УДК 519.246

В. М. Недельк о

К ВОПРОСУ ОБ ЭФФЕКТИВНОСТИ БУСТИНГА В ЗАДАЧЕ КЛАССИФИКАЦИИ*

Исследуются причины высокой эффективности методов, основанных на композициях решающих функций, в частности бустинга. Показано, что одной из главных причин такой эффективности может быть использование эффекта независимости переменных. Для выявления особенностей метода исследуется его работа непосредственно на распределениях. Проводится сравнение аппроксимирующей способности бустинга и сплайнов. Также показана связь сложности композиции с достижимой величиной отступа.

Введение

В настоящее время использование методов построения решающих функций, основанных на композициях [7], является едва ли не необходимым условием для успешного решения практических задач. Об этом свидетельствуют как многочисленные публикации, так и сообщения об ипользуемых при решении задач методах (например, на портале www.kaggle.com).

Тем не менее, до сих пор остаётся открытым вопрос: за счёт чего композиции, в частности бустинг, зачастую превосходят другие методы.

Сложность выяснения ответа на этот вопрос связана в том числе с тем, что преимущество бустинга обычно выявляется при решении реальных задач, для которых известны только данные, но, как правило, нет информации о свойствах и особенностях этих данных. При этом, даже если подвергнуть данные детальному анализу для выявления их особенностей, реальные данные обладают очень большим разнообразием свойств, которые могут потенциально влиять на работу методов построения решающих функций, так что выявить, какие именно свойства обеспечивают успешность решения задачи [8] [9] [3], весьма затруднительно.

Исходя из сказанного, напрашивается идея о целесообразности исследования методов на синтетических данных, все свойства которых известны по построению [18] [19]. Эта идея близка одному из принципов, которые были заложены ещё в 1980-е годы в системе «Полигон» [13]. Принцип состоял в том, что для каждого метода в тестовый набор включается «эталонная» задача, на которой метод должен работать лучше других.

Мы будем в качестве задачи использовать распределение, исследуя, на каких распределениях бустинг работает хорошо и почему.

Кроме того, мы будем запускать бустинг на самих распределениях. Хотя качество решения на распределении не обязательно отражает качество выборочного решения, это

*Работа выполнена при финансовой поддержке РФФИ, проекты № 14-01-00590 и № 14-07-00249

ISSN 1818-7943. Вестник НГУ. Серия: Математика, механика, информатика. 2099. Т. 99, вып. 5. С. 1–19.

даёт представление об аппроксимационных возможностях метода. Кроме этого, эффективность метода зависит от его обобщающей способности.

При исследованиях эффективности методов анализа данных возникает также и методологическая проблема, связанная с отсутствием приемлемого формального определения эффективности [10] [11] [12] [5] метода. Обычно под эффективностью понимается в некотором смысле интегральное качество (в смысле вероятности ошибочной классификации или другого варианта риска) получаемых решений на некотором классе задач. Однако на разных задачах (на разных распределениях) даже одного класса заданный метод может демонстрировать существенно различное качество работы. И введение обобщённой по классу характеристики качества является нетривиальной задачей.

Отсутствие интегрального критерия качества не позволяет ввести понятие оптимального метода.

Действительно, даже для нормальных распределений оптимальный метод классификации неизвестен. Так при известных распределениях оптимальным будет квадратичный дискриминант. Однако прямолинейная оценка параметров распределений с последующей подстановкой в разделяющую функцию, очевидно, не может считаться оптимальным методом, в виду, в частности, такого факта, что при малой выборке лучше строить линейное правило, даже если известно, что ковариационные матрицы не равны. Напрашивается предположение, что оптимальный метод построения выборочной решающей функции должен предоставлять возможность плавного увеличения сложности решения (переход от линейных к квадратичным функциям) при увеличении объёма выборки.

Заметим, что задача построения решающих функций близка задаче проверки статистических гипотез. Действительно, метод построения решающих функций — это отображение множества выборок во множество решающих функций. Статистический критерий — это отображение множества выборок во множество, состоящее из двух решений: принять или отвергнуть заданную гипотезу. Таким образом статистический критерий можно считать частным случаем метода построения решающих функций. При этом понятия оптимального статистического критерия (в случае отсутствия альтернатив) не существует [15] [16].

В работе [17] предложен минимаксный подход к определению интегральной по классу задач характеристики качества, который теоретически позволяет ввести понятие оптимального на классе распределений метода построения решающих функций.

Заметим, что для таблицы данных понятие оптимального метода построения решающих функций вообще не имеет смысла, а для фиксированного распределения оптимальный метод вырожден — он любой выборке сопоставляет байесовское решающее правило. Об оптимальности метода можно говорить только для класса распределений.

Исходя из сказанного, в данной работе исследование качества бустинга будет проводиться на модельных классах распределений.

1. Задача построения решающей функции

Основные понятия. Пусть X — пространство значений переменных, используемых для прогноза, а $Y = \{-1, 1\}$ — пространство значений прогнозируемых переменных, и пусть C — множество всех вероятностных мер на заданной σ-алгебре подмножеств множества $D = X \times Y$. При каждом $c \in C$ имеем вероятностное пространство $\langle D, B, \mathsf{P}_c \rangle$, где B — σ-алгебра, P_c — вероятностная мера.

Решающей функцией называется соответствие $\lambda\colon X \to Y$.

Качество принятого решения оценивается заданной функцией потерь $\mathcal{L}\colon Y^2 \to [0, \infty)$. Под риском будем понимать средние потери:

$$R(c, \lambda) = \mathsf{E}\mathcal{L}(y, \lambda(x)) = \int_D \mathcal{L}(y, \lambda(x))\ \mathsf{P}_c(dx, dy), \quad x \in X, y \in Y. \tag{1}$$

При $\mathcal{L}(y, \tilde{y}) = I(y = \tilde{y})$, где $I(\cdot)$ – индикаторная функция (равна 1, если условие истинно, и 0 — иначе), риск есть вероятность ошибочной классификации.

Задача классификации заключается в построении решающей функции, которая бы минимизировала риск.

Заметим, что значение риска зависит от c — распределения, которое неизвестно. Поэтому в приведённой формулировке задача некорректна. Однако полностью строгой общей постановки задачи распознавания в настоящее время не существует [17]. На практике либо решаются более частные строго поставленные задачи, либо разрабатываются эвристические методы, например, методы, минимизирующие выборочную оценку риска.

Пусть $V = \big((x^i, y^i) \in D \,|\, i = 1, \ldots, N\big)$, $V \in D^N$ — случайная независимая выборка из распределения P_c.

Метод построения решающих функций есть отображение $Q\colon D^N \to \Lambda$, где Λ — заданный класс решающих функций, а $\lambda_{Q,V}$ — функция, построенная по выборке V методом Q.

Для оценивания риска обычно используются эмпирические функционалы качества, т. е. точечные оценки риска, такие как эмпирический риск, оценка скользящего экзамена, оценка bootstrap и т.п.

