Логические алгоритмы классификации

К. В. Воронцов, А.В. Зухба
vokov@forecsys.ru
a__l@mail.ru

март 2016

Содержание

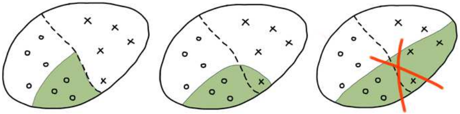
Логическая закономерность

$X^\ell = (x_i, y_i)_{i=1}^{\ell} \subset X \times Y$ — обучающая выборка, $y_i = y(x_i)$.

Логическая закономерность (правило, rule) — это предикат $R\colon X \to \{0, 1\}$, удовлетворяющий двум требованиям:

1. *интерпретируемость*:
   1) R записывается на естественном языке;
   2) R зависит от небольшого числа признаков (1–7);
2. *информативность* относительно одного из классов $c \in Y$:
   $p_c(R) = \#\bigl\{x_i\colon R(x_i){=}1 \text{ и } y_i{=}c\bigr\} \to \max;$
   $n_c(R) = \#\bigl\{x_i\colon R(x_i){=}1 \text{ и } y_i{\neq}c\bigr\} \to \min;$

Если $R(x) = 1$, то говорят «R выделяет x» (R covers x).



Требование интерпретируемости

1) $R(x)$ записывается на естественном языке;
2) $R(x)$ зависит от небольшого числа признаков (1–7);

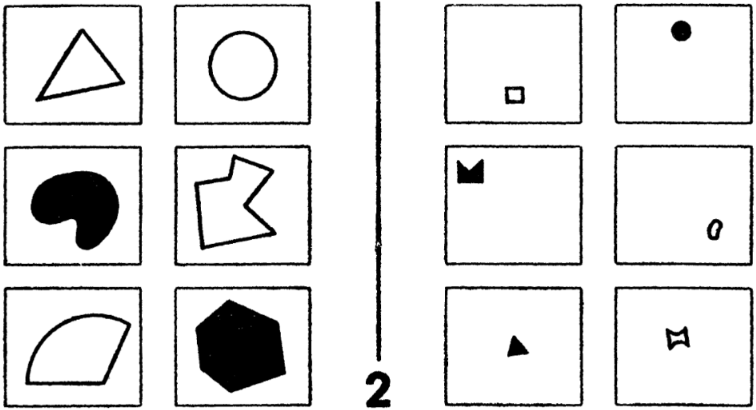
Пример (из области медицины)

Если *«возраст > 60»* ***и*** *«пациент ранее перенёс инфаркт»,*
то *операцию не делать, риск отрицательного исхода 60%.*

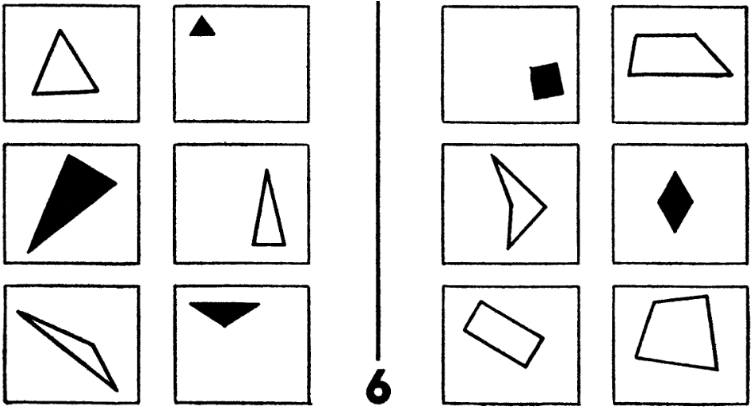
Пример (из области кредитного скоринга)

Если *«в анкете указан домашний телефон»*
и *«зарплата $>$ \$2000»* ***и*** *«сумма кредита $<$ \$5000»*
то *кредит можно выдать, риск дефолта 5%.*

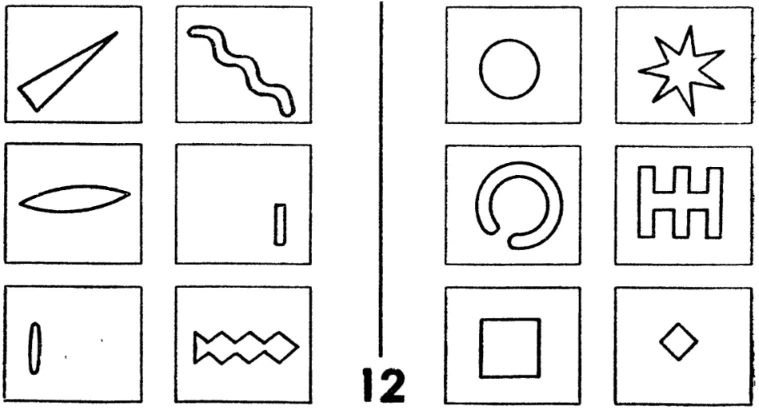
Тесты М. М. Бонгарда [Проблема узнавания, 1967]



