Точные оценки вероятности переобучения

Задачи по спецкурсу «Теория надёжности обучения по прецедентам»

К. В. Воронцов (www.ccas.ru/voron)

18 марта 2009 г.

Точные оценки вероятности переобучения — это новое направление исследований в теории статистического обучения, в котором доказанных теорем (пока) гораздо меньше, чем открытых проблем, а многие нерешённые задачи настолько просты, что вполне по силам студентам 3 курса. Чтобы понимать, о чём эти задачи, необходимо (и достаточно) проработать статьи [1, 2] или ходить на спецкурс[1].

О структуре данного документа. Все задачи разделены на секции по направлениям исследований, каждая секция разделена на параграфы. Обозначения и предположения, вводимые в начале каждого параграфа, распространяются на все задачи внутри данного параграфа. Перед каждым блоком задач даётся минимум необходимых пояснений, включая мотивации, которые привели к данной постановке.

О формальностях. Для сдачи спецкурса необходимо набрать определённое количество баллов, которое будет объявлено на лекциях. Стоимость каждой задачи в баллах указана в скобках после номера задачи.

Страница курса: вики-ресурс www.MachineLearning.ru/wiki, страница «Теория надёжности обучения по прецедентам (курс лекций, К.В.Воронцов)». Адрес этого документа: www.MachineLearning.ru/wiki/images/6/6b/Voron09problems.pdf.

[1] Весной 2009: ВМиК МГУ, ауд. 606, по понедельникам 16:20.

Содержание

1 Классические оценки в слабой вероятностной аксиоматике

Задачи этой секции направлены на то, чтобы потренироваться в доказательстве оценок вероятности больших уклонений или вероятности переобучения, пользуясь только слабой вероятностной аксиоматикой.

Напоминание. *Слабая вероятностная аксиоматика* содержит единственную аксиому: дана конечная выборка $\mathbb{X} = \{x_1, \ldots, x_L\}$, и все её разбиения $X \cup \bar{X} = \mathbb{X}$ на наблюдаемую (обучающую) выборку X длины ℓ и скрытую (контрольную) выборку $\bar{X}$ длины k равновероятны.

§1.1 Закон больших чисел

Для любого алгоритма a, допускающего $m = n(a, \mathbb{X})$ ошибок на полной выборке, справедлива точная оценка вероятности большого уклонения частоты ошибок [3]:

$$\mathsf{P}\bigl[\delta(a, X) \geqslant \varepsilon\bigr] = \sum_{s=s_0}^{s_1(\varepsilon)} h_L^{\ell,m}(s) \equiv H_L^{\ell,m}(s_1(\varepsilon)) . \tag{1.1}$$

Это выражение можно рассматривать как оценку скорости сходимости в законе больших чисел для слабой аксиоматики, если положить $\ell, k \to \infty$.

Задача 1.1. (5) *Провести обзор известных верхних оценок «левого хвоста» гипергеометрического распределения. Какие из оценок наиболее точны? Сравнить с оценкой из книг [4, стр. 236] или [5, стр. 225]. Какие из оценок являются аналогами неравенств Чернова, Бернштейна, Хёффдинга [6]? Показать, что из этих оценок следует стремление левой части (1.1) к нулю при $\ell, k \to \infty$.*

§1.2 Некоторые статистические критерии

Многие статистические критерии, предназначенные для проверки гипотез однородности, независимости, случайности, стационарности (критерии Смирнова, Вилкоксона, ранговые и перестановочные критерии и т. д.), выводятся именно комбинаторными методами. Их (пере)формулировка и (пере)доказательство в слабой аксиоматике практически очевидны. Задачи данного параграфа не предполагают перехода к асимптотическим оценкам, как это обычно принято в математической статистике.

