

В. Н. Малоземов, Г. Ш. Тамасян

СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ КОЗИНЦА, MDM И SMO РЕШЕНИЯ ЗАДАЧИ ЖЕСТКОГО SVM-ОТДЕЛЕНИЯ

В статье приводится сравнительный анализ трех родственных алгоритмов решения задачи жесткого SVM-отделения двух конечных множеств в евклидовом пространстве. Имеются в виду алгоритмы Козинца, MDM и SMO. Единый подход к анализу этих алгоритмов оказался возможным благодаря тому, что были введены *оценки планов* рассматриваемых экстремальных задач. Оценка плана всегда неотрицательна и обращается в ноль тогда и только тогда, когда план оптимальный. Положительная оценка позволяет улучшить план. Это служит основой для построения минимизирующей последовательности планов.

В статье предложены и сравниваются *рабочие* схемы алгоритмов, более эффективные, чем исходные (принципиальные) схемы. Все необходимые теоретические результаты приведены в десяти докладах, прочитанных на семинаре “O&ML”, и в библиографии к этим докладам.

§1. ЖЕСТКОЕ SVM-ОТДЕЛЕНИЕ [1, 2, 3].

Пусть в пространстве \mathbb{R}^n с евклидовой нормой заданы два конечных множества, состоящие из попарно различных точек,

$$P_1 = \{p_j\}_{j=1}^s \quad \text{и} \quad P_2 = \{p_j\}_{j=s+1}^m.$$

Здесь $s \in 1 : (m - 1)$. Будем считать, что выпуклые оболочки C_1 и C_2 этих множеств не имеют общих точек. Задача жесткого SVM-отделения множеств P_1 и P_2 ставится так:

построить гиперплоскость, разделяющую множества P_1 и P_2 , при условии, что расстояние от этой гиперплоскости до объединения $P_1 \cup P_2$ максимально.

Ключевые слова: квадратичное программирование, оценка плана, жесткое SVM-отделение, алгоритм Козинца, MDM-алгоритм, SMO-алгоритм.

Результаты параграфов 6 и 7 получены в Институте проблем машиноведения РАН при финансовой поддержке Российского научного фонда (проект No. 23-41-00060).

Данную задачу можно формализовать следующим образом:

$$\frac{1}{2}\|x - y\|^2 \rightarrow \min_{x \in C_1, y \in C_2}. \quad (1)$$

Очевидно, что задача (1) имеет решение (x^*, y^*) , вообще говоря, не единственное. Но единственным будет вектор $w^* = x^* - y^*$. Так как $C_1 \cap C_2 = \emptyset$, то $w^* \neq \mathbf{0}$.

Обозначим через c середину отрезка $[x^*, y^*]$, $c = \frac{1}{2}(x^* + y^*)$. Уравнение

$$\langle w^*, x - c \rangle = 0$$

определяет искомую гиперплоскость, разделяющую множества P_1 и P_2 .

§2. АЛГОРИТМ КОЗИНЦА [4, 5].

Алгоритм Козинца ориентирован на решение задачи (1). Вводится следующая оценка $\Delta(x, y)$ плана (x, y) задачи (1):

$$\Delta(x, y) = \max\{\Delta^{(1)}(x, y), \Delta^{(2)}(x, y)\},$$

где частные оценки $\Delta^{(1)}(x, y)$ и $\Delta^{(2)}(x, y)$ определяются так:

$$\Delta^{(1)}(x, y) = \max_{j \in 1:s} \langle p_j - x, y - x \rangle,$$

$$\Delta^{(2)}(x, y) = \max_{\ell \in (s+1):m} \langle p_\ell - y, x - y \rangle.$$

Показывается, что оценки $\Delta^{(1)}(x, y)$, $\Delta^{(2)}(x, y)$ и $\Delta(x, y)$ неотрицательны на всех планах (x, y) задачи (1) и что $\Delta(x, y) = 0$ тогда и только тогда, когда (x, y) — оптимальный план.

Опишем рабочую схему алгоритма Козинца.

В качестве начального приближения берем произвольный план (x_0, y_0) задачи (1).

Пусть уже имеется k -е приближение (x_k, y_k) , где $x_k \in C_1$, $y_k \in C_2$. Вычислим частную оценку

$$\Delta^{(1)}(x_k, y_k) := \max_{j \in 1:s} \langle p_j - x_k, y_k - x_k \rangle = \langle p_{j_k} - x_k, y_k - x_k \rangle.$$

- Если $\Delta^{(1)}(x_k, y_k) > 0$, то находим $x_{k+1} = x_k + \lambda_k(p_{j_k} - x_k)$, где

$$\lambda_k = \min \left\{ 1, \frac{\Delta^{(1)}(x_k, y_k)}{\|p_{j_k} - x_k\|^2} \right\}.$$

- Если $\Delta^{(1)}(x_k, y_k) = 0$, то полагаем $x_{k+1} = x_k$.

Далее вычисляем вторую частную оценку на промежуточном плане (x_{k+1}, y_k) :

$$\Delta^{(2)}(x_{k+1}, y_k) := \max_{\ell \in (s+1):m} \langle p_\ell - y_k, x_{k+1} - y_k \rangle = \langle p_{\ell_k} - y_k, x_{k+1} - y_k \rangle.$$

- Если $\Delta^{(2)}(x_{k+1}, y_k) > 0$, то находим $y_{k+1} = y_k + \lambda_k(p_{\ell_k} - y_k)$, где

$$\lambda_k = \min \left\{ 1, \frac{\Delta^{(2)}(x_{k+1}, y_k)}{\|p_{\ell_k} - y_k\|^2} \right\}.$$

- Если $\Delta^{(2)}(x_{k+1}, y_k) = 0$, то полагаем $y_{k+1} = y_k$.

