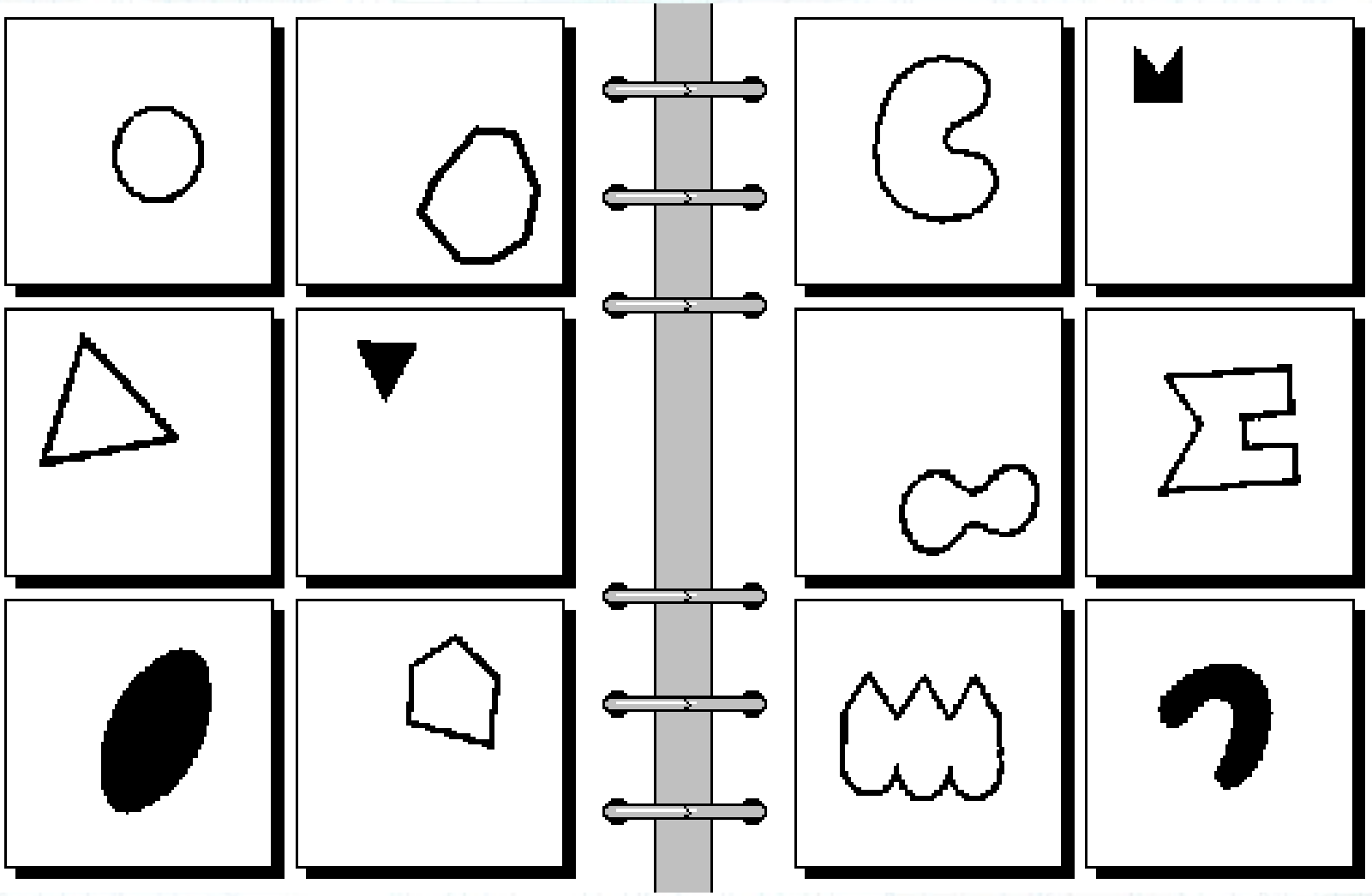
Машинное обучение
Логические методы классификации



https://yandexdataschool.ru/edu-process/courses/machine-learning

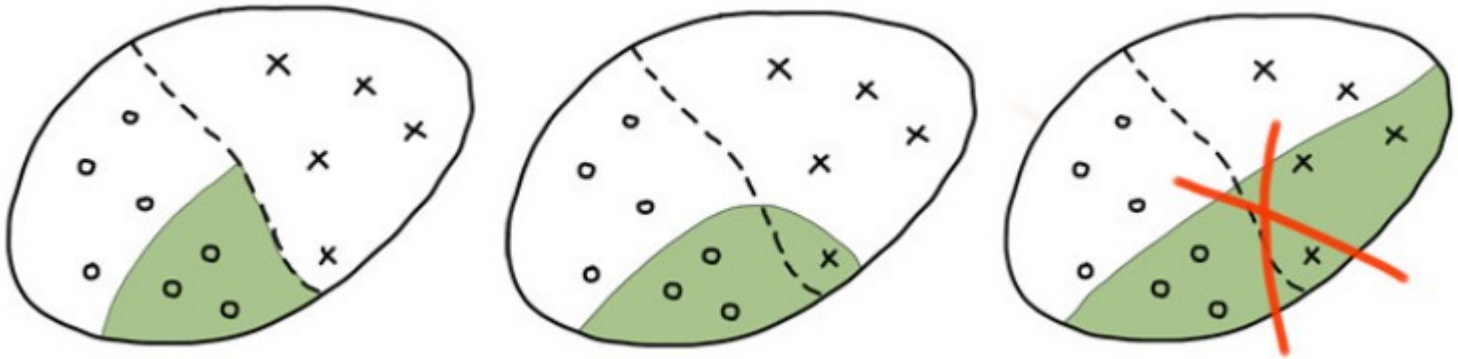
Содержание лекции

- Понятие закономерности
- Критерий качества закономерностей
- Поиск закономерностей
- Алгоритмы классификации на основе логических закономерностей

Понятие закономерности

- Предикат R: X → {0,1} – закономерность, если он выделяет (R(x)=1) достаточно много объектов одного класса C и практически не выделяет объектов других классов

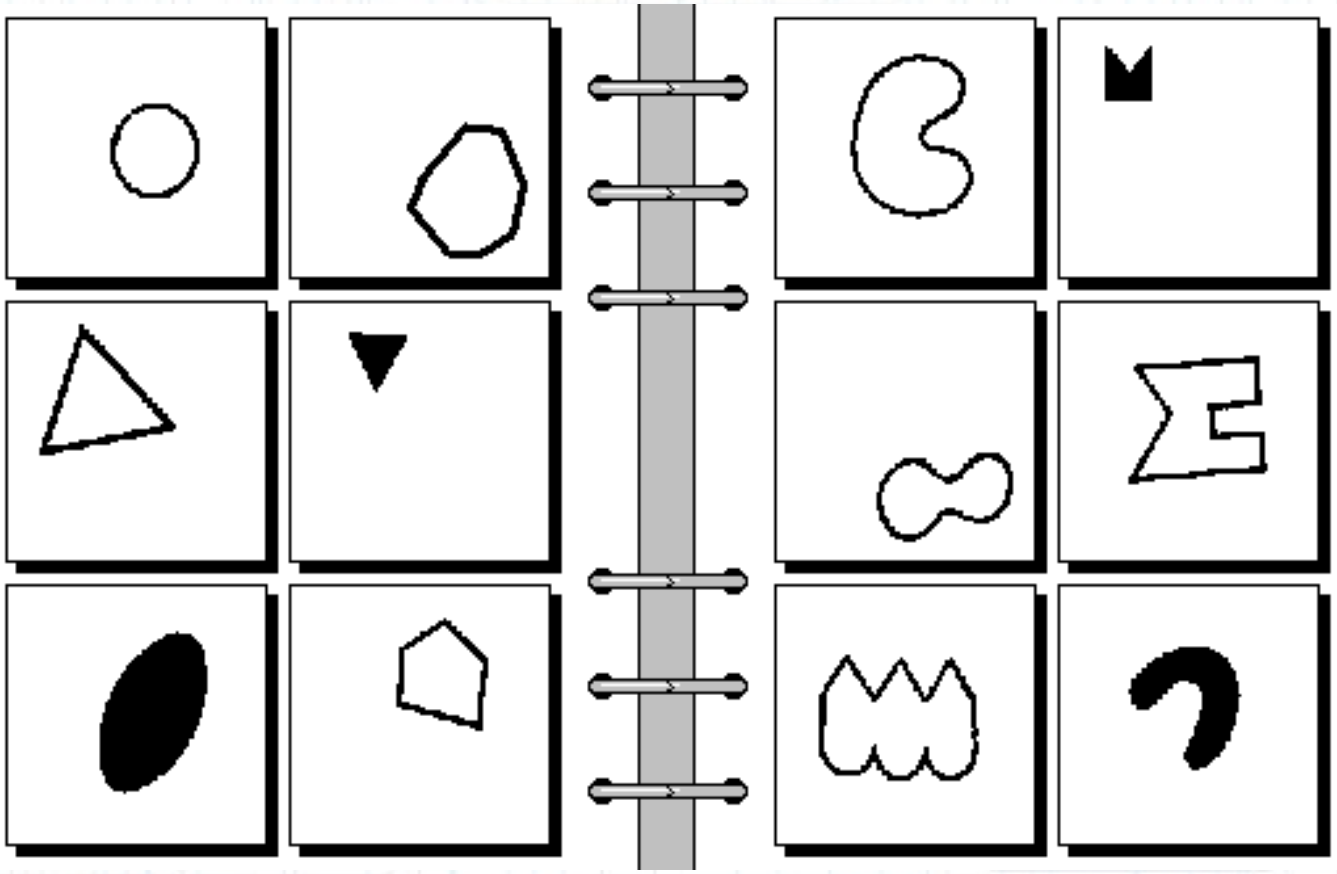
$$p_c(R) = \#\{x_i : R(x_i)=1 \text{ и } y_i=c\} \to \max;$$
$$n_c(R) = \#\{x_i : R(x_i)=1 \text{ и } y_i\neq c\} \to \min;$$