Задача 1.2. (1) *Критерий Вилкоксона-Манна-Уитни.*

Задача 1.3. (1) *Критерий серий Вальда-Вольфовица.*

Задача 1.4. (1) *Критерий знаков.*

§1.3 Оценки вероятности переобучения

Следующие две задачи заключаются в том, чтобы аккуратно воспроизвести в слабой аксиоматике результаты Бакса и Силла, которые оценивали другой функционал (функционал равномерной сходимости). Позволяет ли слабая аксиоматика упростить доказательства? Улучшить оценку?

Задача 1.5. (3) *Следуя [7], показать, что если множество векторов ошибок* $\bigl\{(a)_{\mathbb{X}}\colon a \in \mathbb{A}\bigr\}$ *кластеризуется по расстоянию Хэмминга на* $S(r)$ *кластеров радиуса* r *каждый, то*

$$\mathsf{P}\bigl[\delta_\mu(X) \geqslant \varepsilon + \tfrac{r}{\ell}\bigr] \leqslant S(r) \cdot \max_{m=1,\dots,L} H_L^{\ell,m}\left(s_1(\varepsilon)\right).$$

Задача 1.6. (3) *Следуя [8, 9] показать, что если семейство* A *связно, то*

$$\mathsf{P}\bigl[\delta_\mu(X) \geqslant \varepsilon\bigr] \leqslant \frac{1}{\sqrt{\pi L}}\,|A| \max_{m=1,\dots,L} H_L^{\ell,m}\left(s_1(\varepsilon)\right).$$

2 Исследования свойства связности

Достаточно просто должны доказываться утверждения о том, что при непрерывном изменении вектора параметров вдоль непрерывной траектории почти всегда образуется цепочка векторов ошибок. Что значит «почти», надо уточнять.

§2.1 Цепочки, порождаемые линейными классификаторами

Пусть $\mathbb{X}$ — множество, состоящее из L точек в $\mathbb{R}^n$; $\mathbb{Y} = \{-1, +1\}$ — множество меток классов; $\mathbb{A}$ — семейство n-мерных линейных классификаторов:

$$\mathbb{A} = \bigl\{a\colon \mathbb{X} \to \mathbb{Y} \bigm| a(x) = \operatorname{sign}(x_1 w_1 + \dots + x_n w_n),\ w \in \mathbb{R}^n\bigr\}, \tag{2.1}$$

где $w \in \mathbb{R}^n$ — вектор параметров. Предполагая, что существует функция истинной классификации $y\colon \mathbb{X} \to \mathbb{Y}$, введём бинарную функцию потерь $I(a, x) = \bigl[a(x) \neq y(x)\bigr]$.

Задача 2.1. (1) *Доказать, что множество векторов ошибок* $\bigl\{a(x, w + t\delta)\colon t \in \mathbb{R}\bigr\}$ *при некоторых дополнительных ограничениях образует цепочку. Сформулировать эти ограничения. Какова максимальная возможная длина цепочки?*

Говорят, что множество объектов $\mathbb{X}$ *линейно разделимо*, если существует направляющий вектор разделяющей гиперплоскости $w^* \in \mathbb{R}^n$, при котором алгоритм $a(x, w^*)$ не допускает ошибок на $\mathbb{X}$.

Задача 2.2. (1) *В каких случаях множество векторов ошибок* $\bigl\{a(x, w^* + t\delta)\colon t \geqslant 0\bigr\}$, *где* $\delta \in \mathbb{R}^n$ *— фиксированный вектор, образует монотонную цепочку с* $m = 0$?

Задача 2.3. (1) *В каких случаях множество векторов ошибок* $\bigl\{a(x, w^* + t\delta)\colon t \in \mathbb{R}\bigr\}$ *где* $\delta \in \mathbb{R}^n$ *— фиксированный вектор, образует унимодальную цепочку с* $m = 0$?

