

Методы машинного обучения. Комбинаторная теория переобучения

Воронцов Константин Вячеславович

www.MachineLearning.ru/wiki?title=User:Vokov

вопросы к лектору: k.vorontsov@iai.msu.ru

материалы курса:

github.com/MSU-ML-COURSE/ML-COURSE-24-25

орг.вопросы по курсу: ml.cmc@mail.ru

ВМК МГУ • 29 октября 2024

1 Проблема оценивания переобучения

- Вероятность переобучения
- Теория Вапника–Червоненкиса
- Бритва Оккама

2 Эксперименты с переобучением

- Монотонная цепь алгоритмов
- Переобучение цепей
- Переобучение при выборе из двух алгоритмов

3 Комбинаторная теория переобучения

- Граф расслоения–связности
- Порождающие и запрещающие множества
- Основная оценка расслоения–связности

Проблема оценивания обобщающей способности

Дано:

$X^L = \{x_1, \dots, x_L\}$ — генеральное множество объектов

$X^L = X^\ell \sqcup X^k$ — разбиение на обучающую и контрольную части

$A = \{a: X \times W \rightarrow Y\}$ — модель, семейство алгоритмов

$\mu: (X \times Y)^\ell \rightarrow A$ — метод обучения

$\mathcal{L}(a, x)$ — функция потерь алгоритма a на объекте x

$Q(a, U) = \frac{1}{|U|} \sum_{x \in U} \mathcal{L}(a, x)$ — средняя потеря на выборке U

Найти:

способ оценивать и минимизировать $Q(\mu(X^\ell), X^k)$, не зная X^k ,
для широкого класса задач (Y, A, W) и методов (μ) ;

при упрощающих предположениях:

— функция потерь бинарная;

— все разбиения $X^\ell \sqcup X^k$ случайны и равновероятны.

Бинарная функция потерь. Матрица ошибок

$X^L = \{x_1, \dots, x_L\}$ — конечное *генеральное* множество объектов
 $A = \{a_1, \dots, a_D\}$ — конечное множество (семейство) *алгоритмов*
 $\mathcal{L}(a, x) \equiv I(a, x) = [a \text{ ошибается на } x]$ — *индикатор ошибки*

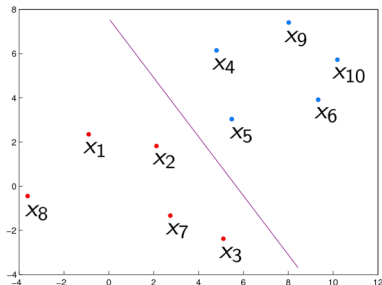
$L \times D$ -матрица ошибок с попарно различными столбцами:

	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^ℓ — наблюдаемая (обучающая) выборка длины ℓ
\dots	0	0	0	0	1	1	\dots	1	
x_ℓ	0	0	1	0	0	0	\dots	0	
$x_{\ell+1}$	0	0	0	1	1	1	\dots	0	X^k — 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	

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

$\nu(a, X) = n(a, X)/|X|$ — *частота ошибок* a на выборке X

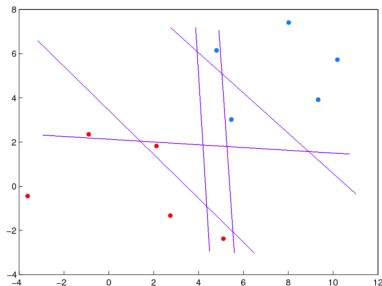
Пример. Матрица ошибок линейных классификаторов



1 вектор с 0 ошибками

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

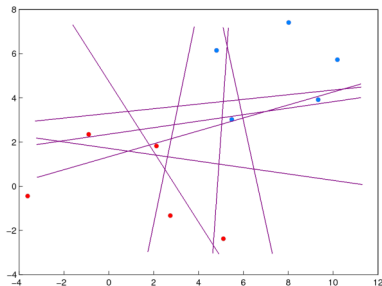
Пример. Матрица ошибок линейных классификаторов



1 вектор с 0 ошибками
5 векторов с 1 ошибкой

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



1 вектор с 0 ошибками
5 векторов с 1 ошибкой
8 векторов с 2 ошибками
и т. д...

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	0	1	0	1	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	...

Задача оценивания вероятности переобучения

Переобученность — разность частот ошибок на X^k и на X^ℓ :

$$\delta(\mu, X^\ell, X^k) = \nu(\mu(X^\ell), X^k) - \nu(\mu(X^\ell), X^\ell).$$

Переобучение — это событие $\delta(\mu, X^\ell, X^k) \geq \varepsilon$.

