© 2013 г. Е.В. БУРНАЕВ, канд. физ.-мат. наук,
П.В. ПРИХОДЬКО,
(ООО "Датадванс", Институт проблем передачи информации имени А.А. Харкевича РАН, Москва; Московский физико-технический институт, Долгопрудный)

ОБ ОДНОЙ МЕТОДИКЕ ПОСТРОЕНИЯ АНСАМБЛЕЙ РЕГРЕССИОННЫХ МОДЕЛЕЙ[1]

Предложена методика построения ансамблей регрессионных моделей. Проводится её теоретический и экспериментальный анализ.

1. Введение

В работе рассматривается стандартная постановка регрессионной задачи. Имеется выборка значений неизвестной функции f и метод построения базовых аппроксиматоров g из некоторого семейства G. Необходимо построить достаточно близкий к f аппроксиматор $\hat{f}$ при условии, что метод построения базовых аппроксиматоров g не обеспечивает необходимой точности. В ряде работ (например, в [1, 2, 3]) показано, что точность аппроксимации может быть существенно повышена, если аппроксиматор $\hat{f}$ определенным образом строится как линейная комбинация (ансамбль) базовых аппроксиматоров из G.

Основными методами построения ансамблей являются методы Беггинг (Bagging) [1, 4] и Бустинг (Boosting) [2, 5, 6]. В данной работе предлагается новый метод построения ансамблей базовых аппроксиматоров. На примере применения к искуственным нейронным сетям при некоторых дополнительных предположениях доказывается сходимость метода, а также проводится сравнение предлагаемого метода с основными подходами к построению ансамблей, описанными в публикациях.

Статья имеет следующую структуру. В разделе 2 приводится постановка задачи восстановления зависимости и построения ансамбля и дается краткий обзор основных методов построения ансамблей. В разделе 3 описывается предлагаемый в статье подход и приводятся результаты о некоторых теоретических свойствах метода. Кроме того, проводится сравнение предлагаемого метода с другими современными методами построения ансамблей. В разделе 4 проводится экспериментальный анализ работы предложенного метода. В разделе 5 приведены выводы из результатов работы.

2. Постановка задачи построения ансамблей

Пусть задана обучающая выборка $S_m = (X_m, Y_m)$, $X_m = \{x_i, i = 1, \ldots, m\}$, $Y_m = \{y_i, i = 1, \ldots, m\}$ такая, что

[1]Работа выполнена при поддержке Лаборатории структурных методов анализа данных в предсказательном моделировании МФТИ, грант правительства РФ договор 11.G34.31.0073 и Российского Фонда Фундаментальных Исследований (проект 13-01-00521).

— входные векторы (входы) $x_i \in \mathbf{X} \subset \mathbf{R}^d$ ($\mathbf{X}$ — произвольное компактное множество) порождаются независимым образом некоторым неизвестным непрерывным распределением $P(x)$, $x \in \mathbf{X}$,

— между выходным значением (выходом) $y_i \in \mathbf{Y} \subset \mathbf{R}^1$ и входным вектором $x_i \in \mathbf{X}$ есть некоторая неизвестная функциональная зависимость $y_i = f(x_i)$.

Задача восстановления неизвестной зависимости $y = f(x)$ состоит в построении по обучающей выборке S_m регрессионной зависимости (аппроксиматора) $\hat{f}(x) = \hat{f}\,(x|S_m)$ такой, что её обобщающая способность [7, 8, 9] высока, т.е. величина (риск)

$$er_{P,f}\left(\hat{f}\right) = \mathbb{E}\left(f(x) - \hat{f}(x)\right)^2 = \int_{\mathbf{X}} \left(f(x) - \hat{f}(x)\right)^2 dP(x) \tag{1}$$

мала с вероятностью, близкой к единице. Здесь и далее $\mathbb{E}$ обозначает математическое ожидание по распределению $P(x)$.

На практике распределение $P(x)$ неизвестно, поэтому вместо (1) оценивают среднеквадратичную ошибку (эмпирический риск)

$$\hat{er}_{S_m}\left(\hat{f}\right) = \left\|f - \hat{f}\right\|^2_{X_m}, \tag{2}$$

где $\|v\|_{X_m} = \sqrt{\frac{1}{m}\sum_{i=1}^m (v(x_i))^2}$ обозначает выборочную норму функции v.

Введем определение базового аппроксиматора g. Обозначим через G пространство гипотез, т.е. рассматриваемое семейство функций, которые осуществляют отображение из $\mathbf{X}$ в $\mathbf{Y}$.

Будем называть базовым аппроксиматором функцию $g = \mathrm{Train}(S_m, w) \in G$, порождаемую некоторым методом обучения $\mathrm{Train}(\cdot)$, применённым к обучающей выборке S_m и, возможно, зависящим от некоторого набора гиперпараметров $w \in \Omega$ (например, от случайной инициализации начальных значений параметров базового аппроксиматора).

В качестве метода обучения могут быть использованы многие алгоритмы машинного обучения, такие как: линейная регрессия, регрессия на основе гауссовских процессов (кригинг), регрессия на основе опорных векторов, искуственные нейронные сети (ИНС), регрессионные деревья (CART) (детальнее см. в [10, 11]).

Теоретические результаты, представленные в данной работе, приводятся для случая, когда базовый аппроксиматор порождается ИНС [9, 11]. Аппроксиматор ИНС может быть представлена в виде

$$g(x) = \sum_{k=1}^{p} u_k \psi(x, \theta_k), \tag{3}$$

где $u_k \in \mathbf{R}^1$, $k = 1, \ldots, p$, сигмоидальная функция активации (сигмоид) имеет вид

$$\psi(x, \theta) = \sigma(\beta^{\mathrm{T}} \cdot x + \beta^0), \quad \sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}, \quad z \in \mathbf{R}^1, \tag{4}$$

$\beta \in \mathbf{R}^d$, $\beta^0 \in \mathbf{R}^1$, $\theta = (\beta, \beta^0) \in \Theta \subset \mathbf{R}^{d+1}$ (Θ — произвольное компактное множество), T — обозначает знак транспонирования.

Под построением ансамбля базовых аппроксиматоров для решения задачи восстановления неизвестной зависимости на основе данных S_m понимается способ получения аппроксиматора $\hat{f}(x)$ в виде некоторого функционала $\hat{f}(x) = f_n(g_1(x), \ldots, g_n(x))$, где $g_i(x)$, $i = 1, \ldots, n$, являются базовыми аппроксиматорами из некоторого заданного класса, построенными на основе данных S_m. Далее везде в работе рассматривается случай, когда ансамбль f_n линеен по $g_1, \ldots, g_n$, т.е.

$$f_n = \sum_{i=1}^{n} \alpha_i g_i \tag{5}$$

для некоторых коэффициентов $\{\alpha_i\}_{i=1}^n$.

Таким образом, алгоритм построения ансамбля сводится к описанию того, как

— построить базовые модели $g_1, \ldots, g_n$,
— задать веса $\{\alpha_i\}_{i=1}^n$.

3. Процедура БегБуст для построения ансамблей аппроксиматоров

Опишем предлагаемый метод построения ансамблей БегБуст (BagBoost). Для дальнейшего изложения введем обозначения:

— G — некоторое подмножество функционального гильбертова пространства H с функциональной нормой $\|\cdot\|$;
— $co(G) = \{g : g = \sum_{i=1}^n \alpha_i g_i,\ \alpha_i \geqslant 0, \quad \sum_{i=1}^n \alpha_i = 1, \quad g_i \in G, \quad n \geqslant 1\}$ — выпуклая оболочка множества G;
— $\overline{co}(G)$ — замыкание выпуклой оболочки $co(G)$.

Естественным примером гильбертова пространства H является пространство $L^2(\mathbf{X}, P)$, $\mathbf{X} \subset \mathbf{R}^d$, в котором $\|f\|^2 = \int_{\mathbf{X}} (f(x))^2 dP(x) < \infty$ для $f \in H$.

