© 2022 г. З.М. ШИБЗУХОВ, д-р физ.-мат. наук (intellimath@mail.ru)
(Институт математики и информатики Московского педагогического государственного университета; Московский физико-технический институт)

ОБ ОДНОЙ РОБАСТНОЙ СХЕМЕ ГРАДИЕНТНОГО БУСТИНГА НА ОСНОВЕ АГРЕГИРУЮЩИХ ФУНКЦИЙ, НЕЧУВСТВИТЕЛЬНЫХ К ВЫБРОСАМ[1]

Предложена одна новая робастная схема построения алгоритмов градиентного бустинга. Она основана на применении дифференцируемых оценок среднего значения, нечувствительных или малочувствительных к выбросам, при построении робастного функционала эмпирического риска. Это позволило применить метод итеративного перевозвешивания для для поиска очередной базовой функции и ее веса. Такая процедура градиентного бустинга позволяет находить искомую зависимость по данным, которые содержат относительно большую долю выбросов.

Ключевые слова: градиентный бустинг, робастная оценка, регрессия, классификация.

DOI: 10.31857/S0005231022100142, **EDN:** ALQEEL

1. Введение

Методы бустинга [1] являются разновидностью методов машинного обучения для построения *ансамблей базовых алгоритмов*. Модель базовых алгоритмов позволяет строить *слабые алгоритмы*, которые имеют относительно небольшую сложность и заведомо не являются переобученными. Модель базовых алгоритмов также может позволять строить *сложные алгоритмы* с высокими показателями качества, но склонные к переобучению. В таких случаях в методах бустинга они, как правило, используются с ограничениями на сложность, которые позволяют исключить переобучение базовых алгоритмов, но в то же время делают их более слабыми. Целевой алгоритм, как правило, строится в виде линейной комбинации базовых алгоритмов. Такой подход к построению алгоритмов машинного обучения позволяет строить сильные алгоритмы машинного обучения из более слабых алгоритмов.

Метод *градиентного бустинга* направлен на решение задачи построения линейной композиции некоторого заранее неизвестного количества базовых алгоритмов, которые минимизируют оценку эмпирического риска на обучающем множестве примеров. В классической схеме построения алгоритмов машинного обучения для решения задач регрессии и классификации эмпириче-

[1] Работа выполнена при поддержке научного проекта № АААА-А20-120122190034-9 Московского педагогического государственного университета.

ский риск оценивается как среднее арифметическое от потерь:

$$\mathcal{Q}(w) = \frac{1}{N}\sum_{k=1}^{N} \ell(f(\tilde{x}_k; w), \tilde{y}_k), \tag{1}$$

где $f(x; w)$ — параметризованная зависимость, $\{\tilde{x}_1, \ldots, \tilde{x}_N\} \subset \mathbb{R}^n$ — обучающие входы, $\{\tilde{y}_1, \ldots, \tilde{y}_N\}$ — ожидаемые значения на выходе, $\ell(y, \tilde{y})$ – неотрицательная дифференцируемая функция потерь. Например:

1) в задаче регрессии $\ell(y, \tilde{y}) = \varrho(y - \tilde{y})$, где $\varrho(r)$ — квазивыпуклая функция с минимумом в нуле, например $\varrho(r) = r^2$;

2) в задаче классификации для двух классов $\ell(y, \tilde{y}) = \varrho(1 - \tilde{y}y)$, где $\varrho(r)$ — монотонно убывающая функция, строго положительная при $r < 0$ и стремящаяся к нулю при $r \to +\infty$, например, $\varrho(r) = \max(0, 1 - \tilde{y}y)$ (функция Хинжа).

Требуется найти

$$w^\star = \arg\min_w \mathcal{Q}(w).$$

Для повышения робастности ранее предлагалось использовать более робастные функции потерь [2]. Например, в задаче регрессии:

1) $\varrho(r) = \sqrt{\varepsilon^2 + r^2} - \varepsilon$ ($\varrho'(r)$ ограничена);

2) $\varrho(r) = \ln(a^2 + r^2) - 2\ln a$ ($\varrho'(r) \to 0$ при $r \to \pm\infty$);

3) $\varrho(r) = |r| / \sqrt{\varepsilon^2 + r^2}$ ($\varrho(r)$ — ограничена),

а в задаче классификации:

1) $\varrho(r) = \ln(1 + \max(0, 1 - r))$ ($\varrho'(r) \to 0$ при $r \to \pm\infty$);

2) $\varrho(r) = \eta(\max(0, 1 - r))$, где $\eta(s)$ монотонно возрастающая ограниченная функция при $s > 0$.

