

Просеминар кафедры ММП, ВМК МГУ

.....

# Комбинаторная теория переобучения и её применения

Воронцов Константин Вячеславович

27 февраля 2012

- 1 Обучение и переобучение**
  - Задачи обучения по прецедентам
  - Методы обучения по прецедентам
  - Явление переобучения
- 2 Комбинаторная теория переобучения**
  - Основные понятия и классические оценки
  - Эксперименты
  - Комбинаторные оценки переобучения
- 3 Результаты, открытые проблемы, планы**
  - Логические алгоритмы классификации
  - Отбор эталонов в методе ближайшего соседа
  - Открытые проблемы и планы

## Основные определения и обозначения

$\mathbb{X}$  — объекты;  $\mathbb{Y}$  — ответы (классы);

$y^*: \mathbb{X} \rightarrow \mathbb{Y}$  — неизвестная зависимость.

**Дано:**  $x_i = (x_i^1, \dots, x_i^n)$  — обучающие объекты с известными ответами  $y_i = y^*(x)$ ,  $i = 1, \dots, \ell$ :

$$\begin{pmatrix} x_1^1 & \dots & x_1^n \\ \dots & \dots & \dots \\ x_\ell^1 & \dots & x_\ell^n \end{pmatrix} \xrightarrow{y^*} \begin{pmatrix} y_1 \\ \dots \\ y_\ell \end{pmatrix}$$

**Найти:** алгоритм  $a: \mathbb{X} \rightarrow \mathbb{Y}$ , способный давать правильные ответы на новых объектах  $\tilde{x}_i = (\tilde{x}_i^1, \dots, \tilde{x}_i^n)$ ,  $i = 1, \dots, k$ :

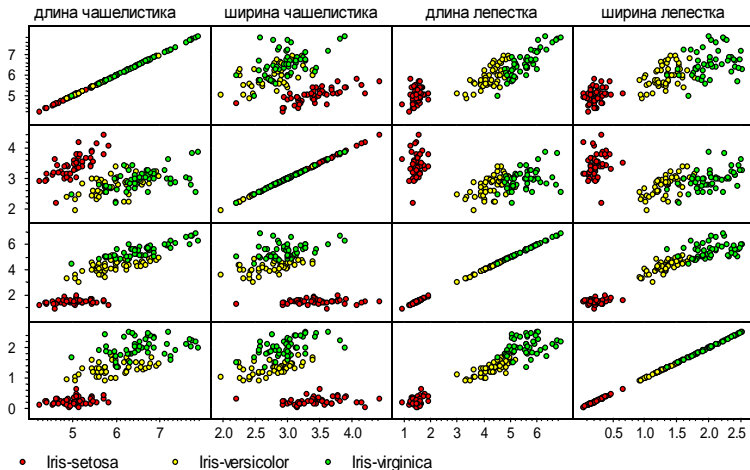
$$\begin{pmatrix} \tilde{x}_1^1 & \dots & \tilde{x}_1^n \\ \dots & \dots & \dots \\ \tilde{x}_k^1 & \dots & \tilde{x}_k^n \end{pmatrix} \xrightarrow{a?} \begin{pmatrix} a(\tilde{x}_1) \\ \dots \\ a(\tilde{x}_k) \end{pmatrix}$$

## Примеры прикладных задач обучения по прецедентам

- Распознавание, классификация, принятие решений ( $|\mathbb{Y}| < \infty$ ):
  - $x$  — пациент;  $y$  — долгосрочный исход операции;
  - $x$  — заёмщик;  $y$  — кредит выдать / не выдать;
  - $x$  — курсы акций;  $y$  — купить / продать.
  - $x$  — абонент;  $y$  — уйдёт / не уйдёт к другому оператору;
  - $x$  — фотопортрет;  $y$  — идентификатор личности;
  - $x$  — фрагмент ДНК;  $y$  — функция: промотор / ген;
  - $x$  — фрагмент белка;  $y$  — тип вторичной структуры;
  - $x$  — текстовое сообщение;  $y$  — спам / не спам;
- Регрессия и прогнозирование ( $\mathbb{Y} = \mathbb{R}$  или  $\mathbb{R}^m$ ):
  - $x$  — структура химического соединения;  $y$  — его свойство;
  - $x$  — параметры технолог. процесса;  $y$  — свойство продукции;
  - $x$  — история продаж;  $y$  — прогноз потребительского спроса;
  - $x$  — данные о недвижимости;  $y$  — продажная стоимость;
  - $x$  — пара (клиент, товар);  $y$  — рейтинг товара.

## Пример: задача классификации цветков ириса [Фишер, 1936]

$n = 4$  признака,  $|\mathbb{Y}| = 3$  класса, длина выборки  $\ell = 150$ .



*Модель алгоритмов* — параметрическое семейство отображений

$$A = \{g(x, \theta) \mid \theta \in \Theta\},$$

где  $g: \mathbb{X} \times \Theta \rightarrow \mathbb{Y}$  — фиксированная функция,  
 $\Theta$  — множество допустимых значений параметра  $\theta$ .

*Метод обучения* (learning algorithm)  $\mu$  по произвольной выборке  $X = (x_i, y_i)_{i=1}^{\ell}$  строит алгоритм  $\mu(X)$  из  $A$ :

$$\mu: (\mathbb{X} \times \mathbb{Y})^{\ell} \rightarrow A.$$

В задачах обучения по прецедентам всегда есть два этапа:

- *Этап обучения*:  $a = \mu(X)$ .
- *Этап применения*:  $\tilde{y}_i = a(\tilde{x}_i)$ ,  $i = 1, \dots, k$ .

## Примеры методов обучения

из курса «Математические методы распознавания образов»:

- Байесовский классификатор
- Метод ближайших соседей
- Метод потенциальных функций
- Метод опорных векторов
- Логистическая регрессия
- Многослойная нейронная сеть
- Сеть радиальных базисных функций
- Бустинг
- Решающее дерево
- Алгоритм вычисления оценок
- ... ..

## Принцип минимизации эмпирического риска

*Эмпирический риск* — частота ошибок алгоритма  $a$  на  $X$ :

$$Q(a, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} [a(x_i) \neq y_i].$$

*Метод минимизации эмпирического риска:*

$$\mu(X) = \arg \min_{a \in A} Q(a, X).$$

**Проблема обобщающей способности:**

- будет ли алгоритм  $a = \mu(X)$  приближать  $y^*$  на всём  $\mathbb{X}$ ?
- найдём ли мы «закон природы» или *переобучимся*, т. е. подгоним функцию  $g(x, \theta)$  под заданные точки  $(x_i, y_i)$ ?
- будет ли  $Q(a, \bar{X})$  мало́ на новых данных — *контрольной выборке*  $\bar{X} = (\tilde{x}_i, \tilde{y}_i)_{i=1}^k$ ,  $\tilde{y}_i = y^*(\tilde{x}_i)$ ?



