

# Комбинаторная теория надёжности обучения по прецедентам

Воронцов Константин Вячеславович

Диссертация на соискание ученой степени  
доктора физико-математических наук  
05.13.17 — теоретические основы информатики

Научный консультант — чл.-корр РАН К. В. Рудаков

ВЦ РАН, 22 апреля 2010

## Содержание

- 1 Проблема переобучения**
  - Задача оценивания вероятности переобучения
  - Проблема завышенности оценок
  - Комбинаторно-дискретная постановка задачи
- 2 Эксперименты: анализ факторов завышенности**
  - Измерение факторов завышенности
  - Эксперименты с цепочками алгоритмов
- 3 Оценки вероятности переобучения**
  - Простые частные случаи
  - Порождающие и запрещающие множества
  - Модельные семейства алгоритмов
  - Рекуррентное вычисление вероятности переобучения
- 4 Оценки полного скользящего контроля**
  - Функционал полного скользящего контроля
  - Метод ближайшего соседа
  - Монотонные алгоритмы классификации

## Задача обучения по прецедентам

$X = \{x_1, \dots, x_\ell\} \subset \mathbb{X}$  — обучающая выборка объектов;

$A$  — множество (семейство, модель) алгоритмов;

$\mu: \mathbb{X}^\ell \rightarrow A$  — метод обучения;

$a = \mu X$  — алгоритм, построенный методом  $\mu$  по выборке  $X$ ;

$\nu(a, X)$  — частота ошибок алгоритма  $a$  на выборке  $X$ ;

$\bar{X} = \{x'_1, \dots, x'_k\} \subset \mathbb{X}$  — контрольная выборка;

$\nu(a, \bar{X})$  — частота ошибок алгоритма  $a$  на выборке  $\bar{X}$ ;

### Проблема переобучения («переподгонки», overfitting)

На практике  $\nu(a, \bar{X})$ , как правило, превышает  $\nu(a, X)$ .

## Проблема переобучения (пример)

Зависимость  $\nu(\mu X, \bar{X})$  от  $\nu(\mu X, X)$  при различных  $\mu$ .

Частота ошибок на контроле, %

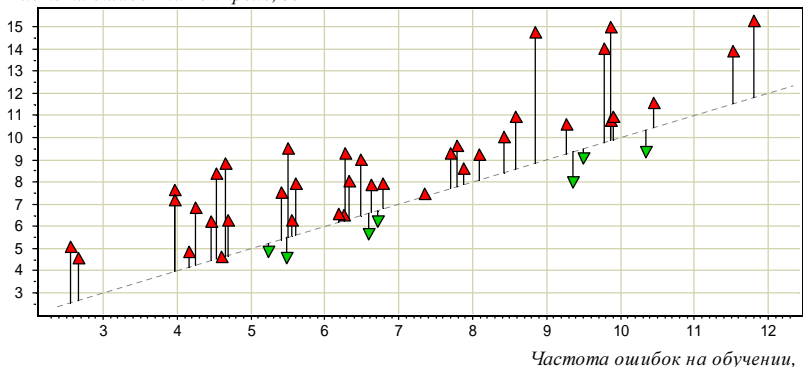


График построен по реальным данным (задача прогнозирования отдалённых результатов хирургического лечения атеросклероза).

## Задача оценивания вероятности переобучения

Дано:

- наблюдаемая обучающая выборка  $X \subset \mathbb{X}$ ;
- семейство алгоритмов  $A$ ;
- метод обучения  $\mu: \mathbb{X}^\ell \rightarrow A$ .

Требуется:

- оценить  $\nu(\mu X, \bar{X})$  на скрытой контрольной выборке  $\bar{X} \subset \mathbb{X}$ ;
- для этого достаточно оценить *вероятность переобучения*:

$$Q_\varepsilon = P_{X, \bar{X}} \left[ \underbrace{\nu(\mu X, \bar{X}) - \nu(\mu X, X)}_{\delta_\mu(X, \bar{X})} \geq \varepsilon \right].$$

Опр.  $\delta_\mu(X, \bar{X})$  — переобученность алгоритма  $\mu X$  на  $(X, \bar{X})$ .

# Теория восстановления зависимостей по эмпирическим данным

## Теорема (Вапник и Червоненкис, 1971)

Для любой меры  $P$ , метода  $\mu$ , конечного  $A$  и  $\varepsilon \in (0, 1)$

$$\begin{aligned}
 Q_\varepsilon &= P[\nu(\mu X, \bar{X}) - \nu(\mu X, X) \geq \varepsilon] \leq \\
 &\stackrel{(1)}{\leq} P\left[\max_{a \in A} (\nu(a, \bar{X}) - \nu(a, X)) \geq \varepsilon\right] \leq \\
 &\stackrel{(2)}{\leq} |A| \cdot P[\nu(a, \bar{X}) - \nu(a, X) \geq \varepsilon] \stackrel{(3)}{\leq} |A| \cdot \frac{3}{2} e^{-\varepsilon \ell^2}.
 \end{aligned}$$

- (1) принцип равномерной сходимости;
- (2) неравенство Буля (union bound);
- (3) неравенство Хёффдинга (концентрация вероятностной меры).

## Проблема завышенности VC-оценки

### Анализ шагов доказательства:

- 1 оценка равномерной сходимости сильно завышена, когда в  $A$  есть мало «хороших» и много «плохих» алгоритмов;
- 2 неравенство Буля сильно завышено, когда в  $A$  есть много схожих алгоритмов;
- 3 неравенство Хёффдинга завышено в несколько раз.