Для поиска $w^\star$, которая минимизирует $\mathcal{Q}(w)$ с более "робастными" функциями потерь, применяется *метод итеративного перевзвешивания* [3, 4]. Например,

1) в случае робастной регрессии решение задачи

$$w^\star = \arg\min_w \frac{1}{N}\sum_{k=1}^{N} \varrho\left(f(\tilde{x}_k; w) - \tilde{y}_k\right)$$

сводится к решению цепочки задач:

$$w^{t+1} = \arg\min \sum_{k=1}^{N} v_k^t \left(f(\tilde{x}_k; w) - \tilde{y}_k\right)^2, \tag{2}$$

где

$$v_k^t = \varphi\left(f(\tilde{x}_k; w^t) - \tilde{y}_k\right), \quad \varphi(r) = \varrho'(r)/r;$$

2) в случае задачи классификации решение задачи

$$w^\star = \arg\min_w \frac{1}{N}\sum_{k=1}^{N} \varrho\left(\max\left(0, 1 - \tilde{y}_k f(\tilde{x}_k; w)\right)\right)$$

сводится к решению цепочки задач:

$$w^{t+1} = \arg\min \sum_{k=1}^{N} v_k^t \max\left(0, 1 - \tilde{y}_k f(\tilde{x}_k; w)\right), \tag{3}$$

где

$$v_k^t = \varphi\left(1 - \tilde{y}_k f(\tilde{x}_k; w^t)\right), \quad \varphi(r) = \varrho'(r)/r \ \text{ при } \ r < 0 \ \text{ и } \ \varphi(r) = 0 \ \text{ при } \ r \geqslant 0.$$

Здесь на каждом шаге процедуры итерационного перевзвешивания минимизируется взвешенная сумма квадратов ошибки (в задаче регрессии) или взвешенная сумма отступов с обратным знаком (в задаче классификации). Подобные схемы хорошо известны. Однако если обучающие данные содержат выбросы, из-за которых распределение значений потерь неизбежно будет содержать выбросы, то такой подход сталкивается с трудностями из-за неустойчивости среднего арифметического. Поэтому для преодоления этой проблемы было предложено использовать оценки среднего значения, которые нечувствительны или малочувствительны к выбросам [4, 5]. В этом случае робастная оценка средних потерь имеет вид

$$\mathcal{Q}(w) = \mathsf{M}\left\{\ell(f(\tilde{x}_1; w), \tilde{y}_1), \ldots, \ell(f(\tilde{x}_N; w), \tilde{y}_N)\right\},$$

где $\mathsf{M}\{z_1, \ldots, z_N\}$ — усредняющая агрегирующая функция. В [6, 7] было предложено использовать дифференцируемые оценки среднего, которые являются сглаженными вариантами известных робастных оценок среднего — медианы, α-квантиля и винзоризированного среднего арифметического. Это позволяет тоже применить метод итеративного перевзвешивания, но с другой схемой пересчета весов в (2) и (3). В настоящей работе эта робастная схема распространяется на метод градиентного бустинга. Далее сначала опишем классическую схему градиентного бустинга, а затем — робастную.

2. Классическая схема градиентного бустинга

Классическую схему метода *градиентного бустинга* [8] можно описать следующим образом. Рассмотрим класс функций $L(\mathcal{H})$, состоящий из линейных комбинаций *базовых функций* из некоторого класса функций $\mathcal{H}$

$$H(x) = \sum_{j=1}^{p} \alpha_j h_j(x),$$

где $\alpha_j \in \mathbb{R}$, $h_j \in \mathcal{H}$, $x \in \mathbb{R}^n$.

В классе $L(\mathcal{H})$ ищется оптимальная функция H^*, которая доставляет минимум

$$H^* = \arg \min_{H \in L(\mathcal{H})} \mathcal{Q}(H)$$

функционалу $\mathcal{Q}(H)$:

$$\mathcal{Q}_V(H) = \sum_{k=1}^{N} v_k \ell(H(\tilde{x}_k), \tilde{y}_k), \tag{4}$$

где $V = \{v_k \colon k = 1, \ldots, N\}$, $v_k \geqslant 0$ — веса примеров, такие что $v_1 + \cdots + v_N = 1$. Например, $v_k = 1/N$.

Для произвольных $\alpha \in \mathbb{R}$ и $h \in \mathcal{H}$ рассматривается функционал

$$\mathcal{Q}_V\big(h, \alpha\big) = \mathcal{Q}_V\big(H + \alpha h\big) = \sum_{k=1}^{N} v_k \ell(\tilde{H}_k + \alpha h(\tilde{x}_k), \tilde{y}_k), \tag{5}$$

где $\tilde{H}_k = H(\tilde{x}_k)$.