Эмпирический риск определяется как средние потери на выборке:

$$\widetilde{R}(V, \lambda) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y^i, \lambda(x^i)).$$

Выражение для эмпирического риска является частным случаем формулы (1), если в неё подставить эмпирическое распределение. Этот факт обозначается термином Fisher consistency.

Приведённые понятия являются базовым набором понятий в задаче распознавания. Однако для дальнейших рассуждений нам потребуется ввести некоторые их обобщения.

Обобщение базовых понятий. Основное обобщение касается метода классификации. Большинство известных методов допускают использование не обычной выборки, а выборки с весами. Многие методы позволяют вместо выборки в качестве входных данных

использовать распределение. Заметим, что распределение является самым общим случаем входных данных, поскольку обычной выборке соответствует эмпирическое распределение, а взвешенной выборке — дискретное распределение.

В данной работе для простоты ограничимся дискретными распределениями. В контексте рассматриваемых вопросов это не будет существенным ограничением общности.

Под методом построения решающих функций будем понимать отображение $Q\colon D^N \times W \to \Lambda$, где Λ — заданный класс решающих функций, а $\lambda_{Q,V}$ — функция, построенная по выборке V методом Q, а W – пространство весов, образованное векторами вида $w = (w_1, ..., w_N)$, где $w_i \geq 0$ и $\sum_{i=1}^{N} w_i = 1$.

Понятие решающей функции обобщается посредством введения пространства оценок [7], когда в качестве множества решений выступает не Y, а расширенное множество $\widetilde{Y}$, например, множество всех действительных чисел.

Решающей функцией в этом случае будет соответствие $\lambda\colon X \to \widetilde{Y}$.

Для таких решающих функций вместо функции потерь вводят понятие отступа: $\mathcal{M}(y, \tilde{y}) = y\tilde{y}$.

Вместо минимизации эмпирического риска осуществляется максимизация среднего отступа:

$$\widetilde{M}(V, w, \lambda) = \sum_{i=1}^{N} w_i \mathcal{M}(y^i, \lambda(x^i)).$$

Универсальные методы классификации. Универсальным называется такой класс решающих функций [14], в котором для любого распределения и для любой заданной (ненулевой) погрешности найдётся функция, позволяющая с заданной точностью приблизиться к байесовскому уровню ошибки. Примерами универсальных классов являются функции, кусочно-постоянные на интервалах, а также деревья решений.

Для методов построения решающих функций существует понятие, обозначаемое термином universal consistency, который может быть переведён как «универсальная состоятельность». Метод называется состоятельным для заданного класса распределений, если при увеличении объёма выборки величина риска для получаемых решений сходится по вероятности к байесовскому уровню ошибок. Если данный факт имеет место для любых распределений, то такой метод называется universally consistent, т.е. универсально состоятельным. Состоятельность метода означает, что он обладает обобщающей способностью, однако не гарантирует, что она высокая.

Введём также понятие универсального метода, как метода, использующего в качестве решений универсальный класс решающих функций.

Примером универсального метода классификации являются нейронные сети.

Универсальность — положительное свойство метода, однако, само по себе оно недостаточно для того, чтобы метод был эффективным. Так, например, можно предложить тривиальный алгоритм построения дерева решений, которое сколь угодно точно аппроксимирует любую зависимость, но такой метод будет бесполезен на практике ввиду низкой обобщающей способности. С другой стороны, например, метод AdaBoost на пороговых классификаторах на практике зарекомендвал себя достаточно эффективным, но он не универсален, что демонстрируется на рис. 1, где приведён пример разделяющей

функции (в виде эллипса) между классами, которая не может быть точно воспроизведена этим методом.

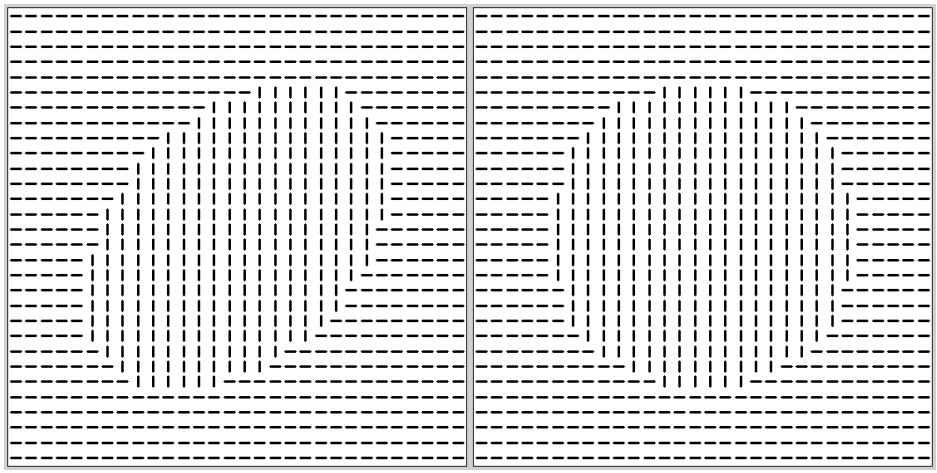

Рис. 1: Байесовское решающее правило и решение, построенное бустингом на пороговых классификаторах.

2. Алгоритмические композиции

Алгоритмические композиции [7] позволяют обогатить класс решающих путём конструирования новых функций в виде суперпозиции исходных функций и некоторой агрегирующей функции.

Чаше всего в качестве агрегирующей используется линейная функция

$$\beta(x) = \sum_{t=1}^{T} \alpha_t \lambda_t(x), \tag{2}$$

где λ_t – базовые решающие функции, $\alpha_t \geq 0$ – параметры (веса). Окончательное решение о принадлежности объекта классу принимается пороговой функцией $\bar{\lambda}(x) = \text{sign}(\beta(x))$.

Функция $\beta(x)$ позволяет также упорядочивать объекты по степени (или вероятности) принадлежности их к заданному классу. Кроме того, путём специального преобразования (часто это логистическая функция) из неё может быть получена оценка вероятности принадлежности объекта классу.

Возникает вопрос, почему использование композиции предпочтительнее, чем использование изначально богатого класса решающих функций, например, деревьев решений, которые сами по себе являются универсальным классом [14].

Существует мнение (основанное на многочисленных исследованиях), что композиции меньше подвержены переобучению.

Однако, следует отметить, что это утверждение пока не только никем не доказано, но даже и не сформулировано в математически строгом смысле. Более того, утверждение может быть справедливым только при указании дополнительных уточняющих условий, поскольку можно привести тривиальные контрпримеры. Так, композиция пороговых

классификаторов в одномерном случае эквивалентна методу ближайшего соседа (который имет бесконечную ёмкость), если не ограничивать размер композиции.

С другой стороны, также не доказано, что низкая переобучаемость — это безусловное достоинство метода. Во-первых, с практической точки зрения обычно важен только полученный уровень риска, т.е. само значение вероятности ошибочной классификации, а насколько эта вероятность отличается от эмпирического риска — роли не играет. Во-вторых, по убеждению автора, совсем не переобучаются только те методы, которые не обучаются, и существует некоторое оптимальное значение этой характеристики.