При выполнении условий $\Delta^{(1)}(x_k, y_k) = 0$, $\Delta^{(2)}(x_{k+1}, y_k) = 0$ план (x_k, y_k) будет оптимальным. Иначе (x_{k+1}, y_{k+1}) — очередное приближение с меньшим значением целевой функции.

Описание рабочей схемы алгоритма Козинца завершено.

§3. MDM-АЛГОРИТМ [6, 7].

Обозначим через ξ вектор из \mathbb{R}^m с компонентами $\xi_j = 1$ при $j \in 1 : s$, $\xi_j = -1$ при $j \in (s+1) : m$ и через A — матрицу со столбцами $\xi_1 p_1, \xi_2 p_2, \dots, \xi_m p_m$. Пусть $x \in C_1$, $y \in C_2$. Согласно определению выпуклой оболочки конечного множества имеем

$$x - y = \sum_{j=1}^s u[j] p_j - \sum_{j=s+1}^m u[j] p_j = \sum_{j=1}^m \xi_j u[j] p_j = Au, \quad (2)$$

где вектор u принадлежит \mathbb{R}^m и удовлетворяет условиям

$$\begin{aligned} \sum_{j=1}^s u[j] &= 1, & \sum_{j=s+1}^m u[j] &= 1, \\ u[j] &\geq 0 \text{ при всех } j \in 1 : m. \end{aligned} \quad (3)$$

При этом

$$\frac{1}{2} \|x - y\|^2 = \frac{1}{2} \|Au\|^2 = \frac{1}{2} \langle A^T Au, u \rangle.$$

Множество векторов u из \mathbb{R}^m , удовлетворяющих условиям (3), обозначим Ω . Теперь задачу жесткого SVM-отделения множеств P_1 и P_2 можно записать в развернутом виде

$$\frac{1}{2} \langle A^T Au, u \rangle \rightarrow \min_{u \in \Omega}. \quad (4)$$

Очевидно, что задача (4) имеет решение u^* . По u^* в силу (2) восстанавливается решение задачи (1):

$$x^* = \sum_{j=1}^s u^*[j]p_j, \quad y^* = \sum_{j=s+1}^m u^*[j]p_j.$$

MDM-алгоритм ориентирован на решение задачи (4).

Пусть u — произвольный план задачи (4), $w = Au$ и

$$M_1^+(u) = \{j \in 1 : s \mid u[j] > 0\}, \quad M_2^+(u) = \{\ell \in (s+1) : m \mid u[\ell] > 0\}.$$

Введем оценку $\Delta(u)$ плана u :

$$\Delta(u) = \max\{\Delta^{(1)}(u), \Delta^{(2)}(u)\},$$

где частные оценки $\Delta^{(1)}(u)$ и $\Delta^{(2)}(u)$ определяются следующим образом:

$$\begin{aligned} \Delta^{(1)}(u) &= \max_{j \in M_1^+(u)} \langle p_j, w \rangle - \min_{j \in 1:s} \langle p_j, w \rangle, \\ \Delta^{(2)}(u) &= \max_{\ell \in M_2^+(u)} \langle p_\ell, -w \rangle - \min_{\ell \in (s+1):m} \langle p_\ell, -w \rangle. \end{aligned}$$

Показывается, что оценки $\Delta^{(1)}(u)$, $\Delta^{(2)}(u)$ и $\Delta(u)$ на всех планах задачи (4) неотрицательны и что $\Delta(u) = 0$ тогда и только тогда, когда u — оптимальный план.

Опишем рабочую схему MDM-алгоритма.

В качестве начального приближения возьмем произвольный план u_0 задачи (4). Вычислим $w_0 = Au_0$.

Пусть уже имеются k -е приближение u_k и вектор $w_k := Au_k$. Найдем частную оценку плана u_k :

$$\Delta^{(1)}(u_k) = \max_{j \in M_1^+(u_k)} \langle p_j, w_k \rangle - \min_{j \in 1:s} \langle p_j, w_k \rangle = \langle p_{j'_k} - p_{j''_k}, w_k \rangle.$$

Здесь $j'_k \in M_1^+(u_k)$, $j''_k \in 1 : s$.

- Если $\Delta^{(1)}(u_k) > 0$, то введем промежуточный план \tilde{u}_k с компонентами

$$\tilde{u}_k[j] = \begin{cases} u_k[j'_k] - \lambda_k, & \text{если } j = j'_k, \\ u_k[j''_k] + \lambda_k, & \text{если } j = j''_k, \\ u_k[j], & \text{при остальных } j \in 1 : m, \end{cases}$$

где

$$\lambda_k = \min \left\{ u_k[j'_k], \frac{\Delta^{(1)}(u_k)}{\|p_{j'_k} - p_{j''_k}\|^2} \right\}.$$

С \tilde{u}_k связан вектор $\tilde{w}_k := A\tilde{u}_k = w_k - \lambda_k(p_{j'_k} - p_{j''_k})$.

- Если $\Delta^{(1)}(u_k) = 0$, то полагаем $\tilde{u}_k = u_k$ и $\tilde{w}_k = w_k$.