Пусть $\mathbb{A}$ — произвольное семейство классификаторов, не обязательно линейное:

$$\mathbb{A} = \bigl\{a\colon \mathbb{X} \to \mathbb{Y} \bigm| a(x) = \operatorname{sign}\bigl(f(x, w)\bigr),\ w \in \mathbb{R}^n\bigr\}. \tag{2.2}$$

Следующая задача показывает, что, рассматривая множества векторов ошибок, порождаемых семействами классификаторов, не обязательно вводить множество меток классов и функцию истинной классификации.

Задача 2.4. (1) *Доказать, что если взять множество алгоритмов* $\mathbb{W} = \mathbb{R}^n$ *и индикатор ошибок* $I(w, x) = \bigl[\varphi(w, x) \geqslant 0\bigr]$, *то можно так подобрать непрерывную функцию* $\varphi(w, x)$, *что множество векторов ошибок* $\bigl\{(w)_{\mathbb{X}}\colon w \in \mathbb{W}\bigr\}$ *совпадёт с множеством векторов ошибок* $\bigl\{(a)_{\mathbb{X}}\colon a \in \mathbb{A}\bigr\}$. *Какие дополнительные предположения необходимо для этого принять? Насколько они обременительны?*

§2.2 Графы связности, порождаемые линейными классификаторами

Рассматривается множество всех векторов ошибок, порождаемых линейными классификаторами (2.1) на заданной выборке $\mathbb{X}$ в пространстве размерности $n = 2$.

В *графе связности* вершинами являются все векторы ошибок $\bigl\{(a)_{\mathbb{X}}\colon a \in \mathbb{A}\bigr\}$; рёбрами соединяются векторы, имеющие хэммингово расстояние 1.

Задача 2.5. (1) *Привести пример выборки $\mathbb{X}$, для которой граф связности представляет собой двумерную решетку (у каждой вершины не более четырёх рёбер). Привести примеры выборок, для которых в графе связности имеются вершины с числом рёбер более четырёх.*

Задача 2.6. (1) *Доказать, что для любого L существует выборка длины L, для которой в графе связности имеется вершина с числом рёбер $L - 1$.*

Задача 2.7. (10) *Верна ли гипотеза, что линейный классификатор в пространстве размерности n порождает граф связности, у которого среднее число инцидентности вершин стремится к $2n$ при $L \to \infty$?*

§2.3 Общий случай: связные семейства

В работах Силла [9, 8] доказывается, что семейства функций, непрерывных по параметрам, порождают связные графы векторов ошибок.

Пусть $\mathbb{A} = \mathbb{R}^h$ и функция потерь имеет вид $I(a, x) = \bigl[\varphi(a, x) \geqslant 0\bigr]$, где функция $\varphi(a, x)$ непрерывна по a. Рассмотрим множества алгоритмов $A_r \subset \mathbb{A}$, в которых изменение вектора ошибок происходит одновременно на r объектах:

$$A_r = \bigl\{a \in A \bigm| \forall\varepsilon > 0 \;\; \exists a' \in A\colon \|a - a'\| \leqslant \varepsilon,\; \rho(a, a') = r\bigr\},$$

где $\|\cdot\|$ — евклидова норма в $\mathbb{R}^n$, $\rho(a, a')$ — хэммингово расстояние между векторами ошибок алгоритмов a и a'.

Задача 2.8. (3) *Можно ли утверждать, что $A_1 \neq \varnothing$, $A_2 = \cdots = A_L = \varnothing$ почти всегда? При каких ограничениях это действительно так? Насколько эти ограничения обременительны?*

Задача 2.9. (3) *Пусть генеральная выборка $\mathbb{X}$ длины L получена случайно и независимо из непрерывного распределения $p(x)$ на $\mathbb{X}$. Можно ли утверждать, что с вероятностью 1 (т. е. для почти всех выборок) $A_2 = \cdots = A_L = \varnothing$?*

3 Экспериментальные исследования (практикум)

Реализация следующих программ и экспериментирование с ними поможет не только проверять правильность теоретических оценок, но и замечать новые интересные факты, выдвигать и проверять новые гипотезы.

Графики желательно генерировать в виде chd-файлов в формате ChartLib.

