Методы машинного обучения. Предобработка данных и оценивание моделей

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-25-26
орг.вопросы по курсу: ml.cmc@mail.ru

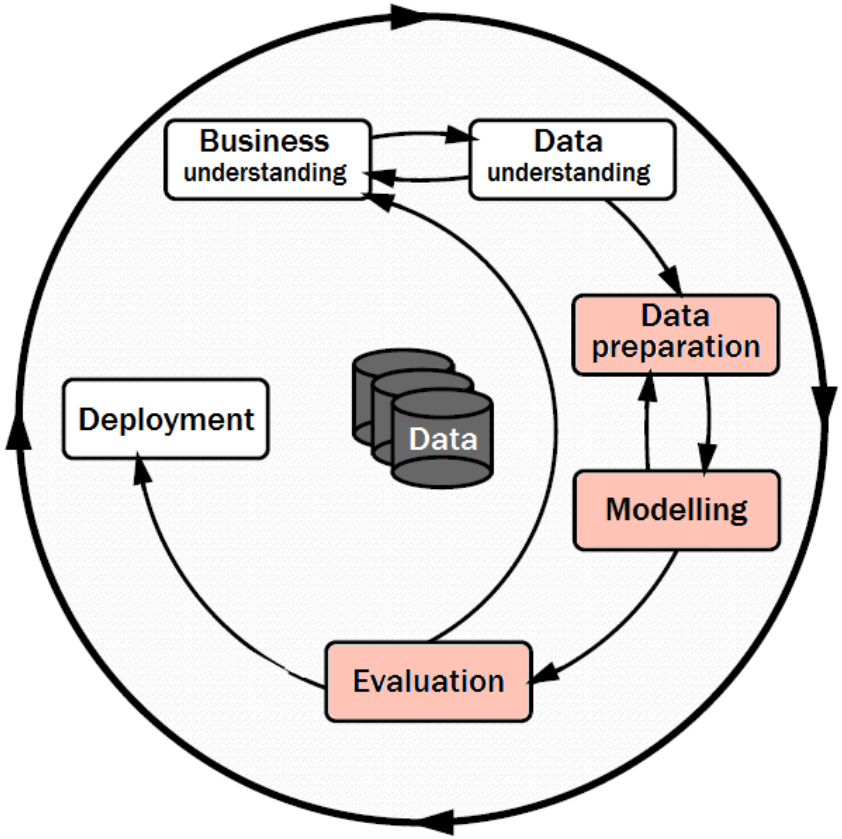
ВМК МГУ • 7 октября 2025

Содержание

Практические аспекты машинного обучения

Межотраслевой стандарт интеллектуального анализа данных

CRISP-DM (1999): CRoss Industry Standard Process for Data Mining



Компании-инициаторы:

- SPSS
- Teradata
- Daimler AG
- NCR Corp.
- OHRA

Шаги процесса:

- понимание бизнеса
- понимание данных
- предобработка данных
- **моделирование**
- оценивание моделей
- внедрение

Шкалы измерения

Измерительная шкала — множество Z допустимых значений, получаемых в результате измерения признака $f(x)$, $f: X \to Z$

Тип шкалы определяется множествами

- допустимых биективных преобразований $\psi: Z \to Z'$
- допустимых операций над значениями из шкалы Z

Классификация типов измерительных шкал по Стивенсу:

шкала	Z	$\psi(z)$	операции
логическая (boolean)	$0, 1$	биективные	$\vee \wedge \neg$
номинальная (nominal)	$< \infty$	биективные	$= \neq \in$
порядковая (ordinal)	$< \infty$	монотонные	$= \neq \in < >$
интервальная (interval)	$\mathbb{R}$	$az + b$	$< > + -$
отношений (ratio)	$\mathbb{R}$	az	$< > + - \times \div$
абсолютная (absolute)	$\mathbb{R}$	z	любые

S.S.Stevens. On the Theory of Scales of Measurement // Science, 1946.

Примеры величин, измеряемых в различных шкалах

- **Логическая**
  наличие/отсутствие свойства, ответ «да/нет»
- **Номинальная** (можно переименовать или перенумеровать)
  идентификаторы классов, людей, регионов, фирм, товаров
- **Порядковая** (порядок частичный или линейный)
  уровень образования, тяжесть болезни, степень согласия
- **Ранговая** (частный случай порядковой: $1, 2, 3, \ldots, N$)
  оценка в баллах, шкалы Рихтера, Бофорта, Мооса, Бека
- **Интервальная** (можно сдвигать положение нуля)
  время, географическая широта, температура (°C, °F)
- **Отношений** (можно менять единицы измерения)
  масса, скорость, объём, сила, давление, заряд, яркость, °K
- **Абсолютная**
  число предметов, частота события, оценка вероятности

Ослабление шкалы

Номинальный $\to$ **много бинарных** (one-hot-encoding):

- $f_v(x) = \left[f(x) = v\right]$, индикатор значения v признака
- $f_A(x) = \left[f(x) \in A\right]$, индикатор подмножества A значений

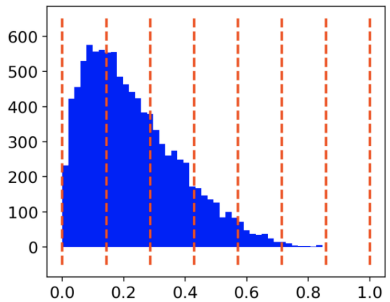
Числовой или порядковый $\to$ **бинарный:**

- $f_{a,b}(x) = \left[a \leqslant f(x) \leqslant b\right]$ для заданного отрезка $[a, b]$

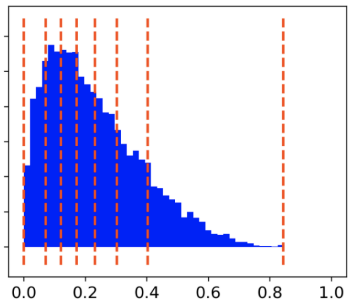
Числовой $\to$ **ранговый** (data binning, quantization):

- $f_a(x) = \sum_{k=1}^{K} \left[f(x) \geqslant a_k\right]$, номер интервала сетки $a_1, \ldots, a_K$

равномерная сетка



квантильная сетка



Ослабление шкалы всегда влечёт потерю информации

Усиление шкалы

Номинальный $\to$ числовой:

- категория заменяется частотой:
$$\tilde{f}(x) = \frac{1}{\ell}\sum_{i=1}^{\ell}\big[f(x_i) = f(x)\big]$$
- условное среднее числового признака $g(x)$:
$$\tilde{f}(x) = \text{mean}\big(g|f(x)\big) = \frac{\sum_{i=1}^{\ell} g(x_i)\big[f(x_i) = f(x)\big]}{\sum_{i=1}^{\ell}\big[f(x_i) = f(x)\big]},$$
- условное среднее целевой величины $y(x)$:
$\tilde{f}(x) = \text{mean}\big(y|f(x)\big)$, возможно переобучение!

