МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ (государственный университет)
ФАКУЛЬТЕТ УПРАВЛЕНИЯ И ПРИКЛАДНОЙ МАТЕМАТИКИ
ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР ИМ. А. А. ДОРОДНИЦЫНА РАН
КАФЕДРА «ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ»

Гуз Иван Сергеевич

Синтез монотонных алгоритмических композиций на основе анализа их обобщающей способности

511656 — Математические и информационные технологии

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРА

Научный руководитель:
н.с. ВЦ РАН, к.ф.-м.н. К. В. Воронцов

Москва 2008 г.

Содержание

ВВЕДЕНИЕ

Проблема синтеза алгоритмов на основе обучения по конечным выборкам прецедентов и изучения их качества на всем множестве прецедентов является одним из важнейших вопросов теории обучения систем. В качестве прецедентов рассматриваются пары: объект, описанный набором признаков, и класс, к которому принадлежит объект. Задача классификации состоит в том, чтобы на основании известного конечного множества прецедентов научиться определять классы для априори неизвестных объектов. Обучение или настройка параметров алгоритма на обучающей выборке происходит путем решения задачи численной оптимизации. Практика показала, что чем точнее настраивать алгоритм на обучающей выборке, тем хуже настроенный алгоритм будет классифицировать контрольную выборку. Таким образом, встает вопрос о качественной настройки алгоритма с точки зрения способности к обобщению.

На текущий момент существует несколько научных подходов к изучению данного вопроса. В основе первого подхода лежит статистическая теория Вапника-Червоненкиса. В ней обобщающая способность определяется как вероятность ошибки алгоритма, полученного в результате обучения, либо как частота его ошибок на некоторой независимой и, вообще говоря, неизвестной контрольной выборке. К сожалению, оценки Вапника-Червоненкиса сильно завышены, что приводит к требованию слишком длинных обучающих выборок (10^5–10^6 объектов), а некоторые семейства алгоритмов находятся за пределами теории и для них данный подход не дает оценок. Второй подход использует комбинаторные методы для расчета оценок обобщающей способности. Оценки, полученные таким способом, справедливы для любого метода обучения и любой конечной выборки, не обязательно случайной, независимой, одинаково распределённой.

Работа состоит из двух глав. В первой главе исследуются границы применимости оценок семейства монотонных алгоритмов, полученных в [2] и предпринимается попытка их улучшения. Во второй главе объясняются причины переобучения алгоритма построения монотонных алгоритмических композиций, описанного в [3] и доказывается новый алгоритм синтеза монотонных алгоритмических композиций на основе анализа функционала обобщающей способности.

Глава 1. Исследование оценок функционала обобщающей способности для монотонных классификаторов.

В данной главе исследуются границы применимости оценок обобщающей способности семейства монотонных классификаторов [1], полученных в [2] с помощью комбинаторного подхода, и предпринимается попытка их улучшения. В конце главы описаны вычислительно эффективные алгоритмы расчета полученных оценок и проводятся численные эксперименты, показывающие качество полученных оценок и границы их применимости.

1.1. Улучшение существующей оценки

В работе рассматривается задача классификации, в которой множество объектов $X \subset R^k$ частично упорядочено, а множество классов состоит из двух элементов $Y = \{-1,1\}$, а в качестве семейства классификаторов A рассматриваются монотонные функции. К требованиям монотонности часто прибегают на практике, например в меди-

цинских задачах, когда рассуждения о состоянии больного имеют вид «чем больше температура больного – тем больше вероятность осложнения». Поэтому, исследование обобщающей способности монотонных классификаторов, имеет огромное практическое значение.

Основная цель данной работы – улучшение оценок обобщающей способности, полученных в [2]. Для этого рассмотрим несколько понятий, введенных в [2].

Степенью немонотонности выборки X^L называется наименьшая частота ошибок, допускаемая на ней монотонным классификатором:

$$\delta(X^L) = \min_{a \in A} \nu(a, X^L). \qquad (1.1)$$

При этом выборка называется монотонной, если

$$x_i \le x_j \Rightarrow y_i \le y_j \ \forall i, j = 1 \ldots L. \qquad (1.2)$$

В разделе 3 будет описан эффективный алгоритм приближенного подсчета степени немонотонности для произвольной выборки объектов.

Для каждого объекта выборки рассмотрим 4 множества, которые будем называть конусами:

$$W_i^+(x_i) = \begin{cases} x_k \in X^L \mid x_i < x_k ; y_i = y_k = -1 \\ x_k \in X^L \mid x_k < x_i ; y_i = y_k = +1 \end{cases} \text{ – прямой положительный клин;}$$

$$W_{-i}^+(x_i) = \begin{cases} x_k \in X^L \mid x_i < x_k ; y_i = -1; y_k = +1 \\ x_k \in X^L \mid x_k < x_i ; y_i = +1; y_k = -1 \end{cases} \text{ – прямой отрицательный клин;}$$

$$W_i^-(x_i) = \begin{cases} x_k \in X^L \mid x_i > x_k ; y_i = y_k = -1 \\ x_k \in X^L \mid x_k > x_i ; y_i = y_k = +1 \end{cases} \text{ – обратный положительный клин;}$$

$$W_{-i}^-(x_i) = \begin{cases} x_k \in X^L \mid x_i > x_k ; y_i = -1; y_k = +1 \\ x_k \in X^L \mid x_k > x_i ; y_i = +1; y_k = -1 \end{cases} \text{ – обратный отрицательный клин.}$$

Определим вес каждого клина w как мощность соответствующего множества.

Профилем монотонности выборки X^L называется называется функция $M(m, X^L)$, выражающая долю объектов выборки мощности m:

$$M(m, X^L) = \frac{1}{L} \sum_{i=1}^{L} [w_i^+ = m], \text{ где } i = 1 \ldots m.$$

В [2] получена следующая оценка обобщающей способности класса монотонных функций:

$$Q^{lk}(\mu, X^L) \le \sum_{m=0}^{\delta L+k-1} M(m, X^L) \sum_{s=\max\{0,m-k+1\}}^{\min\{\delta L,l,m\}} \frac{C_m^s C_{L-1-m}^{l-s}}{C_{L-1}^l}. \qquad (1.3)$$

Здесь μ – метод, минимизирующий эмпирический риск, то есть настраивающий монотонный классификатор наилучшим образом на обучающей выборке; l – длина обучающей выборки, а k – длина контрольной выборки.

Данная оценка выводится из точной оценки обобщающей способности:

$$Q^{lk} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{k}\sum_{x\in X_n^k} I(x, \mu(X_n^l)) = \frac{1}{k}\sum_{i=1}^{L}\frac{1}{N}\underbrace{\sum_{n=1}^{N}[x_i \in X_n^k]I(x_i, \mu(X_n^l))}_{N_i}. \qquad (1.4)$$

Здесь $N = C_L^k$ – число способов разбиения выборки на обучающую и контрольную, а N_i выражает число разбиений выборки, при которых объект x_i оказывается в контрольной подвыборке и монотонный алгоритм, обученный на обучающей подвыборке, допускает на нем ошибку классификации.

