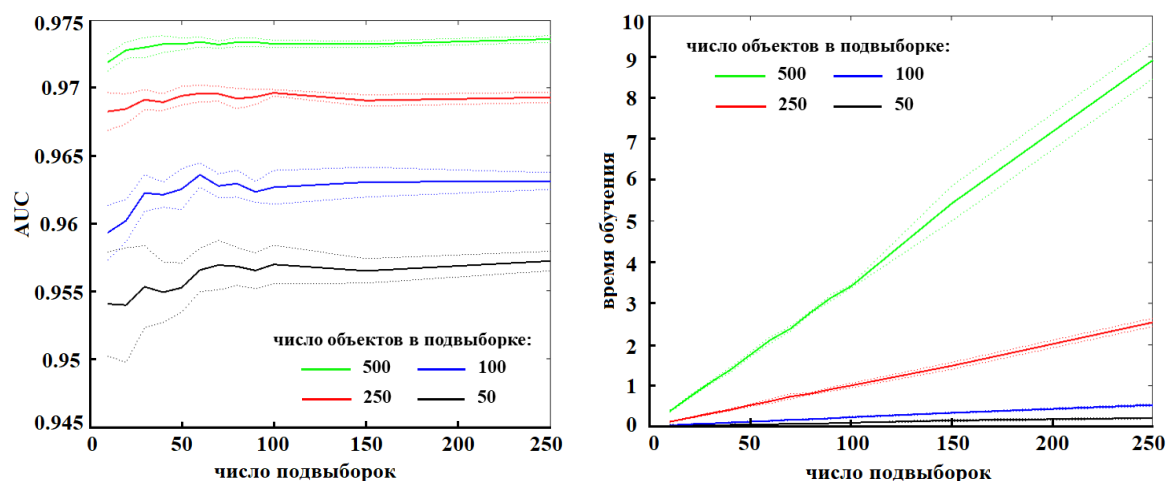


Рисунок 2. Зависимости AUC (слева) и времени обучения в секундах (справа) от числа подвыборок.

На рисунке 2 (справа) представлены зависимости времени обучения (в секундах) от числа подвыборок. Как и следовало ожидать, увеличение значений параметров метода KMDR приводит к увеличению времени обучения, но при этом время растет линейно относительно числа подвыборок, в результате чего не приводит к катастрофически большим абсолютным значениям времени работы.

Таким образом, предложенный метод позволяет балансировать между качеством и скоростью решения в зависимости от имеющихся предпочтений.

4.5. Результаты сравнения KMDR с другими реализациями SVM

В данном разделе приводится сравнение KMDR с другими реализациями SVM, позволяющими строить нелинейную разделяющую границу, такими, как LibSVM [8,9], SVMlight [25] и piSVM [26].

Эксперименты проводились на модельных данных, сгенерированных согласно модели, описанной в разделе 4.1 для разного числа объектов и признаков.

Для введения нелинейности применялись радиальные потенциальные функции (1) со значением параметра радиальности $g = 0.75$. Параметр c метода опорных векторов во всех экспериментах был принят равным 10, а число объектов в подвыборке в методе KMDR – равным 50.

Сравнение качества решения проводилось по площади под ROC-кривой (AUC) для тестовой совокупности. Результаты сравнения приведены в таблице 1, в которой показатели KMDR усреднены по 10 запускам.

Как видно из таблицы 1, предложенный метод позволяет получить достаточно близкие по качеству решения по сравнению со стандартными библиотеками, но при этом имеет существенно преимущество по скорости обучения, которое становится все более очевидным с увеличением числа объектов в обучающей совокупности, оставаясь на уровне всего нескольких секунд даже для 100 000 объектов.

Время распознавания пока остается достаточно большим, поскольку во многом определяется задачей (числом опорных объектов). Но это время может быть легко уменьшено путем привлечения технологий параллельных и распределенных вычислений.

Время обучения для предлагаемого подхода тоже может быть уменьшено за счет высокой степени параллелизма по данным.

5. Заключение

В данной работе предложено обобщение метода средних решающих правил (MDR) для построения нелинейных границ, разделяющих объекты двух классов (KMDR).