Функция h и параметр α в (5) выбираются в результате решения задачи минимизации:

$$h^\star, \alpha^\star = \arg \min_{h, \alpha} \mathcal{Q}_V(h, \alpha). \tag{6}$$

Для поиска минимума $\mathcal{Q}_V(h, \alpha)$ можно применить процедуру поиска h и α из известных алгоритмов градиентного бустинга, которые основаны на минимизации взвешенной суммы потерь. Для нахождения экстремума $\mathcal{Q}_V$ будем применять итеративный метод *поочередной минимизации* (*alternating minimization*) [9]

$$\begin{aligned} h^{p+1} &= \arg \min_{h} \mathcal{Q}_V(h, \alpha^p) \\ \alpha^{p+1} &= \arg \min_{\alpha} \mathcal{Q}_V(h^{p+1}, \alpha). \end{aligned} \tag{7}$$

На каждом шаге итерации сначала решается первая задача для поиска h^{p+1}, а затем вторая задача для поиска α^{p+1}. Итерационный процесс завершается, если $\big|\mathcal{Q}(h^{p+1}, \alpha^{p+1}) - \mathcal{Q}(h^p, \alpha^p)\big| < \varepsilon$ для заданного $\varepsilon > 0$, или если $t = t_{\max}$, где $t_{\max}$ — максимальное число шагов итерации. Для упрощения вычислений иногда в алгоритмах градиентного бустинга выполняется *только один* шаг метода (7). Практика также показала, что достаточно использовать небольшое число таких шагов. В некоторых случаях α^{p+1} можно вычислить явно (опираясь на необходимое условие экстремума $\mathcal{Q}_V$ по α), например

1) для задачи регрессии с $\ell(y, \tilde{y}) = \frac{1}{2}(y - \tilde{y})^2$ следующим образом:

$$\alpha^{p+1} = \frac{\sum\limits_{k=1}^{N} v_k (H_k - \tilde{y}_k) h^{p+1}(\tilde{x}_k)}{\sum\limits_{k=1}^{N} v_k \left(h^{p+1}(\tilde{x}_k)\right)^2};$$

2) для задачи классификации с $\ell(y, \tilde{y}) = \max(0, 1-\tilde{y}y)$ следующим образом:

$$\alpha^{p+1} = \frac{\sum\limits_{k \in I_p} \tilde{v}_k (1 - \tilde{y}_k H_k) \tilde{y}_k h^{p+1}(\tilde{x}_k)}{\sum\limits_{k \in I_p} \tilde{v}_k \left(\tilde{y}_k h^{p+1}(\tilde{x}_k)\right)^2},$$

где

$$\tilde{v_k} = \frac{v_k}{1 - \tilde{y}_k H_k - \alpha^p \tilde{y}_k h^{p+1}(\tilde{x}_k)},$$

а

$$I_p = \left\{k \colon 1 - \tilde{y}_k H_k - \alpha^p \tilde{y}_k h^{p+1}(\tilde{x}_k) > 0\right\}.$$

В целом алгоритм градиентного бустинга можно выразить при помощи следующего псевдокода:

def gb_fit(M,V):
| $H_0 = 0$
| **for** j **in** $[1, \ldots, M]$:
| | $h_j, \alpha_j = \arg\min\limits_{h,\alpha} \mathcal{Q}_V\big(H_{j-1} + \alpha h\big)$.
| | $H_j = H_{j-1} + \alpha_j h_j(x)$
| **return** H_M

3. Робастная схема градиентного бустинга

Эмпирическое распределение значений

$$\left\{z_k = z_k(h, \alpha) = \ell(\tilde{H}_k + \alpha h(\tilde{x}_k), \tilde{y}_k) \colon k = 1, \ldots, N\right\}$$

может содержать выбросы из-за искажений в данных или неадекватности части данных по отношению к выбранной модели зависимости, особенно на начальной стадии градиентного бустинга. Так как среднее арифметическое чувствительно к выбросам, то в результате минимизации (5), как правило, получаются искаженные h и α.

Проблему выбросов можно было бы решить путем подбора набора весов $v_1, \ldots, v_N$, так чтобы для индексов k, соответствующих выбросам, значения v_k были достаточно малы, чтобы невелировать их влияние. Однако задача поиска таких значений весов по сложности сопоставима с задачей идентификации выбросов. Ниже сформулируем подход, который может позволить преодолеть влияние выбросов, а также найти соответствующие значения весов $v_1, \ldots, v_N$.

Для этого сформулируем более робастную постановку задачи:

$$h^\star, \alpha^\star = \arg\min_{h,\alpha} \mathcal{Q}_{\mathsf{M}}(h, \alpha), \tag{8}$$

где

$$\mathcal{Q}_{\mathsf{M}}(h,\alpha) = \mathsf{M}\{z_1(h,\alpha),\ldots,z_N(h,\alpha)\},$$

где $\mathsf{M}\{z_1,\ldots,z_N\}$ — дифференцируемая усредняющая агрегирующая функция, более устойчивая к выбросам в данных [10].