Для оценки N_i рассмотрим произвольный объект x_i из обучающей выборки. Рассмотрим число s объектов обучающей выборки, которое может находиться внутри прямого положительного клина объекта x_i. В [2] показано, что это число ограничено:

$$s \in (\max\{0, w_i^+ - k + 1\}, \min\{\delta L, l, w_i^+\}).$$

Имеется $C_{w_i^+}^s$ способов выбрать s обучающих объектов из клина w_i^+. Для каждого из этих способов имеется $C_{L-1-w_i^+}^{l-s}$ вариантов выбрать $l-s$ обучающих объектов из множества $X^L / (W_i^+ \cup x_i)$. В итоге получаем оценку числа разбиений:

$$N_i \le \sum_{s=\max\{0,w_i^+-k+1\}}^{\min\{\delta L,l,m\}} C_{w_i^+}^s C_{L-1-w_i^+}^{l-s}. \qquad (1.5)$$

Неточность формулы (1.3) заключается в неточности оценки N_i. Рассмотрим несколько причин этой неточности.

1. Рассмотрим линейно упорядоченное множество из четырех объектов, имеющих один признак, причем два объекта принадлежат классу -1, а два оставшихся – классу +1. Пусть $l = k = 2$. Монотонный классификатор в данном случае будет являться решающим правилом

$$F(x) = \begin{cases} -1, \textit{ если } x < c \\ +1, \textit{ если } x \ge c \end{cases}. \qquad (1.6)$$

Рассчитаем точную оценку функционала обобщающей способности по формуле (1.4). Для этого изобразим все $N = C_4^2 = 6$ случаев разбиения выборки на контрольную и обучающую подвыборки. Настройка классификатора на обучающей выборке будет сводиться к выбору параметра c в (1.6) таким образом, чтобы число ошибок на обучающей подвыборке было минимальным, а число ошибок на контрольной подвыборке – максимальным, поскольку контрольная подвыборка априори неизвестена и мы

должны рассчитывать на наихудший вариант классификации. На рисунках символом I обозначено значение параметра c. Эффективный приближенный алгоритм настройки монотонного классификатора для общего случая $X \subset R^n$ описан в разделе 3.

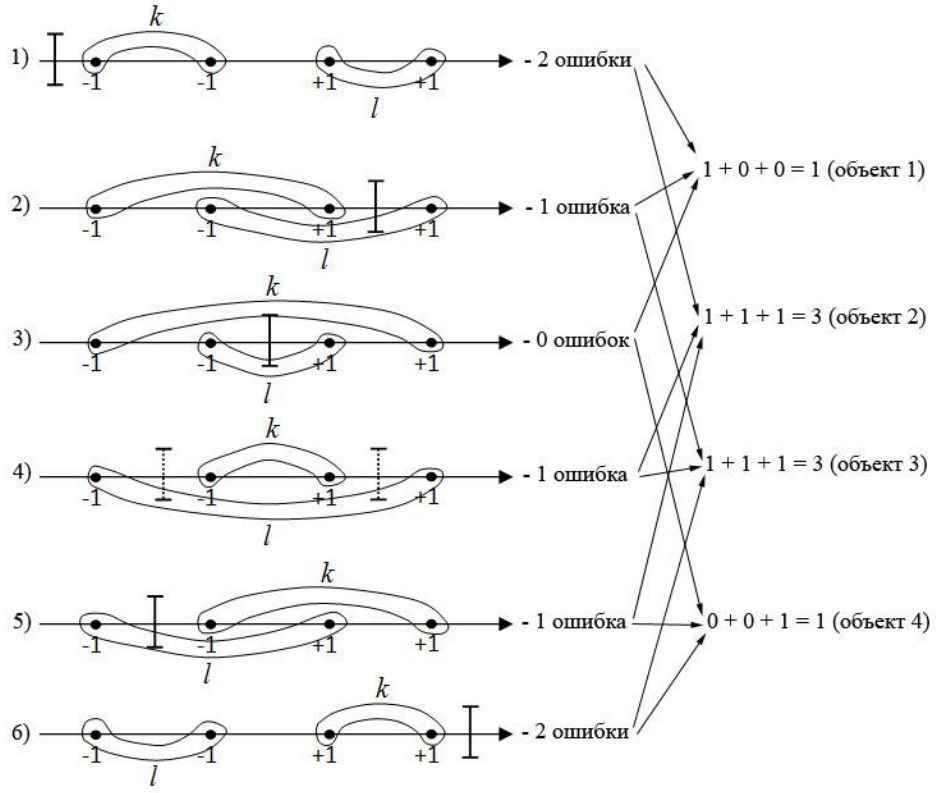


Рис. 1

Для каждого случая на рисунке 1 подсчитано число ошибок на контрольной подвыборке. Для точной оценки функционала число ошибок равно 2 + 1 + 0 +1 + 1 + 2 = 7. Подсчет N_i где $i = 1 \ldots 4$ из (1.5) осуществлен в правой части рисунка, где стрелочками указано, в каких случаях соответствующий объект попадает в контрольную подвыборку. Объекты нумеруются слева направо с 1 до 4. Общее число ошибок равно 1 + 3 + 3 + 1 = 8 > 7. Дополнительная ошибка возникла в результате неоднозначности выбора параметра c в случае 4). На рисунке изображено 2 способа выбора этого параметра, при которых число ошибок на обучении равно 0, а число ошибок на контроле равно 1. При подсчете N_2 был учтен один способ выбора c, а при подсчете N_3 – другой, хотя выбор этого параметра для каждого случая однозначен. Таким образом, неоднозначность построения монотонного классификатора на обучении приводит к завышению оценки (1.3) на число этих неоднозначностей.

2. Рассмотрим случай, изображенный на рисунке 2, а именно подсчет N_3

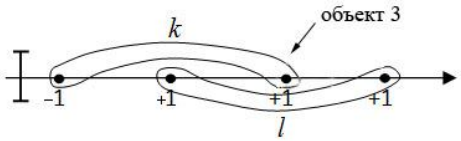


Рис. 2

Этот случай дает вклад в N_3 хотя при настройке на указанной обучающей выборке монотонный алгоритм никогда не даст ошибку на объекте 3. То есть, в оценке (1.3) учитываются случаи, нарушающие монотонность классификатора.

3. Рассмотрим еще один случай для подсчета N_2, изображенный на рисунке 3.

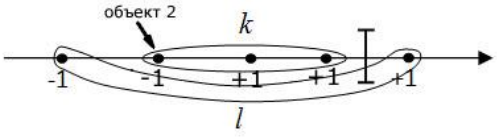


Рис. 3

В данной ситуации монотонный алгоритм настроится на контрольной выборке способом, максимизирующим число ошибок на контроле и на объекте 2 ошибки не будет, хотя при расчете N_2 ошибка будет учтена. В общем случае это означает, что если для какого-то объекта, принадлежащего классу -1, в клине W_{-i}^{+} оказываются хотя бы 2 объекта, меньшие любого объекта из W_i^{+}, то из N_i необходимо вычесть 1.

