ЭФФЕКТЫ РАССЛОЕНИЯ И СХОДСТВА В СЕМЕЙСТВАХ АЛГОРИТМОВ И ИХ ВЛИЯНИЕ НА ВЕРОЯТНОСТЬ ПЕРЕОБУЧЕНИЯ*

К. В. Воронцов
vokov@forecsys.ru, http://www.ccas.ru/voron

Computing Centre RAS, Vavilov st. 40, 119333 Moscow, Russia

Аннотация

Показано, что численно точные оценки вероятности переобучения возможно получить только путём совместного учёта двух свойств семейства алгоритмов: расслоения по уровням ошибок и сходства алгоритмов. Для семейства, состоящего только из двух алгоритмов, получена точная оценка вероятности переобучения и показано, что даже в этом простейшем случае возникает переобучение и проявляются эффекты расслоения и сходства, снижающие вероятность переобучения. Для более сложного случая — цепочки алгоритмов — проведён эксперимент, в котором удалось раздельно оценить влияние расслоения и сходства. Показано, что приемлемо низкие вероятности переобучения возможны только для тех семейств, которые обладают обоими свойствами.

Получение достаточно точных верхних оценок вероятности переобучения является одной из основных проблем в теории статистического обучения (statistical learning theory). Она остаётся открытой уже более сорока лет, начиная с появления VC-теории В. Н. Вапника и А. Я. Червоненкиса [16, 15]. Наиболее точные из известных оценок всё ещё сильно завышены [10]. Завышенность приводит к необоснованному требованию увеличивать длину обучающей выборки до 10^5–10^8 объектов [17], а в методе структурной минимизации риска — к чрезмерному упрощению алгоритмов [8]. Известные оценки лишь на качественном уровне описывают связь переобучения со сложностью семейства алгоритмов, но не всегда подходят для точных количественных предсказаний и управления процессом обучения. Остаётся открытым вопрос, не связано ли переобучение с какими-то более тонкими и пока не изученными явлениями.

Данное исследование направлено на выявление причин завышенности и поиск путей улучшения оценок. Показано, что вероятность переобучения существенно зависит не только от сложности семейства (числа различных алгоритмов в нём), но ещё

*Работа выполнена при поддержке РФФИ (проект № 08-07-00422) и программы ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения».

и от степени их различности. Для получения точных оценок необходимо одновременно учесть два эффекта: степень сходства алгоритмов в семействе и расслоение семейства по уровням частоты ошибок. Пренебрежение одним из этих эффектов сводит на нет все усилия, направленные на учёт второго. Данный вывод косвенно подтверждается и тем, что известные попытки учесть эти факторы по отдельности [2, 14, 10] не приносили радикального улучшения точности.

В разделе 1 вводятся необходимые понятия и определения, в том числе слабая (перестановочная) вероятностная аксиоматика. Раздел 2 носит обзорный характер; в нём приводятся оценки Вапника-Червоненкиса и некоторые их улучшения, связанные с попытками учесть различность алгоритмов. В разделе 3 выводится точная комбинаторная оценка вероятности переобучения для семейства, состоящего из двух алгоритмов. Это простейший частный случай, в котором уже наблюдается переобучение, причём усиление свойств расслоения и сходства снижает вероятность переобучения. В разделе 4 рассматриваются семейства алгоритмов специального вида, называемые цепочками. Модельные эксперименты показывают, что получение численно точных оценок вероятности переобучения возможно только путём совместного учёта эффектов расслоения и сходства алгоритмов.

1 Задача оценивания вероятности переобучения

Задано конечное множество $\mathbb{X} = \{x_1, \ldots, x_L\}$, называемое *полной* или *генеральной* выборкой. Элементы множества $\mathbb{X}$ называются *объектами*. Задано множество $\mathbb{A}$, элементы которого называются *алгоритмами*. Существует бинарная функция $I\colon \mathbb{A}\times\mathbb{X} \to \{0,1\}$, называемая *индикатором ошибки*. Если $I(a,x) = 1$, то говорят, что алгоритм a допускает ошибку на объекте x.

Числом ошибок алгоритма a на выборке $X \subseteq \mathbb{X}$ называется величина

$$n(a,X) = \sum_{x\in X} I(a,x).$$

Частотой ошибок или *эмпирическим риском* алгоритма a на выборке X называется величина $\nu(a,X) = \frac{1}{|X|}n(a,X)$. Она принимает значения из отрезка $[0,1]$.

Обозначим через $\mathbb{X}_L^\ell$ множество всех ℓ-элементных подмножеств генеральной выборки $\mathbb{X}$. Очевидно, $|\mathbb{X}_L^\ell| = C_L^\ell$.

Методом обучения называется отображение $\mu\colon \mathbb{X}_L^\ell \to \mathbb{A}$, которое произвольной *обучающей выборке* $X \in \mathbb{X}_L^\ell$ ставит в соответствие некоторый алгоритм $a = \mu X$ из $\mathbb{A}$.

Методом *минимизации эмпирического риска* называется метод обучения вида

$$\mu X = \arg\min_{a\in\mathbb{A}} n(a,X). \tag{1.1}$$

Поясним введённые понятия. В задачах классификации алгоритм — это вычислимая функция $a\colon \mathbb{X} \to \mathbb{Y}$, которая каждому объекту x из $\mathbb{X}$ ставит в соответствие номер класса из заданного конечного множества $\mathbb{Y}$; индикатор ошибки имеет вид

$$I(a,x) = \big[a(x) \neq y(x)\big],$$

где $y\colon \mathbb{X} \to \mathbb{Y}$ — неизвестная функция классификации. Здесь и далее квадратные скобки — это нотация Айверсона [5], переводящая логическое значение в число 0 или 1

по правилам $[\mathit{истина}] = 1$, $[\mathit{ложь}] = 0$. В роли множества $\mathbb{A}$ выступает некоторое параметрическое семейство алгоритмов, например разделяющие гиперплоскости, нейронные сети или решающие деревья [6]. Метод обучения μ настаивает параметры алгоритма по заданной обучающей выборке X с известными классификациями $y_i = y(x_i)$. Говорят также, что метод μ *восстанавливает зависимость* $y(x)$ по эмпирическим данным $(x_i, y_i)_{i=1}^{\ell}$. Примерами широко известных методов обучения являются: машины опорных векторов SVM для линейных разделителей; обратное распространение ошибок BackProp для нейронных сетей; C4.5 для решающих деревьев [6].