§3.1 Эмпирическое исследование вероятности переобучения

Задача 3.1. (5) *Написать программу, позволяющую:*

- *генерировать последовательности векторов ошибок $a_1, \ldots, a_D$ в виде бинарной матрицы ошибок размера $L \times D$ (в матрице ошибок не должно быть одинаковых столбцов — векторов ошибок); легко заменять генераторы данных;*
- *вычислять точные верхние и нижние оценки вероятности переобучения, если соответствующие формулы известны;*
- *вычислять эмпирические оценки вероятности переобучения методом Монте-Карло, т. е. по случайному подмножеству из N разбиений $(X, \bar{X})$; требуется вычислять две оценки, соответствующие двум стратегиям выбора алгоритма в случаях неоднозначного минимума эмпирического риска:*
  — верхняя оценка $\bar{Q}_\varepsilon$ соответствует стратегии «худший из лучших» (метод $\mu_х$);
  — нижняя оценка $\underline{Q}_\varepsilon$ соответствует стратегии «лучший из лучших» (метод $\mu_л$).
- *строить графики, в которых по оси X откладывается порядковый номер алгоритма d, по оси Y:*
  — эмпирические оценки $\bar{Q}_\varepsilon$ и $\underline{Q}_\varepsilon$ для подпоследовательности $a_1, \ldots, a_d$;
  — точные значения $\bar{Q}_\varepsilon$ и $\underline{Q}_\varepsilon$ для $a_1, \ldots, a_d$ (если формулы известны);
  — число ошибок $n(a_d, \mathbb{X})$;
  — доля разбиений, на которых $\mu_х X = a_d$;
  — доля разбиений, на которых $\mu_л X = a_d$.
- *строить графики, в которых по оси Y откладываются точные (если известны) и эмпирические значения $\bar{Q}_\varepsilon$ и $\underline{Q}_\varepsilon$, по оси X:*
  — число ошибок лучшего алгоритма m;
  — длина обучения ℓ;
  — длина контроля k.

Следующие три задачи направлены на экспериментальную проверку теоретических оценок и исследование перечисленных выше зависимостей. Число разбиений N должно быть достаточно большим (например, десять тысяч), чтобы совпадение теоретических и эмпирических оценок было очевидно.

Задача 3.2. (3) *Исследовать $\bar{Q}_\varepsilon$, $\underline{Q}_\varepsilon$ монотонной цепочки.*

Задача 3.3. (3) *Исследовать $\bar{Q}_\varepsilon$, $\underline{Q}_\varepsilon$ унимодальной цепочки.*

Задача 3.4. (3) *Исследовать $\bar{Q}_\varepsilon$, $\underline{Q}_\varepsilon$ единичной окрестности лучшего алгоритма.*

Следующий эксперимент направлен на проверку гипотезы, что монотонная цепочка и цепочка случайных инверсий [1] ведут себя практически одинаково с точки зрения переобучения. Если это подтвердится, то можно будет ограничиться изучением монотонных цепочек, как достаточно точной модели реальных цепочек.

Задача 3.5. (3) *Сравнить, представив на одном графике, точные значения $\bar{Q}_\varepsilon$ для монотонной цепочки, их эмпирические оценки и эмпирические оценки $\bar{Q}_\varepsilon$ для цепочки случайных инверсий при одинаковых m. Увеличиваются ли различия между $\bar{Q}_\varepsilon$ монотонной цепочки и цепочки случайных инверсий с ростом ℓ и m?*

§3.2 Визуализация графов связности

Задача 3.6. (5) *Написать программу, позволяющую:*

- *генерировать двумерные модельные задачи классификации на два класса;*
- *строить точечный график выборки;*
- *строить бинарную матрицу ошибок, порождаемую всевозможными двумерными линейными классификаторами на заданной выборке (в матрице ошибок не должно быть одинаковых столбцов — векторов ошибок);*
- *отображать граф связности в виде плоского точечного графика, со всеми рёбрами, желательно без самопересечений.*

Задача 3.7. (3) *Отобразить выборки и графы связности для Задач 2.5 и 2.6.*

4 Нижние и асимптотические оценки

В [2] получены точные верхние оценки вероятности переобучения для четырёх частных случаев: монотонной цепочки, унимодальной цепочки, единичной окрестности лучшего алгоритма и пары алгоритмов. Эти результаты можно развивать по нескольким направлениям.