Основное вероятностное предположение:

$\mathbf{P} \equiv \mathbf{E} \equiv \frac{1}{C_L^L} \sum_{X^\ell \sqcup X^k = X^L}$ — все разбиения $X^\ell \sqcup X^k = X^L$ равновероятны

Интерпретация 1: это CCV, полный скользящий контроль.

Интерпретация 2: это гипотеза независимости выборки X^L .

Основная задача — оценить *вероятность переобучения*:

$$R_\varepsilon(\mu, X^L) = \mathbf{P}[\delta(\mu, X^\ell, X^k) \geq \varepsilon].$$

$\hat{\mathbf{P}} \equiv \hat{\mathbf{E}} \equiv \frac{1}{|N|} \sum_{X^\ell \in N}$ — эмпирическая оценка методом Монте-Карло по случайному подмножеству разбиений N

Простейший, но важный частный случай

Пусть $A = \{a\}$ — одноэлементное множество, $m = n(a, X^L)$.

Тогда вероятность переобучения есть вероятность большого отклонения частот ошибок алгоритма a в двух подвыборках:

$$R_\varepsilon(a, X^L) = P[\delta(a, X^\ell, X^k) \geq \varepsilon] = P[\nu(a, X^k) - \nu(a, X^\ell) \geq \varepsilon].$$

Теорема

Для любого X^L , любого $\varepsilon \in [0, 1]$ вероятность переобучения описывается функцией гипергеометрического распределения:

$$R_\varepsilon(a, X^L) = \mathcal{H}_L^{\ell, m} \left(\frac{\ell}{L} (m - \varepsilon k) \right),$$

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

Доказательство

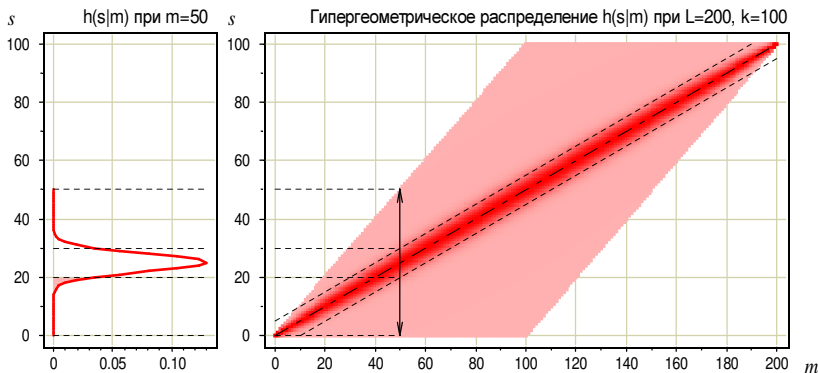
1. Обозначим $s = n(a, X^\ell)$.
2. «Школьная» задача по теории вероятностей:
в урне L шаров, m из них чёрные; извлекаем ℓ шаров наугад.
Какова вероятность того, что s из них чёрные?

$$P[n(a, X^\ell) = s] = C_m^s C_{L-m}^{\ell-s} / C_L^\ell.$$

3. Распишем R_ε , подставив $\nu(a, X^k) = \frac{m-s}{k}$, $\nu(a, X^\ell) = \frac{s}{\ell}$:

$$\begin{aligned} R_\varepsilon(a, X^L) &= P[\nu(a, X^k) - \nu(a, X^\ell) \geq \varepsilon] = \\ &= \sum_{s=0}^{\ell} \underbrace{\left[\frac{m-s}{k} - \frac{s}{\ell} \geq \varepsilon \right]}_{s \leq \frac{\ell}{L}(m-\varepsilon k)} \underbrace{P[n(a, X^\ell) = s]}_{C_m^s C_{L-m}^{\ell-s} / C_L^\ell} = \\ &= \mathcal{H}_L^{\ell, m} \left(\frac{\ell}{L}(m - \varepsilon k) \right). \quad \blacksquare \end{aligned}$$

Гипергеометрическое распределение $h(s|m) = C_m^s C_{L-m}^{\ell-s} / C_L^\ell$



Концентрация вероятностной меры, закон больших чисел:
предсказание числа $m = n(a, X^L)$ по числу $s = n(a, X^\ell)$
возможно благодаря узости гипергеометрического пика,
причём при $\ell, k \rightarrow \infty$ он сужается, и $\nu(a, X^\ell) \rightarrow \nu(a, X^k)$

Принцип равномерной сходимости частот

Рассмотрим случай, когда A произвольное, конечное.