### Анализ того, какая информация используется:

- оценка зависит только от  $|A|$  и  $\ell$ ;
- не учтены свойства конкретного метода обучения  $\mu$ ;
- не учтены свойства конкретной выборки  $\mathbb{X}$ ;

**Вывод:** VC-оценка — это оценка «худшего случая».

## Проблема завышенности VC-оценки

Зависимость достаточной длины обучения  $\ell$   
от ёмкости  $h$ ,  $|A| = \frac{3}{2} \frac{(2\ell)^h}{h!}$ , точности  $\varepsilon$  и надёжности  $\eta$ :

$h$	$\eta = 0.01$			$\eta = 1$		
	$\varepsilon = 0.05$	0.1	0.2	$\varepsilon = 0.05$	0.1	0.2
0	2404	601	150	562	140	35
2	9012	1946	408	6963	1423	273
5	19884	4192	848	17823	3664	711
10	38160	7974	1589	36095	7444	1452
20	74855	15572	3082	72789	15043	2944
50	185193	38433	7575	183127	37903	7437
100	369275	76581	15075	367208	76051	14937

На практике, как правило, достаточно существенно меньшей  $\ell$ .

**Вывод:** случаи малых выборок и сложных семейств  
лежат за границами применимости VC-теории.



## Развитие теории статистического обучения, 1968–2009

- Теория равномерной сходимости [Вапник, Червоненкис, 1968]
- Корректные алгебры ограниченной ёмкости [Матросов, 1980]
- Theory of learnable (PAC-learning) [Valiant, 1982]
- Data-dependent bounds [Haussler, 1992]
- **Connected function classes [Sill, 1995]**
- **Similar classifiers VC bounds [Bax, 1997]**
- Margin based bounds [Bartlett, 1998]
- Self-bounding learning algorithms [Freund, 1998]
- **Rademacher complexity [Koltchinskii, 1998]**
- Adaptive microchoice bounds [Langford, Blum, 2001]
- Algorithmic stability [Bousquet, Elisseeff, 2002]
- Algorithmic luckiness [Herbrich, Williamson, 2002]
- **Shell bounds [Langford, 2002]**
- **PAC-Bayes bounds [McAllester, 1999; Langford, 2005]**

## Актуальность данного исследования

- Ни один из известных подходов
  - не устраняет *всех* причин завышенности;
  - не даёт *точных* оценок вероятности переобучения;
- Большинство подходов используют принцип равномерной сходимости.
- Многие подходы используют неравенство Буля.
- Все подходы используют завышенные или асимптотические неравенства концентрации вероятностной меры (Хёффдинга, Чернова, МакДиармида, Талаграна, и т.п.).
- Незначительные улучшения оценок достигаются путём значительного усложнения математического аппарата.

**Вывод:** в теории статистического обучения необходимы новые подходы и методы.

## Цели и методы исследования

### Основная цель диссертационной работы

Создание нового математического аппарата для получения точных оценок вероятности переобучения.

### Основные этапы исследования

- 1 Экспериментальное измерение факторов завышенности и понимание причин завышенности VC-оценок (Глава 3).
- 2 Разработка общих методов получения точных оценок и исследование модельных частных случаев (Глава 4).
- 3 Применение — создание новых методов обучения (Глава 5).

## Матрица ошибок множества алгоритмов на выборке

$\mathbb{X} = \{x_1, \dots, x_L\}$  — конечное множество объектов;

$A = \{a_1, \dots, a_D\}$  — конечное множество алгоритмов;

$I(a, x) = [\text{алгоритм } a \text{ ошибается на объекте } x];$

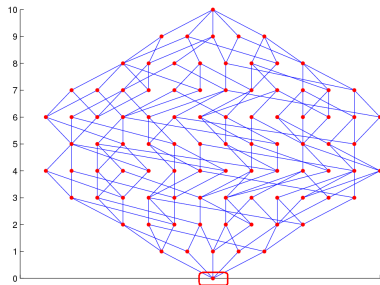
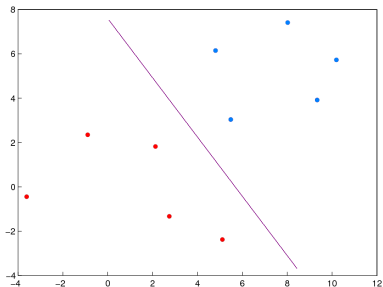
$L \times D$ -матрица ошибок с попарно различными столбцами:

	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$\dots$	$a_D$	
$x_1$	1	1	0	0	0	1	$\dots$	1	$X$ — наблюдаемая (обучающая) выборка длины $\ell$
$\dots$	0	0	0	0	1	1	$\dots$	1	
$x_\ell$	0	0	1	0	0	0	$\dots$	0	
$x_{\ell+1}$	0	0	0	1	1	1	$\dots$	0	$\bar{X}$ — скрытая (контрольная) выборка длины $k = L - \ell$
$\dots$	0	0	0	1	0	0	$\dots$	1	
$x_L$	0	1	1	1	1	1	$\dots$	0	

$n(a, X)$  — число ошибок алгоритма  $a$  на выборке  $X \subset \mathbb{X}$ ;

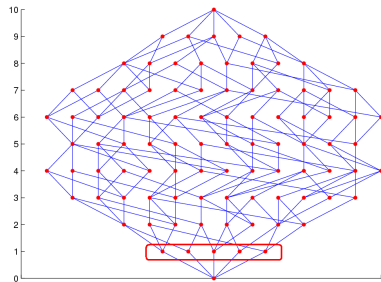
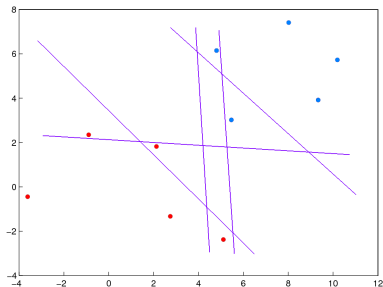
$\nu(a, X) = n(a, X)/|X|$  — частота ошибок  $a$  на выборке  $X \subset \mathbb{X}$ ;

## Пример. Матрица ошибок линейных классификаторов



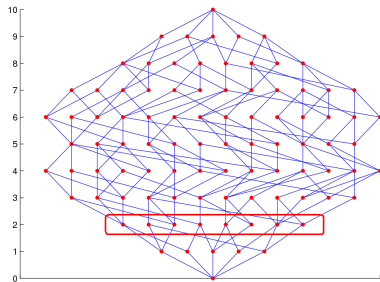
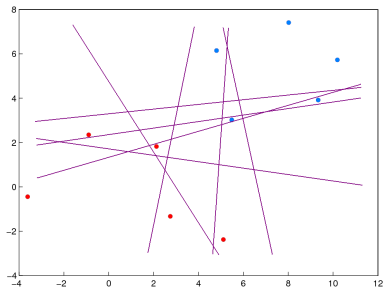
$$\begin{array}{l} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{array} \left| \begin{array}{l} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right|$$