Далее вычисляем вторую частную оценку плана \tilde{u}_k :

$$\Delta^{(2)}(\tilde{u}_k) = \max_{\ell \in M_2^+(\tilde{u}_k)} \langle p_\ell, -\tilde{w}_k \rangle - \min_{\ell \in (s+1):m} \langle p_\ell, -\tilde{w}_k \rangle = \langle p_{\ell'_k} - p_{\ell''_k}, -\tilde{w}_k \rangle.$$

Здесь $\ell'_k \in M_2^+(\tilde{u}_k)$, $\ell''_k \in (s+1) : m$.

- Если $\Delta^{(2)}(\tilde{u}_k) > 0$, то находим план u_{k+1} с компонентами

$$u_{k+1}[\ell] = \begin{cases} \tilde{u}_k[\ell'_k] - \lambda_k, & \text{если } \ell = \ell'_k, \\ \tilde{u}_k[\ell''_k] + \lambda_k, & \text{если } \ell = \ell''_k, \\ \tilde{u}_k[\ell], & \text{при остальных } \ell \in 1 : m, \end{cases}$$

где

$$\lambda_k = \min \left\{ \tilde{u}_k[\ell'_k], \frac{\Delta^{(2)}(\tilde{u}_k)}{\|p_{\ell'_k} - p_{\ell''_k}\|^2} \right\}.$$

С u_{k+1} связан вектор $w_{k+1} := Au_{k+1} = \tilde{w}_k + \lambda_k(p_{\ell'_k} - p_{\ell''_k})$.

- Если $\Delta^{(2)}(\tilde{u}_k) = 0$, то полагаем $u_{k+1} = \tilde{u}_k$ и $w_{k+1} = \tilde{w}_k$.

При выполнении условий $\Delta^{(1)}(u_k) = 0$, $\Delta^{(2)}(\tilde{u}_k) = 0$ план u_k будет оптимальным. Иначе u_{k+1} — очередное приближение с меньшим значением целевой функции. С u_{k+1} связан вектор w_{k+1} .

Описание рабочей схемы MDM-алгоритма завершено.

§4. SMO-АЛГОРИТМ [8, 9, 10].

Задача жесткого SVM-отделения множеств P_1 и P_2 , наряду с формализацией (1), допускает другую формализацию:

$$\begin{aligned} \frac{1}{2} \|w\|^2 &\rightarrow \min, \\ \langle w, p_j \rangle + \beta &\geq 1, \quad j \in 1 : s, \\ \langle w, p_j \rangle + \beta &\leq -1, \quad j \in (s+1) : m. \end{aligned} \tag{5}$$

Ограничения задачи (5) имеют нестандартный вид, однако к нему сводится естественная пара ограничений

$$\begin{aligned}\langle w, p_j \rangle &\geq a, & j \in 1 : s, \\ \langle w, p_j \rangle &\leq b, & j \in (s+1) : m,\end{aligned}$$

где $a > b$. Действительно, достаточно положить $c = -\frac{1}{2}(a+b)$, $d = \frac{1}{2}(a-b)$, переписать данные ограничения в эквивалентном виде

$$\begin{aligned}\langle w, p_j \rangle + c &\geq d, & j \in 1 : s, \\ \langle w, p_j \rangle + c &\leq -d, & j \in (s+1) : m,\end{aligned}$$

и поделить все неравенства на положительное число d .

Задача (5) является задачей выпуклого квадратичного программирования. В силу условия $C_1 \cap C_2 = \emptyset$ и теоремы о строгой отделимости множество её планов непусто. Вместе с ограниченностью снизу целевой функции это гарантирует существование оптимального плана (w°, β°) . Отдельно доказывается его единственность.

Очевидно, что $w^\circ \neq \mathbf{0}$. Уравнение $\langle w^\circ, x \rangle + \beta^\circ = 0$ определяет гиперплоскость, разделяющую множества P_1 и P_2 . При этом ширина нейтральной полосы принимает наибольшее возможное значение, равное $\frac{2}{\|w^\circ\|}$.

Отметим, что ограничения задачи (5) с помощью вектора ξ (см. §3) можно представить в виде одного неравенства

$$\xi_j (\langle w, p_j \rangle + \beta) \geq 1, \quad j \in 1 : m. \quad (6)$$

Запишем задачу, двойственную к задаче (5) с ограничениями в форме (6):

$$\begin{aligned}-\frac{1}{2}\|v\|^2 + \sum_{j=1}^m u[j] &\rightarrow \max, \\ v &= \sum_{j=1}^m u[j] \xi_j p_j,\end{aligned} \quad (7)$$

$$\sum_{j=1}^m u[j] \xi_j = 0; \quad u[j] \geq 0 \text{ при всех } j \in 1 : m.$$

По первой теореме двойственности (в квадратичном программировании) задача (7) также имеет решение (v°, u°) . По второй теореме двойственности справедливы соотношения

$$\begin{aligned} w^\circ &= v^\circ, \\ u^\circ[j] \left(\xi_j (\langle w^\circ, p_j \rangle + \beta^\circ) - 1 \right) &= 0, \quad j \in 1 : m. \end{aligned} \quad (8)$$

В частности,

$$\xi_j (\langle w^\circ, p_j \rangle + \beta^\circ) = 1 \text{ при } j \in J^\circ, \quad (9)$$

где $J^\circ = \{j \in 1 : m \mid u^\circ[j] > 0\}$ – носитель вектора u° . Так как

$$w^\circ = v^\circ = \sum_{j=1}^m u^\circ[j] \xi_j p_j \quad \text{и} \quad w^\circ \neq \mathbf{0},$$

то $u^\circ \neq \mathbf{0}$, так что $J^\circ \neq \emptyset$.