Примеры

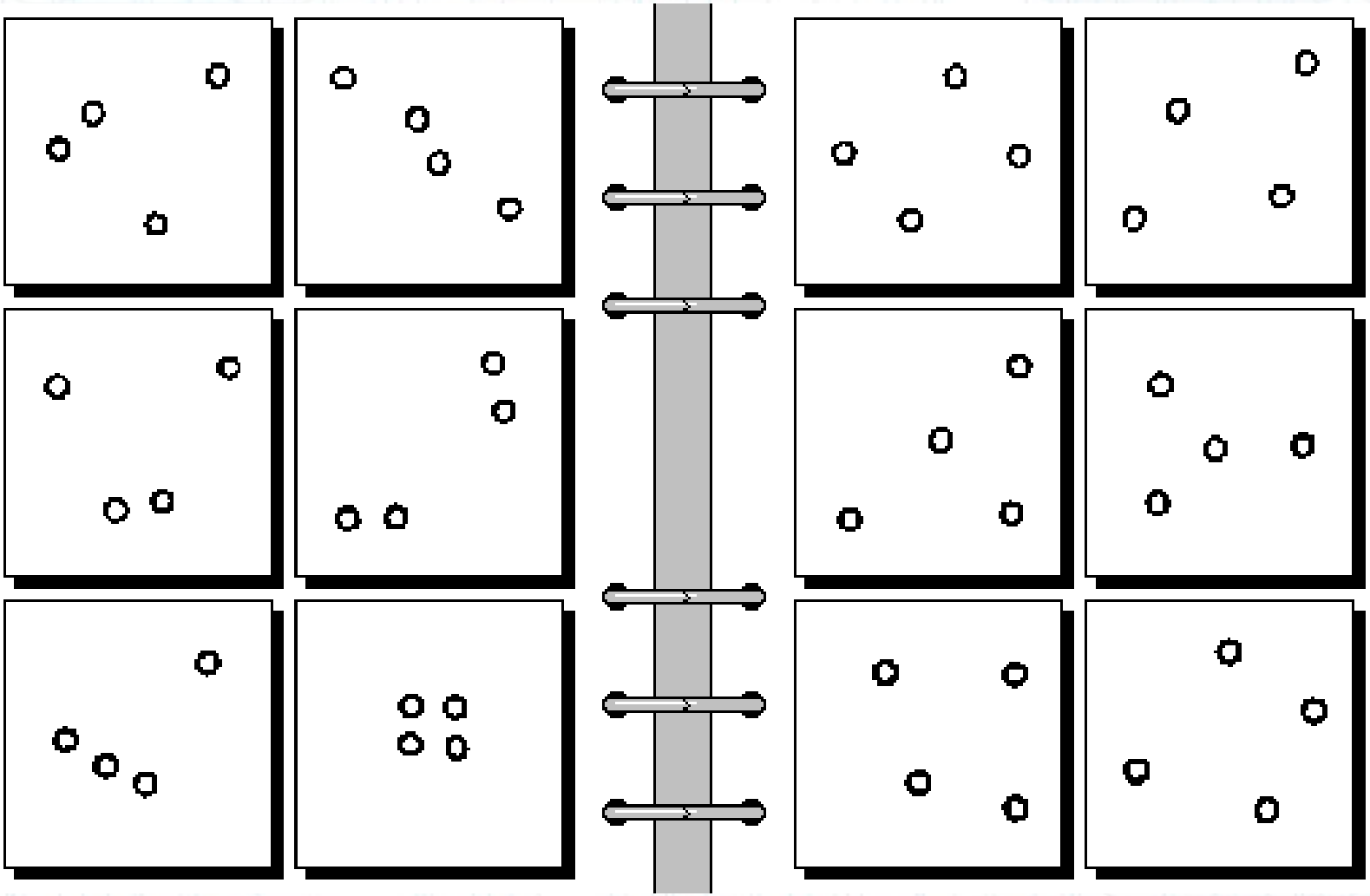
- Если «возраст > 60» и «пациент ранее перенёс инфаркт», то операцию не делать, риск отрицательного исхода 60%.
- Если «в анкете указан домашний телефон» и «зарплата > $2000» и «сумма кредита < $5000» то кредит можно выдать, риск дефолта 5%.

Тесты М.М.Бонгарда

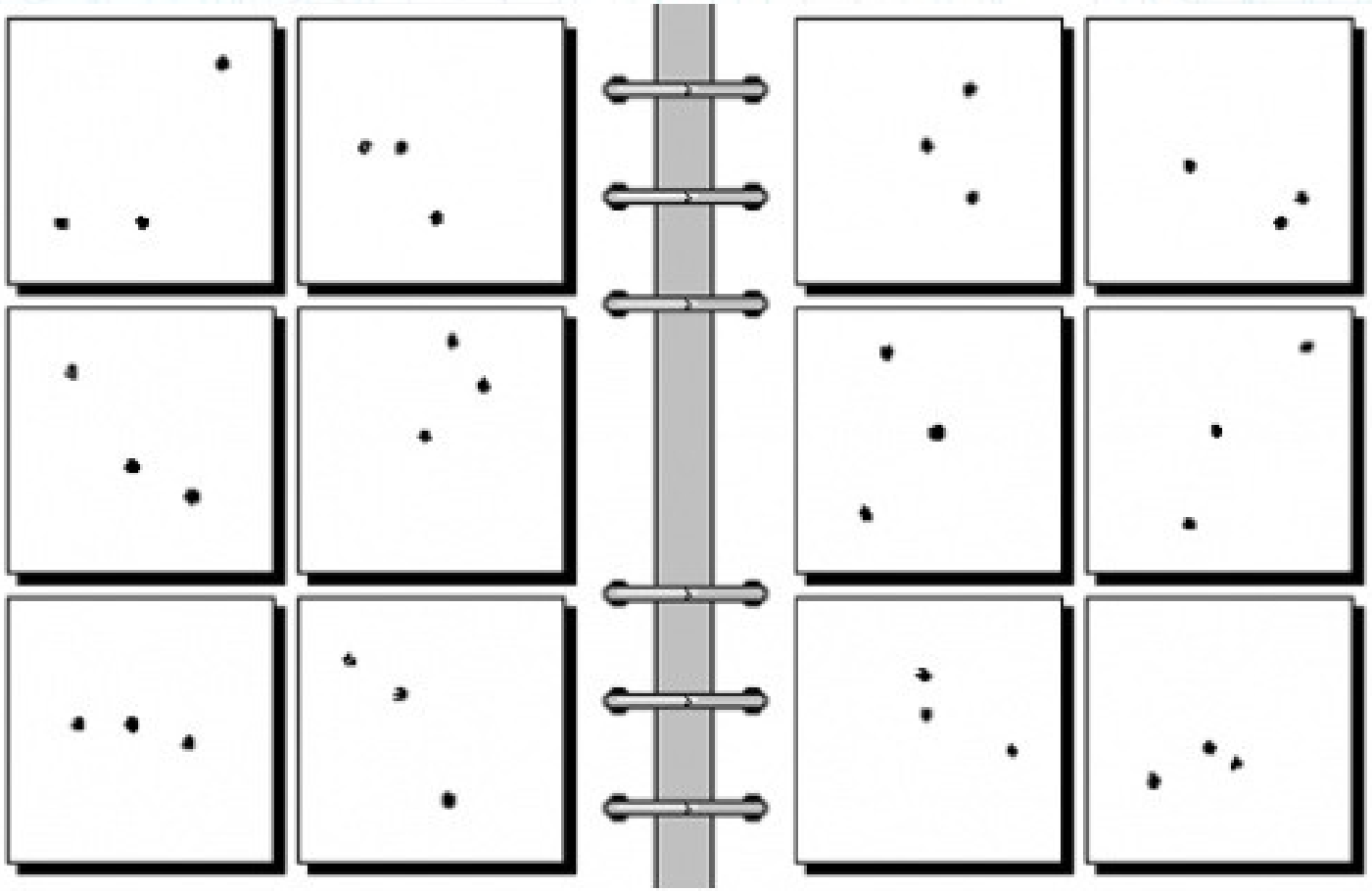


Этот тип головоломки изобрёл выдающийся русский кибернетик, основоположник теории распознавания образов Михаил Моисеевич Бонгард: в 1967-м году он впервые опубликовал одну из них в своей книге "Проблема узнавания"

Тесты М.М.Бонгарда



Тесты М.М.Бонгарда



Как сравнивать закономерности?

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

$$I(p,n) = \frac{p}{p+n} \to \max \quad \text{(precision)};$$

$$I(p,n) = p - n \to \max \quad \text{(accuracy)};$$

$$I(p,n) = p - Cn \to \max \quad \text{(linear cost accuracy)};$$

$$I(p,n) = \frac{p}{P} - \frac{n}{N} \to \max \quad \text{(relative accuracy)};$$

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

Как сравнивать закономерности?

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

p	n	p/(p+n)	p-n	p-5n	p/P-n/N
10	0	1	10	10	0,05
200	10	0,95	190	150	0,9
10	0	1	10	10	0,05
60	50	0,55	10	-190	-0,2
200	40	0,83	160	0	0,6
5	1	0,83	4	0	0,02
10	0	1	10	10	0,05
200	95	0,68	105	-275	0,05

Вероятностный подход

- Рассмотрим опыт – отбор предикатом объектов обучающей выборки
- Предикат – закономерность ттт,к события: “объект отобран предикатом” и “объект имеет класс **c**” зависимы
- Качество закономерности = мера зависимости случайных событий

Точный тест Фишера

- Предположим, что события “объект отобран предикатом” и “объект имеет класс **c**” независимы
- Тогда вероятность отобрать p объектов класса **c** и n – других классов:

Точный тест Фишера

- Предположим, что события “объект отобран предикатом” и “объект имеет класс **c**” независимы
- Тогда вероятность отобрать p объектов класса **c** и n – других классов: $\frac{C_P^p C_N^n}{C_{P+N}^{p+n}}$
- Это правдоподобие гипотезы независимости событий. Чем меньше данная вероятность, тем более зависимы события

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

Точный тест Фишера

p	n	p/(p+n)	p-n	p-5n	p/P-n/N	Вероятность
10	0	1	10	10	0,05	0,016
200	10	0,95	190	150	0,9	8,80E-66
10	0	1	10	10	0,05	0,016
60	50	0,55	10	-190	-0,2	3,50E-04
200	40	0,83	160	0	0,6	1,50E-36
5	1	0,83	4	0	0,02	0,26
10	0	1	10	10	0,05	0,016
200	95	0,68	105	-275	0,05	0,0038