Введем два *предположения*:

— $\mathbf{A}_1$. *Подмножество G имеет вид*

$$G = \left\{ \alpha\, \psi(\cdot, \theta) \middle| |\alpha| \leqslant B, \quad \theta \in \Theta \right\} \subset L^2(\mathbf{X}, P), \tag{6}$$

где $\alpha \in \mathbf{R}^1$, $\psi(x, \theta)$ — сигмоидальная функция активации вида (4), множество значений параметров $\theta \in \Theta \subset \mathbf{R}^{d+1}$ (Θ — произвольное компактное множество), $0 < B < \infty$ — заданная константа;

— $\mathbf{A}_2$. *Неизвестная функция f принадлежит $\overline{co}(G)$.*

Выпуклая оболочка множества G будет иметь вид

$$co(G) = \left\{ f \in L^2(\mathbf{X}, P) \middle| f = \sum_{k=1}^{p} \alpha_k \psi(\cdot, \theta_k), \quad \sum_{k=1}^{p} |\alpha_k| \leqslant B, \quad \theta_k \in \Theta, \quad p \in \mathbb{N} \right\}, \tag{7}$$

при этом знаки параметров $\{\alpha_k\}_{k=1}^p$ значения не имеют, поскольку исходное множество G симметрично. Более того, сумма $\sum_{k=1}^p |\alpha_k|$ может быть строго меньше B, так как нулевая функция также является элементом множества G. Замыкание выпуклой оболочки множества G будет иметь вид

$$\overline{co}(G) = \left\{ f \in L^2(\mathbf{X}, P) \middle| f = \int_{\Theta} \psi(\cdot, \theta) d\mu(\theta), \quad \mu \in \mathcal{M}, \quad \|\mu\|_{\mathcal{M}} \leqslant B \right\}, \tag{8}$$

где $\mathcal{M}$ — множество всех мер Радона на Θ. Таким образом, $\overline{co}(G)$ содержит все функции, имеющие представление вида $\sum_i \alpha_i \psi(\cdot, \theta_i)$ и $\int_\Theta \alpha(\theta)\psi(\cdot,\theta)d\theta$, и тем самым является достаточно общим функциональным классом (см. [12]).

Заметим, что функции, принадлежащие $\overline{co}(G)$ (8), принимают значения в отрезке $[-B, B]$. В таком случае $\|f\|_{X_m} \leqslant B$ для любого $f \in \overline{co}(G)$.

3.1. Алгоритм БегБуст

Опишем схему работы предложенной процедуры БегБуст.

Алгоритм

1. Положим начальное значение выхода ансамбля $f_0(x) = 0$.
2. Для $k = 1, 2, \ldots$:
   а. Определим новую выборку данных

   (9) $S_m^k = \left\{\left(x_i, y_i^k\right), i = 1, \ldots, m\right\}$, $y_i^k = k \cdot y_i - (k-1) \cdot f_{k-1}(x_i)$, $\left(x_i, y_i\right) \in S_m$,

   где $f_{k-1}(x)$ — ансамбль, полученный на предыдущем $k-1$ шаге и состоящий из $k-1$ аппроксиматоров $g_i(x)$, $i = 1, \ldots, k-1$.

   б. Построим k-й базовый аппроксиматор $g_k = \mathrm{Train}(S_m^k) \in G$ ансамбля.

   в. Полагаем значение выхода ансамбля равным $f_k(x) = \frac{(k-1)}{k} f_{k-1}(x) + + \frac{1}{k} g_k(x) = \frac{1}{k}\sum_{i=1}^k g_i(x)$.

Таким образом, выход ансамбля $f_k \in co(G)$ равен арифметическому среднему выходов аппроксиматоров $g_i \in G$, $i = 1, 2, \ldots, k$, составляющих ансамбль.

3.2. Оценка скорости сходимости

Оценим сверху ошибки аппроксимации $\hat{er}_{S_m}(f_k)$, $k = 1, 2, \ldots$, последовательности приближений f_1, $f_2, \ldots$, порождаемой описанным выше алгоритмом БегБуст, при условии что выполняются предположения $\mathbf{A}_1$ и $\mathbf{A}_2$.

Пусть

$$\varepsilon_k = \hat{er}_{S_m^k}(g_k) - \min_{g \in G} \hat{er}_{S_m^k}(g), \quad k = 1, 2, \ldots, \tag{10}$$

$\varepsilon_n^{\max} = \max_{k=1,\ldots,n}\{\varepsilon_k\}$. Справедлив следующий основной результат.

Теорема 1. Для фиксированной выборки S_m и любого $k = 1, 2, \ldots$ выполнено, что

$$\hat{er}_{S_m}(f_k) \leqslant \frac{\Delta + \varepsilon_k^{\max}}{k}, \tag{11}$$

где $\Delta = B^2 - \frac{1}{m}\sum_{i=1}^m y_i^2 > 0$.

Следствие 1. Если в п. 2б на каждом шаге алгоритма БегБуст при построении базового аппроксиматора g_k достигается минимум ошибки $\hat{er}_{S_m^k}(g_k)$, $k = 1, 2, \ldots$, т.е. $\varepsilon_k = 0$ в (10), то ошибка $\hat{er}_{S_m}(f_k)$ убывает как $\frac{\Delta}{k}$.

Для доказательства теоремы 1 необходима следующая лемма, доказательство которой приведено в Приложении.

Лемма 1. Для любой $f \in \overline{co}(G)$, фиксированной выборки S_m, $h \in co(G)$ и $\alpha \in [0,1]$ справедливо, что

$$\min_{g \in G} \hat{er}_{S_m}(\alpha h + (1-\alpha)g) \leqslant \alpha^2 \hat{er}_{S_m}(h) + (1-\alpha)^2 \left(\max_{g \in G} \|g\|^2_{X_m} - \hat{er}_{S_m}(0) \right).$$

Следствие 2. Для любой $f \in \overline{co}(G)$, фиксированной выборки S_m, $h \in co(G)$ и $\alpha \in [0,1]$ справедливо, что

$$\min_{g \in G} \hat{er}_{S_m}(\alpha h + (1-\alpha)g) \leqslant \alpha^2 \hat{er}_{S_m}(h) + (1-\alpha)^2 \Delta.$$

Для оценки обобщающей способности предложенного алгоритма БегБуст необходимо выполнение следующего предположения.

— $\mathbf{A}_3$. *Для некоторого $\varepsilon > 0$ при $k \geqslant 1$ выполнено, что*

$$\varepsilon_k \leqslant \varepsilon. \tag{12}$$

Следствие 3. Если выполнено неравенство (12)*, то ошибка $\hat{er}_{S_m}(f_k)$ убывает как $\frac{\Delta+\varepsilon}{k}$.*

При выполнении предположений $\mathbf{A}_1$, $\mathbf{A}_2$ и $\mathbf{A}_3$ верна следующая теорема.

Теорема 2. Для любой $f \in \overline{co}(G)$, для всех $\gamma \in (0,1)$ и $\rho > 0$, если описанный в подразделе 3.1 алгоритм БегБуст применяется к выборке S_m размера по крайней мере $m(\rho,\gamma) = \frac{C_1}{\rho^2}\left(\ln \frac{4}{\gamma} + \frac{C_2}{\rho^2} \ln \frac{C_3}{\rho^5} \right)$, где $C_1 = 2^{12}B^4$, $C_2 = 5 \cdot 2^{14}(d+3)B^4$ и $C_3 = \frac{2^{21}eB^6}{(d+1)}C_2$, то с вероятностью не менее $1-\gamma$ через $n = \lceil \frac{2\delta}{\rho} \rceil$ шагов алгоритма БегБуст ошибка аппроксимации ансамбля f_n будет удовлетворять неравенству $er_{P,f}(f_n) < \rho$.

3.3. Обсуждение результатов

1. Приведем обзор основных методов построения ансамблей. Идея процедуры Беггинг (Bagging) [1, 3, 4] состоит в том, что базовые модели g_i, $i = 1, \ldots, n$, независимо и параллельно обучаются на различных (возможно, частично перекрывающихся) бутстреп-репликациях объема m, полученных из исходного обучающего множества S_m. Предсказание ансамбля f_n, состоящего из аппроксиматоров $g_i(x)$, $i = 1, \ldots, n$, подсчитывается согласно формуле

$$f_n(x) = \frac{1}{n} \sum_{i=1}^{n} g_i(x). \tag{13}$$

В публикациях не приводится результатов, доказывающих, что увеличение количества n базовых моделей в случае процедуры Беггинг позволяет достигнуть любую заданную ошибку приближения при достаточном увеличении объема m выборки. Отметим также, что кроме усреднения (13) с равными весами в (5) возможно использовать оптимальные веса $\{\alpha_i\}_{i=1}^n$, минимизирующие ошибку $er_{P,f}(f_n)$ [13]. Однако на практике результаты применения метода Беггинг с оптимальными весами вычислительно нестабильны, так как обычно матрица ковариаций между выходными значениями базовых аппроксиматоров, используемая при вычислении оптимальных весов, плохо обусловлена.