Предложенный метод позволяет быстро найти приближенное, но не сильно отличающееся от точного решение двухклассовой задачи SVM для больших обучающих совокупностей, при этом с ростом числа объектов преимущество по скорости обучения перед стандартными библиотеками становится все более очевидным.

Таблица 1. Результаты тестирования на модельных данных.

Число объектов	Число признаков	Метод	Размер подвыборки	AUC	Время обучения (с)	Время распознавания (с)
10 000	10	LibSVM	-	0.9873	6.977966	2.368463
		SVMlight	-	0.987299	6.91	2.32
		piSVM	-	0.987294	4.81	3.71
		KMDR	50	0.9702±0.0015	0.0199±0.009	1.099±0.04
			500	0.9849±0.0004	0.3493±0.029	2.431±0.31
			1000	0.9867±0.0001	0.9841±0.015	2.68±0.021
10 000	100	LibSVM	-	0.977504	92.589	32.242
		SVMlight	-	0.977092	41.93	14.92
		piSVM	-	0.976734	12.04	27.8
		KMDR	50	0.9552±0.0018	0.059±0.014	7.97±0.12
			500	0.9732±0.0004	1.749±0.054	28.75±0.19
			1000	0.9760±0.0003	5.645±0.293	32.08±1.27
100 000	10	LibSVM	-	0.990	3371.334	216.5087
		SVMlight	-	0.990	1871.75	203.07
		piSVM	-	0.990	1124.75	284.55
		KMDR	50	0.970±0.0015	0.0554±0.01	13.36±1.067
			500	0.9836±0.0003	0.4211±0.023	44.962±1.43
			1000	0.9861±0.00008	1.0425±0.039	113.223±0.83
100 000	100	LibSVM	-	0.985	4360.219	1053.589
		SVMlight	-	0.985	8558.51	1873.66
		piSVM	-	0.984	3708.81	2269.96
		KMDR	50	0.960±0.002	0.192±0.063	95.608±3.422
			500	0.977±0.0002	1.83±0.132	690.98±25.44
			1000	0.979±0.0002	5.488±0.26	1097.26±44.61

Параметры метода позволяют осуществить балансирование между качеством и скоростью решения задачи SVM.

Кроме этого, метод KMDR обладает высокой степенью параллелизма по данным, в результате чего его производительность может быть дополнительно повышена путем привлечения технологий параллельных и распределенных вычислений.

6. Благодарности

Работа выполнена при поддержке Российского Фонда Фундаментальных Исследований, грант 18-07-01087.

7. Литература

- [1] Вапник, В.Н. Теория распознавания образов / В.Н. Вапник, А.Я. Червоненкис – М.: Наука, 1974. – 415 с.
- [2] Vapnik, V. Statistical Learning Theory – John-Wiley & Sons, Inc., 1998.
- [3] Айзерман, М.А. Метод потенциальных функций в теории обучения машин / М.А. Айзерман, Э.М. Браверман, Л.И. Розоноэр – М.: Наука, 1970. – 384 с.
- [4] Hoi, S. Online Learning: A Comprehensive Survey / S. Hoi, D. Sahoo, J. Lu, P. Zhao – Technical report, 2018. – 100 p.
- [5] Boser, B.E. A Training Algorithm for Optimal Margin Classifiers // B.E. Boser, I.M. Guyon, V. Vapnik // Fifth Annual Workshop on Computational Learning Theory, ACM, 1992.