## Пример переобучения. Модельная задача регрессии

Зависимость  $y^*(x) = \frac{1}{1 + 25x^2}$  на отрезке  $x \in [-2, 2]$ .

Признаковое описание  $x \mapsto (1, x^1, x^2, \dots, x^n)$ .

Алгоритм полиномиальной регрессии

$$a(x, \theta) = \theta_0 + \theta_1 x + \dots + \theta_n x^n \quad \text{— полином степени } n.$$

Обучение методом наименьших квадратов:

$$Q(\theta, X) = \sum_{i=1}^{\ell} (\theta_0 + \theta_1 x_i + \dots + \theta_n x_i^n - y_i)^2 \rightarrow \min_{\theta_0, \dots, \theta_n}.$$

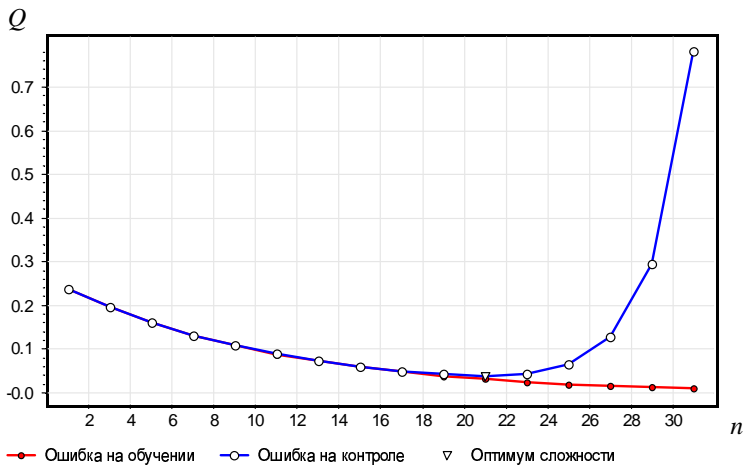
Обучающая выборка:  $X = \{x_i = 4 \frac{i-1}{\ell-1} - 2 \mid i = 1, \dots, \ell\}$ .

Контрольная выборка:  $\bar{X} = \{\tilde{x}_i = 4 \frac{i-0.5}{\ell-1} - 2 \mid i = 1, \dots, \ell - 1\}$ .

Что происходит с  $Q(\mu(X), X)$  и  $Q(\mu(X), \bar{X})$  при увеличении  $n$ ?

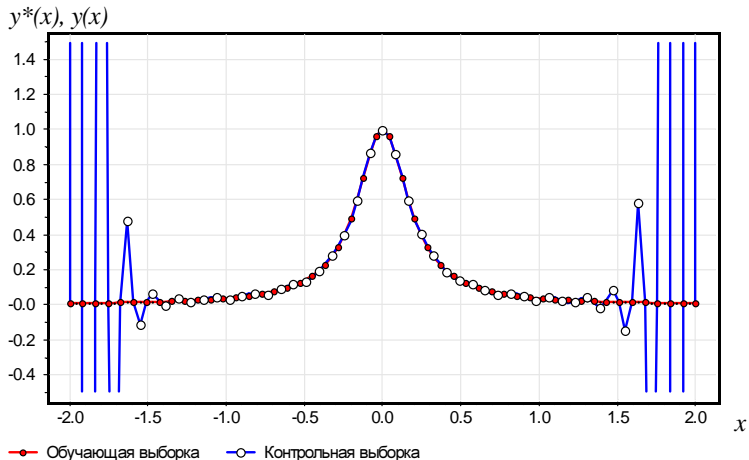
Пример переобучения: эксперимент при  $\ell = 50$ ,  $n = 1..31$ 

Переобучение — это когда  $Q(\mu(X), \bar{X}) \gg Q(\mu(X), X)$ :



Пример переобучения: эксперимент при  $\ell = 50$ 

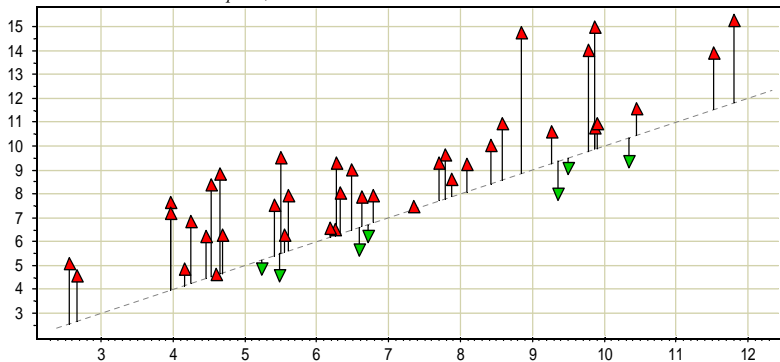
Переобучение, «вид изнутри»: что происходит с полиномами слишком высоких степеней (в данном случае  $n = 40$ )



## Пример переобучения. Реальная задача классификации

Задача предсказания отдалённого результата хирургического лечения атеросклероза,  $L = 98$ . Точки — различные алгоритмы.

*Частота ошибок на контроле, %*



*Частота ошибок на обучении, %*

## Матрица ошибок

$\mathbb{X} = \{x_1, \dots, x_L\}$  — конечное *генеральное 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 $l$
$\dots$	0	0	0	0	1	1	$\dots$	1	
$x_l$	0	0	1	0	0	0	$\dots$	0	
$x_{l+1}$	0	0	0	1	1	1	$\dots$	0	$\bar{X}$ — скрытая (контрольная) выборка длины $k = L - l$
$\dots$	0	0	0	1	0	0	$\dots$	1	
$x_L$	0	1	1	1	1	1	$\dots$	0	

$n(a, X) = \sum_{x \in X} I(a, x)$  — число ошибок  $a \in A$  на выборке  $X \subset \mathbb{X}$ ;

$\nu(a, X) = \frac{1}{|X|} n(a, X)$  — частота ошибок  $a$  на выборке  $X$ ;

## Вероятностные определения обобщающей способности

### Основная вероятностная аксиома

Все разбиения  $X \sqcup \bar{X} = \mathbb{X}$  равновероятны,  $|X| = \ell$ ,  $|\bar{X}| = k$ .

В этом случае  $P \equiv E \equiv \frac{1}{C_L^\ell} \sum_{X \subset \mathbb{X}}$  — доля разбиений выборки.

