

Строгое полиномиальное отделение двух множеств

В. Н. Малоземов, А. В. Плоткин

Санкт-Петербургский государственный университет,
Российская Федерация, 199034, Санкт-Петербург, Университетская наб., 7–9

Для цитирования: Малоземов В. Н., Плоткин А. В. Строгое полиномиальное отделение двух множеств // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2019. Т. 6 (64). Вып. 2. С. 232–240.
<https://doi.org/10.21638/11701/spbu01.2019.205>

Одной из основных задач математической диагностики является строгое отделение двух конечных множеств в евклидовом пространстве. Широко известно *строгое линейное* отделение, которое сводится к решению задачи линейного программирования. Мы вводим понятие *строгого полиномиального* отделения и показываем, что строгое полиномиальное отделение двух множеств также сводится к решению задачи линейного программирования. Целевая функция предложенной в статье задачи линейного программирования имеет следующую особенность: ее оптимальное значение может равняться только нулю или единице — нулю, если множества допускают строгое полиномиальное отделение, и единице в противном случае. Приведены наглядные примеры строгого отделения двух множеств на плоскости с помощью алгебраических полиномов четвертой степени от двух переменных. Анализируется эффективность применения строгого полиномиального отделения при решении задач бинарной классификации данных.

Ключевые слова: строгое линейное отделение, строгое полиномиальное отделение, линейное программирование, математическая диагностика.

1. Введение. Пусть в пространстве \mathbb{R}^n заданы два конечных множества

$$A = \{a_i\}_{i=1}^m \text{ и } B = \{b_j\}_{j=1}^k.$$

Определение 1. Множества A и B называются *строго линейно отделимыми*, если существует ненулевой вектор $w \in \mathbb{R}^n$ и вещественное число γ такие, что

$$\begin{aligned} \langle w, a_i \rangle - \gamma &\geq 1 \text{ при всех } i \in 1 : m, \\ \langle w, b_j \rangle - \gamma &\leq -1 \text{ при всех } j \in 1 : k. \end{aligned} \quad (1)$$

При выполнении условий (1) говорят также, что гиперплоскость, определяемая уравнением $\langle w, x \rangle = \gamma$, строго отделяет множество A от множества B .

Следуя [1], введем функцию

$$f(g) = \frac{1}{m} \sum_{i=1}^m [-\langle w, a_i \rangle + \gamma + 1]_+ + \frac{1}{k} \sum_{j=1}^k [\langle w, b_j \rangle - \gamma + 1]_+, \quad (2)$$

где $g = (w, \gamma)$ и $[u]_+ = \max\{0, u\}$. Ясно, что $f(g) \geq 0$ при всех $g \in \mathbb{R}^{n+1}$.

Предложение 1. Множества A и B строго линейно отделимы тогда и только тогда, когда существует вектор g_* , на котором $f(g_*) = 0$.

Если $f(g_*) = 0$ на некотором векторе $g_* = (w_*, \gamma_*)$, то необходимо $w_* \neq 0$ и все слагаемые

$$[-\langle w_*, a_i \rangle + \gamma_* + 1]_+ \quad \text{и} \quad [\langle w_*, b_j \rangle - \gamma_* + 1]_+$$

равны нулю. Последнее справедливо лишь тогда, когда

$$\begin{aligned} -\langle w_*, a_i \rangle + \gamma_* + 1 &\leq 0 \text{ при всех } i \in 1 : m, \\ \langle w_*, b_j \rangle - \gamma_* + 1 &\leq 0 \text{ при всех } j \in 1 : k. \end{aligned} \quad (3)$$

Неравенства (3) обеспечивают выполнение условий строгой линейной отделимости (1) с $w = w_*$ и $\gamma = \gamma_*$.

Предложение 1 показывает, что строгое линейное отделение множеств A и B сводится к минимизации функции $f(g)$.