§4.1 Точные нижние оценки

Точные *верхние* оценки вероятности переобучения $\bar{Q}_\varepsilon$ получены при условии, что в случаях неоднозначного выбора алгоритма, доставляющего минимальное значение эмпирическому риску, выбирается наихудший алгоритм (стратегия «худший из лучших»). Точные *нижние* оценки $\underline{Q}_\varepsilon$ могут быть получены аналогичным образом, если придерживаться стратегии «лучший из лучших».

Если окажется, что верхние и нижние оценки достаточно близки, то впредь можно будет ограничиваться получением верхних оценок.

Задача 4.1. (2) *Выписать точные нижние оценки вероятности переобучения для монотонной цепочки. Исследовать зависимость разности точной верхней и точной нижней оценок от длины выборки L.*

Задача 4.2. (2) *Выписать точные нижние оценки вероятности переобучения для унимодальной цепочки. Исследовать зависимость разности точной верхней и точной нижней оценок от длины выборки L.*

Задача 4.3. (2) *Выписать точные нижние оценки вероятности переобучения для пары алгоритмов. Исследовать зависимость разности точной верхней и точной нижней оценок от длины выборки L.*

§4.2 Асимптотические оценки

Точные оценки вероятности переобучения имеют громоздкий вид и сложны для вычислений. Наверняка эти комбинаторные выражения можно упростить. В асимптотике $L \to \infty$ будем предполагать, что $\frac{\ell}{L} \to \mathrm{const}$, $\frac{m}{L} \to \mathrm{const}$.

Задача 4.4. (3) *Найти асимптотическое выражение точной верхней оценки вероятности переобучения для монотонной цепочки.*

Задача 4.5. (3) *Найти асимптотическое выражение точной верхней оценки вероятности переобучения для унимодальной цепочки.*

Задача 4.6. (3) *Найти асимптотическое выражение точной верхней оценки вероятности переобучения для единичной окрестности лучшего алгоритма.*

5 Обобщения цепочек и окрестностей

Семейства алгоритмов, рассмотренные в [2], являются искусственными примерами, цель которых — продемонстрировать применимость общего подхода. Для перехода от модельных примеров к реальным семействам будем постепенно усложнять задачи, учитывая следующие соображения:
— реальные семейства, как правило, являются связными и расслоенными;
— вероятность переобучения определяется окрестностью лучшего алгоритма;
— сначала попробуем получить оценки для окрестностей «простого вида»;
— возможно, они окажутся достаточно точными и в более общих случаях;

Во всех задачах данной секции требуется выписать точные оценки P_a и Q_ε.

§5.1 Обобщения монотонных цепочек

Обобщения монотонных цепочек направлены на постепенный переход к параметрическим семействам алгоритмов большей размерности. Конечная цель — понять, как размерность пространства параметров влияет на вероятность переобучения. С ростом размерности она должна увеличиваться, однако не так быстро, как предсказывают VC-оценки.

Задача 5.1. (3) *Пара параллельных монотонных цепочек* — это множество векторов ошибок, состоящее из двух монотонных цепочек $a_0, \dots, a_d$ и $a'_0, \dots, a'_d$ таких, что $\rho(a_t, a'_t) = 1$ *и* $n(a_t, \mathbb{X}) = n(a'_{t-1}, \mathbb{X}) = m + t$.