Необходимое условие экстремума дает систему уравнений

$$\sum_{k=1}^{N} v_k(h,\alpha)\nabla_{h,\alpha}\ell(\tilde{H}_k + \alpha h(\tilde{x}_k),\tilde{y}_k) = 0,$$

где

$$\nu_k(h,\alpha) = \frac{\partial \mathsf{M}\{z_1(h,\alpha),\ldots,z_N(h,\alpha)\}}{\partial z_k}. \tag{9}$$

Дифференцируемые усредняющие агрегирующие функции $\mathsf{M}\{z_1,\ldots,z_N\}$, по построению, такие что $\partial\mathsf{M}/\partial z_k \geqslant 0$ для всех $k = 1,\ldots,N$ и

$$\partial\mathsf{M}/\partial z_1 + \cdots + \partial\mathsf{M}/\partial z_N = 1.$$

Для поиска оптимальных значений $h^\star$ и $\alpha^\star$ (решения задачи (8)) будем применять процедуру итеративного перевзвешивания, следуя [11]:

$$h^t,\alpha^t = \arg\min_{h,\alpha}\sum_{k=1}^{N}\nu_k\left(h^{t-1},\alpha^{t-1}\right)\ell(\tilde{H}_k + \alpha h(\tilde{x}_k),\tilde{y}_k). \tag{10}$$

Данная схема итеративного перевзвешивания возникает в результате применения общего метода Якоби для решения системы нелинейных уравнений

$$\begin{cases} v_k = \dfrac{\partial \mathsf{M}\{z_1(h,\alpha),\ldots,z_N(h,\alpha)\}}{\partial z_k} \\ \displaystyle\sum_{k=1}^{N} v_k\nabla_{h,\alpha}\ell(\tilde{H}_k + \alpha h(\tilde{x}_k),\tilde{y}_k) = 0, \end{cases}$$

которая возникает из необходимого условия экстремума для (8).

В этой итеративной схеме на шаге t осуществляется минимизация взвешенной суммы потерь

$$\mathcal{Q}^t_{V_t}(h,\alpha) = \sum_{k=1}^{N} v_k^t\ell(\tilde{H}_k + \alpha h(\tilde{x}_k),\tilde{y}_k),$$

где

$$V_t = \left\{v_k^t = \nu_k(h^{t-1},\alpha^{t-1})\colon k = 1,\ldots,N\right\}.$$

def gb_fit_step_M(H, $t_{\max}$):
| инициализация h^0, α^0
| $\tilde{H}_k = H(\tilde{x}_k)$, $k = 1, \ldots, N$
| **for** $t = 1, \ldots, t_{\max}$:
| | $h^t, \alpha^t = \arg\min\limits_{h,\alpha} \mathcal{Q}^t_{V_t}(h, \alpha)$.
| | **if** выполнено условие останова:
| | | **break**
| **return** h^t, α^t

def gb_fit_M(M):
| $H_0 = 0$
| **for** $j = 1, \ldots, M$:
| | $h_j, \alpha_j =$ gb_fit_step_M(H_{j-1}, $t_{\max}$)
| | $H_j = H_{j-1} + \alpha_j h_j(x)$
| **return** H_M

Для поиска решения задачи минимизации $\mathcal{Q}^t_{V_t}(h, \alpha)$ будем применять процедуру *альтернативной минимизации* (*alternating minimization*) [9]

$$h^t_p = \arg\min_h \sum_{k=1}^N v^t_k \ell(\tilde{H}_k + \alpha^{t-1}_{p-1} h(\tilde{x}_k), \tilde{y}_k)$$

$$\alpha^t_p = \arg\min_\alpha \sum_{k=1}^N v^t_k \ell(\tilde{H}_k + \alpha h^t_p(\tilde{x}_k), \tilde{y}_k),$$

где $h^{t-1}_0 = h^{t-1}$, $\alpha^{t-1}_0 = \alpha^{t-1}$.

Для решения приведенных задач минимизации использовался метод градиентного спуска с применением схемы ADAM [12].

Рассмотрим отдельно некоторые варианты реализации метода робастного градиентного бустинга для задачи регрессии и задачи классификации, которые можно получить в рамках предложенной выше схемы.

3.1. Задача регрессии

В задаче регрессии функция потерь, как правило, имеет вид: $\ell(y, \tilde{y}) = \varrho(y - \tilde{y})$, где ϱ — неотрицательная дифференцируемая квазивыпуклая унимодальная функция, $0 \in \arg\min \varrho(r)$.

