МАШИННОЕ ОБУЧЕНИЕ
линейные модели классификации и регрессии

Воронцов Константин Вячеславович
ФУПМ МФТИ • ВМК МГУ • Яндекс • FORECSYS

6 июля 2016
Сочи, Сириус • Проектная смена • 1–24 июля 2016

Содержание

Восстановление зависимости по эмпирическим данным

Задача восстановления зависимости $y = f(x)$
по точкам *обучающей выборки* (x_i, y_i), $i = 1, \ldots, \ell$.

Дано: векторы $x_i = (x_i^1, \ldots, x_i^n)$ — объекты обучающей выборки,
$y_i = f(x_i)$ — правильные ответы, $i = 1, \ldots, \ell$:

$$\begin{pmatrix} x_1^1 & \ldots & x_1^n \\ \ldots & \ldots & \ldots \\ x_\ell^1 & \ldots & x_\ell^n \end{pmatrix} \xrightarrow{f} \begin{pmatrix} y_1 \\ \ldots \\ y_\ell \end{pmatrix}$$

Найти: функцию $a(x)$, способную давать правильные ответы
на *тестовых объектах* $\tilde{x}_i = (\tilde{x}_i^1, \ldots, \tilde{x}_i^n)$, $i = 1, \ldots, k$:

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

Типы признаков и типы задач

Типы признаков, $x_i^j \in D_j$, в зависимости от множества D_j:

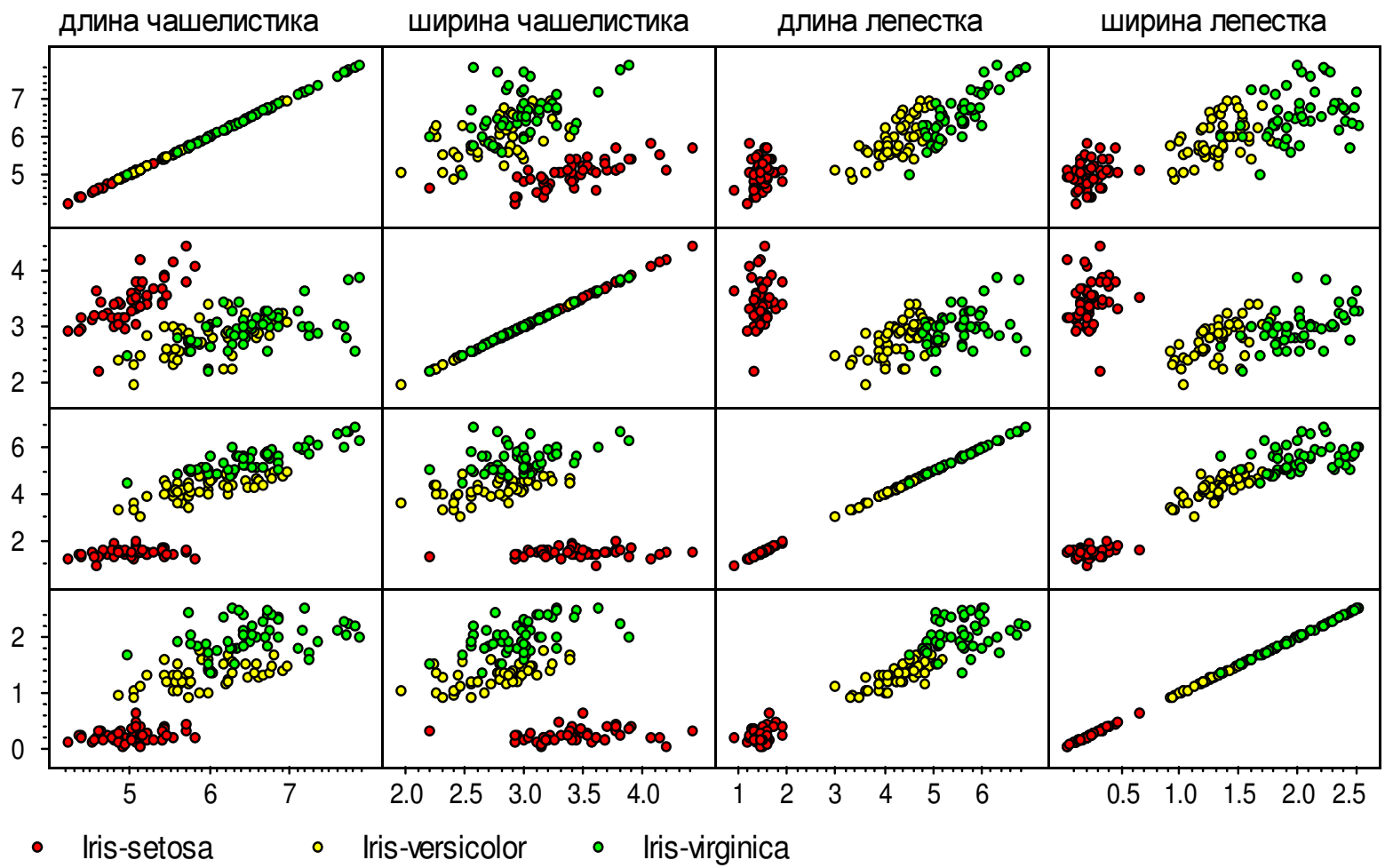
- $D_j = \{0, 1\}$ — *бинарный* признак;
- $|D_j| < \infty$ — *номинальный* признак;
- D_j упорядочено — *порядковый* признак;
- $D_j = \mathbb{R}$ — *количественный* признак.

Типы задач, $y_i \in Y$, в зависимости от множества Y:

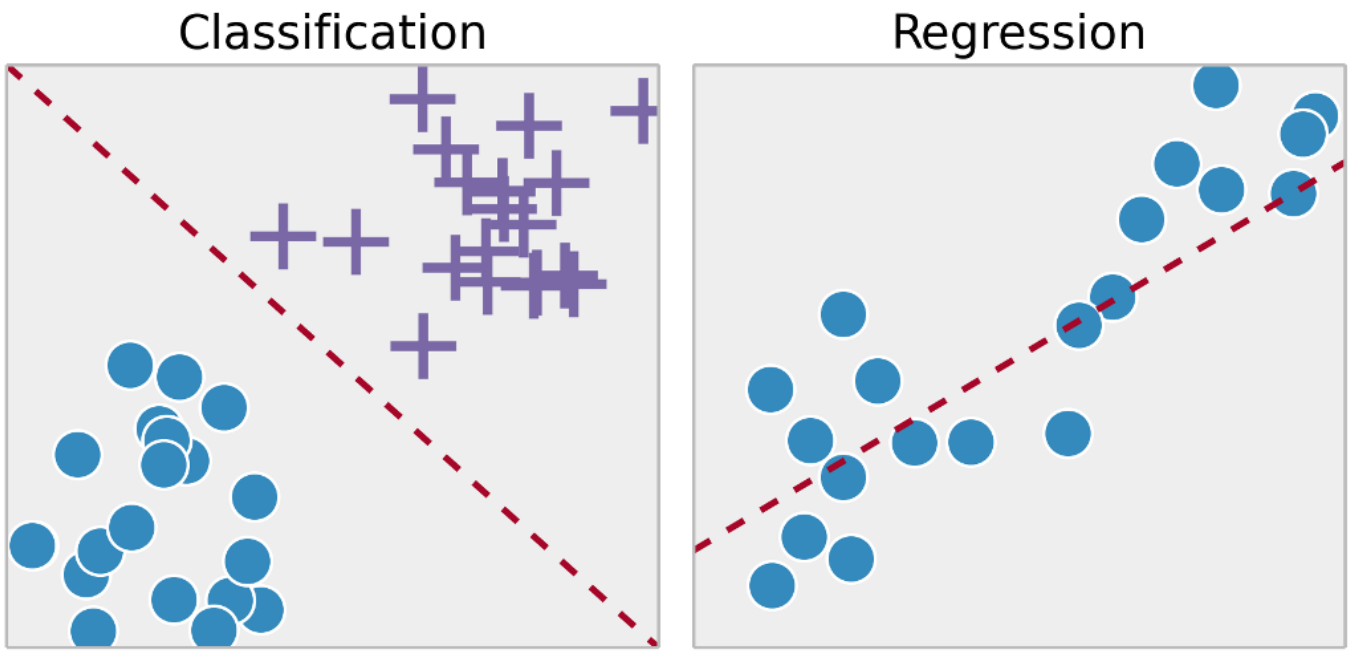
- $Y = \{0, 1\}$ или $Y = \{-1, +1\}$ — *классификация* на 2 класса;
- $Y = \{1, \ldots, M\}$ — на M непересекающихся классов;
- $Y = \{0, 1\}^M$ — на M классов, которые могут пересекаться;
- $Y = \mathbb{R}$ — задача восстановления регрессии;
- Y упорядочено — задача ранжирования (learning to rank).

Пример. Задача классификации цветков ириса [Фишер, 1936]

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



Линейные модели классификации и регрессии



Почему мы так любим линейные модели?
Как обучать параметры линейных моделей?
Как проверять качество предсказательной модели?

Восстановление регрессии — это оптимизация

Задача восстановления регрессионной зависимости, $y_i \in \mathbb{R}$

1. Выбираем *модель регрессии*, например, линейную:
$$a(x, w) = \langle x, w \rangle = \sum_{j=1}^{n} x^j w_j, \qquad x, w \in \mathbb{R}^n$$
2. Выбираем функцию потерь, например, квадратичную:
$$\mathscr{L}(a, y) = (a - y)^2$$
3. Минимизируем потери *методом наименьших квадратов*:
$$Q(w) = \frac{1}{\ell} \sum_{i=1}^{\ell} \big(a(x_i, w) - y_i\big)^2 \to \min_w$$
4. Проверяем прогностическую (обобщающую) способность:
$$\tilde{Q}(w) = \frac{1}{k} \sum_{i=1}^{k} \big(a(\tilde{x}_i, w) - \tilde{y}_i\big)^2$$

Обучение классификации — тоже оптимизация

Задача классификации, $y_i \in \{-1, +1\}$

1. Выбираем *модель классификации*, например, линейную:
$$a(x, w) = \text{sign}\langle x, w\rangle$$
2. Выбираем функцию потерь, например, *бинарную*:
$$\mathscr{L}(a, y) = \big[a(x_i, w)y_i < 0\big]$$
3. Минимизируем *частоту ошибок* на обучающей выборке:
$$Q(w) = \frac{1}{\ell}\sum_{i=1}^{\ell}\big[a(x_i, w)y_i < 0\big] \to \min_w$$
4. Проверяем прогностическую (обобщающую) способность:
$$\tilde{Q}(w) = \frac{1}{k}\sum_{i=1}^{k}\big[a(\tilde{x}_i, w)\tilde{y}_i < 0\big]$$

Обучение классификации — сглаживание функции потерь

Задача классификации, $y_i \in \{-1, +1\}$

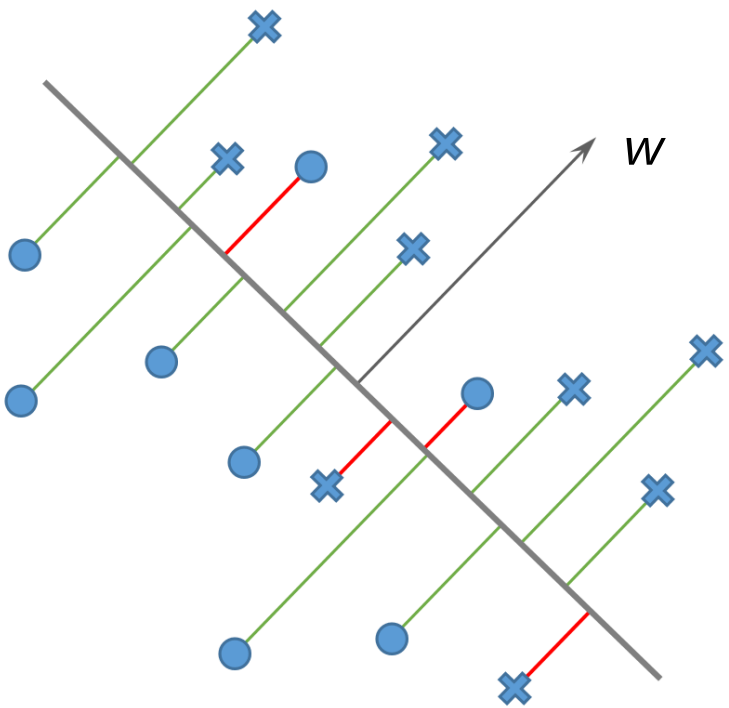
1. Выбираем *модель классификации*, например, линейную:
$$a(x, w) = \text{sign}\langle x, w\rangle$$
2. Мажорируем пороговую функцию потерь непрерывной:
$$\big[M_i < 0\big] \leqslant \mathscr{L}\big(M_i\big), \qquad M_i = \langle x_i, w\rangle y_i \;\; — \;\; \text{отступ (margin)}$$
3. Минимизируем *сглаженную частоту ошибок*:
$$Q(w) = \frac{1}{\ell}\sum_{i=1}^{\ell} \mathscr{L}\big(\langle x_i, w\rangle y_i\big) \to \min_w$$
4. Проверяем прогностическую (обобщающую) способность:
$$\tilde{Q}(w) = \frac{1}{k}\sum_{i=1}^{k} \big[\langle \tilde{x}_i, w\rangle \tilde{y}_i < 0\big]$$