В задачах восстановления регрессионных зависимостей алгоритмы — это, как правило, функции вида $a\colon \mathbb{X} \to \mathbb{R}$; индикатор ошибки может задаваться в виде

$$I(a, x) = \big[|a(x) - y(x)| \geqslant \delta\big],$$

где $y\colon \mathbb{X} \to \mathbb{Y}$ — неизвестная функция регрессии, δ — порог ошибки.

Для целей данного исследования нет необходимости конкретизировать понятие «алгоритма». Достаточно считать алгоритмы элементами некоторого абстрактного множества $\mathbb{A}$, предполагая лишь, что существует способ определить, допускает ли алгоритм a ошибку на объекте x. Такое понимание «алгоритма» с одной стороны расширяет класс рассматриваемых задач, но с другой стороны ограничивает его теми задачами, в которых не важна величина ошибки.

Уклонением частот ошибок алгоритма a на двух выборках X и $\bar{X} = \mathbb{X} \setminus X$ будем назвать разность частот $\delta(a, X) = \nu(a, \bar{X}) - \nu(a, X)$.

Переобученностью метода μ на выборке X будем называть уклонение частот ошибок алгоритма $a = \mu(X)$:

$$\delta_\mu(X) = \delta\big(\mu(X), X\big) = \nu\big(\mu(X), \bar{X}\big) - \nu\big(\mu(X), X\big).$$

Будем говорить, что метод μ *переобучен* на выборке X, если $\delta_\mu(X) \geqslant \varepsilon$, где ε — параметр, называемый *порогом переобучения*.

Обычно термин «переобучение» вводится неформально и обозначает часто встречающееся на практике нежелательное явление, когда алгоритм, настроенный по обучающей выборке, заметно хуже ведёт себя на новых контрольных данных. Здесь этому термину придаётся более строгий формальный смысл.

Будем придерживаться *слабой вероятностной аксиоматики* [18], в которой делается одно единственное вероятностное предположение. Предполагается, что все C_L^ℓ разбиений генеральной выборки $\mathbb{X}$ на *наблюдаемую* обучающую выборку X длины ℓ и *скрытую* контрольную выборку $\bar{X}$ длины $k = L - \ell$ реализуются с равной вероятностью. Данное предположение фактически эквивалентно стандартной гипотезе о независимости элементов выборки $\mathbb{X}$. Однако существование вероятностной меры на всём пространстве объектов не предполагается, и даже само это пространство не вводится. В слабой аксиоматике событиями являются подмножества разбиений выборки $\mathbb{X}$. Точнее, для произвольного предиката $\beta\colon \mathbb{X}_L^\ell \to \{\mathit{истина}, \mathit{ложь}\}$ вероятность события $\beta(X)$ определяется как доля разбиений, при которых $\beta(X)$ истинно:

$$\mathsf{P}\,[\beta(X)] = \frac{1}{C_L^\ell} \sum_{X \in \mathbb{X}_L^\ell} \big[\beta(X)\big].$$

В рамках слабой аксиоматики будем рассматривать одну из основных задач теории статистического обучения. Требуется получить как можно более точные верхние

оценки *вероятности переобучения* для заданного метода μ:

$$Q_\varepsilon(\mu, \mathbb{X}) = \mathsf{P}\left[\delta_\mu(X) \geqslant \varepsilon\right]. \tag{1.2}$$

Введение слабой аксиоматики мотивируется следующими соображениями.

Во-первых, в задачах анализа данных выборки могут быть только конечными, будь то уже известные наблюдаемые данные, или скрытые данные, которые станут известны в будущем. В некоторых задачах число предсказаний k настолько мало́, что вводить «вероятность ошибки» как предел частоты ошибок при $k \to \infty$ просто некорректно. Слабая аксиоматика позволяет получать содержательные результаты, справедливые при любых конечных ℓ и k, чисто комбинаторными методами, причём во многих случаях оценки получаются точными. Понятие «вероятность ошибки» в слабой аксиоматике вообще не определяется. Качество алгоритмов характеризуется частотой их ошибок на конечных выборках. Понятие переобученности определяется как отклонение частоты ошибок в двух подвыборках, а не как отклонение частоты ошибок от её вероятности. Заметим, что такой подход не является новым в теории статистического обучения. Первые работы Вапника и Червоненкиса [16] также основывались на оценках уклонения частот в двух подвыборках.

Во-вторых, вероятности, определяемые через «долю разбиений выборки», легко оценивать эмпирически, заменяя среднее по всем разбиениям средним по случайному подмножеству разбиений (метод Монте-Карло). Эта методика напоминает скользящий контроль [4, 9], но отличается от него тем, что оценивается не эмпирическое среднее частоты ошибок на контроле $\nu\big(\mu(X), \bar{X}\big)$, а эмпирическое распределение переобученности δ_μ. Именно это позволило в [18] выделить и численно сравнить четыре основных фактора завышенности классических оценок Вапника-Червоненкиса. Вообще, в слабой аксиоматике яснее прослеживается связь теоретических оценок с эмпирическими методиками типа перестановочных тестов или скользящего контроля.

В-третьих, оценки вида $Q_\varepsilon(\mu, \mathbb{X}) \leqslant \eta(\varepsilon)$ при необходимости легко переносятся из слабой аксиоматики в сильную (колмогоровскую). Для этого вводится дополнительное предположение, что объекты $\mathbb{X}$ выбраны случайно и независимо из некоторого неизвестного вероятностного распределения. Тогда достаточно взять математическое ожидание по полной выборке $\mathbb{X}$ от обеих частей неравенства:

$$\mathsf{P}_{\mathbb{X}}\big\{\delta_\mu(X) \geqslant \varepsilon\big\} = \mathsf{E}_{\mathbb{X}} Q_\varepsilon(\mu, \mathbb{X}) \leqslant \mathsf{E}_{\mathbb{X}}\eta(\varepsilon).$$

Если оценка $\eta(\varepsilon)$ не зависит от полной выборки $\mathbb{X}$, то она непосредственно переносится из слабой аксиоматики в сильную. Если оценка зависит от некоторой функции полной выборки $T(\mathbb{X})$, то значение этой функции надо либо интерпретировать как априорное знание, либо оценивать по наблюдаемой части выборки. Во всех этих случаях вид оценки не меняется при переходе от слабой аксиоматики к сильной. Поэтому вполне допустимо оставаться в рамках слабой аксиоматики.