Итерационная схема (10) принимает вид:

$$h^t, \alpha^t = \arg\min_{h,\alpha} \sum_{k=1}^N \nu_k(h^{t-1}, \alpha^{t-1}) \varrho\big(\tilde{H}_k - \tilde{y}_k + \alpha h(\tilde{x}_k)\big),$$

где $\tilde{H}_k - \tilde{y}_k + \alpha h(\tilde{x}_k)$ — величина ошибки для k-го прецедента.

Типичный пример $\varrho(r) = r^2$. В рамках классической схемы построения робастной регрессии [13] можно построить следующую процедуру итеративного перевзвешивания:

$$h^t = \arg\min_{h\in\mathcal{H}} \sum_{k=1}^{N} v_k(h^{t-1}, \alpha^{t-1}) \left(\tilde{H}_k - \tilde{y}_k + \alpha^{t-1} h(\tilde{x}_k)\right)^2$$

$$\alpha^t = \arg\min_{\alpha} \sum_{k=1}^{N} v_k(h^{t-1}, \alpha^{t-1}) \left(\tilde{H}_k - \tilde{y}_k + \alpha h^t(\tilde{x}_k)\right)^2,$$

где $v_k(h, \alpha) = \nu_k(h, \alpha)\varphi\big(\tilde{H}_k - \tilde{y}_k + \alpha h(\tilde{x}_k)\big)$, $\varphi(r) = \varrho'(r)/r$, $\nu_k(h, \alpha)$ вычисляется по формуле (9).

Величину α^t в данной схеме можно вычислить явно

$$\alpha^t = \frac{\sum\limits_{k=1}^{N} v_k(h^{t-1}, \alpha^{t-1})(\tilde{y}_k - \tilde{H}_k) h^t(\tilde{x}_k)}{\sum\limits_{k=1}^{N} v_k(h^{t-1}, \alpha^{t-1}) \left(h^t(\tilde{x}_k)\right)^t}.$$

3.2. Задача классификации

В задаче классификации для двух классов функция потерь может иметь вид $\ell(y, \tilde{y}) = \varrho(1 - y\tilde{y})$, где $\varrho(r)$ — неотрицательная монотонно возрастающая функция, $\lim\limits_{r\to+\infty} \varrho(r) = +\infty$, $\varrho(r) > 0$ при $r < 0$.

Итерационная схема (10) принимает вид:

$$h^t, \alpha^t = \arg\min_{h,\alpha} \sum_{k=1}^{N} \nu_k(h^{t-1}, \alpha^{t-1}) \varrho\big(1 - \tilde{y}_k \tilde{H}_k - \alpha \tilde{y}_k h(\tilde{x}_k)\big),$$

где $\tilde{y}_k \tilde{H}_k + \alpha \tilde{y}_k h(\tilde{x}_k)$ — величина отступа для k-го прецедента.

Приведем примеры:

1) $\varrho(r) = \max(0, r)$;
2) $\varrho(r) = \dfrac{1}{\lambda} \ln(1 + e^{\lambda r})$;
3) $\varrho(r) = \dfrac{1}{2}\big(-r + \sqrt{\varepsilon^2 + r^2}\big)$.

В рамках классической схемы построения робастной регрессии [13] построим следующую процедуру итеративного перевзвешивания:

$$h^t = \arg\min_{h\in\mathcal{H}} \sum_{k=1}^{N} v_k(h^{t-1}, \alpha^{t-1}) \left(1 - \tilde{y}\tilde{H}_k - \alpha^{t-1} \tilde{y}_k h(\tilde{x}_k)\right)^2$$

$$\alpha^t = \arg\min_{\alpha} \sum_{k=1}^{N} v_k(h^{t-1}, \alpha^{t-1}) \left(1 - \tilde{y}_k \tilde{H}_k - \alpha \tilde{y}_k h^t(\tilde{x}_k)\right)^2,$$

где

$$v_k(h,\alpha) = \nu_k(h,\alpha)\varphi\big(1 - \tilde{y}_k\tilde{H}_k - \alpha^{t-1}\tilde{y}_k h^{t-1}(\tilde{x}_k)\big),$$

$$\varphi(r) = \varrho'(r)/r \quad \text{при} \quad r < 0$$

и

$$\varphi(r) = 0 \quad \text{при} \quad r \geqslant 0.$$

Величину α^t можно вычислить явно:

$$\alpha^t = \frac{\sum\limits_{k=1}^{N} v_k(h^{t-1},\alpha^{t-1})(1 - \tilde{y}_k\tilde{H}_k)\tilde{y}_k h^{t-1}(\tilde{x}_k)}{\sum\limits_{k=1}^{N} v_k(h^{t-1},\alpha^{t-1})\left(\tilde{y}_k h^{t-1}(\tilde{x}_k)\right)^2}.$$