Для улучшения оценки (1.3) постараемся как можно сильнее уменьшить вклад второго случая в общую неточность оценки. Этого уменьшения можно добиться двумя независимыми способами.

1. Рассмотрим подвыборку, изображенную на рисунке 4.

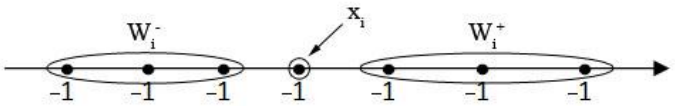


Рис. 4

Очевидно, что если вся обучающая выборка распределится частично внутри клина W_i^{+} и частично внутри клина W_i^{-}, то на объекте x_i монотонный классификатор никогда не допустит ошибку. Поэтому, получаем первую поправку, которую необходимо вычесть из оценки для N^i:

$$N_i^{(1)} = \sum_{s=\max\{1,l-w_i^-,w_i^+-k+1\}}^{\min\{\delta L,l,w_i^+\}} C_{w_i^+}^{s} C_{w_i^-}^{l-s}. \qquad (1.7)$$

2. Если в клине W_i^{+} окажется больше половины обучающей подвыборки, то монотонный классификатор также никогда не ошибется на объекте x_i, поскольку в этом случае он ошибся бы более чем на половине объектов обучения, что невозможно. Получаем вторую поправку:

$$N_i^{(2)} = \sum_{s=[\frac{l}{2}]+1}^{\min\{\delta L,l,w_i^+\}} (L-w_i^+-w_i^-) C_{w_i^+}^{s} C_{L-w_i^+}^{l-s-1}. \qquad (1.8)$$

Коэффициент $(L-w_i^+-w_i^-)=C_{(L-w_i^+-w_i^-)}^{1}$ введен для того, чтобы в выборке объектов, не попавших в клин W_i^{+}, был хотя бы один объект, не попавший в клин W_i^{-}, иначе $N_i^{(1)}$ может частично учитываться в $N_i^{(2)}$.

Таким образом, улучшенная оценка для N_i имеет следующий вид:

$$N_i \le \sum_{s=\max\{0,w_i^+-k+1\}}^{\min\{\delta L,l,m\}} C_{w_i^+}^s C_{L-1-w_i^+}^{l-s} - \sum_{s=\max\{1,l-w_i^-,w_i^+-k+1\}}^{\min\{\delta L,l,w_i^+\}} C_{w_i^+}^s C_{w_i^-}^{l-s} - \sum_{s=[\frac{l}{2}]+1}^{\min\{\delta L,l,w_i^+\}} (L-w_i^+-w_i^-) C_{w_i^+}^s C_{L-w_i^+}^{l-s-1}.$$

1.2. Построение эффективных алгоритмов.

1.2.1. Алгоритм приближенного подсчета степени немонотонности для произвольной выборки объектов

Алгоритм состоит из двух этапов:

1) Из исходной выборки выделяем монотонную выборку наилучшим образом. Для каждого объекта из выборки определим число объектов, с которыми он нарушает монотонность. Будем последовательно исключать из выборки объекты, у которых это число максимально до тех пор, пока выборка не станет монотонной, то есть не останется объектов, у которых это число отлично от нуля. Обозначим оставшееся множество M, оно состоит из двух подмножеств: $M^- = \{x_i \in M : y_i = -1\}$ и $M^+ = \{x_i \in M : y_i = +1\}$
2) Воспользуемся монотонной функцией F, построенной в [1] для классификации исключенных объектов.

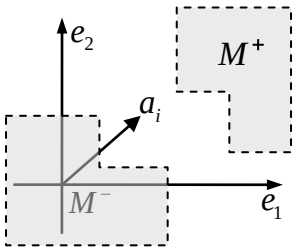


Рис. 5

Введем расстояние от классифицируемого объекта a_i до каждого из множеств:

Расстояние до множества M^-: $\rho(a_i, M^-) = \min_{x \in M^-}\{\max((a_i^1 - x^1)_+, \ldots, (a_i^n - x^n)_+)\}$

Расстояние до множества M^+: $\rho(a_i, M^+) = \min_{x \in M^+}\{\max((x^1 - a_i^1)_+, \ldots, (x^n - a_i^n)_+)\}$

Индекс «+» обозначает операцию срезки: $x_+ = x$ при $x \ldots 0$, и $x_+ = 0$ при $x < 0$.

Если $M^- = \{\varnothing\} : \rho(a_i, M^-) = +\infty \; \forall a_i \in R^n$

Если $M^+ = \{\varnothing\} : \rho(a_i, M^+) = +\infty \; \forall a_i \in R^n$

Если $\rho(a_i, M^-) < \rho(a_i, M^+)$, то $F(a_i) = -1$, в противном случае $F(a_i) = +1$, то есть к какому из множеств ближе в смысле введенного расстояния объект – к соответствующему классу он и принадлежит. В [1] доказывается, что построенная функция является монотонной.

В качестве степени немонотонности δ будем брать долю ошибок классификации удаленных из выборки объектов с помощью описанной функции к общему числу объектов в выборке.

1.2.2. Алгоритм оптимальной настройки монотонного классификатора на обучающей выборке, дающего наихудший результат на контрольной выборке.

Данный алгоритм используется при подсчете числа ошибок при вычислении точного значения функционала обобщающей способности (1.4)

Алгоритм состоит из двух этапов:

1) Из обучающей выборки X^l строим монотонную выборку методом, описанным в пункте 1 предыдущего алгоритма. После этого, аналогичным образом строим множества M^- и M^+. Если множество M^- оказалось пустым, то добавляем в него произвольный объект x^-, такой, что $x^- < x_i \ \forall x_i \in X^L$. Если множество M^+ оказалось пустым, то добавляем в него произвольный объект x^+, такой, что $x^+ > x_i \ \forall x_i \in X^L$.
2) Упорядочим все объекты контрольной выборки X^k по убыванию величины $\rho(x_i, M^+) - \rho(x_i, M^-)$. Решающее правило монотонного классификатора будет иметь вид:

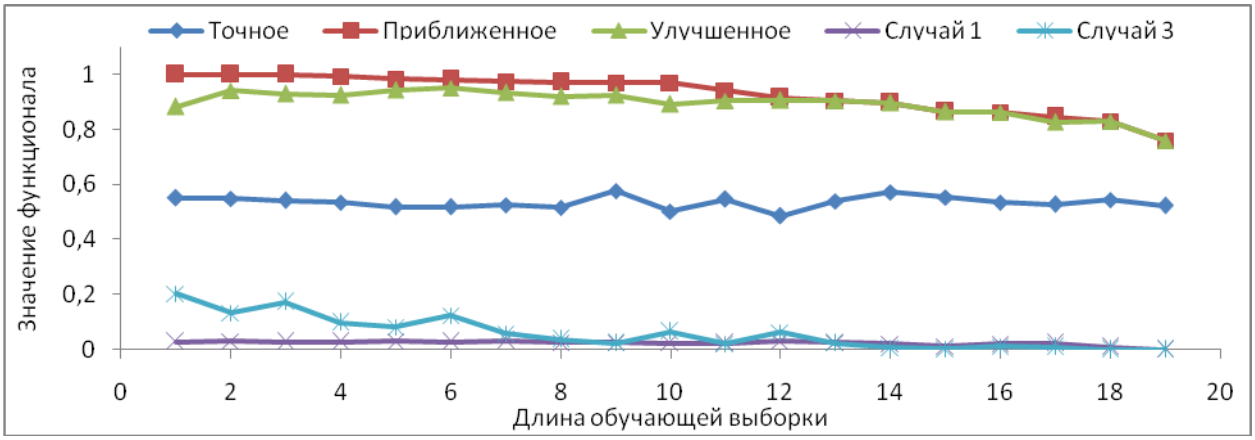
$$F(x) = \begin{cases} -1, \text{ если } \rho(x_i, M^+) - \rho(x_i, M^-) < c \\ +1, \text{ если } \rho(x_i, M^+) - \rho(x_i, M^-) \geq c \end{cases}.$$