Таким образом, вопрос, за счёт чего композиции зачастую наиболее эффективны при решении реальных задач, остаётся открытым. Естественно предположить, что они «угадывают» какие-то особенности, характерные для распределений, которым подчиняются реальные данные.

Поиск таких особенностей и есть цель данной работы.

Помимо того, чтобы угадать модель, требуется ещё эффективно распорядиться выборкой, чтобы подобрать параметры этой модели.

Чтобы отделить влияние аппроксимирующй способности метода на его эффективность от влияния статистической устойчивости, мы будем испытывать методы на самих распределениях.

Заметим, что важен не только используемый класс решающих функций, но и сам метод построения функции по выборке. Для повышения статистической устойчивости, при построении композиции часто используют регуляризацию. Один из вариантов регуляризации — построение композиции минимального размера. Примером может служить минимальный комитет [20].

Другой путь регуляризации — направленная оптимизация, что используется, например, в бустинге.

3. Бустинг

Отличительной особенностью бустинга является то, что этот метод не использует класс базовых решающих функций в явном виде. Для бустинга достаточно иметь метод классификации в виде «чёрного ящика», который сначала обучается на основе поданной на вход выборки, а затем может принимать решение в любой точке.

Отсутствие явного представления класса решающих функций накладывает серьёзные ограничения на способ формирования композиции: всё, что мы можем сделать для получения нового базового классификатора — это сформировать новую выборку и подать её на вход базового метода.

Алгоритм AdaBoost. Данный алгоритм является исторически первым [21] и самым простым вариантом бустинга, однако, как будет показано в дальнейшем, именно он обладает важной особенностью, которая, вероятно, соответствует некоторым типичным свойствам реальных задач.

В методе AdaBoost решение строится в виде композиции (2), где базовые классификаторы и их веса находятся следующим образом.

Первый базовый классификатор строится базовым методом на основе исходной выборки, объектам которой приписаны начальные веса $w^1 = (w_1^1, ..., w_N^1)$.

Заметим, что мы будем задавать начальные веса объектам в соответствии с выбранным распределением, но в стандартном варианте метода начальные веса выбираются одинаковыми, т.е. $w_i^1 = \frac{1}{N}$.

Вес построенного базового классификатора в композиции определяется по формуле

$$\alpha_t = \frac{1}{2} \ln \frac{\widetilde{M}^+(V, w^t, \lambda_t)}{\widetilde{M}^-(V, w^t, \lambda_t)},$$

где

$$\widetilde{M}^+(V, w, \lambda) = \sum_{i=1}^{N} w_i \cdot I(y^i = \lambda(x^i)), \quad \widetilde{M}^-(V, w, \lambda) = \sum_{i=1}^{N} w_i \cdot I(y^i = -\lambda(x^i)).$$

Следующие базовые классификаторы строятся тем же базовым методом по выборке, веса объектов в которой вычисляются по формулам

$$w_i^{t+1} = \frac{\bar{w}_i^{t+1}}{\sum_{i=1}^{N} \bar{w}_i^{t+1}}, \quad \bar{w}_i^{t+1} = w_i^t \cdot e^{-\alpha_t y^i \lambda_t(x^i)}.$$

Смысл этих формул в том, что веса правильно классифицированных объектов умножаются на $e^{-\alpha_t}$, а веса неправильно классифицированных объектов умножаются на e^{α_t}. После этого веса нормируются на 1. В результате веса неправильно классифицируемых объектов увеличиваются.

Оптимизационная задача. Бустинг можно рассматривать как способ оптимизации композиции.

Чтобы математически определить метод построения решающих функций, достаточно указать класс функций и (эмпирический) критерий качества. Однако точное решение получившейся оптимизационной задачи может быть практически недостижимым, или неоправданным с точки зрения сложности, в виду чего используют приближённые оптимизационные алгоритмы. При этом обычно полагается, что использование приближённого решения — это компромисс, и в идеале желательно иметь точное решение.

Однако фактическое качество решения определяется величиной риска, а вовсе не достигнутым значением эмпирического критерия. Отсюда имеем возможность такой парадоксальной ситуации, когда приближённый алгоритм решения оптимизационной задачи может оказаться лучше (по вероятности ошибки) точного алгоритма. При этом направленность поиска может играть роль регуляризации. Можно добиваться упрощения и большей устойчивости решений, вводя регуляризующие поправки в эмпирический критерий, а можно упростить решение, упрощая алгоритм его поиска.

Например, можно искать дерево заданного размера с минимальным числом ошибок на обучающей выборке, а можно направленым поиском найти неоптимальное дерево большего размера с тем же числом ошибок. Что лучше с точки зрения ожидаемой вероятности ошибочной классификации новых объектов — неизвестно.

С бустингом ситуация схожая. Существуют различные модификации бустинга (многие из которых можно уложить в схему градиентного бустинга), которые используют

различные эмпирические критерии качества и имеют различную эффективность по оптимизации этих критериев. Однако подобные показатели эффективности, включая длину построенной композиции, напрямую не отражают реальное качество метода [23].

Оценивание качества различных модификаций бустинга — область дальнейших исследований. В данной работе ограничимся методом AdaBoost.

Бустинг на распределениях. Бустинг может быть легко запущен непосредственно на распределениях вместо выборки. Для этого необходимо и достаточно, чтобы базовый метод классификации мог работать с распределениями.

Мы ограничимся дискретными распределениями. Технически это сводится к тому, что начальные веса объектов выбираются не равными, а пропорциональными вероятностям.

4. Случай независимых переменных

Методы построения решающих функций, использующие гипотезу независимости переменных, в зарубежной терминологии принято называть наивными байесовскими классификаторами.

В большинстве реальных задач предполагать независимость переменных нет оснований, однако с теоретической точки зрения этот случай очень важен. Если в общем случае чем больше переменных, тем больший объём выборки требуется (при прочих равных) для построения решающей функции, то при независимых переменных ситуация противоположная: чем больше переменных, тем меньший объём выборки требуется для достижения той же точности распознавания.

Гипотеза независимости переменных является очень сильной, и естественно желание иметь промежуточные варианты.

Для пространства бинарных переменных такая модель известна: это ряд Бахадура, который позволяет последовательно усложнять модель (ослаблять гипотезу) — от полной независимости, через учёт только парных зависимостей, зависимостей в тройках и т.д. до общего случая любых зависимостей.

К сожалению, обобщения ряда Бахадура для случая переменных произвольных типов автору неизвестны.

Более полную информацию, по сравнению с решающей функцией, несёт функция условной вероятности $g(x) = \mathsf{P}(y = 1 \mid x)$, которая определяет вероятность заданного класса в каждой точке пространства переменных.

Выведем формулу для вычисления условной вероятности в случае независимых переменных.

Из формулы Байеса можем записать

$$g(x) = \mathsf{P}(y = 1 \mid x) = \frac{\mathsf{P}(dx, y = 1)}{\mathsf{P}(dx, y = 1) + \mathsf{P}(dx, y = -1)} = \frac{1}{1 + \frac{1-p}{p} \cdot \frac{\mathsf{P}(dx|y=-1)}{\mathsf{P}(dx|y=1)}},$$

где $p = \mathsf{P}(y = 1)$.