Рассмотрим экстремальную задачу

$$f(g) \rightarrow \min_{g \in \mathbb{R}^{n+1}}, \quad (4)$$

где $f(g)$ — функция вида (2). Эта задача эквивалентна следующей задаче линейного программирования:

$$\begin{aligned} F(w, \gamma, y, z) &:= \frac{1}{m} \sum_{i=1}^m y_i + \frac{1}{k} \sum_{j=1}^k z_j \rightarrow \min, \\ \langle w, a_i \rangle - \gamma + y_i &\geq 1, \quad i \in 1 : m; \\ -\langle w, b_j \rangle + \gamma + z_j &\geq 1, \quad j \in 1 : k; \\ y_i &\geq 0, \quad i \in 1 : m; \quad z_j \geq 0, \quad j \in 1 : k. \end{aligned} \quad (5)$$

Множество планов задачи (5) непусто и целевая функция ограничена снизу нулем. Значит, задача (5) имеет решение. По эквивалентности существует решение и у задачи (4), причем минимальные значения целевых функций у этих задач равны между собой. Это общее значение обозначим через μ . Отметим, что если $(w_*, \gamma_*, y_*, z_*)$ — решение задачи (5), то $g_* := (w_*, \gamma_*)$ — решение задачи (4), при этом $f(g_*) = \mu$. Если $\mu = 0$, то гиперплоскость, определяемая уравнением $\langle w_*, x \rangle = \gamma_*$, строго линейно отделяет множество A от множества B .

Мы кратко напомнили некоторые результаты из работы [1] в том виде, который позволит нам обобщить понятие строгой линейной отделимости. Предварительно введем обозначения, связанные с вектором $x = (x_1, \dots, x_n)$:

$$\varphi_0(x) \equiv 1, \quad w_0 = -\gamma; \quad \varphi_s(x) = x_s \text{ при } s \in 1 : n;$$

$$P(w, x) = \sum_{s=0}^n w_s \varphi_s(x).$$

Выражение $P(w, x)$ есть алгебраический полином первой степени от n переменных. Условия строгой линейной отделимости (1) можно переписать в виде

$$\begin{aligned} P(w, a_i) &\geq 1 \text{ при всех } i \in 1 : m; \\ P(w, b_j) &\leq -1 \text{ при всех } j \in 1 : k. \end{aligned}$$

В следующем параграфе мы введем понятие строгого полиномиального отделения и покажем, что строгое полиномиальное отделение двух множеств также сводится к решению задачи линейного программирования.

2. Строгое полиномиальное отделение. По-прежнему считаем, что в пространстве \mathbb{R}^n заданы два конечных множества

$$A = \{a_i\}_{i=1}^m \text{ и } B = \{b_j\}_{j=1}^k.$$

Введем обобщенный полином

$$P(w, x) = \sum_{s=0}^r w_s \varphi_s(x), \quad (6)$$

где $\varphi_0(x), \varphi_1(x), \dots, \varphi_r(x)$ — произвольные (но фиксированные) базисные функции, заданные и непрерывные на выпуклой оболочке множества $A \cup B$, и $w = (w_0, w_1, \dots, w_r)$ — вектор коэффициентов полинома.

Определение 2. Множества A и B назовем *строго полиномиально отделимыми*, если найдется вектор коэффициентов w такой, что

$$\begin{aligned} P(w, a_i) &\geq 1 \text{ при всех } i \in 1 : m; \\ P(w, b_j) &\leq -1 \text{ при всех } j \in 1 : k. \end{aligned} \quad (7)$$

При выполнении условий (7) будем говорить, что гиперповерхность, определяемая уравнением $P(w, x) = 0$, строго отделяет множество A от множества B .

Введем целевую функцию

$$f(w) = \max \left\{ \max_{i \in 1:m} [1 - P(w, a_i)]_+, \max_{j \in 1:k} [1 + P(w, b_j)]_+ \right\}. \quad (8)$$

Очевидно, что $f(w) \geq 0$ при всех w .

Лемма 1. Равенство $f(w_0) = 0$ выполняется тогда и только тогда, когда полином $P(w_0, x)$ строго отделяет множества A и B .

