Введение
в машинное обучение

Н.Ю. Золотых

zolotykh@vmk.unn.ru

Нижегородский гос. университет им. Н. И. Лобачевского

При поддержке компании Интел

План

- Что такое машинное обучение?
- Обучение с учителем
  - Метод k ближайших соседей
  - Машина опорных векторов (SVM)
  - Деревья решений
  - Бустинг
  - Баггинг
    - Баггинг
    - Random Forests
- Обучение без учителя: кластеризация

1. Что такое *машинное обучение* (*machine learning*)?

Машинное обучение — процесс, в результате которого машина (компьютер) способна показывать поведение, которое в нее не было *явно* заложено (запрограммировано).

A.L. Samuel Some Studies in Machine Learning Using the Game of Checkers // IBM Journal. July 1959. P. 210–229.

Говорят, что компьютерная программа *обучается* на основе опыта E по отношению к некоторому классу задач T и меры качества P, если качество решения задач из T, измеренное на основе P, улучшается с приобретением опыта E.

T.M. Mitchell Machine Learning. McGraw-Hill, 1997.

- На практике фаза обучения может предшествовать фазе работы алгоритма (например, детектирование лиц на фотокамере)
- или обучение (и дополнительное обучение) может проходить в процессе функционирования алгоритма (например, определение спама).

1.1. Сферы приложения

- Компьютерное зрение
- Распознавание речи
- Компьютерная лингвистика и обработка естественных языков
- Медицинская диагностика
- Биоинформатика
- Техническая диагностика
- Финансовые приложения
- Поиск и рубрикация текстов
- Интеллектуальные игры
- Экспертные системы
- . . .

1.2. Смежные области

- Pattern Recognition (распознавание образов)
- Data Mining (интеллектуальный анализ данных)
- Artificial Intelligence (искусственный интеллект)

1.3. Ресурсы

- Wiki-портал http://www.machinelearning.ru
- Мой курс: http://www.uic.unn.ru/~zny/ml (презентации лекций, лабораторные работы, описание системы R, ссылки, ML для «чайников» и др.)
- *Воронцов К. В.* Машинное обучение (курс лекций)
  см. http://www.machinelearning.ru
- *Дьяконов А. Г.* Анализ данных, обучение по прецедентам, логические игры, системы WEKA, RapidMiner и MatLab (Практикум на ЭВМ кафедры математических методов прогнозирования). – М.: Издат. отдел ф-та ВМК МГУ им. М. В. Ломоносова, 2010.
- *Ng A.* Machine Learning Course (video, lecture notes, presentations, labs)
  http://ml-class.org
- *Hastie T., Tibshirani R., Friedman J.* The elements of statistical learning: Data Mining, Inference, and Prediction. 2nd Edition. Springer, 2009
  http://www-stat.stanford.edu/~tibs/ElemStatLearn/
- . . .

1.4. Software

- Библиотека машинного зрения OpenCV (C, C++, интерфейс для Python) (раздел ML)
- Система для статистических вычислений R
- Библиотека алгоритмов для анализа данных Weka (Java)
- Пакет для решения задач машинного обучения и анализа данных Orange
- Система для решения задач машинного обучения и анализа данных RapidMiner
- . . .
- *Данные для экспериментов*: UCI Machine Learning Repository
  http://archive.ics.uci.edu/ml/

1.5. Классификация задач машинного обучения

- Дедуктивное обучение
  (по общим правилам вывести следствие, применительное к конкретному случаю)
- Индуктивное обучение ($\approx$ статистическое обучение)
  (по эмпирическим данным восстановить общую закономерность):
  - Обучение с учителем:
    - классификация
    - восстановление регрессии
    - ...
  - Обучение без учителя:
    - кластеризация
    - понижение размерности
    - ...
  - Обучение с подкрепление (reinforcement learning)
  - Активное обучение
  - ...

2. Обучение с учителем

Множество $\mathscr{X}$ — *объекты, примеры, ситуации, входы* (samples)

Множество $\mathscr{Y}$ — *ответы, отклики, «метки», выходы* (responses)

Имеется некоторая зависимость (детерминированная или вероятностная), позволяющая по $x \in \mathscr{X}$ предсказать $y \in \mathscr{Y}$.

т. е. если зависимость детерминированная, существует функция $f^* : \mathscr{X} \to \mathscr{Y}$.

Зависимость известна только на объектах из *обучающей выборки*:

$$\{(x^{(1)}, y_1),\ (x^{(2)}, y_2),\ \ldots,\ (x^{(N)}, y_N)\}$$

Прецедент $(x^{(i)}, y_i) \in \mathscr{X} \times \mathscr{Y}$.

Задача *обучения с учителем*: *восстановить* (*аппроксимировать*) зависимость, т. е. построить функцию (*решающее правило*) $f : \mathscr{X} \to \mathscr{Y}$, по новым объектам $x \in X$ предсказывающую $y \in Y$:

$$y = f(x) \approx f^*(x).$$

2.1. Признаковые описания

Вход:

$$x = (x_1, x_2, \ldots, x_d) \in \mathscr{X} = Q_1 \times Q_2 \times \ldots \times Q_d,$$

где $Q_j = \mathbf{R}$ или Q_j — конечно

x_j — j-й *признак* (*свойство*, *атрибут*, *feature*) объекта x.

- Если Q_j конечно, то j-й признак — *номинальный* (*категориальный* или *фактор*).
  Если $|Q_j| = 2$, то признак *бинарный* и можно считать, например, $Q_j = \{0, 1\}$.
- Если Q_j конечно и упорядочено, то признак *порядковый*.
  Например, $Q = \{\text{холодно}, \text{прохладно}, \text{тепло}, \text{жарко}\}$
- Если $Q_j = \mathbf{R}$, то признак *количественный*.

Выход: $y \in \mathscr{Y}$

- $\mathscr{Y} = \mathbf{R}$ — *задача восстановления регрессии*
- $\mathscr{Y} = \{1, 2, \ldots, K\}$ — *задача классификации*. Номер класса $k \in \mathscr{Y}$

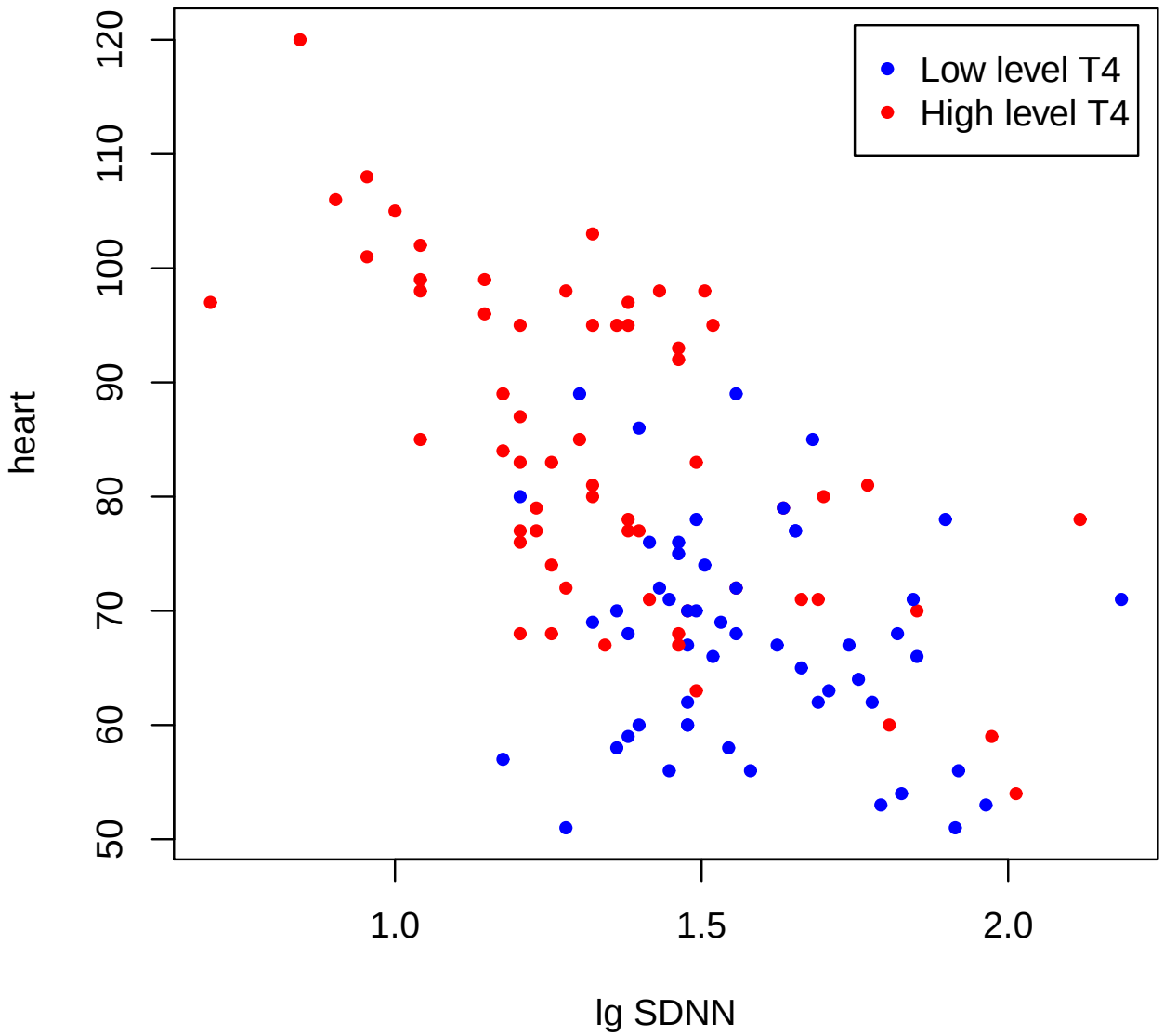
Пример 1. Медицинская диагностика

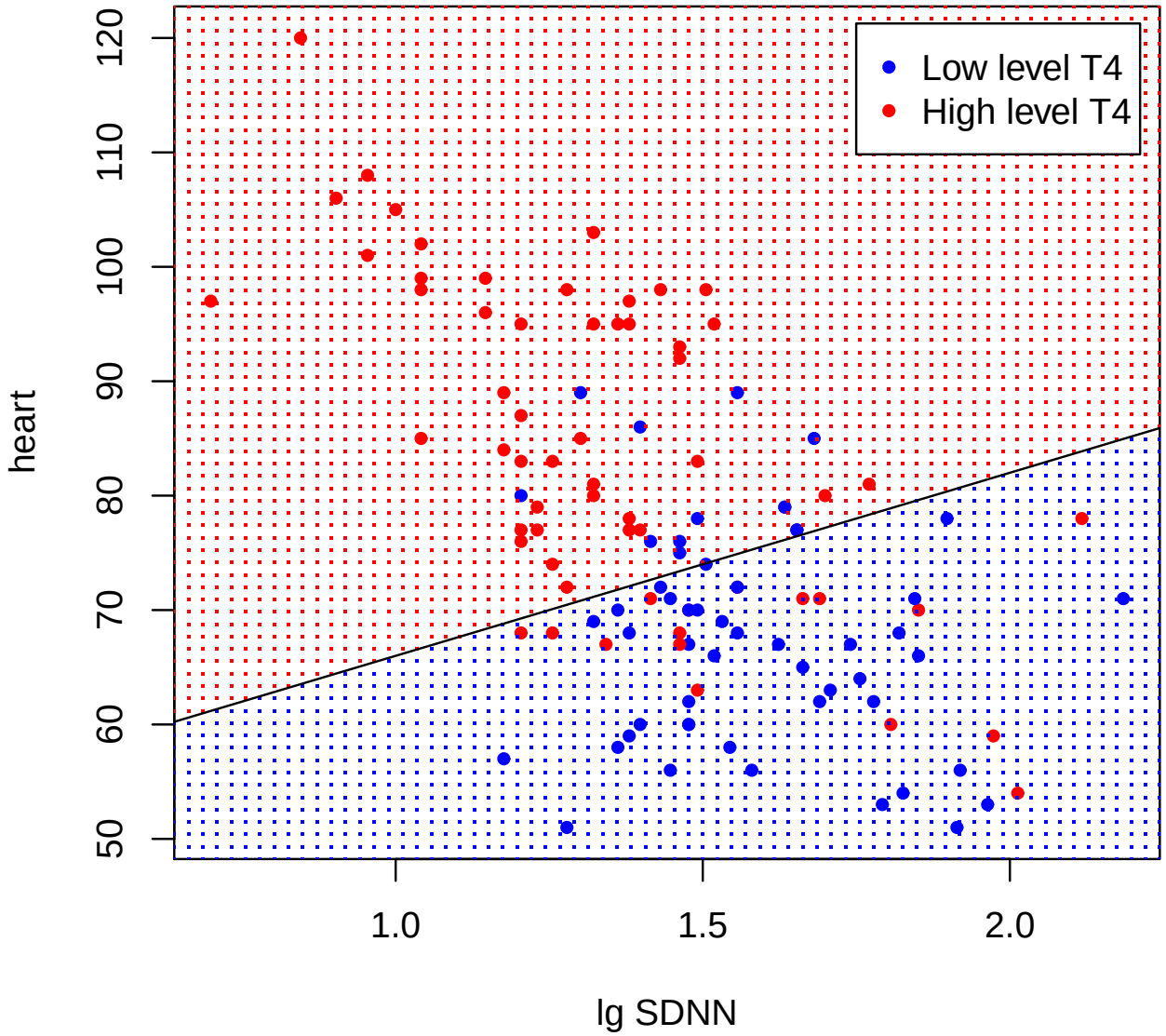
Имеются данные о 114 лицах с заболеванием щитовидной железы.
У 61 — повышенный уровень свободного гормона T4,
у 53 — уровень гормона в норме.