2 Оценки Вапника-Червоненкиса и их улучшения

Рассмотрим сначала простейший случай, когда метод μ по любой выборке $X \subset \mathbb{X}$ строит один и тот же алгоритм $a = \mu(X)$. Фактически это означает, что никакого обучения нет. Для фиксированного алгоритма a оценивается отклонение частоты его ошибок на скрытой выборке от частоты ошибок на наблюдаемой выборке [18].

Теорема 2.1. *Пусть алгоритм a допускает m ошибок на полной выборке: $n(a, \mathbb{X}) = m$. Тогда для любого $\varepsilon \in [0, 1)$ справедлива точная оценка:*

$$\mathsf{P}\big[\delta(a, X) \geqslant \varepsilon\big] = \sum_{s=s_0}^{s_1(\varepsilon)} h_L^{\ell,m}(s) \equiv H_L^{\ell,m}(s_1(\varepsilon)), \tag{2.1}$$

где $h_L^{\ell,m}(s) = C_m^s C_{L-m}^{\ell-s}/C_L^\ell$ — гипергеометрическое распределение, $s_0 = \max\{0, m-k\}$, $s_1(\varepsilon) = \big\lfloor \frac{\ell}{L}(m - \varepsilon k)\big\rfloor$.

При $\ell, k \to \infty$ правая часть (2.1) стремится к нулю. Поэтому Теорему 2.1 можно рассматривать как аналог закона больших чисел в слабой аксиоматике. Известно много верхних оценок, описывающих скорость сходимости в законе больших чисел — неравенства Чебышёва, Бернштейна, Хёффдинга, Чернова [12]. Однако все они являются завышенными верхними оценками *точного* равенства (2.1).

Чтобы обобщить Теорему 2.1 на случай произвольного метода обучения μ, потребуется ввести ещё несколько понятий.

Вектором ошибок алгоритма a на полной выборке $\mathbb{X}$ будем называть L-мерный бинарный вектор $(a)_\mathbb{X} = \big(I(a, x_i)\big)_{i=1}^L$. Поскольку в дальнейшем нас будут интересовать не сами алгоритмы, а, главным образом, их векторы ошибок, для краткости будем использовать обозначение a вместо $(a)_\mathbb{X}$ и говорить «вектор a».

Коэффициентом разнообразия (shatter coefficient) множества алгоритмов $\mathbb{A}$ на выборке $\mathbb{X}$ называется число различных векторов ошибок $(a)_\mathbb{X}$, порождаемых всевозможными алгоритмами $a \in \mathbb{A}$.

Обозначим через A множество векторов ошибок, порождаемых алгоритмами вида $a = \mu X$ на всевозможных обучающих подвыборках X:

$$A = \big\{(\mu X)_\mathbb{X} \colon X \in \mathbb{X}_L^\ell\big\}.$$

Заметим, что мощность множества алгоритмов $\big\{\mu X \colon X \in \mathbb{X}_L^\ell\big\}$ не превосходит C_L^ℓ. Она может оказаться и строго меньше C_L^ℓ, поскольку метод μ может строить по различным выборкам совпадающие алгоритмы. Коэффициент разнообразия $|A|$ может оказаться ещё меньше, поскольку различные алгоритмы могут порождать совпадающие векторы ошибок. В общем случае $|A| \leqslant C_L^\ell$.

Множество векторов ошибок A разбивается на $L+1$ непересекающихся подмножеств $A = A_0 \cup \cdots \cup A_L$, где $A_m = \big\{a \in A \colon n(a, \mathbb{X}) = m\big\}$ — множество векторов с m ошибками. Будем говорить, что A *расслаивается по уровням ошибок*.

Последовательность коэффициентов разнообразия $|A_m|$, $m = 0, \ldots, L$, называется *профилем разнообразия* множества алгоритмов $\mathbb{A}$ на выборке $\mathbb{X}$ [18].

Чтобы получить верхние оценки вероятности переобучения, справедливые для любого метода μ, в VC-теории [16, 15] и ряде последующих работ (см. обзоры [3, 1]) вводится *принцип равномерной сходимости*. Функционал Q_ε подменяется его верхней оценкой $\widetilde{Q}_\varepsilon$ — вероятностью наибольшего уклонения частот в двух подвыборках:

$$Q_\varepsilon \leqslant \widetilde{Q}_\varepsilon = \mathsf{P}\Big[\max_{a \in A} \delta(a, X) \geqslant \varepsilon\Big]. \tag{2.2}$$

В исходных работах [16, 15] использовалась ещё более грубая оценка — максимум брался по всем алгоритмам исходного множества алгоритмов $\mathbb{A}$.

Теорема 2.2. *Если метод μ минимизирует эмпирический риск, и все векторы $a \in A$ имеют одинаковый уровень ошибок $m = n(a, \mathbb{X})$, то верхняя оценка* (2.2) *обращается в точное равенство: $Q_\varepsilon = \widetilde{Q}_\varepsilon$.*

Доказательство. Минимизация эмпирического риска $\nu(a, X) = \frac{s}{\ell}$ при фиксированном m эквивалентна максимизации переобученности $\delta(a, X) = \frac{m-s}{k} - \frac{s}{\ell} = \frac{m\ell - sL}{\ell k}$. ■

Если же множество A *расслаивается по уровням ошибок*, то неравенство (2.2) становится завышенной верхней оценкой, так как максимум переобученности достигается на алгоритмах a, у которых не только мало $s = n(a, X)$, но и велико $m = n(a, \mathbb{X})$. При решении прикладных задач эффект расслоения возникает практически всегда. Это связано с универсальностью применяемых семейств алгоритмов $\mathbb{A}$. Для каждой конкретной задачи, определяемой индикатором ошибки I и выборкой $\mathbb{X}$, лишь малая доля алгоритмов семейства имеет низкий уровень ошибок. Подавляющее большинство алгоритмов «предназначены» для других задач, и в данной задаче допускают около 50% ошибок. Эксперименты подтверждают, что распределение алгоритмов по уровням ошибок $m = 0, \ldots, L$ имеет форму узкого пика, сконцентрированного в окрестности $m = L/2$ [11, 10].