1. Вероятность переобучения оценим сверху вероятностью большого *равномерного отклонения* частот: для любых X^L, μ

$$\begin{aligned} R_\varepsilon(\mu, X^L) &= P[\delta(\mu, X^\ell, X^k) \geq \varepsilon] \leq \\ &\leq P\left[\max_{a \in A} \delta(a, X^\ell, X^k) \geq \varepsilon\right] = \tilde{R}_\varepsilon(A, X^L). \end{aligned}$$

2. Оценим вероятность объединения событий суммой их вероятностей (неравенство Буля, union bound):

$$\begin{aligned} \tilde{R}_\varepsilon(A, X^L) &= P \max_{a \in A} [\delta(a, X^\ell, X^k) \geq \varepsilon] \leq \\ &\leq P \sum_{a \in A} [\delta(a, X^\ell, X^k) \geq \varepsilon] = \sum_{a \in A} \underbrace{P[\delta(a, X^\ell, X^k) \geq \varepsilon]}_{R_\varepsilon(a, X^L)}. \end{aligned}$$

Оценка Вапника–Червоненкиса (VC bound)

Таким образом, доказана

Лемма. Для любых X^L , μ , конечного A и $\varepsilon \in [0, 1]$

$$\tilde{R}_\varepsilon(A, X^L) \leq \sum_{a \in A} \mathcal{H}_L^{\ell, m} \left(\frac{\ell}{L} (m - \varepsilon k) \right), \quad m = n(a, X^L).$$

Теорема (Вапник и Червоненкис, 1968)

Для любых X^L , μ , конечного A и $\varepsilon \in [0, 1]$

$$\begin{aligned} \tilde{R}_\varepsilon(A, X^L) &\leq |A| \cdot \max_m \mathcal{H}_L^{\ell, m} \left(\frac{\ell}{L} (m - \varepsilon k) \right) \leq \\ &\leq |A| \cdot \frac{3}{2} \exp(-\varepsilon^2 \ell), \quad \text{при } \ell = k. \end{aligned}$$

В.Н.Вапник, А.Я.Червоненкис. О равномерной сходимости частот появления событий к их вероятностям. 1968.

Обобщение на случай бесконечных семейств A

Функция роста $\Delta^A(L)$ семейства A — это максимальное по X^L число различных векторов ошибок $\vec{a} = (I(a, x_1), \dots, I(a, x_L))$.
В оценке можно заменить $|A|$ на функцию роста $\Delta^A(L)$.

Ёмкость (размерность Вапника–Червоненкиса) семейства A — это максимальная длина выборки h , для которой $\Delta^A(h) = 2^h$.

Теорема

Если такое h существует, то $\Delta^A(L) \leq C_L^0 + \dots + C_L^h \leq \frac{3}{2} \frac{L^h}{h!}$.

Теорема

Ёмкость семейства линейных классификаторов на два класса

$$a(x) = \text{sign}(w_1 x^1 + \dots + w_n x^n), \quad x = (x^1, \dots, x^n) \in X.$$

равна размерности пространства параметров, $\text{VCdim}(A) = n$.

Обращение оценки Вапника–Червоненкиса (при $\ell = k$)

$$1. \text{ Оценка: } P\left[\max_{a \in A}(\nu(a, X^k) - \nu(a, X^\ell)) \geq \varepsilon\right] \leq \Delta \frac{3}{2} \exp(-\ell \varepsilon^2) = \eta$$

Тогда для любого $a \in A$ с вероятностью не менее $(1 - \eta)$

$$\nu(a, X^k) \leq \underbrace{\nu(a, X^\ell)}_{\text{эмпирический риск}} + \underbrace{\sqrt{\frac{1}{\ell} \ln \Delta + \frac{1}{\ell} \ln \frac{3}{2\eta}}}_{\text{штраф за сложность}}.$$

$$2. \text{ Оценка: } P\left[\max_{a \in A}(\nu(a, X^k) - \nu(a, X^\ell)) \geq \varepsilon\right] \leq \frac{3}{2} \frac{L^h}{h!} \frac{3}{2} \exp(-\ell \varepsilon^2) = \eta$$

Тогда для любого $a \in A$ с вероятностью не менее $(1 - \eta)$

$$\nu(a, X^k) \leq \underbrace{\nu(a, X^\ell)}_{\text{эмпирический риск}} + \underbrace{\sqrt{\frac{h}{\ell} \ln \frac{2e\ell}{h} + \frac{1}{\ell} \ln \frac{9}{4\eta}}}_{\text{штраф за сложность}}.$$

Метод структурной минимизации риска (СМР)

Дано: система вложенных подсемейств возрастающей ёмкости

$$A_0 \subset A_1 \subset \dots \subset A_h \subset \dots$$

Найти: оптимальную ёмкость h^* , такую, что

$$\nu(a, X^k) \leq \underbrace{\min_{a \in A_h} \nu(a, X^\ell)}_{\text{минимизация эмпирического риска}} + \underbrace{\sqrt{\frac{h}{\ell} \ln \frac{2e\ell}{h} + \frac{1}{\ell} \ln \frac{9}{4\eta}}}_{\text{штраф за сложность}} \rightarrow \min_h$$

Недостатки СМР:

- верхняя оценка R_ε очень сильно завышена
- следовательно, h^* может оказаться заниженной
- на практике предпочитают эмпирические оценки CV

В.Н.Вапник, А.Я.Червоненкис. Теория распознавания образов. М.: Наука, 1974.