Точный тест Фишера

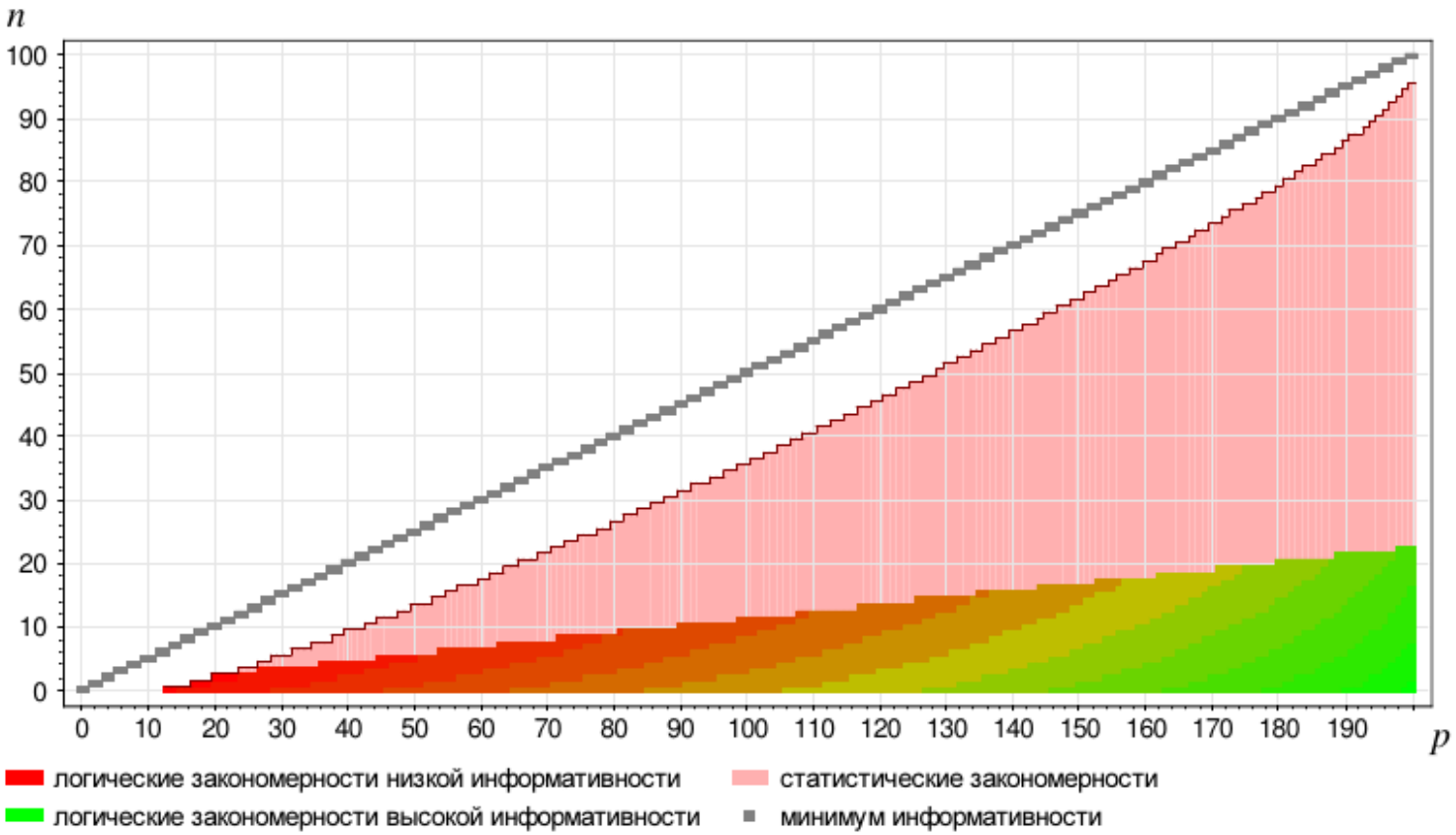
p	n	p/(p+n)	p-n	p-5n	p/P-n/N	Вероятность	Плотность
10	0	1	10	10	0,05	0,016	0,16
200	10	0,95	190	150	0,9	8,80E-66	1,848E-063
10	0	1	10	10	0,05	0,016	0,16
60	50	0,55	10	-190	-0,2	3,50E-04	0,0385
200	40	0,83	160	0	0,6	1,50E-36	3,6E-034
5	1	0,83	4	0	0,02	0,26	1,56
10	0	1	10	10	0,05	0,016	0,16
200	95	0,68	105	-275	0,05	0,0038	1,121

Для дискретных распределений модели с меньшим числом значений случайной величины более правдоподобны, чем с б**о**льшим. Это – переобучение. Лучше переходить к приближению дискретной плотности непрерывной функцией, например, кусочно-постоянной т.е. умножать вероятность теста Фишера на (p+n) – количество различных значений случайной величины.

Логические и статистические закономерности

Логические закономерности: $\frac{n}{p+n} \leqslant 0.1, \quad \frac{p}{P+N} \geqslant 0.05.$

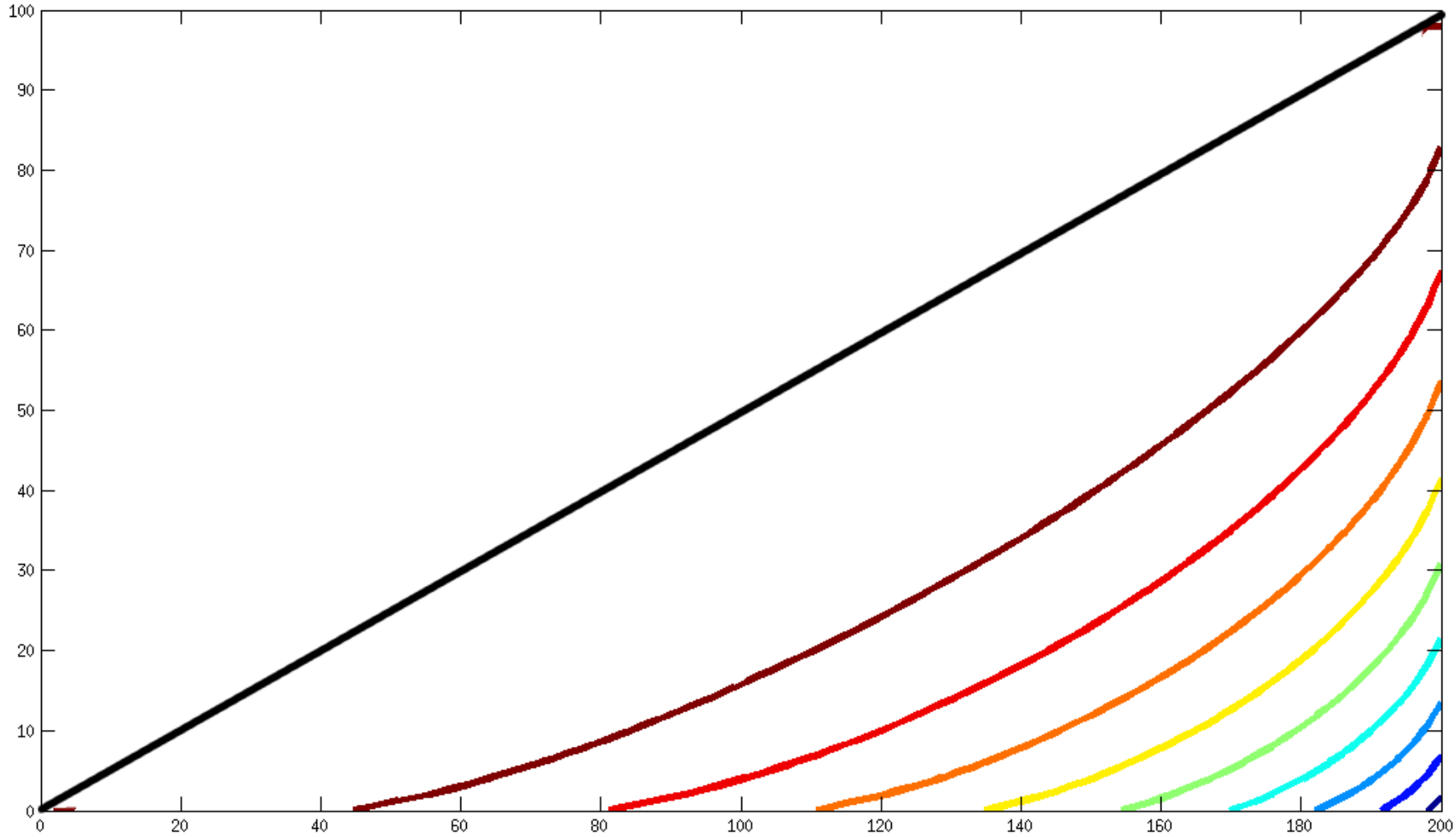
Статистические закономерности: $\mathrm{IStat}(p, n) \geqslant 3.$