### Функционалы обобщающей способности

- ожидаемая частота ошибок на контроле:

$$CCV(\mu, \mathbb{X}) = E \nu(\mu(X), \bar{X}).$$

- вероятность большой частоты ошибок на контроле:

$$R_\varepsilon(\mu, \mathbb{X}) = P[\nu(\mu(X), \bar{X}) \geq \varepsilon].$$

- вероятность переобучения:

$$Q_\varepsilon(\mu, \mathbb{X}) = P[\nu(\mu(X), \bar{X}) - \nu(\mu(X), X) \geq \varepsilon].$$

## Теория Вапника–Червоненкиса

### Теорема (Вапник, Червоненкис, 1974)

Для любых  $\mathbb{X}$ ,  $A$ ,  $\mu$  и  $\varepsilon \in [0, 1]$ , при  $\ell = k$

$$Q_\varepsilon(\mu, \mathbb{X}) \leq |A| \cdot \frac{3}{2} \exp(-\varepsilon^2 \ell).$$

**Проблема завышенности:**

- эта оценка завышена в  $10^8$ – $10^{11}$  раз;
- что приводит к оценкам длины обучения  $\ell = 10^6$ – $10^{10}$ , когда на самом деле достаточно  $\ell = 10^2$ – $10^3$ .

**Причина завышенности** — это оценка «худшего случая»:

- она зависит только от размеров матрицы ошибок  $L \times D$ ;
- не зависит от её содержимого  $l(a, x)$ , выборки  $\mathbb{X}$ , метода  $\mu$ .

## Два мысленных эксперимента

1. Пусть в семействе есть один очень хороший алгоритм,  $n(a_0, \mathbb{X}) = 0$ , и много плохих алгоритмов  $a$ :  $n(a, \mathbb{X}) \gg 0$ . Тогда  $a_0$  почти всегда будет лучшим и на обучающей выборке.

Результат: можно полагать  $|A| \approx 1$ .

В общем случае **надо учитывать расслоение семейства  $A$** ,  $A_m = \{a \in A: n(a, \mathbb{X}) = m\}$ , наиболее важны нижние слои.

2. Пусть в семействе есть алгоритм  $a_0$ , и все остальные очень похожи на него. Тогда это «почти один и тот же алгоритм».

Результат: можно полагать  $|A| \approx 1$ .

В общем случае **надо учитывать связность семейства  $A$** , т.е. сколько в  $A$  вместе с каждым  $a$  содержится  $b$ :  $\|a - b\| = 1$ .



## Эксперименты с модельными семействами алгоритмов

*Физика — экспериментальная, естественная наука, часть естествознания. Математика — это та часть физики, в которой эксперименты дешёвы. [В.И.Арнольд]*


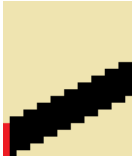


Две идеи организации эксперимента:

- 1 Будем изучать модельные семейства алгоритмов, задавая их непосредственно своими матрицами ошибок.
- 2 Будем оценивать вероятность методом Монте–Карло — как долю разбиений выборки из случайного подмножества  $N$  разбиений,  $|N|$  порядка  $10^3$ – $10^4$ :

$$\hat{Q}_\varepsilon(\mu, \mathbb{X}) = \frac{1}{|N|} \sum_{(\bar{X}, X) \in N} \left[ \nu(\mu(X), \bar{X}) - \nu(\mu(X), X) \geq \varepsilon \right].$$

## Эксперимент с четырьмя модельными семействами

Матрицы ошибок: строки — объекты, столбцы — алгоритмы;  
 лучший алгоритм одинаков во всех четырёх семействах.

	есть расслоение по числу ошибок	нет расслоения по числу ошибок
<b>есть связность</b> , соседние алгоритмы отличаются на одном объекте, образуется <i>цепь</i>		
<b>нет связности</b> , соседние алгоритмы существенно различны, <i>цепь</i> не образуется		

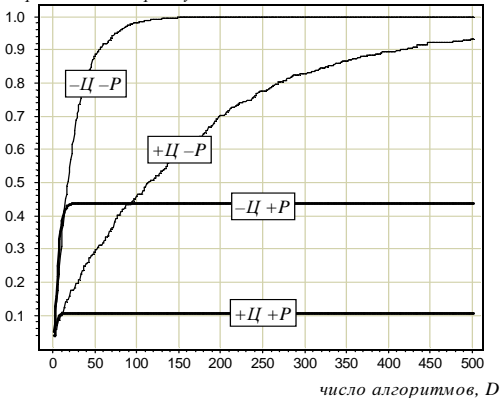
Результаты эксперимента (при  $\ell = k = 100$ ,  $\varepsilon = 0.05$ ,  $|N| = 10^4$ )**Условные обозначения:**

- +Ц — цепь;
- Ц — не цепь;
- +Р — с расслоением;
- Р — без расслоения;

**Связность** замедляет  
темп роста  $Q_\varepsilon(D)$

**Расслоение** понижает  
уровень горизонтальной  
асимптоты  $Q_\varepsilon(D)$

Вероятность переобучения



**Вывод:** получение точных оценок вероятности переобучения невозможно без учёта эффектов расслоения и связности.

## Граф расслоения–связности множества алгоритмов

Определим бинарные отношения на множестве алгоритмов  $A$ :  
 частичный порядок  $a \leq b$ :  $I(a, x) \leq I(b, x)$  для всех  $x \in \mathbb{X}$ ;  
 предшествование  $a \prec b$ :  $a \leq b$  и  $\|b - a\| = 1$ .

## Определение

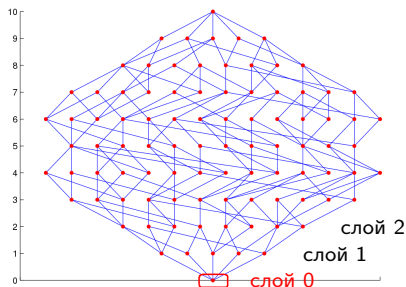
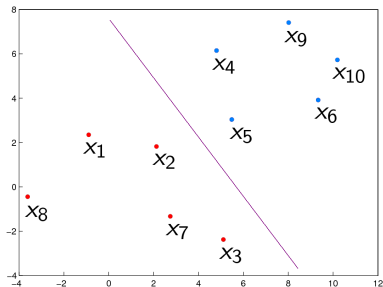
Граф расслоения–связности  $\langle A, E \rangle$ :

$A$  — множество попарно различных векторов ошибок;  
 $E = \{(a, b) : a \prec b\}$ .