В.Н.Вапник, А.Я.Червоненкис. Восстановление зависимостей по эмпирическим данным. М.: Наука, 1979.

Причины завышенности оценок Вапника-Червоненкиса

- **Оценка равномерного отклонения (uniform bound)**
сильно завышена, когда большая часть алгоритмов имеет исчезающе малую вероятность быть результатом обучения

На практике распределение

$$q(a) = P[\mu(X^\ell) = a], \quad a \in A$$

как правило, существенно неравномерно!

Будем называть это **эффектом расслоения семейства A** .

- **Неравенство Буля (union bound)** сильно завышено, когда среди бинарных векторов ошибок есть много похожих

Будем называть это **эффектом сходства алгоритмов**

K. V. Vorontsov. Splitting and similarity phenomena in the sets of classifiers and their effect on the probability of overfitting. 2008.

Оценка «бритва Оккама» (Occam's Razor Bound)

Мы не можем знать распределение $q(a) = P[\mu(X^\ell) = a]$,
но можем попробовать его «угадать».

Теорема (Лангфорд, 2002)

Для произвольной нормированной функции, $p(a)$, $\sum_{a \in A} p(a) = 1$,
любого $\eta \in (0, 1)$, любого $a \in A$ с вероятностью не менее $1 - \eta$

$$\nu(a, X^k) \leq \nu(a, X^\ell) + \sqrt{\frac{1}{\ell} \ln \frac{1}{p(a)} + \frac{1}{\ell} \ln \frac{3}{2\eta}}$$

Утв 1. Если угадали, $p(a) = q(a)$, то $E \ln \frac{1}{p(\mu(X^\ell))}$ минимально.

Утв 2. Бритва Оккама может учитывать эффект расслоения,
но не учитывает эффект сходства, поэтому тоже завышена.

John Langford. Quantitatively tight sample complexity bounds. 2002.

Оценка «бритва Оккама»: как задать $p(a)$?

Пример 1.

A — конечное множество.

Равномерное распределение $p(a) = \frac{1}{|A|}$

даёт оценку Вапника–Червоненкиса:

для любых $a \in A$, $\eta \in (0, 1)$ с вероятностью не менее $1 - \eta$

$$\nu(a, X^k) \leq \nu(a, X^\ell) + \sqrt{\frac{1}{\ell} \ln |A| + \frac{1}{\ell} \ln \frac{3}{2\eta}}.$$

Оценка «бритва Оккама»: как ещё задать $p(a)$?

Пример 2. Задача классификации на 2 класса $Y = \{-1, +1\}$,
 A — линейные классификаторы в \mathbb{R}^n :

$$a(x) = \text{sign}(w_1 x^1 + \dots + w_n x^n), \quad x = (x^1, \dots, x^n) \in X.$$

Гауссовское распределение: веса $w \in \mathbb{R}^n$ — независимые с.в.,
с нулевым ожиданием и равными дисперсиями σ^2 :

$$p(a) = Z \exp\left(-\frac{1}{2\sigma^2} \|w\|^2\right),$$

где Z — нормировочный множитель, не зависящий от w .

Подставляя в «бритву Оккама», получаем... регуляризацию!

$$\nu(a, X^k) \leq \nu(a, X^\ell) + \sqrt{\frac{\|w\|^2}{2\ell\sigma^2} + \frac{1}{\ell} \ln \frac{3}{2\eta Z}}.$$

Как учесть не только расслоение, но и сходство алгоритмов?

Цель порождается непрерывным изменением параметра, например, пороговым решающим правилом над признаком f :

$$A = \{a_d(x) = [f(x) \geq \theta_d] : d = 0, \dots, D\}$$

Пример:

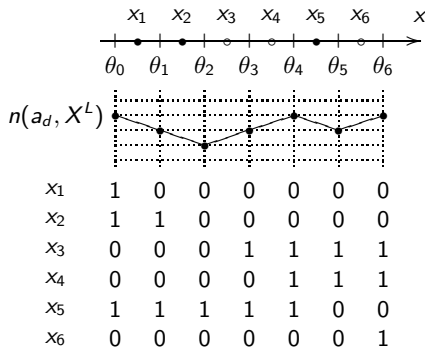
2 класса $\{\bullet, \circ\}$

6 объектов

7 порогов-алгоритмов

Матрица ошибок \rightarrow

В цепи каждый следующий алгоритм отличается от предыдущего только на одном объекте


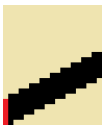




Как зависит переобучение от содержимого матрицы ошибок?