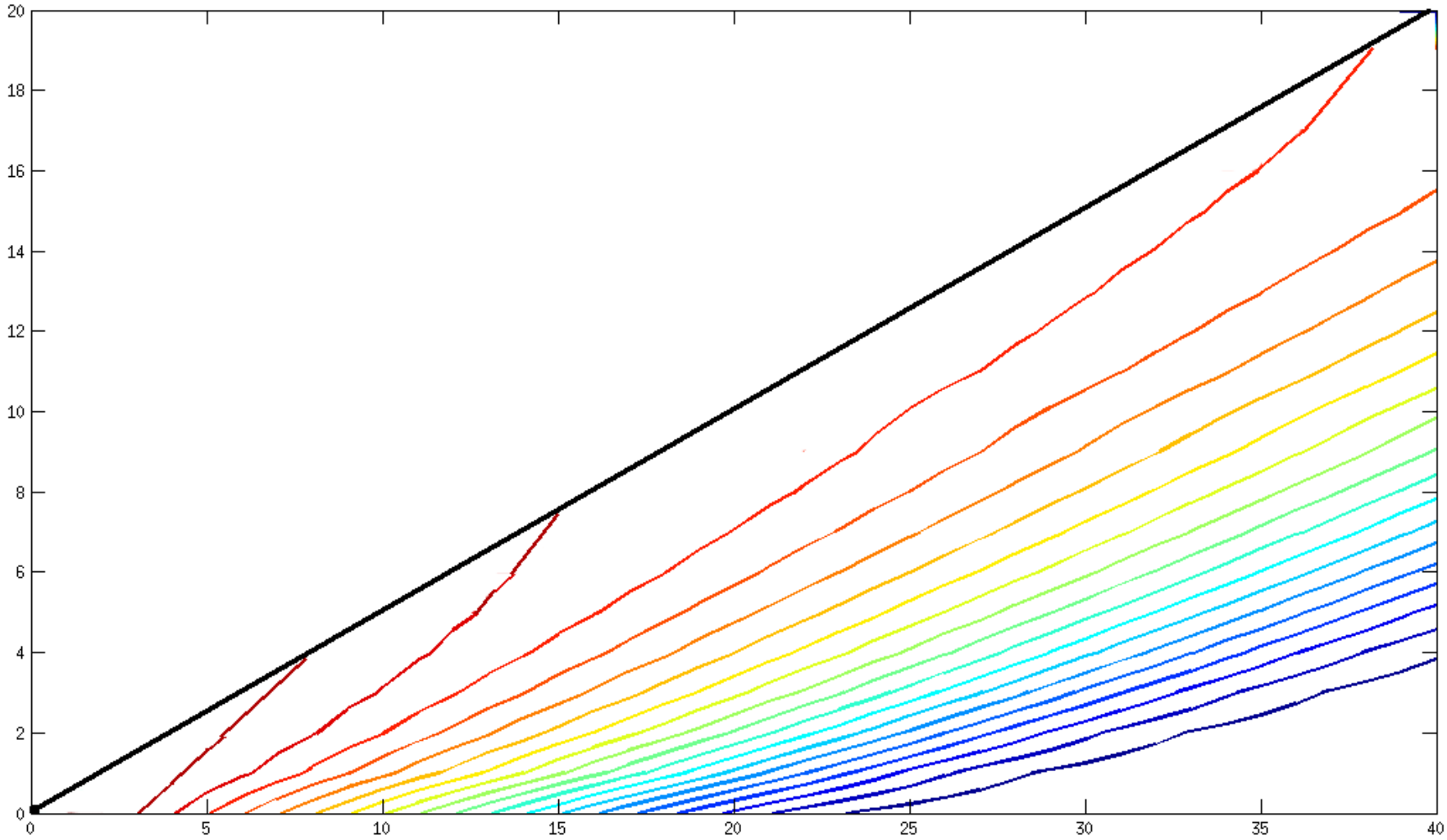
$P = 200$

$N = 100$

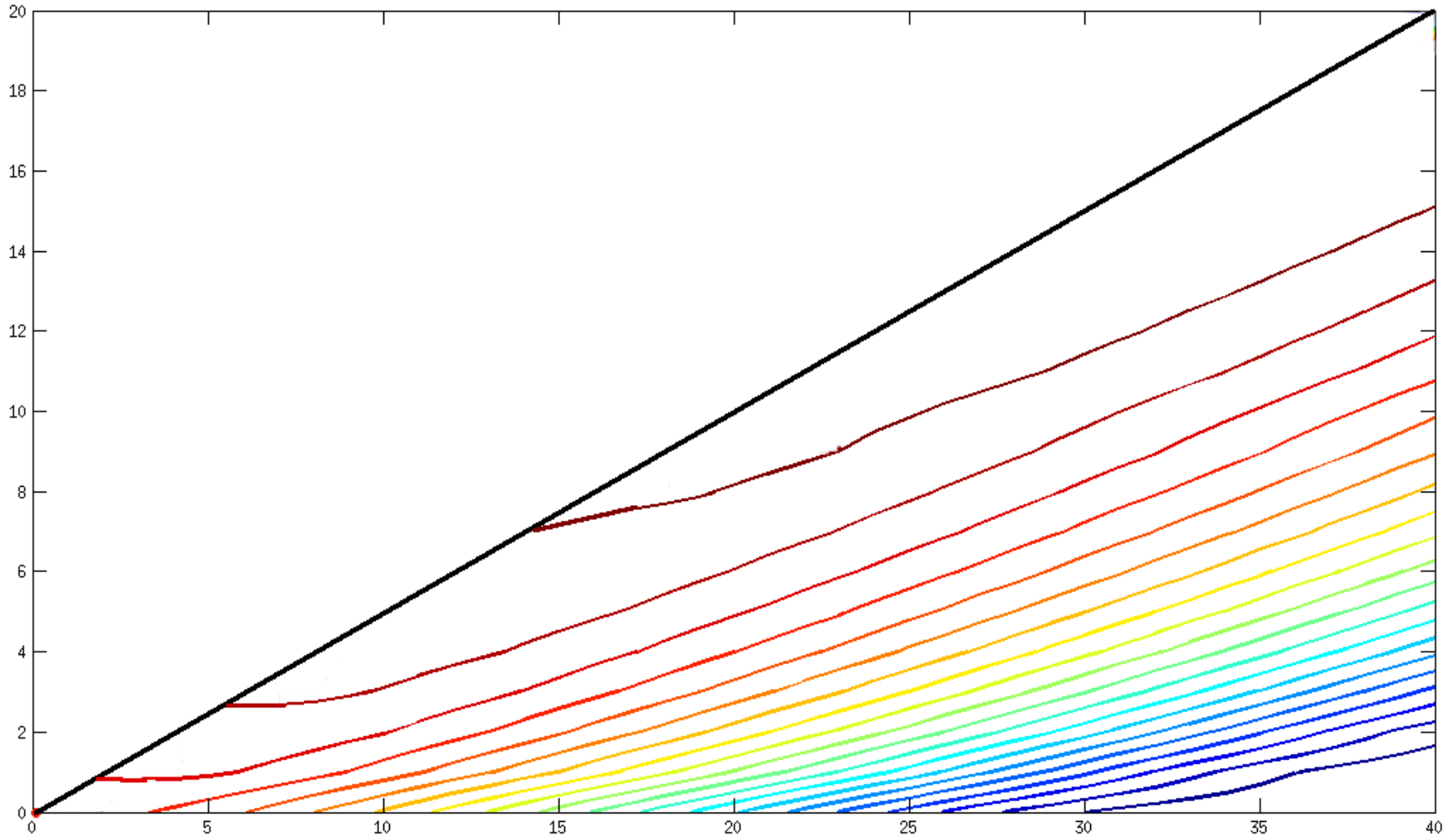
Линии уровня теста Фишера



Линии уровня теста Фишера
Малые p и n



Линии уровня плотности вероятности. Малые p и n



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 $y{=}c$, $y{\neq}c$:

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

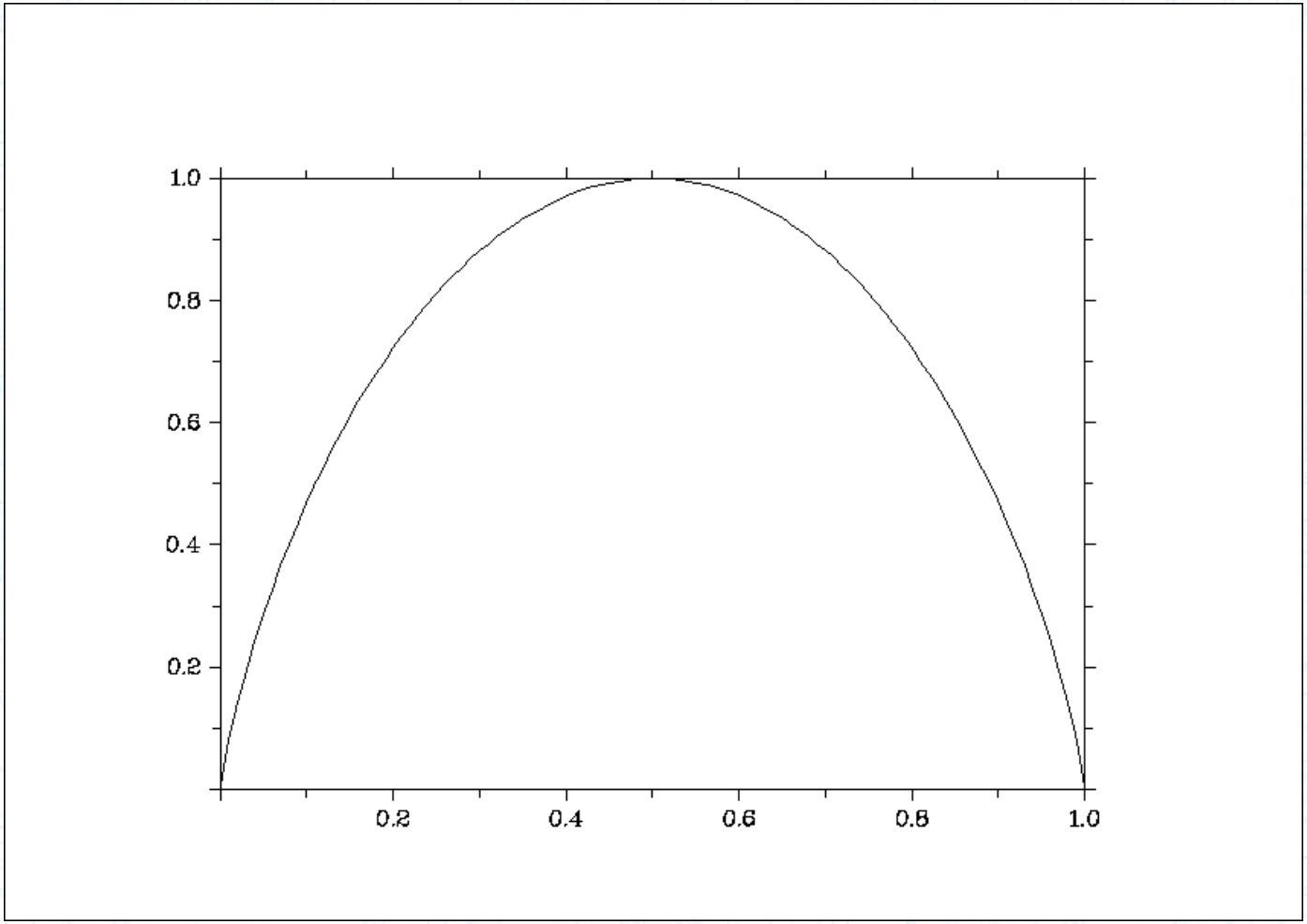
Энтропия выборки X^ℓ после получения информации $R(x_i)_{i=1}^{\ell}$:

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

Прирост информации (Information gain, IGain):

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

Энтропия для различных q



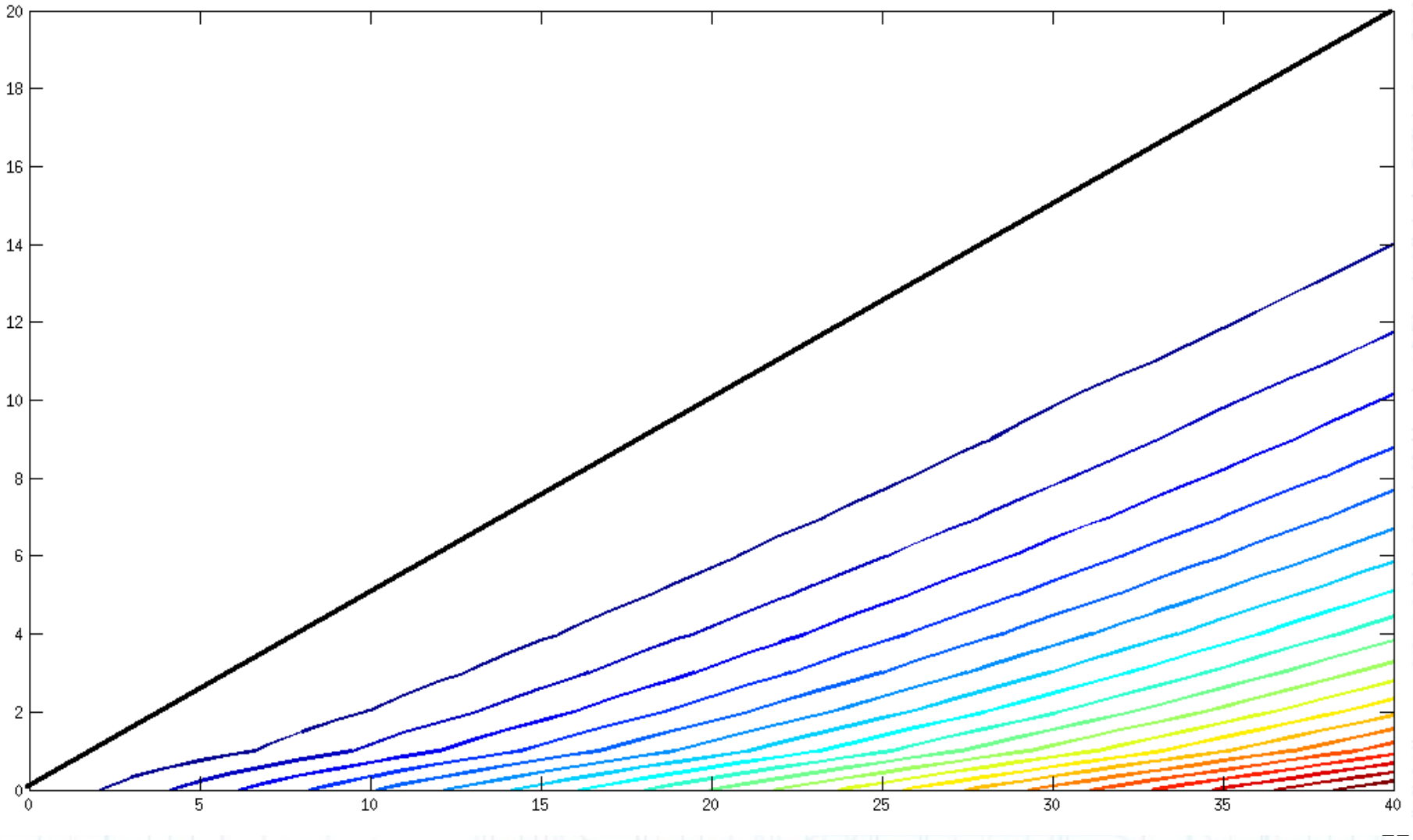
Соотношение статистического и энтропийного критериев

Энтропийный критерий IGain асимптотически эквивалентен статистическому IStat:

$$\mathsf{IStat}(p, n) \to \mathsf{IGain}(p, n) \quad \textit{при } \ell \to \infty$$

Доказательство: применить формулу Стирлинга к критерию IStat.