Задача 5.2. (6) *Монотонная сетка* — это множество векторов ошибок a_{st}, $(s,t) \in$ $\in \{0, \dots, d\}^2$, *такое, что* $\rho(a_{st}, a_{s,t+1}) = 1$, $\rho(a_{st}, a_{s+1,t}) = 1$, $n(a_{st}, \mathbb{X}) = m + s + t$.

Задача 5.3. (1) *Построить пример двумерной модельной задачи классификации, в которой множество векторов ошибок является монотонной сеткой.*

Задача 5.4. (10) *Монотонная h-мерная сетка* — это множество векторов ошибок a_J, *индексируемых h-мерным целочисленным вектором* $J = (j_1, \dots, j_h) \in \{0, \dots, d\}^h$, *такое, что* $\rho(a_J, a_{J'}) = 1$ *для всех пар индексов* J, J', *отличающихся на единицу только в одной координате, и* $n(a_J, \mathbb{X}) = m + j_1 + \dots + j_h$. *Учесть, что «верхушка» сетки может срезаться условием* $m + j_1 + \dots + j_h \leqslant L$.

§5.2 Обобщения единичной окрестности

Единичная окрестность лучшего алгоритма является первым шагом к рассмотрению многомерного семейства с расслоением. Следующим шагом логично было бы увеличить радиус окрестности.

Задача 5.5 является искусственной. Пока не ясно, существуют ли реальные семейства, устроенные таким образом. Однако решение задачи представляется относительно несложным, а разработка техники её решения — полезной.

Задача 5.5. (3) *Множество векторов ошибок является хэмминговым шаром радиуса r с центром в некотором векторе $a_ц$. Известно, что $n(a_ц, X^L) = m_ц \geqslant r$. При этом наилучший алгоритм, очевидно, допускает $m = m_ц - r$ ошибок.*

В следующей задаче требуется уточнять саму постановку. Идея в том, что в пространствах фиксированной размерности окрестность наилучшего алгоритма не может включать в себя все возможные векторы ошибок. В результате из хэммингова шара (точнее, из его верхнего полушария) выделяется некоторое «многообразие меньшей размерности».

Задача 5.6. (20) *Множество векторов ошибок A является подмножеством хэммингова шара радиуса r с центром в некотором векторе a_0 с известной частотой ошибок $m = n(a_0, \mathbb{X})$. Подмножество определяется двумя ограничениями. Во-первых, в A нет векторов лучше a_0: $n(a, \mathbb{X}) \geqslant n(a_0, \mathbb{X})$ для любого вектора $a \in A$ (таким образом, выделяется верхнее полушарие хэммингова шара). Во-вторых, для любого вектора $a \in A$ существует не более d векторов $a' \in A$ на единичном расстоянии от него $\rho(a, a') = 1$. Как Q_ε зависит от d и m?*

§5.3 Реальные семейства алгоритмов

Задача 5.7. (10) *Параметрическое семейство A линейных алгоритмов классификации над n вещественными признаками $f_1(x), \dots, f_n(x)$ с параметрами $w_1, \dots, w_n$:*

$$a(x) = \operatorname{sign}\left(\sum_{j=1}^{n} w_j f_j(x)\right).$$

Задача 5.8. (5) *Предыдущая задача при $n = 2$.*

Задача 5.9. (10) *Параметрическое семейство A конъюнкций над n вещественными признаками $f_1(x), \dots, f_n(x)$: с параметрами $w_1, \dots, w_n$:*

$$a(x) = \bigwedge_{j=1}^{n} \bigl[f_j(x) \leqslant w_j\bigr].$$

Задача 5.10. (5) *Предыдущая задача при $n = 2$.*

Задача 5.11. (20) *Рассматривается семейство линейных алгоритмов классификации над n вещественными признаками. На норму вектора параметров накладывается дополнительное ограничение регуляризации $\|w\| \leqslant \tau$. Как при этом меняется вероятность переобучения?*

6 Семейства, задаваемые перечислением векторов ошибок

В этой секции рассматриваются семейства A, заданные не слишком длинной совокупностью векторов ошибок, в общем случае произвольных. «Не слишком длинной» означает, что оценки должны вычисляться за приемлемое время при условии, что векторы ошибок в явном виде хранятся в памяти компьютера.