Свойства графа расслоения–связности:

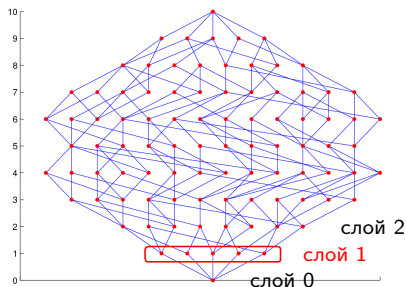
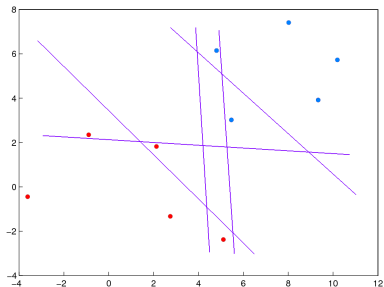
- это подграф графа Хассе отношения порядка  $\leq$  на  $A$ ;
- каждому ребру  $(a, b)$  соответствует объект  $x_{ab} \in \mathbb{X}$ , такой, что  $I(a, x_{ab}) = 0$ ,  $I(b, x_{ab}) = 1$ ;
- граф является многодольным со слоями  
 $A_m = \{a \in A : n(a, \mathbb{X}) = m\}$ ,  $m = 0, \dots, L$ ;

## Пример. Семейство линейных алгоритмов классификации



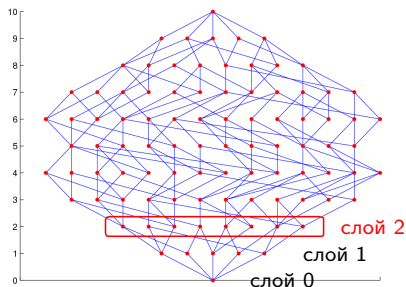
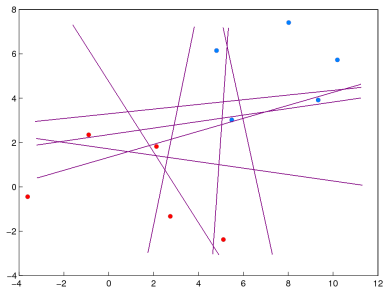
	слой 0
$x_1$	0
$x_2$	0
$x_3$	0
$x_4$	0
$x_5$	0
$x_6$	0
$x_7$	0
$x_8$	0
$x_9$	0
$x_{10}$	0

## Пример. Семейство линейных алгоритмов классификации



	слой 0	слой 1				
$x_1$	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. Семейство линейных алгоритмов классификации



	слой 0	слой 1						слой 2								
$x_1$	0	1	0	0	0	0	0	1	0	0	0	0	1	1	0	...
$x_2$	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	...
$x_3$	0	0	0	1	0	0	0	0	1	1	0	0	0	0	1	...
$x_4$	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	...
$x_5$	0	0	0	0	0	1	0	0	0	0	1	1	1	0	0	...
$x_6$	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	...
$x_7$	0	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	0	...
$x_9$	0	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	0	...

## Характеристики расслоения и связности алгоритма $a \in A$

### Определение

*Порождающее множество* алгоритма  $a$  — множество объектов, соответствующих всем рёбрам, исходящим из  $a$ :

$$X_a = \{x_{ab} \in \mathbb{X} \mid a \prec b\};$$

$u(a) = |X_a|$  называется *верхней связностью* алгоритма  $a$ .

### Определение

*Запрещающее множество* алгоритма  $a$  — множество объектов, соответствующих всем рёбрам на путях, ведущих в  $a$ :

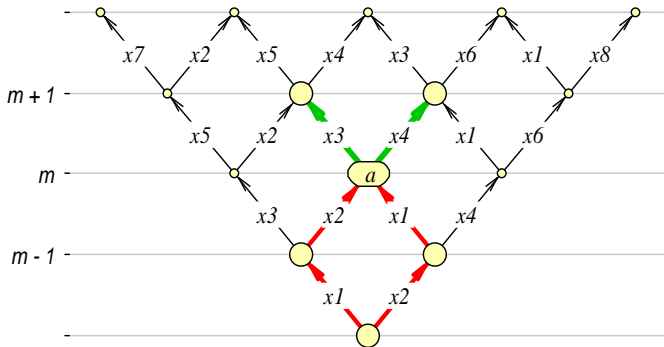
$$X'_a = \{x \in \mathbb{X} \mid \exists b \in A: b \prec a, I(b, x) < I(a, x)\};$$

$q(a) = |X'_a|$  называется *неполноценностью* алгоритма  $a$ .



## Пример: двумерная сеть алгоритмов

Верхняя связность алгоритма  $a$ :  $X_a = \{x3, x4\}$ ,  $u(a) = |X_a| = 2$ ;  
 Неполноценность алгоритма  $a$ :  $X'_a = \{x1, x2\}$ ,  $q(a) = |X'_a| = 2$ ;



Основная лемма: если  $\mu X = a$ , то  $X_a \subseteq X$  и  $X'_a \subseteq \bar{X}$ .

## Монотонные методы обучения

**Опр.** Метод обучения  $\mu$  называется *монотонным*, если

$$\mu(X) \in A(X) = \operatorname{Arg} \min_{a \in A} K(a, X),$$

где  $K(a, X)$  — строго монотонная функция вектора ошибок  $a$ :

$$\forall X \subset \mathbb{X}, \forall a, b \in A \text{ если } a < b, \text{ то } K(a, X) < K(b, X).$$

**Опр.** Метод  $\mu$  называется *пессимистичным*, если

$$\mu(X) = \arg \max_{a \in A(X)} \left( \nu(\mu(X), \bar{X}) - \nu(\mu(X), X) \right).$$

### Основная лемма

Если метод обучения  $\mu$  монотонный и пессимистичный, то

$$[\mu X = a] \leq [X_a \subseteq X] [X'_a \subseteq \bar{X}].$$

## Верхняя оценка вероятности переобучения

## Теорема (Воронцов, Решетняк, Ивахненко, 2010)

Для любого монотонного метода  $\mu$ , любых  $\mathbb{X}$ ,  $A$  и  $\varepsilon \in (0, 1)$

$$Q_\varepsilon(\mu, \mathbb{X}) \leq \sum_{a \in A} \frac{C_{L-u-q}^{\ell-u}}{C_L^\ell} \mathcal{H}_{L-u-q}^{\ell-u, m-q} \left( \frac{\ell}{L} (m - \varepsilon k) \right),$$

где  $u = |X_a|$  — верхняя связность алгоритма  $a$ ,

$q = |X'_a|$  — неполноценность алгоритма  $a$ ,

$m = n(a, \mathbb{X})$  — число ошибок алгоритма  $a$ ,

$$\mathcal{H}_L^{\ell, m}(z) = \sum_{s=0}^{\lfloor z \rfloor} \frac{C_m^s C_{L-m}^{\ell-s}}{C_L^\ell}, \quad z = 0, \dots, \ell$$

— функция гипергеометрического распределения:

Следствие:  $P[\mu X = a] \leq C_{L-u-q}^{\ell-u} / C_L^\ell$ .