Доказательство. Условия строгой отделимости (7) можно переписать в эквивалентном виде

$$\max_{i \in 1:m} [1 - P(w, a_i)]_+ = 0 \text{ и } \max_{j \in 1:k} [1 + P(w, b_j)]_+ = 0.$$

Теперь заключение леммы становится очевидным. □

Лемма показывает, что задача строгого полиномиального отделения множеств A и B сводится к минимизации функции $f(w)$.

Рассмотрим экстремальную задачу

$$f(w) \rightarrow \min_{w \in \mathbb{R}^{n+1}}, \quad (9)$$

где $f(w)$ — функция вида (8).

Теорема 1. *Задача (9) эквивалентна следующей задаче линейного программирования:*

$$\begin{aligned} t &\rightarrow \min \\ P(w, a_i) + t &\geq 1, \quad i \in 1 : m; \\ -P(w, b_j) + t &\geq 1, \quad j \in 1 : k; \\ t &\geq 0. \end{aligned} \tag{10}$$

ДОКАЗАТЕЛЬСТВО. Ограничения задачи (10) можно переписать так:

$$\begin{aligned} t &\geq [1 - P(w, a_i)]_+, \quad i \in 1 : m; \\ t &\geq [1 + P(w, b_j)]_+, \quad j \in 1 : k. \end{aligned} \tag{11}$$

Напомним, что две задачи на минимум называются эквивалентными, если любому плану каждой из этих задач можно сопоставить план другой задачи с равным или меньшим значением целевой функции. В данном случае плану $w \in \mathbb{R}^{r+1}$ задачи (9) сопоставим вектор (w, t) , где $t = f(w)$. Согласно (11), вектор (w, t) является планом задачи (9), при этом $t = f(w)$.

Наоборот, пусть (w, t) — план задачи (10). Тогда w — план задачи (9). При этом, в силу (11), $f(w) \leq t$.

Теорема доказана. \square

Задача линейного программирования (10) имеет решение, поскольку множество ее планов непусто (планом является вектор (w, t) с $w = \mathbb{0}$ и $t = 1$) и целевая функция ограничена снизу нулем. По лемме об эквивалентных задачах [2, с. 11] задача (9) также имеет решение, причем минимальные значения у целевых функций задач (9) и (10) совпадают. Обозначим это общее значение через t_* .

Теорема 2. *Величина t_* может принимать только два значения — ноль или единица.*

ДОКАЗАТЕЛЬСТВО. Пусть (w_*, t_*) — решение задачи (10). Ясно, что $t_* \geq 0$. Так как пара $w = \mathbb{0}$, $t = 1$ является планом задачи (10) со значением целевой функции, равным единице, то $t_* \leq 1$. Таким образом, $t_* \in [0, 1]$.

Предположим, что $t_* \in (0, 1)$. Согласно (10) для полинома $P(w_*, t)$ выполняются неравенства

$$\begin{aligned} P(w_*, a_i) &\geq 1 - t_*, \quad i \in 1 : m; \\ -P(w_*, b_j) &\geq 1 - t_*, \quad j \in 1 : k. \end{aligned}$$

Пара $w_0 = \frac{1}{1-t_*} w_*$, $t_0 = 0$ удовлетворяет ограничениям задачи (10). Значит, необходимо $t_* \leq t_0$, что противоречит условию $t_* \in (0, 1)$.

Для t_* остается только две возможности: $t_* = 0$ или $t_* = 1$.

Теорема доказана. \square

Подведем итог. Строгое полиномиальное отделение множеств A и B сводится к решению задачи линейного программирования (10). Если (w_*, t_*) — решение этой задачи, то при $t_* = 0$ полином $P(w_*, x)$ строго отделяет множества A и B . При $t_* = 1$ строгое отделение множеств A и B с помощью полиномов вида (6) невозможно.

Замечание 1. Частным случаем строгого полиномиального отделения является строгое линейное отделение (когда в качестве обобщенного полинома $P(w, x)$ берется алгебраический полином первой степени от n переменных).