Геометрический смысл отступов объектов

w — нормаль (направляющий вектор) разделяющей гиперплоскости, направлена в сторону класса $+1$

$\langle x_i, w \rangle$ — проекция вектора x_i на вектор нормали w

$\langle x_i, w \rangle y_i = M_i$ — проекция на нормаль в сторону класса y_i



— объекты класса $+1$

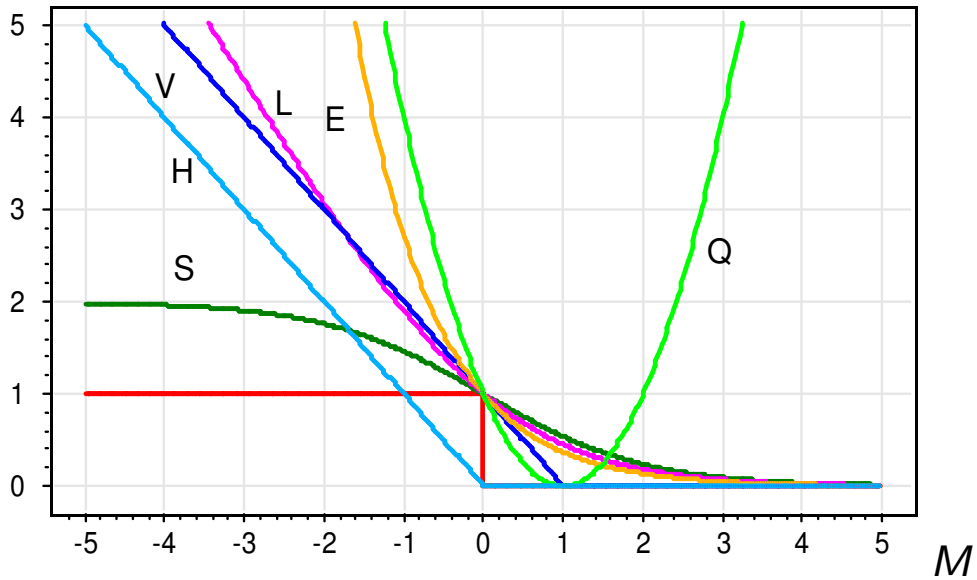
— объекты класса -1

— нет ошибки, $M_i > 0$

— ошибка, $M_i < 0$

Непрерывные аппроксимации пороговой функции потерь

Часто используемые непрерывные функции потерь $\mathscr{L}(M)$:



$V(M) = (1 - M)_+$ — кусочно-линейная (SVM)

$H(M) = (-M)_+$ — кусочно-линейная (Hebb's rule)

$L(M) = \log_2(1 + e^{-M})$ — логарифмическая (LR, Logistic Regression)

$Q(M) = (1 - M)^2$ — квадратичная (Fisher's Linear Discriminant)

$S(M) = 2(1 + e^{M})^{-1}$ — сигмоидная (ANN, Artificial Neural Network)

$E(M) = e^{-M}$ — экспоненциальная (AdaBoost)

$[M < 0]$ — пороговая функция потерь.

Общие подходы к решению оптимизационных задач

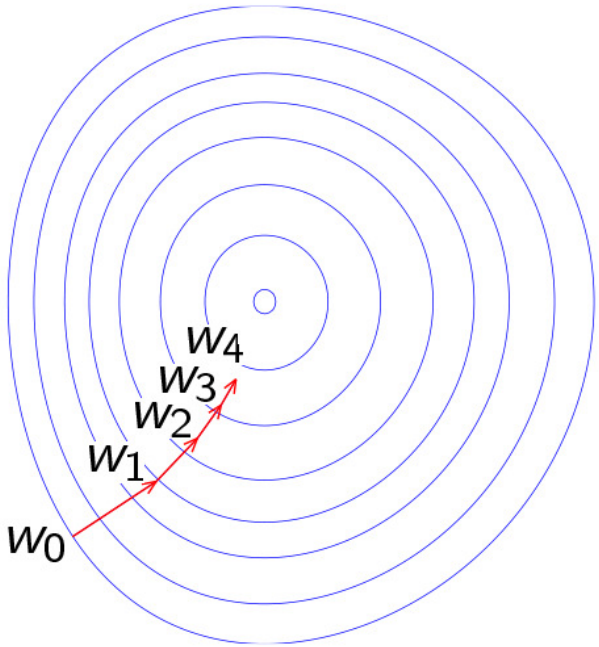
Аналитический подход (напр. метод наименьших квадратов):
Если w — точка минимума *гладкой* функции $Q(w)$, то

$$\frac{\partial Q(w)}{\partial w_j} = 0, \quad j = 1, \ldots, n.$$

Это система n уравнений с n неизвестными.

Численный метод — градиентный спуск:

1 начальное приближение w^0, $t := 0$;
2 **повторять**
3 $\quad w_j^{t+1} := w_j^t - h^t \cdot \frac{\partial Q(w^t)}{\partial w_j}$, $j = 1, \ldots, n$;
4 $\quad t := t + 1$;
5 **пока** процесс не сойдётся;



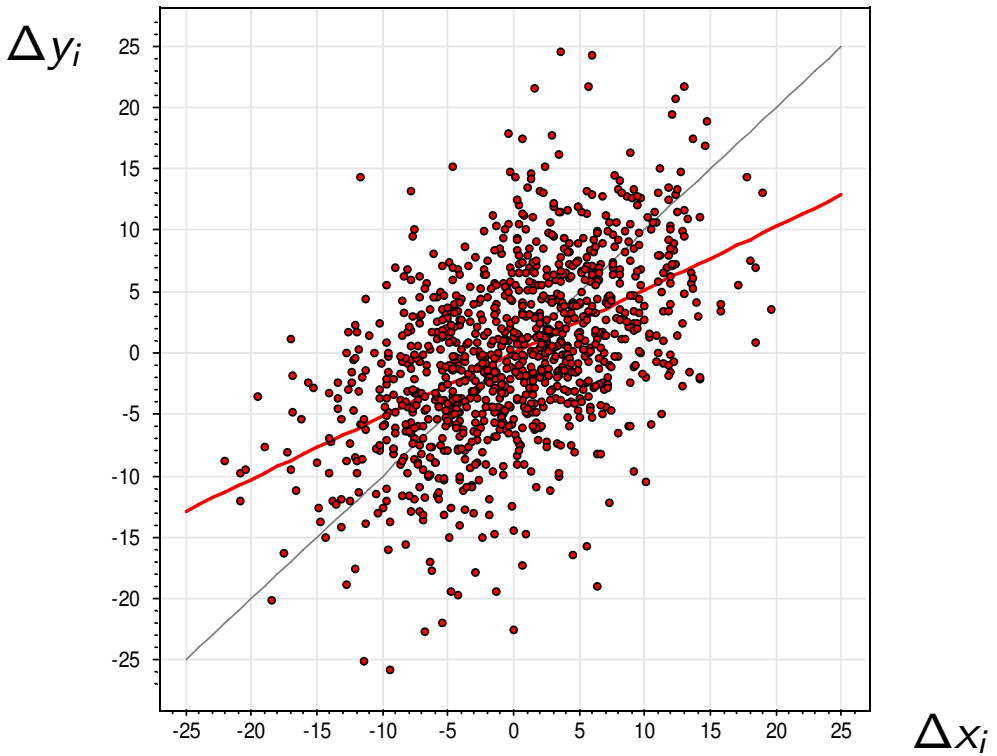
Откуда пошло название «регрессия» (Гальтон, 1886)