4. Иллюстративные примеры

В следующих примерах будет использоваться робастная оценка среднего

$$\mathsf{WM}_\alpha\{z_1,\ldots,z_N\} = \frac{1}{N}\sum_{k=1}^{N}\min(z_k,\bar{z}_\alpha),$$

где

$$\bar{z}_\alpha = \mathsf{M}_\alpha\{z_1,\ldots,z_N\} = \arg\min_u \sum_{k=1}^{N}\rho_\alpha(z_k - u),$$

$$\rho_\alpha(r) = \begin{cases} \alpha\rho(r), & \text{если } r \geqslant 0 \\ (1-\alpha)\rho(r), & \text{если } r < 0, \end{cases} \qquad \rho(r) = \sqrt{\varepsilon^2 + r^2} - \varepsilon.$$

Здесь M_α — «гладкий вариант» α-квантиля, $\varepsilon = 0{,}001$. Робастная оценка WM_α — среднее арифметическое предварительно отцензурированных неотрицательных значений при помощи порогового значения $\bar{z}_\alpha$. В качестве функции потерь в задачах регрессии будет выступать $\ell(y,\tilde{y}) = \frac{1}{2}(y-\tilde{y})^2$ — квадрат ошибки.

Функции $h(x,w)$ выбираются из класса сигмоидальных нейронов

$$h(x,w) = \sigma(w_0 + w_1x_1 + \cdots + w_nx_n),$$

где $\sigma(s) = \operatorname{th}\lambda s$ (по умолчанию $\lambda = 1$, если не оговорено иное). Таким образом, класс функций $L(\mathcal{H})$ описывает функции преобразования нейронной сети со скрытым слоем из сигмоидальных нейронов. Количество нейронов в скрытом слое относительно небольшое во избежание переобучения.

Все вычисления выполнены с помощью языка программирования python и библиотеки mlgrad (https://bitbucket.org/intellimath/mlgrad).

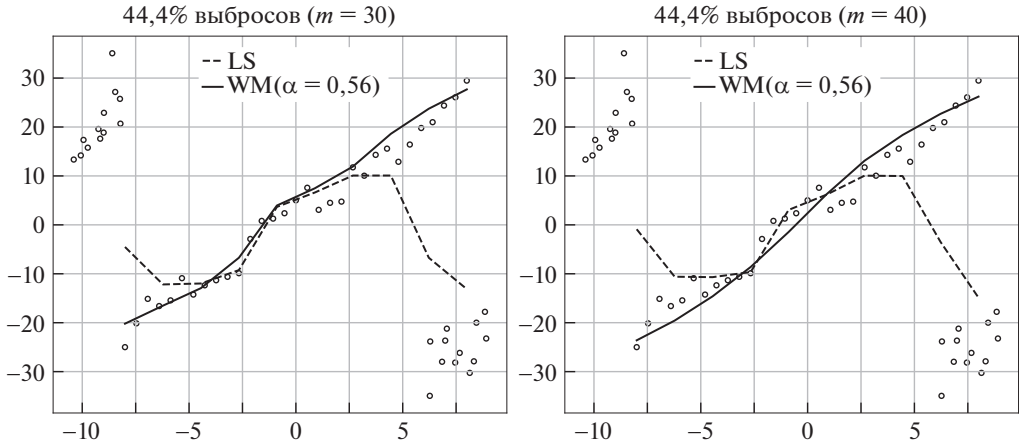

Рис. 1. Графики восстановленных функций для примера с линейной регрессией.

1. Наглядный пример с линейной регрессией. В этом примере выбран набор точек на плоскости, расположенных вдоль некоторой прямой линии. К ним добавлены новые точки — выбросы, которые расположены кучно по разные стороны от прямой линии, так чтобы при восстановлении линейной функции при помощи метода наименьших квадратов найденная прямая линия сильно поворачивалась, притягиваясь к выбросам. Выбросы составляют 44% выборки. Параметр $\lambda = 0{,}5$ в $\sigma(s)$. На рис. 1 приведены графики восстановленных функций.

2. Набор данных breast_cancer.[2] Ко входным векторам предварительно была применена процедура стандартного масштабирования при помощи преобразования $\{x_k\} \to \left\{\frac{x_k - \bar{x}}{\sigma}\right\}$, где $\bar{x}$ — среднее арифметическое, а σ — стандартное отклонение, для приведения значений признаков ко взаимно сопоставимым масштабам значений. Для этого набора строились два варианта функции $H(x)$, которые содержат небольшое число слагаемых ($m = 20$ и $m = 30$). В робастном варианте $\alpha = 0{,}95$. На рис. 3 построены кривые распределения абсолютных значений ошибок в логарифмических координатах. Нетрудно увидеть, что применение более робастной функции оценки среднего значения может позволить уменьшить абсолютную величину ошибок для подавляющего большинства примеров.