Помимо процедуры Беггинг для построения ансамблей аппроксиматоров также используются подходы типа Бустинг (Boosting). Основную идею подходов этого типа можно описать следующим образом:

1) по исходной обучающей выборке строится модель g_1;

2) обучающая выборка модифицируется с учетом точности модели g_1;

3) по модифицированной обучающей выборке строится модель g_2;

4) шаги 2) и 3) повторяются до тех пор, пока не выполнен критерий останова.

Опишем идеи двух основных подходов к реализации процедур типа Бустинг.

Первый подход состоит в том, что при обучении базового аппроксиматора каждой точке $(x, y) \in S_m$ обучающей выборки присваивается свой вес. Вес характеризует то, насколько хорошо эта точка была приближена уже построенными базовыми аппроксиматорами ансамбля [5, 6] (например, большие веса соответствуют большим значениям ошибки аппроксимации). Новый аппроксиматор строится с учетом весов так, чтобы компенсировать наиболее существенные ошибки предыдущих базовых аппроксиматоров ансамбля (например, новый аппроксиматор строится только по тем точкам обучающей выборки, которые имеют наибольшие значения весов, либо используется взвешенная норма ошибки). Наиболее известным примером метода, реализующего описанную концепцию, является алгоритм AdaBoost.R [6, 14, 15]. В рамках этого подхода для сходимости ошибки $\hat{er}_{S_m}(f_n)$ к нулю требуется обеспечить малость ошибки (2) базовых аппроксиматоров (weak hypothesis), что не всегда возможно на практике. По этой причине контролировать скорость сходимости ошибки $\hat{er}_{S_m}(f_n)$ к нулю затруднительно.

Второй подход заключается в том, что на каждой итерации процедуры Бустинг приближается некоторая функция от точек обучающей выборки $(x, y) \in S_m$ и предсказаний аппроксиматоров, построенных на предыдущих шагах алгоритма. Например, пусть $f_{n-1}(x)$ — ансамбль, полученный за предыдущие $n-1$ шагов и состоящий из $(n-1)$-й базовой модели $g_j(x)$, $j = 1, \ldots, n-1$. Тогда на n-м шаге алгоритма базовая модель $g_n(x)$ строится по выборке $S_m^n = \left\{\left(x_i, y_i^n\right), i = 1, \ldots, m\right\}$, где $y_i^n = y_i - f_{n-1}(x_i)$, $\left(x_i, y_i\right) \in S_m$, $i = 1, \ldots, m$. В таком случае предсказание ансамбля аппроксиматоров на n-й итерации имеет вид $f_n(x) = f_{n-1}(x) + \alpha_n g_n(x)$, где α_n — некоторый "поправочный" коэффициент. Наиболее известным примером метода, реализующего описанную концепцию, является алгоритм SquareLev.R [2].

Обозначим:

— $\bar{\mathbf{z}}$ — выборочное среднее компонент вектора $\mathbf{z}$, $\langle \mathbf{z}, \mathbf{z}' \rangle$ — скалярное произведение векторов $\mathbf{z}$ и $\mathbf{z}'$, $\|\mathbf{z}\| = \sqrt{\langle \mathbf{z}, \mathbf{z} \rangle}$ — евклидову норму вектора $\mathbf{z}$,

— f_n — ансамбль, полученный на n-й итерации алгоритма и состоящий из аппроксиматоров $g_1, \ldots, g_n$,

— $\mathbf{g} = g_n(X_m)$ — вектор прогнозов базового аппроксиматора g_n и $\mathbf{r} = Y_m - f_n(X_m)$ — вектор остатков, полученные на n-й итерации алгоритма.

Утверждения о свойствах алгоритма SquareLev.R доказаны в [2] при условии выполнения следующих *предположений*:

— $\mathbf{B}_1$. *Можно выбрать такое число* $\epsilon_{\min} > 0$, *что для всех итераций*

$$\frac{\langle (\mathbf{r} - \bar{\mathbf{r}}\mathbf{1}), (\mathbf{g} - \bar{\mathbf{g}}\mathbf{1}) \rangle}{\|\mathbf{r} - \bar{\mathbf{r}}\mathbf{1}\| \, \|\mathbf{g} - \bar{\mathbf{g}}\mathbf{1}\|} \geqslant \epsilon_{\min}, \tag{14}$$

где $\mathbf{1}$ — *вектор размерности* m, *состоящий из единиц;*

— $\mathbf{B}_2$. *Можно выбрать такое число* $c > 0$, *что для всех итераций*

$$\frac{\|\mathbf{r} - \bar{\mathbf{r}}\mathbf{1}\|}{\|\mathbf{g} - \bar{\mathbf{g}}\mathbf{1}\|} \leqslant c. \tag{15}$$

Отметим, что нет никакой возможности заранее проверить выполнение этих предположений. В [2] авторы проводят серию экспериментов, чтобы проверить, выполняются ли эти предположения при построении ансамблей в некоторых конкретных задачах.

При сделанных выше предположениях для метода SquareLev.R в [2] доказана теорема.

Теорема 3. Пусть данные порождены распределением $P(x)$, $x \in \mathbf{X}$, $\mathbf{Y} = \left[-\frac{B}{2}, \frac{B}{2}\right]$, $\mathcal{F}$ — *класс функций со значениями в* $[-1, 1]$ *и псевдоразмерностью* $Pdim(\mathcal{F})$, *на каждой итерации алгоритма SquareLev.R базовая модель* $g \in \mathcal{F}$ *и выполнены предположения* $\mathbf{A}_1$ *и* $\mathbf{A}_2$. *Тогда найдется такая константа* $A > 0$, *что для всех* $\gamma \in (0, 1)$ *и* $\rho > 0$, *если SquareLev.R применен к случайной выборке размера не меньше* $m(\rho, \gamma) = \left(\frac{AK^4}{\rho^2}\right)\left(\log\left(\frac{1}{\gamma}\right) + Pdim(\mathcal{F})\left\lceil\frac{K^4}{\rho^2}\right\rceil \log\left(\frac{K^{8\left\lceil\frac{K^4}{\rho^2}\right\rceil}}{\rho^3}\right)\right)$, *где* $K = 2\max\left(c\frac{\log(B^2/2\rho)}{\epsilon^2_{\min}},\, B\right)$, *то с вероятностью не меньше* $1-\gamma$ *после* $n = \left\lceil\frac{\log(B^2/2\rho)}{\epsilon^2_{\min}}\right\rceil$ *итераций будет выполнено, что* $er_{P,f}\left(f_n + \bar{\mathbf{r}}\right) < \rho$ *для ансамбля* f_n, *т.е. ошибка аппроксимации удовлетворяет неравенству*

$$er_{P,f}\left(f_n\right) < \rho + 2\bar{\mathbf{r}}\mathbb{E}\left(f(x) - f_n(x)\right) - (\bar{\mathbf{r}})^2.$$

2. Идея метода БегБуст состоит в том, чтобы сохранить форму предсказания ансамбля в виде усреднения предсказаний всех входящих в этот ансамбль аппроксиматоров и одновременно последовательно обучать базовые аппроксиматоры не на первоначальной выборке S_m, а на некоторой разности выходных значений из выборки и предсказаний аппроксиматоров ансамбля, построенных на соответствующих предыдущих итерациях.

3. Можно обратить внимание, что формулы для необходимых объемов обучающей выборки в теоремах 3 и 2 для алгоритмов SquareLev.R и БегБуст соответственно имеют похожую структуру и одинаковый порядок зависимости от параметров ρ и γ. Более того, если передоказать теорему 3 для алгоритма SquareLev.R в случае, когда выполняются предположения $\mathbf{A}_1$ и $\mathbf{A}_2$, то для обоих методов выражение требуемого объема $m(\rho, \gamma)$ обучающей выборки принимает вид $m(\rho, \gamma) \geqslant \frac{C_1K^4}{\rho^2}\left(\ln\frac{4}{\gamma} + \frac{C_2K^4}{\rho^2}\ln\frac{C_3K^{10}}{\rho^5}\right)$, где константы C_1, C_2 и C_3 совпадают для обоих методов, $K = \max\left(c\frac{\log(B^2/2\rho)}{\epsilon^2_{\min}},\, B\right)$ для SquareLev.R и $K = B$ для БегБуст. Таким образом, алгоритму БегБуст требуется меньше наблюдений для достижения сравнимой верхней границы ошибки.