Замечание 2. Строгое линейное отделение имеет следующую особенность: для всех точек x из выпуклой оболочки множества A сохраняется неравенство $P(w, x) \geq 1$, а для всех точек x из выпуклой оболочки множества B сохраняется неравенство $P(w, x) \leq -1$. В случае строгого полиномиального отделения ситуация более сложная. Чтобы избежать неправильной идентификации, рекомендуется использовать обобщенные полиномы невысоких порядков. Также необходимо, чтобы количество элементов множества $A \cup B$ значительно превосходило размерность пространства \mathbb{R}^n .

3. Примеры строгого полиномиального отделения на плоскости. На рис. 1 представлены примеры строгого полиномиального отделения двух множеств на плоскости с помощью алгебраических полиномов четвертой степени от двух переменных. На рис. 1, *a* и *b* получилось стандартное нелинейное отделение, в то время как отделение на рис. 1, *в* выглядит нестандартно (эффект «переотделения»). На рис. 1, *г* отделяющая кривая распалась на две бесконечные компоненты. На рис. 1, *д* и *е* она распалась на две конечные компоненты.

Рассмотрим более подробно еще один пример. Выполним строгое полиномиальное отделение точек двух видов, представленных на рис. 2. Координаты точек не превосходят по модулю единицы. С помощью полиномов первой, второй и третьей степени строгое отделение получить не удалось. Эту задачу решает полином четвертой степени. В результате приходим к кривой, изображенной на рис. 3 тонкой линией.

Разделяющий полином можно упростить. Выделим в его представлении базисную функцию, соответствующую наименьшему по модулю коэффициенту. После этого выполним строгое полиномиальное отделение с помощью полинома, у которого отсутствует выделенная базисная функция. Допустим, что отделение прошло успешно. Тогда данную процедуру можно повторить. Уменьшение количества базисных функций можно проводить, пока очередной полином реализует строгое отделение.

В рассматриваемом примере итоговый полином имеет вид

$$P(w_*, x) = -19366 x_1^4 - 49259 x_2^4 + 608 x_1^3 x_2 - 25706 x_1 x_2^3 - 883 x_1 x_2^2 + 4173 x_1^2 + 10383 x_2^2.$$

Соответствующая ему разделяющая кривая изображена на рис. 3 жирной линией. Количество коэффициентов сократилось с 15 до 7.

4. Приложение к бинарной классификации данных. Рассмотрим вопрос о применении строгого полиномиального отделения к бинарной классификации данных. Напомним относящиеся к этой теме понятия.

Пусть X — множество описаний объектов и $Y = \{-1, 1\}$. Имеется конечная обучающая выборка $C^r = \{(x_1, y_1), \dots, (x_r, y_r)\}$, где все x_k принадлежат X и все y_k принадлежат Y . Требуется построить алгоритм $G : X \rightarrow Y$, способный на основе информации об обучающей выборке C^r классифицировать произвольный объект $x \in X$.