3. Сгенерированная однослойная нейронная сеть с одним скрытым слоем. Это искусственно сгенерированный набор данных на основе функции

$$H(x) = \sum_{k=1}^{m} \alpha_j \sigma(w_{j,0} + w_{j,1}x_1 + w_{j,2}x_2), \quad m = 40,$$

в которой значения весов $w_{j,0}$, $w_{j,1}$, $w_{j,2}$ и коэффициентов α_j для простоты выбраны случайно из равномерного распределения на $[-1, 1]$ (то, что значения выбраны из равномерного распределения принципиального значения не

[2] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer

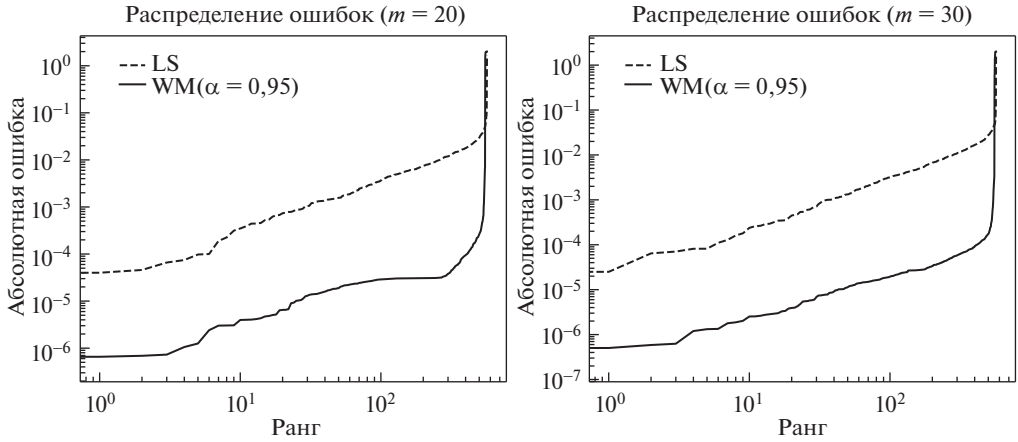

Рис. 2. Графики распределения абсолютных значений ошибок в примере 2.

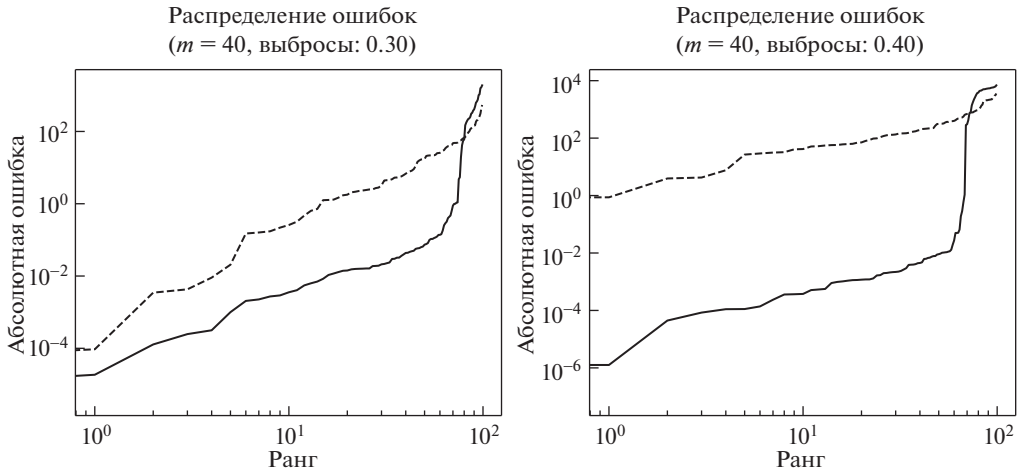

Рис. 3. Графики распределения абсолютных значений ошибок в примере 3.

имеет). Аналогично случайно выбирается набор входов $\{x_k\colon k = 1, \ldots, 100\} \subset$ $\subset [-3, 3]^2$. Для всех k вычисляются значения $y_k = H(x_k)$. Из этого набора данных создаются два набора с долями выбросов $M = 30\%$ и $M = 40\%$. Значение $\tilde{y}_k$ в точке выбросов увеличивается в 10 раз. На рис. 3 построены кривые распределения абсолютных значений ошибок в логарифмических координатах. На рисунках сплошная кривая соответствует робастному варианту градиентного бустинга. Нетрудно увидеть, что применение более робастной функции среднего значений может позволить ощутимо уменьшить абсолютную величину ошибок практически для всех выбросов. При применении стандартной процедуры градиентного бустинга в точках, которые не являются выбросами, наблюдаются очень большие значения ошибок. В результате применения робастной процедуры градиентного бустинга ошибки для нормальных точек могут стать достаточно малы.

