

Комитетный бустинг: минимизация числа базовых алгоритмов при простом голосовании

Маценов А. А.

amascenov@yandex.ru

Москва, МФТИ

Построение композиций алгоритмов является универсальным способом повышения качества классификации в сложных прикладных задачах. Вполне естественно требовать, чтобы композиция была не избыточной, т. е. состояла из минимального числа базовых алгоритмов.

Пусть X — множество допустимых описаний объектов, $Y = \{-1, 1\}$ — множество меток классов, $X^\ell = (x_i, y_i)_{i=1}^\ell \subset X \times Y$ — обучающая выборка, $b_t: X \rightarrow Y$ — базовые алгоритмы, $t = 1, \dots, T$. *Комитетом большинства* [1] называется алгоритмическая композиция вида

$$a(x) = \text{sign}(b_1(x) + \dots + b_T(x)), \quad (1)$$

при условии, что она корректно классифицирует объекты обучающей выборки: $a(x_i) = y_i$. Комитет, состоящий из минимального возможного числа алгоритмов T , называется *минимальным*. В данной работе понятие комитета (1) обобщается сразу по нескольким направлениям.

Рассматриваются базовые алгоритмы $b_t: X \rightarrow \mathbb{R}$, которые могут выдавать как ± 1 , так и числовую оценку принадлежности объекта классу $+1$. Для многих приложений гарантировать корректность не обязательно (в таких случаях комитет называется *грубым* [1]). Гораздо важнее, чтобы композиция обладала хорошей обобщающей способностью, т. е. допускала как можно меньше ошибок на независимых контрольных данных. Минимальность T также не является жестким требованием; достаточно обеспечить значение T , близкое к минимальному. Но тогда можно отказаться от решения NP-трудной задачи построения минимального комитета [2] и применять разного рода эвристики для построения «почти минимального» грубого комитета. Такими эвристиками, по сути дела, являются бустинг [3] и бэггинг [4]. Однако в них задача минимизации T не ставится даже приближённо, что приводит к чрезмерно сложным композициям из десятков и сотен базовых алгоритмов.

Метод ComBoost

Степенью граничности или *отступом* (margin) объекта x_i называется величина $m_T(x_i) = y_i \sum_{t=1}^T b_t(x_i)$. Отступ отрицателен тогда и только тогда, когда $a(x)$ допускает ошибку на объекте x_i . Поэтому число ошибок композиции $a(x)$ на обучающей выборке X^ℓ определяется как

$$Q(a, X^\ell) = \sum_{i=1}^{\ell} [m_T(x_i) < 0].$$

Известно, что оптимальным с точки зрения понижения вероятности ошибки является такое распределение отступов, при котором все они принимают одинаковое и как можно большее значение [5]. В то же время, в реальных задачах всегда встречаются объекты-выбросы, которые плохо описываются как базовыми алгоритмами, так и их композицией. Ошибки на таких объектах неизбежны, а их отступы, соответственно, отрицательны. При этом заранее неизвестно, какие именно объекты являются выбросами. Изложенные соображения приводят к новому методу построения композиций, основанному на идентификации выбросов.

Базовый алгоритм b_1 строится по выборке X^ℓ некоторым стандартным методом. Рассмотрим t -й шаг, когда базовые алгоритмы $b_1, \dots, b_{t-1}$ уже есть, и требуется построить алгоритм b_t . Упорядочим выборку X^ℓ по возрастанию отступов $m_{t-1}(x_i)$. Объекты с одинаковыми отступами договоримся располагать в случайном порядке. Обучим алгоритм b_t по подвыборке длины $(\ell_2 - \ell_1)$, с $\ell_1 + 1$ -го по ℓ_2 -й объекты включительно. Параметр ℓ_1 ограничивает количество выбросов; параметр ℓ_2 позволяет найти компромисс между качеством и различностью базовых алгоритмов.

Значения параметров ℓ_1, ℓ_2 могут либо фиксироваться, либо подбираться на каждом шаге t по критерию минимума числа ошибок композиции на всей выборке X^ℓ . В экспериментах второй вариант практически всегда давал лучший результат. Оптимизация ℓ_1, ℓ_2 может быть очень эффективной, если существует метод быстрой перенастройки базового алгоритма при добавлении одного объекта в обучающую выборку.

По мере увеличения числа алгоритмов в композиции повышается надёжность идентификации выбросов. При этом алгоритмы, которые были обучены ранее, оказываются неоптимальными, так как их обучающие выборки содержали выбросы. Поэтому, начиная с некоторого шага t_0 , перед обучением нового алгоритма «наиболее старый» удаляется.

Описанный алгоритм был назван ComBoost — *комитетный бустинг*, за очевидное сходство и с бустингом, и с методом комитетов.