Порядковый $\to$ числовой (монотонное преобразование)

- значение заменяется частотой:
$$\tilde{f}(x) = \frac{1}{\ell}\sum_{i=1}^{\ell}\big[f(x_i) \leqslant f(x)\big]$$

Нормализация и стандартизация числовых шкал

Многие методы накапливают меньше вычислительных погрешностей, если признаки приведены к одному масштабу

- $\tilde{f}_j(x) = \dfrac{f_j(x) - f_j^{\min}}{f_j^{\max} - f_j^{\min}}$ — нормализация, приведение к $[0, 1]$
- $\tilde{f}_j(x) = \dfrac{f_j(x)}{|f_j|^{\max}}$ — масштабирование с сохранением нуля
- $\tilde{f}_j(x) = \dfrac{f_j(x) - \mu_j}{\sigma_j}$ — стандартизация

$f_j^{\max}$, $|f_j|^{\max}$, $f_j^{\min}$, μ_j, σ_j определяются по обучающей выборке

Для повышения устойчивости к выбросам можно отбрасывать 5% наименьших и наибольших значений признака

Трансформация вида распределения

F_j — функция распределения (c.d.f.) признака f_j
Эмпирическая функция распределения (кусочно-постоянная):

$$\hat{F}_j(z) = \frac{1}{\ell} \sum_{i=1}^{\ell} \big[f_j(x_i) \leqslant z \big]$$

- $\tilde{f}_j(x) = F_j(f_j(x))$ — преобразование $f_j(x)$ в равномерную на отрезке $[0, 1]$ случайную величину
- $\tilde{f}_j(x) = \Phi^{-1}\big(F_j(f_j(x))\big)$ — преобразование $f_j(x)$ в случайную величину с заданной функцией распределения Φ (например, в нормальную)
- $\tilde{f}_j(x) = \ln\big(1 + f_j(x)\big)$ — преобразование неотрицательной случайной величины «с тяжёлым правым хвостом» (объёмы производства, перевозок, продаж)

Подходы к обработке пропущенных значений

- Игнорировать объекты или признаки с пропусками
  — ведёт к потере информации :(
- Заполнить пропущенные значения признака f:
  — средним или медианным значением $\bar{f}$
- Прогнозировать значения признака f по остальным:
  — регрессия для вещественного признака f
  — классификация для дискретного признака f
  — матричные разложения, например, разреженный SVD
- Использовать модели, способные обрабатывать пропуски:
  — решающие деревья
  — голосование низкоразмерных базовых предикторов
- Ввести бинарный признак $\tilde{f}(x) = [f(x) \text{ не известно}]$

Непараметрическая регрессия для заполнения пропусков

Формула Надарая–Ватсона, ядерное сглаживание:

$$\hat{f}_j(x_i) = \frac{\sum_u f_j(u) S(u, x_i)}{\sum_u S(u, x_i)}$$

где $\sum_u$ — сумма по всем объектам $u \in X^\ell$ с известным $f_j(u)$

Возможные конструкции функций сходства $S(u, x)$:

- $S(u,x) = K\left(\frac{\rho(u,x)}{h}\right), \quad \rho^2(u,x) = \frac{1}{|J_{ux}|} \sum_{j \in J_{ux}} \big(f_j(u) - f_j(x)\big)^2$
- $S(u,x) = \frac{1}{|J_{ux}|} \sum_{j \in J_{ux}} f_j(u) f_j(x)$ — скалярное произведение
- $S(u,x) = \frac{\sum_{j \in J_{ux}} f_j(u) f_j(x)}{\sqrt{\sum_{j \in J_{ux}} f_j^2(u)} \sqrt{\sum_{j \in J_{ux}} f_j^2(x)}}$ — косинусная ф.сх.

где J_{ux} — множество признаков j с известными $f_j(x)$ и $f_j(u)$

Разреженное низкоранговое матричное разложение

Дано: матрица $F = \big(f_{ij} = f_j(x_i)\big)_{\ell \times n}$, $\Omega \subseteq \{1, \ldots, \ell\} \times \{1, \ldots, n\}$

Найти: матрицы $G = (g_{it})_{\ell \times k}$ и $U = (u_{jt})_{n \times k}$ такие, что

$$\|F - GU^{\mathsf{T}}\| = \sum_{(i,j) \in \Omega} \underbrace{\big(f_{ij} - \langle g_i, u_j \rangle\big)}_{\varepsilon_{ij}}^2 = \sum_{(i,j) \in \Omega} \Big(f_{ij} - \sum_{t=1}^{k} g_{it} u_{jt}\Big)^2 \to \min_{G,U}$$

Классический SVD неприменим для разреженной задачи.

Метод стохастического градиента: перебираем $(i,j) \in \Omega$ в случайном порядке, делаем градиентные шаги $(\varepsilon_{ij})^2 \to \min\limits_{g_i, u_j}$:

$$g_{it} := g_{it} + \eta \varepsilon_{ij} u_{jt}, \quad t = 1, \ldots, k$$
$$u_{jt} := u_{jt} + \eta \varepsilon_{ij} g_{it}, \quad t = 1, \ldots, k$$

$\hat{f}_j(x_i) = \langle g_i, u_j \rangle$ — восстановление пропущенных значений

g_{it} — новые признаки x_i в пространстве размерности k

Классические подходы к конструированию признаков

Feature Engineering: признаки вычисляются по формулам, которые зависят от задачи, требуют изобретательности и знаний предметной области. Долго, дорого.

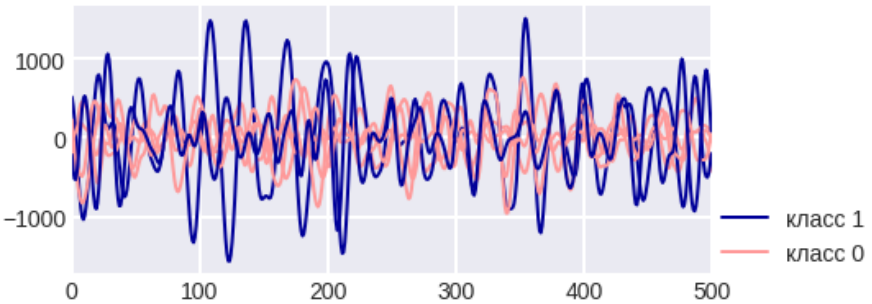
Примеры:

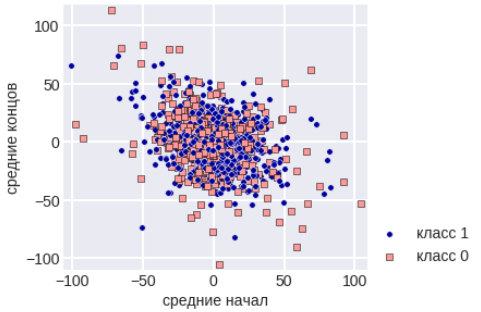
- Прогнозирование временных рядов:
  признаки агрегируются по предыстории различной глубины
- Распознавание лиц:
  признаки размера и формы черт лица
- Классификация и поиск текстов:
  признаки частоты слов, терминов, названий, синонимов
- Распознавание речи:
  спектральные, фонетические, лингвистические признаки

Иногда удачные признаки решают задачу без ML

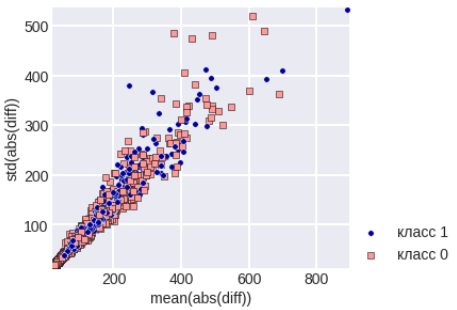
Соревнование «Ford Classification Challenge» (2008)
Задача детектирования поломок по сигналу датчика