Таким образом, требование равномерной сходимости является чрезмерно сильным. Оно даёт лишь достаточное условие обучаемости.

Попытка учесть расслоение в рамках слабой аксиоматики была предпринята в [18], где была получена оценка, зависящая от профиля разнообразия $|A_m|_{m=0}^L$, а не от коэффициента разнообразия $|A|$. Ниже приводится более краткое доказательство этой же оценки. Причём здесь она выводится через принцип равномерной сходимости. С учётом теоремы 2.2 это означает, что данная оценка учитывает расслоение лишь частично, хотя и зависит от профиля разнообразия.

Теорема 2.3. *Для любых μ, $\mathbb{X}$ и $\varepsilon \in [0, 1)$ справедливы оценки:*

$$Q_\varepsilon \leqslant \sum_{m=0}^{L} |A_m| \, H_L^{\ell,m}\left(s_1(\varepsilon)\right) \leqslant \tag{2.3}$$

$$\leqslant |A| \max_{m=1,\ldots,L} H_L^{\ell,m}\left(s_1(\varepsilon)\right). \tag{2.4}$$

Доказательство. Покажем, что эти оценки справедливы для функционала $\widetilde{Q}_\varepsilon$. Оценим максимум величин $\big[\delta(a, X) \geqslant \varepsilon\big]$ их суммой (неравенство Буля), затем воспользуемся расслоением по уровням ошибок $|A| = |A_0| + \cdots + |A_L|$:

$$\widetilde{Q}_\varepsilon = \mathsf{P}\Big[\max_{a \in A} \delta(a, X) \geqslant \varepsilon\Big] = \mathsf{P} \max_{a \in A} \big[\delta(a, X) \geqslant \varepsilon\big] \leqslant$$

$$\leqslant \sum_{a \in A} \mathsf{P}\big[\delta(a, X) \geqslant \varepsilon\big] = \sum_{m=0}^{L} \sum_{a \in A_m} \mathsf{P}\big[\delta(a, X) \geqslant \varepsilon\big] =$$

$$= \sum_{m=0}^{L} |A_m| \cdot H_L^{\ell,m}\left(s_1(\varepsilon)\right) \leqslant |A| \cdot \max_m H_L^{\ell,m}\left(s_1(\varepsilon)\right).$$

■

Эмпирический анализ факторов завышенности оценки (2.4), проведённый в [18], показал, что наиболее существенны два фактора. Первый — пренебрежение эффектом расслоения; он приводит к завышению оценки в 10^3–10^5 раз. Второй — пренебрежение эффектом сходства алгоритмов; он приводит к завышению в 10^3–10^4 раз.

Остальные факторы носят технический характер, относительно легко устраняются и совместно дают завышение в 10^1–10^2 раз. В частности, третий фактор завышенности, связанный с заменой профиля разнообразия $|A_m|$ одним скалярным коэффициентом разнообразия $|A|$ (переход от (2.3) к (2.3)) оказался не столь существенным, как можно было бы ожидать.

Явление расслоения и связанные с ним оценки переобучения (shell bounds) изучались в работах Дж. Лэнгфорда [11, 10]. К сожалению, они обладают рядом недостатков. Во-первых, они довольно громоздки как для записи, так и для вычислений. Для оценивания профиля разнообразия приходится прибегать к имитационному моделированию, порождая случайное подмножество алгоритмов из $\mathbb{A}$ методом Монте-Карло. Во-вторых, они не дают радикального выигрыша в точности по сравнению с классическими VC-оценками. Другой подход связан с введением *алгоритмической функции везения* (algorithmic luckiness function), с помощью которой все алгоритмы семейства ранжируются по их предпочтительности относительно заданной выборки; затем, следуя классической VC-теории, применяется неравенство Буля и оцениваются мощности покрытия (covering numbers) [7].

Второй фактор завышенности — пренебрежение сходством алгоритмов — возникает в результате применения неравенства Буля. Эта оценка тем сильнее завышена, чем более схожи векторы ошибок алгоритмов. Влияние сходства алгоритмов на вероятность переобучения почти не изучалось, за исключением работ Э. Бакса [2] и Дж. Силла [14], в которых радикального улучшения оценок добиться не удалось.

Следуя работе Бакса [2], нетрудно уточнить Теорему 2.3, показав, что если множество векторов ошибок A кластеризуется по расстоянию Хэмминга на $S(r)$ кластеров радиуса r каждый, то

$$\mathsf{P}\left[\delta_\mu(X) \geqslant \varepsilon + \tfrac{r}{\ell}\right] \leqslant S(r) \max_m H_L^{\ell,m}\left(s_1(\varepsilon)\right).$$

В частности, в [2] показано, что если семейство $\mathbb{A}$ линейно по параметрам, то $S(r) \leqslant \frac{1}{2r+1}|A|$. К сожалению, даже после оптимизации по r эта оценка остаётся сильно завышенной.

В работах Силла [13, 14] рассматриваются параметрические семейства алгоритмов $\mathbb{A} = \left\{a(x,\gamma)\colon \gamma \in \mathbb{R}^d\right\}$, обладающие свойством *связности*, которое заключается в следующем. При непрерывном изменении любой из координат вектора параметров γ каждое изменение вектора ошибок алгоритма $a(x,\gamma)$ происходит только на одном объекте. Можно доказать (при некоторых технических предположениях), что одновременное изменение нескольких координат реализуется с нулевой вероятностью. Благодаря этому свойству множество векторов ошибок всех алгоритмов семейства почти всегда образует связный граф, рёбра которого соответствуют парам векторов, отличающихся только на одном объекте. Свойством связности обладают многие алгоритмы классификации с непрерывной по параметрам разделяющей поверхностью: линейные классификаторы, машины опорных векторов с непрерывными ядрами, нейронные сети с непрерывными функциями активации, решающие деревья с пороговыми условиями, и многие другие. Переписав выкладки [13, 14] в слабой аксиоматике, нетрудно доказать, что для связного семейства $\mathbb{A}$ справедлива оценка

$$\mathsf{P}\left[\delta_\mu(X) \geqslant \varepsilon\right] \leqslant \frac{1}{\sqrt{\pi L}}\,|A| \cdot \max_m H_L^{\ell,m}\left(s_1(\varepsilon)\right),$$

отличающаяся от (2.4) только множителем $\sqrt{\pi L}$, который существенно меньше степени завышенности, следовательно, не даёт радикального улучшения точности.