Для каждого пациента известны следующие показатели:

- $x_1 = \texttt{heart}$ — частота сердечных сокращений (пульс),
- $x_2 = \texttt{SDNN}$ — стандартное отклонение длительности интервалов между синусовыми сокращениями RR.

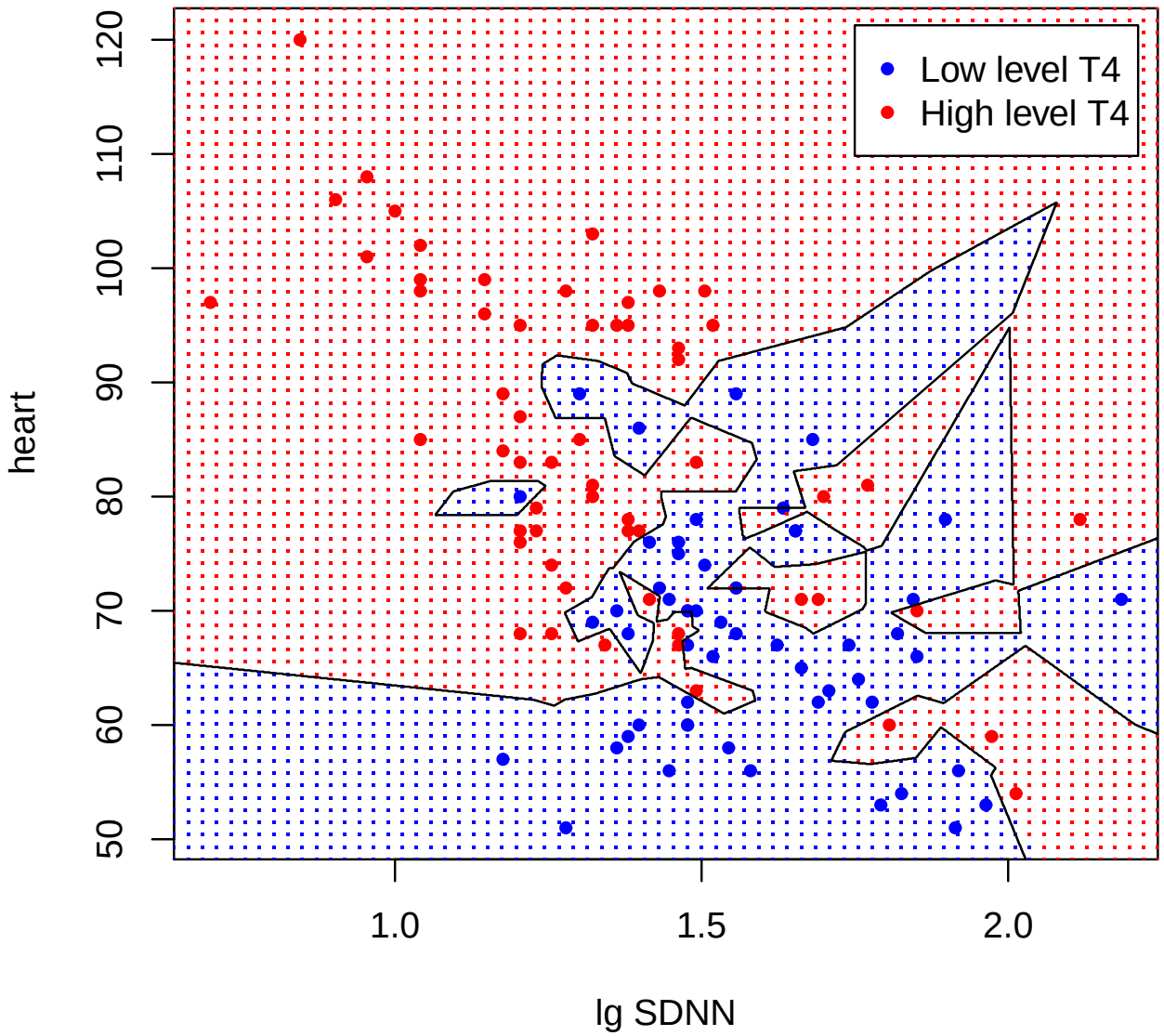
Можно ли научиться предсказывать (допуская небольшие ошибки) уровень свободного T4 по `heart` и `SDNN`?


Low level T4
High level T4
heart
50
60
70
80
90
100
110
120
1.0
1.5
2.0
lg SDNN



$16 \cdot \lg \texttt{SDNN} - \texttt{heart} + 50 = 0$

Ошибка на обучающей выборке — 23 %. Можно ли ее сделать меньше?



Метод ближайшего соседа (с масштабированием)

Ошибка на обучающей выборке — 0 %.

Малая ошибка на *обучающей выборке* не означает, что мы хорошо классифицируем *новые* объекты.

Итак, *малая ошибка на данных, по которым построено решающее правило, не гарантирует, что ошибка на новых объектах также будет малой*.

Обобщающая способность (*качество*) *решающего правила* — это способность решающего правила правильно предсказывать выход для новых объектов, не вошедших в обучающую выборку.

Переобучение — решающее правило хорошо решает задачу на обучающей выборке, но имеет плохую обобщающую способность.

Пример 2. Распознавание рукописных символов (цифр)

Научиться распознавать рукописный символ по его изображению.

Бинарное изображение — битовая матрица размера 32×32:

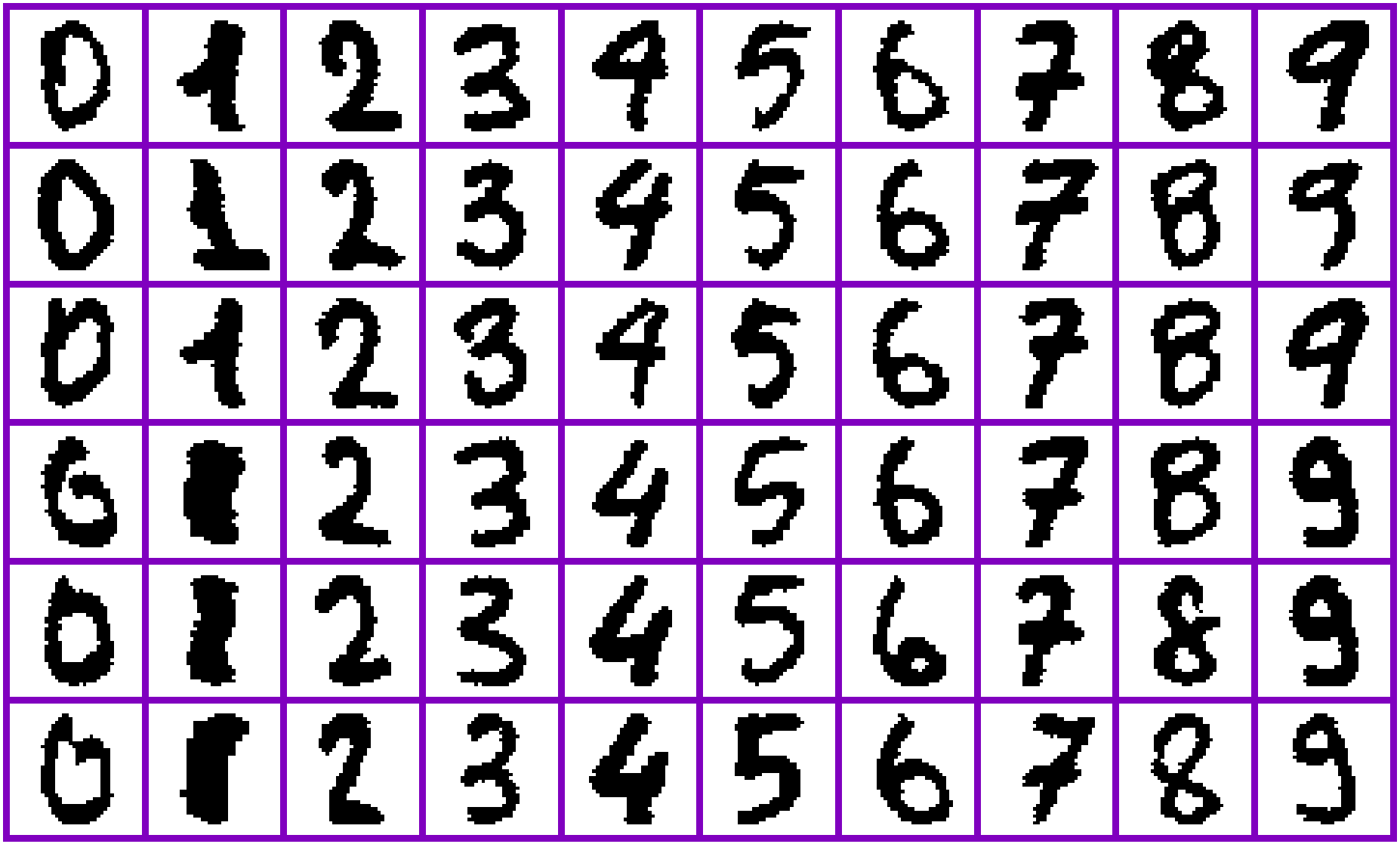
$$x \in \mathscr{X} = \{0,1\}^{32\times 32} = \{0,1\}^{1024}$$

$$\mathscr{Y} = \{0,1,2,\ldots,9\}$$

Это задача классификации.

optdigit http://www.ics.uci.edu/~mlearn/MLRepository.html — 1934 прецедента.

Некоторые объекты из обучающей выборки



Проблема построения признакового описания.

В задаче распознавания символов можно использовать признаковое описание на основе анализа контура изображения.

В примере `letter-recognition` `http://www.ics.uci.edu/~mlearn/MLRepository.html` распознавания печатных заглавных букв (26 классов) для кодирования изображений используется другой подход.