Пусть условные распределения всех переменных X_j при условии обоих классов независимы, т.е. $\mathsf{P}(dx \mid y) = \prod_{j=1}^{n} \mathsf{P}(dx_j \mid y)$.

Подставив это произведение в предыдущее выражение, после преобразований имеем

$$\frac{p}{1-p} \cdot \left(\frac{1}{g(x)} - 1 \right) = \prod_{j=1}^{n} \frac{p}{1-p} \cdot \left(\frac{1}{g_j(x_j)} - 1 \right),$$

где $g_j(x_j) = \mathsf{P}(y = 1 \mid x_j) = \frac{\mathsf{P}(dx_j, y=1)}{\mathsf{P}(dx_j)}$.

Логарифмируем последнее выражение и получаем

$$\sigma^{-1}(g(x)) = (n-1)(\ln p - \ln(1-p)) + \sum_{j=1}^{n} \sigma^{-1}(g_j(x_j)),$$

где $\sigma^{-1}(\cdot)$ — функция, обратная сигмоиду $\sigma(z) = \frac{1}{1+e^{-z}}$.

Заметим, что полученное выражение имеет вид логистической регрессии, а именно

$$g(x) = \sigma \left(u_0 + \sum_{j=1}^{n} u_j \sigma^{-1}(g_j(x_j)) \right),$$

при $u_0 = (n-1)(\ln p - \ln(1-p))$, $u_j = 1$.

Обычно логистическую кривую получают, исходя из предположений о виде распределения, однако сейчас мы предположили независимость переменных, но не ограничивали вид распределений.

Данное выражение справедливо не только при независимых переменных, а в несколько более общем случае, поскольку здесь мы имеем лишь одно соотношение, а независимость независимость переменных требует выполнения мультипликативности условного распределения для каждого класса, что даёт число соотношений по числу классов.

Ещё более расширить область применимости можно, если считать веса свободными параметрами.

Дальнейшее обобщение возможно, если допустить произвольные оценочные функции

$$g(x) = \sigma \left(u_0 + \sum_{j=1}^{n} u_j s_j(x_j) \right). \tag{3}$$

Мы получили метод, который можно считать разновидностью метода логистической регрессии, а также разновидностью наивного байесовского классификатора (хоть он и не вляется частным случаем последнего).

В рамках данной статьи будем называть метод, дающий решения, представимые в форме (3), обобщённым наивным байсовским классификатором.

Функции $s_j(x_j)$ можно оптимизировать напрямую (вместе с весами u_j), например, по критерию максимального правдоподбия.

В примере будут использованы кубические сплайны.

Формулу (3) можно естественным образом обобщить по аналогии с рядом Бахадура, включив возможность учитывать зависимости между переменными, последовательно

добавляя парные зависимости, зависимости в тройках и т.д.

$$g(x) = \sigma\left(u_0 + \sum_{j=1}^{n} u_j s_j(x_j) + \sum_{j,k} u_{jk} s_{jk}(x_j, x_k) + \sum_{j,k,l} u_{jkl} s_{jkl}(x_j, x_k, x_l) + \ldots\right). \qquad (4)$$

Как будет показано в следующем разделе, бустинг как раз и позволяет строить подобные модели. Например, метод AdaBoost на деревьях с числом конечных вершин не более M соответствует (4) с ограничением суммирования вплоть до слагаемых, сответствующих всевозможным сочетаниям из n по M переменных.

5. Свойства бустинга

Бустинг позволяет оценивать условную вероятность. Этот факт известен, но мы здесь приведём наиболее краткий вывод формулы, выражающей эту оценку.

Для простоты будем считать, что пространство X дискретно. На практике это не ограничивает общности, поскольку любое распределение с практически приемлемой точностью можно приблизить дискретным распределением. Дискретные распределения нам удобны тем, что их можно представлять в виде выборки с весами. Конечно, под «выборкой» в данном контексте мы подразумеваем просто конечное множество объектов, не предполагая их случайного выбора.

Условную вероятность $g(x) = \mathsf{P}(y = 1\,|x)$ можно задать, поместив в точку x два объекта: класса 1 с весом $w_0 g(x)$ и класса -1 с весом $w_0(1 - g(x))$.

В результате выполнения бустинга вес первого объекта станет равным

$$w^{+1}(x) = w_0 g(x) \cdot A e^{-\beta(x)},$$

где константа A есть произведение всех нормировочных множителей.

В справедливости этой формулы легко убедиться непосредственно, вспомнив, что на каждом шаге вес объекта умножается на $e^{-\alpha_t}$ в случае правильной классификации и на e^{α_t} при неправильной. Но поскольку объект принадлежит классу 1, правильная классификация получается при $\lambda_t(x) = 1$, неправильная — при $\lambda_t(x) = -1$. Видим, что в обоих вариантах вес объекта умножается на $e^{-\alpha_t \lambda_t(x)}$. Произведение сомножителей по всем шагам и даёт $e^{-\beta(x)}$.

Аналогичным образом получаем, что конечный вес второго объекта есть

$$w^{-1}(x) = w_0(1 - g(x)) \cdot A e^{\beta(x)}.$$

Если положить $w^{+1}(x) = w^{-1}(x)$, то получим

$$g(x) = \frac{1}{1 + e^{-2\beta(x)}}. \qquad (5)$$

Возникает вопрос, на каком основании мы приравняли конечные веса объектов.

Вообще говоря, в результате бустинга веса объектов не обязаны становиться равными, а величина $\frac{1}{1+e^{-2\beta(x)}}$ не обязательно сходится к функции условной вероятности $g(x)$. Контрпример приведён на рис. 1.

Однако, если во всех точках пространства веса объектов противоположных классов сравнялись, то метод AdaBoost остановится, поскольку не будет возможности уменьшить критерий ошибок и веса последующих классификаторов станут нулевыми.

Мы сейчас не ставим цели выяснить условия, при которых это происходит. Цель приведённых рассуждений — вывести формулу (5) и проиллюстрировать её содержательный смысл, но не устанавливать границы её применимости.

Заметим, что мы обнаружили любопытное свойство бустинга — способность самостоятельно «останавливаться». Проявляется оно, однако, лишь при запуске метода на самих распределениях, причём таких, для которых $0 < g(x) < 1$, либо на выборках, где в каждой «непустой» точке пространства есть объекты обоих классов. Это свойство подсказывает идею регуляризации для алгоритма бустинга, состоящую в том, чтобы для каждого объекта из исходной выборки добавить в выборку объект противоположного класса с такими же значениями x, приписав ему некоторый достаточно малый вес. Это обеспечит самостоятельную остановку процесса наращивания композиции.

Бустинг на пороговых классификаторах.

Пороговым классификатором называется решающая функция вида $\lambda(x) = \pm\,\mathrm{sign}(x - c)$, где c - некоторое пороговое значение, а $\mathrm{sign}(\cdot)$ – стандартная функция знака (возвращает $+1$, -1 или 0 при соответственно положительном, отрицательном и нулевом значениях аргумента).