Возникает вопрос: в чём причина неудач? Казалось бы, усилия были направлены на то, чтобы учесть расслоение семейства и сходство алгоритмов, то есть на устранение основных факторов завышенности.

3 Семейство из двух алгоритмов

Данный раздел преследует две цели. Во-первых, показать принципиальную возможность получать *точные* оценки вероятности переобучения, опираясь только на слабую вероятностную аксиоматику и простые комбинаторные рассуждения. Во-вторых, показать, что явление переобучения возникает даже в самом простом случае, причём эффекты расслоения и сходства снижают вероятность переобучения.

Рассмотрим семейство из двух алгоритмов, $\mathbb{A} = \{a_1, a_2\}$. Возьмём в качестве μ метод *минимизации эмпирического риска*. В случае неоднозначности выбора лучшего алгоритма на обучающей выборке, когда $\nu(a_1, X) = \nu(a_2, X)$, будем ориентироваться на худший случай, полагая, что выбирается алгоритм с бóльшим числом ошибок на полной выборке.

Теорема 3.1. *Пусть в выборке $\mathbb{X}$ имеется m_0 объектов, на которых оба алгоритма допускают ошибку; m_1 объектов, на которых только a_1 допускает ошибку; m_2 объектов, на которых только a_2 допускает ошибку; m_3 остальных объектов (на которых оба алгоритма не допускают ошибку), и пусть для определённости $m_1 \leqslant m_2$:*

$$a_1 = (1, \ldots, 1,\ 1, \ldots, 1,\ 0, \ldots, 0,\ 0, \ldots, 0\,);$$
$$a_2 = (\underbrace{1, \ldots, 1}_{m_0},\ \underbrace{0, \ldots, 0}_{m_1},\ \underbrace{1, \ldots, 1}_{m_2},\ \underbrace{0, \ldots, 0}_{m_3}).$$

Тогда для любого $\varepsilon \in [0, 1)$ справедлива точная оценка:

$$Q_\varepsilon = \sum_{s_0=0}^{m_0} \sum_{s_1=0}^{m_1} \sum_{s_2=0}^{m_2} \sum_{s_3=0}^{m_3} \frac{C_{m_0}^{s_0} C_{m_1}^{s_1} C_{m_2}^{s_2} C_{m_3}^{s_3}}{C_L^\ell} \big[s_0 + s_1 + s_2 + s_3 = \ell\big] \times$$
$$\times \Big(\big[s_1 < s_2\big] \big[s_0 + s_1 \leqslant \tfrac{\ell}{L}(m_0 + m_1 - \varepsilon k)\big] +$$
$$+ \big[s_1 \geqslant s_2\big] \big[s_0 + s_2 \leqslant \tfrac{\ell}{L}(m_0 + m_2 - \varepsilon k)\big] \Big).$$

Доказательство. Метод минимизации эмпирического риска выбирает алгоритм a_1 при $\nu(a_1, X) < \nu(a_2, X)$ и алгоритм a_2 в противном случае. Следовательно,

$$Q_\varepsilon = \frac{1}{C_L^\ell} \sum_{X \in \mathbb{X}_L^\ell} \big[\nu(a_1, X) < \nu(a_2, X)\big] \big[\nu(a_1, \bar{X}) - \nu(a_1, X) \geqslant \varepsilon\big] +$$
$$+ \frac{1}{C_L^\ell} \sum_{X \in \mathbb{X}_L^\ell} \big[\nu(a_1, X) \geqslant \nu(a_2, X)\big] \big[\nu(a_2, \bar{X}) - \nu(a_2, X) \geqslant \varepsilon\big].$$

Разобьём множество $\mathbb{X}$ на 4 подмножества: X_0 — объекты, на которых ошибаются оба алгоритма; X_1 — объекты, на которых ошибается только a_1; X_2 — объекты,

на которых ошибается только a_2; X_3 — все остальные объекты. Очевидно, $m_i = |X_i|$. Положим $s_i = |X_i \cap X|$ — число объектов из X_i, попавших в обучающую выборку X.

В этих обозначениях частоты ошибок алгоритмов a_1, a_2 на выборках X, $\bar{X}$ есть

$$\nu(a_1, X) = \tfrac{s_0+s_1}{\ell}; \qquad \nu(a_1, \bar{X}) = \tfrac{m_0+m_1-s_0-s_1}{k};$$
$$\nu(a_2, X) = \tfrac{s_0+s_2}{\ell}; \qquad \nu(a_2, \bar{X}) = \tfrac{m_0+m_2-s_0-s_2}{k}.$$

Число разбиений, при которых реализуется набор значений (s_0, s_1, s_2, s_3), есть

$$\sum_{X \in \mathbb{X}_L^\ell} \prod_{i=0}^{3} \big[s_i = |X_i \cap X|\big] = C_{m_0}^{s_0} C_{m_1}^{s_1} C_{m_2}^{s_2} C_{m_3}^{s_3}. \tag{3.1}$$

Отсюда следует, что s_0, s_1, s_2, s_3 должны удовлетворять ограничениям

$$0 \leqslant s_0 \leqslant m_0; \quad 0 \leqslant s_1 \leqslant m_1; \quad 0 \leqslant s_2 \leqslant m_2; \quad 0 \leqslant s_3 \leqslant m_3.$$