Входы (входы отмасштабированы и округлены, так, чтобы они принимали целые значения от 0 до 15; база содержит 20000 прецедентов):

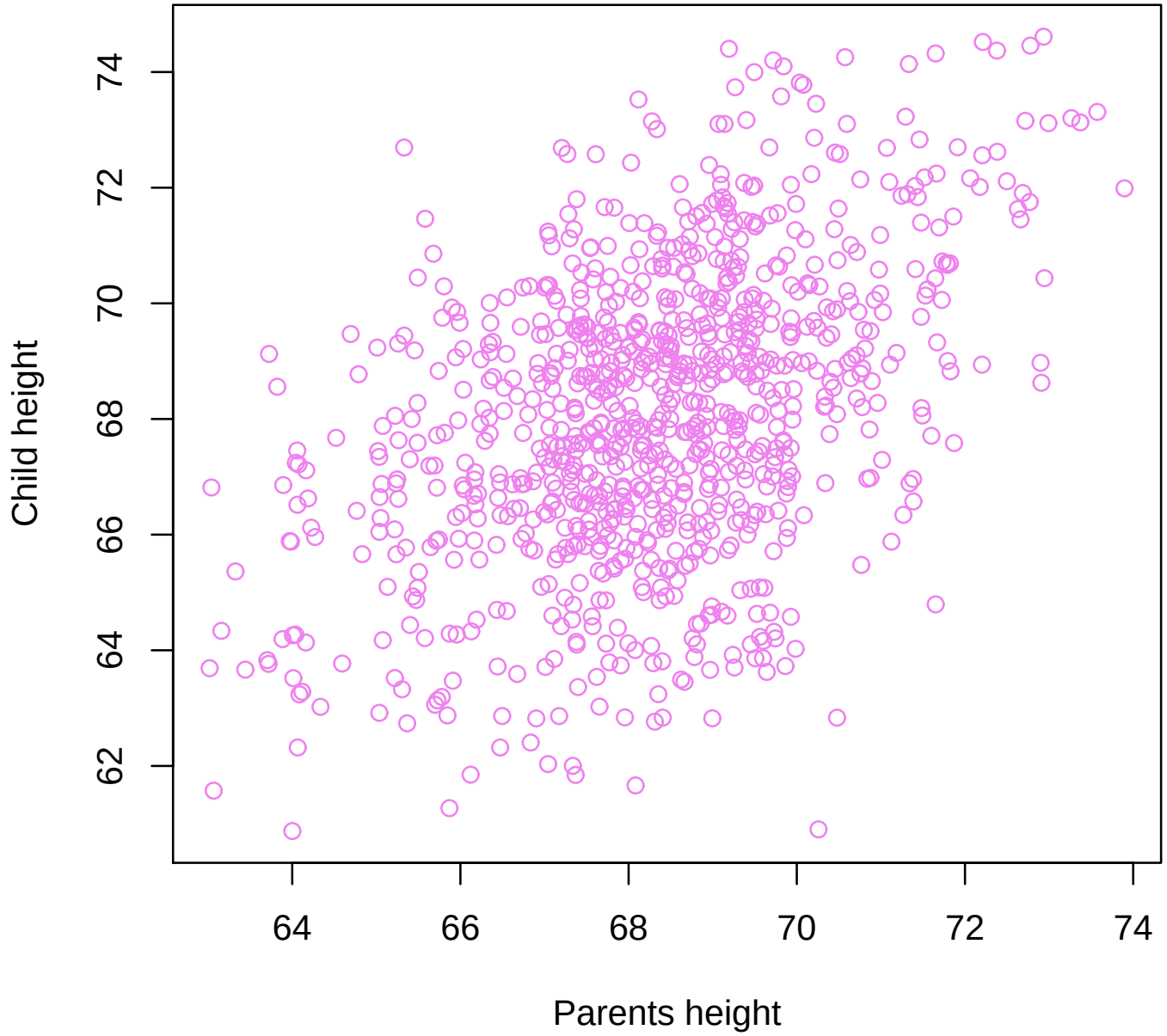
1. `x-box` — координата x левого нижнего угла обрамляющего прямоугольника,
2. `y-box` — координата y левого нижнего угла обрамляющего прямоугольника,
3. `width` — ширина прямоугольника,
4. `high` — высота прямоугольника,
5. `onpix` — общее число подсвеченных пикселей
6. `x-bar` — среднее значение координаты x для подсвеченных пикселей
7. `y-bar` — среднее значение координаты y для подсвеченных пикселей
8. `x2bar` — стандартное отклонение для координаты x подсвеченных пикселей
9. `y2bar` — стандартное отклонение для координаты y подсвеченных пикселей
10. `xybar` — коэффициент корреляции x и y подсвеченных пикселей
11. `x2ybr` — среднее значение x^2y
12. `xy2br` — среднее значение xy^2
13. `x-ege` — среднее значение числа отрезков при просмотре слева направо
14. `xegvy` — коэффициент корреляции между средним значением числа отрезков при просмотре слева направо и y
15. `y-ege` — среднее значение числа отрезков при просмотре снизу вверх
16. `yegvx` — коэффициент корреляции между средним значением числа отрезков при просмотре снизу вверх и x

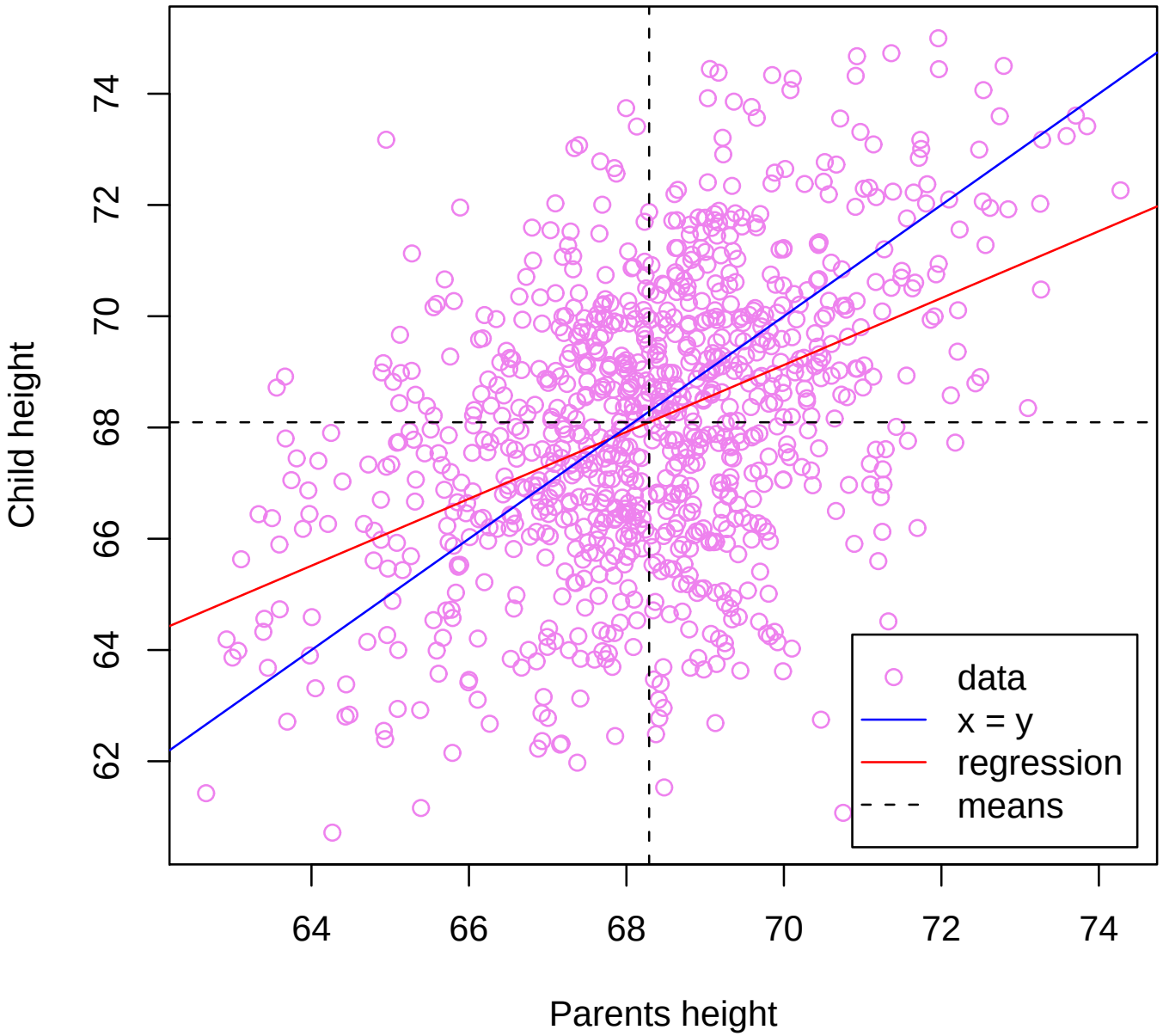
Пример 3. «Регрессия к середине» Ф. Гальтона

Фрэнсис Гальтона (1822–1911)

«Регрессия к середине в наследовании роста» (1885)

Зависимость роста взрослого ребенка от роста родителей в исследовании Ф. Гальтона