Линии уровня энтропийного критерия. Малые p и n



Построение закономерностей (rule induction)

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

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

2. *Конъюнкция* пороговых условий:

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

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

$$R(x) = \left[\sum_{j \in J} \big[a_j \leqslant f_j(x) \leqslant b_j\big] \geqslant d\right],$$

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

Построение закономерностей (rule induction)

4. *Полуплоскость* — линейная пороговая функция:

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

5. *Шар* — пороговая функция близости:

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

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

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

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

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

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;

1: начальное множество правил Z;
2: **повторять**
3: $\quad Z' := $ множество модификаций правил $R \in Z$;
4: $\quad$ удалить слишком похожие правила из $Z \cup Z'$;
5: $\quad$ оценить информативность всех правил $R \in Z'$;
6: $\quad Z :=$ наиболее информативные правила из $Z \cup Z'$;
7: **пока** правила продолжают улучшаться
8: **вернуть** Z.

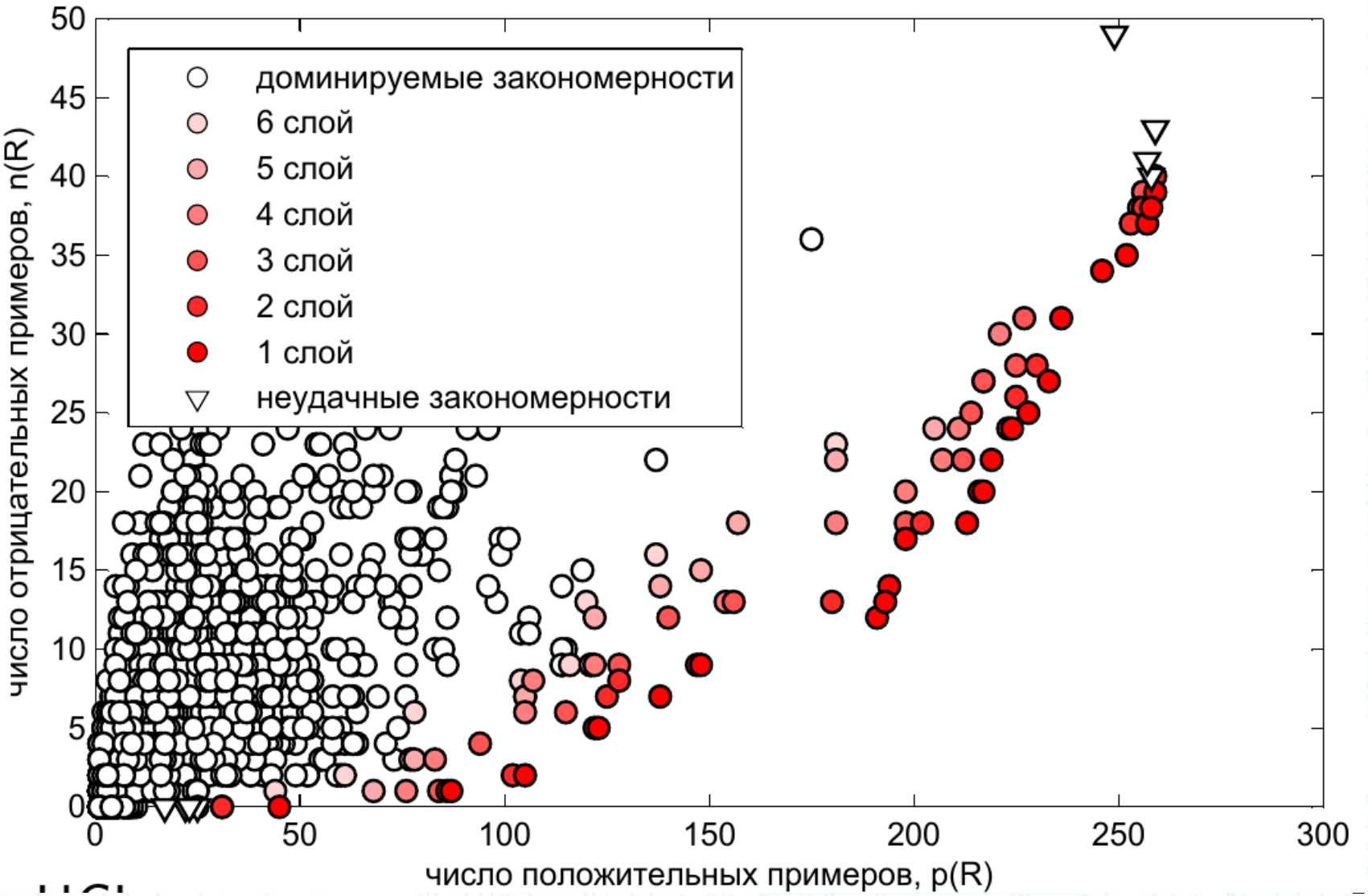
Частные случаи:

— стохастический локальный поиск (SLS)

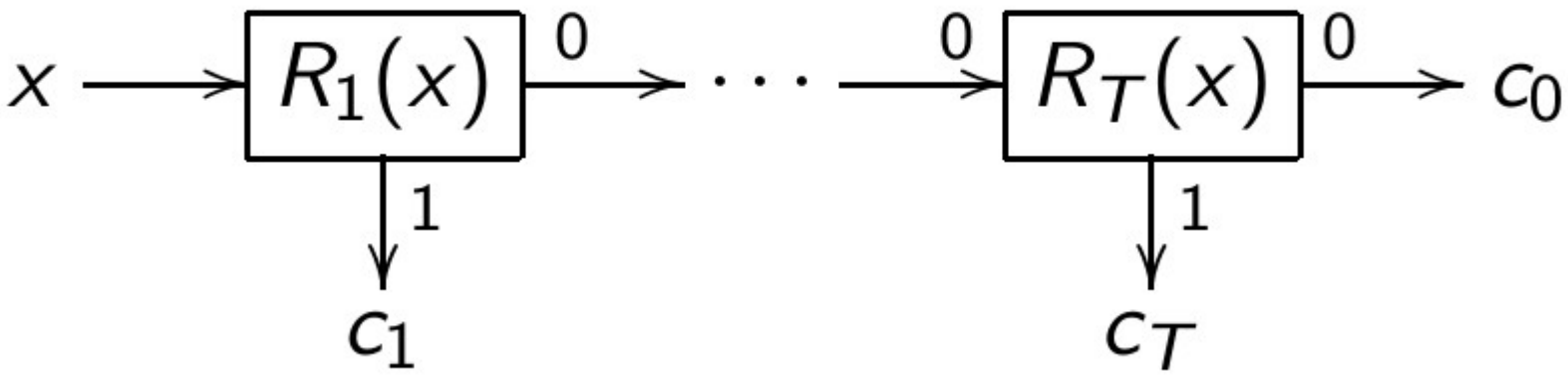
— генетические (эволюционные) алгоритмы

— метод ветвей и границ

Отбор закономерностей по Парето



Алгоритмы классификации. Решающий список



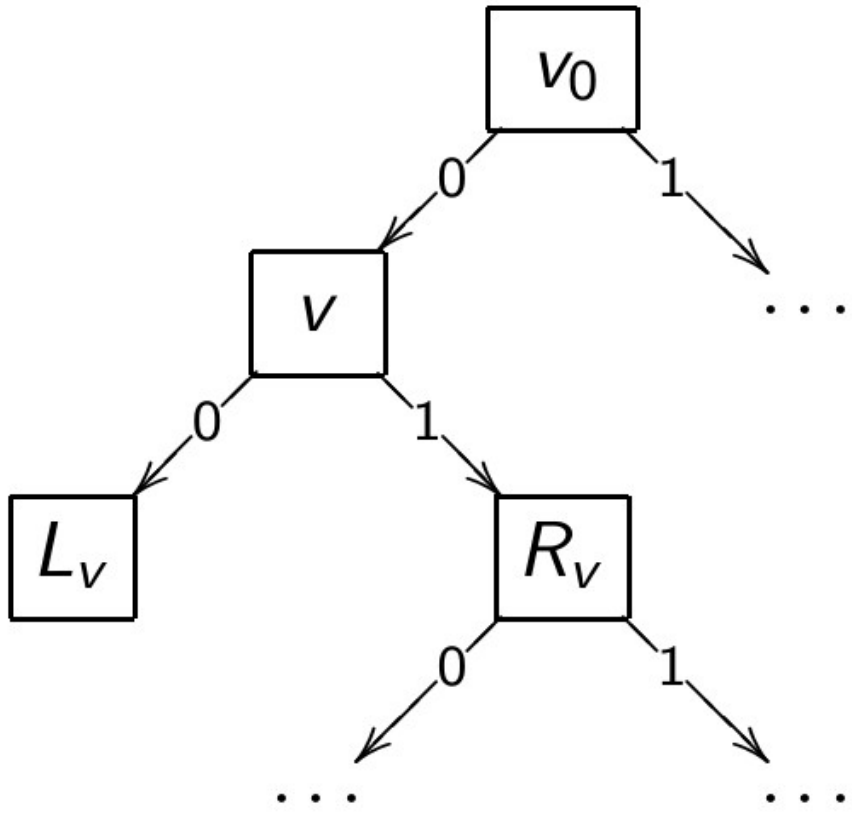
1: **для всех** $t = 1, \ldots, T$
2: **если** $R_t(x) = 1$ **то**
3: **вернуть** c_t;
4: **вернуть** c_0 — отказ от классификации объекта x

Жадный алгоритм построения решающего списка

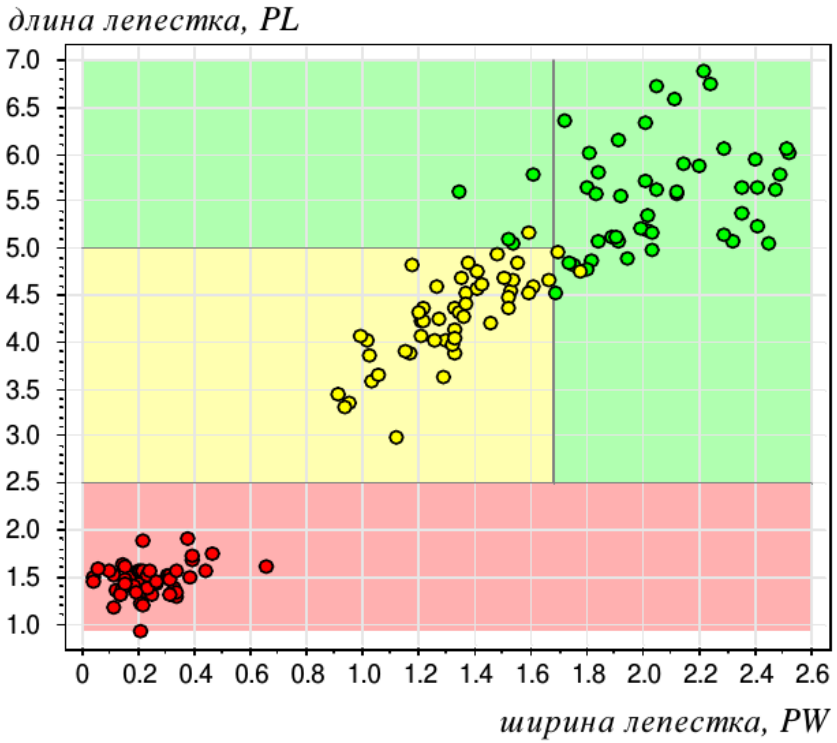
- $U := X_\ell$; $t := 1$
- Повторять:
  - Выбор класса =: c_t
  - Выбор предиката R_t: $I(R_t,U) \to \max$
  - $U := \{ x \in U \mid R_t(x) = 0 \}$
- Пока: $I > I_{min}$; $|U| > \ell_0$

