Профили компактности и выделение опорных объектов в метрических алгоритмах классификации

К. В. Воронцов, А. О. Колосков
(Москва)

Известно много методов классификации, в которых алгоритм строится по специально отобранному подмножеству обучающих объектов. Классический пример такого метода — машины опорных векторов, SVM [1]. Опорными векторами в SVM оказываются объекты, примыкающие к границе классов. Данный принцип отбора опорных объектов неоднократно подвергался критике, поскольку граничными объектами в условиях неполных и неточных данных часто становятся шумовые выбросы. Идея брать в качестве опорных объекты, «слегка отодвинутые» от границы классов, лежит в основе метода релевантных векторов, RVM [2]. Это приводит к построению более гладкой разделяющей поверхности и повышению качества классификации.

Для метрических алгоритмов классификации, таких как алгоритм ближайших соседей, метод потенциальных функций или RBF-сети, выделение опорных объектов позволяет существенно уменьшить объёмы хранимых данных, повысить скорость и качество классификации.

В метрических алгоритмах СТОЛП и λ-СТОЛП на этапе обучения применяется «жадная» стратегия последовательного добавления опорных объектов [3]. На каждом шаге алгоритма к множеству опорных объектов присоединяется «наиболее напряжённый» объект, в окрестности которого плотность чужих классов максимальна. В результате опорными становятся пограничные объекты, аналогично тому, как это происходит в SVM. Для реализации метрического алгоритма с более разумным механизмом отбора опорных объектов, подобным RVM, необходим более тонкий критерий, способный обоснованно решать, насколько далеко от границы классов должны отстоять опорные объекты.

В данной работе предлагается критерий, основанный на понятии профиля компактности и комбинаторных формулах эффективного вычисления оценок скользящего контроля [4].

Пусть задана выборка $X^L = \{x_1, \ldots, x_L\}$. Обозначим через (X_n^l, X_n^k), $n = 1, \ldots, N$, $N = C_L^l$, всевозможные разбиения выборки X^L на обучающую и контрольную подвыборки длиной l и k соответственно. Обозначим через $\nu(X_n^l, X_n^k)$ частоту ошибок алгоритма, построенного по обучающей выборке X_n^l, на контрольной выборке X_n^k. Введём функционал полного скользящего контроля (complete cross-validation), характеризующий обобщающую способность метода обучения:

$$\mathrm{CCV} = \frac{1}{N} \sum_{n=1}^{N} \nu(X_n^l, X_n^k) .$$

Определение. *Профилем компактности* выборки X^L называется функция $P(i)$, $i = 1,\ldots,L-1$, выражающая долю объектов выборки, для которых правильный ответ не совпадает с правильным ответом на i-м соседе, если соседей нумеровать в порядке возрастания расстояний.

Теорема [4]. В методе ближайшего соседа функционал полного скользящего контроля может быть эффективно вычислен по формуле

$$\mathrm{CCV} = \sum_{i=1}^{k} P(i)\Gamma(i), \text{ где } \Gamma(i) = C_{L-1-i}^{l-1} \Big/ C_{L-1}^{l}.$$

Комбинаторный множитель $\Gamma(i)$ быстро убывает с ростом i. Поэтому для минимизации функционала CCV достаточно, чтобы при малых i профиль $P(i)$ принимал значения, близкие к нулю. Но это и означает, что близкие объекты должны лежать преимущественно в одном классе. Таким образом, профиль $P(i)$ является формальным выражением гипотезы компактности. На Рис. 1 показаны профили компактности для трёх плоских модельных задач, $L = 100$. Чем ниже проходит начальный участок профиля, тем выше обобщающая способность метода ближайшего соседа.