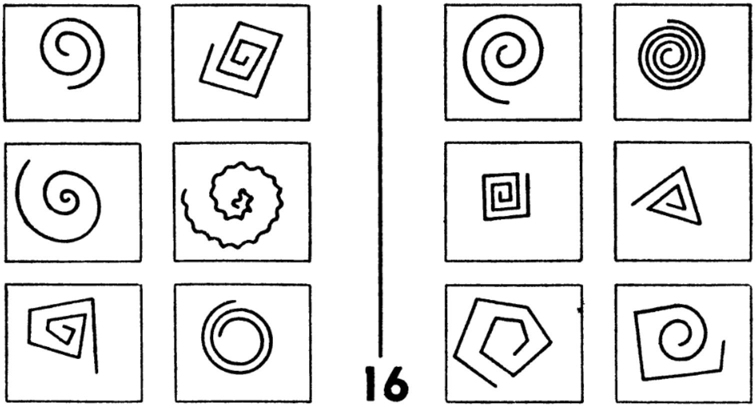
Тесты М. М. Бонгарда [Проблема узнавания, 1967]



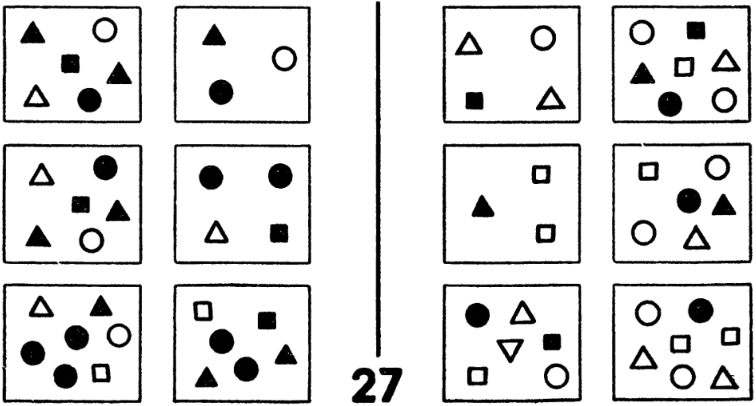
Тесты М. М. Бонгарда [Проблема узнавания, 1967]



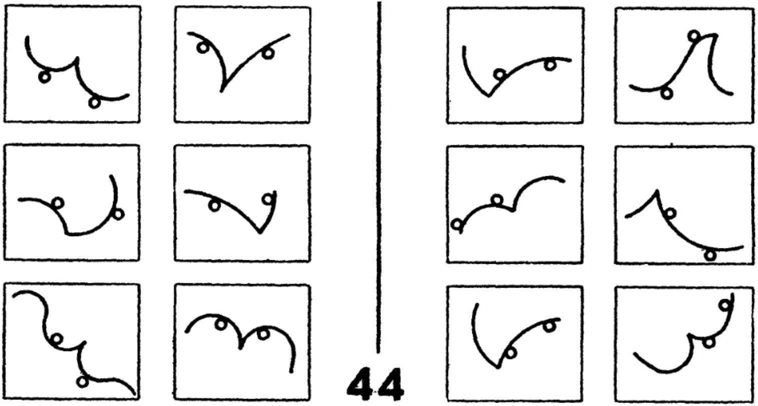
Тесты М. М. Бонгарда [Проблема узнавания, 1967]



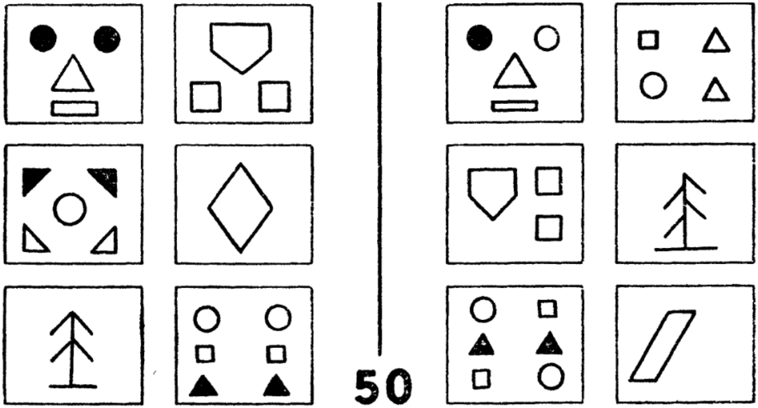
Тесты М. М. Бонгарда [Проблема узнавания, 1967]



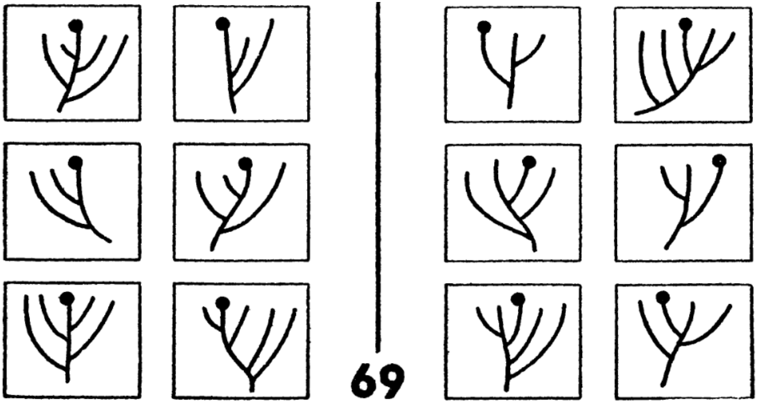
Тесты М. М. Бонгарда [Проблема узнавания, 1967]



Тесты М. М. Бонгарда [Проблема узнавания, 1967]



Тесты М. М. Бонгарда [Проблема узнавания, 1967]



Основные вопросы построения логических алгоритмов

1. Как изобретать признаки $f_1(x), \ldots, f_n(x)$?
— не наука, а искусство (размышления, озарения, эксперименты, консультации, мозговые штурмы,...)
2. Какого вида закономерности $R(x)$ нам нужны?
— простые формулы от малого числа признаков
3. Как определять информативность?
— так, чтобы одновременно $p \to \max$, $n \to \min$
4. Как искать закономерности?
— перебором подмножеств признаков
5. Как объединять закономерности в алгоритм?
— любым классификатором ($R(x)$ — это тоже признаки)

Закономерность — интерпретируемый высокоинформативный одноклассовый классификатор с отказами.