Переменную v в двойственной задаче (7) можно исключить. Воспользуемся тем, что $v = Au$, где, как и раньше, A – матрица со столбцами $\xi_1 p_1, \xi_2 p_2, \dots, \xi_m p_m$. Поменяв знак у целевой функции, придем к вспомогательной задаче квадратичного программирования

$$\begin{aligned} D(u) &:= \frac{1}{2} \langle A^T A u, u \rangle - \sum_{j=1}^m u[j] \rightarrow \min, \\ \sum_{j=1}^m \xi_j u[j] &= 0; \quad u[j] \geq 0 \text{ при всех } j \in 1 : m. \end{aligned} \quad (10)$$

Множество планов задачи (10) обозначим U . Очевидно, что U – выпуклый конус в пространстве \mathbb{R}^m с вершиной в начале координат. Отметим важное для дальнейшего свойство конуса U : при любом ненулевом $u \in U$ вектор $w = Au$ будет ненулевым. Это следует из свойства $C_1 \cap C_2 = \emptyset$.

Пусть u° – решение задачи (10). Тогда (v°, u°) , где $v^\circ = Au^\circ$, – решение задачи (7). С помощью формул (8) и (9) можно получить решение (w°, β°) задачи (5).

Отметим, что решение u° задачи (10) и решение u^* задачи (4) связаны простым соотношением $u^* = \frac{1}{b} u^\circ$, где

$$b = \frac{1}{2} \sum_{j=1}^m u^\circ[j].$$

Это позволяет представить решение (x^*, y^*) задачи (1) в виде

$$x^* = \frac{1}{b} \sum_{j=1}^s u^\circ[j] p_j, \quad y^* = \frac{1}{b} \sum_{j=s+1}^m u^\circ[j] p_j.$$

SMO-алгоритм ориентирован на решение задачи (10).

Возьмем план $u \in U$ задачи (10) и вычислим вектор $w = Au$. Оценкой плана u называется величина

$$\Delta(u) = \max_{j \in J''(u)} \{ \langle w, p_j \rangle - \xi_j \} - \min_{j \in J'(u)} \{ \langle w, p_j \rangle - \xi_j \}, \quad (11)$$

где индексные множества $J'(u)$ и $J''(u)$ определяются так:

$$J'(u) = (1 : s) \cup \{ j \in (s+1) : m \mid u[j] > 0 \}, \\ J''(u) = \{ j \in 1 : s \mid u[j] > 0 \} \cup ((s+1) : m).$$

Показывается, что оценка $\Delta(u)$ неотрицательна на всех планах задачи (10) и что $\Delta(u) = 0$ тогда и только тогда, когда u оптимальный план.

Опишем рабочую схему SMO-алгоритма.

В качестве начального приближения берем произвольный ненулевой план u_0 задачи (10). Вычисляем вектор $w_0 = Au_0$.

Пусть уже имеется k -е приближение $u_k \in U \setminus \{0\}$ и соответствующий ему вектор w_k . Вычислим скаляр $\gamma_k = \frac{1}{\|w_k\|^2} \sum_{j=1}^m u_k[j]$. Определим $\tilde{u}_k = \gamma_k u_k$, $\tilde{w}_k = \gamma_k w_k$. Найдем оценку плана \tilde{u}_k :

$$\Delta(\tilde{u}_k) = \max_{j \in J''(\tilde{u}_k)} \{ \langle \tilde{w}_k, p_j \rangle - \xi_j \} - \min_{j \in J'(\tilde{u}_k)} \{ \langle \tilde{w}_k, p_j \rangle - \xi_j \} = \\ = \langle \tilde{w}_k, p_{j_k''} - p_{j_k'} \rangle - (\xi_{j_k''} - \xi_{j_k'}).$$

Здесь $j_k' \in J'(\tilde{u}_k)$, $j_k'' \in J''(\tilde{u}_k)$.

- Если $\Delta(\tilde{u}_k) = 0$, то план \tilde{u}_k – решение задачи (10). Вектор \tilde{w}_k известен.
- Если $\Delta(\tilde{u}_k) > 0$, то полагаем $u_{k+1} = \tilde{u}_k + \lambda_k (\xi_{j_k'} e_{j_k'} - \xi_{j_k''} e_{j_k''})$, где e_j – орты в пространстве \mathbb{R}^m . Шаг λ_k определяется в два приема. Вначале находим предварительный шаг $\tilde{\lambda}_k$,

$$\tilde{\lambda}_k = \frac{\Delta(\tilde{u}_k)}{\|p_{j_k'} - p_{j_k''}\|^2},$$

после чего вычисляем λ_k по формуле

$$\lambda_k = \begin{cases} \tilde{\lambda}_k, & \text{если } j'_k \in 1 : s, j''_k \in (s+1) : m; \\ \min\{\tilde{\lambda}_k, \tilde{u}_k[j''_k]\}, & \text{если } j'_k \in 1 : s, j''_k \in 1 : s; \\ \min\{\tilde{\lambda}_k, \tilde{u}_k[j'_k]\}, & \text{если } j'_k \in (s+1) : m, j''_k \in (s+1) : m; \\ \min\{\tilde{\lambda}_k, \tilde{u}_k[j'_k], \tilde{u}_k[j''_k]\}, & \text{если } j'_k \in (s+1) : m, j''_k \in 1 : s. \end{cases}$$

К этому нужно добавить, что

$$w_{k+1} := Au_{k+1} = \tilde{w}_k + \lambda_k(p_{j'_k} - p_{j''_k}).$$

Описание рабочей схемы СМО-алгоритма завершено.