Утверждение 1. *Бустинг на пороговых классификаторах («пнях») является разновидностью обобщённого наивного байесовского классификатора.*

Действительно, каждая $\lambda_t(x)$ в композиции

$$\beta(x) = \sum_{t=1}^{T} \alpha_t \lambda_t(x))$$

зависит только от одной переменной X_{i_t}, поэтому после группировки слагаемых выражение можно привести к виду

$$2\beta(x) = \sum_{j=1}^{n} u_j s_j(x_j)).$$

Подставив в выражение для $g(x)$, получим искомый вид (3).

Лемма 1. *Метод AdaBoost на пороговых классификаторах в одномерном случае позволяет с любой заданной точностью приблизить произвольную функцию условной вероятности.*

Данный факт является известным, поэтому мы не будем приводить доказательство (и даже не будем конкретизировать, в какой метрике рассматривается приближение, поскольку это потребует неоправданого загромождения изложения). Лемма понадобится в дальнейшем при исследовании отступов.

Свойство использования эффекта независимости мы установили для метода AdaBoost. Для других модификаций бустинга соотношение (3) может и не выполняться, однако, из этого не следует, что такие модификации будут менее эффективны, поскольку соотношение (3) выведено для распределений. Прямолинейный перенос его на выборку,

возможно, не является лучшим решением, учитывая, например, то, что из мультипликативности вероятностей не следует, что на выборке нужно перемножать частоты [1].

6. Аппроксимирующая способность бустинга

Мы выяснили, что AdaBoost на пороговых классификаторах является по сути способом аппроксимации функции условной вероятности в виде (3). Попробуем выяснить, в чём специфика данного способа аппроксимации, сравнив с классическим подходом к аппроксимации функций с использованием сплайнов.

Для экспериментального исследования зададим (на интервале $[-1, 1]$) следующую функцию условной вероятности

$$g(x) = \frac{1}{2}\left(1 + \sin\left(\frac{2\pi}{1{,}5x + 0{,}3\,\mathrm{sign}(x)}\right)\right)$$

Функция изображена на первой диаграмме рис. 2. Следующие диаграммы изображают аппроксимацию этой функции кубическим сплайном на 20 интервалах по методу максимального правдоподобия и аппроксимации бустингом при 50 и 250 итерациях.

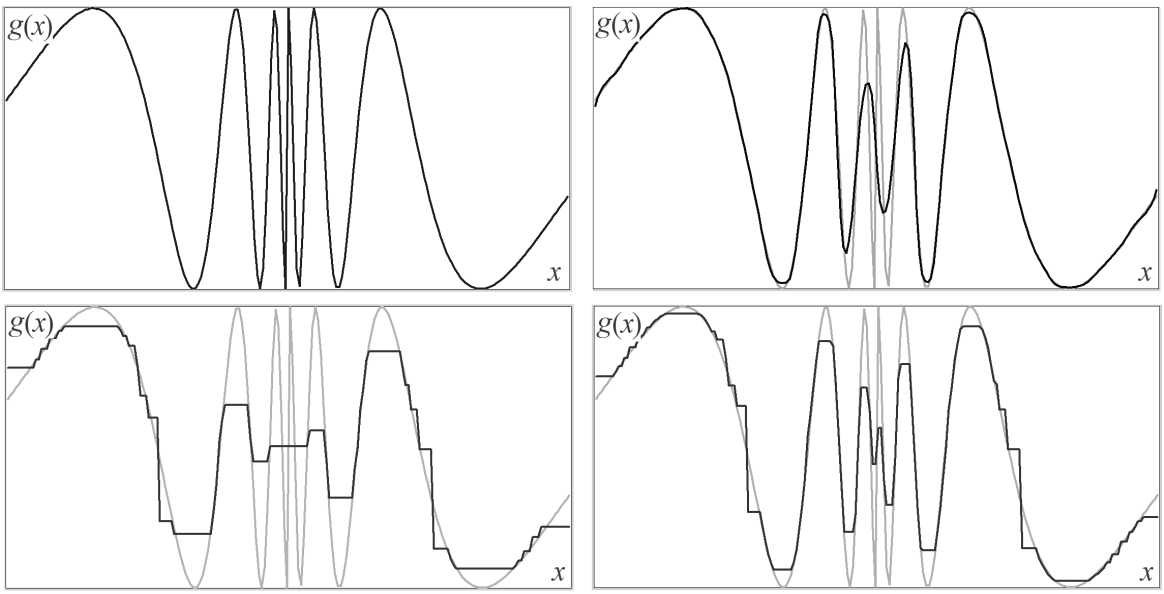

Рис. 2: Модельная функция условной вероятности и её аппроксимации кубическим сплайном на 20 интервалах и бустингом при 50 и 250 итерациях.

На данных иллюстрациях можно заметить, что, в отличие от сплайнов, бустинг производит адаптивную аппроксимацию, в том смысле, что в первую очередь функция приближается на участках, где она изменяется медленно, и только при существенном увеличении числа итераций начинается аппроксимация быстро меняющихся фрагментов. Данное свойство является, очевидно, достоинством бустинга.

Вместе с тем, очевиден и недостаток данного метода: ступенчатый вид решения. Напрашивается идея объединить достоинства подходов, путём конструирования гладких аппроксимаций с учётом адаптации к локальной сложности оцениваемой функции.

7. Распределение отступов

Часто для обоснования бустинга приводят оценки для вероятности ошибочной классификации, использующие понятие отступа.

Для случая конечного множества Λ базовых классификаторов имеет место оценка риска, в соответствии с которой с вероятностью, не меньшей $1-\delta$, для $\theta > 0$ выполняется неравенство:

$$R(\lambda, c) = \mathsf{P}_c(y\beta(x) \leqslant 0) \leqslant \widetilde{\mathsf{P}}(y\beta(x) \leqslant \theta\varkappa) + O\left(\frac{1}{\sqrt{N}}\left(\frac{\ln N \ln|\Lambda|}{\theta^2} - \ln\delta\right)^{1/2}\right), \quad (6)$$

где $\varkappa = \sum_{t=1}^{T} \alpha_t$, а $\widetilde{\mathsf{P}}(\cdot)$ – частота события на выборке.

Выражение в правой части не меньше чем

$$\widetilde{F}(\theta) + O\left(\frac{1}{\theta} \cdot \sqrt{\frac{\ln|\Lambda|}{N}}\right),$$

где $\widetilde{F}(\theta) = \widetilde{\mathsf{P}}(y\beta(x) < \theta\varkappa)$ – эмпирическая (выборочная) функция распредления для величины $\frac{y\beta(x)}{\varkappa}$, которую принято называть отступом.

Такое огрубление сделано, чтобы выделить, от чего наиболее существенно зависит оценка риска, а именно, чтобы показать, что прогнозируемое значение риска тем меньше, чем больше эмпирическое распределение отступа сдвинуто вправо. При этом в качестве важной особенности данной оценки принято отмечать то, что она не зависит явно от количества функций в композиции.

В качестве обоснования (объяснения) эффективности бустинга часто приводится утверждение, что бустинг максимизирует отступ, а в силу приведённой оценки, чем больше отступ, тем меньше предполагаемая вероятность ошибки.