4. Существенно, что при доказательстве теорем 3 и 2 используются разные предположения:

— В случае алгоритма БегБуст накладывается ограничение на функциональный класс (предположения $\mathbf{A}_1$ и $\mathbf{A}_2$, см. (8)) и предполагается, что выполнено неравенство (12) (предположение $\mathbf{A}_3$);

— В случае алгоритма SquareLev.R предполагается существование констант $\epsilon_{\min}$ и c, для которых выполняются неравенства (14) и (15) (предположения $\mathbf{B}_1$ и $\mathbf{B}_2$).

5. Неравенство (12) фактически означает, что на каждом шаге алгоритма БегБуст базовый аппроксиматор g_k может быть "хуже" оптимального базового аппроксиматора в смысле значения среднеквадратичной ошибки на выборке S_m^k, но не более чем на величину ε. При этом алгоритм будет иметь гарантированную скорость сходимости $\frac{\Delta+\varepsilon}{k}$ (см. оценку сверху (11) ошибки аппроксимации). Естественно, на практике на каждой итерации алгоритма БегБуст делается попытка найти точный минимум среднеквадратичной ошибки и хорошо, если это удается, однако даже если ошибка построенного базового аппроксиматора больше ошибки оптимального базового аппроксиматора, алгоритм БегБуст до некоторой степени устойчив к этому. Отметим также, что требование (12) представляется менее сильным, чем требование, накладываемое на базовые модели (weak hypothesis) в алгоритме AdaBoost.R [14].

6. Для небольших размерностей входного вектора неравенство (12) принципиально может быть выполнено, если дискретизовать пространство параметров, задающих базовый аппроксиматор, и последующий подбор значений параметров проводить не с помощью градиентного алгоритма оптимизации, а выполняя полный перебор по дискретизованному множеству значений. Этот же подход устраняет влияние на результат обучения начальной инициализации значений параметров, необходимой при использовании градиентных алгоритмов оптимизации.

7. Предположение о том, что $f \in \overline{co}(G)$, означает, что возможно построить сходящуюся по интегральной норме $\|\cdot\|$ (см. определение в разделе 3) к f последовательность $f_1,\ f_2, \ldots \in co(G)$. Однако факт, что при выполнении предположения $\mathbf{A}_3$ такую сходящуюся последовательность можно построить и по норме $\|\cdot\|_{X_m}$ для любой выборки S_m и любой функции $f \in \overline{co}(G)$, является нетривиальным, для доказательства этого существенно используется явный вид элементов из множества G, заданного в предположении $\mathbf{A}_1$. Кроме того, алгоритм БегБуст предлагает конструктивный способ построения такой последовательности.

8. Несмотря на то, что утверждения о свойствах алгоритма доказаны для случая, когда $g_i(x)$, $i = 1, 2 \ldots$, — это "одиночные" сигмоидальные функции из множества G (6), в расчетах часто бывает эффективнее использовать линейные комбинации $p > 1$ сигмоидальных функций [9, 11], т.е. функции $g_i(x)$, $i = 1, 2, \ldots$, в таком случае будут выбираться из множества $co(G)$ (7) и иметь вид (3). Вообще говоря, на практике рассматриваемые алгоритмы построения ансамблей могут использовать любой тип базовых моделей или даже комбинацию нескольких типов базовых моделей.

9. Заметим, что применительно к ИНС построение ансамблей может быть более предпочтительным, чем прямое увеличение сложности ИНС (число сигмоид p в (3)) и последующая настройка параметров ИНС с применением методов глобального поиска (например, мультистартов), так как порождает меньше артефактов в модели и является более дешевым по времени и памяти способом повысить производительность. Например, при настройке параметров ИНС с помощью стандартного алгоритма Левенберга–Марквардта [16] наблюдается кубический по времени и квадратичный по памяти рост необходимых вычислительных ресурсов при увеличении числа сигмоид, в то время как построение ансамбля приводит только к линейному росту.

4. Экспериментальные результаты

На наборе искусственных задач построения регрессии предложенный метод Бег-Буст (BagBoost) сравнивается с основными методиками построения ансамблей. Кроме этого, проводится экспериментальное исследование свойств предложенного подхода.

4.1. Условия экспериментов

В качестве базового аппроксиматора используется однослойная искусственная нейронная сеть (ИНС) с сигмоидальной передаточной функцией (см. [11]). Структура такого аппроксиматора задается формулой (3). Стандартный метод обучения Train$(\cdot)$ ИНС устроен следующим образом [17, 18] (см. также другие примеры алгоритмов обучения базового аппроксиматора в [10, 18, 19]):

— Выборка S_m разделяется на подвыборки S_{train} и S_{val};

— Параметры $\{u_k, \theta_k\}_{k=1}^p$ настраиваются за счет минимизации $\hat{er}_{S_{train}}(g)$ с помощью градиентных алгоритмов оптимизации. При этом для получения более робастной модели g нередко вводят ограничение на коэффициенты $\{u_k\}_{k=1}^p$ вида $\sum_{k=1}^p |u_k| \leqslant B$ для некоторого заданного B (см. [11]). В рассматриваемом случае минимизация $\hat{er}_{S_{train}}(g)$ проводилась с помощью стандартного алгоритма Левенберга–Марквардта [16];

— Обобщающая способность модели g контролируется посредством ранней остановки алгоритма минимизации [11], осуществляемой при росте ошибки $\hat{er}_{S_{val}}(g)$.

Заметим, что функция ошибки $\hat{er}_{S_{train}}(g)$ не является выпуклой и результаты обучения будут сильно зависеть от начальных условий, а именно от:

— инициализации значений весов $\{u_k, \theta_k\}_{k=1}^p$;

— разбиения выборки S_m на подвыборки S_{train} и S_{val},

осуществляемых обычно случайным образом. Указанные начальные условия и представляют собой гиперпараметры w метода обучения Train$(\cdot)$ ИНС.

Таким образом, в случае ИНС существует два источника неопределенности, влияющих на построение модели g:

— случайность выборки S_m;

— выбор гиперпараметров w.

Замечание 1. При изучении теоретических свойств алгоритмов обучения обычно предполагается, что значения гиперпараметров w алгоритма обучения Train$(\cdot)$ не оказывают влияния на свойства базовых моделей, построенных с помощью этого алгоритма. Данное предположение является стандартным при анализе теоретических свойств алгоритмов машинного обучения. Например, при изучении теоретических свойств алгоритмов аппроксимации на основе ИНС не учитывается влияние на результаты обучения начальных значений весов ИНС и то, каким образом обучающая выборка была разбита на части [9].

Если не оговорено обратное, то размер скрытого слоя будем фиксировать равным $p = 10$ нейронам. При обучении ИНС выборка S_m разбивается на обучающую S_{train} и валидационную S_{val} части в соотношении 80 % к 20 %. Остановка обучения производится при росте ошибки на валидационной части в течение 5 итераций алгоритма обучения. Для экспериментов использовалась реализация ИНС, доступная в Neural Network Toolbox пакета MatLab.

Эксперименты проводились на наборе из 19 функций, обладающих различными особенностями [20]:

— Простые гладкие функции (например, квадратичная),

— Сложные гладкие функции (например, Mystery, Michalewicz),

— Сложные гладкие функции, имеющие периодическую структуру (например, функции Rastrigin, Schwefel),

— Функции с разрывом,

— Функции, демонстрирующие характерное поведение зависимостей, возникающих в аэродинамике.

Обучающей выборкой ($train$) будем называть выборку, используемую при построении модели. Проверочной выборкой ($test$) будем называть независимую выборку большого размера, недоступную при построении модели, которая, однако, может быть использована для оценки качества работы метода, характеризуемого среднеквадратичной ошибкой (СКО) (2).

Отметим, что в экспериментах есть два фактора, привносящих случайность в результаты:

— Начальная случайная инициализация параметров ИНС (использовался алгоритм инициализации SCAWI [21], прописанный в настройках алгоритма обучения ИНС по умолчанию),

— Случайно генерируемая обучающая выборка и ее случайное разбиение на подвыборки, одна из которых используется для настройки параметров аппроксиматора, а другая — в критерии остановки обучения.