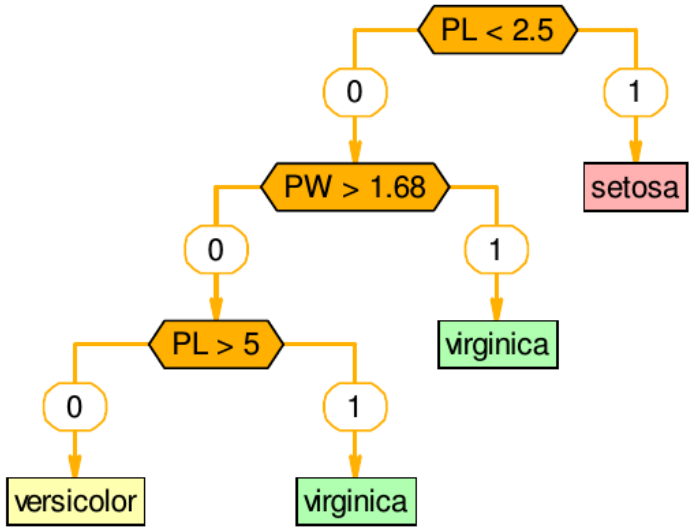
Решающее дерево

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.



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	$r_4(x) = [PL > 2.5] \wedge [PL \leqslant 5] \wedge [PW < 1.68]$

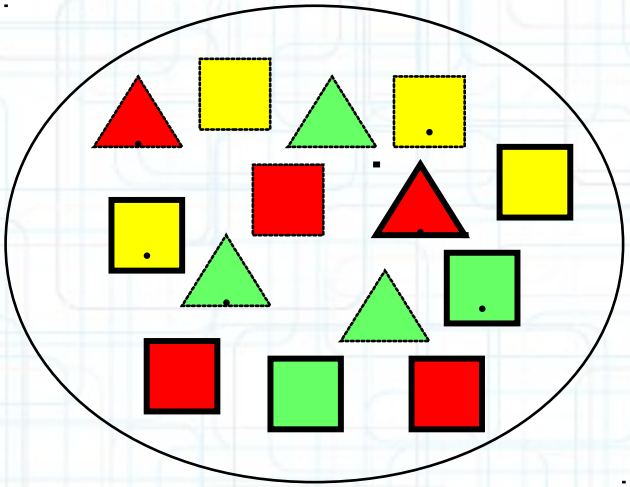
Жадный алгоритм построения решающего дерева

- Функция:
- Tree buildTree(U) {
  - Выбор предиката β_v: $I(\beta_v, U) \to \max$
  - $U_0 := \{ x \in U \mid \beta_v(x) = 0 \}$
  - $U_1 := \{ x \in U \mid \beta_v(x) = 1 \}$
  - Если $|U_0| < \ell_0$ или $|U_1| < \ell_0$ вернуть лист
  - Иначе:
    - L_v := buildTree(U_0)
    - R_v := buildTree(U_1)
- }

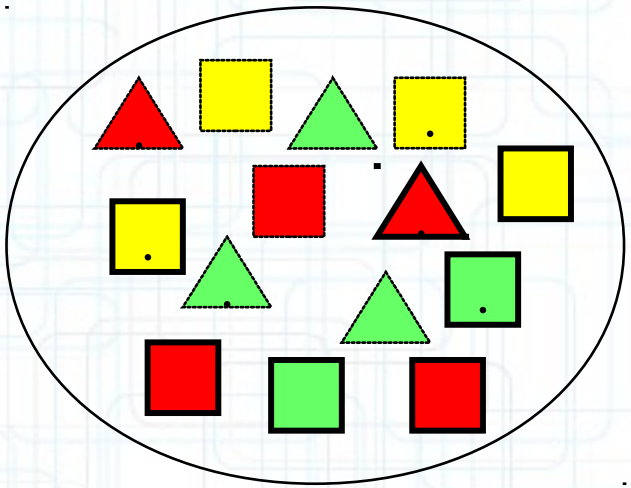
Пример: треугольники и квадраты

#	Attribute			Shape
	Color	Outline	Dot	
1	green	dashed	no	triange
2	green	dashed	yes	triange
3	yellow	dashed	no	square
4	red	dashed	no	square
5	red	solid	no	square
6	red	solid	yes	triange
7	green	solid	no	square
8	green	dashed	no	triange
9	yellow	solid	yes	square
10	red	solid	no	square
11	green	solid	yes	square
12	yellow	dashed	yes	square
13	yellow	solid	no	square
14	red	dashed	yes	triange

Обучающая выборка



Энтропия



- 5 треугольников
- 9 квадратов
- Вероятности классов

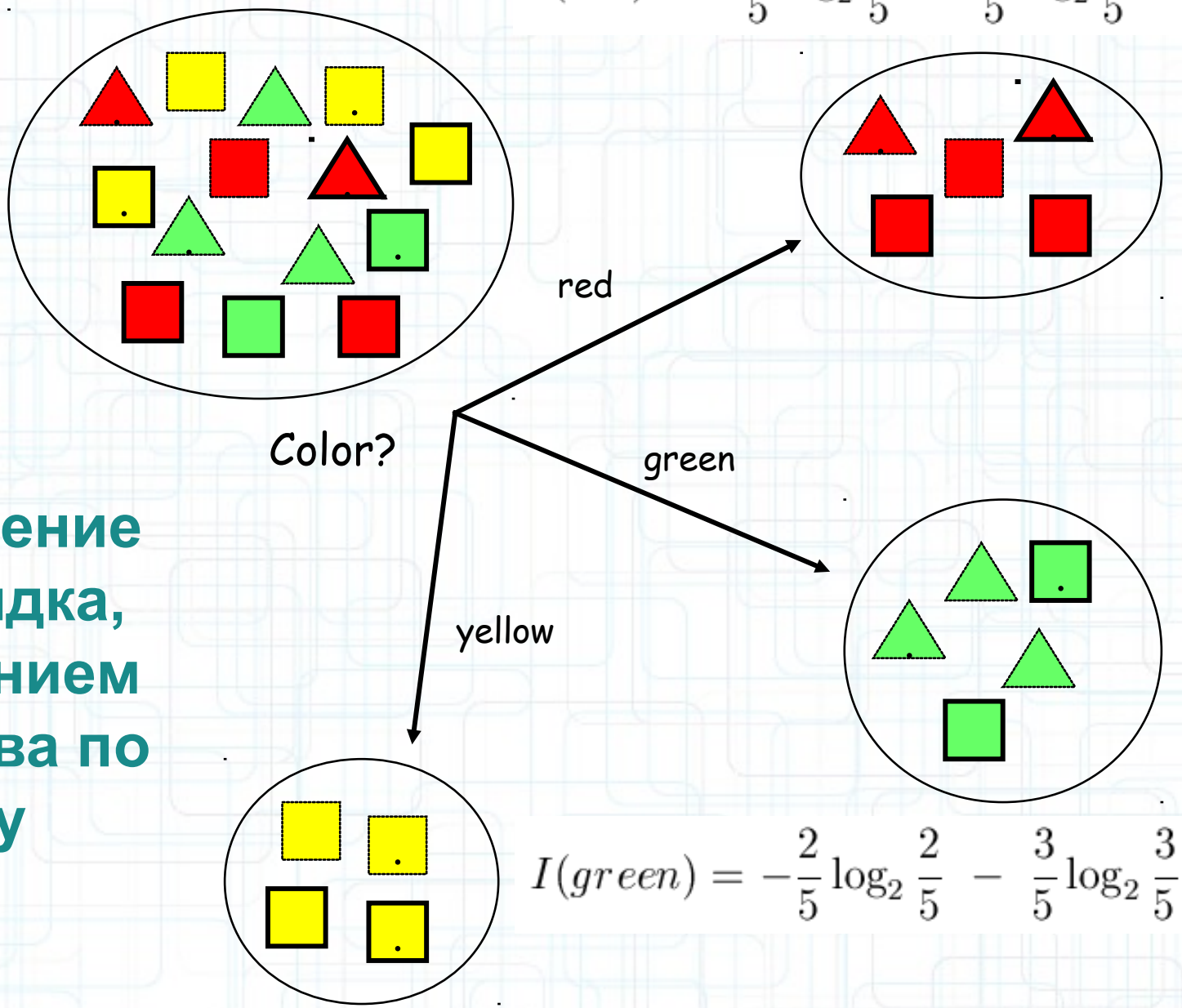
$$p(\square) = \frac{9}{14}$$

$$p(\triangle) = \frac{5}{14}$$

энтропия

$$I = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.940 \text{ bits}$$

$$I(red) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.971 \text{ bits}$$

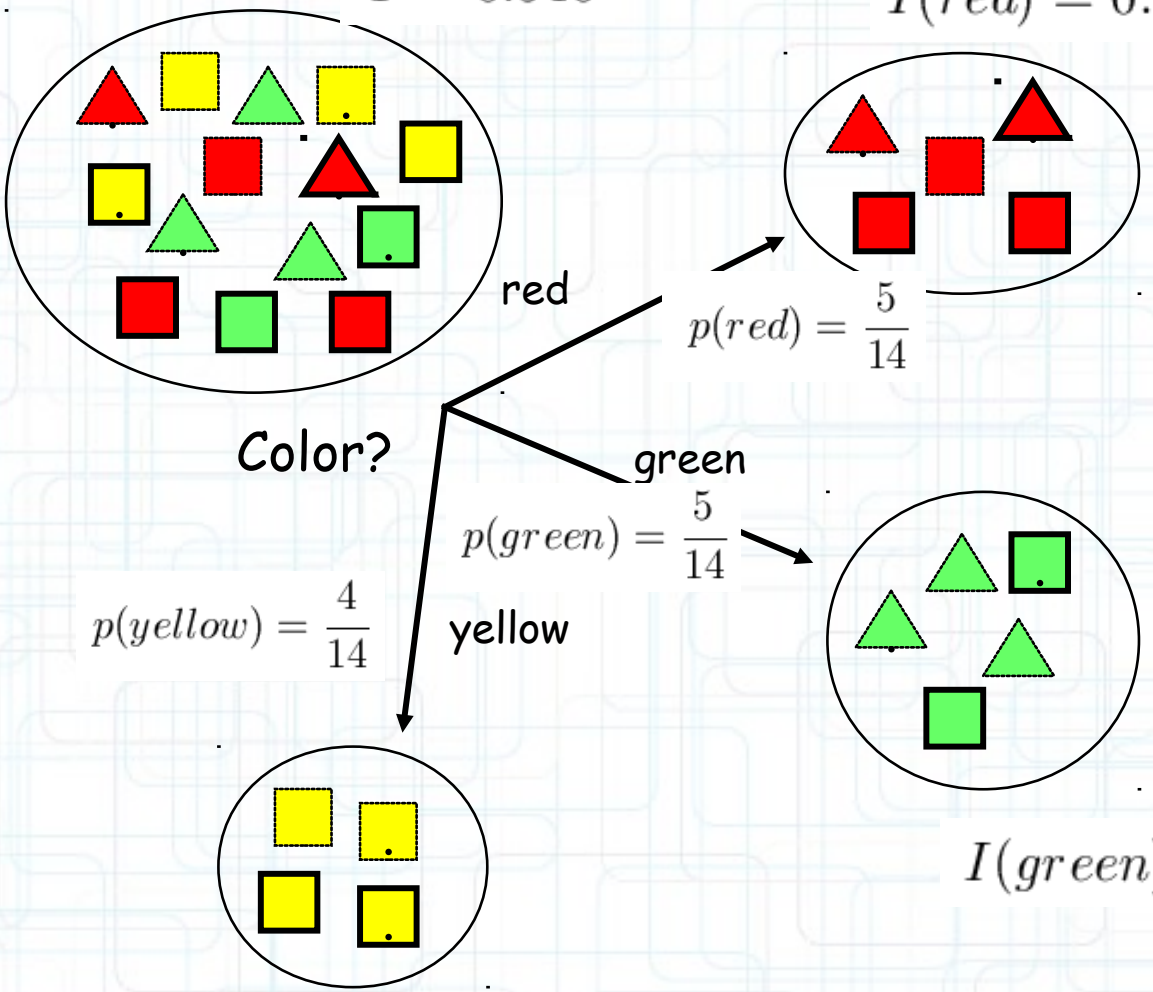


Уменьшение беспорядка, разделением множества по цвету

$$I(green) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0.971 \text{ bits}$$