§5. ВЫБОР НАЧАЛЬНОГО ПРИБЛИЖЕНИЯ И ПРАВИЛО ОСТАНОВА.

Дадим предварительный комментарий к приводимому ниже сравнению алгоритмов Козинца, MDM и СМО.

Обозначим через \bar{x} и \bar{y} центры масс множеств P_1 и P_2 , так что

$$\bar{x} = \frac{1}{s} \sum_{j=1}^s p_j, \quad \bar{y} = \frac{1}{m-s} \sum_{j=s+1}^m p_j.$$

Для корректности сравнения во всех случаях в качестве начального приближения бралась пара (\bar{x}, \bar{y}) и вычисления завершались при близости к нулю оценки очередного плана.

В алгоритме Козинца использовалось прямое равенство $(x_0, y_0) = (\bar{x}, \bar{y})$. Вычисления завершались на плане (x_k, y_k) при выполнении условий

$$\Delta^{(1)}(x_k, y_k) < \varepsilon, \quad \Delta^{(2)}(x_{k+1}, y_k) < \varepsilon.$$

Здесь $\varepsilon > 0$ – параметр точности.

В MDM-алгоритме начальный план u_0 должен обеспечивать равенство

$$w_0 := Au_0 = \bar{x} - \bar{y}.$$

Этому условию удовлетворяет план u_0 с компонентами

$$u_0[j] = \begin{cases} \frac{1}{s} & \text{при } j \in 1 : s, \\ \frac{1}{m-s} & \text{при } j \in (s+1) : m. \end{cases}$$

У данного плана все m компонент положительны. Так как у оптимального плана количество положительных компонент имеет порядок n , то желательно, чтобы у начального плана количество положительных

компонент было небольшим. Мы покажем, как построить начальный план с минимальным количеством положительных компонент, равным двум.

Для этого рассмотрим расширенный вариант задачи (4). Множества P_1 и P_2 заменим на P'_1 и P'_2 , положив

$$P'_1 = \{p_1, \dots, p_s, \bar{x}\}, \quad P'_2 = \{p_{s+1}, \dots, p_m, \bar{y}\}.$$

В этом случае матрице A будет соответствовать матрица A' со столбцами

$$\xi_1 p_1, \xi_2 p_2, \dots, \xi_s p_s, \xi_{s+1} \bar{x}, \xi_{s+2} p_{s+1}, \xi_{s+3} p_{s+2}, \dots, \xi_{m+1} p_m, \xi_{m+2} \bar{y},$$

где

$$\xi_j = \begin{cases} 1 & \text{при } j \in 1 : s+1, \\ -1 & \text{при } j \in (s+2) : (m+2). \end{cases}$$

Задача (4) примет вид

$$\begin{aligned} & \frac{1}{2} \langle A' u', A' u' \rangle \rightarrow \min, \\ & \sum_{j=1}^{s+1} u'[j] = 1, \quad \sum_{j=s+2}^{m+2} u'[j] = 1, \\ & u'[j] \geq 0 \text{ при } j \in 1 : (m+2). \end{aligned} \quad (12)$$

Очевидно, что выпуклые оболочки множеств P'_1 и P'_2 совпадают с C_1 и C_2 соответственно. Поэтому решение основной задачи (1) сводится к решению задачи (12). По оптимальному плану задачи (12), так же как по оптимальному плану задачи (4), легко восстанавливается решение задачи (1).

В качестве начального плана для решения задачи (12) можно взять вектор $u'_0 \in \mathbb{R}^{m+2}$ с компонентами

$$u'_0[j] = \begin{cases} 1 & \text{при } j = s+1 \text{ и } j = m+2, \\ 0 & \text{при остальных } j \in 1 : (m+2). \end{cases}$$

В этом случае выполняется равенство $w'_0 := A' u'_0 = \bar{x} - \bar{y}$.

Вычисления при решении задачи (12) МДМ-алгоритмом завершались на плане u'_k при выполнении условий

$$\Delta^{(1)}(u'_k) < \varepsilon, \quad \Delta^{(2)}(\tilde{u}'_k) < \varepsilon.$$

В SMO-алгоритме начальное приближение строилось так же, как в MDM-алгоритме. Вычисления завершались на плане \tilde{u}'_k при выполнении условия $\Delta(\tilde{u}'_k) < \varepsilon$.

§6. ГЕНЕРАЦИЯ ТЕСТОВЫХ ЗАДАЧ.

При генерации тестовых задач используются, наряду с n , s и m , параметры r_1 и r_2 . Они задают количество “граничных точек” в множествах P_1 и P_2 соответственно.

Пусть x^* и y^* случайные не равные между собой точки из \mathbb{R}^n . Поместим точку x^* в множество P_1 , а точку y^* – в P_2 . Нейтральная полоса между множествами P_1 и P_2 будет ограничиваться гиперплоскостями

$$L_1 := \{x \in \mathbb{R}^n \mid \langle w^*, x - x^* \rangle = 0\}, \quad L_2 := \{x \in \mathbb{R}^n \mid \langle w^*, x - y^* \rangle = 0\}.$$

Здесь вектор $w^* = x^* - y^*$ – нормаль оптимальной гиперплоскости.