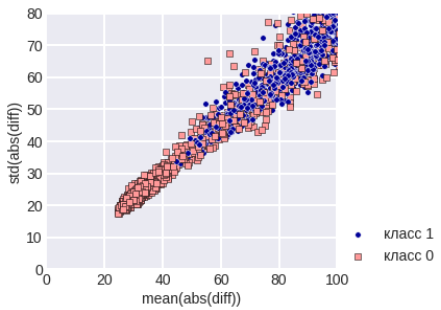
Признаки, генерируемые по исходным временным рядам, слабы:

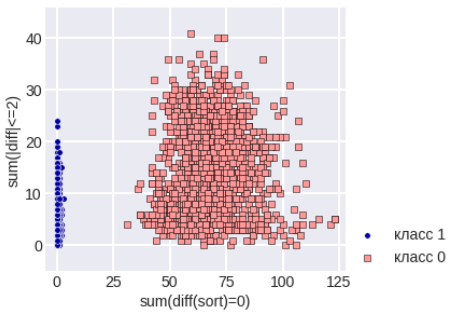




Среди признаков рядов их производных оказывается идеальный:



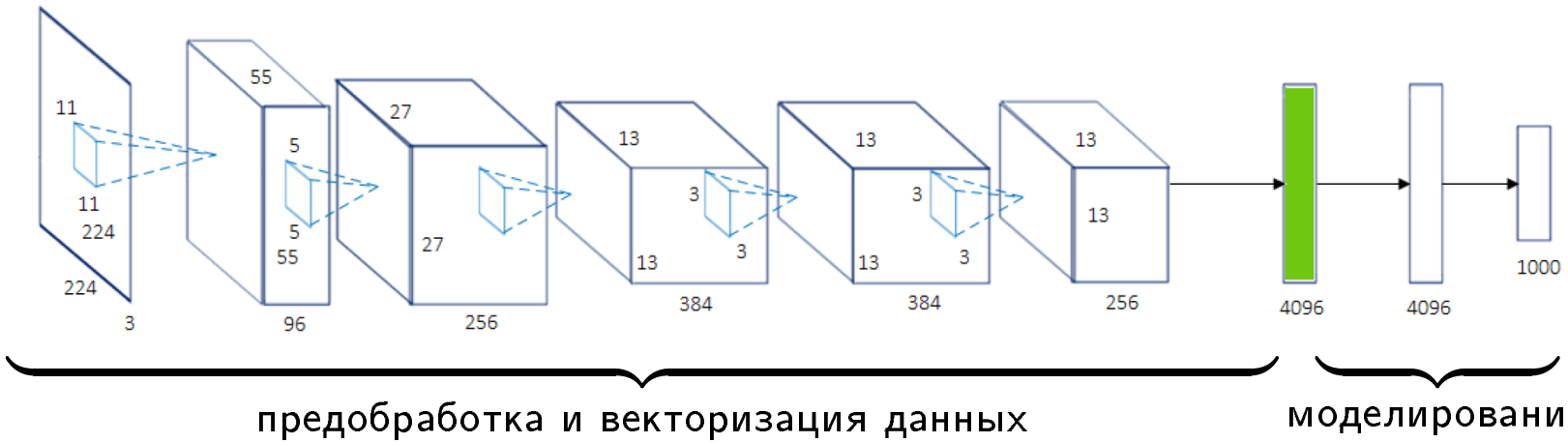




https://dyakonov.org/2018/06/28/простые-методы-анализа-данных

Обучаемая векторизация данных

Глубокие нейронные сети объединяют два этапа обработки данных: векторизацию и предсказательное моделирование



- компьютерное зрение
- обработка текстов естественного языка
- анализ сигналов, распознавание и синтез речи
- анализ графов и транзакционных данных

— немного в следующем семестре, много в курсе DL

Krizhevsky A., Sutskever I., Hinton G. ImageNet classification with deep convolutional neural networks. 2012.

Резюме. Предварительная обработка данных

Данные могут быть

- разнородные (признаки измерены в разных шкалах)
- неполные (измерены не все, имеются пропуски)
- неточные (измерены с погрешностями)
- противоречивые (объекты одинаковые, ответы разные)
- избыточные (сверхбольшие, не помещаются в память)
- недостаточные (объектов меньше, чем признаков)
- сложно структурированные (нет признаковых описаний)

— и для всех этих проблем в ML известны решения!

«Грязные» данные — единственная проблема, для которой нет решения кроме «снова улучшать процессы сбора данных»

Анализ ошибок классификации

Задача бинарной классификации: $y_i,\ a(x_i) \in \{-1, +1\}$.

	модель классификации	учитель
TP, True Positive	$a(x_i) = +1$	$y_i = +1$
TN, True Negative	$a(x_i) = -1$	$y_i = -1$
FP, False Positive	$a(x_i) = +1$	$y_i = -1$
FN, False Negative	$a(x_i) = -1$	$y_i = +1$

FP: ложноположительно, ошибка I рода, «ложная тревога»
FN: ложноотрицательно, ошибка II рода, «пропуск цели»

Правильность классификации (чем больше, тем лучше):

$$\text{Accuracy} = \frac{1}{\ell}\sum_{i=1}^{\ell}\big[a(x_i) = y_i\big] = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{FN} + \text{TP} + \text{TN}}$$

Недостаток: не учитывается дисбаланс численности классов, а также различие цены ошибки I и II рода.

ROC-кривая (Receiver Operating Characteristic)

Модель бинарной классификации: $a(x; w, w_0) = \text{sign}\big(g(x, w) - w_0\big)$
Кривая ROC: как меняется качество при варьировании w_0
(чем больше w_0, тем больше x_i, на которых $a(x_i) = -1$)

- по оси X: доля *ошибочных положительных классификаций* (FPR — false positive rate):

$$\text{FPR}(a) = \frac{\text{FP}}{\text{FP} + \text{TN}} = \frac{\sum_{i=1}^{\ell} [y_i = -1][a(x_i; w, w_0) = +1]}{\sum_{i=1}^{\ell} [y_i = -1]};$$