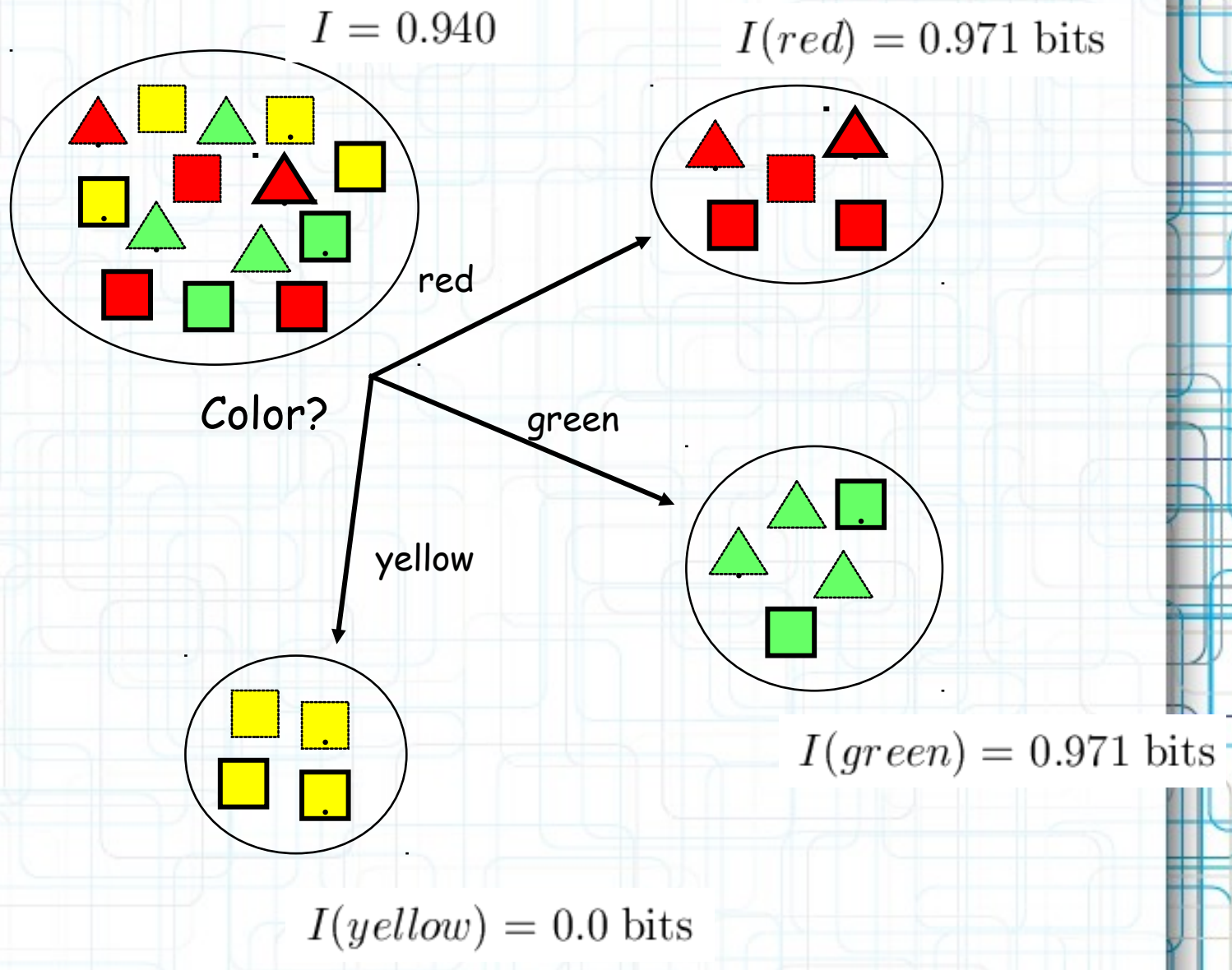
$$I(yellow) = -\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4} = 0.0 \text{ bits}$$

Энтропия после разделения



$$I_{res}(\text{Color}) = \sum p(v)I(v) = \frac{5}{14}0.971 + \frac{5}{14}0.971 + \frac{4}{14}0.0 = 0.694 \; bits$$