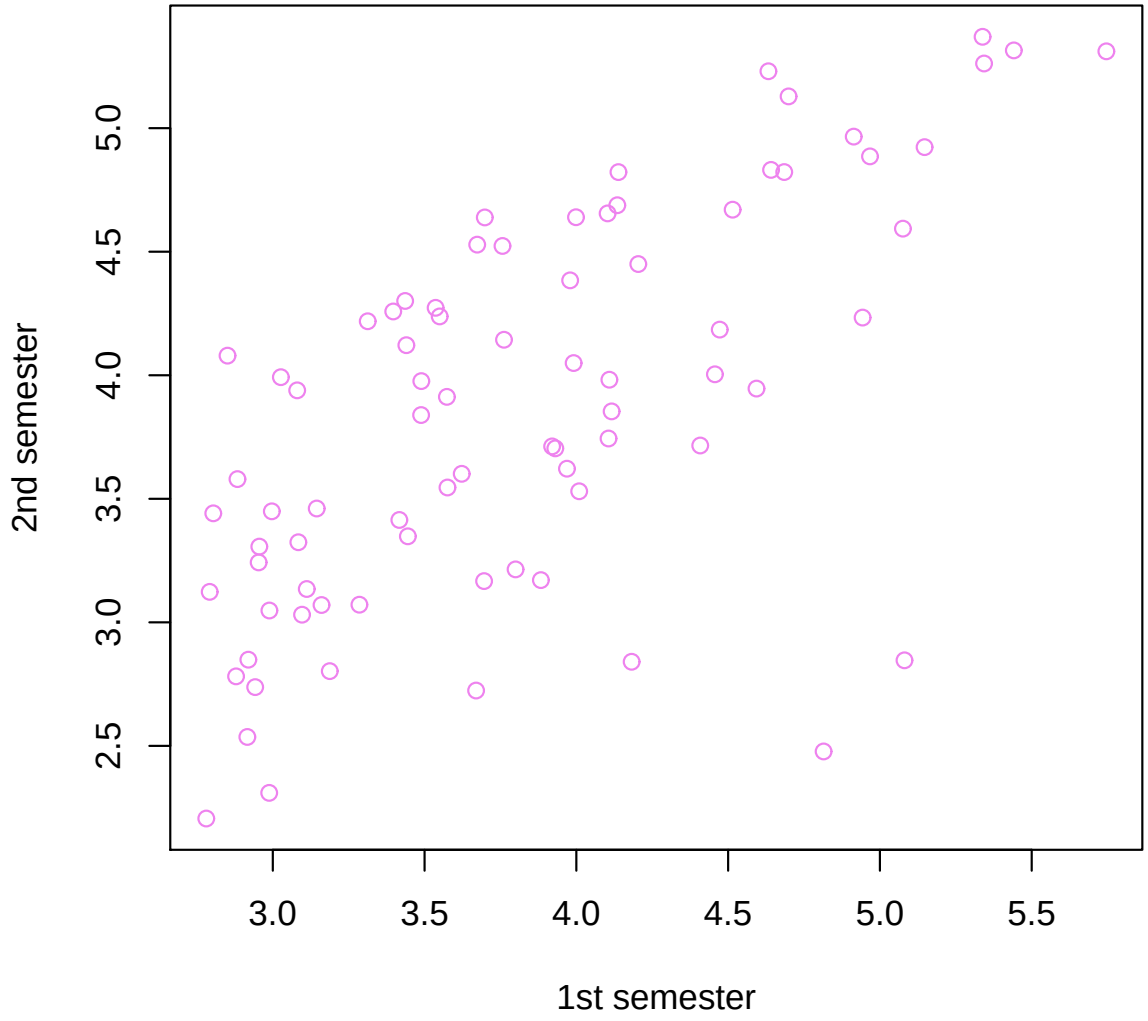




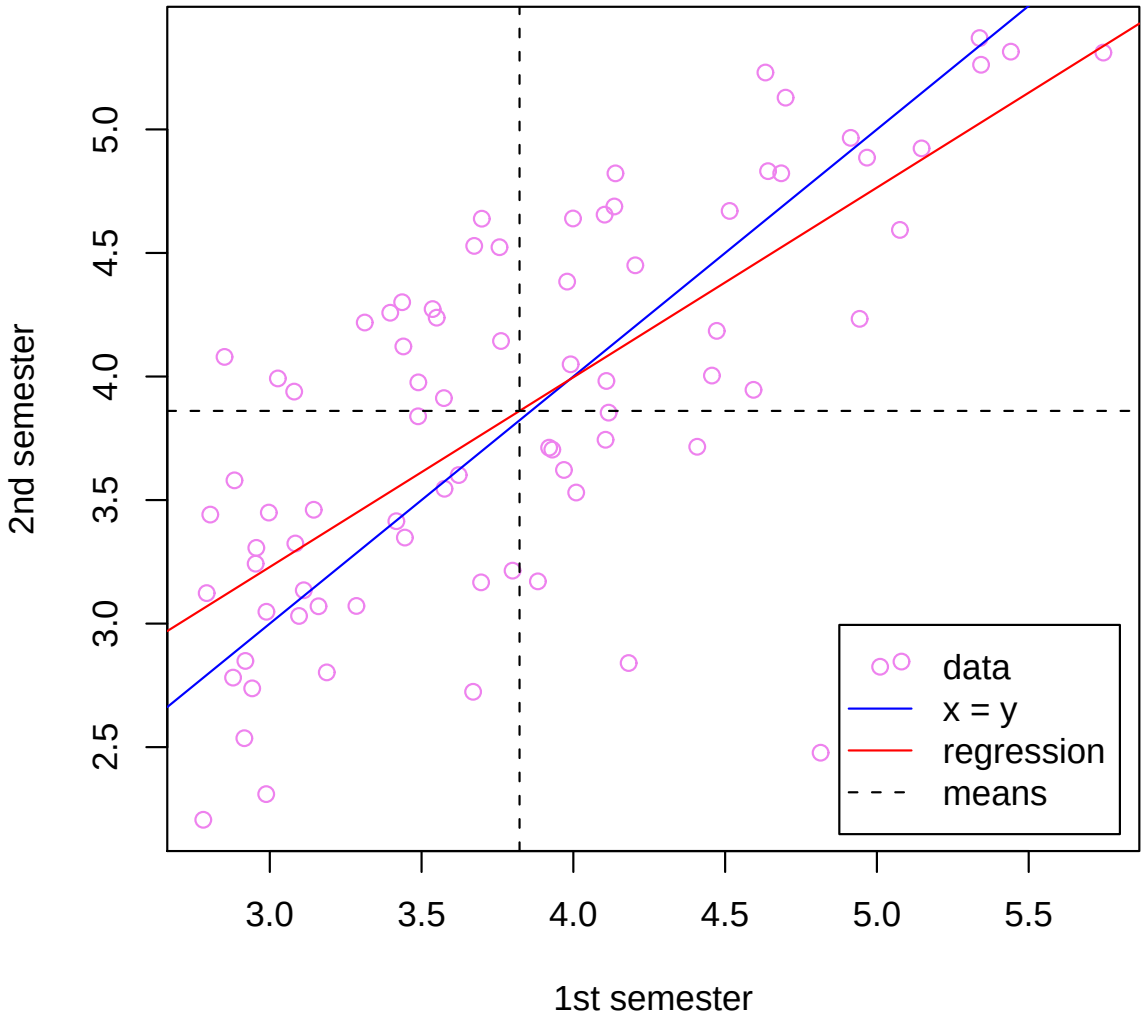
928 наблюдений $y = 0.65x + 24 = 68.2 + 0.65 \times (x - 68.2)$

x = средняя оценка по мат. анализу и алгебре в 1-м семестре

y = средняя оценка по мат. анализу, алгебре и программированию во 2-м семестре



79 студентов



$y = 0.93 + 0.77 \times x \approx 3.86 + 0.77 \times (x - 3.82)$

3.82 — средняя оценка по всем студентам в 1-м семестре

3.86 — средняя оценка по всем студентам во 2-м семестре

Пример 4. Оценка стоимости дома

Предположим, что имеются данные о жилых загородных домах в некоторой местности.

Для каждого дома известна его цена, состояние, жилая площадь, количество этажей, количество комнат, время постройки, удаленность до основных магистралей, наличие инфраструктуры, экологическая обстановка в районе и т. п.

Требуется научиться оценить цену по остальной информации.

Объектами являются дома, входами — их характеристики, а выходом — цена дома.

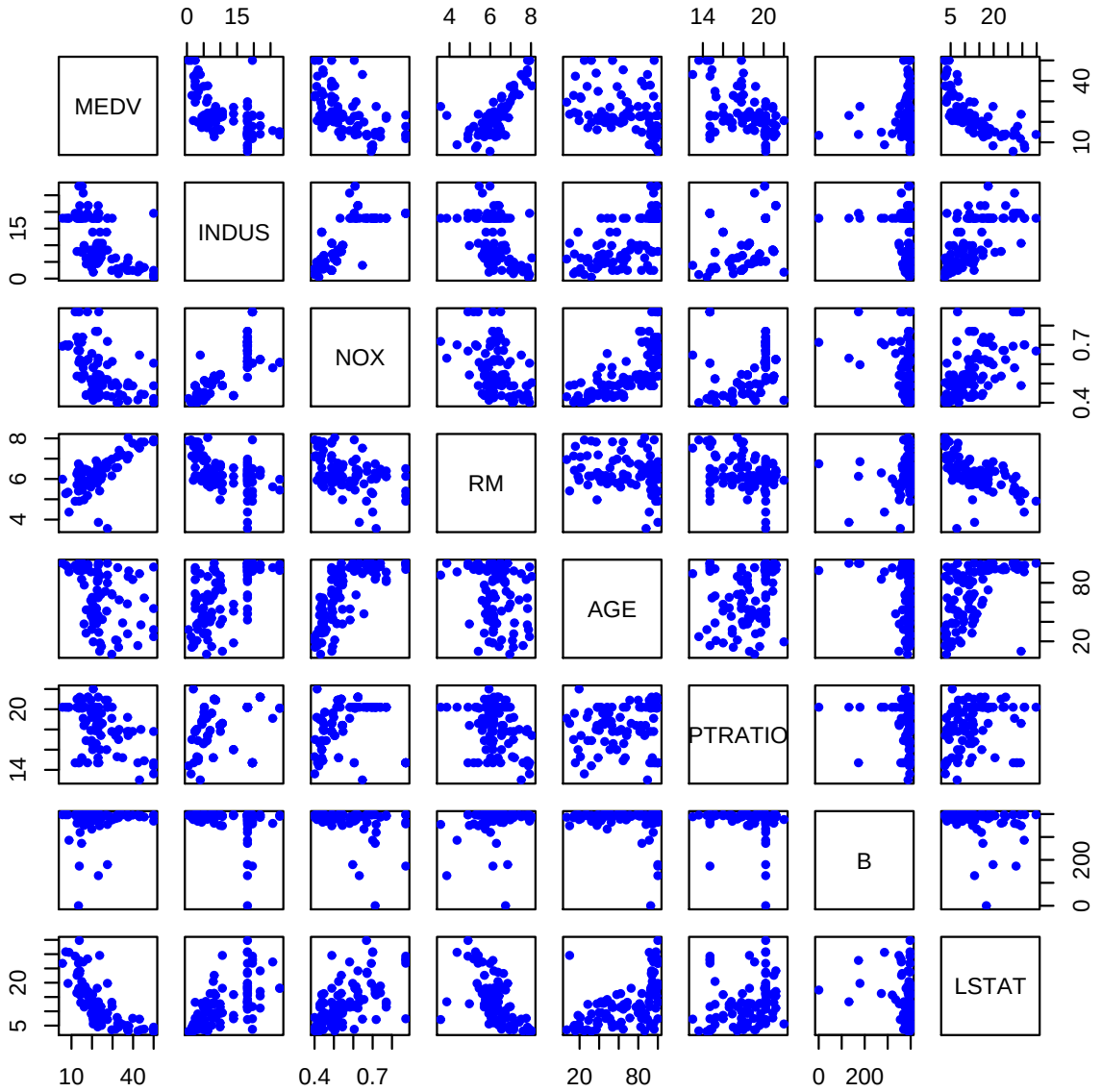
Это задача восстановления регрессии.

Boston Housing Data http://archive.ics.uci.edu/ml/datasets/Housing

Информация агрегирована: территория поделена на участки и дома, стоящие на одном участке, собраны в группы. Нужно оценить среднюю цену дома. Таким образом, объектами являются сами эти группы. Их общее количество — 506.
Признаки

1. CRIM — уровень преступности на душу населения,
2. ZN — процент земли, застроенной жилыми домами (только для участков площадью свыше 25000 кв. футов),
3. INDUS — процент деловой застройки,
4. CHAS — 1, если участок граничит с рекой; 0 в противном случае (бинарный признак),
5. NOX — концентрация оксида азота, деленная на 10^7,
6. RM — среднее число комнат (по всем домам рассматриваемого участка),
7. AGE — процент домов, построенных до 1940 г. и занимаемых владельцами,
8. DIS — взвешенное расстояние до 5 деловых центров Бостона,
9. RAD — индекс удаленности до радиальных магистралей,
10. TAX — величина налога в $10000,
11. PTRATIO — количество учащихся, приходящихся на одного учителя (по городу),
12. B = $1000(\mathtt{AA} - 0.63)^2$, где AA — доля афро-американцев,
13. LSTAT — процент жителей с низким социальным статусом.

Диаграммы рассеяния для каждой пары переменных MEDV, INDUS, NOX, RM, AGE, PTRATIO, B. Значение переменной MEDV нужно научиться предсказывать по значениям остальных переменных. Изображены только по 100 случайных точек.



2.2. Некоторые методы обучения с учителем

- Линейный метод наименьших квадратов
- Линейный и квадратичный дискриминантный анализ
- Логистическая регрессия
- Метод k ближайших соседей
- Наивный байесовский классификатор
- Деревья решений (C4.5, CART и др.)
- Персептрон и нейронные сети
- Машина опорных векторов (SVM)
- Ансамбли решающих правил (бустинг, баггинг и т. п.)
- . . .

См., например,

Top 10 algorithms in data mining // Knowl. Inf. Syst. 2008. № 14. P. 1–37

2.3. Метод ближайшего соседа

kNN — k nearest neighbours

Пусть $N_k(x)$ — множество k ближайших (относительно некоторой выборки в пространстве признаков) к x объектов из обучающей выборки.

Пусть $I_k(x, y)$ — множество тех объектов $x^{(i)}$ из $N_k(x)$, для которых $y_i = y$.