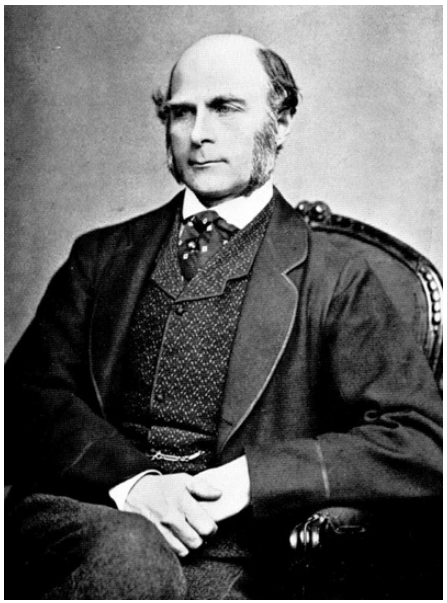
Исследование наследственности роста.

отклонение роста от среднего в популяции:

Δx_i — отклонение роста отца

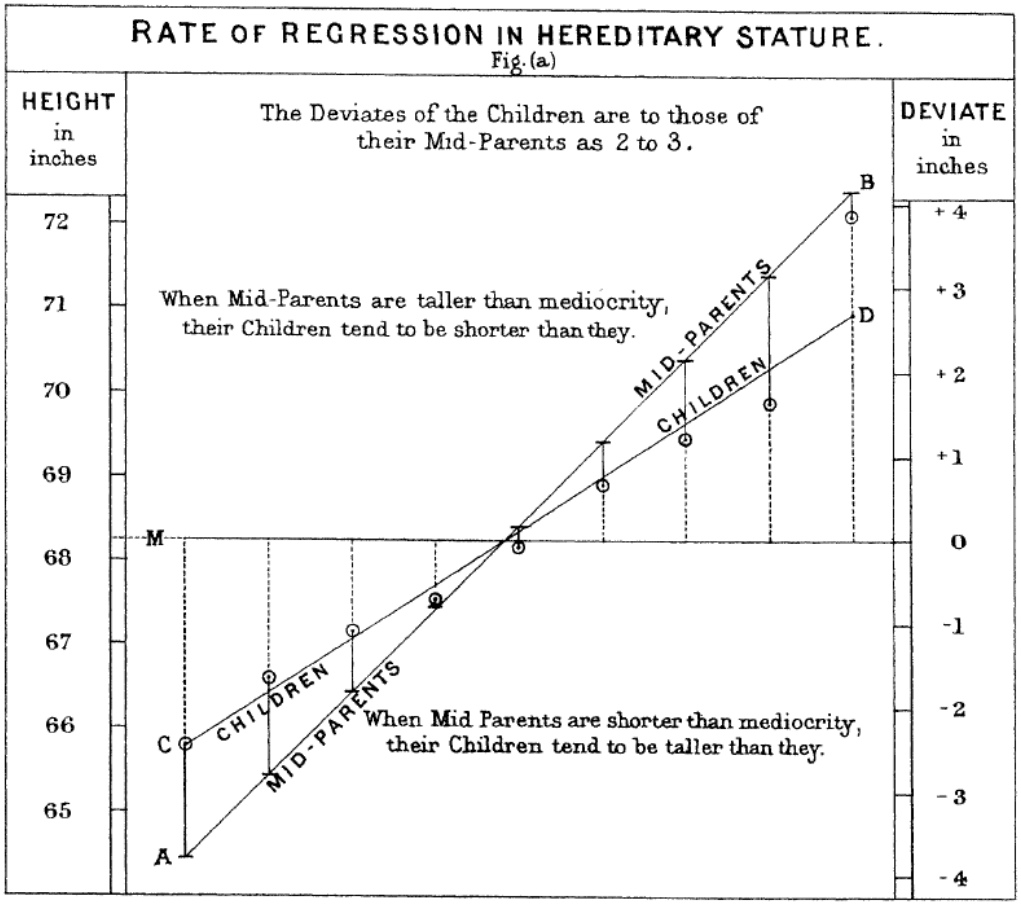
Δy_i — отклонение роста взрослого сына

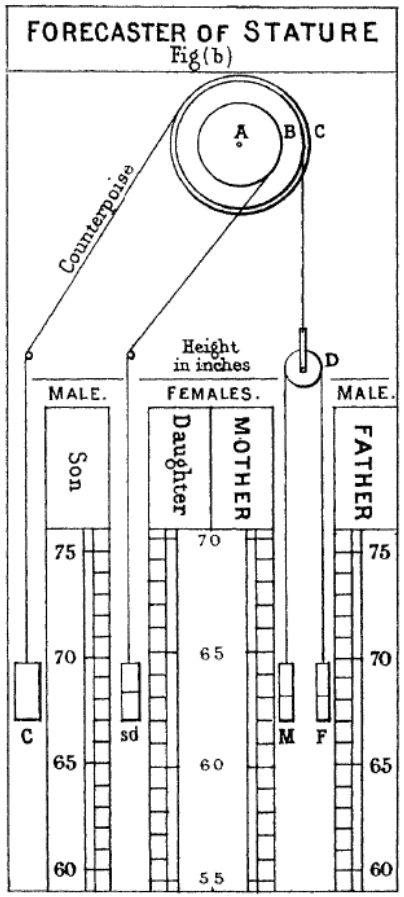




Фрэнсис Гальтон
(1822–1911)

Скрытый смысл: «регрессия» — сначала данные, потом модель





Galton F. Regression Towards Mediocrity in Hereditary Stature. 1886.