Кроме того, s_0, s_1, s_2, s_3 должны удовлетворять соотношению $s_0 + s_1 + s_2 + s_3 = \ell$. Таким образом,

$$Q_\varepsilon = \frac{1}{C_L^\ell} \sum_{X \in \mathbb{X}_L^\ell} \sum_{s_0=0}^{m_0} \sum_{s_1=0}^{m_1} \sum_{s_2=0}^{m_2} \sum_{s_3=0}^{m_3} \big[s_0 + s_1 + s_2 + s_3 = \ell\big] \prod_{i=0}^{3} \big[s_i = |X_i \cap X|\big] \times$$
$$\times \Big(\big[s_1 < s_2\big] \big[\tfrac{m_0+m_1-s_0-s_1}{k} - \tfrac{s_0+s_1}{\ell} \geqslant \varepsilon\big] + \big[s_1 \geqslant s_2\big] \big[\tfrac{m_0+m_2-s_0-s_2}{k} - \tfrac{s_0+s_2}{\ell} \geqslant \varepsilon\big] \Big).$$

Переставляя знаки суммирования и подставляя сюда (3.1), получаем требуемую точную оценку. ■

В экспериментах наряду с вероятностью переобучения удобно оценивать *эффективный локальный коэффициент разнообразия* (ЭЛКР), введённый в [18]. Это такое значение коэффициента разнообразия $|A|$, при котором оценка (2.4) не является завышенной. Сопоставляя (2.1) и (2.4), получаем выражение для ЭЛКР:

$$\Delta = \frac{\mathsf{P}\big[\delta_\mu(X) \geqslant \varepsilon\big]}{\max\limits_{a \in A} \mathsf{P}\big[\delta(a, X) \geqslant \varepsilon\big]}.$$

В данной работе оценивалась верхняя оценка ЭЛКР, которая имеет более естественную содержательную интерпретацию. Она показывает, во сколько раз вероятность переобучения метода μ превышает вероятность большого уклонения частот для наилучшего алгоритма в семействе:

$$\bar{\Delta} = \frac{\mathsf{P}\big[\delta_\mu(X) \geqslant \varepsilon\big]}{\min\limits_{a \in A} \mathsf{P}\big[\delta(a, X) \geqslant \varepsilon\big]}.$$

Очевидно, в случае двухэлементного семейства алгоритмов $1 \leqslant \bar{\Delta} \leqslant 2$.

На рис. 1, 2 показана зависимость верхней оценки ЭЛКР $\bar{\Delta}$ от различности алгоритмов при $\ell = k = 100$, $\varepsilon = 0.05$. В качестве естественной меры различности

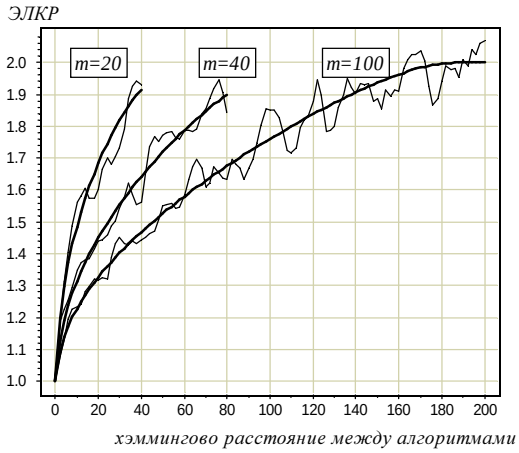

Рис. 1. Зависимость верхней оценки ЭЛКР $\bar{\Delta}$ от различности алгоритмов, когда алгоритмы допускают одинаковое число ошибок, $m_1 = m_2$. Три графика соответствуют трём разным значениям числа ошибок на полной выборке $m = n(a_i, \mathbb{X}) = m_1 + m_0 \in \{20, 40, 100\}$.

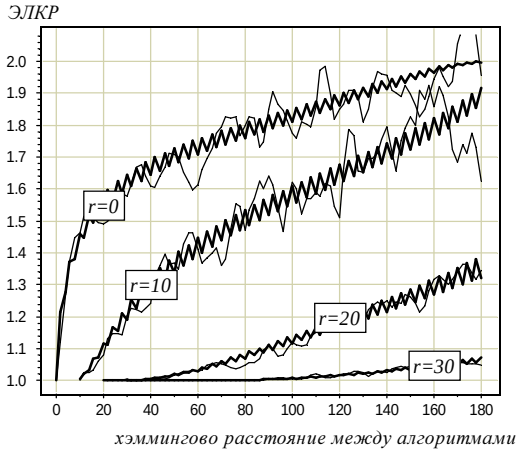

Рис. 2. Зависимость верхней оценки ЭЛКР $\bar{\Delta}$ от различности алгоритмов, когда $m_0 = 20$ и второй алгоритм допускает на r ошибок больше, $m_2 = m_1 + r$. Четыре графика соответствуют четырём разным значениям $r \in \{0, 10, 20, 30\}$.

взято хэммингово расстояние между векторами ошибок $\rho(a_1, a_2) = m_1 + m_2$. Тонкими линиями показаны оценки ЭЛКР методом Монте-Карло по 1000 случайных разбиений.

Графики позволяют сделать следующие выводы.

1. Переобучение связано с выбором алгоритма по неполной выборке $X \subset \mathbb{X}$. Оно возникает даже если выбор производится всего лишь из двух алгоритмов.

2. Если алгоритмы допускают на $\mathbb{X}$ одинаковое число ошибок ($m_1 = m_2$), но при этом максимально различны ($m_0 = 0$), то вапниковская оценка $\bar{\Delta} = 2$ достигается или почти достигается.

3. Если алгоритмы схожи, то ЭЛКР приближается к 1, то есть два схожих алгоритма с точки зрения переобучения ведут себя практически как один алгоритм.

4. Если алгоритмы различны по числу ошибок, $r = m_2 - m_1 > 0$, то вапниковская оценка также не достигается. Чем больше r, тем меньше вероятность переобучения.