- Для задачи восстановления регрессии:

$$f(x) = \frac{1}{k} \sum_{x^{(i)} \in N_k(x)} y_i.$$

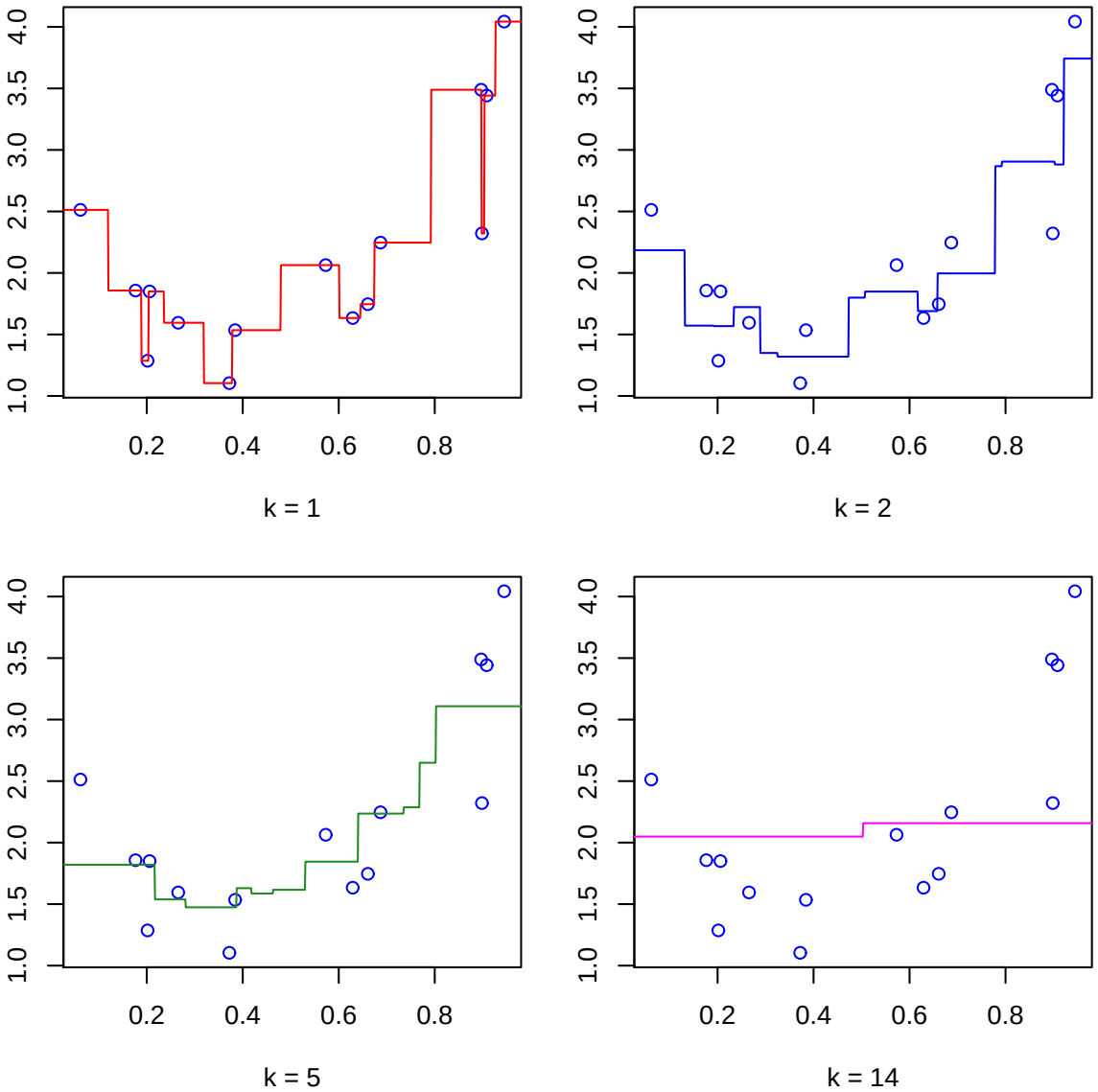
- Для задачи классификации:

$$f(x) = \operatorname*{argmax}_y |I_k(x, y)|.$$

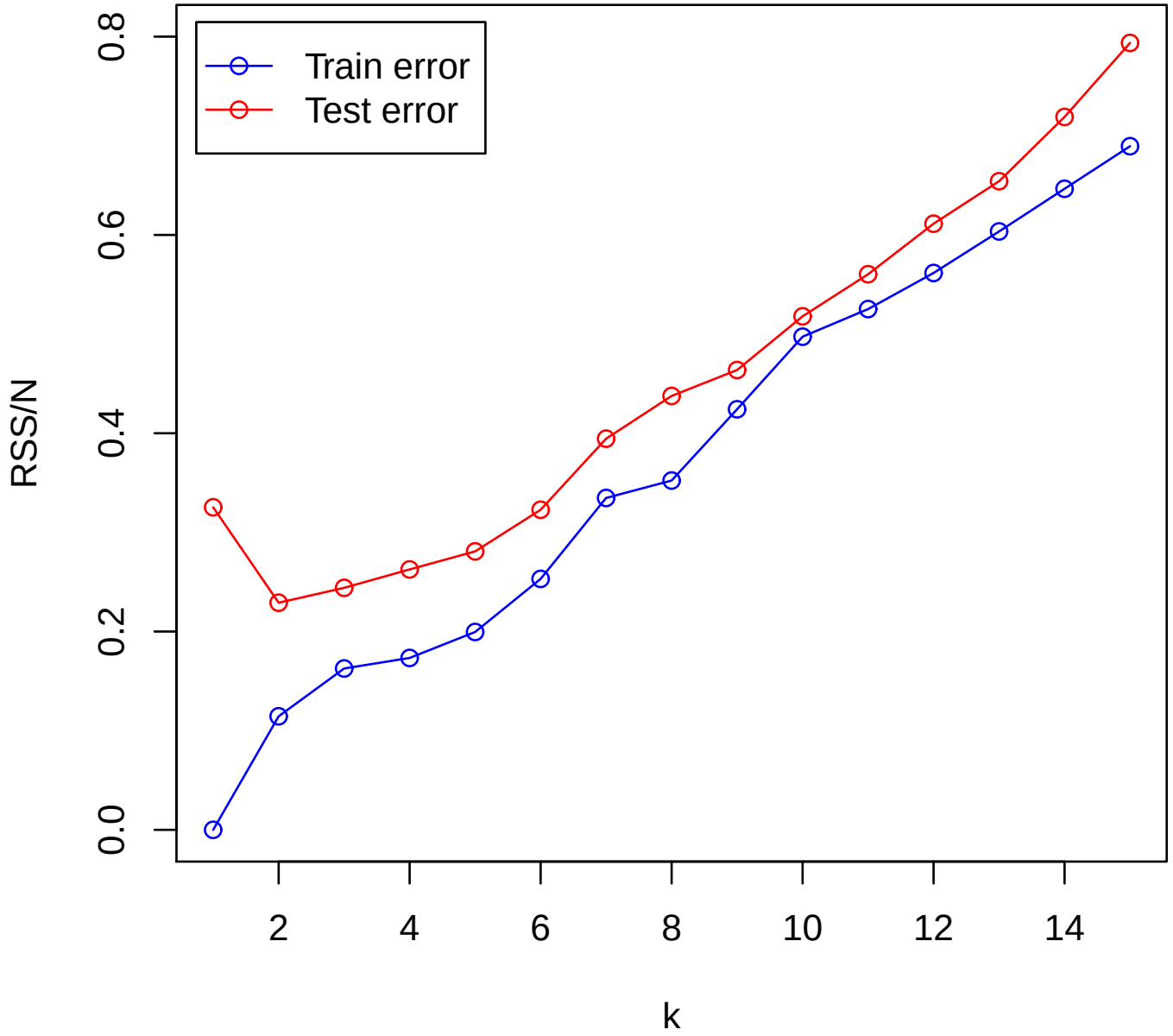
Пример. Задача восстановления регрессии

Метод k ближайших соседей для задачи восстановления регрессии

$y = 8x^2 - 6.4x + 2.5 + N(0, 0.4)$



Средняя квадратическая ошибка



Пример. Задача классификации

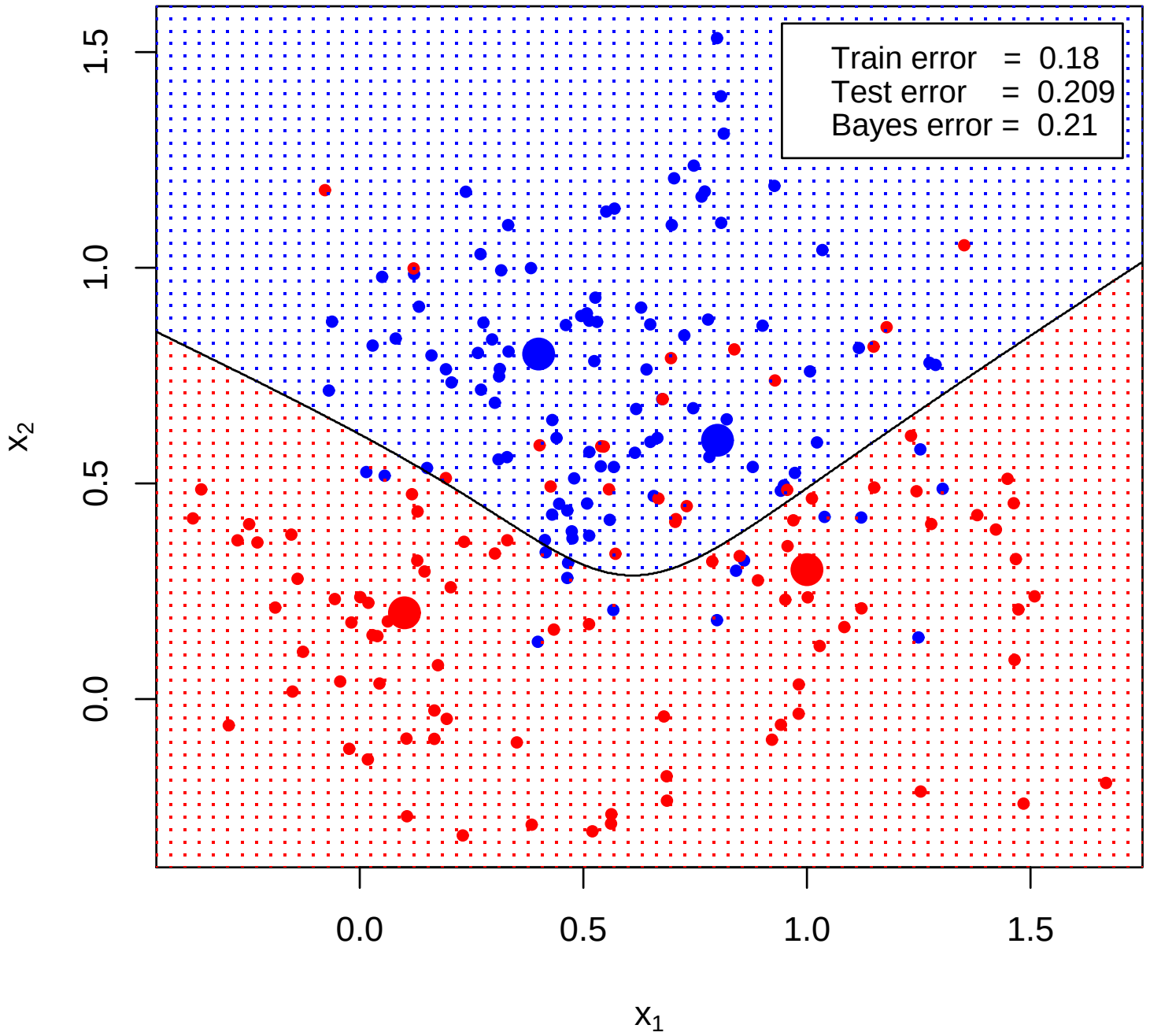
Рассмотрим обучающую выборку для задачи классификации на два класса.

Каждый класс — 100 прецедентов.

Распределение в каждом классе — смесь (взвешенная сумма) 2-х нормальных распределений (гауссианов).

$\mu_1 = (0.4, 0.8)$, $\mu_2 = (0.8, 0.6)$, $\mu_3 = (1.0, 0.3)$, $\mu_4 = (0.1, 0.2)$.

Матрица ковариации $\sigma^2\mathbf{I}$, где $\sigma = 0.3$.