Задача проведения прямой через заданные точки

Дано: $x_i, y_i \in \mathbb{R}, \;\; i = 1, \ldots, \ell$

Найти: параметры $w = (\alpha, \beta)$ линейной модели $y = \alpha x + \beta$

Критерий: $Q(\alpha, \beta) = \sum_{i=1}^{\ell} (\alpha x_i + \beta - y_i)^2 \;\to\; \min$

Аналитический метод решения:

$$\frac{\partial Q}{\partial \alpha} = 0 \quad \Rightarrow \quad \alpha \sum_{i=1}^{\ell} x_i^2 + \beta \sum_{i=1}^{\ell} x_i - \sum_{i=1}^{\ell} x_i y_i = 0$$

$$\frac{\partial Q}{\partial \beta} = 0 \quad \Rightarrow \quad \alpha \sum_{i=1}^{\ell} x_i + \beta \sum_{i=1}^{\ell} 1 - \sum_{i=1}^{\ell} y_i = 0$$

Это система линейных уравнений 2×2:

$$\begin{cases} \alpha S_{xx} + \beta S_x = S_{xy} \\ \alpha S_x + \beta S_1 = S_y \end{cases} \quad \Rightarrow \quad \begin{cases} \alpha = \dfrac{S_{xy} S_1 - S_x S_y}{S_{xx} S_1 - S_x S_x} \\ \beta = \dfrac{S_{xx} S_y - S_{xy} S_x}{S_{xx} S_1 - S_x S_x} \end{cases}$$

Метод стохастического градиента (SG, Stochastic Gradient)

Задача классификации: $y_i \in \{-1, +1\}$, $a(x, w) = \text{sign}\langle w, x\rangle$.
Минимизация сглаженной частоты ошибок:

$$Q(w) = \sum_{i=1}^{\ell} \mathscr{L}\big(\langle w, x_i\rangle y_i\big) \to \min_w .$$

Один шаг *градиентного спуска*:

$$w^{t+1} := w^t - h^t \sum_{i=1}^{\ell} \mathscr{L}'\big(\langle w^t, x_i\rangle y_i\big)\, x_i\, y_i.$$

Идея ускорения сходимости: брать (x_i, y_i) по одному в случайном порядке и сразу обновлять вектор весов,

$$w^{t+1} := w^t - h^t \mathscr{L}'\big(\langle w^t, x_i\rangle y_i\big)\, x_i\, y_i.$$

Алгоритм SG (Stochastic Gradient)

Вход: выборка $(x_i, y_i)_{i=1}^{\ell}$;
Выход: веса $w_1, \ldots, w_n$;

1 инициализировать веса w_j, $j = 1, \ldots, n$;
2 **повторять**
3 выбрать случайный объект (x_i, y_i) из обучающей выборки;
4 выбрать величину градиентного шага h;
5 выполнить градиентный шаг:
$w_j := w_j - h\mathscr{L}'\big(\langle w, x_i\rangle y_i\big)x_i^j y_i$ для всех $j = 1, \ldots, n$;
6 **пока** процесс не сойдётся куда-нибудь;

Преимущества и недостатки:

⊕ можно брать какие угодно модели и функции потерь $\mathscr{L}$
⊕ хорошо работает на больших выборках
⊖ возможно застревание в локальных экстремумах

Эвристики

- Выбор начального приближения, например, так:

$$w_j^0 := \frac{\langle y, f_j \rangle}{\langle f_j, f_j \rangle} \quad \text{(из одномерной линейной регрессии)}$$

$f_j = (x_i^j)_{i=1}^{\ell}$ — вектор значений j-го признака,
$y = (y_i)_{i=1}^{\ell}$ — вектор ответов.

- Выбор темпа обучения (градиентного шага) h^t:
сходимость гарантируется для выпуклых $Q(w)$ при

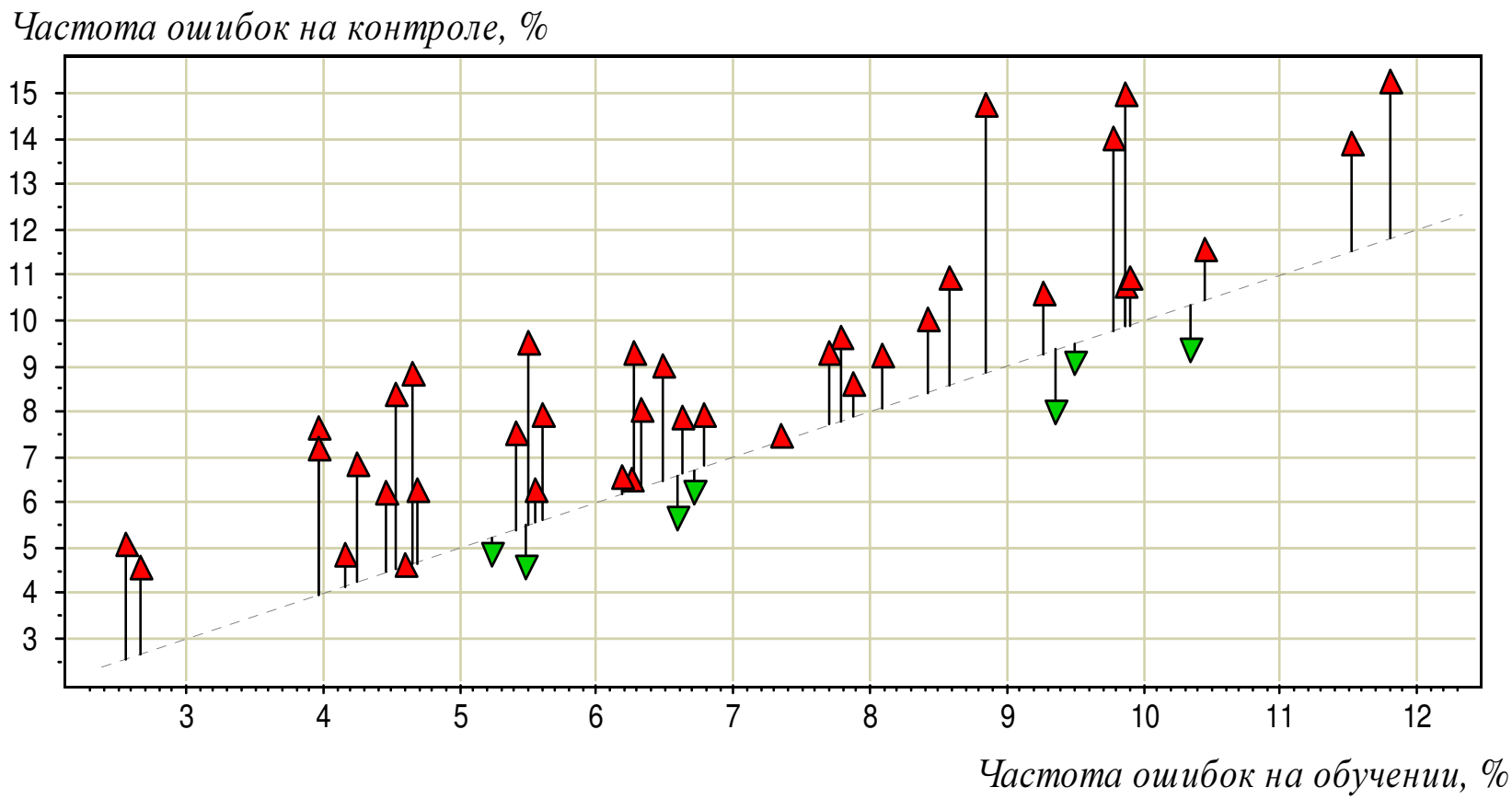
$$h^t \to 0, \quad \sum_{t=1}^{\infty} h^t = \infty, \quad \sum_{t=1}^{\infty} (h^t)^2 < \infty,$$