## Идея доказательства

1. Пусть  $\mu$  — произвольный монотонный метод обучения,  $\bar{\mu}$  — монотонный пессимистичный метод обучения. Тогда

$$Q_\varepsilon(\mu, \mathbb{X}) \leq Q_\varepsilon(\bar{\mu}, \mathbb{X}).$$

2. Если  $\bar{\mu}(X) = a$ , то  $\begin{cases} X_a \subseteq X \text{ в силу пессимистичности } \bar{\mu}, \\ X'_a \subseteq \bar{X} \text{ в силу монотонности } \bar{\mu}. \end{cases}$

$$3. P[\bar{\mu}(X) = a] \leq P[\underbrace{X_a \subseteq X \text{ и } X'_a \subseteq \bar{X}}_{S(a, X)}] = \frac{C_{L-|X_a|-|X'_a|}^{\ell-|X_a|}}{C_L^\ell} = \frac{C_{L-u-q}^{\ell-u}}{C_L^\ell}.$$

4. По формуле полной вероятности:

$$Q_\varepsilon(\bar{\mu}, \mathbb{X}) = \sum_{a \in A} \underbrace{P[S(a, X)]}_{C_{L-u-q}^{\ell-u} / C_L^\ell} \cdot \underbrace{P[\delta(a, X) \geq \varepsilon \mid S(a, X)]}_{\mathcal{H}_{L-u-q}^{\ell-u, m-q}(\frac{\ell}{L}(m - \varepsilon k))}. \quad \blacksquare$$

## Свойства оценки

$$Q_\varepsilon \leq \sum_{a \in A} \frac{C_{L-u-q}^{\ell-u}}{C_L^\ell} \mathcal{H}_{L-u-q}^{\ell-u, m-q} \left( \frac{\ell}{L} (m - \varepsilon k) \right)$$

- 1 Вклад алгоритма  $a \in A$  убывает экспоненциально по  $u(a) \Rightarrow$  **связные семейства меньше переобучаются;**  
по  $q(a) \Rightarrow$  **только нижние слои вносят вклад в  $Q_\varepsilon$ .**
- 2 При  $q = u = 0$  и  $\ell = k$  это оценка Вапника-Червоненкиса:

$$Q_\varepsilon \leq \sum_{a \in A} \mathcal{H}_L^{\ell, m} \left( \frac{\ell}{L} (m - \varepsilon k) \right) \leq |A| \cdot \frac{3}{2} \exp(-\varepsilon \ell^2).$$

- 3 При  $|A| = 1$  это вариант закона больших чисел (сходимость частот в двух подвыборках):

$$\nu(a, \bar{X}) \xrightarrow{P} \nu(a, X) \text{ при } \ell, k \rightarrow \infty.$$

- 4 Оценка обращается в равенство в случае многомерных монотонных сетей алгоритмов [**Павел Ботов**]
- 5 Получен критерий точности оценки [**Никита Животовский**]

## Верхние оценки средней частоты ошибок на контроле

## Теорема

Для любого монотонного метода  $\mu$ , любых  $\mathbb{X}$  и  $A$

$$\text{CCV}(\mu, \mathbb{X}) \leq \sum_{a \in A} \frac{C_{L-u-q}^{\ell-u}}{C_L^\ell} \left( \frac{m}{k} - \frac{(m-q)(\ell-u)}{k(L-u-q)} \right).$$

где  $u = |X_a|$  — верхняя связность алгоритма  $a$ ,

$q = |X'_a|$  — неполноценность алгоритма  $a$ ,

$m = n(a, \mathbb{X})$  — число ошибок алгоритма  $a$ .

**Преимущество** оценки  $\text{CCV}$  по сравнению с оценками  $Q_\varepsilon$  и  $R_\varepsilon$  — она вычислительно эффективнее, т.к. не нужно вычислять и обращать функцию гипергеометрического распределения.

## Основные результаты в комбинаторной теории переобучения

- 1 Оценки  $Q_\epsilon$  для пороговых логических закономерностей и новые критерии отбора признаков [Андрей Ивахненко]
- 2 Точные оценки для многомерных сетей алгоритмов и новые методы обучения деревьев решений [Павел Ботов]
- 3 Точные оценки  $CCV$  для метода  $k$  ближайших соседей и новые методы отбора эталонных объектов [Максим Иванов, Анастасия Зухба]
- 4 Верхние оценки  $CCV$  для монотонных классификаторов и новые методы обучения композиций классификаторов [Иван Гуз, Галина Махина]
- 5 Оценки  $Q_\epsilon$  для рандомизированных методов обучения и симметричных семейства на основе теории групп [Александр Фрей, Илья Толстихин]

## Понятие логической закономерности

*Закономерность* класса  $y$  — это предикат  $r: \mathbb{X} \rightarrow \{0, 1\}$ , который выделяет ( $r(x) = 1$ ) много объектов класса  $y$ :

$$p(r, X) = \sum_{x_i \in X} r(x_i) [y_i = y] \rightarrow \max,$$

и как можно меньше объектов всех остальных классов:

$$n(r, X) = \sum_{x_i \in X} r(x_i) [y_i \neq y] \rightarrow \min.$$

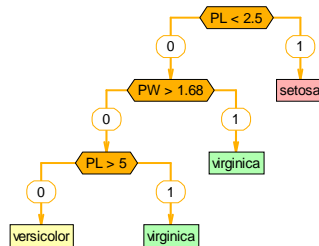
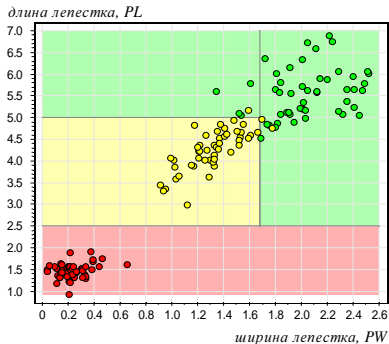
*Логическая закономерность* — конъюнкция пороговых условий:

$$r(x) = \bigwedge_{j \in J} [f_j(x) \leq \theta_j],$$

где  $f_j(x)$  — числовые признаки,  $\theta_j$  — пороги,  $j = 1, \dots, n$ ;  
 $J \subseteq \{1, \dots, n\}$  — подмножество признаков, обычно  $|J| = 1..5$ .