Далее, если речь идет о случайных запусках, то имеется в виду, что для каждого запуска были заново сгенерированы обучающая выборка и ее разбиение на подвыборки, а также заново инициализированы параметры аппроксиматора.

4.2. Кривые Долана–Мора

Для сравнения результатов работы методов будут рисоваться кривые Долана–Мора [22]. Опишем методологию сравнения методов построения ансамблей на основе кривых Долана–Мора, используя обозначения:

— A_k, $k = 1, \ldots, K$, — сравниваемые методы построения ансамблей,

— T_l, $l = 1, \ldots, L$, — различные задачи,

— $e(A_k, T_l)$ — среднеквадратичная ошибка (2) метода A_k на задаче T_l,

— $\widetilde{e}(T_l) = \min_k e(A_k, T_l)$ — минимальная ошибка среди всех методов построения ансамблей на задаче T_l.

Для рассматриваемого метода построения ансамблей A_k и масштабного множителя $a \geqslant 1$ определим величину $P_k(a) = \frac{\#\{l:\, e(A_k,T_l) \leqslant a \cdot \widetilde{e}(T_l)\}}{L}$, где $\#(M)$ — мощность множества M. Фактически, величина $P_k(a)$ показывает, на какой части задач ошибки аппроксимации, полученной с помощью метода построения ансамблей A_k, не больше чем в a раз минимальных (среди всех рассматриваемых методов построения ансамблей) ошибок для соответствующих задач. При этом чем выше кривая $P_k(a)$ и чем быстрее она выходит на единицу, тем более высокое качество имеет соответствующий метод построения ансамблей. Величина $P_k(1)$ равна доле задач, на которой метод построения ансамблей A_k имеет наименьшую ошибку.

4.3. Проверка справедливости теоретических предположений

Итак, согласно теореме 1 для обеспечения гарантированной скорости сходимости на каждом шаге алгоритма БегБуст достаточно построить базовый аппроксиматор g_n, который может быть "хуже" оптимального базового аппроксиматора в смысле значения среднеквадратичной ошибки не более чем на ε (см. неравенство (12)). При этом ошибка на обучающей выборке будет убывать в зависимости от номера итерации n как $\sim \frac{1}{n}$. Приведем типичный пример того, что при применении алгоритма БегБуст на практике действительно можно наблюдать такое поведение ошибки на обучающей выборке. Как и в доказанных теоремах 1 и 2, базовую модель будем выбирать из подмножества G (6) (сдвиги/растяжения сигмоидной функции), а качество работы метода будем оценивать с помощью среднеквадратичной ошибки (СКО).

На рис. 1 приведены графики зависимости ошибки на обучающей и проверочной выборках от номера итерации алгоритма БегБуст (изображены черной сплошной и серой штриховой линиями соответственно), усредненные по 5 запускам, и контрольная кривая $\frac{\delta}{n}$ (изображена черной пунктирной линией). Видно, что кривая ошибки на обучающей выборке довольно близка по поведению к контрольной кривой, в том числе ясно видно постепенное убывание ошибки до нуля с ростом числа итераций. Заметим, что в случае обучающей выборки размера $m = 100$ точек ошибка на проверочной выборке сначала убывает, а потом стабилизируется на некотором ненулевом уровне. В случае же обучающей выборки размера $m = 1000$ точек ошибка на проверочной выборке убывает на протяжении всех итераций алгоритма БегБуст и фактически неотличима от значения ошибки на обучающей выборке. Эти наблюдения соответствуют сути утверждения теоремы 2. Рис. 1

Таким образом, несмотря на то, что для доказательства теорем 1 и 2 потребовалось ввести ряд ограничивающих предположений, утверждения этих теорем оказываются выполнены на практике.

4.4. Сравнение метода БегБуст со стандартными методами построения ансамблей

Представим результаты сравнения работы предложенного подхода с современными методами построения ансамблей. Для сравнения были рассмотрены следующие алгоритмы построения ансамблей: Bagging [1, 4], Gradient boosting [5], AdaBoost.R [6], SquareLev.R [2].

На рис. 2 приведено сравнение результатов работы методов для разных функций и разных размеров выборки. Результаты работы усреднялись по 10-и запускам. Цифра после названия метода построения ансамбля, указанного на рис. 2, задает количество итераций соответствующего метода построения ансамбля. Как можно видеть, для всех методов применение ансамблей снижает разброс результатов по запускам, кроме того, в случае большой выборки это также позволяет существенно уменьшить среднюю ошибку. Рис. 2

На рис. 3 и 4 приведено сравнение методов построения ансамблей, проведенное на всех 19-и тестовых функциях. При визуализации кривых Долана–Мора $P_k(a)$ по оси абсцисс отложен десятичный логарифм от a. Количество итераций для каждого из методов равнялось 20. Рис. 3

Согласно полученным результатам, например, в случае размера выборки $m =$ Рис. 4

1000 на $\sim$ 55 процентах задач предложенный алгоритм БегБуст позволяет получить наилучшее качество аппроксимации. Отметим также, что профиль $P(a)$, соответствующий предложенному алгоритму БегБуст, лежит выше профилей остальных методов построения ансамблей и достаточно быстро сходится к единице, что также говорит о более высоком качестве предложенного метода.

5. Заключение

В работе предложен метод БегБуст (BagBoost) для построения ансамблей, при некоторых дополнительных предположениях доказана сходимость метода. Кроме того, проведенное экспериментальное исследование и сравнение с современными методами построения ансамблей показали, что соответствующий алгоритм БегБуст

— уменьшает нестабильность аппроксимации, обусловленную случайностью в базовых аппроксиматорах и/или в выборке,

— позволяет получить более высокое качество аппроксимации.

Авторы выражают благодарность А.В. Бернштейну за плодотворные дискуссии, способствовавшие существенному улучшению качества статьи.

ПРИЛОЖЕНИЕ 1

Доказательство леммы 1. Сначала докажем, что $\forall \varepsilon > 0 \quad \exists f^* \in co(G)$ такая, что $|y_i - f^*(x_i)| < \varepsilon$, $i = 1, \ldots, m$.

Очевидно, что для любого $\epsilon > 0$ можно построить такое разбиение компактного множества $\Theta = \cup_{i=1}^{n(\epsilon)} \Theta_i$ на непересекающиеся множества Θ_i, $i = 1, \ldots, n(\epsilon)$, что для любого $\theta \in \Theta$ найдется $i = i(\theta) \in \{1, 2, \ldots, n(\epsilon)\}$, и для всех $\theta' \in \Theta_i$ будет выполнено неравенство $\|\theta - \theta'\| \leqslant \epsilon$.

Произвольным образом выберем в каждом множестве Θ_i, $i = 1, \ldots, n(\epsilon)$, элемент θ_i, $i = 1, \ldots, n(\epsilon)$. Положим $f_\epsilon(x) = \sum_{i=1}^{n(\epsilon)} \psi(x, \theta_i)\mu(\Theta_i)$. Заметим, что $\sum_{i=1}^{n(\epsilon)} |\mu(\Theta_i)| \leqslant \sum_{i=1}^{n(\epsilon)} (\mu^+(\Theta_i) + \mu^-(\Theta_i)) = \sum_{i=1}^{n(\epsilon)} \int_{\Theta_i} |\mu(d\theta)| = \int_\Theta |\mu(d\theta)| = B$, т.е. $f_\epsilon \in co(G)$. При этом $\sup_{x \in \mathbf{X}} |f(x) - f_\epsilon(x)| = \sup_{x \in \mathbf{X}} \left| \int_\Theta \psi(x, \theta)\mu(d\theta) - \sum_{i=1}^{n(\epsilon)} \psi(x, \theta_i)\mu(\Theta_i) \right| = \sup_{x \in \mathbf{X}} \left| \sum_{i=1}^{n(\epsilon)} \int_{\Theta_i} (\psi(x, \theta) - \psi(x, \theta_i))\, \mu(d\theta) \right| \leqslant \sum_{i=1}^{n(\epsilon)} \int_{\Theta_i} \sup_{x \in \mathbf{X}} |\psi(x, \theta) - \psi(x, \theta_i)|\, |\mu(d\theta)|$.