## Пример. Матрица ошибок линейных классификаторов



$x_1$	0	1	0	0	0	0
$x_2$	0	0	1	0	0	0
$x_3$	0	0	0	1	0	0
$x_4$	0	0	0	0	1	0
$x_5$	0	0	0	0	0	1
$x_6$	0	0	0	0	0	0
$x_7$	0	0	0	0	0	0
$x_8$	0	0	0	0	0	0
$x_9$	0	0	0	0	0	0
$x_{10}$	0	0	0	0	0	0

# Пример. Матрица ошибок линейных классификаторов



$x_1$	0	1	0	0	0	0	1	0	0	0	0	1	1	0	...
$x_2$	0	0	1	0	0	0	1	1	0	0	0	0	0	0	...
$x_3$	0	0	0	1	0	0	0	1	1	0	0	0	0	1	...
$x_4$	0	0	0	0	1	0	0	0	1	1	0	0	0	0	...
$x_5$	0	0	0	0	0	1	0	0	0	1	1	1	0	0	...
$x_6$	0	0	0	0	0	0	0	0	0	1	0	1	0	0	...
$x_7$	0	0	0	0	0	0	0	0	0	0	0	0	0	1	...
$x_8$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	...
$x_9$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	...
$x_{10}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	...

## Единственное вероятностное допущение

Пусть  $\mathbb{X} = \{x_1, \dots, x_L\}$  — конечное множество объектов.

### Аксиома

Все  $C_L^\ell$  разбиений  $X \sqcup \bar{X} = \mathbb{X}$  равновероятны, где

$X$  — наблюдаемая обучающая выборка,  $|X| = \ell$ ;

$\bar{X}$  — скрытая контрольная выборка,  $|\bar{X}| = k = L - \ell$ ;

*Вероятность* определяется как доля разбиений выборки:

$$Q_\varepsilon = \mathbf{P}[\delta_\mu(X, \bar{X}) \geq \varepsilon] = \frac{1}{C_L^\ell} \sum_{\substack{X, \bar{X} \\ X \sqcup \bar{X} = \mathbb{X}}} [\delta_\mu(X, \bar{X}) \geq \varepsilon].$$

Это аналог стандартной гипотезы о *независимости* наблюдений.  
Теория меры и предельный переход  $|\mathbb{X}| \rightarrow \infty$  не используются.



## Преимущества комбинаторно-дискретной постановки задачи

- Легко измерить факторы завышенности VC-оценки экспериментально, по конечной выборке  $\mathbb{X} = \{x_1, \dots, x_L\}$ .
- Комбинаторные методы позволяют получать точные оценки вероятности переобучения.
- Появляется возможность исследовать и учитывать в оценках структуру графа расслоения и связности семейства алгоритмов.

## Экспериментальное измерение факторов завышенности

Несмещённая оценка вероятности переобучения по случайному подмножеству разбиений  $X_n \sqcup \bar{X}_n = \mathbb{X}$ ,  $n = 1, \dots, N$ :

$$\hat{Q}_\varepsilon = \frac{1}{N} \sum_{n=1}^N [\delta_\mu(X_n, \bar{X}_n) \geq \varepsilon] \approx Q_\varepsilon \leq |A| \cdot \frac{3}{2} e^{-\varepsilon \ell^2}.$$

Измерение  $\hat{Q}_\varepsilon$  позволяет также оценить:

- 1 *эффективный локальный коэффициент разнообразия* — значение  $\Delta = |A|$ , при котором оценка не завышена:

$$\hat{Q}_\varepsilon = \hat{\Delta} \cdot \frac{3}{2} e^{-\varepsilon \ell^2} \Rightarrow \hat{\Delta} = \frac{\hat{Q}_\varepsilon}{\frac{3}{2} e^{-\varepsilon \ell^2}}.$$

- 2 все три фактора завышенности:

$$\hat{Q}_\varepsilon \cdot r_1 \cdot r_2 \cdot r_3 = |A| \cdot \frac{3}{2} e^{-\varepsilon \ell^2}.$$

## Результаты эксперимента на 6 задачах классификации

Факторы завышенности  $r_1, r_2, r_3$

и оценка  $\hat{\Delta}$  с доверительным интервалом  $[\hat{\Delta}_1; \hat{\Delta}_2]$ :

Задача	класс $y$	$r_1$	$r_2$	$r_3$	$\hat{\Delta}$	$[\hat{\Delta}_1; \hat{\Delta}_2]$
crx	0	2 759	680	32.6	24	[10; 41]
	1	1 104	1700	11.6	12	[11; 180]
german	1	15 215	1500	10.9	54	[38; 530]
	2	44 400	9000	9.9	1.9	[1.0; 2.2]
hepatitis	0	308	280	9.5	83	[11; 148]
	1	132	680	22.5	15	[12; 27]
horse-colic	1	151	4500	7.2	7	[2; 9]
	2	504	3400	7.3	6	[3; 6]
hypothyroid	0	1 964 200	400	16.5	21	[3; 220]
	1	581 400	460	28.7	30	[2; 44]
promoters	0	555	340	9.8	72	[36; 230]
	1	510	790	6.9	18	[9; 22]

## Выводы из экспериментов

- $r_1 = 10^2 \dots 10^5$  — существенный фактор,  
**не учитывается *расслоение* множества алгоритмов:**  
чем выше  $\nu(a, \mathbb{X})$ , тем меньше  $P[\mu X = a]$ ;  
эффективное число алгоритмов  $\hat{\Delta} \sim 10^1 \dots 10^2$ .
- $r_2 = 10^3 \dots 10^4$  — существенный фактор,  
**не учитывается *сходство* алгоритмов.**
- $r_3 = 10^1 \dots 10^2$  — несущественный фактор.