Проблема оценивания информативности

Проблема: надо сравнивать закономерности R.
Как свернуть два критерия в один критерий информативности?

$$\begin{cases} p(R) \to \max \\ n(R) \to \min \end{cases} \quad \stackrel{?}{\Longrightarrow} \quad I(p, n) \to \max$$

Очевидные, но не всегда адекватные свёртки:

- $I(p, n) = \dfrac{p}{p + n} \to \max$ (precision);
- $I(p, n) = p - n \to \max$ (accuracy);
- $I(p, n) = p - Cn \to \max$ (linear cost accuracy);
- $I(p, n) = \dfrac{p}{P} - \dfrac{n}{N} \to \max$ (relative accuracy);

  $P_c = \#\{x_i : y_i{=}c\}$ — число «своих» во всей выборке;
  $N_c = \#\{x_i : y_i{\neq}c\}$ — число «чужих» во всей выборке.

Нетривиальность проблемы свёртки двух критериев

Пример:

при $P = 200$, $N = 100$ и различных p и n.

Простые эвристики не всегда адекватны:

p	n	$p-n$	$p-5n$	$\frac{p}{P}-\frac{n}{N}$	$\frac{p}{n+1}$	IStat$\cdot\ell$	IGain$\cdot\ell$	$\sqrt{p}-\sqrt{n}$
50	0	**50**	50	0.25	50	22.65	23.70	7.07
100	50	**50**	−150	0	1.96	2.33	1.98	2.93
50	9	41	**5**	0.16	**5**	7.87	7.94	4.07
5	0	5	**5**	0.03	**5**	2.04	3.04	2.24
100	0	**100**	100	**0.5**	100	52.18	53.32	10.0
140	20	**120**	40	**0.5**	6.67	37.09	37.03	7.36

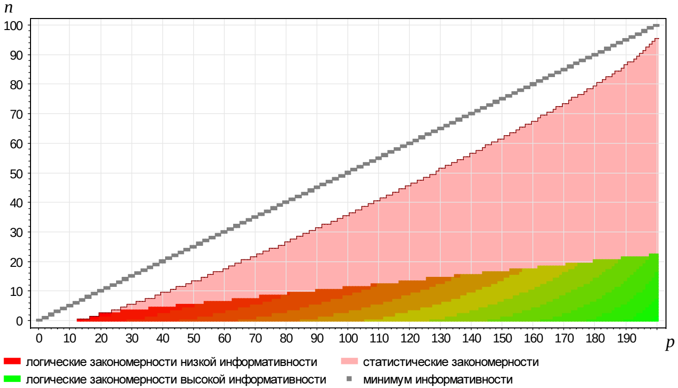
Часто используемые критерии информативности

Адекватные, но неочевидные критерии:

- энтропийный критерий прироста информации:
  $$\mathrm{IGain}(p,n) = h\left(\tfrac{P}{\ell}\right) - \tfrac{p+n}{\ell} h\left(\tfrac{p}{p+n}\right) - \tfrac{\ell-p-n}{\ell} h\left(\tfrac{P-p}{\ell-p-n}\right) \to \max,$$
  где $h(q) = -q\log_2 q - (1-q)\log_2(1-q)$
- критерий Джини (Gini impurity):
  $\mathrm{IGini}(p,n) = \mathrm{IGain}(p,n)$ при $h(q) = 4q(1-q)$
- точный статистический тест Фишера (Fisher's Exact Test):
  $\mathrm{IStat}(p,n) = -\frac{1}{\ell}\log_2 \frac{C_P^p C_N^n}{C_{P+N}^{p+n}} \to \max$
- критерий бустинга:
  $\sqrt{p} - \sqrt{n} \to \max$
- нормированный критерий бустинга:
  $\sqrt{p/P} - \sqrt{n/N} \to \max$

Где находятся закономерности в (p, n)-плоскости

Логические закономерности: $\frac{n}{p+n} \leqslant 0.1, \quad \frac{p}{P+N} \geqslant 0.05.$
Статистические закономерности: $\mathrm{IStat}(p, n) \geqslant 3.$



$P = 200$
$N = 100$

Вывод: неслучайность — ещё не значит закономерность.

Энтропийный критерий информативности

Пусть ω_0, ω_1 — два исхода с вероятностями q и $1-q$.
Количество информации: $I_0 = -\log_2 q$, $I_1 = -\log_2(1-q)$.
Энтропия — математическое ожидание количества информации:

$$h(q) = -q\log_2 q - (1-q)\log_2(1-q).$$

Энтропия выборки X^ℓ, если исходы — это классы $y{=}c$, $y{\neq}c$:

$$H(y) = h\left(\frac{P}{\ell}\right).$$

Энтропия выборки X^ℓ после получения информации $R(x_i)_{i=1}^{\ell}$:

$$H(y|R) = \frac{p+n}{\ell}h\left(\frac{p}{p+n}\right) + \frac{\ell-p-n}{\ell}h\left(\frac{P-p}{\ell-p-n}\right).$$

Прирост информации (Information gain, IGain):

$$\mathrm{IGain}(p,n) = H(y) - H(y|R).$$

Статистический критерий информативности

Точный тест Фишера. Пусть X — в.п., выборка X^ℓ — i.i.d.
Гипотеза H_0: $y(x)$ и $R(x)$ — независимые случайные величины.
Тогда вероятность реализации пары (p, n) описывается *гипергеометрическим распределением*:

$$\mathrm{P}(p, n) = \frac{C_P^p C_N^n}{C_{P+N}^{p+n}}, \quad 0 \leqslant p \leqslant P, \quad 0 \leqslant n \leqslant N,$$

где $C_N^n = \frac{N!}{n!(N-n)!}$ — биномиальные коэффициенты.