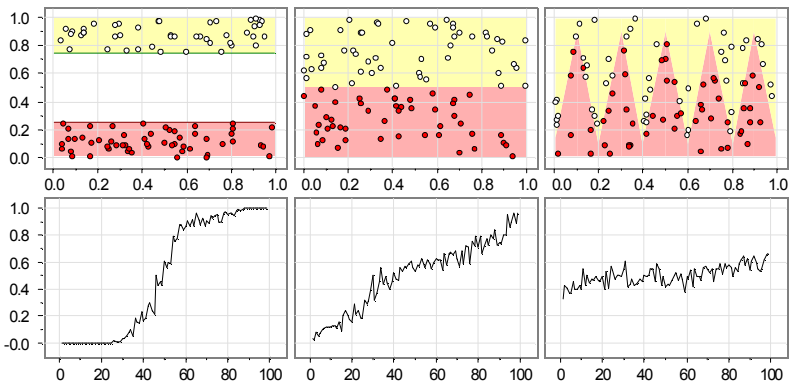

Рис 1. Профили компактности для трёх модельных задач.

Процедура оптимизации профиля компактности была положена в основу нового алгоритма выделения опорных объектов. В отличие от СТОЛП, данный алгоритм работает в противоположном направлении — начиная с полной выборки, он последовательно исключает объекты. На каждом шаге выбирается тот объект, исключение которого минимизирует CCV. Оказалось, что процесс отсева объектов разбивается на две стадии. Сначала исключаются шумовые выбросы, что приводит к уменьшению CCV. Затем исключаются неинформативные периферийные объекты, при этом значение функционала не изменяется или несущественно увеличивается. Процесс останавливается, когда остаются объекты, исключение которых заметно

увеличивает CCV. Таким образом, возникает естественное деление обучающих объектов на шумовые, периферийные и опорные.

На Рис. 2. показана зависимость CCV от числа исключённых объектов для модельной задачи, соответствующей среднему графику Рис. 1. На обучающую выборку накладывался дополнительный шум путём инвертирования ответов для 10% объектов. Для измерения обобщающей способности алгоритма вычислялась частота ошибок на независимой контрольной выборке, не содержавшей шума (тонкая линия на графике). Оба графика практически синхронно проходят участки понижения, постоянства и повышения. Этот факт позволяет утверждать, что оптимизация множества опорных объектов по функционалу CCV не ведёт к переобучению.

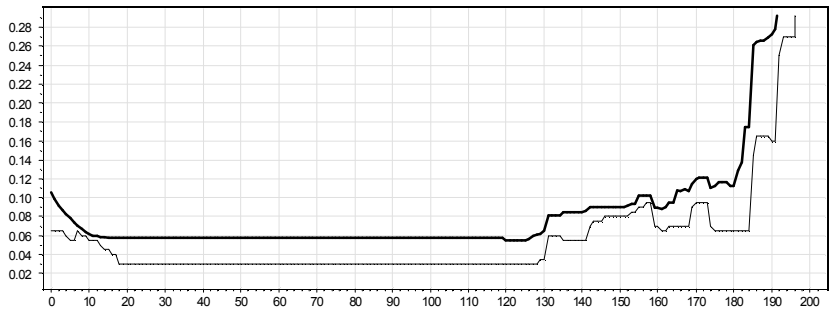

Рис 2. Изменение CCV в процессе последовательного исключения не-опорных объектов (эксперимент на модельных данных).

Работа выполнена в рамках проектов РФФИ 05-01-00877, 05-07-90410 и программы ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики».

Литература

1. Vapnik V. Statistical Learning Theory. — Wiley, New York, 1998.
2. Tipping M. The relevance vector machine // Advances in Neural Information Processing Systems, San Mateo, CA. — Morgan Kaufmann, 2000.
3. Загоруйко Н. Г. Прикладные методы анализа данных и знаний. — Новосибирск: ИМ СО РАН, 1999.
4. Воронцов К. В. Комбинаторный подход к оценке качества обучаемых алгоритмов // Математические вопросы кибернетики. — 2004. — No. 13. — С. 5–36.