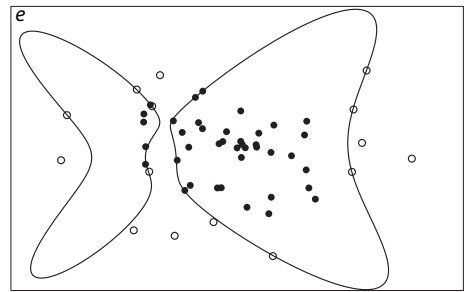
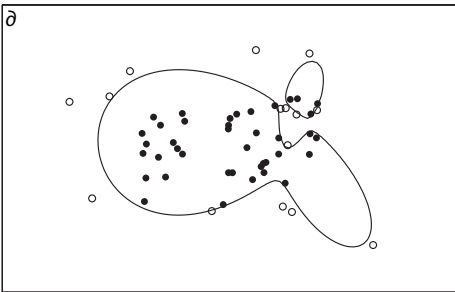
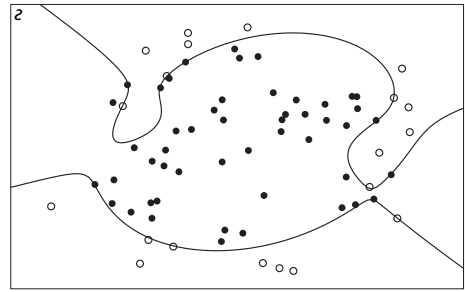
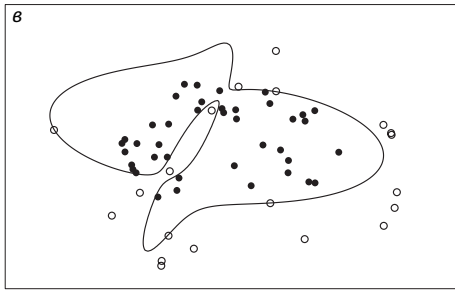
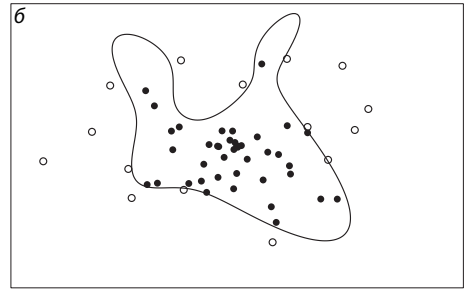
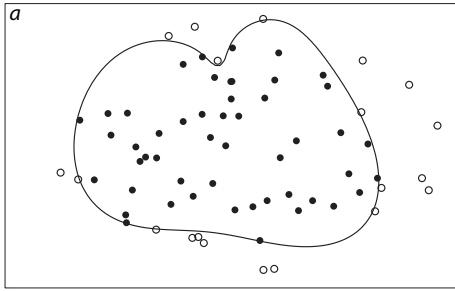


Рис. 1. Строгое полиномиальное отделение на плоскости.

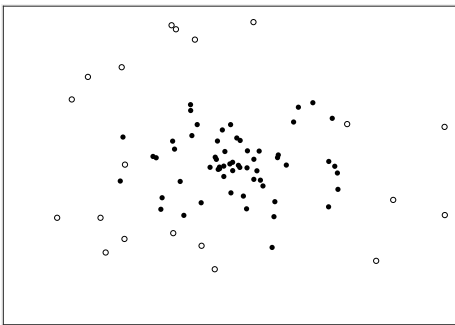


Рис. 2. Исходные множества.

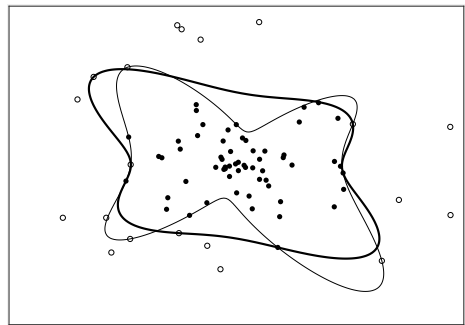


Рис. 3. Две отделяющие линии.

Точность алгоритмов бинарной классификации оценивается следующим образом. Из обучающей выборки исключим некоторый набор объектов. Исключенную часть назовем тестовой выборкой. Построим алгоритм G по урезанной обучающей выборке и применим его к объектам тестовой. Процент объектов, для которых клас-

совая принадлежность была предсказана верно, считают точностью алгоритма G . Очевидно, что при таком определении точности ее значение сильно зависит от конкретной тестовой выборки. Для получения более объективного результата этот эксперимент проводится несколько раз с разными тестовыми выборками, а точностью алгоритма считается среднее арифметическое вычисленных значений точности. Метод кросс-валидации [3, с. 241–249] является наиболее распространенным подходом к такой тестовой процедуре.

В подавляющем большинстве случаев все рассматриваемые объекты можно представить в виде векторов пространства \mathbb{R}^n . По этой причине далее будем считать, что $X = \mathbb{R}^n$.