Определение

Информативность предиката $R(x)$ относительно класса $c \in Y$:

$$\mathrm{IStat}(p, n) = -\frac{1}{\ell} \log_2 \frac{C_P^p C_N^n}{C_{P+N}^{p+n}},$$

$\mathrm{IStat}(p, n) \geqslant I_0$ — *статистическая закономерность класса c.*

Соотношение статистического и энтропийного критериев

Определение

Предикат R — закономерность по энтропийному критерию, если $\mathrm{IGain}(p,n) > G_0$ при некотором G_0.

Теорема

Энтропийный критерий IGain *асимптотически эквивалентен статистическому* IStat*:*

$$\mathrm{IStat}(p,n) \to \mathrm{IGain}(p,n) \quad \text{при } \ell \to \infty.$$

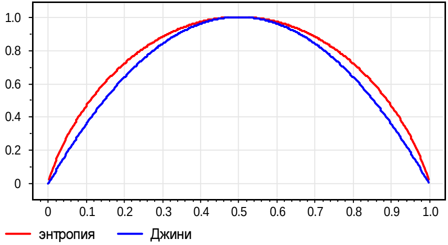
Доказательство:
применить формулу Стирлинга к критерию IStat.

Соотношение критерия Джини и энтропийного критериев

Критерий прироста информации:

$$\mathrm{IGain}(p, n) = h\left(\tfrac{P}{\ell}\right) - \tfrac{p+n}{\ell} h\left(\tfrac{p}{p+n}\right) - \tfrac{\ell-p-n}{\ell} h\left(\tfrac{P-p}{\ell-p-n}\right) \to \max,$$

- энтропийный критерий:
  $h(q) = -q \log_2 q - (1-q) \log_2 (1-q)$
- критерий Джини (Gini impurity):
  $h(q) = 4q(1-q)$



Часто используемые виды правил

1. Пороговое условие (решающий пень, decision stump):

$$R(x) = \bigl[f_j(x) \leqslant a_j\bigr] \text{ или } \bigl[a_j \leqslant f_j(x) \leqslant b_j\bigr].$$

2. *Конъюнкция* пороговых условий:

$$R(x) = \bigwedge_{j \in J} \bigl[a_j \leqslant f_j(x) \leqslant b_j\bigr].$$

3. *Синдром* — выполнение не менее d условий из J, (при $d = |J|$ это конъюнкция, при $d = 1$ — дизъюнкция):

$$R(x) = \Biggl[\sum_{j \in J} \bigl[a_j \leqslant f_j(x) \leqslant b_j\bigr] \geqslant d\Biggr],$$

Параметры J, a_j, b_j, d настраиваются по обучающей выборке путём оптимизации *критерия информативности*.

Часто используемые виды закономерностей

4. *Полуплоскость* — линейная пороговая функция:

$$R(x) = \Bigl[\sum_{j \in J} w_j f_j(x) \geqslant w_0 \Bigr].$$

5. *Шар* — пороговая функция близости:

$$R(x) = \bigl[r(x, x_0) \leqslant w_0 \bigr],$$

АВО — алгоритмы вычисления оценок [Ю. И. Журавлёв, 1971]:

$$r(x, x_0) = \max_{j \in J} w_j \bigl| f_j(x) - f_j(x_0) \bigr|.$$

SCM — машины покрывающих множеств [M. Marchand, 2001]:

$$r(x, x_0) = \sum_{j \in J} w_j \bigl| f_j(x) - f_j(x_0) \bigr|^{\gamma}.$$

Параметры J, w_j, w_0, x_0 настраиваются по обучающей выборке путём оптимизации *критерия информативности*.

Поиск информативных закономерностей

Вход: выборка X^ℓ;
Выход: множество закономерностей Z;

начальное множество правил Z;
повторять
  $Z' :=$ множество модификаций правил $R \in Z$;
  удалить слишком похожие правила из $Z \cup Z'$;
  оценить информативность всех правил $R \in Z'$;
  $Z :=$ наиболее информативные правила из $Z \cup Z'$;
пока правила продолжают улучшаться
вернуть Z.

Задача перебора конъюнкций

Пусть $\mathscr{B}$ — конечное множество *элементарных предикатов*.
Множество конъюнкций с ограниченным числом термов из $\mathscr{B}$:

$$\mathscr{K}_K[\mathscr{B}] = \bigl\{\varphi(x) = \beta_1(x) \wedge \dots \wedge \beta_k(x) \bigm| \beta_1, \dots, \beta_k \in \mathscr{B},\ k \leqslant K\bigr\}.$$

Число допустимых конъюнкций: $O(|\mathscr{B}|^K)$.

Семейство методов локального поиска

Окрестность $V(\varphi)$ — все конъюнкции, получаемые из $V(\varphi)$ добавлением, изъятием или модификацией одного из термов.