в частности можно положить $h^t = \frac{1}{t}$;

- Выбор порядка предъявления объектов:
— случайно, но попеременно из разных классов;
— чаще брать пограничные объекты с малым $|M_i|$;

Пример. Переобучение в задаче медицинской диагностики

Задача предсказания отдалённого результата хирургического лечения атеросклероза. Точки — различные алгоритмы.



Пример: переобучение полиномиальной регрессии

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

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

Модель полиномиальной регрессии

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

Обучение методом наименьших квадратов:

$$Q(a, X^\ell) = \sum_{i=1}^{\ell} (\theta_0 + \theta_1 x_i + \cdots + \theta_n x_i^n - y_i)^2 \to \min_{\theta_0,\ldots,\theta_n}.$$

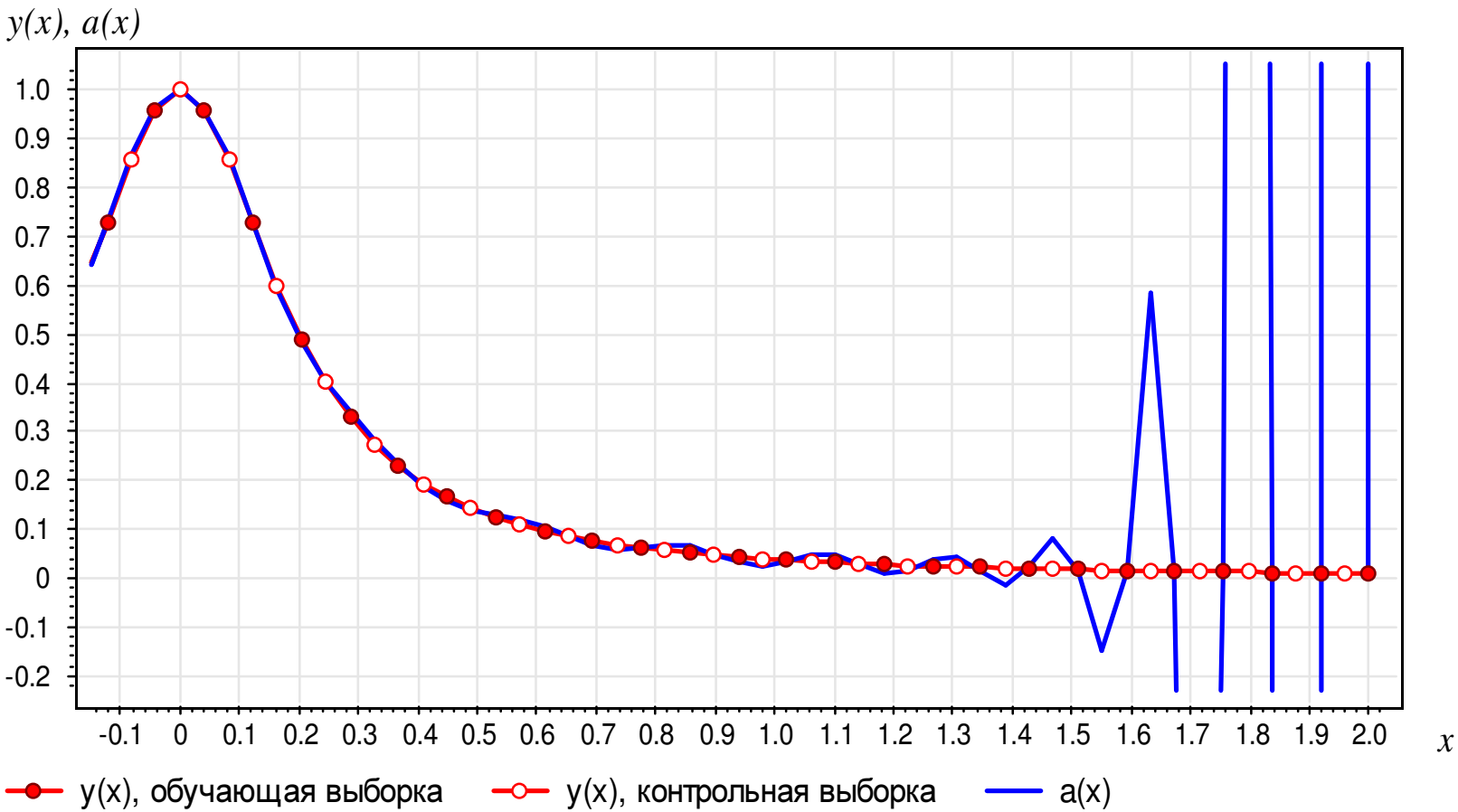
Обучающая выборка: $X^\ell = \left\{ x_i = 4\frac{i-1}{\ell-1} - 2 \;\middle|\; i = 1, \ldots, \ell \right\}$.

Контрольная выборка: $X^k = \left\{ x_i = 4\frac{i-0.5}{\ell-1} - 2 \;\middle|\; i = 1, \ldots, \ell-1 \right\}$.

Что происходит с $Q\left(a, X^\ell\right)$ и $Q\left(a, X^k\right)$ при увеличении n?