Найдется такой вектор параметров $\theta^* = (\beta^*, \beta^{*,0}) \in \Theta_i$, для которого

$$
\begin{aligned}
&|\psi(x, \theta) - \psi(x, \theta_i)| = \left|\sigma\left(\beta^{\mathrm{T}} \cdot x + \beta^0\right) - \sigma\left(\beta_i^{\mathrm{T}} \cdot x + \beta_i^0\right)\right| = \\
&= \left|\sigma'\left((\beta^*)^{\mathrm{T}} \cdot x + \beta^{*,0}\right)\right| \cdot \left|\left(\beta^{\mathrm{T}} - \beta_i^{\mathrm{T}}\right) \cdot x + \left(\beta^0 - \beta_i^0\right)\right| \leqslant \\
&\leqslant |\psi'_\theta(x, \theta^*)| \cdot \|(x, 1)\| \cdot \|\theta - \theta_i\| \leqslant \mathrm{const} \cdot \epsilon,
\end{aligned}
$$

где $\mathrm{const} = \sup_{x \in \mathbf{X}, \theta \in \Theta} |\psi'_\theta(x, \theta)| \cdot \|(x, 1)\| < \infty$, поскольку множества $\mathbf{X}$ и Θ — компактны и $\|\theta - \theta_i\| \leqslant \epsilon$ по определению Θ_i, т.е. для всех $\theta \in \Theta_i$ выполнено неравенство $\sup_{x \in \mathbf{X}} |\psi(x, \theta) - \psi(x, \theta_i)| \leqslant \mathrm{const} \cdot \epsilon$.

Таким образом, получаем, что $\sup_{x \in \mathbf{X}} |f(x) - f_\epsilon(x)| \leqslant \sum_{i=1}^{n(\epsilon)} \int_{\Theta_i} (\mathrm{const} \cdot \epsilon) |\mu(d\theta)| = (\mathrm{const} \cdot \epsilon) B = \varepsilon$ при достаточно малом ϵ.

Итак, найдется такая функция $f^* \in co(G)$, что выполнено неравенство

$$
\hat{er}_{S_m}(f^*) < \epsilon \tag{П.1}
$$

для заданного $\epsilon > 0$, при этом $f^* = \sum_{i=1}^k \gamma_i g_i^*$, где $g_i^* \in G$, $\sum_{i=1}^k \gamma_i = 1$, $\gamma_i \geqslant 0$.

Значит, для произвольных $g \in G$ и $h \in co(G)$ справедливо неравенство

$$\hat{er}_{S_m}(\alpha h + (1-\alpha)g) \leqslant \|\alpha(f-h) + (1-\alpha)(f^*-g)\|^2_{X_m} + \epsilon. \tag{П.2}$$

Очевидно, что

$$\begin{aligned} &\|\alpha(f-h) + (1-\alpha)(f^*-g)\|^2_{X_m} = \\ &= \alpha^2 \hat{er}_{S_m}(h) + (1-\alpha)^2 \|f^*-g\|^2_{X_m} + 2\alpha(1-\alpha)\langle f-h, f^*-g\rangle_{X_m}, \end{aligned} \tag{П.3}$$

где $\langle f, f'\rangle_{X_m} = \frac{1}{m}\sum_{i=1}^m f(x_i)f'(x_i)$ обозначает "выборочное" скалярное произведение.

Пусть g случайно выбирается из множества $\{g_1^*, \ldots, g_k^*\} \subset G$ с вероятностью $P(g = g_i^*) = \gamma_i$, тогда среднее значение последних двух слагаемых в (П.3) равно

$$\begin{aligned} &\sum_{i=1}^k \gamma_i \left((1-\alpha)^2 \|f^*-g_i^*\|^2_{X_m} + 2\alpha(1-\alpha)\langle f-h,\, f^*-g_i^*\rangle_{X_m}\right) = \\ &= (1-\alpha)^2 \sum_{i=1}^k \gamma_i \|f^*-g_i^*\|^2_{X_m} + 0 = (1-\alpha)^2 \left(\sum_{i=1}^k \gamma_i \|g_i^*\|^2_{X_m} - \|f^*\|^2_{X_m}\right) \leqslant \\ &\leqslant (1-\alpha)^2 \left(\sup_{g\in G} \|g\|^2_{X_m} - \|f^*\|^2_{X_m}\right). \end{aligned}$$

Так как это неравенство выполнено для среднего значения, то найдется такое $g \in \{g_1^*, \ldots, g_k^*\}$, что

$$\|\alpha(f-h) + (1-\alpha)(f^*-g)\|^2_{X_m} \leqslant \alpha^2 \hat{er}_{S_m}(h) + (1-\alpha)^2 \left(\sup_{g\in G} \|g\|^2_{X_m} - \|f^*\|^2_{X_m}\right).$$

Тогда, используя неравенство треугольника, из (П.1) получаем неравенство $\|f^*\|^2_{X_m} > \hat{er}_{S_m}(0) - \epsilon$. Учитывая (П.2) и устремляя ϵ к нулю, получаем

$$\inf_{g\in G} \hat{er}_{S_m}(\alpha h + (1-\alpha)g) \leqslant \alpha^2 \hat{er}_{S_m}(h) + (1-\alpha)^2 \left(\sup_{g\in G} \|g\|^2_{X_m} - \hat{er}_{S_m}(0)\right),$$

что и требовалось доказать. Лемма 1 доказана.

Доказательство теоремы 1. Доказательство проведем по индукции. Для $n = 1$ из следствия 2 при $\alpha = 0$ получаем, что

$$\hat{er}_{S_m}(f_1) = \min_{g\in G} \hat{er}_{S_m}(g) + \varepsilon_1 = \Delta + \varepsilon_1.$$

Предположим, что $\hat{er}_{S_m}(f_n) \leqslant \frac{\Delta + \varepsilon_n^{\max}}{n}$. Докажем, что тогда $\hat{er}_{S_m}(f_{n+1}) \leqslant \frac{\Delta + \varepsilon_{n+1}^{\max}}{n+1}$. Применяя следствие 2 с $\alpha = \frac{n}{n+1}$, получаем, что

$$\begin{aligned} &\hat{er}_{S_m}(f_{n+1}) = \min_{g\in G} \hat{er}_{S_m}\left(\frac{n}{n+1}f_n + \frac{1}{n+1}g\right) + \frac{\varepsilon_{n+1}}{(n+1)^2} \leqslant \\ &\leqslant \left(\frac{n}{n+1}\right)^2 \hat{er}_{S_m}(f_n) + \frac{1}{(n+1)^2}\Delta + \frac{\varepsilon_{n+1}}{(n+1)^2} \leqslant \\ &\leqslant \left(\frac{n}{n+1}\right)^2 \cdot \frac{\Delta + \varepsilon_n^{\max}}{n} + \frac{1}{(n+1)^2}\Delta + \frac{\varepsilon_{n+1}}{(n+1)^2} \leqslant \frac{\Delta + \varepsilon_{n+1}^{\max}}{n+1}, \end{aligned}$$

что и требовалось доказать. Теорема 1 доказана.

Доказательство теоремы 2. Зафиксируем ρ и γ. Заметим, что через $n = \left\lceil \frac{2\delta}{\rho} \right\rceil$ шагов алгоритма БегБуст согласно теореме 1 ошибка аппроксимации, оценённая по выборке S_m, будет удовлетворять неравенству $\hat{er}_{S_m}(f_n) \leqslant \frac{\rho}{2}$. При этом если значения базовых аппроксиматоров g_i, $i = 1, \ldots, n$, лежат в отрезке $[-B, B]$, то и значения ансамбля f_n, полученного на n-м шаге алгоритма БегБуст, по построению будут лежать в том же отрезке. Необходимо ограничить вероятность того, что ошибка $er_{P,f}(f_n)$, оценивающая обобщающую способность ансамбля f_n, существенно больше ошибки $\hat{er}_{S_m}(f_n)$.

Чтобы использовать стандартные оценки из [9], необходимо сначала масштабировать диапазон $[-B, B]$ значений функций $f \in \mathcal{F}$ в отрезок $[0, 1]$, где $\mathcal{F} = \overline{co}(G)$. Для этого применим преобразование $f \to \tilde{f}$, где $\tilde{f} = \frac{f}{2B} + \frac{1}{2}$. Обозначим через $\tilde{\mathcal{F}}$ получившийся после масштабирования класс функций, через $\tilde{S}_m = (X_m, \tilde{Y}_m)$ — масштабированную выборку S_m, где $X_m = \{x_i,\, i = 1, \ldots, m\}$, $\tilde{Y}_m = \{\tilde{y}_i,\, i = 1, \ldots, m\}$, $\tilde{y}_i = \tilde{f}(x_i)$. Заметим, что для $f \in co(G)$ коэффициенты выпуклой оболочки (см. (7)) соответствующей масштабированной функции $\tilde{f}$ удовлетворяют неравенству $\sum_{k=1}^{p} |\tilde{\alpha}_k| \leqslant 1$.