Основная идея: на t-й итерации

$$\varphi_t := \arg\max_{\varphi \in V(\varphi_{t-1})} I_c(\varphi, X^\ell).$$

Обобщённый алгоритм локального поиска

Вход: выборка X^ℓ; класс $c \in Y$;
начальное приближение φ_0; параметры $t_{\max}$, d, ε;

Выход: конъюнкция φ;

$I^* := I_c(\varphi_0, X^\ell)$; $\quad \varphi^* := \varphi_0$;
для всех $t = 1, \dots, t_{\max}$
  $\varphi_t := \arg\max\limits_{\varphi \in V(\varphi_{t-1})} I_c(\varphi, X^\ell)$ — наиболее перспективная;
  $\varphi_t^* := \arg\max\limits_{\substack{\varphi \in V(\varphi_{t-1}) \\ E_c(\varphi) < \varepsilon}} I_c(\varphi, X^\ell)$ — лучшая конъюнкция;
  если $I_c(\varphi_t^*) > I^*$ **то**
    $t^* := t$; $\quad \varphi^* := \varphi_t^*$; $\quad I^* := I_c(\varphi^*)$
  если $t - t^* > d$ **то**
    выход;
вернуть φ^*;

Частные случаи

- **жадный алгоритм:**
  $V(\varphi)$ — только добавления термов; $\varphi_0 = \varnothing$;
- **стохастический локальный поиск (SLS):**
  $V(\varphi)$ — случайное подмножество всевозможных добавлений, удалений, модификаций термов; $\varphi_0 = \varnothing$;
- **стабилизация:**
  $V(\varphi)$ — удаления термов или изменение параметров в термах; $\varphi_0 \neq \varnothing$;
- **редукция:**
  $V(\varphi)$ — только удаления термов; $\varphi_0 \neq \varnothing$;
  $I_c(\varphi, X^k)$ оценивается по контрольной выборке X^k.

Поиск закономерностей — это отбор признаков

Отличия от методов отбора признаков:

- вместо внешнего критерия $Q_{\text{ext}} \to \min$ критерий информативности $I_c \to \max$;
- вместо одного набора признаков строится множество закономерностей.

Все методы отбора признаков подходят:

- добавления–удаления;
- поиск в глубину;
- поиск в ширину;
- генетические (эволюционные) алгоритмы;
- случайный поиск с адаптацией.

Задача бинаризации вещественного признака

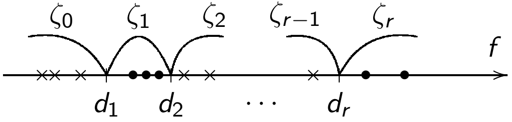
Дано:

вещественный признак $f(x)$;

Построить:

наиболее информативное разбиение области значений признака на относительно небольшое число зон:

$$\zeta_0(x) = [f(x) < d_1];$$
$$\zeta_s(x) = [d_s \leqslant f(x) < d_{s+1}], \qquad s = 1, \dots, r-1;$$
$$\zeta_r(x) = [d_r \leqslant f(x)].$$



Алгоритм разбиения области значений признака на зоны

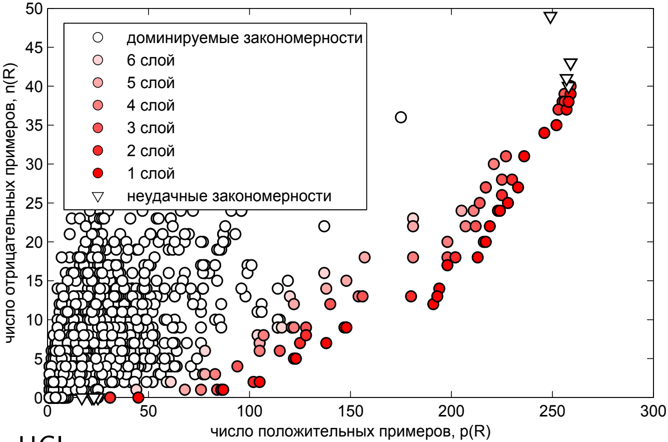
Вход: выборка X^ℓ; класс $c \in Y$; параметры r и δ_0.

Выход: $D = \{d_1 < \dots < d_r\}$ — последовательность порогов;

$D := \varnothing$; упорядочить выборку X^ℓ по возрастанию $f(x_i)$;
для всех $i = 2, \dots, \ell$
  если $f(x_{i-1}) \ne f(x_i)$ и $[y_{i-1} = c] \ne [y_i = c]$ **то**
    добавить порог $\frac{1}{2}(f(x_{i-1}) + f(x_i))$ в конец D;
повторять
  для всех $d_i \in D$, $i = 1, \dots, |D| - 1$
    $\delta I_i := I_c(\zeta_{i-1} \vee \zeta_i \vee \zeta_{i+1}) - \max\{I_c(\zeta_{i-1}), I_c(\zeta_i), I_c(\zeta_{i+1})\}$;
  $i := \arg\max\limits_s \delta I_s$;
  если $\delta I_i > \delta_0$ **то**
    слить зоны ζ_{i-1}, ζ_i, ζ_{i+1}, удалив d_i и d_{i+1} из D;
пока $|D| > r + 1$.