Перейдем к построению алгоритма бинарной классификации на основе строгого полиномиального отделения. Пусть множества

$$A = \{a_i\}_{i=1}^m \text{ и } B = \{b_j\}_{j=1}^k$$

являются объектами обучающей выборки первого ($y = 1$) и второго ($y = -1$) классов соответственно. Решая задачу линейного программирования (10), произведем строгое полиномиальное отделение множеств A и B . Если отделение возможно ($t_* = 0$), то положим

$$G(x) = \text{sign } P(w_*, x) \quad \forall x \in \mathbb{R}^n.$$

Если $G(x) = 1$, то объект x принадлежит первому классу, если $G(x) = -1$, то объект x принадлежит второму классу. При $G(x) = 0$ будем считать, что классифицировать объект x не удалось.

Одним из наиболее популярных алгоритмов бинарной классификации является метод опорных векторов (SVM) [3, с. 417–438]. Проведем сравнение точности и скорости работы предложенного в статье алгоритма с тремя наиболее распространенными вариантами SVM: linear, polynomial, radial [3, с. 424]. Для тестирования было выбрано пять тестовых задач из репозитория UCI Machine Learning Repository [4]: *Flowmeters*, *Mesothelioma*, *Ionosphere*, *Wisconsin*, *Tic-Tac-Toe*. Реализация SVM была взята из библиотеки LIBSVM [5], а алгоритм строгого полиномиального отделения был реализован в среде MATLAB. В качестве разделяющего полинома P для каждой задачи использовался алгебраический полином второй степени. В табл. 1 приведены полученные результаты точности и времени работы алгоритмов при десятиблочной кросс-валидации данных.

Также была рассмотрена задача *Wilt* из того же репозитория тестовых задач. В отличие от рассмотренных ранее задач, для задачи *Wilt* была известна наряду с обучающей некоторая тестовая выборка. Строгое полиномиальное отделение обучающей выборки с помощью алгебраического полинома второй степени оказалось невозможным. В связи с этим, отделение проводилось алгебраическим полиномом третьей степени. В табл. 2 представлены полученные результаты. В первом столбце приведено количество объектов обучающей выборки, а во втором — количество объектов тестовой выборки.

Стоит отметить, что скорость и качество работы алгоритма строгого полиномиального отделения напрямую зависит от метода решения задачи линейного программирования. В нашей реализации использовалась встроенная в MATLAB функция `linprog`. В версии MATLAB R2018a она по умолчанию ориентирована на двойственный симплекс-метод.

Таблица 1. Точность и время работы алгоритмов при десятиблочной кросс-валидации выбранных баз данных

База данных ($m + k$) \times n	Точность (%)			
	Полиномиальное отделение	SVM (Linear)	SVM (Polynomial)	SVM (Radial)
<i>Floumeters</i> 87 \times 36	84.15 0.39	59.07 3.32	61.63 6.20	58.00 0.01
<i>Mesothelioma</i> 324 \times 34	99.14 0.37	87.89 36.72	83.78 68.67	70.38 0.15
<i>Ionosphere</i> 351 \times 34	74.79 1.07	86.84 0.05	64.10 0.07	92.76 0.04
<i>Wisconsin</i> 569 \times 30	95.55 1.16	95.19 6.95	65.77 32.14	62.74 0.25
<i>Tic-Tac-Toe</i> 958 \times 9	96.69 0.77	65.35 0.13	91.03 0.39	87.83 0.19

Таблица 2. Точность и время работы на базе данных *Wilt*

# Train	# Test	n	Полиномиальное отделение	SVM (Linear)	SVM (Polynomial)	SVM (Radial)
4339	500	5	85.80 1.15	71.80 1.04	77.80 4.88	62.80 1.39

5. Заключение. В статье рассматривается строгое полиномиальное отделение двух конечных множеств в евклидовом пространстве, включающее строгое линейное отделение и являющееся его естественным обобщением на нелинейный случай. Строгое полиномиальное отделение сведено к простой задаче линейного программирования со следующей особенностью: оптимальное значение ее целевой функции может равняться только нулю или единице — нулю, когда отделение возможно, и единице в противном случае.