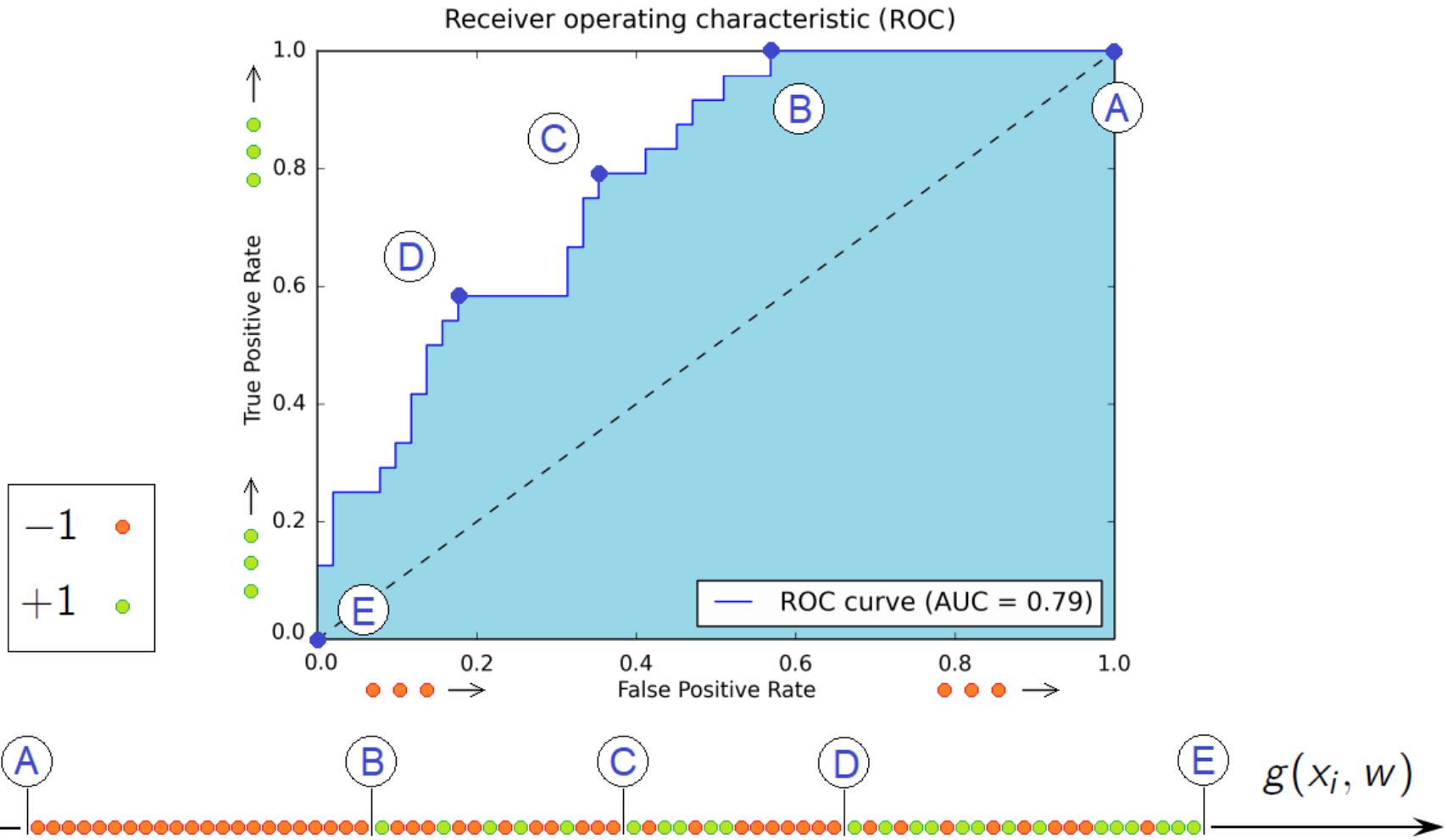
  $1 - \text{FPR}(a)$ называется *специфичностью* алгоритма a.

- по оси Y: доля *правильных положительных классификаций* (TPR — true positive rate):

$$\text{TPR}(a) = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\sum_{i=1}^{\ell} [y_i = +1][a(x_i; w, w_0) = +1]}{\sum_{i=1}^{\ell} [y_i = +1]};$$

  $\text{TPR}(a)$ называется также *чувствительностью* алгоритма a.

ROC-кривая и площадь под кривой AUC (Area Under Curve)



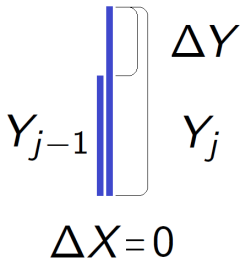
ABCDE — положения порога w_0 на оси значений функции g

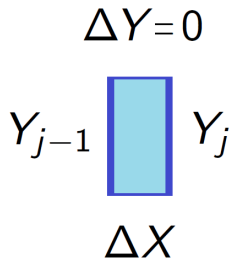
Алгоритм эффективного построения ROC-кривой

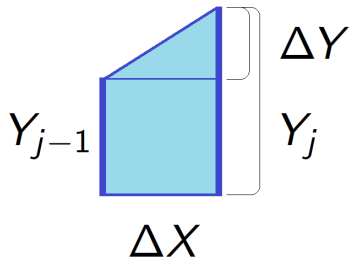
Вход: выборка $\{x_i\}_{i=1}^{\ell}$; дискриминантная функция $g(x, w)$;
Выход: ROC-кривая $(X_j, Y_j)_{j=0}^{k}$, $k \leqslant \ell$ и площадь AUC

$\ell_y := \sum_{i=1}^{\ell} [y_i = y]$, для всех $y \in Y$;
упорядочить $\{x_i\}$ по убыванию $g_i = g(x_i, w)$: $g_1 \geqslant \ldots \geqslant g_\ell$;
$(X_0, Y_0) := (0, 0)$; AUC $:= 0$; $\Delta X := 0$; $\Delta Y := 0$; $j := 1$;

для $i := 1, \ldots, \ell$
- $\Delta X := \Delta X + \frac{1}{\ell_-}[y_i = -1]$;
- $\Delta Y := \Delta Y + \frac{1}{\ell_+}[y_i = +1]$;
- **если** $(g_i \neq g_{i-1})$ **то**
  - $X_j := X_{j-1} + \Delta X$;
  - $Y_j := Y_{j-1} + \Delta Y$;
  - AUC $:=$ AUC $+ \frac{1}{2}(Y_{j-1} + Y_j)\Delta X$;
  - $j := j + 1$; $\Delta X := 0$; $\Delta Y := 0$;







Задача максимизации площади под кривой ROC-AUC

Модель классификации: $a(x_i, w, w_0) = \text{sign}\big(g(x_i, w) - w_0\big)$

AUC — это доля правильно упорядоченных пар (x_i, x_j):

$$\text{AUC}(w) = \frac{1}{\ell_-}\sum_{i=1}^{\ell}\big[y_i = -1\big]\text{TPR}_i =$$

$$= \frac{1}{\ell_-\ell_+}\sum_{i=1}^{\ell}\sum_{j=1}^{\ell}\big[y_i < y_j\big]\big[g(x_i, w) < g(x_j, w)\big] \to \max_w$$

Критерий: максимум аппроксимированного AUC:

$$1 - \text{AUC}(w) \leqslant Q(w) = \sum_{i,j:\, y_i<y_j} L\underbrace{\big(g(x_j, w) - g(x_i, w)\big)}_{M_{ij}(w)} \to \min_w$$

где $L(M)$ — убывающая функция попарного отступа $M_{ij}(w)$
SG: градиентные шаги по парам объектов (x_i, x_j): $y_i < y_j$

Алгоритм SG для максимизации AUC

Возьмём для простоты линейный классификатор:

$$g(x, w) = \langle x, w \rangle, \qquad M_{ij}(w) = \langle x_j - x_i, w \rangle, \qquad y_i < y_j.$$

Вход: выборка X^ℓ, темп обучения h, темп забывания λ;
Выход: вектор весов w;

инициализировать веса w_j, $j = 0, \ldots, n$;
инициализировать оценку: $\bar{Q} := \frac{1}{\ell_+ \ell_-} \sum_{i,j} [y_i < y_j]\, L(M_{ij}(w))$;

повторять
- выбрать пару объектов (i, j): $y_i < y_j$, случайным образом;
- вычислить потерю: $\varepsilon_{ij} := L(M_{ij}(w))$;
- сделать градиентный шаг: $w := w - h\, L'(M_{ij}(w))(x_j - x_i)$;
- оценить функционал: $\bar{Q} := (1 - \lambda)\bar{Q} + \lambda \varepsilon_{ij}$;

пока значение $\bar{Q}$ и/или веса w не сойдутся;

Логарифм правдоподобия, log-loss

Вероятностная модель бинарной классификации, $y_i \in \{-1, +1\}$:

$$a(x, w) = \text{sign}\big(g(x, w) - w_0\big), \qquad g(x, w) = \text{P}(y = +1 | x, w).$$

Проблема: ROC и AUC инвариантны относительно монотонных преобразований дискриминантной функции $g(x, w)$.