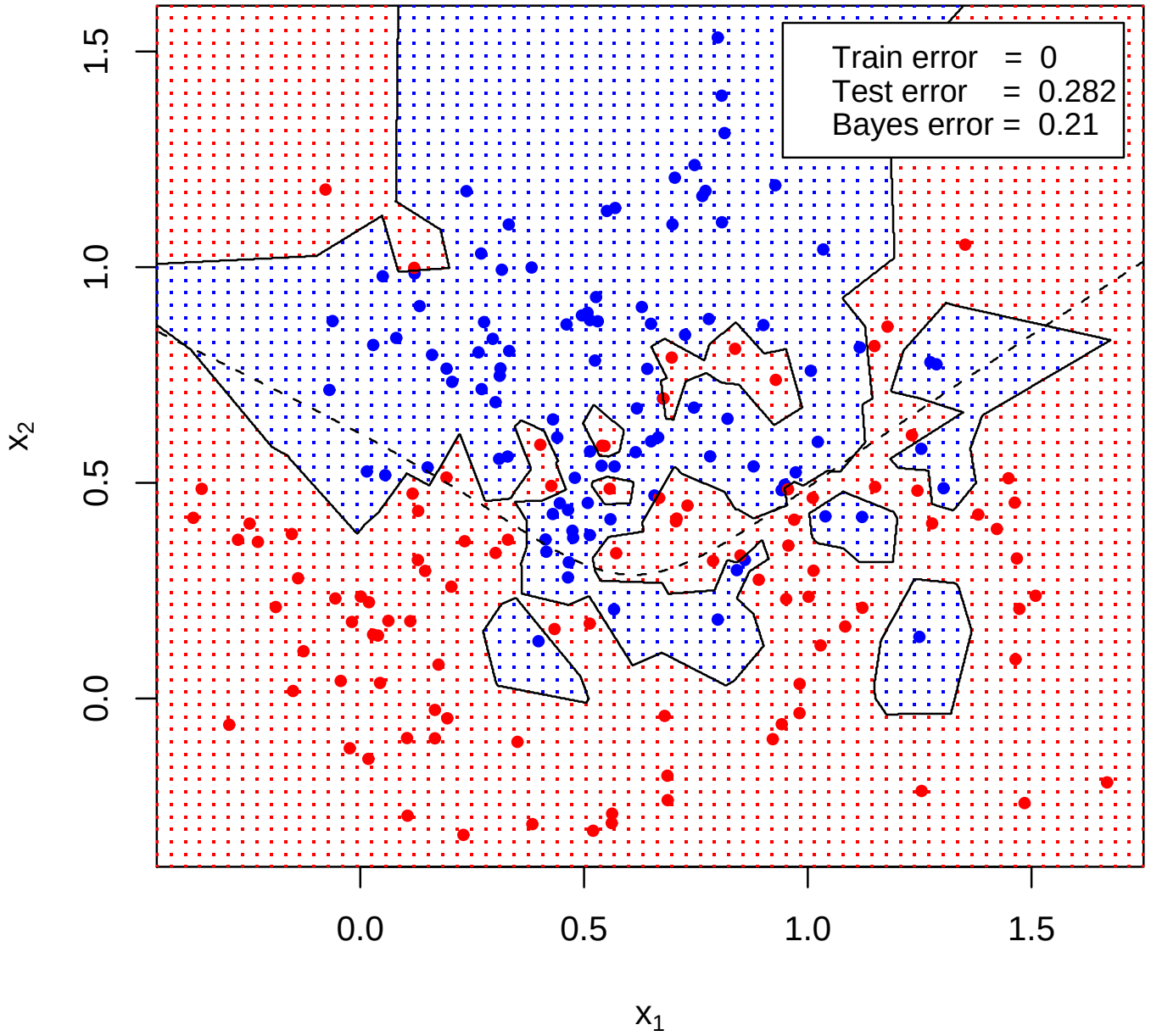

Train error = 0.18
Test error = 0.209
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

Ошибка на заданной выборке: 0.2

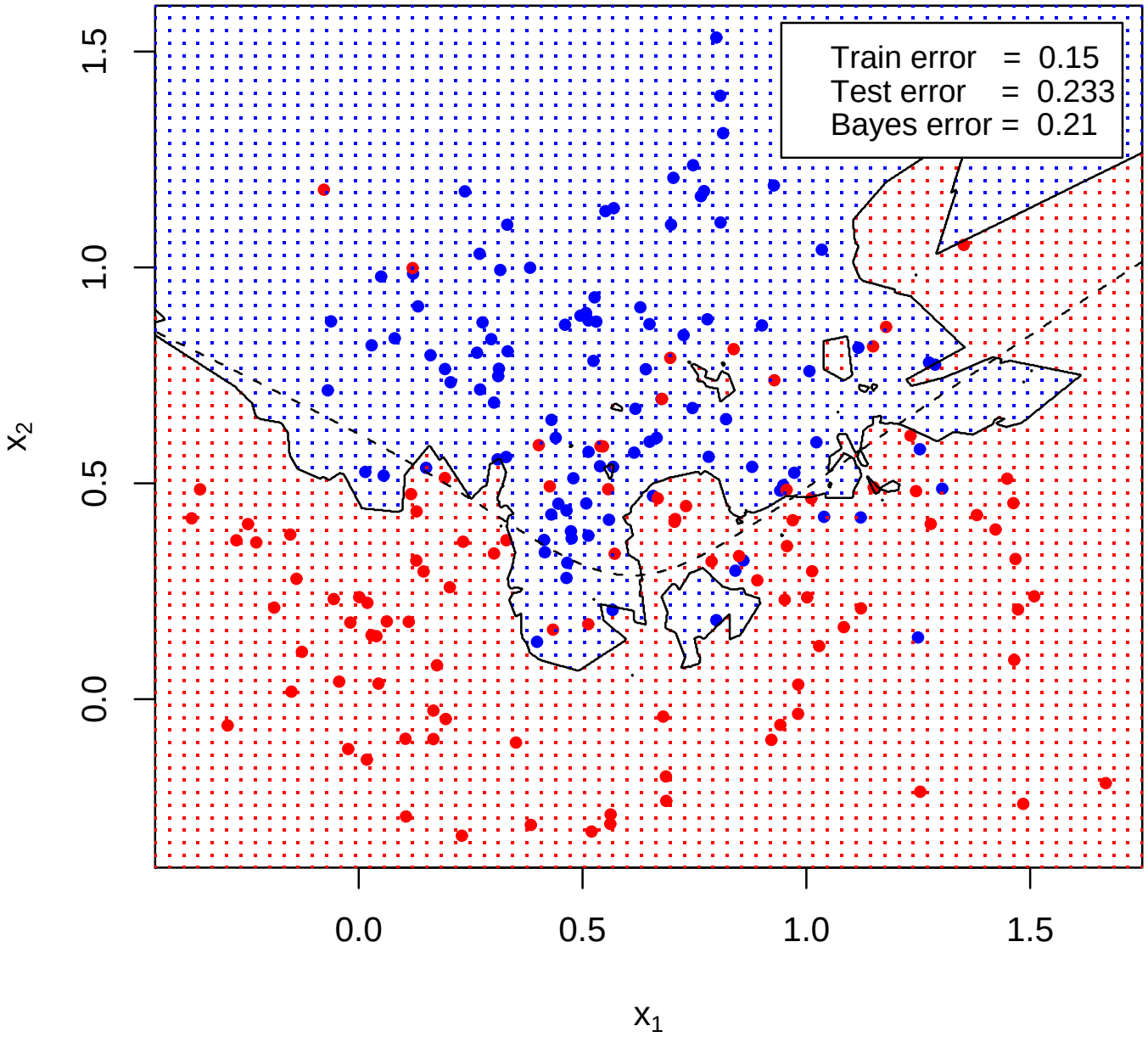
Распределение известно, поэтому можем определить точно:

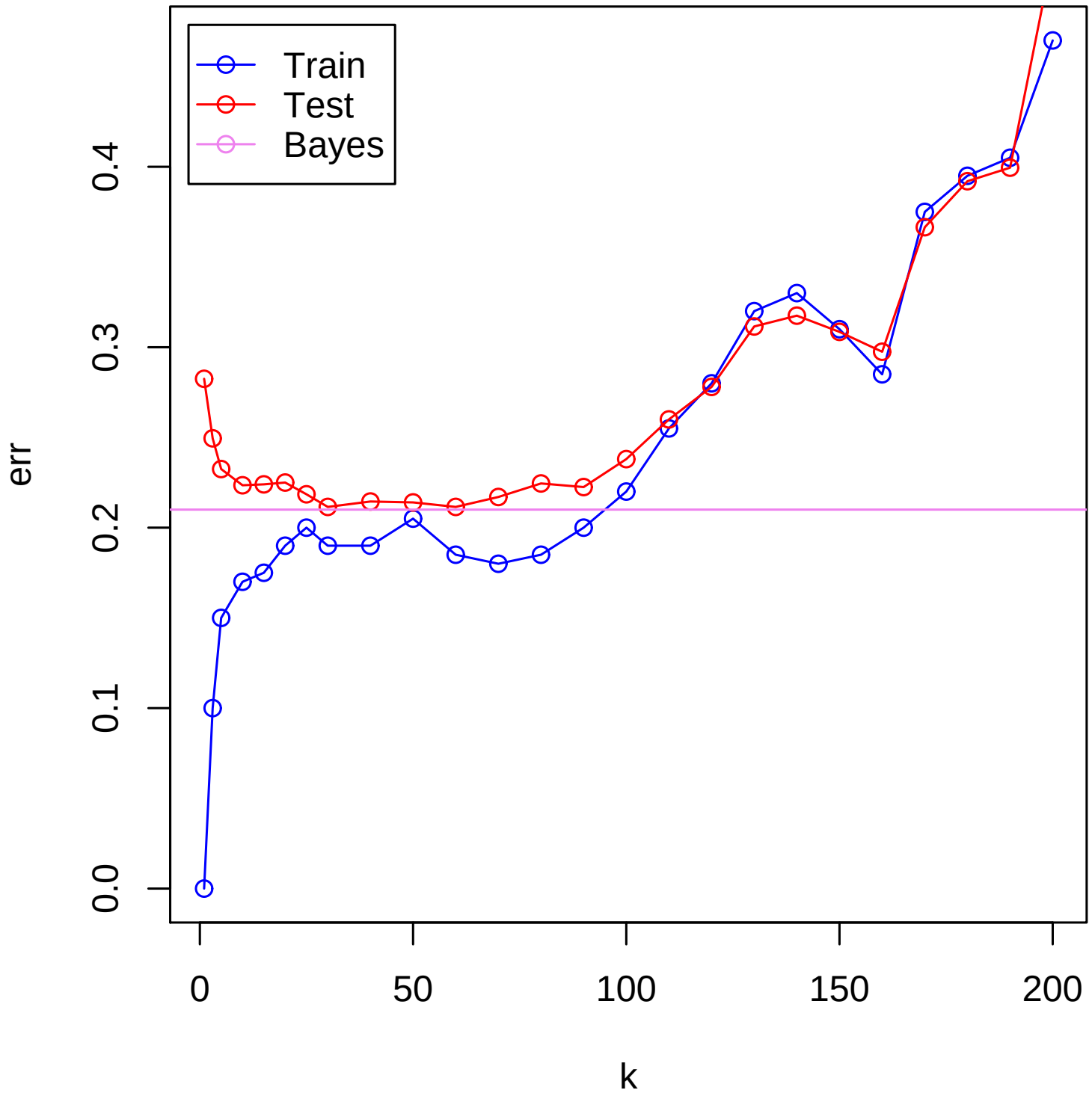
Средний риск R (байесова ошибка): 0.21

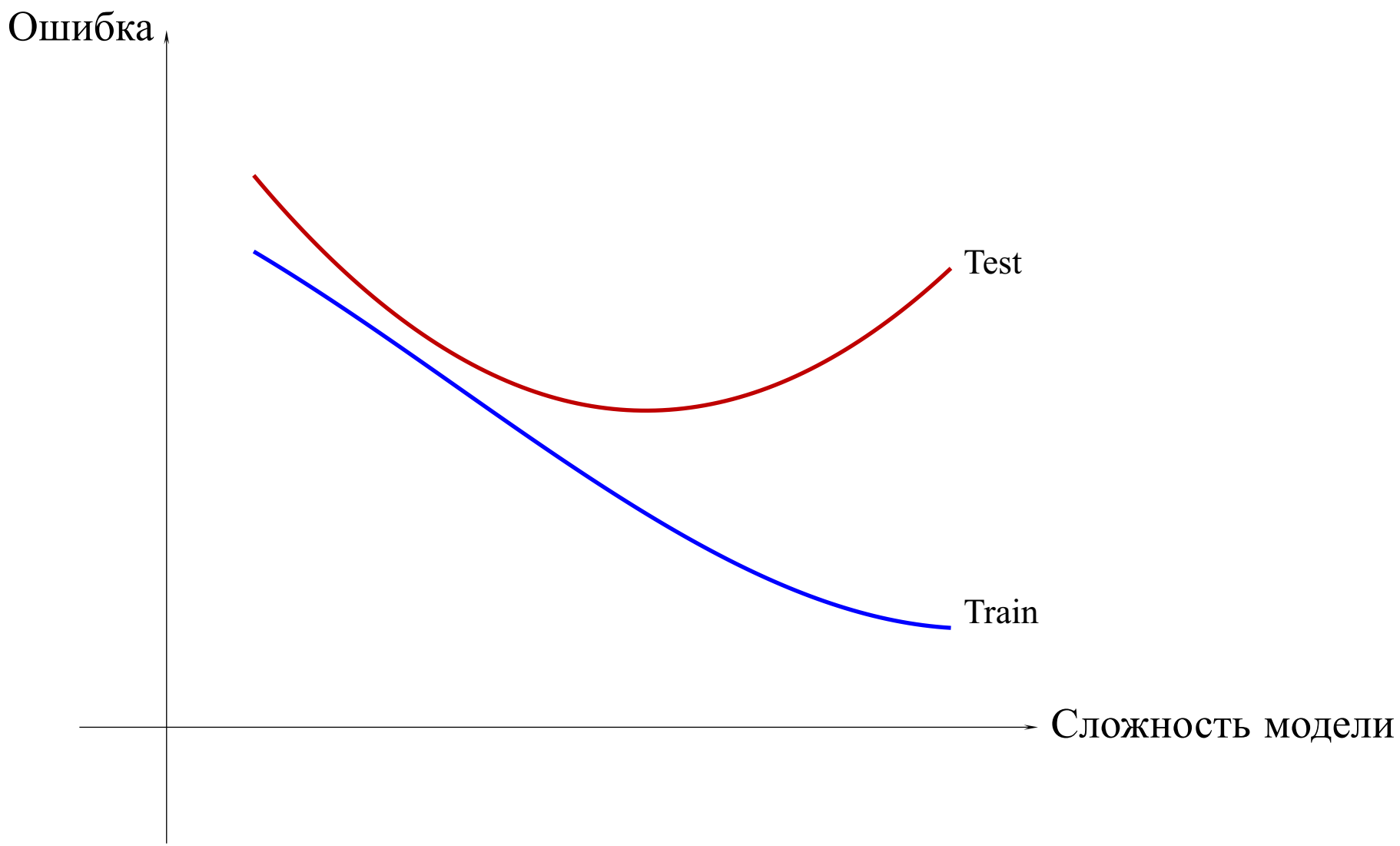
Метод ближайшего соседа



Метод 5 ближайших соседей




Train
Test
Bayes
err
0.0
0.1
0.2
0.3
0.4
0
50
100
150
200
k

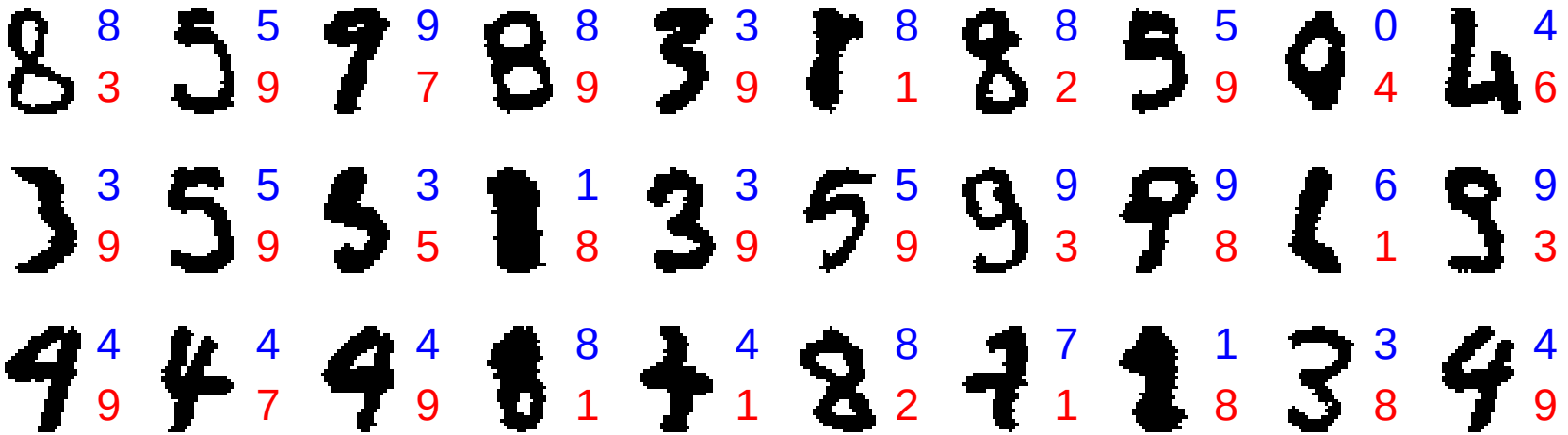

Ошибка
Test
Train
Сложность модели

Задача классификации рукописных цифр

Выборка размера 1934 была случайным образом разбита на две группы: обучающую и тестовую — по 967 объектов в каждой.

k	*Ошибка*	
	на обучающей выборке	*на тестовой выборке*
1	0	0.031
2	0.017	0.046
3	0.014	0.031
5	0.026	0.033
10	0.034	0.038
15	0.042	0.046
20	0.051	0.050

Все случаи неправильной классификации цифр из тестовой выборки в случае $k = 1$.
Красная цифра — ответ классификатора, синяя — верный ответ.