- [6] Bottou, L. Stochastic Learning // *Advanced Lectures on Machine Learning*. – 2004. – Vol. 3176. – P. 146-168.
- [7] Platt, J. Sequential minimal optimization: A fast algorithm for training support vector machines // *Technical Report MSR-TR-98-14*. Microsoft Research, 1998.
- [8] Chang, C.-C. LIBSVM: a library for support vector machines / C.-C. Chang, C.-J. Lin, 2001. URL: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [9] LIBSVM-A Library for Support Vector Machines [Электронный ресурс]. – Режим доступа: <https://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- [10] Joachims, T. “Making large-scale support vector machine learning practical” in *Advances in kernel methods: support vector learning* – Cambridge, USA: MIT Press, 1999. – P. 169-184.
- [11] Dekel, O. Optimal distributed online prediction using mini-batches / O. Dekel, R. Gilad-Bachrach, O. Shamir, L. Xiao // *J. Mach. Learn. Res.* – 2012. – Vol. 13(1). – P. 165-202.
- [12] Agarwal, A. Distributed delayed stochastic optimization / A. Agarwal, J.C. Duchi // *Advances in Neural Information Processing Systems*. – 2011. – P. 873-881
- [13] Jaggi, M. Communication-efficient distributed dual coordinate ascent. In: *Advances in Neural Information Processing Systems* / M. Jaggi, V. Smith, M. Tak'ac, J. Terhorst, S. Krishnan, T. Hofmann, M.I. Jordan, 2014. – P. 3068-3076.
- [14] Ma, C. Adding vs. averaging in distributed primal-dual optimization / C. Ma, V. Smith, M. Jaggi, M.I. Jordan, P. Richt'arik, M. Tak'a'c // *arXiv preprint arXiv:1502.03508*, 2015.
- [15] Yang, T. Trading computation for communication: distributed stochastic dual coordinate ascent // *Advances in Neural Information Processing Systems*, 2013. – P. 629-637.
- [16] Chang, E.Y. Psvm: Parallelizing support vector machines on distributed computers / E.Y. Chang, K. Zhu, H. Wang, H. Bai, J. Li, Z. Qiu, H. Cui // *NIPS*. – 2007. – Vol. 20.
- [17] Joachims, T. Training linear svms in linear time // *ACM KDD*. – 2006. – P. 217-226.
- [18] Chu, C.-T. Map reduce for machine learning on multicore / C.-T. Chu, S.K. Kim, Y.-A. Lin, Y. Yu, G. Bradski, A. Y. Ng, K. Olukotun // *NIPS*, 2006.
- [19] Zhao, H.X. Parallel support vector machines on multi-core and multiprocessor systems / H.X. Zhao, F. Magoules // *11th International Conference on Artificial Intelligence and Applications (AIA)*, Innsbruck, Austria. IASTED, 2011. DOI: 10.2316/P.2011.717-056.
- [20] Макарова, А.И. Быстрое приближенное решение двухклассовой задачи SVM для больших обучающих совокупностей / А.И. Макарова, В.В. Сулимова // *Сборник трудов V международной конференции "Информационные технологии и нанотехнологии" (ИТНТ)*. Т. 4. Науки о данных – Самара: Новая техника, 2019. – С. 25-34.
- [21] Kurbakov, M. High-Performance Solution of the Two-Class SVM Problem for Big Data Sets by the Mean Decision Rule Method / M. Kurbakov, A. Makarova, V. Sulimova // *Proc. of International conference “Supercomputing Russian Days”* – Moscow: Springer CCIS, 2019.
- [22] Моттль, В.В. Метрические пространства, допускающие введение линейных операций и скалярного произведения // *ДАН*. – 2003. – Т. 67(1).
- [23] Duin, R.P.W. Experiments with a featureless approach to pattern recognition / R.P.W. Duin, D. De Ridder, D.M.J. Tax // *Pattern Recognition Letters*. – 1997. – Vol. 18(11-13). – P. 1159-1166.
- [24] Вапник, В.Н. Восстановление зависимостей по эмпирическим данным – М.: Наука, 1979. – 44 с.
- [25] svmLight Support Vector Machine [Электронный ресурс]. – Режим доступа: <http://svmlight.joachims.org>.
- [26] PiSvMSoftware [Электронный ресурс]. – Режим доступа: <http://pisvm.sourceforge.net>.

Mean Decision Rule method for constructing nonlinear boundaries in solving large two-class SVM problems

A.I. Makarova¹, M.Y. Kurbakov¹, V.V. Sulimova¹

¹Tula State University, Lenin ave. 92, Tula, Russia, 300012

Abstract. In the previous work, we proposed the Mean Decision Rule method for fast approximate solution of the two-class SVM problem in a feature space. This paper generalized the proposed approach to the case of training in a space generated by a kernel function (Kernel-based Mean Decision Rule method), which allows to construct nonlinear boundaries separating objects of two classes. Experiments show that even for large training sets Kernel-based MDR, like the initial MDR method, allows to quickly find a solution that does not differ much from the exact one. Besides it is economical in memory and has a high degree of data parallelism and so it can be effectively implemented using parallel and distributed computing technologies.