Значения параметра c, при котором число ошибок на контрольной выборке максимально находится полным перебором, сложность которого равна k:

$$c^* = \arg\min \sum_{x_i \in X^k} (\rho(x_i, M^+) - \rho(x_i, M^-) - c) y_i.$$

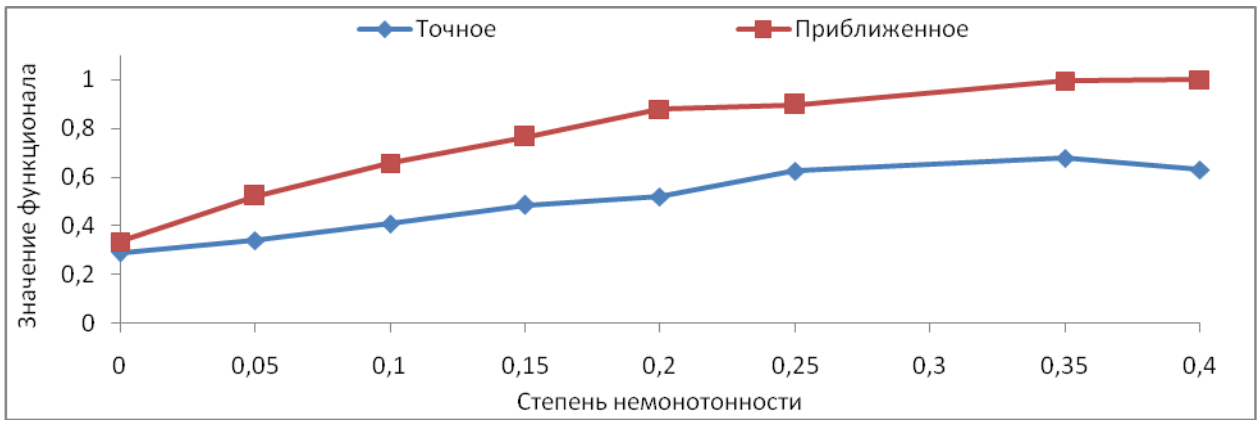
1.3. Численные эксперименты по расчету функционала обобщающей способности и его оценок.

Для сравнения значений приближенного функционала обобщающей способности, предложенного в [2] с улучшенным значением, предложенным в работе, рассматривалась задача с параметрами $L = 20$, число объектов принадлежащих классу -1 равно числу объектов, принадлежащих классу +1 и равно 10. Для каждого значения длины обучающей выборки $l = 1 \ldots 19$ рассматривались 20 случаев различного расположения объектов на плоскости со степенью немонотонности $\delta \in [0.5, 0.4]$. Каждая точка на графике является средним значением по этим 20 измерениям.



Из графика видно, что чем длиннее обучающая выборка, тем точнее становится приближенная оценка, а чем она короче – тем лучше работает предложенное улучшение этой оценки. Эти результаты согласуются с оценками (1.7) и (1.8), поскольку чем меньше длина обучающей выборки, тем больше вероятность, что случаи, которые в них описаны реализуются. С другой стороны, из графика видно, что при длине обучающей выборки большей 80% от общей длины выборки описанные эффекты перестают реализовываться и улучшенная оценка практически совпадает с приближенной. Внизу изображены графики поправок, вносимых случаями 1 и 3, описанных в разделе 2. Видно, что их вклад также не существенен, если длина обучающей выборки становится больше 80%.

На следующем графике показана зависимость функционалов от степени немонотонности выборки. Параметры эксперимента: $L = 20$ $l = 16$.



Таким образом, чем более монотонная выборка, тем точнее приближенное значение функционала (1.3). Поэтому, в качестве оценки функционала обобщающей способности в дальнейшем будет использоваться именно функционал (1.3).

Глава 2. Синтез монотонных алгоритмических композиций на основе анализа их обобщающей способности.

В данной главе рассматривается задача классификации объектов выборки X^L, где $X \subset R^k$ на 2 непересекающихся класса -1 и +1. Для решения данной задачи используется алгебраический подход, в котором в качестве базовых алгоритмов используются алгоритмы B_i, умеющие обучаться на объектах выборки X^L с весами и в качестве результатов классификации выдающие вероятность отнесения объекта к тому или иному классу с соответствующим знаком. В качестве корректирующей операции будут рассматриваться монотонные функции. Под положительным клином далее будем понимать прямой положительный клин.

2.1. Анализ оценки функционала обобщающей способности

Перепишем значение функционала (1.3) в следующем виде:

$$Q(\mu, X^L) = \sum_{m=0}^{\delta L+k-1} M(m, X^L) \sum_{s=\max\{0,m-k+1\}}^{\min\{\delta L,l,m\}} \frac{C_m^s C_{L-1-m}^{l-s}}{C_{L-1}^l} = \sum_{m=0}^{\delta L+k-1} M(m, X^L) C^{'}(m).$$

Назовем множитель $C^{'}(m)=\sum_{s=\max\{0,m-k+1\}}^{\min\{\delta L,l,m\}}\frac{C_m^s C_{L-1-m}^{l-s}}{C_{L-1}^l}$ комбинаторной функцией.

Теорема 1. Если $m\le\delta L\le k-1$, то $C^{'}(m)=1$. Если $m>\delta L$, то $C^{'}(m)<1$.

Доказательство. При указанных в условиях теоремы ограничениях выражение для комбинаторной функции будет иметь следующий вид: $C^{'}(m)=\sum_{s=0}^{m}\frac{C_m^s C_{L-1-m}^{l-s}}{C_{L-1}^l}$. Если множество из $L-1$ объекта разбить на 2 подмножества: m и $L-1-m$, то выражение в числителе будет в точности задавать число способов, которыми из первого подмножества выбираются s объектов, а из второго – $l-s$ объектов. Если просуммировать это выражение по s от 0 до m, то получим в точности число способов, которыми можно из всего множества $L-1$ объектов выбрать l объектов. Таким образом, $C^{'}(m)=\frac{C_{L-1}^l}{C_{L-1}^l}=1$. Очевидно, что C_{L-1}^l – максимальное значение, которое может принять числитель. Из этих соображений следует второе условие теоремы.