Основной вывод: явление переобучения возникает даже в том простейшем слу-

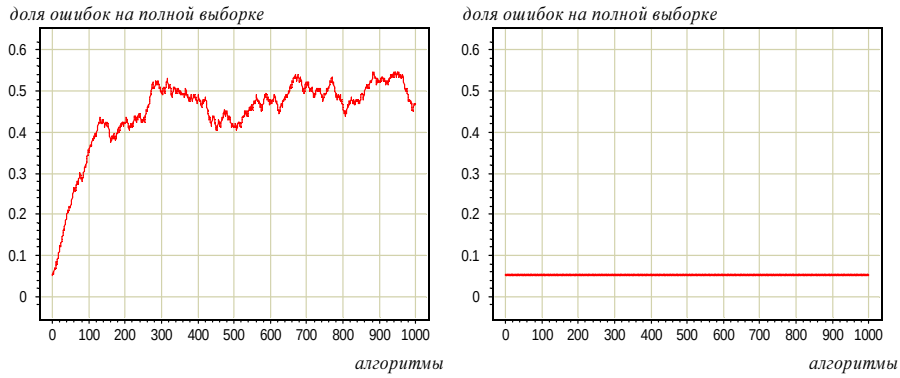

Рис. 3. Цепочки с расслоением и без. Зависимость $\nu(a_t, \mathbb{X})$ от t при $\ell = k = 100,\ m = 10$.

чае, когда алгоритмов только два. При этом наличие свойств расслоения и сходства снижает вероятность переобучения.

4 Эксперименты с цепочками алгоритмов

Цель данного раздела — продемонстрировать на конкретном примере, что численно точные оценки вероятности переобучения возможно получить только путём совместного учёта и расслоения семейства, и сходства алгоритмов.

Последовательность алгоритмов $\{a_1, \ldots, a_D\}$ будем называть *цепочкой*, если хэммингово расстояние между векторами ошибок a_{t-1} и a_t равно 1 для всех $t = 2, \ldots, D$. Цепочка является простейшим примером связного семейства алгоритмов [14].

Зависимость вероятности переобучения от длины цепочки D исследовалась экспериментально. Для этого строились модельные цепочки двух типов, задаваемые непосредственно последовательностью векторов ошибок $a_1, \ldots, a_D$.

1. *Цепочка с расслоением*, рис. 3 слева. Лучший алгоритм a_1 допускает m ошибок на полной выборке. Каждый следующий вектор ошибок a_t получается из a_{t-1} путём инверсии одной случайно выбранной координаты. Если цепочка достаточно длинная ($D \gg L$), то большинство алгоритмов допускают число ошибок m, близкое к $L/2$.

2. *Цепочка без расслоения*, рис. 3 справа. Число ошибок алгоритмов на полной выборке, чередуясь, принимает значения m и $m + 1$.

Для каждой цепочки строилась соответствующая ей *не-цепочка* $\{a'_1, \ldots, a'_D\}$, состоящая из существенно различных алгоритмов. Векторы ошибок a'_t генерировались случайным образом, но так, чтобы $\nu(a'_t, \mathbb{X}) = \nu(a_t, \mathbb{X})$ для всех $t = 1, \ldots, D$. Таким образом, в не-цепочках соседние алгоритмы a_{t-1}, a_t не являлись схожими.

Итого, строилось четыре конечных семейства алгоритмов при одинаковых значениях параметров D и m. Сопоставление этих четырёх случаев позволило разделить влияние *сходства* (цепочки или не-цепочки) и *расслоения* (m ошибок у всех алгоритмов или только у лучшего) на вероятность переобучения.

На рис. 4 и рис. 5 показаны зависимости вероятности переобучения Q_ε и ЭЛКР $\bar{\Delta}$ от числа алгоритмов D для четырёх типов семейств, при $\ell = k = 100,\ \varepsilon = 0.05$. Значения Q_ε вычислялись методом Монте-Карло по 1000 случайных разбиений. Условные обозначения на графиках: $+Ц$ — наличие цепочки, $-Ц$ — отсутствие цепочки, $+Р$ —

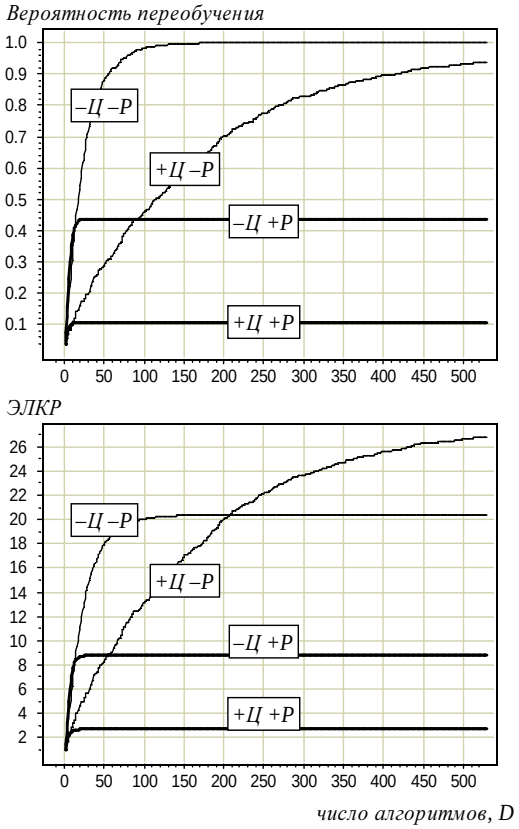

Рис. 4. Зависимость вероятности переобучения Q_ε и ЭЛКР $\bar{\Delta}$ от числа алгоритмов D («простая задача» — частота ошибок лучшего алгоритма $\nu(a_1, \mathbb{X}) = 0.05$).