Используемые на практике множества  $A$ , как правило,

- *расслоены*, т.к. предназначены для решения многих задач, следовательно, содержат много алгоритмов, «плохих» для данной задачи;
- *связны*, т.к. непрерывны по параметрам.

## Эксперимент с монотонной цепочкой алгоритмов

Цель эксперимента: понять, как *связность* и *расслоение* влияют на вероятность переобучения.

Монотонная цепочка:

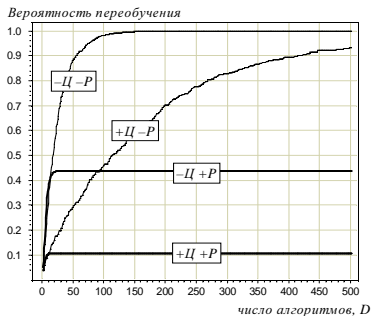
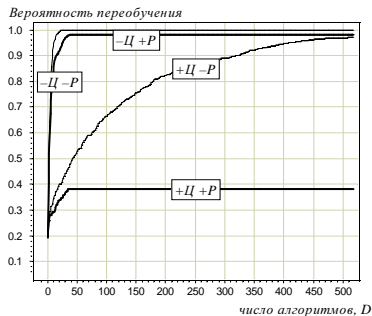
	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$\dots$	$a_D$
$x_1$	1	1	1	1	1	1	1
$x_2$	$0 \rightarrow 1$	1	1	1	1	1	1
$x_3$	0	$0 \rightarrow 1$	1	1	1	1	1
$x_4$	0	0	$0 \rightarrow 1$	1	1	1	1
$x_5$	0	0	0	$0 \rightarrow 1$	1	1	1
$x_6$	0	0	0	0	$0 \rightarrow 1$	1	1

Цепочка без расслоения:

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$\dots$	$a_D$
$x_1$	1	$1 \rightarrow 0$	0	0	0	0	0
$x_2$	$0 \rightarrow 1$	1	$1 \rightarrow 0$	0	0	0	0
$x_3$	0	0	$0 \rightarrow 1$	1	$1 \rightarrow 0$	0	0
$x_4$	0	0	0	0	$0 \rightarrow 1$	1	1
$x_5$	0	0	0	0	0	0	0
$x_6$	0	0	0	0	0	0	0

Для каждой цепочки генерируется *не-цепочка* путём случайной перестановки единиц в каждом столбце.

Итого имеем 4 модельных семейства.

Эксперимент: зависимость  $Q_\varepsilon$  от  $D$  при  $\ell = k = 100$ ,  $\varepsilon = 0.05$ Простая задача,  $n(a_0, \mathbb{X}) = 10$ Трудная задача,  $n(a_0, \mathbb{X}) = 50$ 

## Выводы

- Связность приводит к замедлению роста  $Q_\varepsilon(D)$ .
- Расслоение понижает уровень горизонтальной асимптоты.

## Эксперимент с монотонной цепочкой алгоритмов

### Основные выводы

- Монотонная цепочка почти не переобучается, причём лишь нижние 5–6 алгоритмов дают вклад в переобучение.
- Без расслоения или без связности переобучение ( $Q_\varepsilon = \frac{1}{2}$ ) наступает при  $|A|$  порядка нескольких десятков.
- Поэтому «хорошие» семейства обязаны быть и *расслоенными*, и *связными* (или обладать какой-либо иной структурой сходства алгоритмов).

## Комбинаторные оценки вероятности переобучения

- 1 Один алгоритм,  $A = \{a\}$  (точная оценка);
- 2 Два алгоритма,  $A = \{a_1, a_2\}$  (точная оценка);
- 3 Метод порождающих и запрещающих множеств (теоремы для получения точных оценок);
- 4 Монотонная цепочка алгоритмов (точная оценка);
- 5 Рекуррентный метод (алгоритм вычисления сходящихся верхних и нижних оценок);
- 6 Оценка через граф расслоения и связности (слабо завышенная верхняя оценка).



## Один алгоритм — аналог закона больших чисел

Пусть  $|A| = 1$ ,  $\mu X = a$  для всех  $X \subset \mathbb{X}$ .

Обозначим  $m = n(a, \mathbb{X})$ ,  $s = n(a, X)$ .

### Теорема (точная оценка)

Вероятность большого уклонения частот описывается функцией **гипергеометрического распределения** (ГГР):

$$Q_\varepsilon = H_L^{\ell, m} \left( \frac{\ell}{L} (m - \varepsilon k) \right),$$

где  $H_L^{\ell, m}(z) = \sum_{s=0}^{\lfloor z \rfloor} \frac{C_m^s C_{L-m}^{\ell-s}}{C_L^\ell}$  — левый «хвост» ГГР.

**Вывод:** основная аксиома обеспечивает возможность предсказания скрытого  $n(a, \bar{X})$  по наблюдаемому  $n(a, X)$ .

## Двухэлементное множество алгоритмов

Пусть алгоритмы  $a_1$ ,  $a_2$  допускают  $m_1$ ,  $m_2$  ошибок на  $X^L$ :

$$\begin{aligned}
 a_1 &= (\overbrace{1111111}^{m_1} 000000000000000000); \\
 a_2 &= (000 \overbrace{111111111111}^{m_2} 000000000000).
 \end{aligned}$$