5. Заключение

Предложенный в данной статье подход сравним с известным подходом к повышению робастности алгоритмов регрессии и классификации, основанным на применении более робастных функций потерь. Существенное отличие предложенной выше робастной схемы состоит в способе пересчета весов примеров в процедуре итеративного перевзвешивания. В случае применения в (5) с $v_k = 1/N$ более робастных функций потерь веса примеров вычисляются по формуле вида

$$v_k = \varphi(z_k),$$

где $\varphi(z)$ — неотрицательная, как правило, убывающая функция от z или $|z|$. Эффект снижения влияния выбросов достигается за счет малости весов примеров, которые являются выбросами (как правило, с большими значениями z или $|z|$). В нашем подходе веса пересчитываются по формуле вида:

$$v_k = \psi(z_k - \bar{z}),$$

где $\varphi(z)$ — тоже неотрицательная убывающая функция от z, $\bar{z}$ — величина робастной оценки среднего значения $z_1, \ldots, z_N$, которая нечувствительна или малочувствительна к выбросам. Отличие состоит в том, здесь вес примера является функцией отклонения z_k от среднего значения. Так, в задаче регрессии, когда значения z_k соответствуют ошибкам, в ситуации, со значением $\bar{z}$ существенно отличающимся от нуля, значения весов примеров в предложенном робастном подходе оказываются существенно меньше. Это получается потому, что когда все ошибки существенно отделены от нуля, они оказываются в области значений z, где значение функции φ (в (2) и (3)) убывает медленнее, чем около нуля. В предложенном робастном подходе случае разность $z_k - \bar{z}$ оказывается ближе к нулю и поэтому происходит более быстрое падение значений весов примеров по мере удаления z_k от $\bar{z}$. В результате в рамках предложенного здесь метода примеры, соответствующие выбросам, получают такие малые значения весов (по сравнению с весами примеров, которые не являются выбросами), достаточные для того, чтобы преодолеть их влияние.

СПИСОК ЛИТЕРАТУРЫ

1. *Freund Y., Schapire R.E.* A decision-theoretic generalization of on-line learning and an application to boosting // J. of Comput. and Syst. Sci. 1997. V. 55. No. 1. P. 119–139.
2. *Kanamori T., Takenouchi T., Eguchi S., Murata N.* Robust loss functions for boosting // Neural Computation. 2007. V. 19. No. 8. P. 2183–2244.
3. *Holland P.W., Welsch R.E.* Robust regression using iteratively reweighted least squares // Communications in Statistics — Theory and Methods. 1977. V. 6. No. 9. P. 813–827.
4. *Rousseeuw P.J., Leroy A.M.* Robust Regression and Outlier Detection. New York: John Wiley and Sons. 1987.

5. *Rousseeuw P.J., Hubert M.* High-breakdown estimators of multivariate location and scatter / Becker C., Fried R., Kuhnt S., editors. Robustness and Complex Data Structures. Springer, 2013. P. 49–66.
6. *Шибзухов З.М.* О принципе минимизации эмпирического риска на основе усредняющих агрегирующих функций // Докл. РАН. 2017. Т. 476. № 5. С. 495–499.
7. *Shibzukhov Z.M.* Machine learning based on the principle of minimizing robust mean estimates / Advances in Intelligent Systems and Computing. V. 1310. P. 472–477. Springer International Publishing. 2020.
8. *Friedman J.H.* Greedy function approximation: A gradient boosting machine // Annals Statist. 2001. V. 29. No. 5.
9. *Csiszar I., Tusnady G.* Information geometry and alternating minimization procedures // Statistics and Decisions, Supplement Issue. 1984. No. 1. P. 205–237.
10. *Calvo T., Beliakov G.* Aggregation functions based on penalties // Fuzzy Sets and Systems. 2010. V. 161. No. 10. P. 1420–1436.
11. *Shibzukhov Z.M., Semenov T.A.* Machine learning based on minimizing robust mean estimates. In: Pattern Recognition. ICPR International Workshops and Challenges. P. 112–119. Springer International Publishing. 2021.
12. *Kingma D.P., Ba J.* Adam: A method for stochastic optimization. arXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980
13. *Huber P.J.* Robust Statistics. John Wiley and Sons. 1981.

Статья представлена к публикации членом редколлегии А.А. Лазаревым.

Поступила в редакцию 31.01.2022

После доработки 23.05.2022

Принята к публикации 29.06.2022