Даже если согласиться с этим утверждением, следует заметить, что оно ещё не объясняет, почему бустинг эффективен при решении практических задач, а фактически лишь заменяет этот вопрос вопросом, почему бустинг увеличивает отступ.

При этом с отступом ситуация неоднозначная. Эксперименты показывают, что распределение отступов с ростом композиции двигается немонотонно, более того, средний отступ может иметь тенденцию к уменьшению. Это происходит, в частности, на рис. 3, где для эксперимента, описанного в предыдущем разделе, приводится эмпирическое распределение отступов и зависимость среднего отступа от числа итераций.

Кроме того, построены примеры, когда предельное значение отступа оказывалось обратно пропорциональным числу различных правил в композиции. Иными словами, чем больше слагаемых (без учёта повторов), тем меньше отступ и тем больше оценка риска.

Теория об отступах не отвечает на крайне важный для понимания эффективности бустинга вопрос: почему на некоторых задачах бустинг неэффективен. Таких задач в некотором смысле мало (и они, как правило, искусственные). Но достаточно того, что они есть. А это значит, что объяснение эффективности бустинга должно учитывать не только свойства метода, но и свойства решаемой задачи.

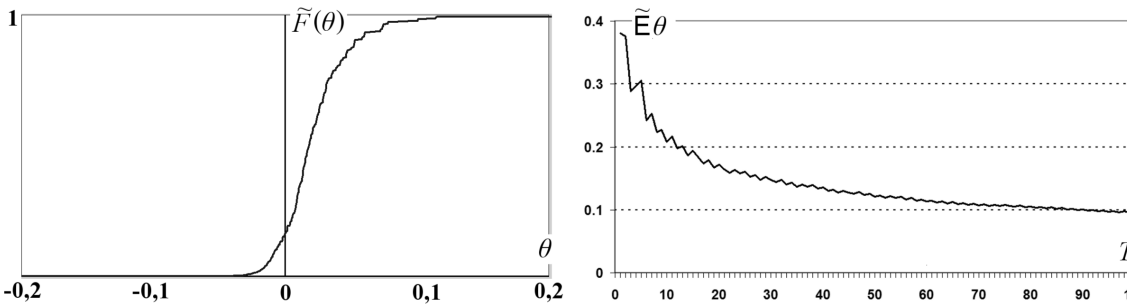

Рис. 3: Эмпирическое распределение отступов после 80000 итераций и зависимость среднего отступа от числа итераций

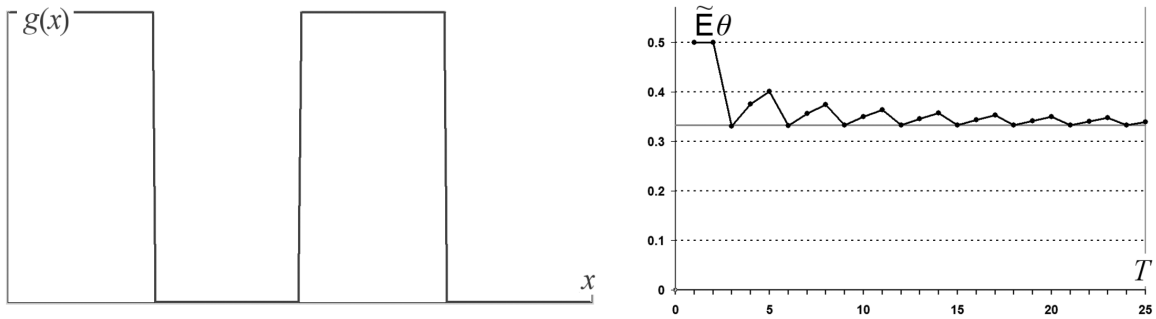

Рис. 4: Модельная функция условной вероятности и зависимость среднего отступа от числа итераций.

Рассмотрим семейство распределений с равномерным $\mathsf{P}(dx)$ на интервале $[0,1]$ и следующей функцией условной вероятности:

$$g^K(x) = (1 - ([Kx] \mod 2))(1-\rho) + 0{,}5\rho, \tag{7}$$

где K – параметр, задающий число областей постоянства, а ρ – параметр, определяющий байесовский уровень ошибки. Пример такой функции при $K = 4$ и $\rho = 0$ приведён на левой диаграмме рис. 4.

Утверждение 2. *Для предельного значения отступа, получаемого методом AdaBoost с пороговыми решающими функциями на модели* (7) *для чётных K при увеличении числа итераций, справедливо следующее выражение*

$$\lim_{T\to\infty} \widetilde{\mathsf{E}}\theta = \frac{1}{K-1} \cdot (1-\rho)$$

ДОКАЗАТЕЛЬСТВО. Для доказательства явным образом найдём композицию, точно описывающую рассматриваемую функцию условной вероятности.

Поскольку заданная $g^K(x)$ изменяется только в точках $x = t/K$, $t = 1, ..., K-1$, искомая композиция имеет вид

$$\beta(x) = a_0 + \sum_{t=1}^{K-1} a_t \operatorname{sign}\left(x - \frac{t}{K}\right)$$

Из равенства $g(x) = \frac{1}{1+e^{-2\beta(x)}}$, или $2\beta(x) = \ln g(x) - \ln(1-g(x))$, получаем систему уравнений для неизвестных a_t

$$\begin{array}{lllllll} a_0 & -a_1 & -a_2 & \dots & -a_{K-1} & = & C \\ a_0 & +a_1 & -a_2 & \dots & -a_{K-1} & = & -C \\ a_0 & +a_1 & +a_2 & \dots & -a_{K-1} & = & C \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_0 & +a_1 & +a_2 & \dots & +a_{K-1} & = & -C, \end{array}$$

где $2C = \ln(0{,}5\rho) - \ln(1-0{,}5\rho)$. Каждое уравнение соответствует интервалу постоянства $g(x)$.

Матрица системы невырождена, поэтому решение единственно. Непосредственной проверкой (по индукции) легко убедиться, что решением является $a_0 = 0$ и $a_t = (-1)^t C$, $t > 0$.

Заметим, что для полученного решения $\varkappa = \sum_{t=1}^{K-1} |a_t| = (K-1) \cdot C$. Модули возникают, поскольку в композиции коэффициенты должны быть неотрицательны, и знаки нужно переместить в базовые классификаторы.

При построении решающих функций на распределениях средним отступом в точке x будет величина $(2g(x)-1) \cdot \frac{\beta(x)}{\varkappa}$, которая получается, если в выражение $\frac{y\beta(x)}{\varkappa}$ подставить $y = (+1) \cdot g(x) + (-1) \cdot (1-g(x))$.

Подставим решение в выражение для отступа. Имеем искомое соотношение

$$(2g(x)-1) \cdot \frac{\beta(x)}{\varkappa} = (1-\rho) \cdot \frac{C}{C \cdot (K-1)} = (\rho - 1) \cdot \frac{-C}{C \cdot (K-1)} = \frac{1-\rho}{K-1}.$$

Согласно Лемме 1, бустинг на пороговых классификаторах при увеличении числа слагаемых в композиции даёт решение, приближающееся к $g(x)$. При этом мы установили, что такое решение единственное с точностью до групировки подобных слагаемых, поэтому отступ для найденного решения совпадает с отступом для результата бустинга.