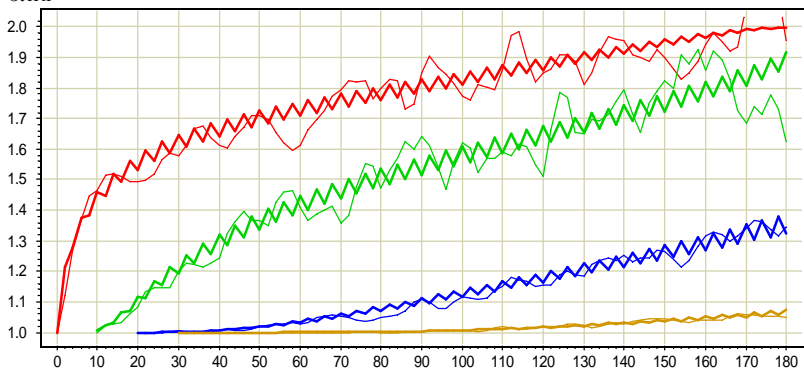
### Теорема (точная оценка вероятности переобучения)

$$\begin{aligned}
 Q_\varepsilon &= \sum_{s_0=0}^{m_0} \sum_{s_1=0}^{m_1} \sum_{s_2=0}^{m_2} \frac{C_{m_0}^{s_0} C_{m_1}^{s_1} C_{m_2}^{s_2} C_{L-m_0-m_1-m_2}^{\ell-s_0-s_1-s_2}}{C_L^\ell} \times \\
 &\quad \times \left( [s_1 < s_2] [s_0 + s_1 \leq \frac{\ell}{L}(m_0 + m_1 - \varepsilon k)] + \right. \\
 &\quad \left. + [s_1 \geq s_2] [s_0 + s_2 \leq \frac{\ell}{L}(m_0 + m_2 - \varepsilon k)] \right).
 \end{aligned}$$

# Эффекты сходства и расслоения для пары алгоритмов

$\ell = k = 100$ ;  $\varepsilon = 0.05$ ;  $m_0 = 20$ ;  $d \equiv m_2 - m_1 = 0, 10, 20, 30$

ЭЛКР



хэммингово расстояние между алгоритмами

— d=0    — d=10    — d=20    — d=30

## Двухэлементное множество алгоритмов

### Выводы

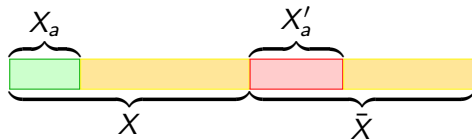
- Оптимизация при неполной информации ведёт к переобучению, даже если вариантов выбора только два.
- Если варианты почти одинаковы, то переобучения почти нет (эффект сходства).
- Если один из вариантов явно хуже, то переобучения почти нет (эффект расслоения).

# Гипотеза о порождающих и запрещающих объектах

## Гипотеза (1)

Для каждого  $a \in A$  можно указать пару непересекающихся подмножеств объектов  $X_a \subset \mathbb{X}$ ,  $X'_a \subset \mathbb{X}$  такую, что:

$$(\mu X = a) \Leftrightarrow (X_a \subseteq X) \text{ и } (X'_a \subseteq \bar{X}), \quad \forall X \subset \mathbb{X}.$$



Опр.  $X_a$  — множество объектов, **порождающих** алгоритм  $a$ .

Опр.  $X'_a$  — множество объектов, **запрещающих** алгоритм  $a$ .

Опр.  $\mathbb{X} \setminus (X_a \cup X'_a)$  — множество объектов, **нейтральных** для  $a$ .

## Обозначения и основная лемма

Введём для каждого  $a \in A$  следующие обозначения:

$L_a = L - |X_a| - |X'_a|$  — число нейтральных объектов;

$\ell_a = \ell - |X_a|$  — число нейтральных обучающих объектов;

### Лемма (о вероятности получения алгоритма)

Если гипотеза (1) справедлива, то вероятность получить в результате обучения алгоритм  $a$  равна *доле разбиений, при которых объекты из  $X_a$  и  $X'_a$  остаются на своих местах*:

$$P_a = P[\mu X = a] = \frac{C_{L_a}^{\ell_a}}{C_L^{\ell}}.$$

## Ещё обозначения и основная теорема

$$m_a = n(a, \mathbb{X}) - n(a, X_a) - n(a, X'_a)$$

— число ошибок алгоритма  $a$  на нейтральных объектах;

$$s_a(\varepsilon) = \frac{\ell}{L} (n(a, \mathbb{X}) - \varepsilon k) - n(a, X_a)$$

— наибольшее число ошибок переобученного алгоритма  $a$  на нейтральных обучающих объектах  $X \setminus X_a$ .

### Теорема (точная оценка вероятности переобучения)

Если гипотеза (1) справедлива, то

$$P[\delta_\mu(X) \geq \varepsilon] = \sum_{a \in A} P_a H_{L_a}^{\ell_a, m_a}(s_a(\varepsilon)).$$

## Обобщение основной гипотезы

### Гипотеза (2)

Для каждого  $a \in A$  можно указать такой набор пар непересекающихся подмножеств объектов  $X_{av}, X'_{av} \subset \mathbb{X}$ ,  $v \in V_a$  и такой коэффициент  $c_{av} \in \mathbb{R}$ , что

$$[\mu X = a] = \sum_{v \in V_a} c_{av} [X_{av} \subseteq X] [X'_{av} \subseteq \bar{X}].$$

Обозначения: для каждого  $a \in A$  и каждого  $v \in V_a$

$$L_{av} = L - |X_{av}| - |X'_{av}|;$$

$$l_{av} = l - |X_{av}|;$$

$$m_{av} = n(a, \mathbb{X}) - n(a, X_{av}) - n(a, X'_{av});$$

$$s_{av}(\varepsilon) = \frac{l}{L} (n(a, \mathbb{X}) - \varepsilon k) - n(a, X_{av}).$$



## Обобщение: лемма и основная теорема

### Лемма (о вероятностях получения алгоритмов)

Если гипотеза (2) справедлива, то вероятность получить в результате обучения алгоритм с вектором ошибок  $a$

$$P[\mu X=a] = \sum_{v \in V_a} c_{av} P_{av}; \quad P_{av} = \frac{C_{L_{av}}^{\ell_{av}}}{C_L^{\ell}}.$$

### Теорема (точная оценка вероятности переобучения)

Если гипотеза (2) справедлива, то

$$Q_\varepsilon = \sum_{a \in A} \sum_{v \in V_a} c_{av} P_{av} H_{L_{av}}^{\ell_{av}, m_{av}}(s_{av}(\varepsilon)).$$

## Сильное ли ограничение накладывает гипотеза (2)?

Оказывается, почти не накладывает. **Это общий случай!**

### Теорема

*Пусть векторы ошибок алгоритмов  $a_1, \dots, a_D$  попарно различны и метод  $\mu$  минимизирует эмпирический риск.*

*Тогда справедлива гипотеза (2), причём  $c_{av} = 1$ .*

Доказательство конструктивно, но «тавтологично» — строится система подмножеств  $(X_{av}, X'_{av})_{v \in V_a} \equiv (X, \bar{X})_{\mu X=a}$ , что приводит к вычислительно неэффективным оценкам.