Отбор закономерностей по информативности в (p, n)-плоскости

Парето-фронт — множество недоминируемых закономерностей (точка R недоминируема, если правее и ниже точек нет)



задача UCI:german

Определение решающего списка

Решающий список (Decision List, DL)
— алгоритм классификации $a\colon X \to Y$, который задаётся закономерностями $R_1(x), \dots, R_T(x)$ классов $c_1, \dots, c_T \in Y$:

$$\begin{tikzcd}
x \arrow[r] & \boxed{R_1(x)} \arrow[r, "0"] \arrow[d, "1"] & \cdots \arrow[r, "0"] & \boxed{R_T(x)} \arrow[r, "0"] \arrow[d, "1"] & c_0 \\
& c_1 & & c_T &
\end{tikzcd}$$

1: **для всех** $t = 1, \dots, T$
2: **если** $R_t(x) = 1$ **то**
3: **вернуть** c_t;
4: **вернуть** c_0 — отказ от классификации объекта x.

$$E(R_t, X^\ell) = \frac{n(R_t)}{n(R_t) + p(R_t)} \to \min \quad \text{— доля ошибок } R_t \text{ на } X^\ell$$

Жадный алгоритм построения решающего списка

Вход: выборка X^ℓ; семейство предикатов $\mathscr{B}$;
параметры: $T_{\max}$, $I_{\min}$, $E_{\max}$, ℓ_0;

Выход: решающий список $\{R_t, c_t\}_{t=1}^{T}$;

1: $U := X^\ell$;

2: **для всех** $t := 1, \dots, T_{\max}$

3: выбрать класс c_t;

4: максимизация информативности $I(R, U)$ при ограничении на число ошибок $E(R, U)$:

$$R_t := \arg\max_{R \in \mathscr{B}\colon E(R,U) \leqslant E_{\max}} I(R, U);$$

5: **если** $I(R_t, U) < I_{\min}$ **то выход**;

6: оставить объекты, не покрытые правилом R_t:

$$U := \{x \in U : R_t(x) = 0\};$$

7: **если** $|U| \leqslant \ell_0$ **то выход**;

Замечания к алгоритму построения решающего списка

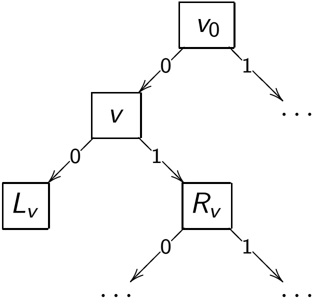
- Параметр $E_{\max}$ управляет сложностью списка:
  $E_{\max} \downarrow \quad \Rightarrow \quad p(R_t) \downarrow, \ T \uparrow.$
- Стратегии выбора класса c_t:
  1) все классы по очереди;
  2) на каждом шаге определяется оптимальный класс.
- Простой обход проблемы пропусков в данных.
- Другие названия:
  комитет с логикой старшинства (Majority Committee)
  голосование по старшинству (Majority Voting)
  машина покрывающих множеств (Set Covering Machine, SCM)
- **Недостаток:** низкое качество классификации

Определение бинарного решающего дерева

Бинарное решающее дерево — алгоритм классификации $a(x)$, задающийся бинарным деревом:

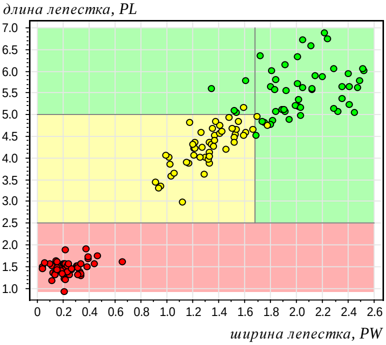
1) $\forall v \in V_{\text{внутр}} \;\to\;$ предикат $\beta_v : X \to \{0, 1\}$, $\beta_v \in \mathscr{B}$
2) $\forall v \in V_{\text{лист}} \;\to\;$ имя класса $c_v \in Y$.

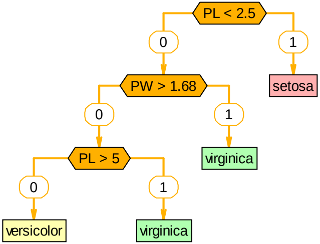
1: $v := v_0$;
2: **пока** $v \in V_{\text{внутр}}$
3:     **если** $\beta_v(x) = 1$ **то**
4:         переход вправо:
        $v := R_v$;
5:     **иначе**
6:         переход влево:
        $v := L_v$;
7: **вернуть** c_v.



Пример решающего дерева

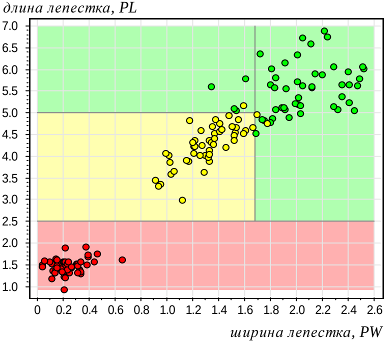
Задача Фишера о классификации цветков ириса на 3 класса, в выборке по 50 объектов каждого класса, 4 признака.

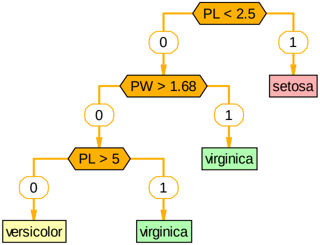




На графике: в осях двух самых информативных признаков (из 4) два класса разделились без ошибок, на третьем 3 ошибки.