Очевидно, что следующие неравенства эквивалентны:

$$\left| er_{P,\tilde{f}}\left(\tilde{f}_n\right) - \hat{er}_{\tilde{S}_m}\left(\tilde{f}_n\right) \right| \geqslant \rho \Leftrightarrow \left| er_{P,f}\left(f_n\right) - \hat{er}_{S_m}\left(f_n\right) \right| \geqslant \rho \cdot 4B^2.$$

Положим $\frac{\tilde{\rho}}{2} = \frac{1}{4B^2}\frac{\rho}{2}$. Теперь можно применить теорему 17.1 из [9]:

$$\begin{aligned} &\mathrm{P}\left(\exists f \in \mathcal{F} :\, \left| er_{P,f}\left(f_n\right) - \hat{er}_{S_m}\left(f_n\right) \right| \geqslant \frac{\rho}{2}\right) = \\ &= \mathrm{P}\left(\exists \tilde{f} \in \tilde{\mathcal{F}} :\, \left| er_{P,\tilde{f}}\left(\tilde{f}_n\right) - \hat{er}_{\tilde{S}_m}\left(\tilde{f}_n\right) \right| \geqslant \frac{\tilde{\rho}}{2}\right) \leqslant 4N_1\left(\frac{\tilde{\rho}}{2^5}, \tilde{\mathcal{F}}, 2m\right) \exp\left(\frac{-\tilde{\rho}^2 m}{2^7}\right), \end{aligned}$$

где N_1 — покрывающее число с метрикой в L_1 (см. раздел 10.4 из [9]), т.е. $N_1(\epsilon, \mathcal{F}, m)$ — это минимальное количество шаров радиуса ϵ, которыми можно покрыть множество $f(x), x \in X_m$, для любой $f \in \mathcal{F}$ и любого набора входных векторов $X_m = \{x_i, i = 1, 2, \ldots, m\}$

Так как $N_1\left(\frac{\tilde{\rho}}{2^5}, \tilde{\mathcal{F}}, 2m\right) \leqslant N_2\left(\frac{\tilde{\rho}}{2^5}, \tilde{\mathcal{F}}, 2m\right)$ (лемма 10.5 в [9]), то применяя теорему 14.15 из [9], получаем неравенство

$$\mathrm{P}\left(\exists \tilde{f} \in \tilde{\mathcal{F}} :\, \left| er_{P,\tilde{f}}\left(\tilde{f}_n\right) - \hat{er}_{\tilde{S}_m}\left(\tilde{f}_n\right) \right| \geqslant \frac{\tilde{\rho}}{2}\right) \leqslant 4\left(\frac{2^7 em}{\tilde{\rho}(d+1)}\right)^{\frac{5120(d+3)}{\tilde{\rho}^2}} \exp\left(\frac{-\tilde{\rho}^2 m}{2^7}\right).$$

Эта вероятность будет меньше γ, если $\gamma > 4\left(\frac{2^7 em}{\tilde{\rho}(d+1)}\right)^{\frac{5120(d+3)}{\tilde{\rho}^2}} \exp\left(\frac{-\tilde{\rho}^2 m}{2^7}\right)$, откуда, положив $q = \frac{5120(d+3)}{\tilde{\rho}^2}$, получаем неравенство

$$\frac{\tilde{\rho}^2 m}{2^7} > q\left(\ln m + \ln\frac{2^7 e}{\tilde{\rho}(d+1)}\right) + \ln\frac{4}{\gamma}. \tag{П.4}$$

Так как $ab \geqslant \ln a + \ln b + 1$, то

$$\frac{m\tilde{\rho}^2}{2^8} \geqslant q\ln m + q\ln\frac{\tilde{\rho}^2}{q2^8}. \tag{П.5}$$

Перепишем неравенство (П.4) в виде $\frac{m\tilde{\rho}^2}{2^8} \geqslant q\ln m + q\ln\frac{\tilde{\rho}^2}{q2^8} - \frac{m\tilde{\rho}^2}{2^8} + q\ln\frac{2^7 e}{\tilde{\rho}(d+1)} - q\ln\frac{\tilde{\rho}^2}{2^8 q} + \ln\frac{4}{\gamma}$, Учитывая (П.5), получаем, что неравенство (П.4) будет заведомо выполнено, если $\frac{m\tilde{\rho}^2}{2^8} \geqslant q\ln\frac{2^7 e}{\tilde{\rho}(d+1)} - q\ln\frac{\tilde{\rho}^2}{2^8 q} + \ln\frac{4}{\gamma}$, откуда получаем ограничение на размер выборки $m \geqslant \frac{2^8}{\tilde{\rho}^2}\left(q\ln\frac{2^{15}eq}{\tilde{\rho}^3(d+1)} + \ln\frac{4}{\gamma}\right) = \frac{C_1}{\rho^2}\left(\ln\frac{4}{\gamma} + \frac{C_2}{\rho^2}\ln\frac{C_3}{\rho^5}\right)$, где $C_1 = 2^{12}B^4$, $C_2 = 5\cdot 2^{14}(d+3)B^4$ и $C_3 = \frac{2^{21}eB^6}{(d+1)}C_2$, что и требовалось доказать. Теорема 2 доказана.

СПИСОК ЛИТЕРАТУРЫ

1. *Breiman L.* Bagging predictors // Machine Learning. 1996. V. 24. No. 2. P. 123–140.

2. *Duffy N., Helmbold D.* Boosting Methods for Regression // Machine Learning. 2002. V. 47. No. 2. P. 153–200.

3. *Ueda N., Nakano R.* Generalization Error of Ensemble Estimators // Proc. Int. Conf. on Neural Networks. 1996. P. 90–95.

4. *Bruhlmann P., Yu B.* Explaining Bagging // Research Report. No. 92. Statistics Department, University of California at Berkeley, 2000.

5. *Friedman J.* Greedy Function Approximation: A Gradient Boosting Machine // Annals of Statistics. 2001. V. 29. No. 5. P. 1189–1232.

6. *Shrestha D. L., Solomatine D. P.* Experiments with AdaBoosting.RT, an Improved Boosting Scheme for Regression // Neural Computation. 2006. V. 18. No. 7. P. 1678–1710.

7. *Вапник В. Н., Червоненкис А. Я.* О методе упорядоченной минимизации риска // АиТ. 1974. № 8. С. 21–31; 1974. № 9. С. 29–40.

   Vapnik V. N., Chervonenkis A. Ja. Ordered Risk Minimization (I and II) // Autom. Remote Control. 1974. V. 34. P. 1226–1235; 1974. V. 34. P. 1403–1412.

8. *Вапник В. Н., Михаильский А. И.* О поиске зависимостей методом упорядоченной минимизации риска // АиТ. 1974. № 10. С. 86–97.

   Vapnik V. N., Michalski A. I. The Search for Dependencies by the Method of Ordered Minimization of Risk // Autom. Remote Control. 1974. V. 35. № 10. part 1. P. 1615–1624.

9. *Anthony M., Bartlett P. L.* Neural Network Learning: Theoretical Foundations. Cambridge: Cambridge University Press, 2002.

10. *Burnaev E., Belyaev M., Prihodko P.* About Hybrid Algorithm for Tuning of Parameters in Approximation Based on Linear Expansions in Parametric Functions // Proc. 8th Int. Conf. “Intelligent Information Processing”. Republic of Cyprus, Paphos. October 17–24. 2010. P. 200–205.

11. *Hastie T., Tibshirani R., Friedman J.* The Elements of Statistical Learning: Data Mining, Inference and Prediction. NY: Springer, 2008.

12. *Pinkus A.* Approximation Theory of the MLP Model in Neural Networks // Acta Numerica. V. 8. Cambridge: Univ. Press, 1999. P. 143–195.

13. *Zhou Z., Wu J., Tang W.* Ensembling Neural Networks: Many Could be Better Than All // Artificial Intelligence. 2002. V. 137 No. 1. P. 239–263.

14. *Freund Y., Schapire R.* A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting // Proc. Second Eur. Conf. Comput. Learning Theory. 1995. P. 23–37.