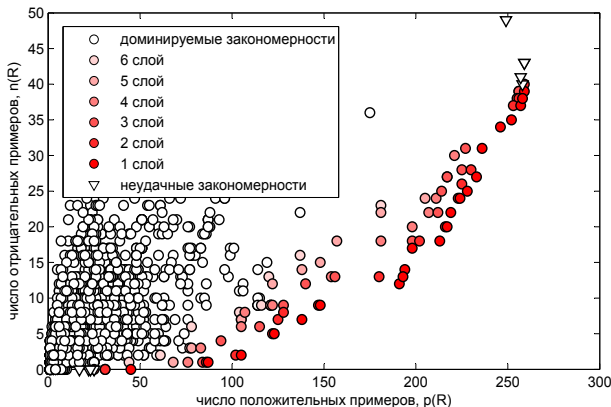
## Пример 1. Закономерности в задаче с ирисами Фишера



setosa	$r_1(x) = [PL \leq 2.5]$
virginica	$r_2(x) = [PL > 2.5] \wedge [PW > 1.68]$
virginica	$r_3(x) = [PL > 5] \wedge [PW \leq 1.68]$
versicolor	$r_4(x) = [PL > 2.5] \wedge [PL \leq 5] \wedge [PW < 1.68]$

## Пример 2. Для классификатора нужно много закономерностей

**Парето-фронт** — множество недоминируемых закономерностей (точка  $(p, n)$  недоминируема, если правее и ниже точек нет)

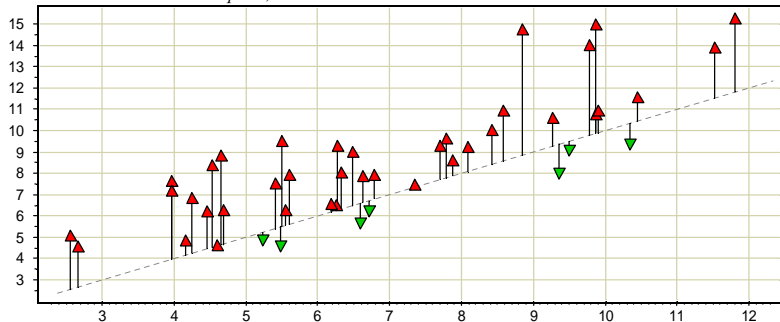


Задача кредитного скоринга german из репозитория UCI.

## Пример 3. Проблема переобучения

Как отбросить переобученные закономерности на этапе обучения?

Частота ошибок на контроле, %



Частота ошибок на обучении, %

**Задача** предсказания отдалённого результата хирургического лечения атеросклероза. Точки — найденные закономерности.

## Модификация критериев $(p, n)$ с поправкой на переобучение

1. Вычислить оценки расслоения–связности как функции  $\varepsilon$ :

$$P\left[\frac{1}{k}n(r, \bar{X}) - \frac{1}{\ell}n(r, X) \geq \varepsilon\right] \leq \eta_n(\varepsilon);$$

$$P\left[\frac{1}{\ell}p(r, X) - \frac{1}{k}p(r, \bar{X}) \geq \varepsilon\right] \leq \eta_p(\varepsilon);$$

2. Обращение оценок: с вероятностью  $1 - \eta$

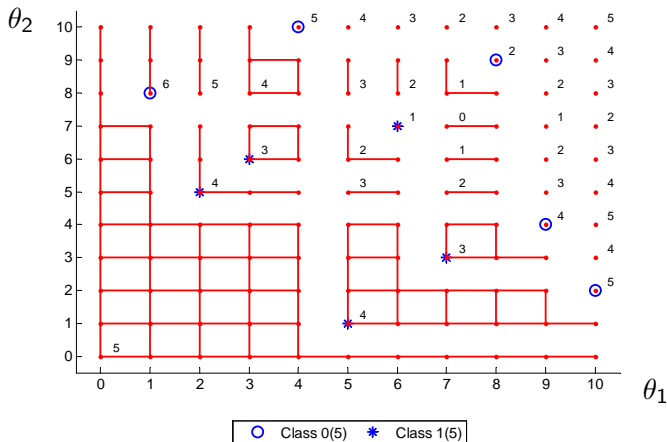
$$n(r, \bar{X}) \leq \underbrace{\frac{k}{\ell}n(r, X) + k\varepsilon_n(\eta)}_{\hat{n}(r, X)};$$

$$p(r, \bar{X}) \geq \underbrace{\frac{k}{\ell}p(r, X) - k\varepsilon_p(\eta)}_{\hat{p}(r, X)}.$$

3. Для поиска закономерностей вместо  $(p \rightarrow \max, n \rightarrow \min)$  использовать модифицированный критерий  $(\hat{p} \rightarrow \max, \hat{n} \rightarrow \min)$ .

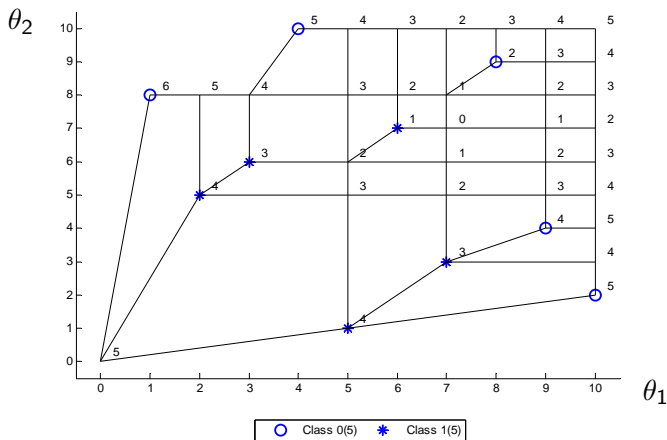
# Множество конъюнктивных закономерностей

Пример: разделимая 2-мерная выборка,  $L = 10$ , два класса.  
закономерности:  $r(x) = [f_1(x) \leq \theta_1] \wedge [f_2(x) \leq \theta_2]$ .



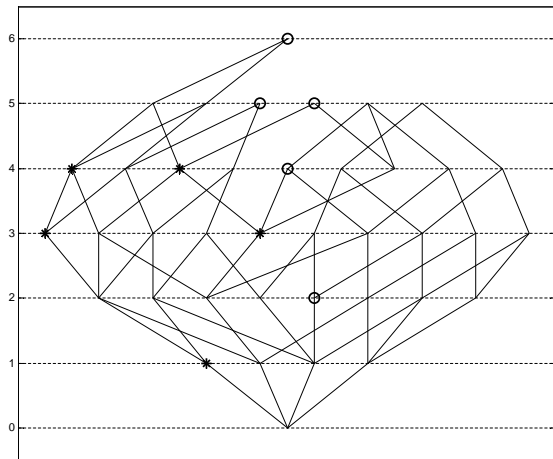
## Классы эквивалентности закономерностей

Пример: разделимая 2-мерная выборка,  $L = 10$ , два класса.  
закономерности:  $r(x) = [f_1(x) \leq \theta_1] \wedge [f_2(x) \leq \theta_2]$ .