Точку p из \mathbb{R}^n принадлежащую одной из гиперплоскостей L_1 или L_2 будем называть *граничной*.

Дальнейшее формирование множеств P_1 и P_2 происходит в два этапа. Вначале вводятся граничные точки множеств P_1 и P_2 , а после по необходимости эти множества пополняются другими точками, пока они не будут содержать s и $m - s$ точек соответственно.

Первый этап. При $r_1 > 1$ генерируем оставшиеся $r_1 - 1$ граничных точек множества P_1 . Они вычисляются по формуле

$$p_j = x^* + \left(E - \frac{w^*(w^*)^T}{\langle w^*, w^* \rangle} \right) z_j, \quad j \in 2 : r_1,$$

где E – единичная матрица порядка n , z_j – случайные ненулевые n -мерные векторы.

Аналогично поступаем и при генерации граничных точек множества P_2 . Если $r_2 > 1$, то они вычисляются по формуле

$$p_{s+j} = y^* + \left(E - \frac{w^*(w^*)^T}{\langle w^*, w^* \rangle} \right) z_j, \quad j \in 2 : r_2.$$

Второй этап. Выбираем случайный n -мерный вектор z . Если для него выполняется неравенство

$$\langle w^*, z - x^* \rangle > 0,$$

то помещаем z в множество P_1 при условии, что оно еще не заполнено.

Если для случайной точки z выполняется неравенство

$$\langle w^*, z - y^* \rangle < 0,$$

то её помещаем в множество P_2 при условии, что оно еще не заполнено.

Процесс повторяется пока множества P_1 и P_2 не будут содержать s и $m - s$ точек соответственно.

§7. СРАВНИТЕЛЬНЫЙ АНАЛИЗ РАБОЧИХ СХЕМ АЛГОРИТМОВ КОЗИНЦА, MDM И SMO.

Далее будет приведен сравнительный анализ описанных выше алгоритмов решения задачи жесткого SVM-отделения. Алгоритмы были реализованы в среде MatLab 2022b на ПК Intel Core i5-10400 CPU 2.90GHz ОЗУ 16Гб. Замерялось время, затраченное каждой схемой на решение одинаковых задач. В каждой серии экспериментов брались по 2000 задач.

Пример 1. Приведем результаты расчетов при $s = 4000$, $m - s = 6000$, $r_1 = r_2 = 1$ (см. таблицы 1 и 2).

$\varepsilon \backslash n$	2	50	100	250	500	750	1000
10^{-2}	1.7	14.3	9.8	11.5	12.9	13.4	13.7

Таблица 1. Отношение медианного значения времени решения задачи алгоритмом Козинца к MDM-алгоритму.

$\varepsilon \backslash n$	2	50	100	250	500	750	1000
10^{-2}	55.3	7.3	4.9	6.4	7.3	7.5	7.8

Таблица 2. Отношение медианного значения времени решения задачи SMO-алгоритмом к MDM-алгоритму.

Таким образом, в наиболее простом случае $r_1 = r_2 = 1$, что соответствует минимальному количеству опорных точек, MDM-алгоритм имеет устойчивое преимущество по сравнению с алгоритма Козинца и SMO.

$\varepsilon \backslash n$	2	50	100	250	500	750	1000
10^{-1}	1.1	6.3	4.7	6.0	5.6	5.7	7.0
10^{-2}	55.4	6.5	4.8	6.1	7.0	7.3	7.5
10^{-3}	1043	385	141	7.5	7.1	7.4	8.5

Таблица 3. Отношение медианного значения времени решения задачи алгоритмом Козинца к MDM-алгоритму.

$\varepsilon \backslash n$	2	50	100	250	500	750	1000
10^{-1}	1.8	12.5	9.3	11.0	10.7	10.6	12.6
10^{-2}	1.7	12.7	9.4	11.0	12.5	13.0	13.2
10^{-3}	1.6	12.8	9.5	10.3	12.6	13.1	13.2

Таблица 4. Отношение медианного значения времени решения задачи SMO-алгоритмом к MDM-алгоритму.

Пример 2. Приведем результаты расчетов при $s = 4000$, $m - s = 6000$, $r_1 = 1$, $r_2 = n$ (см. таблицы 3 и 4).

Получили, что и в случае $r_1 = 1$, $r_2 = n$ MDM-алгоритм имеет преимущество перед алгоритмами Козинца и SMO.

ПРИЛОЖЕНИЕ.

ОСОБЕННОСТЬ РАБОЧЕЙ СХЕМЫ SMO-АЛГОРИТМА

На каждой итерации SMO-алгоритма производится минимизация целевой функции на луче. Поясним это. Напомним, что SMO-алгоритм используется для решения задачи (10).

Перед началом k -й итерации имеются ненулевой план u_k и ненулевой вектор $w_k = Au_k$. Первым действием вычисляется план $\tilde{u}_k = \gamma_k u_k$, где

$$\gamma_k = \frac{1}{\|w_k\|^2} \sum_{j=1}^m u_k[j]. \quad (\text{П.1})$$

Покажем, что γ_k – решение следующей экстремальной задачи:

$$\varphi(\gamma) := D(\gamma u_k) \rightarrow \min_{\gamma > 0}.$$

Имеем

$$\varphi(\gamma) = \frac{1}{2}\gamma^2 \|w_k\|^2 - \gamma \sum_{j=1}^m u_k[j].$$

Видим, что функция $\varphi(\gamma)$ является квадратным трехчленом с положительным старшим коэффициентом. Для его точки минимума γ_k справедлива формула (П.1). При этом $\gamma_k > 0$.