Заметим, что если в построенной композиции окажутся присутствующими противоположные базовые классификаторы ($\operatorname{sign}(x-c)$ и $-\operatorname{sign}(x-c)$), то это не изменит решения, но увеличит $\varkappa$ и, соответственно, уменьшит отступ. Однако, как показывает практика, бустинг не включает в композицию противоположных базовых классификаторов, но даже если окажется, что подобные классификаторы в некоторых случаях всё же возникают, мы можем явным образом добавить в алгоритм бустинга операцию по сокращению слагаемых.

С учётом приведённых замечаний, утверждение доказано. □

Приведённый пример иллюстрирует очень важный факт: оценка (6) существенно зависит от числа классификаторов в композиции, хотя эта зависимость неявная.

Классические сложностные оценки риска [2] зависят от эмпирического риска, сложности класса решающих функций [6] и объёма выборки.

Полученные впоследствии новые оценки, подобные (6), вместо эмпирического риска используют понятие отступа. Так оценка (6) зависит от отступа, сложности базового класса решающих функций и объёма выборки. Если считать отступ непосредственным обобщением эмпирического риска, то возникает иллюзия, что оценка не зависит от сложности композиции (поскольку в оценке фигурирует только сложность базового класса).

Однако непосредственным обобщением эмпирического риска (вернее дополнительной к нему величины — доли правильно классифицированных объектов) следует считать ненормированный отступ, т.е. величину $y\beta(x)$, поскольку именно ненормированная $\beta(x)$ фигурирует в оценке вероятности и именно она имеет смысл степени уверенности в решении.

В оценке (6) фигурирует нормированный отступ, где в качестве нормировки выступает $\varkappa$. В таком виде отступ уже не является непосредственным обобщением эмпирического риска, а включает в себя параметр сложности композиции, роль которого играет $\varkappa$. Если вынести $\varkappa$ из выражения для отступа, то в оценке (6) в роли сложности метода классификации получим величину $\varkappa^2 \ln |\Lambda|$. Как видим, длина композиции входит в оценку риска. При этом учитывается число существенно различных [4] решающих функций.

8. Заключение

Эффективность бустинга на реальных задачах означает, что бустинг «угадывает» какие-то особенности, присущие большинству реальных задач. В настоящей работе на роль такой особенности предлагается свойство независимости. Это, однако, не строгая независимость переменных, а более слабые условия, которые, тем не менее, оправдано считать проявлением эффекта независимости в более общем понимании.

Бустинг на пороговых классификаторах является разновидностью непараметрической логистической регрессии, также его можно считать разновидностью (существенно обобщённого) наивного байесовского классификатора. Бустинг реализует интуитивно «разумный» вариант непараметрической аппроксимации условной вероятности.

Используя в методе AdaBoost в качестве базового классификатора подходящие методы (например, решающие деревья), можно получить решения, являющиеся в некотором смысле аналогом решений на основе ряда Бахадура, используемого для бинарных переменных. Аналогия состоит в том, что бустинг позволяет последовательно увеличивать сложность модели, включая в рассмотрение зависимости в парах переменных, тройках и т.д.

Таким образом, гипотеза независимости представляется актуальной при разработке методов построения решающих функций, но нуждается в разумном ослаблении, одним из вариантов которого связан с методом AdaBoost.

Список литературы

1. *Боровков А. А.* О задаче распознавания образов // Теория вероятностей и ее применение. — 1971. — Т. 16, № 1. — С. 132–136.

2. *Вапник В. Н., Червоненкис А. Я.* Теория распознавания образов. — Москва: Наука, 1974. — 415 с.

3. *Викентьев А. А., Викентьев Р. А.* Расстояния и меры недостоверности на высказываниях n-значной логики // Вестник Новосибирского государственного университета. Серия: Математика, механика, информатика. 2011, /No. 2, С. 51–64.

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

5. *Гимади Э. Х., Истомин А. М., Рыков И. А.* Задача о двух коммивояжерах с ограничениями на пропускные способности ребер графа с различными весовыми функциями // Вестник Новосибирского государственного университета. Серия: Математика, механика, информатика. 2014, /No. 3, С. 3–18.

6. *Донской В. И.* Колмогоровская сложность классов общерекурсивных функций с ограниченной ёмкостью. // *Таврический вестник информатики и математики.* 2005, /No. 1, 25–34.

7. *Журавлев Ю. И.* Об алгебраическом подходе к решению задач распознавания или классиикации // Проблемы кибернетики. Вып. 33. М.: Наука, 1978. — С. 5–68.

8. *Загоруйко Н. Г., Борисова И. А., Кутненко О. А., Дюбанов В. В.* Построение сжатого описания данных с использованием функции конкурентного сходства // Сибирский журнал индустриальной математики. 2013, Т. 16, /No. 1 (53), С. 29–41.

9. *Загоруйко Н. Г., Татарников В. В.* Обнаружение ошибок и заполнение пробелов в кубах данных // Сибирский журнал индустриальной математики. 2014, Т. 17, /No. 2 (58), С. 50–58.

10. *Kel'manov A. V., Khamidullin S. A.* An Approximating Polynomial Algorithm for a Sequence Partitioning Problem // Journal of Applied and Industrial Mathematics. 2014. Vol. 8, No.2. P. 236–244.

11. *Кельманов А. В., Хандеев В. И.* Рандомизированный алгоритм для одной задачи двухкластерного разбиения множества векторов // Журнал вычислительной математики и математической физики. 2015, Т. 55, /No. 2, С. 330–339.

12. *Кочетов Ю. А., Сивых М. Г., Хмелёв А. В., Яковлев А. В.* Методы локального поиска для одной задачи о перестановке столбцов бинарной матрицы // Вестник Новосибирского государственного университета. Серия: Математика, механика, информатика. 2012, /No. 1, С. 91–101.

13. *Лбов Г. С., Старцева Н. Г.* Сравнение алгоритмов распознавания с помощью программной системы «Полигон» // *Анализ данных и знаний в экспертных системах.* Новосибирск, 1990. Вып. 134: Вычислительные системы. С. 56–66.

14. *Лбов Г. С., Старцева Н. Г.* Логические решающие функции и вопросы статистической устойчивости решений. — Новосибирск: Издательство Института математики, 1999. — 212 с.

15. *Лемешко Б. Ю., Лемешко С. Б., Постовалов С. Н.* Сравнительный анализ мощности критериев согласия при близких альтернативах. II. Проверка сложных гипотез // *Сибирский журнал индустриальной математики.* 2008. — Т.11. — №4(36). — С. 78–93.

16. *Лисицин Д. В., Гаврилов К. В.* Оценивание параметров финитной модели, устойчивое к нарушению финитности // Сибирский журнал индустриальной математики. 2013, Т. 16, /No. 2 (54), С. 109–121.