## Классы эквивалентности закономерностей

**Пример:** граф расслоения–связности, изоморфный графу классов эквивалентности с предыдущего слайда.



## Эксперимент на реальных данных

Реальные задачи классификации из репозитория UCI:

задачи	объектов	признаков
australian	690	14
echo cardiogram	74	10
heart disease	294	13
hepatitis	155	19
labor relations	40	16
liver	345	6

Методы обучения композиций логических закономерностей:

- WV (weighted voting) — взвешенное голосование;
- DL (decision list) — решающий список.

Методика тестирования: 10-кратный скользящий контроль.



## Результаты эксперимента на реальных данных

методы	задачи					
	austr	echo	heart	hepa	labor	liver
RIPPER-opt	15.5	2.97	19.7	20.7	18.0	32.7
RIPPER+opt	15.2	5.53	20.1	23.2	18.0	31.3
C4.5 (Tree)	14.2	5.51	20.8	18.8	14.7	37.7
C4.5 (Rules)	15.5	6.87	20.0	18.8	14.7	37.5
C5.0	14.0	4.30	21.8	20.1	18.4	31.9
SLIPPER	15.7	4.34	19.4	17.4	12.3	32.2
LR	14.8	4.30	19.9	18.8	14.2	32.0
WV	14.9	4.37	20.1	19.0	14.0	32.3
DL	15.1	4.51	20.5	19.5	14.7	35.8
<b>WV модиф.</b>	<b>14.1</b>	<b>3.2</b>	<b>19.3</b>	<b>18.1</b>	<b>13.4</b>	<b>30.2</b>
<b>DL модиф.</b>	14.4	3.6	19.5	18.6	13.6	32.3

По каждой задаче **выделено** два лучших результата.

## Классификатор ближайшего соседа (NN, nearest neighbor)

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

$$\mu: X \mapsto a, \quad a(x) = y^* \left( \arg \min_{x' \in X} \rho(x, x') \right).$$

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

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

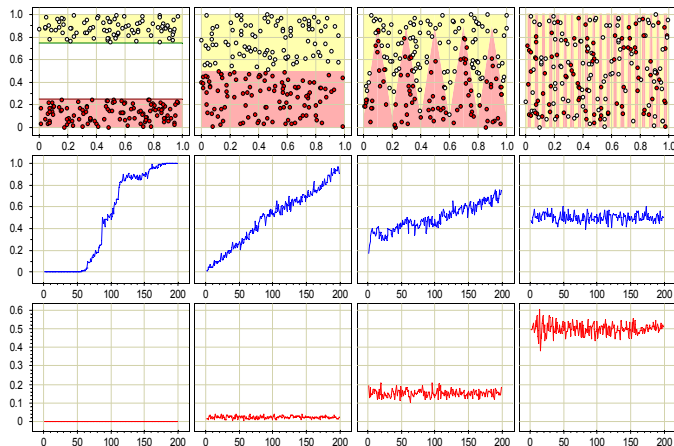
$$K(m, \mathbb{X}) = \frac{1}{L} \sum_{i=1}^L [y^*(x_{im}) \neq y_i]; \quad m = 1, \dots, L-1,$$

### 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 $CCV$

- 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 [Максим Иванов].

**Замечание.** Теория Вапника–Червоненкиса вообще не даёт содержательных оценок для метода ближайшего соседа, т.к. ёмкость данного семейства алгоритмов бесконечна.

## Задача отбора множества эталонов $\Omega \subseteq \mathbb{X}$

Модификация NN  $\mu_\Omega: X \mapsto a$ ,  $a(x) = y(\arg \min_{x' \in \Omega} \rho(x, x'))$ .

**Определение (профиль компактности относительно  $\Omega$ )**

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

где  $x_{im}^\Omega$  —  $m$ -й сосед объекта  $x_i$  из множества  $\Omega$ ;

**Теорема (точное выражение CCV относительно  $\Omega$ )**

$$\text{CCV}(\mu_\Omega, \mathbb{X}) = \sum_{i=1}^L \underbrace{\sum_{m=1}^k [y(x_i) \neq y(x_{im}^\Omega)]}_{T(x_i, \Omega) \text{ — вклад объекта } x_i \text{ в CCV}} \frac{C_{L-1-m}^{\ell-1}}{L C_{L-1}^\ell}.$$

## Задача отбора эталонов (prototype selection)

Задача:  $CCV(\mu_\Omega, \mathbb{X}) \rightarrow \min_{\Omega}$ .

Если  $x$  добавляется в  $\Omega$  или удаляется из  $\Omega$ , то для обновления  $CCV$  достаточно пересчитать вклады только тех объектов, для которых  $x$  является не далее, чем  $k$ -м соседом.

$r_i(x, \Omega)$  — ранг объекта  $x \in \Omega$  во множестве  $\Omega$ , упорядоченном по возрастанию расстояний  $\rho(x_i, x)$ .

### Определение

Обратная окрестность  $m$ -го порядка объекта  $x \in \Omega$  — это множество объектов, для которых объект  $x$  является не далее, чем  $m$ -м соседом:

$$V_m(x, \Omega) = \{x_i \in \mathbb{X} \mid r_i(x, \Omega) \leq m\}, \quad m = 1, \dots, L - 1.$$

## Жадный алгоритм удаления не-эталонов

**Вход:** выборка  $\mathbb{X}$ , параметр  $\delta > 0$ ;

**Выход:** множество эталонов  $\Omega \subseteq \mathbb{X}$ ;

- 
- 1:  $\Omega := \mathbb{X}$ ;  $Q := \sum_{i=1}^L T(x_i, \Omega)$ ;  $Q_{\min} := Q$ ;
  - 2: **повторять**
  - 3:   count := 0;   — счётчик числа удалений
  - 4:   **для всех**  $x \in \Omega$
  - 5:      $\Delta Q := \sum_{x_i \in V_k(x, \Omega)} \left( T(x_i, \Omega \setminus \{x\}) - T(x_i, \Omega) \right)$ ;
  - 6:     **если**  $Q + \Delta Q < Q_{\min} + \delta$  **то**
  - 7:        $Q := Q + \Delta Q$ ;  $\Omega := \Omega \setminus \{x\}$ ;
  - 8:       **если**  $Q < Q_{\min}$  **то**  $Q_{\min} := Q$ ;
  - 9:       count := count + 1;
  - 10: **пока** count > 0;