Полученный результат можно сформулировать иначе: на плане \tilde{u}_k достигается минимальное значение функции $D(u)$ при условии, что u принадлежит лучу $\Gamma_k = \{u = \gamma u_k \mid \gamma > 0\}$.

Отметим два свойства плана \tilde{u}_k :

$$D(\tilde{u}_k) = -\frac{1}{2} \sum_{j=1}^m \tilde{u}_k[j], \quad (\text{П.2})$$

$$D(\tilde{u}_k) - D(u_k) = -\frac{1}{2} \left(1 - \frac{1}{\gamma_k}\right)^2 \sum_{j=1}^m \tilde{u}_k[j]. \quad (\text{П.3})$$

Приведем их простое обоснование.

Согласно определению γ_k имеем

$$\gamma_k \|w_k\|^2 = \sum_{j=1}^m u_k[j]. \quad (\text{П.4})$$

Отсюда следует, что

$$\|A\tilde{u}_k\|^2 = \gamma_k (\gamma_k \|A u_k\|^2) = \gamma_k \sum_{j=1}^m u_k[j] = \sum_{j=1}^m \tilde{u}_k[j].$$

Значит,

$$D(\tilde{u}_k) := \frac{1}{2} \|A\tilde{u}_k\|^2 - \sum_{j=1}^m \tilde{u}_k[j] = -\frac{1}{2} \sum_{j=1}^m \tilde{u}_k[j].$$

Далее, ещё раз воспользуемся формулой (П.4). Запишем

$$D(u_k) = \frac{1}{2\gamma_k} \sum_{j=1}^m u_k[j] - \sum_{j=1}^m u_k[j] = \left(\frac{1}{2\gamma_k} - 1\right) \sum_{j=1}^m u_k[j].$$

Получаем

$$\begin{aligned} D(\tilde{u}_k) - D(u_k) &= \left(-\frac{1}{2} - \frac{1}{\gamma_k} \left(\frac{1}{2\gamma_k} - 1 \right) \right) \sum_{j=1}^m \tilde{u}_k[j] = \\ &= -\frac{1}{2\gamma_k^2} (\gamma_k - 1)^2 \sum_{j=1}^m \tilde{u}_k[j] = -\frac{1}{2} \left(1 - \frac{1}{\gamma_k} \right)^2 \sum_{j=1}^m \tilde{u}_k[j]. \end{aligned}$$

Формулы (П.2) и (П.3) установлены. Из них следует, что при всех k значение $D(\tilde{u}_k)$ отрицательно и что при переходе от u_k к \tilde{u}_k целевая функция $D(u)$ либо строго убывает, либо не меняется (при $\gamma_k = 1$ будет $\tilde{u}_k = u_k$).

Приведем два простых примера на применение рабочей схемы SMO-алгоритма.

Пример П. 1. Рассмотрим два множества на плоскости

$$P_1 = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix} \right\}, \quad P_2 = \left\{ \begin{pmatrix} 3 \\ 0 \end{pmatrix} \right\}.$$

Очевидно, что решением задачи жесткого SVM-отделения в данном случае является пара точек $x^* = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ и $y^* = \begin{pmatrix} 3 \\ 0 \end{pmatrix}$ (см. рис. 1).



Рис. 1. Множества P_1 и P_2 .

В качестве начального приближения для SMO-алгоритма возьмем, согласно рекомендации, план $u_0 = (\frac{1}{2}, \frac{1}{2}, 1)^T$. Ему соответствует вектор $w_0 = Au_0 = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$. Отметим, что $w_0 = x^* - y^*$. Вместе с тем, план u_0 не будет оптимальным. Проверим это.

Имеем $J'(u_0) = J''(u_0) = \{1, 2, 3\}$. Значит

$$\Delta(u_0) = \max\{-1, -1, -8\} - \min\{-1, -1, -8\} = 7,$$

то есть оценка плана u_0 положительна.

Теперь вычислим оценку плана $\tilde{u}_0 = \gamma_0 u_0$, где

$$\gamma_0 = \frac{1}{\|w_0\|^2} \sum_{i=1}^m u_0[i] = \frac{2}{9}.$$

Имеем

$$\tilde{u}_0 = \frac{1}{9}(1, 1, 2)^T, \quad \tilde{w}_0 = \gamma_0 w_0 = \begin{pmatrix} -2/3 \\ 0 \end{pmatrix},$$

$$\Delta(\tilde{u}_0) = \max\{-1, -1, -1\} - \min\{-1, -1, -1\} = 0.$$

Оценка плана \tilde{u}_0 равна нулю. Значит, он оптимальный.

Пример II. 2. Рассмотрим два множества на плоскости

$$P_1 = \left\{ \begin{pmatrix} -1 \\ 2 \end{pmatrix}, \begin{pmatrix} 0 \\ -3 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\}, \quad P_2 = \left\{ \begin{pmatrix} 3 \\ -2 \end{pmatrix} \right\}.$$

Для их жесткого SVM-отделения воспользуемся SMO-алгоритмом. В качестве начального приближения возьмем план $u_0 = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 1)^T$. При $\varepsilon = 0.01$ вычисления завершились на 5-м приближении. Траектория поиска представлена на рис. 2.