В дальнейшем будем предполагать, что исследуются задачи, у которых $l=0.8L, k=0.2L$ и $\delta<0.2$, то есть все условия теоремы 1 выполнены. Ограничение $\delta<0.2$ говорит о том, что рассматриваются достаточно монотонные выборки.

При указанных предположениях оценка функционала обобщающей способности запишется в следующем виде:

$$Q(\mu,X^L)=\sum_{m=0}^{\delta L}M(m,X^L)+\sum_{m=\delta L+1}^{\delta L+k-1}M(m,X^L)C^{'}(m).$$

Перейдем в этой формуле от суммированию по m к суммированию по объектам:

$$Q(\mu,X^L)=\sum_{i=0}^{L}C(w_i^+), \qquad (1.9)$$

где $C(w_i^+)=0$ при $w_i^+>\delta L+k-1$ и $C(w_i^+)=1$ при $w_i^+\le\delta L$.

На рисунке 6 изображен график функции $C(m)$ для случая $L=1000, l=800, \delta L=50$

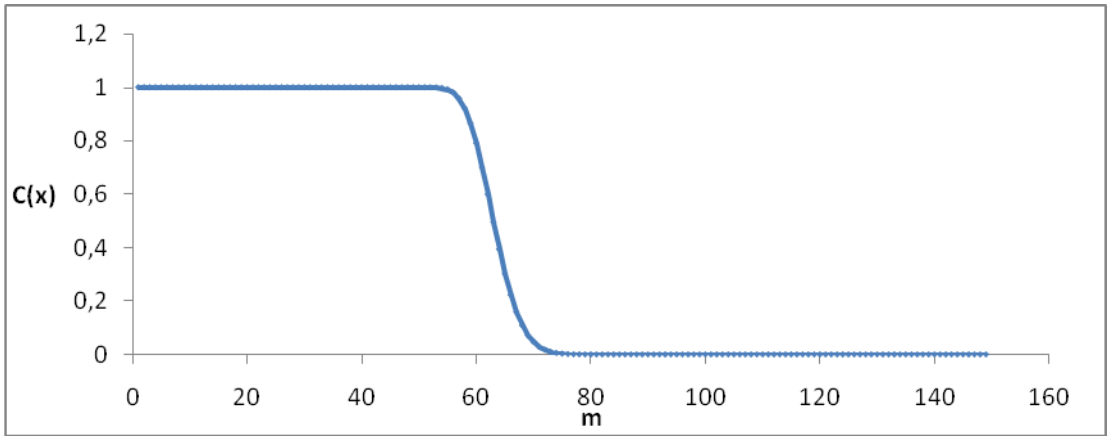


Рис. 6

На графике видно, что при $m \le \delta L = 50$ комбинаторная функция равна 1, а при дальнейшем увеличении m очень быстро спадает до нуля. Таким образом, чем менее монотонная будет выборка, тем больше объектов попадет в область $w_i^+ \le \delta L$ и тем больше будет значение функционала обобщающей способности.

2.2. Причины переобучения алгоритма построения монотонных композиций, полученного в [3].

Используя вид функции $C(w_i^+)$ в (1.9), покажем, как интерпретировать оценку функционала обобщающей способности. Для этого рассмотрим произвольную задачу, в которой для каждого из восьми объектов известен вес его клина w_i^+ и известна степень немонотонности δ всей выборки.

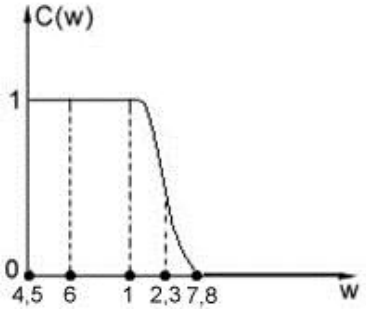


Рис. 7

На рисунке 7 объекты упорядочены по горизонтальной оси по значению мощности клина w_i^+. Для подсчета оценки функционала обобщающей способности необходимо спроецировать эти значения на график комбинаторной функции и усреднить соответствующие значения, принимаемые комбинаторной функцией.

Теперь выясним причины, приводившие к переобучению алгоритма строящего композиции с монотонной корректирующей операцией, полученного в [3]. Рассмотрим работу этого алгоритма на примере задачи, состоящей из четырех объектов.

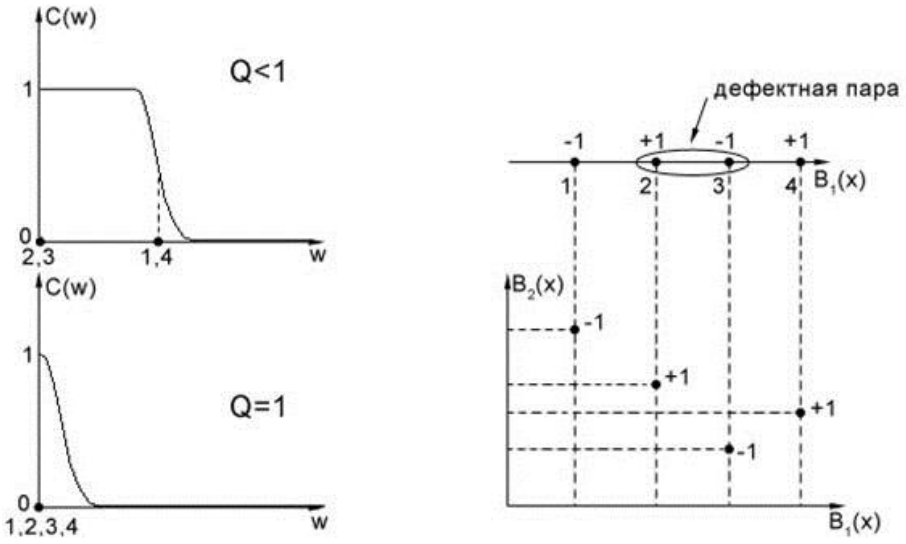


Рис. 8

На рисунке в верхнем левом углу изображен результат настройки первого базового алгоритма на этой выборке. После настройки появилась одна дефектная пара и степень немонотонности стала больше нуля. Вес объектов 1 и 4 равен 1, а вес объектов 2 и 3 равен 0. При подсчете функционала обобщающей способности $C(0)=1, C(1)<1 \Rightarrow Q<1$

Следующий базовый алгоритм должен настраиваться на выборке таким образом, чтобы минимизировать число дефектных пар. После его настройки дефектная пара (2,3) будет устранена, то есть эти объекты перейдут в разряд несравнимых. Однако при настройке этого базового алгоритма совершенно не учитывалось, что объекты 1,3 и 2,4 образовывали клинья, поэтому они также могли перейти в разряд несравнимых (см. рис). Степень немонотонности после такой настройки уменьшилась до нуля, но и веса клиньев всех объектов стали равны нулю. Значение функционала обобщающей способности стало равно 1, то есть увеличилось. Именно этот эффект объясняет процесс переобучения жадного алгоритма.