Решающее дерево → покрывающий набор конъюнкций





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

Жадный алгоритм построения дерева ID3

1: **ПРОЦЕДУРА** LearnID3 ($U \subseteq X^\ell$);

2: **если** все объекты из U лежат в одном классе $c \in Y$ **то**

3:     **вернуть** новый лист v, $c_v := c$;

4: найти предикат с максимальной информативностью:
$\beta := \arg\max\limits_{\beta \in \mathscr{B}} I(\beta, U)$;

5: разбить выборку на две части $U = U_0 \sqcup U_1$ по предикату β:
$U_0 := \bigl\{x \in U \colon \beta(x) = 0\bigr\}$;
$U_1 := \bigl\{x \in U \colon \beta(x) = 1\bigr\}$;

6: **если** $U_0 = \varnothing$ или $U_1 = \varnothing$ **то**

7:     **вернуть** новый лист v, $c_v :=$ Мажоритарный класс(U);

8: создать новую внутреннюю вершину v: $\beta_v := \beta$;
построить левое поддерево: $L_v :=$ LearnID3 (U_0);
построить правое поддерево: $R_v :=$ LearnID3 (U_1);

9: **вернуть** v;

Разновидности многоклассовых критериев ветвления

1. **Отделение одного класса** (слишком сильное ограничение):
$$I(\beta, X^\ell) = \max_{c \in Y} I_c(\beta, X^\ell).$$

2. **Многоклассовый энтропийный критерий:**
$$I(\beta, X^\ell) = \sum_{c \in Y} h\left(\frac{P_c}{\ell}\right) - \frac{p}{\ell} h\left(\frac{p_c}{p}\right) - \frac{\ell - p}{\ell} h\left(\frac{P_c - p_c}{\ell - p}\right),$$

где $P_c = \#\{x_i \colon y_i = c\}$, $p = \#\{x_i \colon \beta(x_i) = 1\}$, $h(z) \equiv -z \log_2 z$.

3. **Критерий Джини:**
$$I(\beta, X^\ell) = \#\big\{(x_i, x_j) \colon \beta(x_i) = \beta(x_j) \text{ и } y_i = y_j\big\}.$$

4. D-**критерий В.И.Донского:**
$$I(\beta, X^\ell) = \#\big\{(x_i, x_j) \colon \beta(x_i) \neq \beta(x_j) \text{ и } y_i \neq y_j\big\}.$$

Обработка пропусков

На стадии обучения:

- $\beta_v(x)$ не определено $\Rightarrow$ x_i исключается из U для $I(\beta, U)$
- $q_v = \frac{|U_0|}{|U|}$ — оценка вероятности левой ветви, $\forall v \in V_{\text{внутр}}$
- $P_v(y|x) = \frac{1}{|U|}\#\bigl\{x_i \in U\colon y_i = y\bigr\}$ для всех $v \in V_{\text{лист}}$

На стадии классификации:

- $\beta_v(x)$ определено $\Rightarrow$ либо налево, либо направо:
  $P_v(y|x) = \bigl(1-\beta_v(x)\bigr)P_{L_v}(y|x) + \beta_v(x)P_{R_v}(y|x).$
- $\beta_v(x)$ не определено $\Rightarrow$ *пропорциональное распределение*:
  $P_v(y|x) = q_v P_{L_v}(y|x) + \bigl(1-q_v\bigr)P_{R_v}(y|x).$
- Окончательное решение — наиболее вероятный класс:
  $$y = \arg\max_{y\in Y} P_{v_0}(y|x).$$

Решающие деревья ID3: достоинства и недостатки

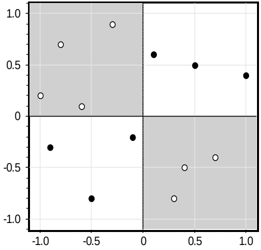
Достоинства:

- Интерпретируемость и простота классификации.
- Гибкость: можно варьировать множество $\mathscr{B}$.
- Допустимы разнотипные данные и данные с пропусками.
- Трудоёмкость линейна по длине выборки $O(|\mathscr{B}|h\ell)$.
- Не бывает отказов от классификации.

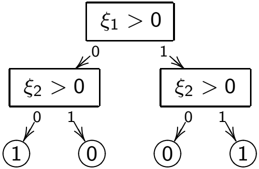
Недостатки:

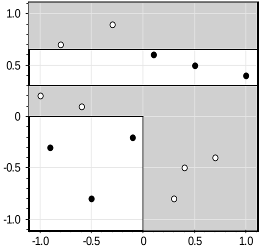
- Жадный ID3 переусложняет структуру дерева, и, как следствие, сильно переобучается.
- Фрагментация выборки: чем дальше v от корня, тем меньше статистическая надёжность выбора β_v, c_v.
- Высокая чувствительность к шуму, к составу выборки, к критерию информативности.

Жадный ID3 переусложняет структуру дерева

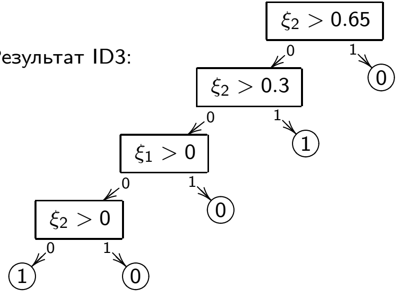


Оптимальное дерево для задачи XOR:





Результат ID3:



Усечение дерева (pruning). Алгоритм C4.5

X^k — независимая контрольная выборка, $k \approx 0.5\ell$.

1: **для всех** $v \in V_{\text{внутр}}$
2: $S_v :=$ подмножество объектов X^k, дошедших до v;
3: **если** $S_v = \varnothing$ **то**
4: **вернуть** новый лист v, $c_v :=$ Мажоритарный класс(U);
5: число ошибок при классификации S_v четырьмя способами:
$r(v)$ — поддеревом, растущим из вершины v;
$r_L(v)$ — поддеревом левой дочерней вершины L_v;
$r_R(v)$ — поддеревом правой дочерней вершины R_v;
$r_c(v)$ — к классу $c \in Y$.
6: в зависимости от того, какое из них минимально:
сохранить поддерево v;
заменить поддерево v поддеревом L_v;
заменить поддерево v поддеревом R_v;
заменить поддерево v листом, $c_v := \arg\min\limits_{c \in Y} r_c(v)$.