Критерий логарифма правдоподобия (log-loss):

$$Q(w) = \sum_{i=1}^{\ell} [y_i = +1] \ln g(x, w) + [y_i = -1] \ln\big(1 - g(x, w)\big) \to \max_w$$

Вероятностная модель многоклассовой классификации:

$$a(x) = \arg\max_{y \in Y} \text{P}(y | x, w);$$

$$Q(w) = \sum_{i=1}^{\ell} \ln \text{P}(y_i | x_i, w) \to \max_w$$

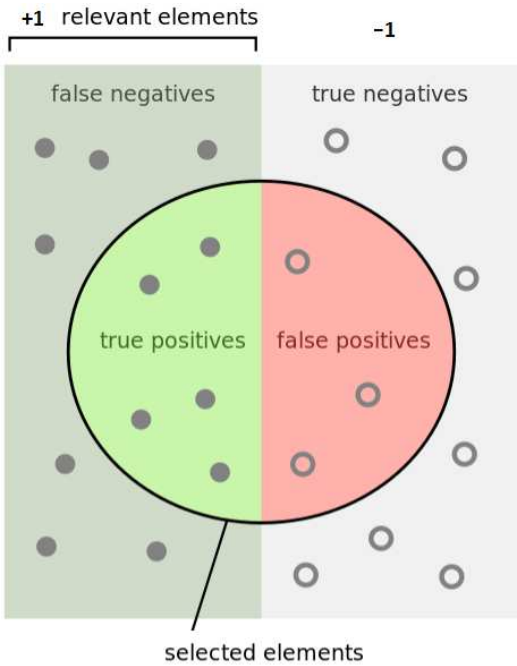
Точность и полнота бинарной классификации

В информационном поиске не важен TN:

$$\text{Точность, Precision} = \frac{TP}{TP+FP}$$

$$\text{Полнота, Recall} = \frac{TP}{TP+FN}$$

Precision — доля релевантных среди найденных
Recall — доля найденных среди релевантных

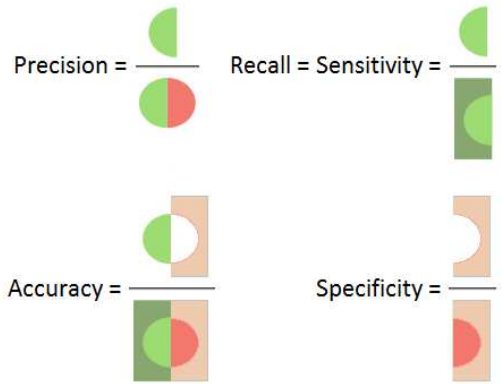


В медицинской диагностике:

$$\text{Чувствительность, Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Специфичность, Specificity} = \frac{TN}{TN+FP}$$

Sensitivity — доля верных положительных диагнозов
Specificity — доля верных отрицательных диагнозов



Точность и полнота многоклассовой классификации

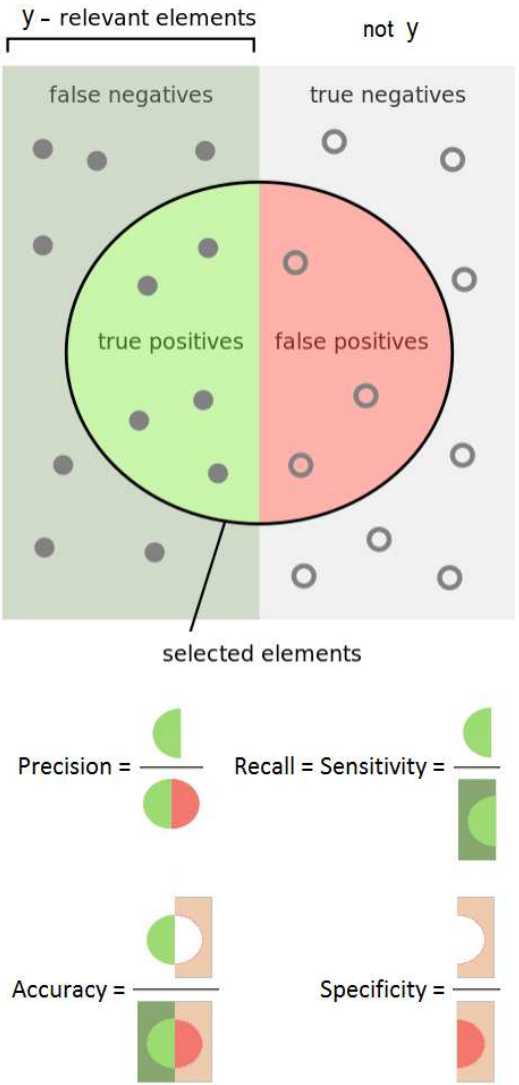
Для каждого класса $y \in Y$:
TP_y — верные положительные
FP_y — ложные положительные
FN_y — ложные отрицательные

Точность и полнота с микроусреднением:

$$\text{Precision: } P = \frac{\sum_y \mathrm{TP}_y}{\sum_y (\mathrm{TP}_y + \mathrm{FP}_y)};$$

$$\text{Recall: } R = \frac{\sum_y \mathrm{TP}_y}{\sum_y (\mathrm{TP}_y + \mathrm{FN}_y)};$$

Микроусреднение не чувствительно
к ошибкам на малочисленных классах



Точность и полнота многоклассовой классификации

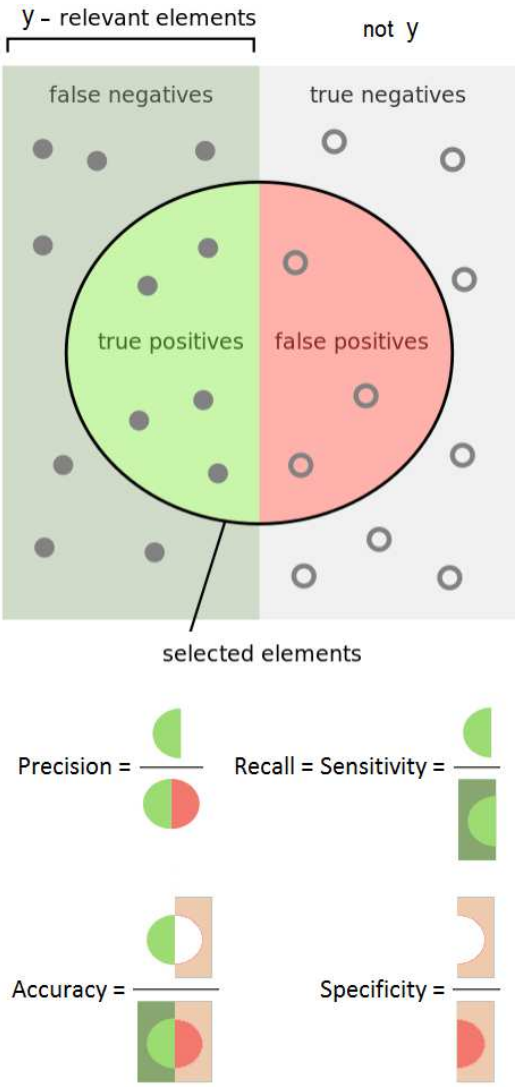
Для каждого класса $y \in Y$:
TP_y — верные положительные
FP_y — ложные положительные
FN_y — ложные отрицательные