Приведены примеры строгого полиномиального отделения на плоскости с помощью алгебраических полиномов четвертой степени от двух переменных. Они показывают, что отделяющая кривая может распадаться на компоненты, ограниченные и неограниченные.

На примере известных баз данных сравнивалась эффективность строгого полиномиального отделения с эффективностью трех вариантов SVM-метода. Подтверждена конкурентоспособность предложенного метода строгого отделения.

Литература

1. Bennett K. P., Mangasarian O. L. Robust linear programming discrimination of two linearly inseparable sets // Optimization Methods and Software. 1992. Vol. 1. P. 23–34.
2. Гавурин М. К., Малоземов В. Н. Экстремальные задачи с линейными ограничениями. Л.: Изд-во ЛГУ, 1984. 176 с.
3. Hastie T., Tibshirani R., Friedman J. H. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. New York: Springer, 2009.
4. Dua D., Karra Taniskidou E. UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science. Available at: <http://archive.ics.uci.edu/ml> (accessed December 20, 2018).
5. Chih-Chung C., Chih-Jen L. LIBSVM: a library for support vector machines // ACM Transactions on Intelligent Systems and Technology. 2011. Vol. 2(3). P. 27:1–27:27. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (accessed December 20, 2018).

Статья поступила в редакцию 26 октября 2018 г.;
после доработки 25 ноября 2018 г.;
рекомендована в печать 20 декабря 2018 г.

Контактная информация:

Малоземов Василий Николаевич — д-р физ.-мат. наук, проф.; v.malozemov@spbu.ru
Плоткин Артем Владимирович — студент; avplotkin@gmail.com

Strict polynomial separation of two sets

V. N. Malozemov, A. V. Plotkin

St. Petersburg State University, Universitetskaya nab., 7–9, St. Petersburg, 199034, Russian Federation

For citation: Malozemov V. N., Plotkin A. V. Strict polynomial separation of two sets. *Vestnik of Saint Petersburg University. Mathematics. Mechanics. Astronomy*, 2019, vol. 6 (64), issue 2, pp. 232–240. <https://doi.org/10.21638/11701/spbu01.2019.205> (In Russian)

One of the main tasks of mathematical diagnostics is the strict separation of two finite sets in Euclidean space. A *strict linear* separation is widely known and is reduced to a linear programming problem. We introduce a *strict polynomial* separation and show that the strict polynomial separation of two sets also is reduced to a linear programming problem. The objective function of the linear programming problem proposed in this paper has the following feature: its optimal value can only be zero or one. It is zero if the sets are strictly polynomially separable; otherwise, it is one. We give illustrative examples of strict separation of two sets with the help of two-variable fourth-degree algebraic polynomials on the plane. We also analyze the efficiency of applying the strict polynomial separation to binary classification problems.

Keywords: strict linear separation, strict polynomial separation, linear programming, mathematical diagnostics.

References

1. Bennett K. P., Mangasarian O. L., “Robust linear programming discrimination of two linearly inseparable sets”, *Optimization Methods and Software* **1**, 23–34 (1992).
2. Gavurin M. K., Malozemov V. N., *Extremal Problems with Linear Constraints* (Leningrad, 1984). (In Russian)
3. Hastie T., Tibshirani R., Friedman J. H., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Springer, New York, 2009).
4. Dua D., Karra Taniskidou E., *UCI Machine Learning Repository* (Irvine, CA, University of California, School of Information and Computer Science). Available at: <http://archive.ics.uci.edu/ml> (accessed December 20, 2018).
5. Chih-Chung C., Chih-Jen L., “LIBSVM: a library for support vector machines”, *ACM Transactions on Intelligent Systems and Technology* **2**(3), 27:1–27:27 (2011). Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (accessed December 20, 2018).

Received: October 26, 2018
Revised: November 25, 2018
Accepted: December 20, 2018

Author’s information:

Vassili N. Malozemov — v.malozemov@spbu.ru
Artem V. Plotkin — avplotkin@gmail.com