Таким образом, после настройки очередного базового алгоритма степень немонотонности всей выборки уменьшается за счет того, что устраняются дефектные пары, но при этом уменьшаются мощности клиньев объектов выборки, так как некоторые объекты становятся несравнимыми.

2.3. Принцип построения композиций алгоритмов на основе оптимизации обобщающей способности

Докажем несколько теорем, которые потребуются в дальнейшем для объяснения основного принципа построения композиций алгоритмов на основании оптимизации обобщающей способности.

Теорема 2. Пусть имеется выборка объектов X^L и для нее уже построено $T-1$ базовых алгоритмов $B_1,\ldots,B_{T-1}$. Тогда после обучения базового алгоритма B_T на выборке X^L степень немонотонности δ выборки $\{B_i(x_j)\}_{i=1..T}^{j=1..L}$ может только уменьшиться.

Доказательство. Пусть построено $T-1$ базовых алгоритмов $B_1,\ldots,B_{T-1}$, тогда для выборки $\{B_i(x_j)\}_{i=1,\ldots,T-1}^{j=1,\ldots,L}$ известна степень немонотонности δ_{T-1}. Это означает, что существует монотонная функция $F_{T-1}(B_1(x_i),\ldots,B_{T-1}(x_i))$, на которой достигается эта степень немонотонности. Допустим, что мы произвольным образом обучили базовый алгоритм B_T на выборке X^L и добавили его в композицию. Степень немонотонности новой выборки $\{B_i(x_j)\}_{i=1..T}^{j=1..L}$ стала равна δ_T и она достигается на некоторой функции $F_T(B_1(x_i),\ldots,B_T(x_i))$, минимизирующей число ошибок новой выборки. Построим функцию F_T следующим образом:

$$F_T(B_1(x_i),\ldots,B_T(x_i)) = F_{T-1}(B_1(x_i),\ldots,B_{T-1}(x_i)).$$

Очевидно, что построенная таким способом функция F_T является монотонной, поскольку F_{T-1} является монотонной и на ней достигается степень немонотонности δ_{T-1}. Однако, возможно, что функцию F_T можно построить более оптимальным способом с точки зрения уменьшения эмпирического риска. Поэтому $\delta_T \le \delta_{T-1}$.

Введем обозначения:

$C^T(m)$ — комбинаторная функция для выборки $\{B_i(x_j)\}_{i=1\ldots T}^{j=1\ldots L}$,

w_i^T — вес положительного клина объекта $\{B_j(x_i)\}_{j=1\ldots T}$ для выборки $\{B_i(x_j)\}_{i=1\ldots T}^{j=1\ldots L}$

K_i^T — множество индексов объектов, для которых объект $\{B_j(x_i)\}_{j=1\ldots T}$ входит в положительный клин.

$Q^T = \frac{1}{L}\sum_{i=1}^{L} C^T(w_i^T)$ — оценка функционала обобщающей способности для выборки $\{B_i(x_j)\}_{i=1\ldots T}^{j=1\ldots L}$

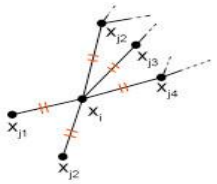
Теорема 3. $\forall i = 1\ldots L \quad C^{T+1}(w_i^{T+1}) \le C^T(w_i^{T+1})$

Доказательство. Воспользуемся тем, что график комбинаторной функции монотонно убывает и максимальное значение аргумента, при котором комбинаторная функция достигает значения 1 равно δL. В теореме 1 доказано, что при добавлении очередного базового алгоритма степень немонотонности δ может только уменьшаться, то есть график комбинаторной функции сдвинется влево. Следовательно $C^{T+1}(x) \le C^T(x) \quad \forall x \ge 0$.

Из теорем 2 и 3 следует справедливость следующей цепочки неравенств:

$$Q^{T+1} - Q^T = \frac{1}{L}\sum_{i=1}^{L}(C^{T+1}(w_i^{T+1}) - C^T(w_i^T)) \le \frac{1}{L}\sum_{i=1}^{L}(C^T(w_i^{T+1}) - C^T(w_i^T)) \le$$
$$\frac{1}{L}\sum_{i=1}^{L}\left(1 - C^T(w_i^T) + \sum_{j\in K_i^T} C^T(w_j^T - 1) - C^T(w_j^T)\right) \qquad (1.10)$$

Каждое слагаемое в сумме описывает вклад объекта i в оценку функционала обобщающей способности, то есть насколько бы она изменилась, если бы все связи в клиньях, в которых участвует данный объект, были бы разрушены.



Из теоремы 2 следует, что при построении следующего базового алгоритма степень немонотонности выборки может только убывать. Поэтому композиция будет становиться все более корректной при добавлении новый базовых алгоритмов. Для того чтобы не происходило переобучения важно следить за тем, чтобы мощности клиньев объектов не сильно уменьшались после добавления очередного базового алгоритма. Для этого необходимо обучаться сильнее на тех объектах, которые своими связями в клиньях сильнее других уменьшают функционал. Веса объектов, с которыми необходимо обучаться очередному базовому алгоритму, получены в (1.10) и равны:

$$W_i = 1 - C^T(w_i^T) + \sum_{j\in K_i^T} C^T(w_j^T - 1) - C^T(w_j^T). \qquad (1.11)$$

2.4. Алгоритм построения алгоритмических композиций на основе оптимизации функционала обобщающей способности

Ниже приведен эффективный алгоритм построения алгоритмических композиций на основе оптимизации функционала обобщающей способности использующий формулу (1.11). В нем используются 2 матрицы, содержащие литералы в качестве элементов для ускорения его работы. Эти матрицы показывают, в каком соотношении находятся 2 объекта: равны; первый больше второго; первый меньше второго; несравнимы. Поэтому будем называть их матрицами соотношений (RelMat).

RelMat^0 – матрица соотношений между объектами класса -1,
I^0 – множество индексов объектов класса -1,
RelMat^1 – матрица соотношений между объектами класса +1,
I^1 – множество индексов объектов класса +1.

После построения первого базового алгоритма матрица соотношений объектов класса -1 инициализируется следующим образом:

$$\mathrm{RelMat}_1^0 = \left\| a_{ij}^1 \right\|_{i,j=1..|I^0|}, \text{ где } a_{ij}^1 = \begin{cases} \textit{меньше, если } B_1(x_i) < B_1(x_j) \\ \textit{больше, если } B_1(x_i) > B_1(x_j) \\ \textit{равно, если } B_1(x_i) = B_1(x_j) \end{cases}.$$

Аналогичным образом инициализируется матрица соотношений объектов класса +1.