Точность и полнота с макроусреднением:

$$\text{Precision: } P = \frac{1}{|Y|} \sum_y \frac{\mathrm{TP}_y}{\mathrm{TP}_y + \mathrm{FP}_y};$$

$$\text{Recall: } R = \frac{1}{|Y|} \sum_y \frac{\mathrm{TP}_y}{\mathrm{TP}_y + \mathrm{FN}_y};$$

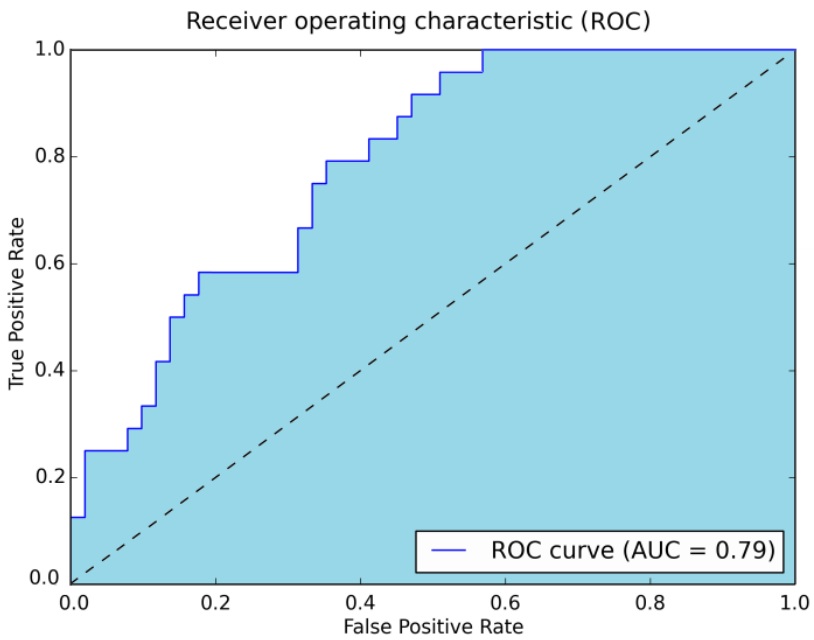
Макроусреднение чувствительно
к ошибкам на малочисленных классах

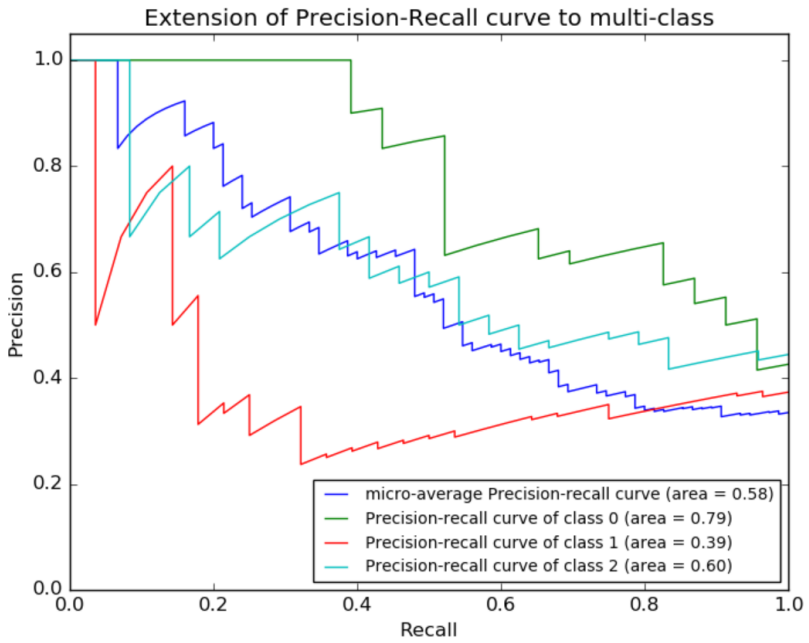


Кривые ROC и Precision-Recall

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

Каждая точка кривой соответствует значению порога w_0





AUROC — площадь под ROC-кривой

AUPRC — площадь под кривой Precision-Recall

Примеры из Python scikit learn: `http://scikit-learn.org/dev`

Резюме. Оценки качества классификации

- Чувствительность и специфичность лучше подходят для задач с несбалансированными классами
- Логарифм правдоподобия (log-loss) лучше подходит для оценки качества вероятностной модели классификации.
- Точность и полнота лучше подходят для задач поиска, когда доля объектов релевантного класса очень мала.

Агрегированные оценки:

- AUC лучше подходит для оценивания качества, когда соотношение цены ошибок не фиксировано.
- AUPRC — площадь под кривой точность–полнота.
- $F_1 = \frac{2PR}{P+R}$ — F-мера, другой способ агрегирования P и R.
- $F_\beta = \frac{(1+\beta^2)PR}{\beta^2 P+R}$ — F_β-мера: чем больше β, тем важнее R.

Задачи оценивания и выбора моделей

Дано:

$X^\ell = (x_1, \ldots, x_\ell)$ — обучающая выборка

$A_t = \{a \colon X \times W_t \to Y\}$ — параметрические модели, $t \in T$

W_t — пространство параметров модели A_t

$\mu_t \colon (X \times Y)^\ell \to W_t$ — методы обучения, $t \in T$

Найти: метод μ_t с наилучшей *обобщающей способностью*.

Частные случаи:

- выбор лучшей модели A_t (model selection);
- выбор метода обучения μ_t для заданной модели A (в частности, оптимизация *гиперпараметров*);
- отбор признаков (feature selection):
  $F = \{f_j \colon X \to D_j \colon j = 1, \ldots, n\}$ — множество признаков;
  метод обучения μ_J использует только признаки $J \subseteq F$.