CART: деревья регрессии и классификации

Обобщение на случай регрессии: $Y = \mathbb{R}, \;\; c_v \in \mathbb{R}$

Пусть U_v — множество объектов x_i, дошедших до вершины v

Значения в терминальных вершинах — МНК-решение:

$$c_v := \hat{y}(U_v) = \frac{1}{|U_v|} \sum_{x_i \in U_v} y_i$$

Критерий информативности — среднеквадратичная ошибка

$$I(\beta, U_v) = \sum_{x_i \in U_v} \bigl(\hat{y}_i(\beta) - y_i\bigr)^2,$$

где $\hat{y}_i(\beta) = \beta(x_i)\hat{y}(U_{v1}) + \bigl(1 - \beta(x_i)\bigr)\hat{y}(U_{v0})$
— прогноз после ветвления β и разбиения $U_v = U_{v0} \sqcup U_{v1}$

CART: критерий Minimal Cost-Complexity Pruning

Среднеквадратичная ошибка со штрафом за сложность дерева

$$C_\alpha = \sum_{x_i=1}^{\ell} \bigl(\hat{y}_i - y_i\bigr)^2 + \alpha |V_{\text{лист}}| \to \min$$

При увеличении α дерево последовательно упрощается.
Причём последовательность вложенных деревьев единственна.

Из этой последовательности выбирается дерево с минимальной ошибкой на тестовой выборке (Hold-Out).

Для случая классификации используется аналогичная стратегия усечения, с критерием Джини.

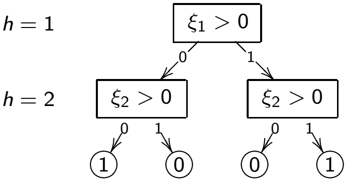
Небрежные решающие деревья — ODT (Oblivious Decision Tree) [1991]

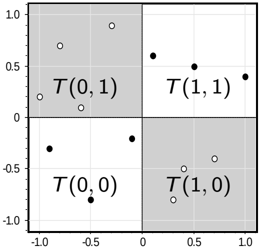
Решение проблемы фрагментации:
строится сбалансированное дерево высоты H;
для всех узлов уровня h условие ветвления $\beta_h(x)$ *одинаково*;
на уровне h ровно 2^{h-1} вершин; X делится на 2^H ячеек.

Классификатор задаётся *таблицей решений* $T\colon \{0,1\}^H \to Y$:

$$a(x) = T\bigl(\beta_1(x), \ldots, \beta_H(x)\bigr).$$

Пример: задача XOR, $H = 2$.





Алгоритм обучения ODT

Вход: выборка X^ℓ; семейство правил $\mathscr{B}$; глубина дерева H;
Выход: условия β_h, $h = 1, \dots, H$; таблица $T\colon \{0,1\}^H \to Y$;

1: **для всех** $h = 1, \dots, H$
2: найти предикат с максимальной информативностью:
$\beta_h := \arg\max\limits_{\beta \in \mathscr{B}} I(\beta_1, \dots, \beta_{h-1}, \beta; X^\ell)$;
3: **для всех** $b \equiv (b_1, \dots, b_H) \in \{0,1\}^H$
4: классификация по мажоритарному правилу:
$T(b_1, \dots, b_H) := \arg\max\limits_{c \in Y} \sum\limits_{i=1}^{\ell} [y_i = c] \prod\limits_{h=1}^{H} \bigl[\beta_h(x_i) = b_h\bigr]$;

$$I(\beta_1, \dots, \beta_h) = \sum_{c \in Y} h\left(\frac{P_c}{\ell}\right) - \sum_{b \in \{0,1\}^h} \frac{|X_b|}{\ell} h\left(\frac{|X_b \cap X_c|}{|X_b|}\right);$$

$$X_b = \bigl\{x_i \colon \beta_s(x_i) = b_s,\ s = 1, \dots, h\bigr\}, \qquad X^\ell = \bigsqcup_{b \in \{0,1\}^h} X_b.$$

Случайный лес (Random Forest)

Голосование деревьев классификации, $Y = \{-1, +1\}$:

$$a(t) = \operatorname{sign} \frac{1}{T} \sum_{t=1}^{T} b_t(x).$$

Голосование деревьев регрессии, $Y = \mathbb{R}$:

$$a(t) = \frac{1}{T} \sum_{t=1}^{T} b_t(x).$$

- каждое дерево $b_t(x)$ обучается по случайной выборке с повторениями
- в каждой вершине признак выбирается из случайного подмножества $\sqrt{n}$ признаков
- признаки и пороги выбираются по критерию Джини
- усечений (pruning) нет

Резюме в конце лекции

- Основные требования к логическим закономерностям:
  - интерпретируемость, информативность, различность.
- Преимущества решающих деревьев:
  - интерпретируемость,
  - допускаются разнотипные данные,
  - возможность обхода пропусков;
- Недостатки решающих деревьев:
  - переобучение,
  - фрагментация,
  - неустойчивость к шуму, составу выборки, критерию;
- Способы устранения этих недостатков:
  - редукция,
  - специальные виды деревьев ODT, ADT и др.
  - композиции (леса) деревьев.

Yandex MatrixNet = градиентный бустинг над ODT.