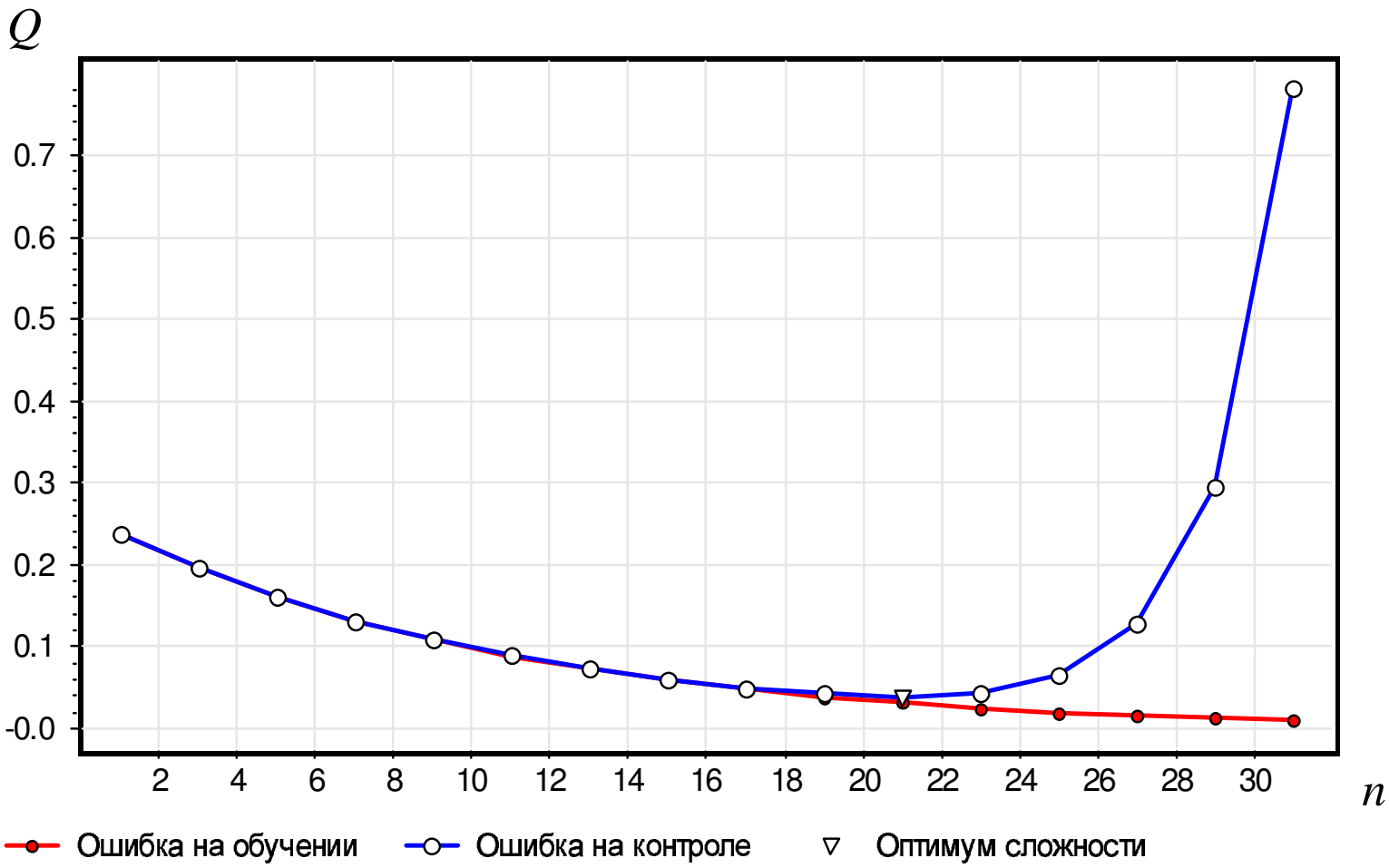
Пример переобучения: эксперимент при $\ell = 50,\ n = 38$

$$y(x) = \frac{1}{1 + 25x^2};\quad a(x) - \text{полином степени } n = 38$$



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

Переобучение — это когда $Q\big(\mu(X^\ell), X^k\big) \gg Q\big(\mu(X^\ell), X^\ell\big)$:



Эмпирические оценки обобщающей способности

Метод обучения μ по выборке $(x_i, y_i)_{i=1}^{\ell}$ строит алгоритм a.

- Среднее значение потерь на тестовых данных (hold-out):

$$\mathrm{HO}(\mu, X^{\ell}, X^{k}) = Q(\mu(X^{\ell}), X^{k}) \to \min$$

- Скользящий контроль (leave-one-out), $L = \ell + 1$:

$$\mathrm{LOO}(\mu, X^{L}) = \frac{1}{L} \sum_{i=1}^{L} \mathscr{L}(\mu(X^{L} \setminus \{x_i\}), x_i) \to \min$$

- Кросс-проверка (cross-validation) по N разбиениям, $X^{L} = X_n^{\ell} \sqcup X_n^{k}$, $L = \ell + k$:

$$\mathrm{CV}(\mu, X^{L}) = \frac{1}{N} \sum_{n=1}^{N} Q(\mu(X_n^{\ell}), X_n^{k}) \to \min$$

Причины переобучения линейных моделей

1. слишком мало объектов; слишком много признаков;
2. линейная зависимость (мультиколлинеарность) признаков:

   пусть построен классификатор: $a(x, w) = \text{sign}\langle x, w\rangle$;
   мультиколлинеарность: $\exists v \in \mathbb{R}^n$: $\forall x \;\; \langle x, v\rangle \approx 0$;
   тогда $\forall \gamma \in \mathbb{R} \quad a(x, w) \approx \text{sign}\langle x, w + \gamma v\rangle$

Последствия:

- решение неединственно и неустойчиво;
- появляются слишком большие веса $+w_j$ или $-w_j$;
- $Q(w)$ на обучении много меньше, чем на контроле;

Спасает *регуляризация* — введение дополнительного критерия:

$$\|w\|^2 = \sum_{j=1}^{n} w_j^2 \;\to\; \min.$$

Метод сокращения весов (weight decay)

Штраф за увеличение нормы вектора весов:

$$Q_\tau(w) = Q(w) + \frac{\tau}{2}\sum_{j=1}^{n} w_j^2 \to \min_w .$$

Компоненты вектора градиента:

$$\frac{\partial}{\partial w_j} Q_\tau(w) = \frac{\partial}{\partial w_j} Q(w) + \tau w_j, \quad j = 1, \ldots, n.$$

Модификация градиентного шага:

$$w_j^{t+1} := w_j^t(1 - h^t\tau) - h^t \frac{\partial}{\partial w_j} Q(w^t), \quad j = 1, \ldots, n.$$

Параметр регуляризации τ подбирается экспериментально, по качеству на контрольной выборке.