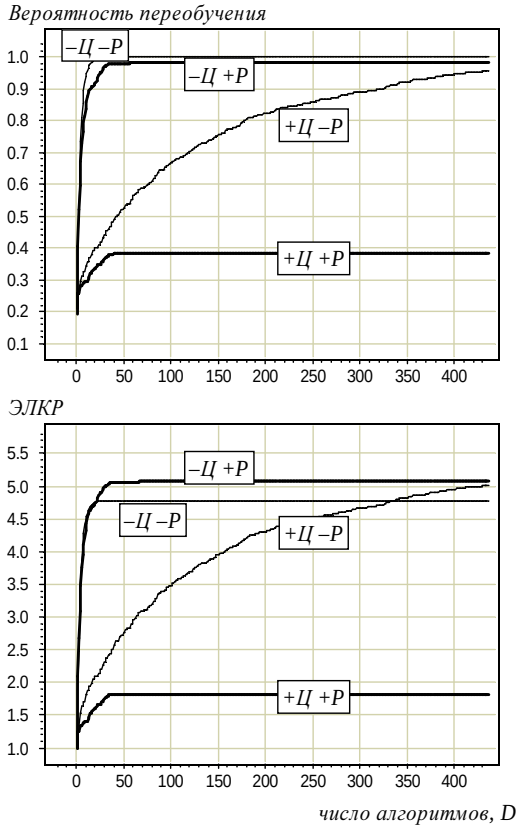

Рис. 5. Зависимость вероятности переобучения Q_ε и ЭЛКР $\bar{\Delta}$ от числа алгоритмов D («трудная задача» — частота ошибок лучшего алгоритма $\nu(a_1, \mathbb{X}) = 0.25$).

наличие расслоения, $-P$ — отсутствие расслоения.

Эти графики позволяют сделать следующие выводы.

1. Зависимость ЭЛКР $\bar{\Delta}$ от числа алгоритмов D показывает, какое значение вместо D должен был бы принимать коэффициент разнообразия, чтобы оценка (2.4) не была завышенной. Это значение может оказаться существенно меньшим, чем D. В какой-то момент вероятность переобучения достигает некоторого максимального значения $Q_{\max}$, при этом ЭЛКР также выходит на горизонтальную асимптоту и перестаёт зависеть от D. В то же время, оценка Вапника-Червоненкиса линейна по D и вообще не имеет горизонтальной асимптоты. Она достигается только для не-цепочек и только при малых D (в данном эксперименте при $D < 8$).

2. При наличии цепочки вероятность переобучения Q_ε растёт существенно медленнее с ростом числа алгоритмов D. Таким образом, благодаря свойству связности, число алгоритмов в семействе может быть намного больше, чем предсказывает VC-теория.

3. При наличии расслоения (толстые кривые на графиках) вероятность переобучения Q_ε может не достигать 1 даже при очень больших D. В то же время, в цепочках без расслоения $Q_{\max}$ всё-таки достигает 1 при D порядка сотен. Таким образом, именно свойство расслоения приводит к понижению горизонтальной асимптоты $Q_{\max}$

до уровня, существенно меньшего единицы. Заметим, что этот эффект невозможно объяснить исходя из принципа равномерной сходимости, так как согласно Теореме 2.2 $Q_\varepsilon = \widetilde{Q}_\varepsilon$ только при отсутствии расслоения.

4. Для относительно простых задач, когда существует алгоритм с низким уровнем ошибок, наличие расслоения сильнее уменьшает вероятность переобучения, чем наличие цепочки, рис. 4. При увеличении сложности задачи влияние расслоения уменьшается, рис. 5.

5. При больших D только одновременное наличие цепочки и расслоения позволяет избежать сильного переобучения (нижние кривые на каждом из графиков). Внушает оптимизм тот факт, что именно этот случай наиболее распространён на практике.

Список литературы

[1] *Воронцов К. В.* Обзор современных исследований по проблеме качества обучения алгоритмов // *Таврический вестник информатики и математики.* — 2004. — № 1. — С. 5–24.

[2] *Bax E. T.* Similar classifiers and VC error bounds: Tech. Rep. CalTech-CS-TR97-14: 6 1997.

[3] *Boucheron S., Bousquet O., Lugosi G.* Theory of classification: A survey of some recent advances // *ESAIM: Probability and Statistics.* — 2005. — no. 9. — Pp. 323–375.

[4] *Efron B.* The Jackknife, the Bootstrap, and Other Resampling Plans. — SIAM, Philadelphia, 1982.

[5] *Graham R. L., Knuth D. E., Patashnik O.* Concrete Mathematics. — Reading, Massachusetts: Addison-Wesley, 1994. — P. 657.

[6] *Hastie T., Tibshirani R., Friedman J.* The Elements of Statistical Learning. — Springer, 2001.

[7] *Herbrich R., Williamson R.* Algorithmic luckiness // *Journal of Machine Learning Research.* — 2002. — no. 3. — Pp. 175–212.

[8] *Kearns M. J., Mansour Y., Ng A. Y., Ron D.* An experimental and theoretical comparison of model selection methods // 8th Conf. on Computational Learning Theory, Santa Cruz, California, US. — 1995. — Pp. 21–30.

[9] *Kohavi R.* A study of cross-validation and bootstrap for accuracy estimation and model selection // 14th International Joint Conference on Artificial Intelligence, Palais de Congres Montreal, Quebec, Canada. — 1995. — Pp. 1137–1145.

[10] *Langford J.* Quantitatively Tight Sample Complexity Bounds: Ph.D. thesis / Carnegie Mellon Thesis. — 2002.

[11] *Langford J., McAllester D.* Computable shell decomposition bounds // Proc. 13th Annu. Conference on Comput. Learning Theory. — Morgan Kaufmann, San Francisco, 2000. — Pp. 25–34.

[12] *Lugosi G.* On concentration-of-measure inequalities. — Machine Learning Summer School, Australian National University, Canberra. — 2003.

[13] *Sill J.* Generalization bounds for connected function classes. — citeseer.ist.psu.edu/127284.html.

[14] *Sill J.* Monotonicity and connectedness in learning systems: Ph.D. thesis / California Institute of Technology. — 1998.

[15] *Vapnik V.* Statistical Learning Theory. — Wiley, New York, 1998.

[16] *Vapnik V., Chervonenkis A.* On the uniform convergence of relative frequencies of events to their probabilities // *Theory of Probability and its Applications.* — 1971. — Vol. 16, no. 2. — Pp. 264–280.

[17] *Vorontsov K. V.* Combinatorial substantiation of learning algorithms // *Comp. Maths Math. Phys.* — 2004. — Vol. 44, no. 11. — Pp. 1997–2009.

[18] *Vorontsov K. V.* Combinatorial probability and the tightness of generalization bounds // *Pattern Recognition and Image Analysis.* — 2008. — Vol. 18, no. 2. — Pp. 243–259.