Верхние оценки VC-теории зависят только от размера матрицы ошибок $L \times |A|$, и потому сильно завышены.

Эксперимент: сравним R_ϵ и CCV для четырёх матриц ошибок:


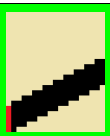
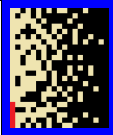

- лучший алгоритм одинаковый
- есть/нет *расслоение* — когда каждый следующий алгоритм допускает на одну ошибку больше, чем предыдущий
- есть/нет *связность* — когда каждый следующий алгоритм лишь на одном объекте отличается от предыдущего

	рассл.	без рассл.
связн.		
без связн.		

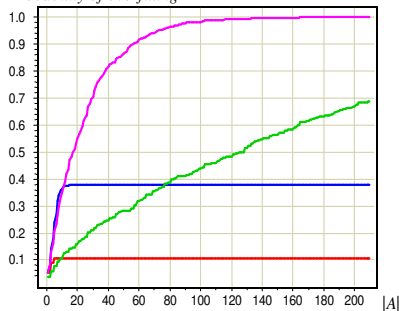
Vorontsov K. V. Splitting and similarity phenomena in the sets of classifiers and their effect on the probability of overfitting. PRIA, 2009.

Зависимость вероятности переобучения от числа алгоритмов

$\ell = k = 100$, $m^* = 10$, $\varepsilon = 0.05$, $|N| = 10^3$ разбиений Монте-Карло

	рассл.	без рассл.
связн.		
без связн.		

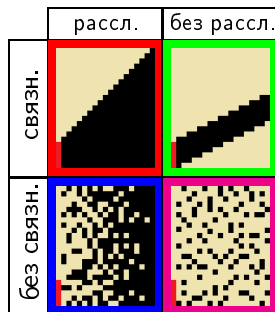
Probability of overfitting



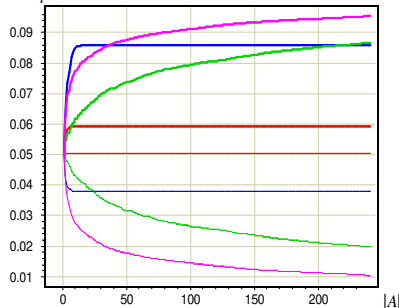
- *связность* замедляет темп роста кривой $R_\varepsilon(|A|)$
- *расслоение* понижает уровень горизонтальной асимптоты
- огромные семейства с P&C могут почти не переобучаться
- VC-оценка линейно мажорирует худшую из этих кривых

Зависимость оценки CCV от числа алгоритмов

$\ell = k = 100$, $m^* = 10$, $|N| = 10^3$ разбиений Монте-Карло



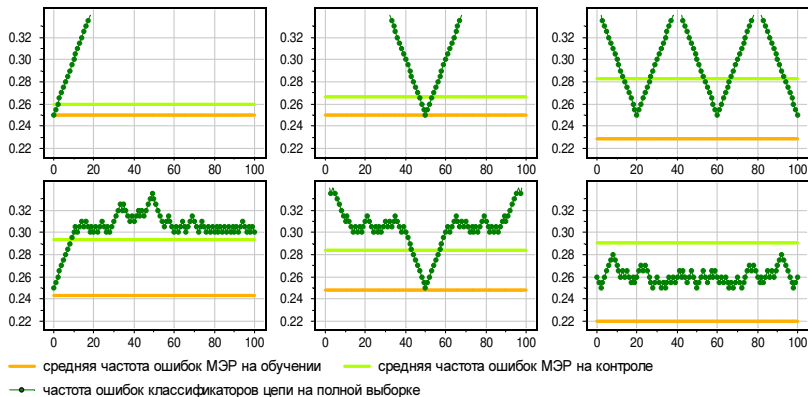
Complete Cross-Validation



- справа: оценки $CCV = \hat{E}_\nu(\mu(X^\ell), X^k)$ и $\hat{E}_\nu(\mu(X^\ell), X^\ell)$
- без P&C даже 10 алгоритмов могут сильно переобучаться
- без учёта эффектов расслоения и связности получение точных оценок вероятности переобучения невозможно

Эксперимент. Переобучение цепей с различным расслоением

Условия эксперимента: $L = 100$, $\ell = 50$, $m^* = 25$,
 $|N| = 10^4$ случайных разбиений Монте-Карло