## Жадный алгоритм добавления эталонов

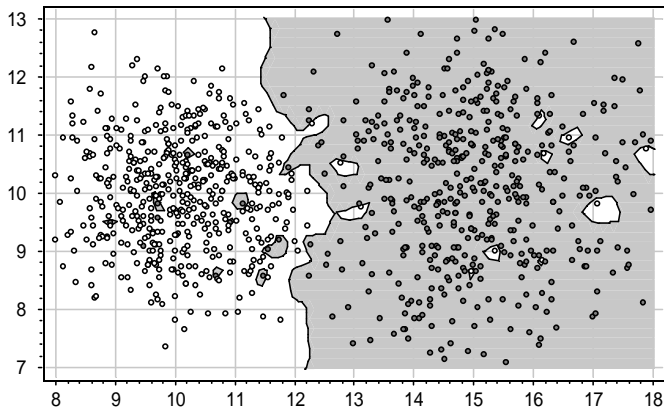
**Вход:** выборка  $\mathbb{X}$ , параметр  $\delta > 0$ ;

**Выход:** множество эталонов  $\Omega \subseteq \mathbb{X}$ ;

- 
- 1:  $\Omega := \{\text{по одному случайному объекту от каждого класса}\};$   
 $Q := \sum_{i=1}^L T(x_i, \Omega); \quad Q_{\min} := Q;$
  - 2: **повторять**
  - 3:  $\text{count} := 0;$  — счётчик числа добавлений
  - 4: **для всех**  $x \in \mathbb{X} \setminus \Omega$
  - 5:  $\Delta Q := \sum_{x_i \in V_k(x, \Omega \cup \{x\})} \left( T(x_i, \Omega \cup \{x\}) - T(x_i, \Omega) \right);$
  - 6: **если**  $Q + \Delta Q < Q_{\min} - \delta$  **то**
  - 7:  $Q := Q + \Delta Q; \quad \Omega := \Omega \cup \{x\};$
  - 8: **если**  $Q < Q_{\min}$  **то**  $Q_{\min} := Q;$
  - 9:  $\text{count} := \text{count} + 1;$
  - 10: **пока**  $\text{count} > 0;$

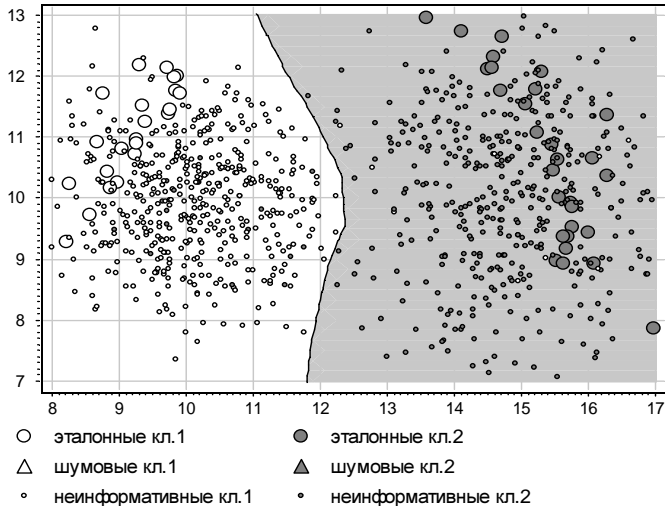


## Модельные данные

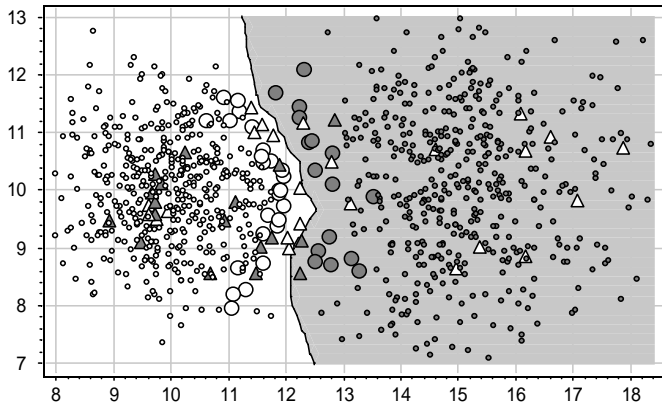


Модельная задача классификации: 1000 объектов, метод NN.

## Жадное добавление эталонных объектов



## Жадное удаление не-эталонных объектов



○ эталонные кл.1

● эталонные кл.2

△ шумовые кл.1

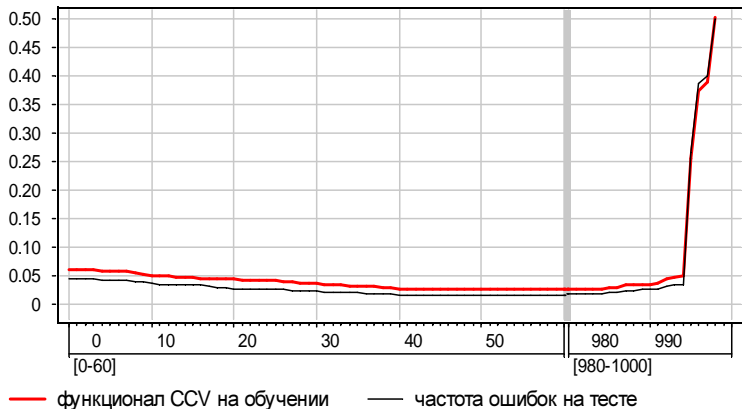
▲ шумовые кл.2

◦ неинформативные кл.1

◐ неинформативные кл.2

## Жадное удаление не-эталонных объектов

Зависимость CCV от числа удаленных неэталонных объектов.



**Чудо: при отборе эталонов переобучения вообще нет!**

## Открытые проблемы

- 1 На практике оценка расслоения–связности вычисляется не по генеральному множеству  $\mathbb{X}$ , а по случайной наблюдаемой подвыборке  $X$ . **Обосновать** эту подмену.
- 2 **Найти** способ быстрого пересчёта оценок при добавлении в выборку ещё одного объекта, ещё одного признака.
- 3 **Уточнить** оценки расслоения–связности с учётом конкуренции между алгоритмами с хэмминговым расстоянием, большим 1.
- 4 **Обобщить** оценки расслоения–связности на случай небинарных функций потерь.
- 5 **Совершенствовать** методы обучения с помощью комбинаторных оценок обобщающей способности.

**Следующий этап — переход от теории к технологии.**

## Спецкурс ТНОП

«Теория надёжности обучения по прецедентам»  
по понедельникам, 18:00, ауд. 615 (ВМК МГУ)

Страницы на [www.MachineLearning.ru](http://www.MachineLearning.ru):

Теория надёжности обучения по прецедентам (курс лекций, К. В. Воронцов)  
✓ **выложено учебное пособие с задачами по курсу ТНОП**

Расслоение и сходство алгоритмов (виртуальный семинар)

Слабая вероятностная аксиоматика

Участник: Vokov