**В общем случае система подмножеств не единственна.**

### Открытая проблема

Как искать системы подмножеств с наименьшими  $|X_{av}|$ ,  $|X'_{av}|$ ?

# Монотонная цепочка алгоритмов

	$a_0$	$a_1$	$a_2$	$a_3$	$\dots$	$a_D$
$x_1$	0	1	1	1	1	1
$x_2$	0	0	1	1	1	1
$x_3$	0	0	0	1	1	1
$x_4$	0	0	0	0	1	1
$\dots$	0	0	0	0	0	1
$\dots$	0	0	0	0	0	0
$\dots$	0	0	0	0	0	0
$\dots$	1	1	1	1	1	1
$x_L$	1	1	1	1	1	1

$(\mu X = a_d) \Leftrightarrow (x_{d+1} \in X) \text{ и } (x_1, \dots, x_d \in \bar{X}), \text{ при } d \leq k;$

$(\mu X = a_d)$  невозможно, при  $d > k.$

Таким образом, справедлива Гипотеза (1).

## Вероятность переобучения монотонной цепочки

Пусть  $\mu$  — пессимистичная минимизация эмпирического риска (выбор алгоритма по принципу «худший из лучших»):

$$A(X) = \text{Arg min}_{a \in A} n(a, X); \quad \mu X = \arg \max_{a \in A(X)} n(a, \bar{X}).$$

### Теорема (точная оценка вероятности переобучения)

Пусть  $a_0, a_1, \dots, a_D$  — монотонная цепочка,  $n(a_0, \mathbb{X}) = m$ ,  $k \leq D \leq L - m$ . Тогда

$$P_d = P[\mu X = a_d] = \frac{C_{L-d-1}^{\ell-1}}{C_L^\ell};$$

$$Q_\varepsilon(\mu, \mathbb{X}) = \sum_{d=0}^k P_d H_{L-d-1}^{\ell-1, m} \left( \frac{\ell}{L} (m + d - \varepsilon k) \right).$$

## Другие модельные семейства алгоритмов

Точные оценки, полученные в данной работе:

- единичная окрестность лучшего алгоритма;
- унимодальная цепочка алгоритмов;
- слой булева куба;
- интервал булева куба;
- $d$  нижних слоёв интервала булева куба.

Оценки, полученные другими авторами:

- связные семейства [Д. Кочедыков, И. Решетняк].
- монотонные и унимодальные  $h$ -мерные сетки [П. Ботов];
- симметричные семейства алгоритмов [А. Фрей];
- пучок монотонных цепочек [А. Фрей];
- хэммингов шар, слои хэммингова шара [И. Толстихин];

## Постановка задачи рекуррентного вычисления $Q_\varepsilon$

$\mathfrak{I}(a) = \langle X_{av}, X'_{av}, c_{av} \rangle_{v \in V_a}$  — информация об алгоритме  $a \in A$ , необходимая для вычисления вероятности переобучения  $Q_\varepsilon$ .

Расслоение:  $n(a_0, \mathbb{X}) \leq n(a_1, \mathbb{X}) \leq \dots \leq n(a_D, \mathbb{X})$ .

Дополнительное предположение:  $n(a_0, \mathbb{X}) = 0$ .

Пусть  $\mu_d$  — пессимистичный метод обучения, выбирающий алгоритмы только из подмножества  $A_d = \{a_0, \dots, a_d\}$ .

**Задача (пересчёт  $Q_\varepsilon$  при добавлении алгоритма  $a_d$ )**

*Известна информация  $\mathfrak{I}(a_t)$  относительно метода  $\mu_{d-1}$  для всех алгоритмов  $a_t$ ,  $t \leq d-1$ .*

*Вычислить информацию  $\mathfrak{I}(a_t)$  относительно метода  $\mu_d$  для всех алгоритмов  $a_t$ ,  $t \leq d$ .*

## Теоремы о рекуррентном вычислении $Q_\varepsilon$

### Теорема (о добавляемом алгоритме $a_d$ )

Порождающее множество:  $X_d = \emptyset$ .

Запрещающее множество:  $X'_d = \{x_i \in \mathbb{X} : I(a_d, x_i) = 1\}$ .

### Теорема (о всех предыдущих алгоритмах $a_t, t < d$ )

Для каждого  $v \in V_t$  такого, что  $X_{tv} \cap X'_d = \emptyset$

- 1) если  $X'_d \setminus X'_{tv} = \{x_i\}$  — одноэлементное множество, то присоединить  $x_i$  к  $X_{tv}$ ;
- 2) если  $|X'_d \setminus X'_{tv}| > 1$ , то добавить в  $V_t$  индекс  $w$ , положив  $c_{tw} = -c_{tv}$ ,  $X_{tw} = X_{tv}$ ,  $X'_{tw} = X'_{tv} \cup X'_d$ ;
- 3) если  $|X'_d \setminus X'_{tv}| = 0$ , то удалить из  $V_t$  индекс  $v$ .

## Упрощённое рекуррентное вычисление $Q_\varepsilon$

### Теорема (о верхних и нижних оценках)

Если иногда пропускать шаг 2) при  $c_{tv} = 1$ ,  
то вычисляемое значение  $Q_\varepsilon$  может только увеличиться.

Если иногда пропускать шаг 2) при  $c_{tv} = -1$ ,  
то вычисляемое значение  $Q_\varepsilon$  может только уменьшиться.

### Теорема (об упрощённом рекуррентном вычислении $Q_\varepsilon$ )

Если всегда пропускать шаг 2), то шаг 3) не будет выполняться  
никогда, и будет получена верхняя оценка  $Q_\varepsilon$ .