Семейство из двух алгоритмов $A = \{a_1, a_2\}$

Пусть для алгоритмов a_1, a_2 известны m_0, m_1, m_2, m_3 :

$$\begin{aligned} a_1 &= (1, \dots, 1, 1, \dots, 1, 0, \dots, 0, 0, \dots, 0); \\ a_2 &= (\underbrace{1, \dots, 1}_{m_0}, \underbrace{0, \dots, 0}_{m_1}, \underbrace{1, \dots, 1}_{m_2}, \underbrace{0, \dots, 0}_{m_3}). \end{aligned}$$

Сходство векторов ошибок измеряется расстоянием Хэмминга:

$$r(a_1, a_2) = \sum_{i=1}^L |I(a_1, x_i) - I(a_2, x_i)| = m_1 + m_2$$

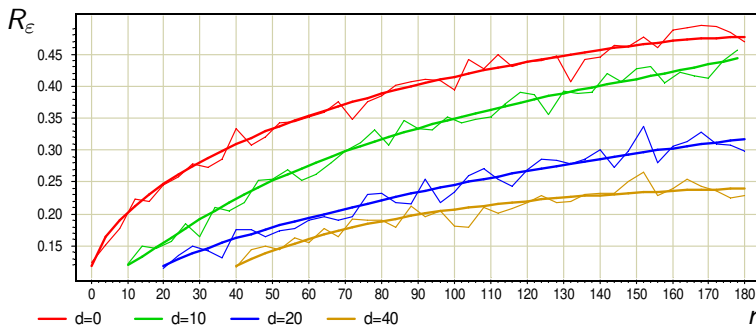
Расслоение измеряется разностью числа ошибок:

$$d(a_1, a_2) = |n(a_1, X^L) - n(a_2, X^L)| = |m_1 - m_2|$$

Условия эксперимента: $\ell = k = 100$, $m_0 = 20$, $\varepsilon = 0.05$,
 метод Монте-Карло по $|N| = 10^4$ случайных разбиений.

Эффекты сходства и расслоения для пары алгоритмов

Зависимость вероятности переобучения R_ϵ
от расстояния Хэмминга r и расслоения d :



- переобучение возникает даже при выборе из двух алгоритмов
- чем более они схожи, тем меньше переобучение
- чем больше расслоение, тем меньше переобучение

Вероятность переобучения семейства из двух алгоритмов

Пусть для алгоритмов a_1, a_2 известны m_0, m_1, m_2, m_3 :

$$a_1 = (1, \dots, 1, 1, \dots, 1, 0, \dots, 0, 0, \dots, 0);$$

$$a_2 = (\underbrace{1, \dots, 1}_{m_0}, \underbrace{0, \dots, 0}_{m_1}, \underbrace{1, \dots, 1}_{m_2}, \underbrace{0, \dots, 0}_{m_3}).$$

Теорема (о вероятности переобучения)

Если $A = \{a_1, a_2\}$ и метод μ минимизирует эмпирический риск (число ошибок на обучении), то для любого $\varepsilon \in [0, 1]$

$$R_\varepsilon(\mu, X^L) = \sum_{s_0=0}^{m_0} \sum_{s_1=0}^{m_1} \sum_{s_2=0}^{m_2} \frac{C_{m_0}^{s_0} C_{m_1}^{s_1} C_{m_2}^{s_2} C_{L-m_0-m_1-m_2}^{\ell-s_0-s_1-s_2}}{C_L^\ell} \times$$

$$\times \left([s_1 < s_2] \left[s_0 + s_1 \leq \frac{\ell}{L}(m_0 + m_1 - \varepsilon k) \right] + \right.$$

$$\left. + [s_1 \geq s_2] \left[s_0 + s_2 \leq \frac{\ell}{L}(m_0 + m_2 - \varepsilon k) \right] \right)$$

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 $x \in X^L$;
предшествование $a \prec b$: $a \leq b$ и $\|b - a\| = 1$.

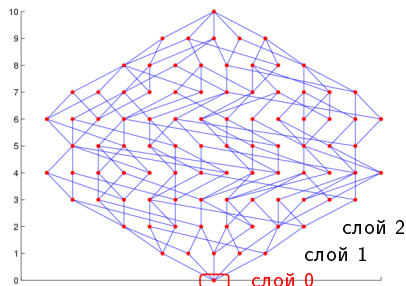
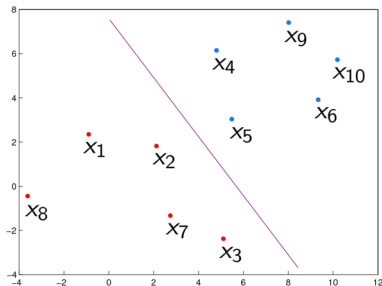
Опр. Граф расслоения–связности $\langle A, E \rangle$:

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

Свойства графа расслоения–связности:

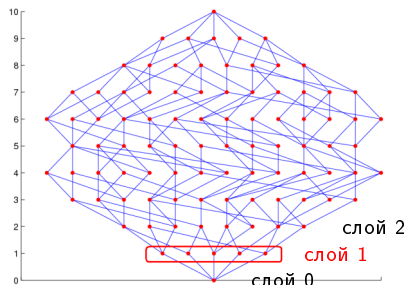
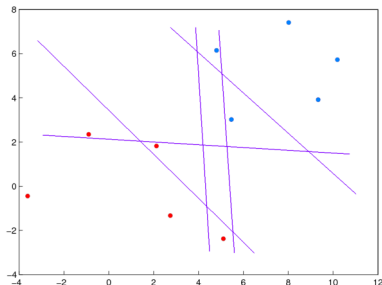
- это подграф графа Хассе отношения порядка \leq на A ;
- каждому ребру (a, b) соответствует *рёберный объект* $x_{ab} \in X^L$, такой, что $I(a, x_{ab}) = 0$, $I(b, x_{ab}) = 1$;
- граф является многодольным со слоями
 $A_m = \{a \in A: n(a, X^L) = m\}$, $m = 0, \dots, L$;

Пример 1. Семейство линейных алгоритмов классификации



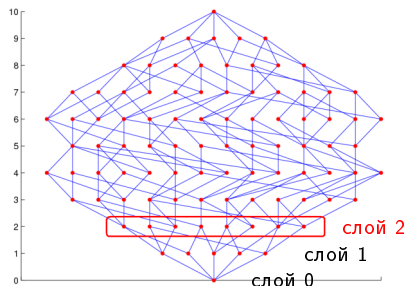
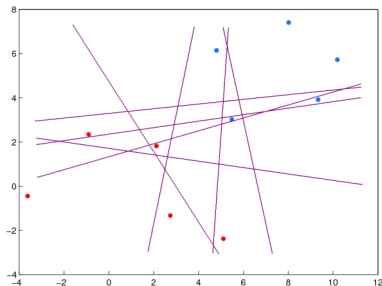
	слой 0
x1	0
x2	0
x3	0
x4	0
x5	0
x6	0
x7	0
x8	0
x9	0
x10	0

Пример 1. 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 1. Семейство линейных алгоритмов классификации



	слой 0	слой 1					слой 2								
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	0	1	0	1	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	...

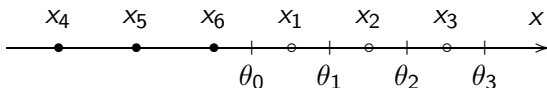
Пример 2. Монотонная цепь

Опр. *Монотонная цепь* алгоритмов: $a_0 \prec a_1 \prec \dots \prec a_D$.

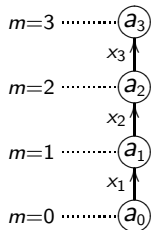
Пример: 1D пороговый классификатор $a_d(x) = [x - \theta_d]$;

2 класса $\{\bullet, \circ\}$

6 объектов



Граф семейства:



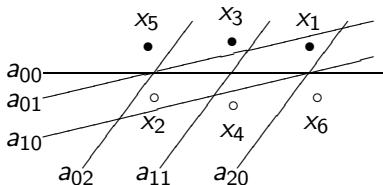
Матрица ошибок:

	a_0	a_1	a_2	a_3
x_1	0	1	1	1
x_2	0	0	1	1
x_3	0	0	0	1
x_4	0	0	0	0
x_5	0	0	0	0
x_6	0	0	0	0

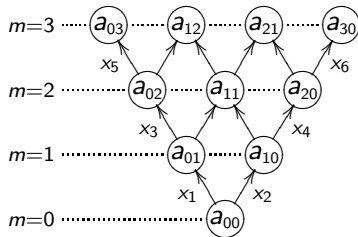
Пример 3. Двумерная сеть классификаторов

Пример:

2D линейный классификатор,
2 класса $\{\bullet, \circ\}$,
6 объектов



Граф семейства:



Матрица ошибок:

	a_{00}	a_{01}	a_{10}	a_{02}	a_{11}	a_{20}	a_{03}	a_{12}	a_{21}	a_{30}
x_1	0	1	0	1	1	0	1	1	1	0
x_2	0	0	1	0	1	1	0	1	1	1
x_3	0	0	0	1	0	0	1	1	0	0
x_4	0	0	0	0	0	1	0	0	1	1
x_5	0	0	0	0	0	0	1	0	0	0
x_6	0	0	0	0	0	0	0	0	0	1

Порождающее и запрещающее множества алгоритмов

Определение

Верхняя связность $u(a)$ алгоритма a — это число всех рёбер, исходящих из вершины a в графе расслоения-связности:

$$u(a) = |X_a|, \quad X_a = \{x_{ab} \in X^L \mid a \prec b\};$$

X_a называется *порождающим множеством* алгоритма a .