17. *Недельк о В. М.* Некоторые вопросы оценивания качества методов построения решающих функций // Вестник Томского государственного университета. Управление, вычислительная техника и информатика, Томск: ТГУ, 2013. № 3 (24). — С. 123–132

18. *Недельк о В. М.* Регрессионные модели в задаче классификации // Сибирский журнал индустриальной математики. 2014, Т. 17, /No. 1 (57), С. 86–98.

19. *Недельк о В. М.* Исследование погрешности оценок скользящего экзамена // Машинное обучение и анализ данных. 2013, Т. 1, /No. 5, С. 526–533.

20. *Хачай М. Ю.* О вычислительной сложности задачи о минимальном комитете и смежных задач // ДАН. — 2006. — Т. 406, № 6. — С. 742–745.

21. *Freund Y., Schapire R. E.* Experiments with a new boosting algorithm // *In Machine Learning: Proceedings of the Thirteenth International Conference*, 1996. Pp. 148–156.

22. *Schapire R. E., Freund Y., Bartlett P., Lee W. S.* Boosting the Margin: a New Explanation for the Effectiveness ot Voting Methods // *The Annals of Statistics*. 1998, Vol. 26, No. 5, 1651–1686.

23. *David Mease and Abraham Wyner.* Evidence Contrary to the Statistical View of Boosting. // *J. Mach. Learn.* Res. 9 (June 2008), 131–156.

Список литературы

1. *Borovkov A. A.* On Pattern Recognition Problem [in Russian] // Theory of Probability and its Applications. 1971. Vol. 16, No 1. P. 132–136.

2. *Vapnik V. N., Chervonenkis A. Ya.* Theory of Pattern Recognition [in Russian]. Moscow: Nauka, 1974. 415 p.

3. *Vikent'ev A. A., Vikent'ev R. A.* Distances and Uncertainty Measures on Satatements of n-fold Logics [in Russian] // Vestnik Novosibirskogo Gosudarstvennogo Universiteta. Seriya Matematika, Mekhanika, Informatika. 2011, No. 2, P. 51–64.

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

5. *Gimadi E. Kh., Istomin A. M., Rykov I. A.* On 2-Capacitated Peripatetic Salesman Problem with Different Weight Functions [in Russian] // Vestnik Novosibirskogo Gosudarstvennogo Universiteta. Seriya Matematika, Mekhanika, Informatika. 2014, No. 3, P. 3–18.

6. *Donskoi V. I.* Kolmogorov complexity of classes of general recursive functions with bounded dimension [in Russian]. // *Tavricheskiĭ Vestnik Informatiki i Matematiki.* 2005, No. 1, 25–34.

7. *Zhuravlev Yu. I.* On Algebraic Approach to Solving of Pattern Recognition or Classification Tasks [in Russian] // Problems of Cybernetics. Issue 33. Moscow: Nauka, 1978. — Pp. 5–68.

8. *Zagoruiko N. G., Borisova I. A., Kutnenko O. A., Dyubanov V. V.* Constructing the compressed description of dataset by the function of rival similarity [in Russian] // Sibirskii Zhurnal Industrial'noi Matematiki. 2013, Vol. 16, No. 1 (53), P. 29–41.

9. *Zagoruiko N. G., Tatarnikov V. V.* Detecting mistakes and filling gaps in data cubes [in Russian] // Sibirskii Zhurnal Industrial'noi Matematiki. 2014, Vol. 17, No. 2 (58), P. 50–58.

10. *Kel'manov A. V., Khamidullin S. A.* An Approximating Polynomial Algorithm for a Sequence Partitioning Problem // Journal of Applied and Industrial Mathematics. 2014. Vol. 8, No.2. P. 236–244.

11. *Kel'manov A. V., Khandeev V. I.* A Randomized Algorithm for Two-Cluster Partition of a Set of Vectors // Computational Mathematics and Mathematical Physics. 2015, Vol. 55, No. 2, pp. 330–339.

12. *Kochetov Yu. A., Sivykh M. G., Khmelev A. V., Yakovlev A. V.* Local search methods for a column permutation problem for the binary matrix [in Russian] // Vestnik Novosibirskogo Gosudarstvennogo Universiteta. Seriya Matematika, Mekhanika, Informatika. 2012, No. 1, P. 91–101.

13. *Lbov G. S., Startseva N. G.* A Comparison of Pattern Recognition Algorithms with the Programm Tool "Poligon"[in Russian] // *Analysis of Data and Knowledges in Expert Systems.* Novosibirsk, 1990. Issue 134: Computational Systems. P. 56–66.

14. *Lbov G. S., Startseva N. G.* Logical Decision Functions and the Problem of Statistical Robustness of Solutions [in Russian]. Novosibirsk: Institute of Mathematics SB RAS, 1999. — 212 p.

15. *Lemeshko B. Yu., Lemeshko S. B., Postovalov S. N.* Comparative Analysis of the Power of the Goodness-of-fit with Similar Alternatives. II. Test Complex Hypotheses [in Russian] // *Sibirskii Zhurnal Industrial'noi Matematiki.* 2008. Vol.11. No 4 (36). Pp. 78–93.

16. *Lisitsin D. V., Gavrilov K. V.* Estimation of the parameters of a compactly-supported model stable under the violation of compact supportedness [in Russian] // Sibirskii Zhurnal Industrial'noi Matematiki. 2013, Vol. 16, No. 2 (54), P. 109–121.

17. *Nedelko V.* Some aspects of estimating a quality of decision functions construction methods [in Russian] // Tomsk state university. Journal of control and computer science, Tomsk: TSU, 2013. N 3 (24). — p. 123–132.

18. *Nedel'ko V. M.* Regression models in the classification problem [in Russian] // Sibirskii Zhurnal Industrial'noi Matematiki. 2014, Vol. 17, No. 1 (57), P. 86–98.

19. *Nedel'ko V. M.* Investigation of accuracy of crossvalidation [in Russian] // Machine Learning and Data Analysis. 2013, Vol. 1, No. 5, P. 526–533.

20. *Khachay M. Yu.* Computational Complexity of the Minimum Committee Problem and Related Problems // Doklady Mathematics. 2006. Vol. 73, No. 1, pp. 138–141.

21. *Freund Y., Schapire R. E.* Experiments with a new boosting algorithm // *In Machine Learning: Proceedings of the Thirteenth International Conference*, 1996. Pp. 148–156.

22. *Schapire R. E., Freund Y., Bartlett P., Lee W. S.* Boosting the Margin: a New Explanation for the Effectiveness ot Voting Methods // *The Annals of Statistics.* 1998, Vol. 26, No. 5, 1651–1686.

23. *David Mease and Abraham Wyner.* Evidence Contrary to the Statistical View of Boosting. // *J. Mach. Learn.* Res. 9 (June 2008), 131–156.

Материал поступил в редколлегию 15.11.2014

Адрес автора
НЕДЕЛЬКО Виктор Михайлович
Институт математики СО РАН
Новосибирский государственный технический университет
Новосибирский государственный университет
ул. Пирогова, 2, Новосибирск, 630090, Россия
e-mail: nedelko@math.nsc.ru