Рекуррентное вычисление  $Q_\varepsilon$  может занять время  $O(L2^D)$ .

Упрощённое рекуррентное вычисление  $Q_\varepsilon$  занимает  $O(LD^2)$ .

Его можно сократить до  $O(LD)$  и даже до  $O(L)$ .

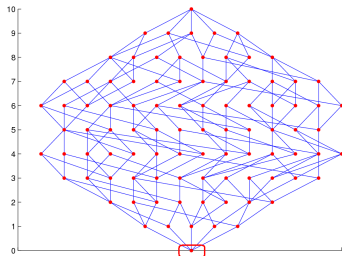
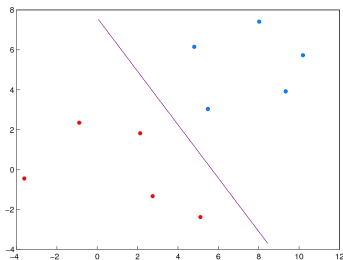


## Расслоение и связность

Расслоение множества алгоритмов  $A = A_0 \sqcup \dots \sqcup A_L$ , где  $A_m = \{a \in A: n(a, \mathbb{X}) = m\}$  —  $m$ -й слой множества алгоритмов.

Связность  $q(a)$  алгоритма  $a \in A$  — число алгоритмов  $a' \in A$  в следующем слое таких, что  $I(a, x) \leq I(a', x)$ ,  $\forall x \in \mathbb{X}$ .

Пример графа расслоения и связности:



## Оценка $Q_\varepsilon$ через профиль расслоения–связности

**Опр.** Профиль расслоения–связности  $\Delta_{mq}$  — это число алгоритмов в  $m$ -м слое со связностью  $q$ .

### Теорема

Пусть векторы ошибок всех алгоритмов  $a \in A$  попарно различны, и в  $A$  есть корректный на  $\mathbb{X}$  алгоритм.

Тогда справедлива верхняя оценка вероятности переобучения

$$Q_\varepsilon \leq \sum_{m=\lceil \varepsilon k \rceil}^L \sum_{q=0}^{\ell} \Delta_{mq} \frac{C_{L-m-q}^{\ell-q}}{C_L^\ell}.$$

# Оценка $Q_\varepsilon$ через профиль расслоения и профиль связности

## Теорема

Пусть справедливы условия предыдущей теоремы и профиль расслоения–связности сепарабелен:

$$\Delta_{mq} \leq \Delta_m \lambda_q.$$

Тогда справедлива верхняя оценка вероятности переобучения

$$Q_\varepsilon \leq \underbrace{\sum_{m=\lceil \varepsilon k \rceil}^k \Delta_m \frac{C_{L-m}^\ell}{C_L^\ell}}_{VC\text{-оценка}} \underbrace{\sum_{q=0}^L \lambda_q \left( \frac{\ell}{L-m} \right)^q}_{\text{поправка на связность}}.$$

При известных  $\Delta_m$ ,  $\lambda_q$  вычисления  $Q_\varepsilon$  займут  $O(L)$ .

## Эксперименты и выводы

В экспериментах с линейными классификаторами:

- **средняя связность = размерности пространства** (с очень высокой точностью);
- гипотеза сепарабельности выполнялась (с достаточной точностью);

### Выводы

- **Учёт расслоения и связности существенно уточняет оценку (экспоненциально по размерности пространства).**
- Оценка зависит не от одной скалярной характеристики сложности, а от «профиля», в отличие от VC-оценок.
- Как использовать эту оценку на практике? (пока открытый вопрос)

## Функционал полного скользящего контроля

Выше рассматривалась только *вероятность переобучения*

$$Q_\varepsilon(\mu, \mathbb{X}) = \frac{1}{C_L^\ell} \sum_{(X, \bar{X})} [\nu(\mu X, \bar{X}) - \nu(\mu X, X) \geq \varepsilon].$$

В работе все результаты перенесены также на функционал

$$R_\varepsilon(\mu, \mathbb{X}) = \frac{1}{C_L^\ell} \sum_{(X, \bar{X})} [\nu(\mu X, \bar{X}) \geq \varepsilon].$$

Однако для функционала *полного скользящего контроля*

$$CCV(\mu, \mathbb{X}) = \frac{1}{C_L^\ell} \sum_{(X, \bar{X})} \nu(\mu X, \bar{X}).$$

техника получения оценок совсем другая.

**Недостаток CCV:** не учитывается дисперсия частоты ошибок.

## Метод ближайшего соседа

Пусть  $\rho(x, x')$  — функция расстояния на множестве  $\mathbb{X}$ .

$$a(x; X) = y(\arg \min_{x' \in X} \rho(x, x')).$$

### Определение (профиль компактности выборки $\mathbb{X}$ )

доля объектов, у которых  $m$ -й сосед  $x_{im}$  лежит в другом классе:

$$K(m, \mathbb{X}) = \frac{1}{L} \sum_{i=1}^L [y(x_i) \neq y(x_{im})]; \quad m = 1, \dots, L-1,$$

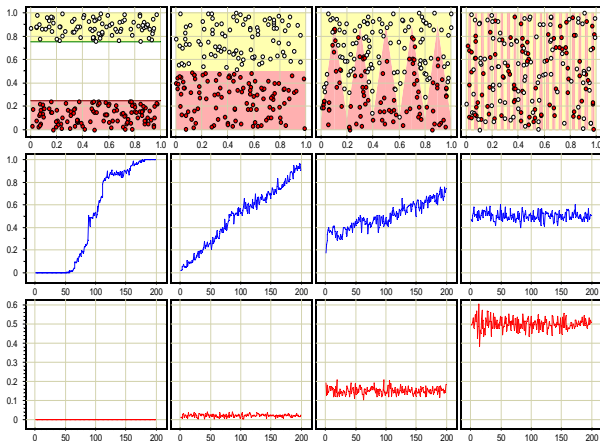
### Теорема (точная оценка для метода ближайшего соседа)

$$\text{CCV}(\mu, \mathbb{X}) = \sum_{m=1}^k K(m, \mathbb{X}) \frac{C_{L-1-m}^{\ell-1}}{C_{L-1}^{\ell}}.$$

## Профили компактности для серии модельных задач

средний ряд: профили компактности,

нижний ряд: зависимость  $CCV$  от длины контроля  $k = |\bar{X}|$ .