Пусть построено $n-1$ базовых алгоритмов $B_1,\ldots,B_{n-1}$ и для выборки объектов класса -1 $\{B_i(x_j)\}_{i=1..n-1}^{j\in I^0}$ известна матрица соотношений RelMat_{n-1}^0. Тогда после добавления в композицию алгоритма B_n матрица соотношений RelMat_n^0 будет пересчитываться следующим способом:

$$\mathrm{RelMat}_n^0 = \left\| a_{ij}^n \right\|_{i,j=1..|I^0|}, \text{ где } a_{ij}^n = \begin{cases} \text{'}\textit{меньше}\text{'}, \text{если } a_{ij}^{n-1} = \text{'}\textit{равно}\text{'} \text{ и } B_n(x_i) < B_n(x_j) \\ \text{'}\textit{больше}\text{'}, \text{если } a_{ij}^{n-1} = \text{'}\textit{равно}\text{'} \text{ и } B_n(x_i) > B_n(x_j) \\ \text{'}\textit{несравним}\text{'}, \text{если } a_{ij}^{n-1} = \text{'}\textit{больше}\text{'} \text{ и } B_n(x_i) < B_n(x_j) \\ \text{'}\textit{несравним}\text{'}, \text{если } a_{ij}^{n-1} = \text{'}\textit{меньше}\text{'} \text{ и } B_n(x_i) > B_n(x_j) \\ a_{ij}^{n-1}, \text{ в остальных случаях} \end{cases}.$$

Аналогичным образом будет пересчитываться матрица объектов класса +1.

Алгоритм: Построение базовых алгоритмов и монотонной корректирующей операции над ними.

Вход:

$\{x_i, y_i\}_{i=1}^{\ell}$ – обучающая выборка;

$\mathfrak{M}^0$ – параметрическое семейство базовых алгоритмов

Выход:

Настроенные базовые алгоритмы $B_1,\ldots,B_n$ и построенная корректирующая операция F.

1: На основе обучающей выборки $\{x_i, y_i\}_{i=1}^{\ell}$ формируем множества I^0 и I^1. Инициализировать текущее число базовых алгоритмов $n=1$ и веса объектов обучения $W_i=1\ \forall i=1\ldots\ell$. Инициализируем степень немонотонности $\delta=+\infty$.

2: Повторять, пока $\delta>0$ или если в течение последних 5 итераций значение функционала обобщающей способности не убывало. (5 итераций обязательно).

3: Настроить алгоритмический оператор B_n на исходных прецедентах с учетом весов объектов W_i и вычислить значения $B_n(x_i)\ \forall i=1\ldots\ell$.

4.1:Если $n=1$:

Для обоих классов инициализировать матрицы соотношений RelMat_1^0 и RelMat_1^1.

4.2:Иначе:

Пересчитать значения матриц соотношений RelMat_n^0 и RelMat_n^1, используя матрицы RelMat_{n-1}^0 и RelMat_{n-1}^1.

5: Вычислить степень немонотонности δ полученной выборки, используя алгоритм 3.1, описанный в разделе 3.

6: Учитывая, что доля объектов обучения составляет 80% и, используя полученное на предыдущем шаге значение δ, рассчитать значение комбинаторной функции $C^n(m)\ \forall m<\delta\ell+0.2\ell-1$.

7: Используя значения комбинаторной функции и матрицы соотношений рассчитать значение оценки функционала обобщающей способности:

$$Q^n=\frac{1}{L}\sum_{i=1}^{L}C^n(w_i{}^n)$$

8: Пересчитать веса объектов обучения:

$$W_i=1-C^n(w_i^n)+\sum_{j\in K_i^n}C^n(w_j^n-1)-C^n(w_j^n),$$

где множества K_i^n определяются следующим образом:

$$K_i^n=\begin{cases} j: a_{ij}^n=\text{'}\textit{больше}\text{'}, \text{если } i\in I^0, \text{где } a_{ij}^n-\text{элемент } \mathrm{RelMat}_n^0 \\ j: a_{ij}^n=\text{'}\textit{меньше}\text{'}, \text{если } i\in I^1, \text{где } a_{ij}^n-\text{элемент } \mathrm{RelMat}_n^1 \end{cases}.$$

10: Оставить первые n базовых алгоритмов, с использованием которых оценка функционала обобщающей способности была минимальна.

11: Над выборкой $\{B_i(x_j)\}_{i=1\ldots n}^{j=1\ldots \ell}$ построить монотонную корректирующую операцию, описанную в 3.1 раздела 3.

Описанный алгоритм обладает двумя очень важными достоинствами:

- Отсутствие переобучения.
- Автоматический выбор оптимального числа базовых алгоритмов в композиции.