§6.1 Оценки по ненаблюдаемым данным

В следующих задачах векторы ошибок предполагаются известными полностью. На практике это невозможно, т. к. при фиксированном разбиении $(X, \bar{X})$ скрытая (контрольная) выборка неизвестна. Тем не менее, получение оценок Q_ε и в этих случаях представляет интерес. Во-первых, это позволит ускорить расчёты экспериментов, заменив эмпирические оценки по методу Монте-Карло точными оценками. Во-вторых, из этих оценок, возможно, будут проще получаться верхние оценки для некоторых семейств алгоритмов.

Простейшим примером такого явно заданного семейства является двухэлементное семейство, для которого точная оценка получена в [1, 2].

Задача 6.1. (3) *Выписать точные оценки P_a и Q_ε для семейства из трёх произвольных алгоритмов $A = \{a_1, a_2, a_3\}$, считая, что заданы восемь параметров*

$$m_{abc} = \#\{x \in \mathbb{X} \colon I(a_1, x) = a,\ I(a_2, x) = b,\ I(a_3, x) = c\}, \quad a, b, c \in \{0, 1\}.$$

Задача 6.2. (2) *В задаче 6.1 рассмотреть частный случай — трёхэлементное семейство без расслоения: $m_{110} = m_{101} = m_{011} = m_{100} = m_{010} = m_{001} = m$. Построить зависимость Q_ε от хэммингова расстояния между векторами ошибок (которое равно $4m$) при нескольких (небольших) значениях m_{111}.*

Задача 6.3. (2) *В задаче 6.1 рассмотреть частный случай — трёхэлементное семейство с расслоением: $m_{011} = m_{001} = m$, $m_{110} = m_{101} = m_{100} = m_{010} = 0$. Построить зависимость Q_ε от хэммингова расстояния между векторами ошибок $\rho(a_1, a_2) = \rho(a_2, a_3) = m$ при нескольких (небольших) значениях m_{111}.*

Задача 6.4. (10) *Задан конечный набор векторов ошибок $A_d = \{a_1, \ldots, a_d\}$. Найти рекуррентную формулу, эффективно вычисляющую Q_ε для A_d, предполагая, что Q_ε уже вычислено для A_{d-1}, и добавляется (полностью известный) вектор a_d. Искомая формула не должна содержать явного суммирования по всем C_L^ℓ разбиениям.*

Подсказка 1: при добавлении нового алгоритма он «забирает» некоторые обучающие выборки X у предыдущих алгоритмов a_t, $(t < d)$, соответственно уменьшая их P_t, и на столько же увеличивая свою P_d.

Подсказка 2: разрешается на каждой итерации сохранять некоторую агрегированную информацию $\mathfrak{I}_{d-1} = \mathfrak{I}(a_1, \ldots, a_{d-1})$ о совокупности всех предыдущих векторов, которая позволяла бы упростить вычисление Q_ε и заодно вычислить $\mathfrak{I}_d$. Что это за информация?

Подсказка 3: разрешается предполагать, что векторы a_t добавляются в порядке неубывания числа ошибок $n(a_t, \mathbb{X})$, или в каком-либо другом удобном порядке.

Решение следующей задачи существенно зависит от предыдущей. С одной стороны, это попытка уйти от непосредственного использования данных, содержащихся в векторах из A. С другой стороны, это ещё одна попытка учесть размерность пространства параметров, альтернативная задачам 5.4 и 5.6.