Терминология, пришедшая из медицинской диагностики

Положительный диагноз — алгоритм предсказывает болезнь
(хотя, казалось бы, что тут положительного...)

Доля больных с верным положительным диагнозом:

$$\text{чувствительность} = \frac{\sum_{i=1}^{\ell}[y_i = +1][a(x_i) = +1]}{\sum_{i=1}^{\ell}[y_i = +1]}$$

Доля здоровых с верным отрицательным диагнозом:

$$\text{специфичность} = \frac{\sum_{i=1}^{\ell}[y_i = -1][a(x_i) = -1]}{\sum_{i=1}^{\ell}[y_i = -1]}$$

Чувствительность и специфичность хотим максимизировать.

⊕ Они не зависят от соотношения мощностей классов.

⊕ Хорошо подходят для несбалансированных выборок.

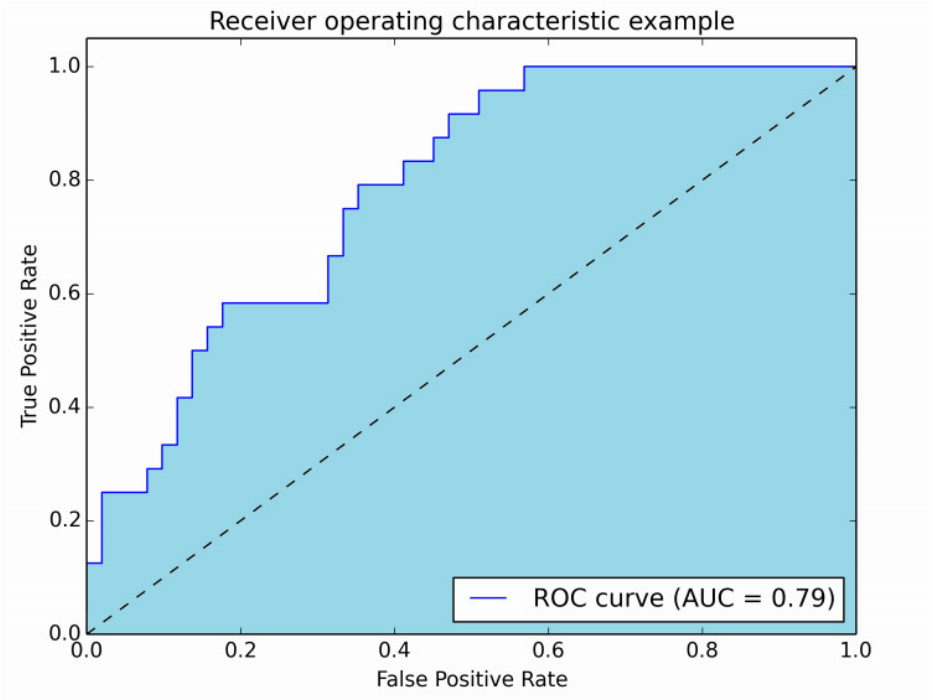
Определение ROC-кривой

Модель классификации: $a(x) = \text{sign}\big(\langle x, w\rangle - w_0\big)$

по оси X: $1 -$ специфичность $=$ FPR, False Positive Rate,

по оси Y: чувствительность $=$ TPR, True Positive Rate

Каждая точка ROC-кривой соответствует значению порога w_0
(ROC — «receiver operating characteristic»),



AUC (area under curve) — площадь под ROC-кривой

Модель классификации: $a(x) = \text{sign}\big(\langle x, w\rangle - w_0\big)$

Доля правильно упорядоченных пар объектов (x_i, x_s) из разных классов, $y_i = -1$, $y_i = +1$ (докажите!):

$$\text{AUC} = \frac{\sum_{i=1}^{\ell}\sum_{s=1}^{\ell}\big[y_i < y_s\big]\big[\langle x_i, w\rangle < \langle x_s, w\rangle\big]}{\sum_{i=1}^{\ell}\sum_{s=1}^{\ell}\big[y_i < y_s\big]}$$

Преимущества AUC:

⊕ не зависит от порога w_0, оценивает только качество w;

⊕ не зависит от численности классов;

⊕ это общепринятая мера качества классификации;

Чтобы измерить предсказательную способностью метода обучения μ, надо вычислять AUC на контрольной выборке.

Явная максимизация сглаженного AUC

Максимизация доли правильно упорядоченных пар (x_i, x_s):

$$\text{AUC} = \sum_{i,s:\ y_i<y_s} \big[\langle x_i, w\rangle < \langle x_s, w\rangle\big] \ \to \ \max_w$$

Максимизация сглаженного AUC:

$$Q(w) = \sum_{i,s:\ y_i<y_s} \mathscr{L}\Big(\underbrace{\langle x_s, w\rangle - \langle x_i, w\rangle}_{M_{is}}\Big) \ \to \ \min_w$$

$\mathscr{L}(M)$ — гладкая убывающая функция отступа,
M_{is} — новое понятие отступа, теперь для пар объектов.

Метод стохастического градиента по парам (x_i, x_s): $y_i < y_s$

$$w^{t+1} := w^t - h^t \mathscr{L}'\big(\langle x_s - x_i, w^t\rangle\big)(x_s - x_i)$$

Метод стохастического градиента для максимизации AUC

Вход: выборка $(x_i, y_i)_{i=1}^{\ell}$, $y_i \in \{-1, +1\}$, параметры h, τ;
Выход: веса $w_1, \ldots, w_n$;

инициализировать веса $w_j = \frac{\langle y, f_j \rangle}{\langle f_j, f_j \rangle}$, $j = 1, \ldots, n$;
повторять
  выбрать случайную пару обучающих объектов x_i, x_s из разных классов: $y_i = -1$, $y_s = 1$;
  выбрать величину градиентного шага h;
  выполнить градиентный шаг:
  $M_{is} := \langle x_s - x_i, w \rangle$;
  для всех $j = 1, \ldots, n$
    $w_j := w_j(1 - \tau h) + h\mathscr{L}'(M_{is})\left(x_s^j - x_i^j\right)$;
пока процесс не сойдётся куда-нибудь;

Резюме

- Обучение — это оптимизации (почти во всех методах)
- Лучшие методы классификации основаны на сглаживании пороговой функции потерь
- Метод наименьших квадратов для линейной регрессии сводится к решению системы линейных уравнений $n \times n$
- Метод стохастического градиента позволяет единообразно решать самые разные задачи, в том числе для Big Data
- Регуляризация помогает против переобучения
- Можно максимизировать непосредственно AUC

Воронцов Константин Вячеславович
voron@forecsys.ru

www.MachineLearning.ru • Участник:Vokov

Если что-то было не понятно,
не стесняйтесь подходить и спрашивать :)