15. *Zhou Z. H.* Ensemble Methods: Foundations and Algorithms. Boca Raton, FL: Chapman & Hall/CRC, 2012.

16. *Marquardt D.* An Algorithm for Least-Squares Estimation of Nonlinear Parameters // SIAM J. Applied Mathematics. 1963. V. 11. No. 2. P. 431–441.

17. *Хайкин С.* Нейронные сети. Полный курс. М.: ООО "И. Д. Вильямс", 2006.

18. Neural Network Toolbox Documentation. The MathWorks, Inc. http://www.mathworks.com/help/nnet/index.html

19. *Беляев М. Г., Любин А. Д.* Особенности оптимизационной задачи, возникающей при построении аппроксимации многомерной зависимости // Тр. конф. "Информационные технологии и системы" (ИТиС'11). 2011. С. 415–422

20. *Global Optimization Test Problems.* http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page364.htm

21. *Drago G. P., Ridella S.* Statistically Controlled Activation Weight Initialization (SCAWI) // IEEE Trans. Neural Networks. 1992. V. 3. No. 4. P. 627–631.

22. *Dolan E. D., More J.* Benchmarking Optimization Software with Performance Profiles // Math. Programming manuscript. 2002. V. 91. No. 2. P. 201–213.

23. *Friedman J., Hastie T., Tibshirani R.* Additive Logistic Regression: a Statistical View of Boosting // Annals of Statistics. 2001. V. 28. No. 2. P. 337–407.

24. *Friedman J.* Greedy Function Approximation: A Gradient Boosting Machine // Annals of Statistics. 2001. V. 29. No. 5. P. 1189–1232.

Бурнаев Е. В., *ИППИ РАН, зав. сектором, Москва*, burnaev@iitp.ru
Приходько П. В., *ИППИ РАН, мнс, Москва*, prikhodkop@gmail.com

Подписи к рисункам

1. Зависимость ошибок на обучающей (*черная сплошная линия*) и проверочной (*серая штриховая линия*) выборках от числа итераций для обучающей выборки объема $m = 100$ (слева) и $m = 1000$ (справа), функция Branin. Для сравнения приведена кривая $\frac{\delta}{n}$ (*черная пунктирная линия*).

2. Разброс СКО для разных методов построения ансамблей в случае функции Six-hump camel back, объем выборки $m = 500$ точек.

3. Кривые Долана–Мора для сравнения различных методов построения ансамблей, объем выборки $m = 50$ точек: кривая *1* – BagBoost, кривая *2* – Bagging, кривая *3* – AdaBoost.R2, кривая *4* – GradBoost, кривая *5* – SqLev.R, кривая *6* – OneANN.

4. Кривые Долана–Мора для сравнения различных методов построения ансамблей, объем выборки $m = 1000$ точек: кривая *1* – BagBoost, кривая *2* – SqLev.R, кривая *3* – GradBoost, кривая *4* – AdaBoost.R2, кривая *5* – Bagging, кривая *6* – One ANN.

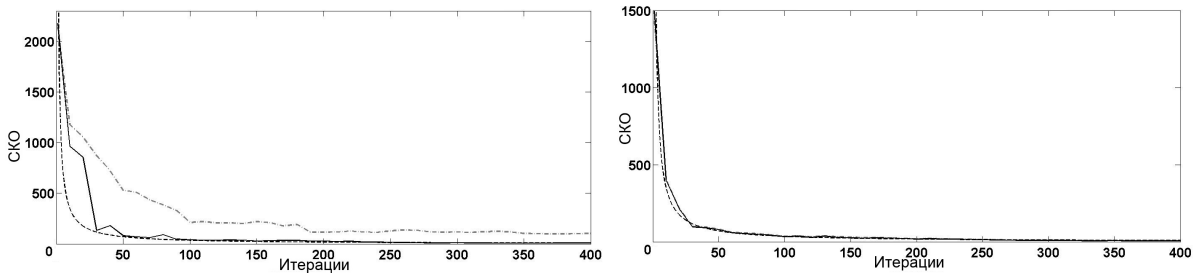

Рис. 1.

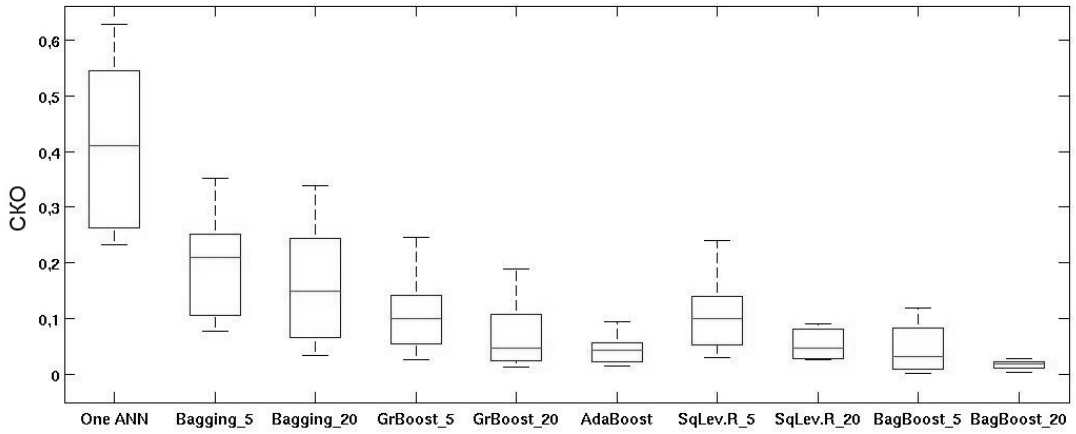

Рис. 2.

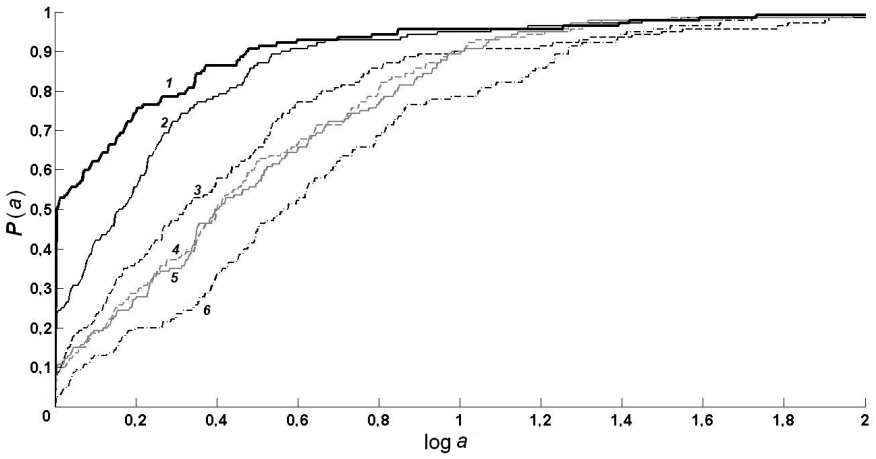

Рис. 3.

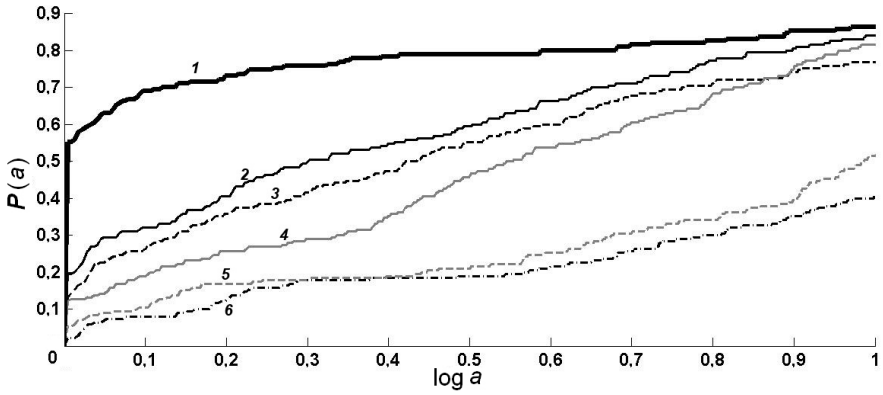
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
P(a)
1
2
3
4
5
6
0,1
0,2
0,3
0,4
0,5
0,6
0,7
0,8
0,9
1
log a

Рис. 4.