Эксперименты и результаты

Исследование качества предложенного метода проводилось на реальных данных из репозитория UCI (в скобках указана длина выборки и число признаков): *Ionosphere* (351×34), *Bupa* (345×6), *House-votes* (435×16), *Pima-diabets* (768×8).

В качестве базовых алгоритмов использовались: (а) SVM с линейным ядром; (б) непараметрический байесовский классификатор с локальным оцениванием плотности по Парзену-Розенблатту, евклидовой метрикой на X и подбором ширины окна по скользящему контролю с одним отделяемым объектом (leave-one-out).

Ошибка %	ionosphere	pima	bupa	votes
SVM	12,9	24,2	42	4,6
ComBoost ₀ [SVM]	12,6	23,1	34,2	4
ComBoost [SVM]	12,3	22,5	30,9	3,8
AdaBoost [SVM]	15	22,7	30,6	4
Parsen	6,3	25,1	41,6	6,9
ComBoost ₀ [Parsen]	6,1	25	38,1	6,8
ComBoost [Parsen]	5,8	24,7	30,6	6,2
AdaBoost [Parsen]	6	24,8	30,5	6,5

Таблица 1. Средняя частота ошибок на контроле по 50 случайным разбиениям в отношении «обучение : контроль» = 4 : 1.

Число базовых алгоритмов	ionosphere	pima	bupa	votes
ComBoost ₀ [SVM]	4	2	5	2
ComBoost [SVM]	5	2	5	3
AdaBoost [SVM]	65	18	15	8
Оценка q	11	87	51	27

Таблица 2. Мощность композиций. Для сравнения приводится классическая верхняя оценка числа членов минимального комитета линейных алгоритмов $q = 2 \left\lceil \frac{1}{n} \left\lfloor \frac{\ell-n}{2} \right\rfloor \right\rceil + 1$, где n — число признаков [6].

Сравнивались три метода построения композиций: ComBoost₀ — фиксация параметров ℓ_1 , ℓ_2 , подобранных для каждой задачи по критерию скользящего контроля; ComBoost — оптимизация параметров ℓ_1 , ℓ_2 на каждом шаге t ; AdaBoost — алгоритм бустинга [3]. Для всех трёх методов использовался один и тот же критерий останова — отсутствие существенного улучшения качества классификации обучающей выборки.

Результаты сравнения представлены в таблицах 1, 2.

Особенности ComBoost

ComBoost не накладывает ограничений ни на вид базовых алгоритмов, ни на метод их обучения. Базовые алгоритмы могут быть как бинарными (возвращать ± 1), так и вещественнозначными (возвращать числовую оценку). В отличие от бустинга, не требуется, чтобы метод обучения был чувствителен к весам объектов.

ComBoost демонстрирует достаточно высокую обобщающую способность даже для устойчивых базовых алгоритмов, коррекцию которых принято считать малоэффективной, в частности, для SVM.

В отличие от бустинга и бэггинга, ComBoost строит короткие композиции за счёт удаления наименее удачных алгоритмов из композиции.

Автоматически обнаруживаются объекты-выбросы. Обучающие объекты ранжируются по отступам, что позволяет выделить среди них шумовые, пограничные, типичные и эталонные объекты.

ComBoost легко обобщается на случай $|Y| > 2$, если воспользоваться понятием многоклассового геометрического отступа [7].

Работа выполнена при поддержке РФФИ, проект № 05-01-00877, и программы ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики».

Литература

- [1] *Мазуров В. Д.* Метод комитетов в задачах оптимизации и классификации. — М.: Наука, 1990.
- [2] *Хачай М. Ю.* О вычислительной сложности задачи о минимальном комитете и смежных задач // Доклады РАН. — 2006. — Т. 406, № 6. — С. 742-745.
- [3] *Freund Y., Schapire R. E.* Experiments with a new boosting algorithm // International Conference on Machine Learning. — 1996. — Pp. 148-156.
- [4] *Breiman L.* Bagging predictors // Machine Learning. — 1996. — Vol. 24, No. 2. — Pp. 123-140.
- [5] *Mason L., Bartlett P., Baxter J.* Direct optimization of margins improves generalization in combined classifiers // Proc. of the 1998 conf. on Advances in Neural Information Processing Systems II. — MIT Press, 1999. — Pp. 288-294.
- [6] *Hachai M. Y., Rybin A. I.* A new estimate of the number of members in a minimum committee of a linear inequalities system // Pattern Recognition and Image Analysis. — 1998. — Vol. 8, No. 4. — Pp. 491-496.
- [7] *Allwein E. L., Schapire R. E., Singer Y.* Reducing multiclass to binary: A unifying approach for margin classifiers // 17th Int'l Conf. on Machine Learning. — Morgan Kaufmann, San Francisco, CA, 2000. — Pp. 9-16.