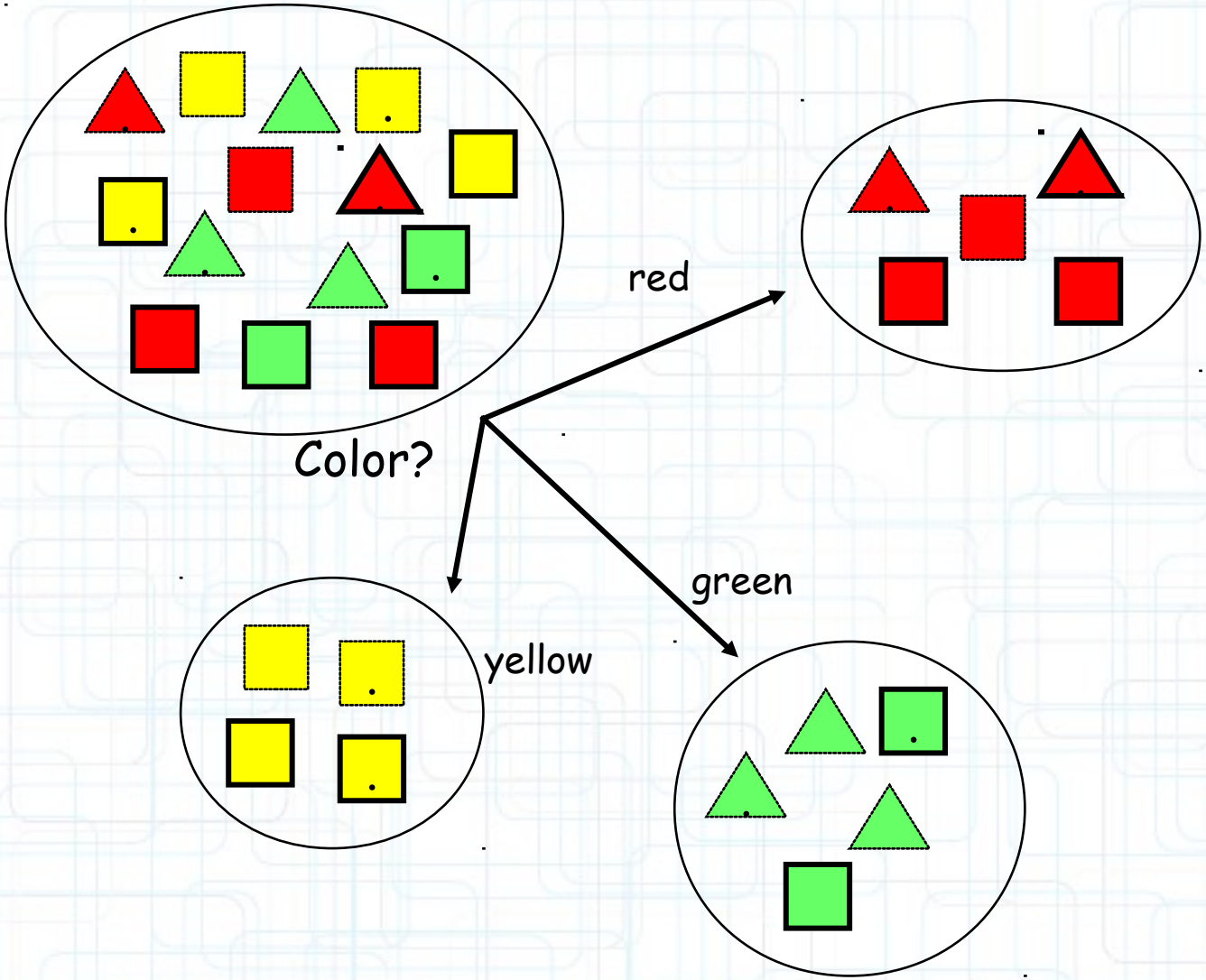
Прирост информации



$$Gain(\text{Color}) = I - I_{res}(\text{Color}) = 0.940 - 0.694 = 0.246\ bits$$

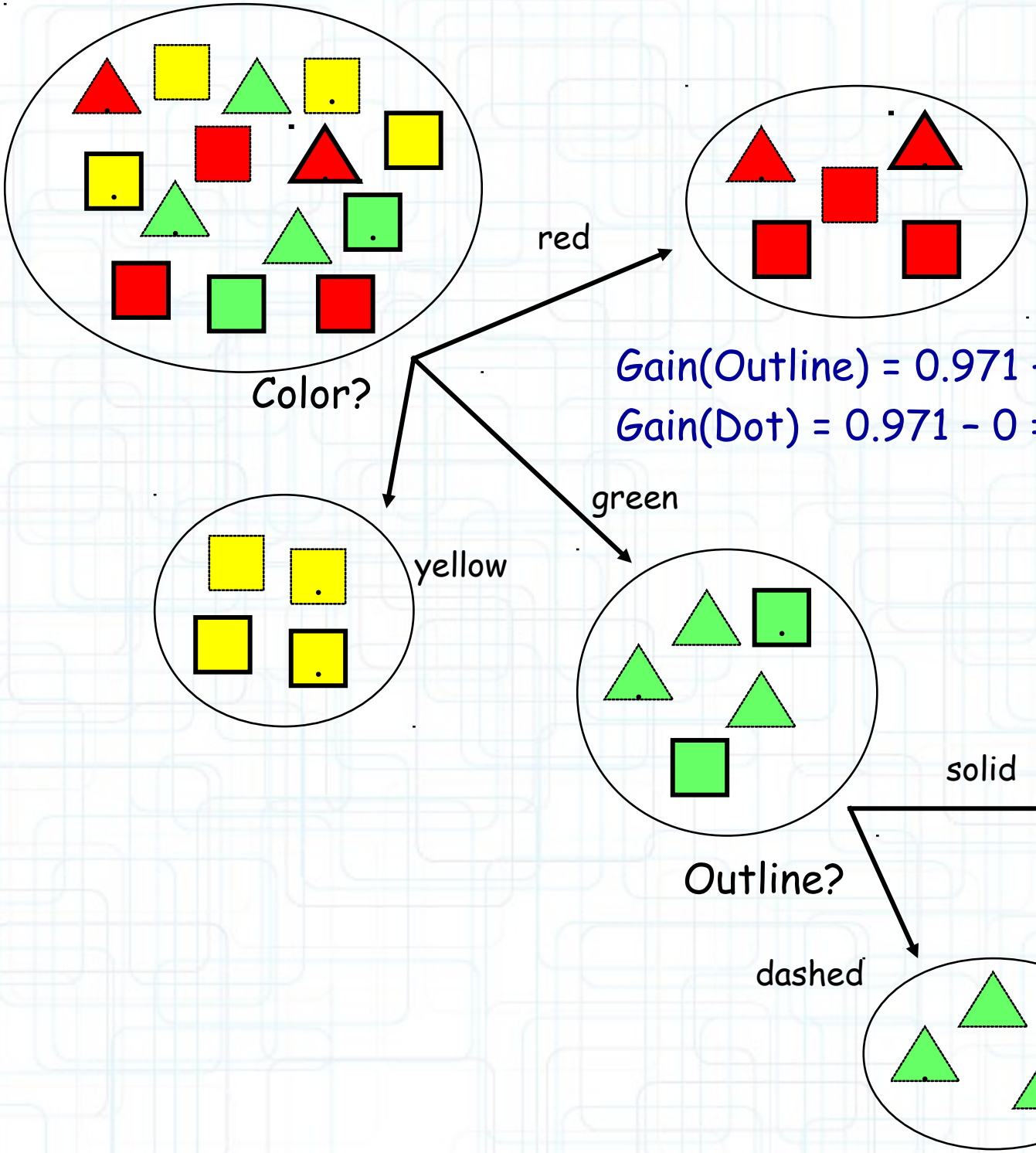
Прирост информации для каждого признака

- Признаки
  - Gain(Color) = 0.246
  - Gain(Outline) = 0.151
  - Gain(Dot) = 0.048
- Лучше всего разбивать множество по признаку, вносящему наибольший порядок



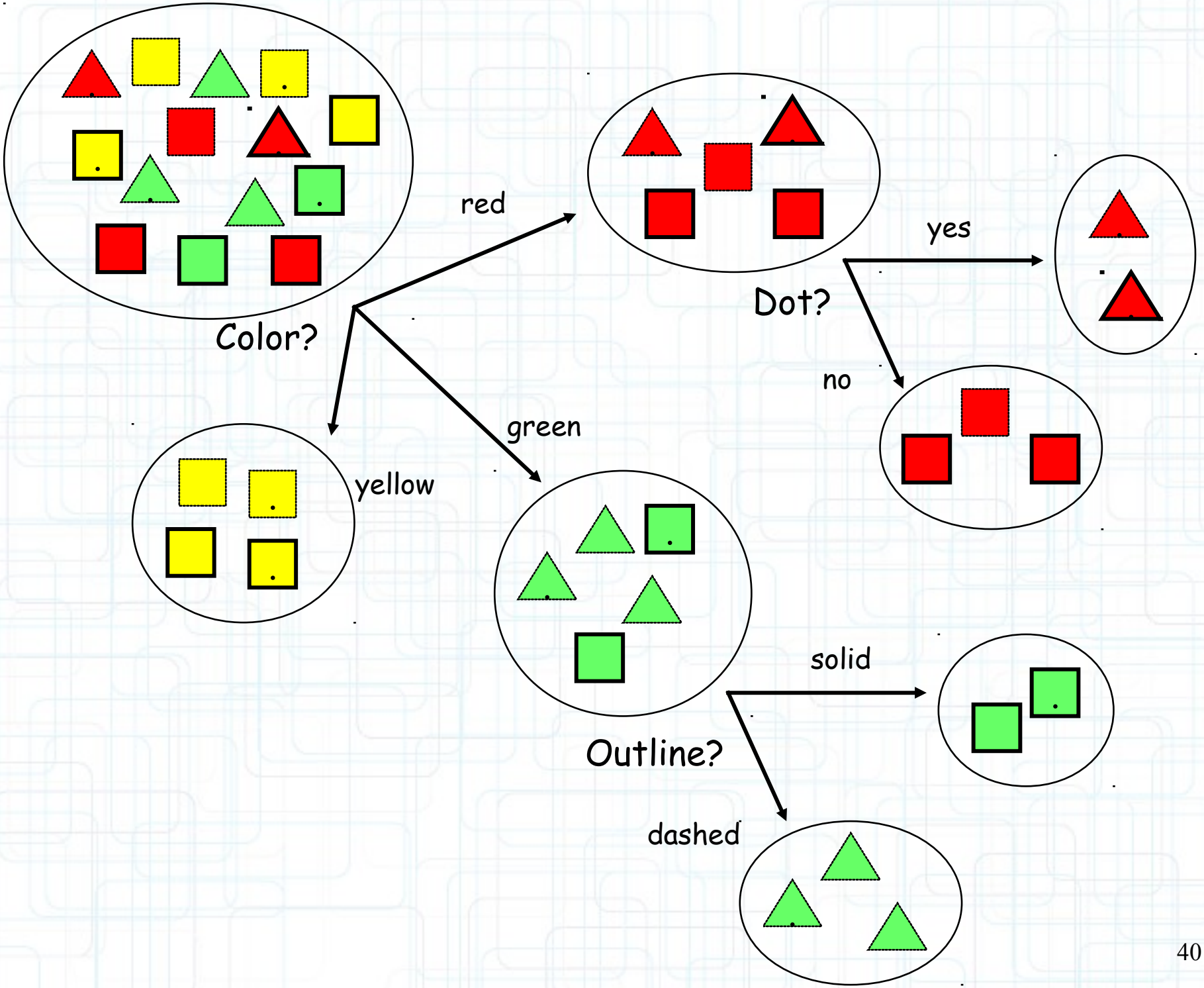
Gain(Outline) = 0.971 – 0 = 0.971 bits

Gain(Dot) = 0.971 – 0.951 = 0.020 bits

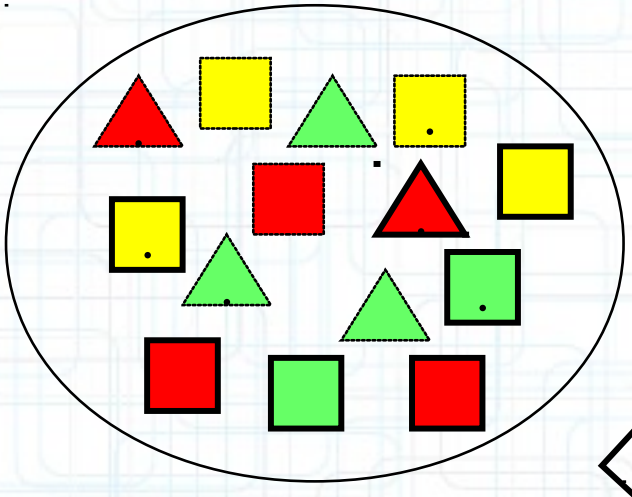


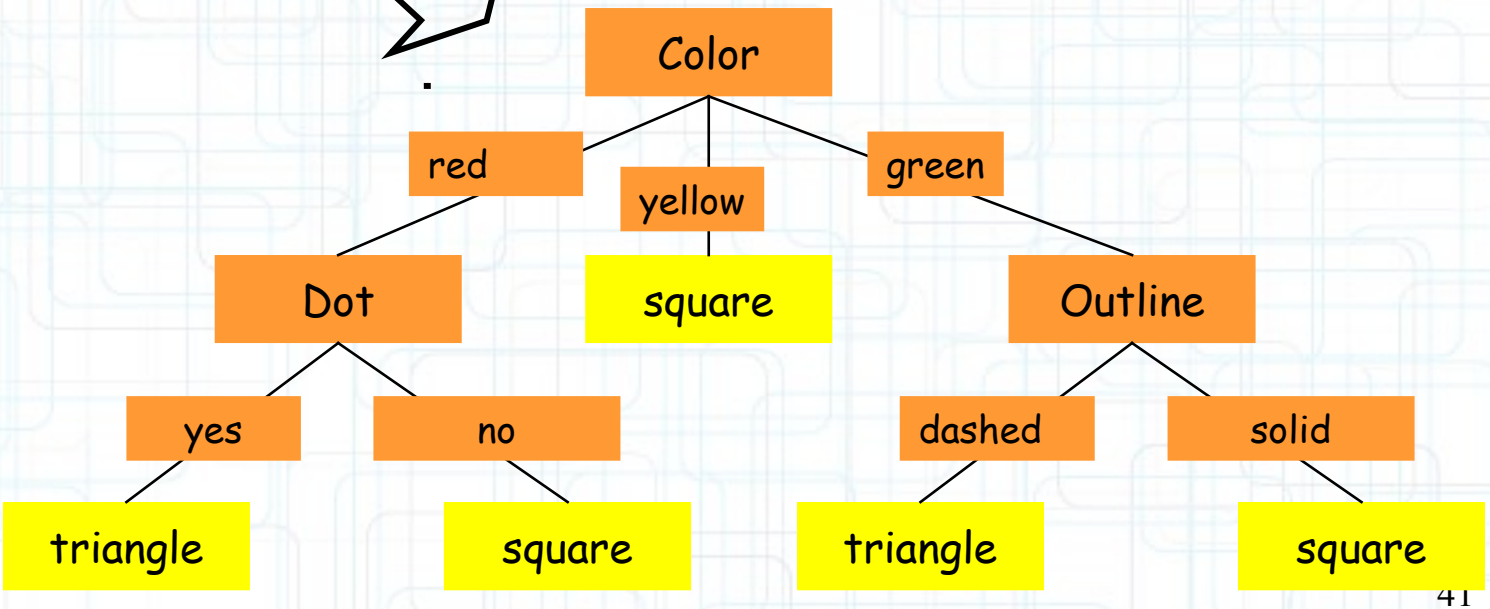
Gain(Outline) = 0.971 – 0.951 = 0.020 bits

Gain(Dot) = 0.971 – 0 = 0.971 bits


Color?
red
yellow
green
Dot?
yes
no
Outline?
solid
dashed

Итоговое дерево





Редукция дерева (pruning)

- Pre-pruning – критерий раннего останова. Досрочное прекращение ветвления, если информативность < порога или глубина велика.
- Post-pruning – пост-редукция. Простматриваем все внутренние вершины дерева и проверяем их качество на тестовой выборке (OutOfBag error). Заменяем листом, где качество после разделения ухудшается

Обобщение на случай задачи регрессии

- В каждом листе целевое значение определяется по методу наименьших квадратов
- Критирий информативности – среднеквадратическая ошибка

Небрежные решающие деревья (Oblivious Decision Tree)

- Для всех узлов на глубине h условие ветвления одинаково
- Дерево получается сбалансированным, на глубине h ровно 2^{h-1} вершин

Пример: задача XOR, $H = 2$.