Внутренние и внешние критерии качества обучения

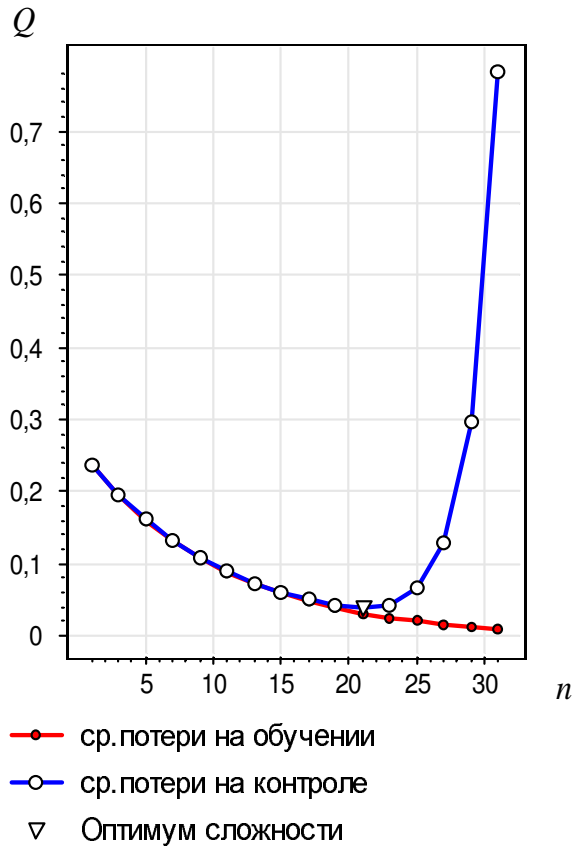
$\mathscr{L}(w, x)$ — функция потерь модели $a(w, x)$ на объекте x

$Q(w, X^\ell) = \frac{1}{\ell} \sum_{i=1}^{\ell} \mathscr{L}(w, x_i)$ — *функционал качества* $a(w, x)$ на X^ℓ

$Q\big(\mu(X^\ell), X^\ell\big)$ — *внутренний критерий*,
убывает с ростом сложности модели
(числа обучаемых параметров),
смещенная оценка $\mathsf{E}_{X^\ell, x} \mathscr{L}\big(\mu(X^\ell), x\big)$

$Q\big(\mu(X^\ell), X^k\big)$ — *внешний критерий*,
имеет минимум по сложности модели,
несмещённая оценка $\mathsf{E}_{X^\ell, x} \mathscr{L}\big(\mu(X^\ell), x\big)$

Недостаток оценки hold-out:
она зависит от разбиения $X^\ell \sqcup X^k$



Кросс-проверка (cross-validation, CV)

Усреднение по множеству разбиений $X^L = X^\ell_s \sqcup X^k_s$, $s \in S$:

$$\mathrm{CV}(\mu, X^L) = \frac{1}{|S|} \sum_{s \in S} Q\big(\mu(X^\ell_s), X^k_s\big)$$

- $|S| = 1$ — единственное (случайное) разбиение: *hold-out*
- S — множество случайных разбиений: *метод Монте-Карло*
- $S = \big\{(X^L \setminus \{x_i\}) \sqcup \{x_i\}\big\}_{i=1..L}$, каждый объект становится контролем один раз, *скользящий контроль* (*leave one out*)
- $S = \big\{(X^L \setminus B_s) \sqcup B_s\big\}_{s=1..q}$, где $B_1 \sqcup \cdots \sqcup B_q = X^L$ — разбиение *на q блоков* равной± 1 длины (q-fold CV), каждый объект участвует в контроле один раз
- $S = \big\{(X^L \setminus B^r_s) \sqcup B^r_s\big\}^{r=1..t}_{s=1..q}$, где $B^r_1 \sqcup \cdots \sqcup B^r_q = X^L$ — t разбиений *на q блоков* равной± 1 длины ($t \times q$-fold CV), каждый объект участвует в контроле ровно t раз
- S — все $C^k_{\ell+k}$ разбиений: *complete cross-validation*, CCV

Методология анализа ошибок (или потерь)

$\mathscr{L}(w, x_i)$ — функция потерь (чем меньше, тем лучше).
Среднее потерь на выборке U и эмпирическое распределение:

$$Q(w, U) = \frac{1}{|U|} \sum_{x_i \in U} \mathscr{L}(w, x_i)$$

$$F(\lambda; w, U) = \frac{1}{|U|} \sum_{x_i \in U} \left[\mathscr{L}(w, x_i) \leqslant \lambda\right]$$

Анализ потерь на обучающей выборке:

- Ранжировать объекты по убыванию потерь $\mathscr{L}_i = \mathscr{L}(w, x_i)$
- Объекты со сверхбольшими потерями — выбросы?
- Если нет, то как улучшить модель на этих объектах?

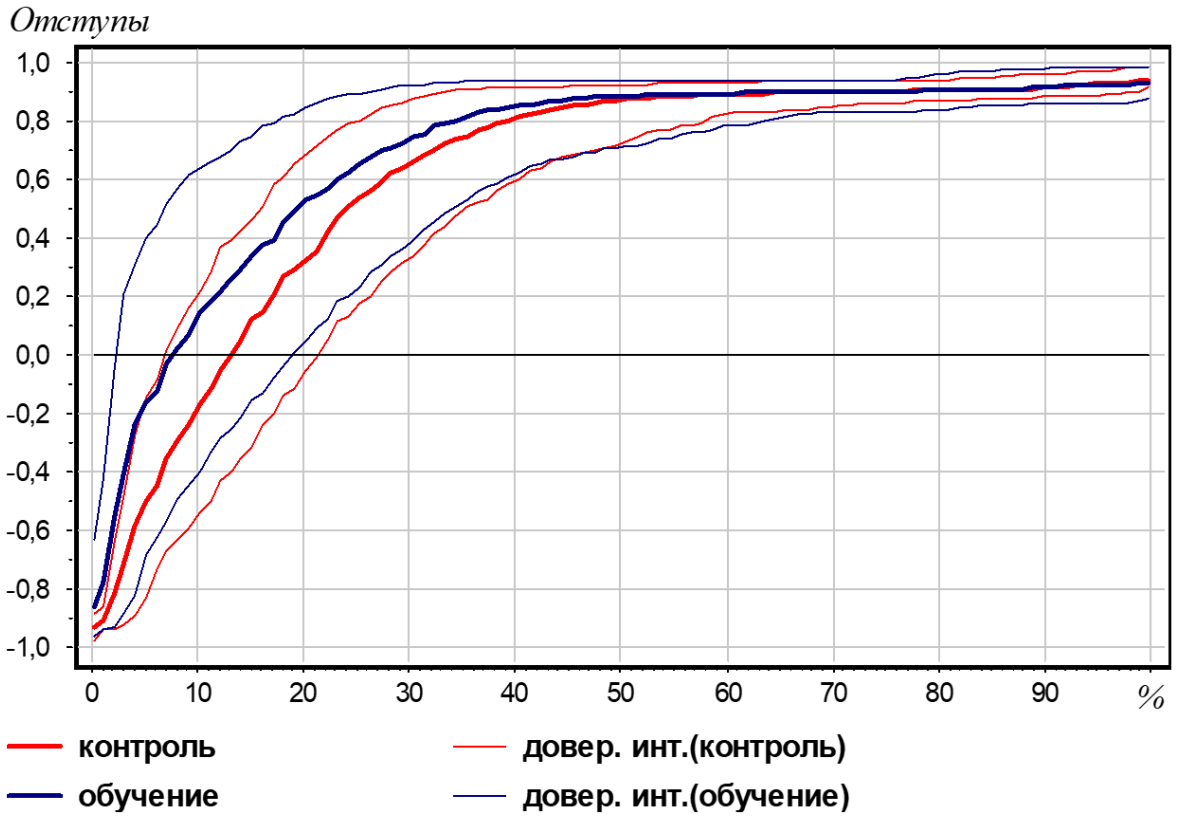
Сравнительный анализ потерь на обучении и тесте:

- Сильно ли отличаются распределения потерь?
- Если сильно, то как устранить переобучение?