Задача 6.5. (15) *Получить оценки для Q_ε, если на каждом шаге t известен не сам добавляемый вектор a_t, а только число ошибок $n(a_t, \mathbb{X})$ и число векторов из A_{t-1}, имеющих единичное хэммингово расстояние до a_t.*

§6.2 Оценки по наблюдаемым данным

В этих задачах предполагается, что векторы ошибок известны не полностью, а только на наблюдаемой выборке X при заданном разбиении $(X, \bar{X})$, причём их можно явным образом перебирать и использовать их данные в вычислениях. На практике возможность явного перебора реализуется не всегда, а только в переборных методах обучения. Примеры таких методов:

— выбор лучшей модели по отложенным данным (hold-out model selection);
— стохастический поиск (например, генетические алгоритмы);
— поиск информативных конъюнкций в логических классификаторах;
— отбор признаков в линейной регрессии с фиксированными коэффициентами.

Предполагается, что получение оценок вероятности переобучения позволит некотором образом улучшить эти методы.

Начнём с простейшего случая — семейства из двух алгоритмов.

Задача 6.6. (10) *Для двухэлементного семейства алгоритмов $A = \{a_1, a_2\}$ при некотором разбиении $(X, \bar{X})$, выбранном случайно и равновероятно, в наблюдаемой подвыборке X оказалось:*

— *s_0 объектов, на которых ошиблись оба алгоритма;*
— *s_1 объектов, на которых ошибся только алгоритм a_1;*
— *s_2 объектов, на которых ошибся только алгоритм a_2.*

Оценить сверху вероятность переобучения Q_ε для метода минимизации эмпирического риска. Как Q_ε зависит от $s_1 + s_2$ — наблюдаемого расстояния между алгоритмами? Как Q_ε зависит от $|s_1 - s_2|$ — наблюдаемой величины расслоения алгоритмов?

Задача 6.7. (∞) *Для d-элементного семейства алгоритмов $A = \{a_1, \dots, a_d\}$ при некотором разбиении $(X, \bar{X})$, выбранном случайно и равновероятно, известны векторы ошибок $(a_1)_X, \dots, (a_d)_X$. Оценить сверху вероятность переобучения Q_ε для метода минимизации эмпирического риска. Как Q_ε зависит от расстояния между алгоритмов? Использование каких характеристик схожести алгоритмов позволяет записать более точную оценку Q_ε?*

Список литературы

[1] *Воронцов К. В.* Эффекты расслоения и сходства в семействах алгоритмов и их влияние на вероятность переобучения // *Pattern Recognition and Image Analysis.* — 2009 (в печати). — Vol. ??, no. ?? — Pp. ??–??
http://www.MachineLearning.ru/wiki/images/0/0e/Voron09roai2008.pdf.

[2] *Воронцов К. В.* Точные оценки вероятности переобучения // *??* — 2009 (в печати). — Vol. ??, no. ?? — Pp. ??–??
http://www.MachineLearning.ru/wiki/images/1/18/Voron09exact.pdf.

[3] *Воронцов К. В.* Комбинаторная вероятность и точность оценок обобщающей способности (Combinatorial probability and the tightness of generalization bounds) // *Pattern Recognition and Image Analysis.* — 2008. — Vol. 18, no. 2. — Pp. 243–259.
http://www.ccas.ru/frc/papers/voron08pria.pdf.

[4] *Вапник В. Н., Червоненкис А. Я.* Теория распознавания образов. — М.: Наука, 1974.

[5] *Вапник В. Н.* Восстановление зависимостей по эмпирическим данным. — М.: Наука, 1979.

[6] *Lugosi G.* On concentration-of-measure inequalities. — Machine Learning Summer School, Australian National University, Canberra. — 2003.
http://citeseer.ist.psu.edu/lugosi98concentrationmeasure.html.

[7] *Bax E. T.* Similar classifiers and VC error bounds: Tech. Rep. CalTech-CS-TR97-14: 6 1997.
http://citeseer.ist.psu.edu/bax97similar.html.

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

[9] *Sill J.* Monotonicity and connectedness in learning systems: Ph.D. thesis / California Institute of Technology. — 1998.
http://etd.caltech.edu/etd/available/etd-09222005-110351/.