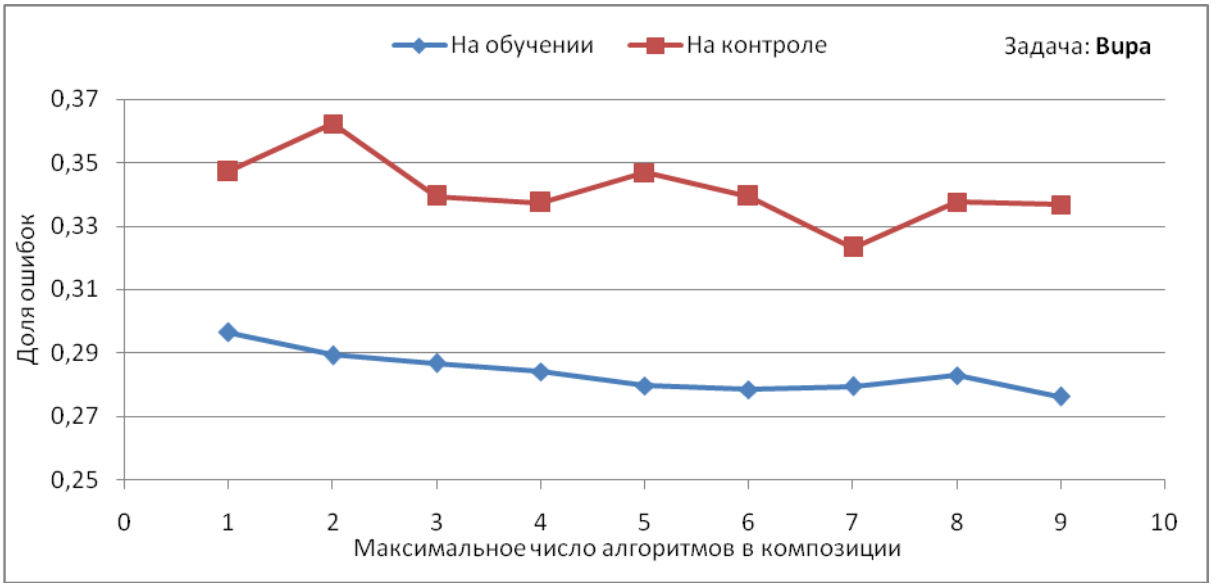
2.5. Результаты экспериментов

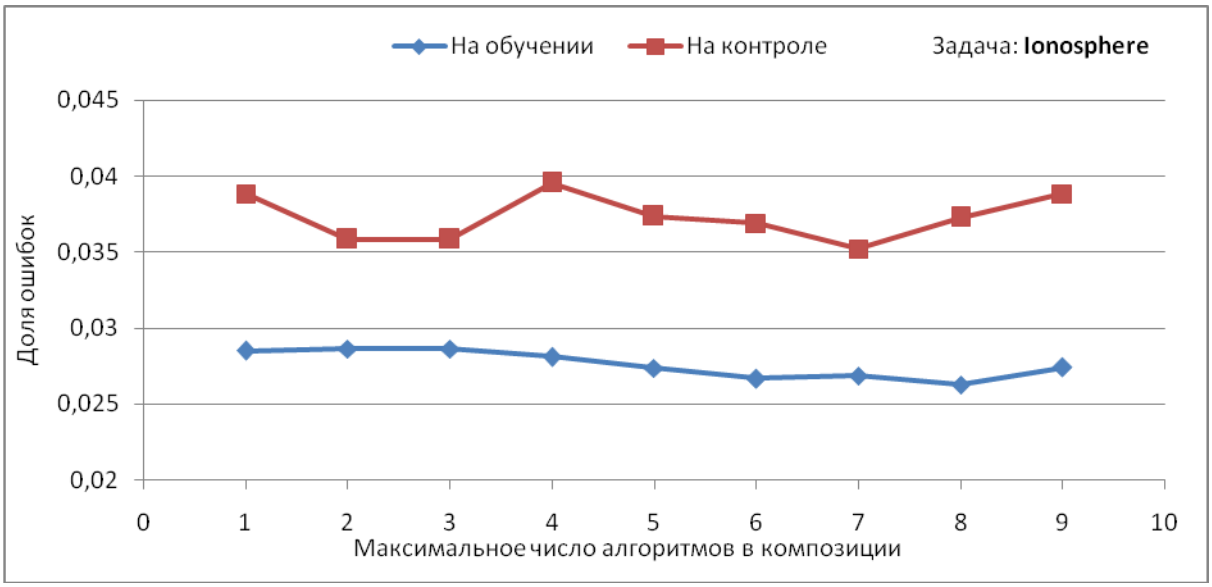
В качестве семейства базовых алгоритмов были использованы алгоритмы SVM (Support Vector Machine), строящие разделяющие гиперповерхности и умеющие обучаться на объектах с весами.

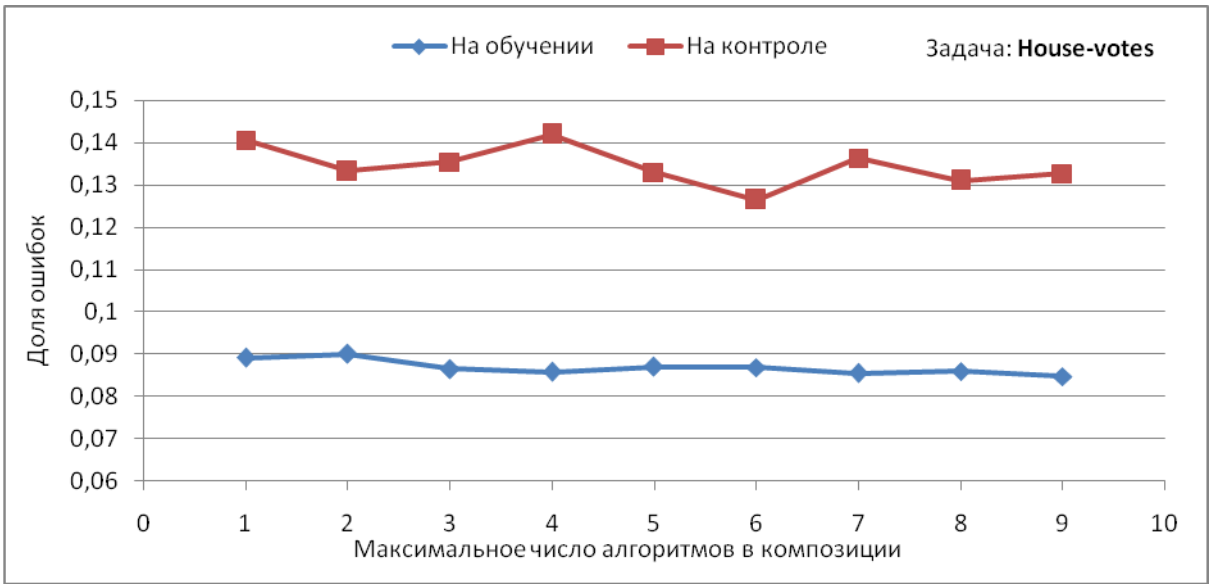
В качестве задач, на которых проводились численные эксперименты, были рассмотрены 3 задачи из репозитария UCI [5].

1. **Ionosphere** – Задача распознавания взаимодействия сигнала радиотелескопа с ионосферой: 34 признака (все вещественные). Всего 351 объект, из них 316 объектов для обучения, 35 для контроля.
2. **Bupa** – Задача диагностики нарушений работы печени: 6 признаков (все вещественные). Всего 345 объектов, из них 310 объектов для обучения, 35 для контроля.
3. **House-votes** – Восстановление принадлежности депутатов Конгресса США к партии (демократы или республиканцы) по их голосованию: 16 признаков, все бинарные. Всего 435 объектов, из них 391 объектов для обучения, 44 для контроля.

Каждая точка на графиках является усреднением по 50 различным способам разбиения объектов задач на обучающую и контрольную подвыборки.







Из графиков видно, что переобучения действительно не происходит.

Оптимальное число алгоритмов в композиции для всех трех задач равно 3.

ЗАКЛЮЧЕНИЕ

В работе получены следующие результаты:

1. Улучшена оценка функционала обобщающей способности для семейства монотонных классификаторов.
2. Показаны границы применимости существующей оценки функционала обобщающей способности для семейства монотонных классификаторов.
3. Объяснена причина переобучения алгоритма построения алгоритмических композиций, минимизирующего число дефектных пар.
4. Разработан и теоретически обоснован алгоритм построения композиций алгоритмов с монотонной корректирующей операцией, минимизирующий оценку функционала обобщающей способности.

5. Разработан вычислительно эффективный способ реализации предложенного алгоритма и с помощью него проведены численные эксперименты на трех реальных задачах.

В качестве направления дальнейших исследований можно указать:

1. Получение теоретических оценок для оптимального числа базовых алгоритмов в композиции.
2. Оценка функционала обобщающей способности принципиально другим способом.
3. Разработка адаптивного алгоритма, который на каждом этапе построения очередного базового алгоритма решал бы, каким способом его оптимальнее настраивать: минимизируя число дефектных пар или минимизируя функционал обобщающей способности.
4. Сравнение алгоритмических композиций с монотонными корректирующими операциями над базовыми алгоритмами SVM с другими типами алгоритмов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Воронцов К. В. Локальные базисы в алгебраическом подходе к проблеме распознавания — Диссертация на соискание ученой степени к.ф.-м.н., М.: ВЦ РАН — 1999.
2. Воронцов К. В. Комбинаторный подход к оценке качества обучаемых алгоритмов // Математические вопросы кибернетики. — 2004. — No. 13. — С. 5–36.
3. Гуз И.С. Нелинейные монотонные композиции классификаторов //ММРО-13 — 2007. — С. 111–114.
4. Blake C., Merz C. UCI repository of machine learning Department of Information and Computer Science, University CA, 1998.
   http://www.ics.uci.edu/~mlearn/MLRepository.html.