Анализ распределения отступов в задаче классификации

Вместо потерь $\mathscr{L}_i = L(M_i)$ можно ранжировать отступы M_i

Видно: переобучение, зону неуверенной классификации

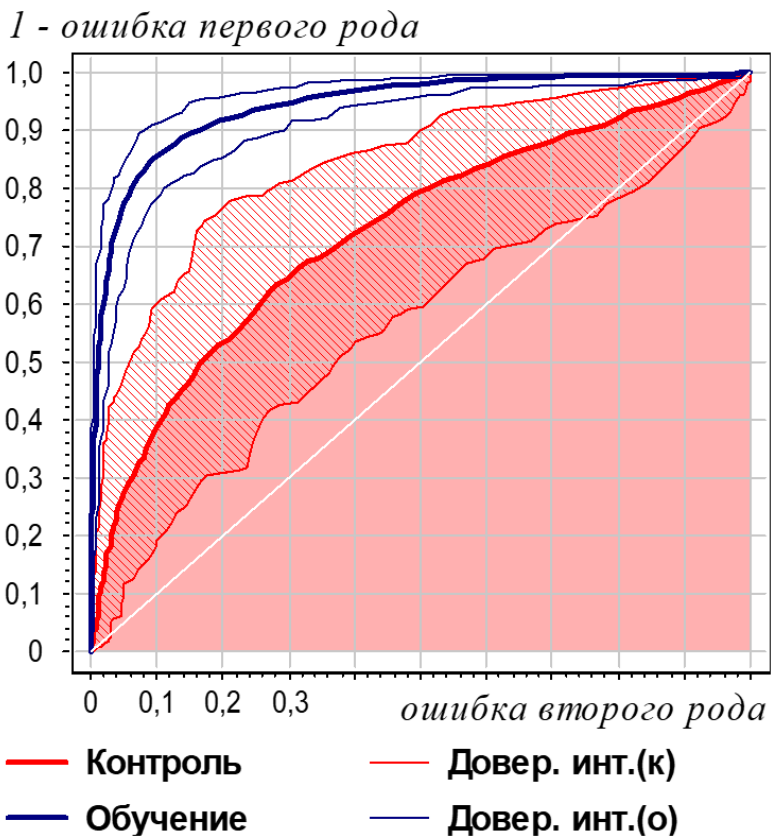


Задача UCI:australian, метод JRip

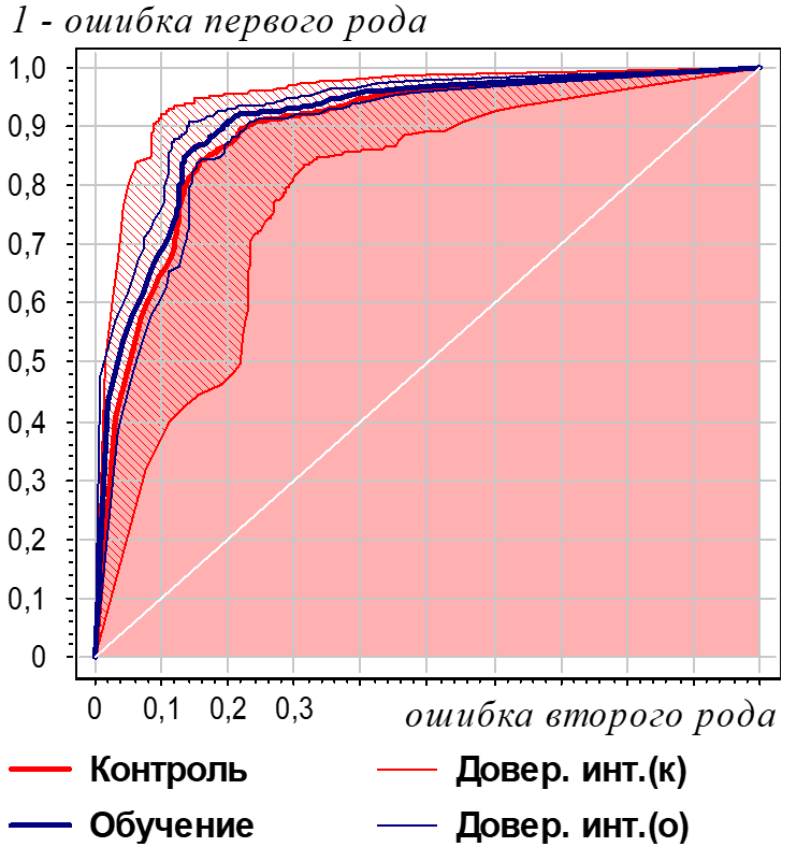
Анализ ROC-кривых

ROC-кривые можно строить отдельно для каждого класса

Видно: переобучение, устойчивость, различия классов



Задача UCI:liver, метод Bagging



Задача UCI:heart, метод Naïve Bayes

A/B тестирование (A/B testing, Split Testing)

Две модели, «базовая A» и «улучшенная B»,
построенные по историческим данным X^ℓ,
тестируются по метрике качества Q на новых данных X^k

В чём отличия A/B тестирования от обычного hold-out?

- X^k — это именно будущие данные (out-of-time), а не часть прошлых данных, исключённых из обучения (out-of-sample)
- больше реализма: за это время могут измениться свойства потока данных, реальные данные не обязаны быть i.i.d.
- однократный выбор модели почти не переобучается
- накопление данных X^k может потребовать много времени
- работа модели может влиять на формирование потока данных (например, в рекомендательных системах)

Мета-обучение (meta-learning, learning to learn)

Проблема: слишком много методов, слишком долго запускать

Дано: выборка «задача, метод» $\to$ критерии качества

Найти: модель многоклассовой классификации, предсказывающую, каким методом решать задачу

Критерий: точность предсказания оптимального метода

Признаки:

- размерные характеристики задачи
- характеристики пространства признаков: типы, выбросы, пропуски, корреляции
- результаты быстрых низкоразмерных методов

Joaquin Vanschoren. Meta-learning Architectures: Collecting, Organizing and Exploiting Meta-knowledge. 2009.

Joaquin Vanschoren. Meta-Learning: A Survey. 2018.

Автоматический выбор моделей и гиперпараметров (AutoML)

Проблема:
подбор структуры модели (архитектуры нейросети)
и гиперпараметров требует слишком много ресурсов

Дано: выборка «задача, структура» $\to$ критерии качества

Найти: какой следующий эксперимент провести с моделью

Критерий:
минимизация затрат ресурсов на автоматический поиск оптимальной модели, сопоставимой по качеству с моделями, построенными профессиональными исследователями

Близкая классическая задача — *планирование экспериментов*

Xin He et al. AutoML: A Survey of the State-of-the-Art. 2019

https://github.com/sberbank-ai-lab/LightAutoML — AutoML от Сбербанка

Резюме. Анализ ошибок и выбор моделей

- Культура анализа данных:
  — смотреть на данные глазами
  — пробовать нетривиальные идеи предобработки, основанные на знаниях предметной области
  — использовать анализ ошибок и визуализацию
  — креативно порождать и оценивать больше гипотез
  — знать и учитывать сильные и слабые стороны методов
- Автоматизация распространяется по схеме CRISP-DM, в перспективе нас ожидает бесшовная интеграция этапов
  — предобработки данных
  — моделирования
  — оценивания и выбора моделей
  — внедрения