Определение

Ошибочность $q(a)$ алгоритма a — это число различных рёберных объектов на всех путях, ведущих в a :

$$q(a) = |X'_a|, \quad X'_a = \{x \in X^L \mid \exists b \in A: b \prec a, l(b, x) < l(a, x)\};$$

X'_a называется *запрещающим множеством* алгоритма a .

Характеристики **расслоения** и **связности** алгоритма

Верхняя связность $u(a) = \#\{x_{ab} \in X^L \mid a \prec b\}$

Нижняя связность $d(a) = \#\{x_{ba} \in X^L \mid b \prec a\}$

Ошибочность $q(a) = \#\{x \in X^L \mid \exists b \in A: b \prec a, I(b, x) < I(a, x)\}$

Число ошибок $m(a) = n(a, X^L)$.

Утв.

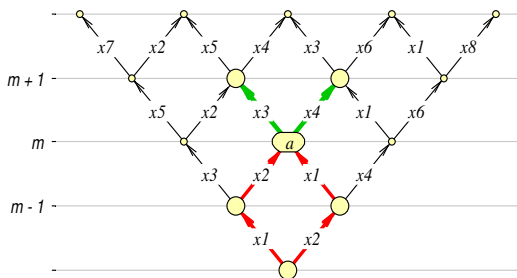
$$d(a) \leq q(a) \leq m(a)$$

Пример: двумерная
сеть алгоритмов

$$u(a) = \#\{x_3, x_4\} = 2$$

$$d(a) = \#\{x_1, x_2\} = 2$$

$$q(a) = \#\{x_1, x_2\} = 2$$



Верхняя оценка расслоения–связности

Метод минимизации эмпирического риска μ *монотонный*, если

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

где $K(a, U)$ — строго монотонная функция вектора ошибок a :
для любых $U \subset X^L$, $a, b \in A$ если $a < b$, то $K(a, X) < K(b, X)$.

Пример. Функция $K(a, U) = \nu(a, U)$ — строго монотонная.

Теорема

Для любого монотонного метода μ , любых X^L , A и $\varepsilon \in (0, 1)$

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

K. V. Vorontsov, A. A. Ivahnenko, I. M. Reshetnyak. Generalization bound based on the splitting and connectivity graph of the set of classifiers. 2010.

Идея доказательства

1. Построим по μ *монотонный пессимистичный* метод $\bar{\mu}$ максимизации переобученности: $\bar{\mu}(X^\ell) = \arg \max_{a \in A_K(X^\ell)} \delta(a, X^\ell, X^k)$.

Тогда $R_\varepsilon(\mu, X^L) \leq R_\varepsilon(\bar{\mu}, X^L)$ — верхняя оценка.

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

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

4. По формуле полной вероятности:

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

Свойства верхней оценки расслоения–связности

- 1 При $|A| = 1$ функция гипергеометрического распределения:

$$R_\varepsilon = \mathcal{H}_L^{\ell, m} \left(\frac{\ell}{L} (m - \varepsilon k) \right) \rightarrow 0 \text{ при } \ell, k \rightarrow \infty.$$

- 2 При $q = u = 0$ и $\ell = k$ это оценка Вапника-Червоненкиса:

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

- 3 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 R_ε** .

- 4 Вероятность получить алгоритм в результате обучения

$$P[\mu(X^\ell) = a] \leq P_a = C_{L-u-q}^{\ell-u} / C_L^\ell$$

- 5 Оценка является точной (обращается в равенство)
в случае многомерных монотонных сетей алгоритмов.

- Без расслоения и связности переобучение наступает уже при нескольких десятках алгоритмов.
- Расслоение и связность сильно уменьшают переобучение.
- На практике семейства, как правило, ими обладают.
- Схема применения оценок вероятности переобучения:
 - 1) оценить $\eta = R_\varepsilon(\mu, X^L)$ по нескольким нижним слоям;
 - 2) применив обращение, оценить ε через η ;
 - 3) использовать оценку $\nu(X^L) + \varepsilon(\eta)$ как внешний критерий или регуляризатор для выбора структуры модели.
- Завышенность оценок R_ε может приводить к занижению сложности (переупрощению) моделей в этой схеме.
- Практичнее пользоваться оценками скользящего контроля.
- Тесные верхние оценки (tight bounds) переобучения выводятся в COLT (Computational Learning Theory)