## Свойства профиля компактности и оценки CCV

### Выводы

- Полученная оценка CCV является *точной* (не завышенной, не асимптотической).
- CCV практически не зависит от длины контроля  $k$  (всегда ли? — открытый вопрос).
- Для минимизации CCV важен только начальный участок профиля, т. к.  $\frac{C_{L-1-m}^{\ell-1}}{C_{L-1}^{\ell}} \rightarrow 0$  экспоненциально по  $m$ .
- Минимизация CCV приводит к эффективному отбору эталонных объектов, без переобучения [М. Иванов, 2009].

**Замечание.** VC-теория вообще не даёт содержательных оценок для метода ближайшего соседа, т.к. ёмкость данного семейства алгоритмов бесконечна.



## Монотонные алгоритмы классификации: определения

Задача классификации:  $\mathbb{X}$  — ч. у. множество,  $Y = \{0, 1\}$ ,  
 $A$  — множество монотонных отображений  $a: \mathbb{X} \rightarrow Y$ .

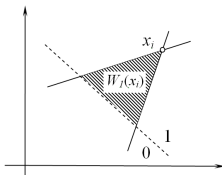
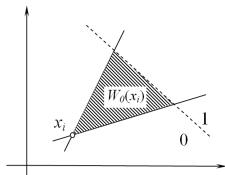
**Опр.** Степень немонотонности выборки  $\mathbb{X}$ :

$$\theta(\mathbb{X}) = \min_{a \in A} \nu(a, \mathbb{X}).$$

**Опр.** Верхний и нижний клин объекта  $x_i \in \mathbb{X}$ :

$$W_0(x_i) = \{x \in \mathbb{X} : x_i < x \text{ и } y(x) = 0\};$$

$$W_1(x_i) = \{x \in \mathbb{X} : x < x_i \text{ и } y(x) = 1\}.$$



## Профиль монотонности выборки

### Определение (профиль монотонности выборки $\mathbb{X}$ )

доля объектов  $x_i \in \mathbb{X}$  с клином мощности  $m$ :

$$M(m, \mathbb{X}) = \frac{1}{L} \sum_{i=1}^L [ |W_{y(x_i)}(x_i)| = m ]; \quad m = 0, \dots, L-1.$$

### Теорема

Пусть  $\mu$  — метод минимизации эмпирического риска в классе всех монотонных функций,  $\theta$  — степень немонотонности выборки  $\mathbb{X}$ . Тогда

$$\text{CCV}(\mu, \mathbb{X}) \leq \sum_{m=0}^{\theta L + k - 1} M(m, \mathbb{X}) H_{L-1}^{\ell, m}(\theta L).$$

## Свойства профиля монотонности и оценки $CCV$

### Выводы

- Невырожденность:  $CCV(\mu, \mathbb{X}) \leq 1$ .
- Для минимизации  $CCV$  важен только начальный участок профиля, т. к.  $H_{L-1}^{\ell, m}(\theta L) \rightarrow 0$  по  $m$  при малых  $\theta$ .
- Для минимизации  $CCV$  отношение порядка на множестве объектов  $\mathbb{X}$  должно быть близко к линейному вблизи границы классов.
- Минимизация  $CCV$  приводит к повышению обобщающей способности алгоритмической композиции с монотонной корректирующей операцией [И. Гуз, 2008].

**Замечание.**  $VC$ -теория даёт сильно завышенные оценки для монотонных семейств алгоритмов (эффективная ёмкость определяется максимальной длиной антицепи).

## Основные публикации

- 1 Рудаков К. В., Воронцов К. В. О методах оптимизации и монотонной коррекции в алгебраическом подходе к проблеме распознавания // **Доклады РАН**. — 1999. — Т. 367, № 3. — С. 314–317.
- 2 Воронцов К. В. Оптимизационные методы линейной и монотонной коррекции в алгебраическом подходе к проблеме распознавания // **ЖВМ и МФ**. — 2000. — Т. 40, № 1. — С. 166–176.
- 3 Воронцов К. В. Комбинаторный подход к оценке качества обучаемых алгоритмов // **Математические вопросы кибернетики** / Под ред. О. Б. Лупанова. — М.: Физматлит, 2004. — Т. 13. — С. 5–36.
- 4 Воронцов К. В. Комбинаторные обоснования обучаемых алгоритмов // **ЖВМ и МФ**. — 2004. — Т. 44, № 11. — С. 2099–2112.
- 5 Воронцов К. В. Комбинаторные оценки качества обучения по прецедентам // **Доклады РАН**. — 2004. — Т. 394, № 2. — С. 175–178.
- 6 Vorontsov K. V. Combinatorial probability and the tightness of generalization bounds // **Pattern Recognition and Image Analysis**. — 2008. — Vol. 18, no. 2. — Pp. 243–259.
- 7 Vorontsov K. V. Splitting and similarity phenomena in the sets of classifiers and their effect on the probability of overfitting // **Pattern Recognition and Image Analysis**. — 2009. — Vol. 19, no. 3. — Pp. 412–420.
- 8 Воронцов К. В. Точные оценки вероятности переобучения // **Доклады РАН**. — 2009. — Т. 429, № 1. — С. 15–18.

## Основные результаты, выносимые на защиту

- 1 Метод измерения факторов завышенности VC-оценок.
- 2 Методы получения точных оценок вероятности переобучения:
  - метод порождающих и запрещающих множеств;
  - рекуррентный метод.
- 3 Точные оценки вероятности переобучения для семи модельных семейств алгоритмов.
- 4 Верхние оценки вероятности переобучения через профиль расслоения и связности.
- 5 Точные оценки полного скользящего контроля:
  - для множества алгоритмов ближайшего соседа;
  - для множества монотонных алгоритмов.