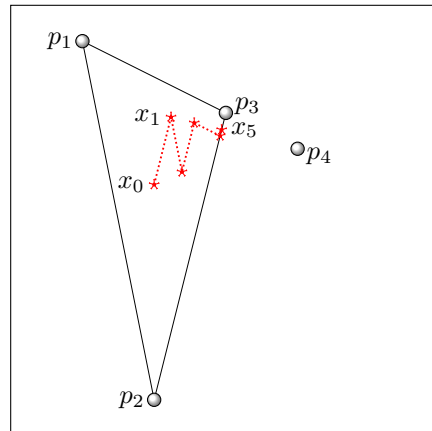
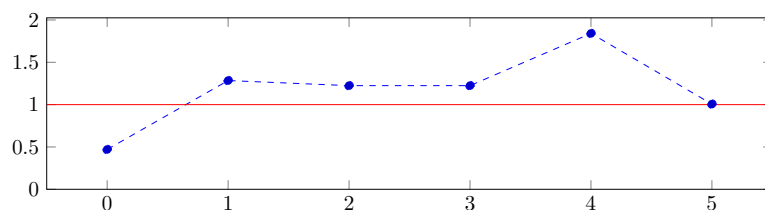


Рис. 2. Траектория поиска.

На рис. 3 изображен график изменения параметра γ_k .

Рис. 3. График изменения параметра γ_k .

СПИСОК ЛИТЕРАТУРЫ

1. В. Н. Малоземов, А. В. Плоткин, *SVM-метод строгого линейного отделения двух конечных множеств*. — Семинар “O&ML”. Избранные доклады. 30 марта 2022 г. (<http://oml.cmlaboratory.com/rep22.shtml#0330>)
2. В. Н. Малоземов, А. В. Плоткин, *SVM-метод: мягкое отделение*. — Семинар “O&ML”. Избранные доклады. 6 апреля 2022 г. (<http://oml.cmlaboratory.com/rep22.shtml#0406>)
3. В. Н. Малоземов, А. В. Плоткин, *Строгое линейное отделение двух конечных множеств и линейное программирование*. — Семинар “O&ML”. Избранные доклады. 24 февраля 2022 г. (<http://oml.cmlaboratory.com/rep22.shtml#0224>)
4. В. Н. Малозёмов, Н. А. Соловьёва, Г. Ш. Тамасян, *Об алгоритме Козинца. I*. — Семинар “O&ML”. Избранные доклады. 14 марта 2024 г. (<http://oml.cmlaboratory.com/rep24.shtml#0314>)
5. В. Н. Малозёмов, Н. А. Соловьёва, Г. Ш. Тамасян, *Об алгоритме Козинца. II*. — Семинар “&ML”. Избранные доклады. 21 марта 2024 г. (<http://oml.cmlaboratory.com/rep24.shtml#0321>)
6. В. Н. Малозёмов, Н. А. Соловьёва, *Общая квадратичная задача математической диагностики*. — Семинар “O&ML”. Избранные доклады. 11 мая 2022 г. (<http://oml.cmlaboratory.com/rep22.shtml#0511>)
7. В. Н. Малозёмов, Н. А. Соловьёва, *МДМ-метод для решения общей квадратичной задачи математической диагностики*. — Семинар “O&ML”. Избранные доклады. 1 июня 2022 г. (<http://oml.cmlaboratory.com/rep22.shtml#0601>)
8. В. Н. Малозёмов, Н. А. Соловьёва, *SMO-алгоритм как обобщение MDM-алгоритма*. — Семинар “O&ML”. Избранные доклады. 16 февраля 2023 г. (<http://oml.cmlaboratory.com/rep23.shtml#0216>)
9. В. Н. Малозёмов, Н. А. Соловьёва, *Картина маслом*. — Семинар “O&ML”. Избранные доклады. 11 мая 2023 г. (<http://oml.cmlaboratory.com/rep23.shtml#0511>)
10. В. Н. Малозёмов, Н. А. Соловьёва, *Сильная сходимость SMO-алгоритма*. — Семинар “O&ML”. Избранные доклады. 30 ноября 2023 г. (<http://oml.cmlaboratory.com/rep23.shtml#1130>)

Malozemov V. N., Tamasyan G. Sh. Comparative analysis of Kozinec, MDM and SMO algorithms for solving the hard SVM separation problem.

The paper provides a comparative analysis of three related algorithms for solving the problem of hard SVM separation of two finite sets in a Euclidean space. These algorithms are the Kozinec, MDM, and SMO algorithms. It is possible to elaborate a unified approach to the analysis of these algorithms due to the fact that “estimates of plans” for the considered extremal problems have been introduced. The estimate of a plan is always non-negative, and it vanishes if and only if the plan is optimal. A positive estimate allows us to improve the plan. This serves as a basis for constructing a minimizing sequence of plans.

The paper proposes and compares “working” schemes of algorithms that are more efficient than the original (principal) schemes. All the necessary theoretical results were presented in ten reports of the “O&ML” seminar and in the bibliographies to these reports.

С.-Петербургский
государственный университет
С.-Петербург, Россия
E-mail: v.malozemov@spbu.ru

Поступило 21 августа 2024 г.

Военно-космическая академия
им. А. Ф. Можайского; Институт проблем машиноведения РАН
С.-Петербург, Россия
E-mail: grigoriytamasjan@mail.ru