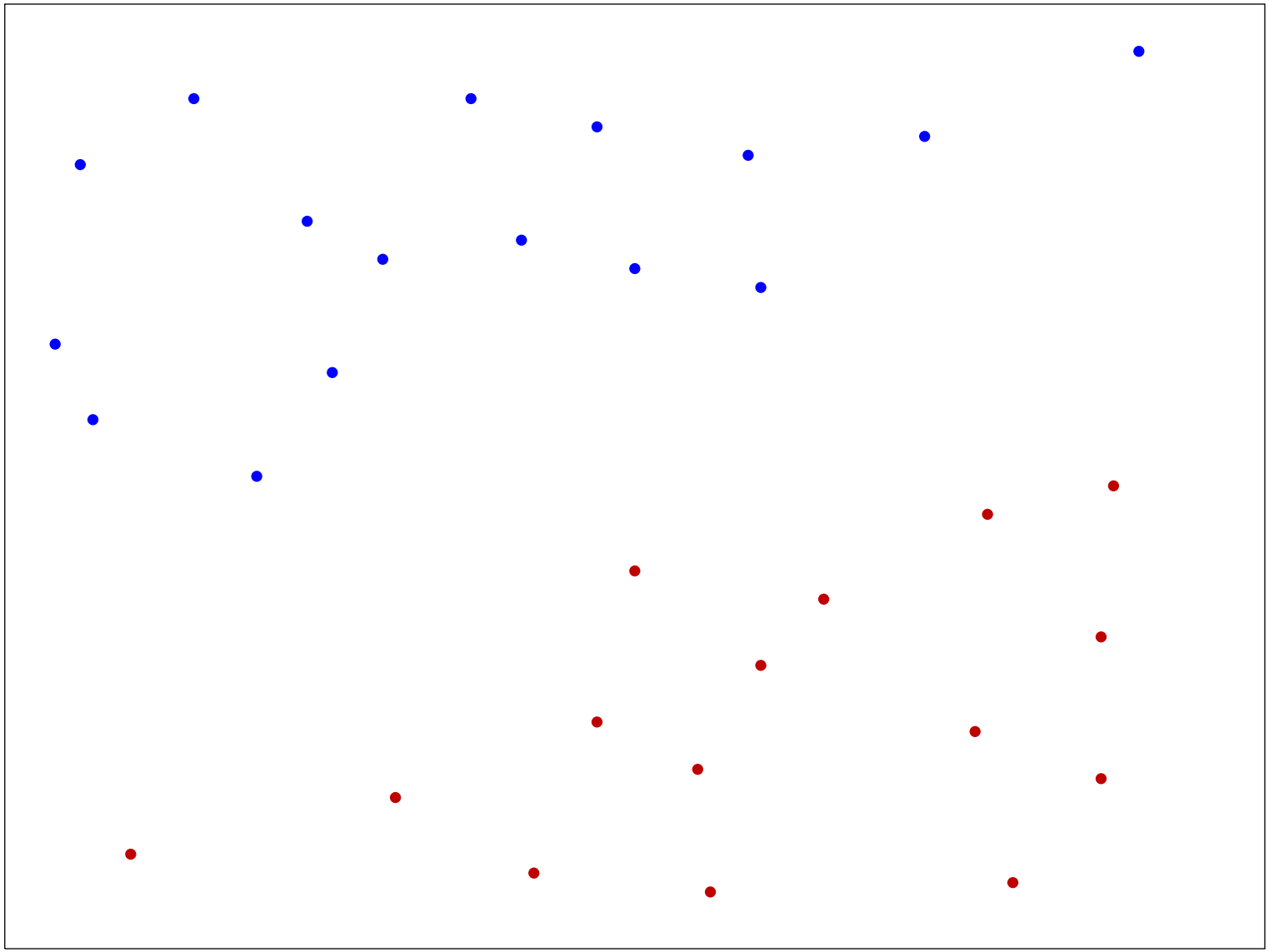


2.4. Машина опорных векторов

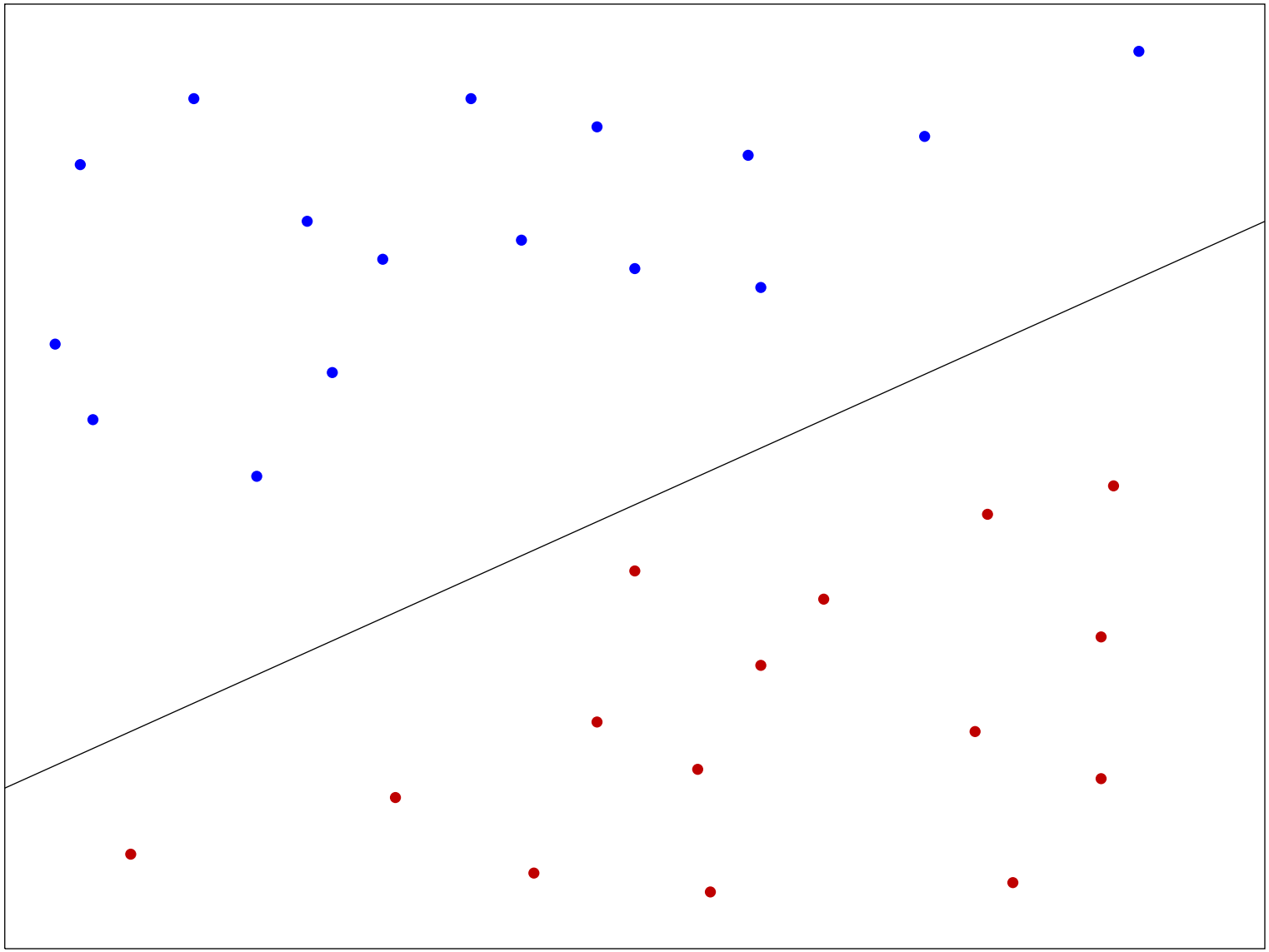
Support Vector Machine (SVM)

Машина опорных векторов (support vector machine) — один из методов построения решающего правила.

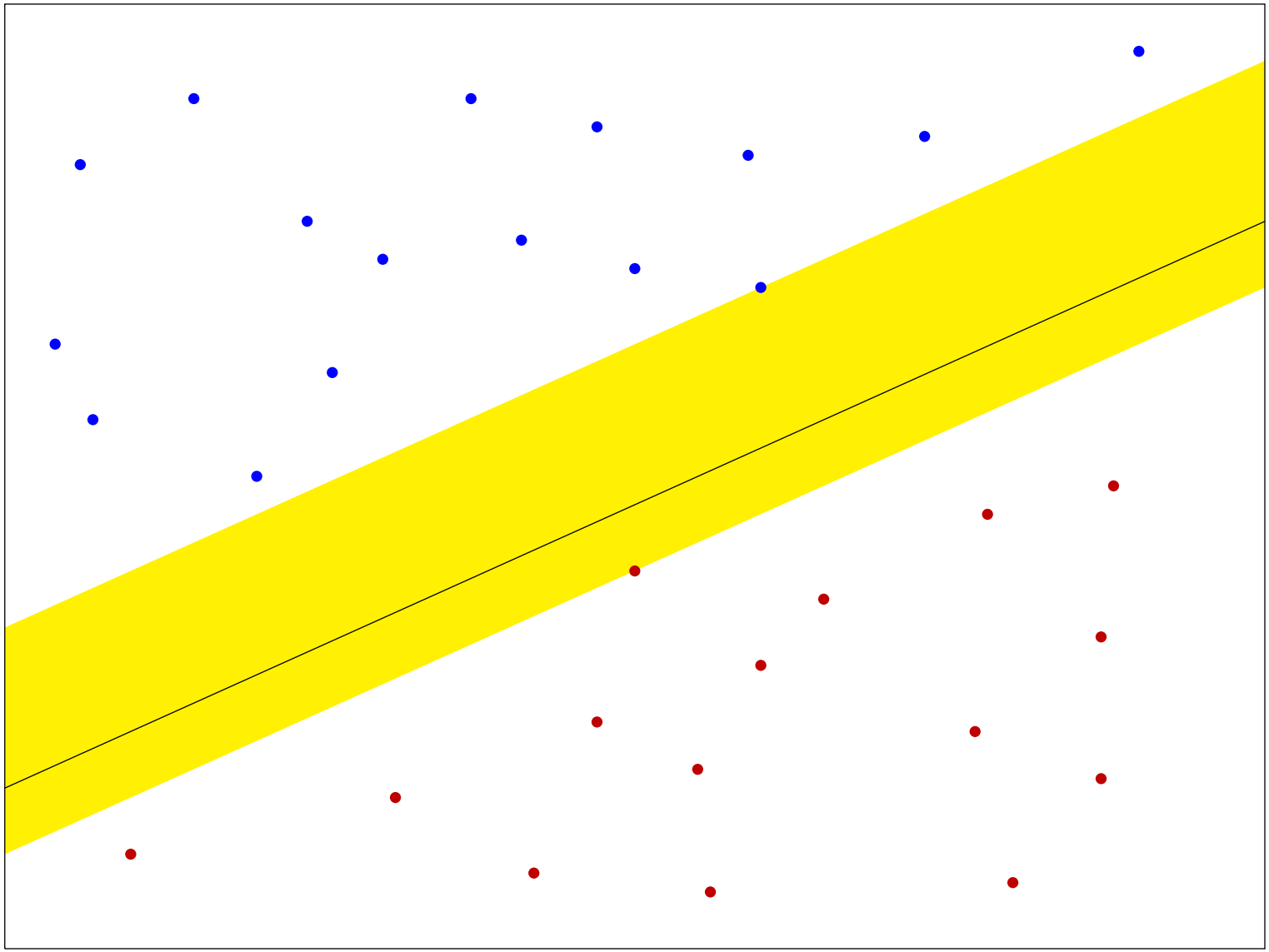
- Метод обобщенного портрета (оптимальная разделяющая гиперплоскость) — 60-70 гг. В. Н. Вапник и др., см. В. Н. Вапник, А. Я. Червоненкис «Теория распознавания образов». М.: Наука, 1974
- Добавлены ядра — [Cortes, Vapnik, 1995]



Два класса



Разделяющая гиперплоскость



Желтая — разделяющая («нейтральная») полоса

Оптимальная разделяющая гиперплоскость — это гиперплоскость, разделяющая объекты двух классов, такая, что расстояние от нее до ближайшей точки из обучающей выборки (не важно из какого класса) максимально.

Т. е. оптимальная разделяющая гиперплоскость лежит в центре разделяющей полосы и толщина этой полосы максимальна.

Она максимизирует *зазор* (*margin*) между плоскостью и данными из обучающей выборки — это приводит, как правило, к хорошим результатам и на тестовых данных.

Обучающая выборка:

$$(x^{(1)}, y_1), (x^{(2)}, y_2), \ldots, (x^{(N)}, y_N),$$

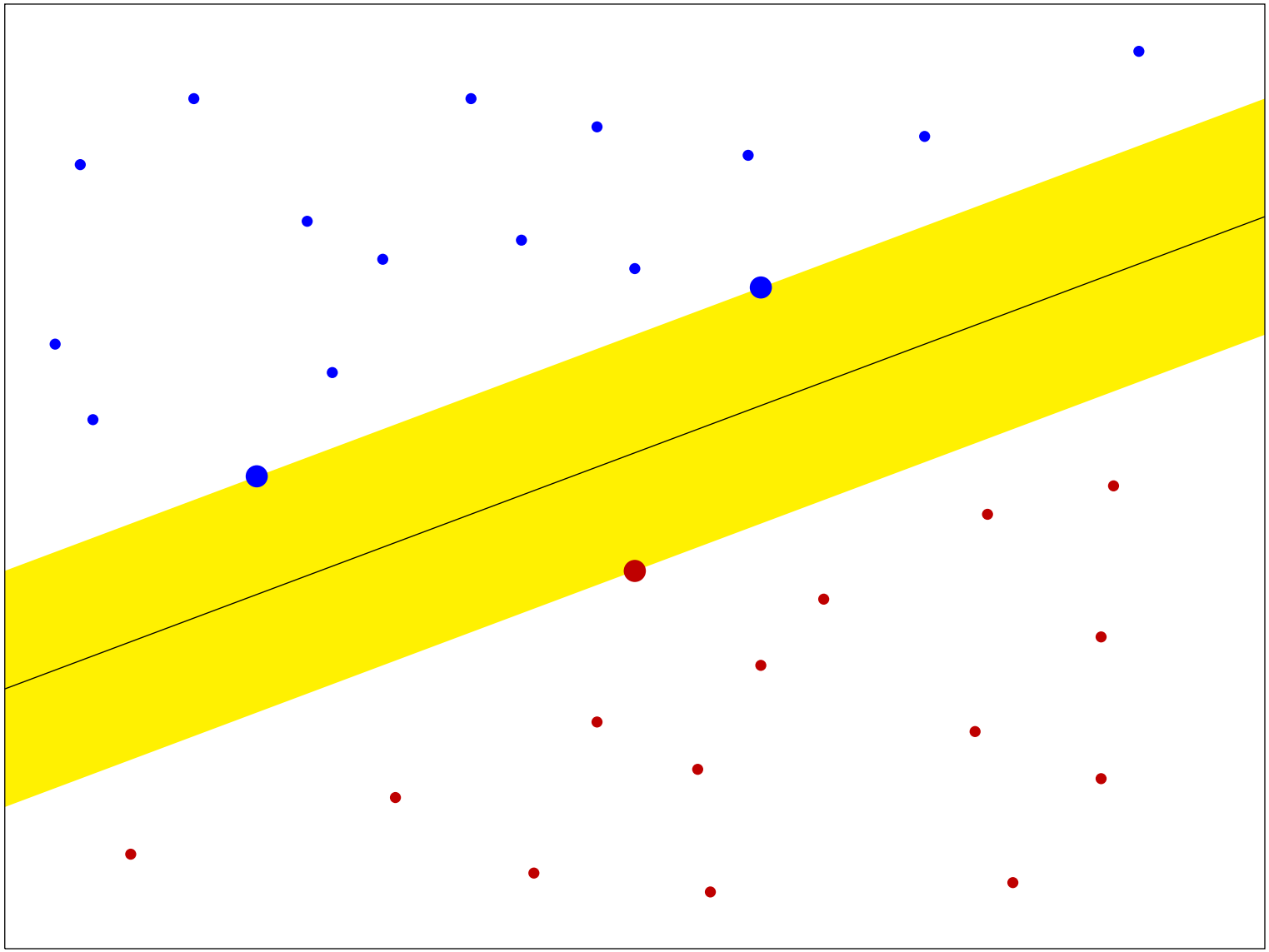
$y_i \in \{-1, 1\}$

Задача нахождения оптимальной гиперплоскости $x\beta + \beta_0 = 0$:

$$\max_{\beta, \beta_0, \|\beta\|=1} C$$

при ограничениях

$$y_i(x^{(i)}\beta + \beta_0) \geq C \quad (i = 1, 2, \ldots, N)$$



А если точки нельзя разделить гиперплоскостью на два заданных класса?

$$\max_{\beta,\, \beta_0,\, \xi_i,\, \|\beta\|=1} C$$

при ограничениях

$$y_i(x^{(i)}\beta + \beta_0) \geq C(1 - \xi_i), \quad \xi_i \geq 0 \quad (i = 1, 2, \ldots, N), \quad \sum_{i=1}^{n} \xi_i \leq K,$$

где K — некоторая константа.

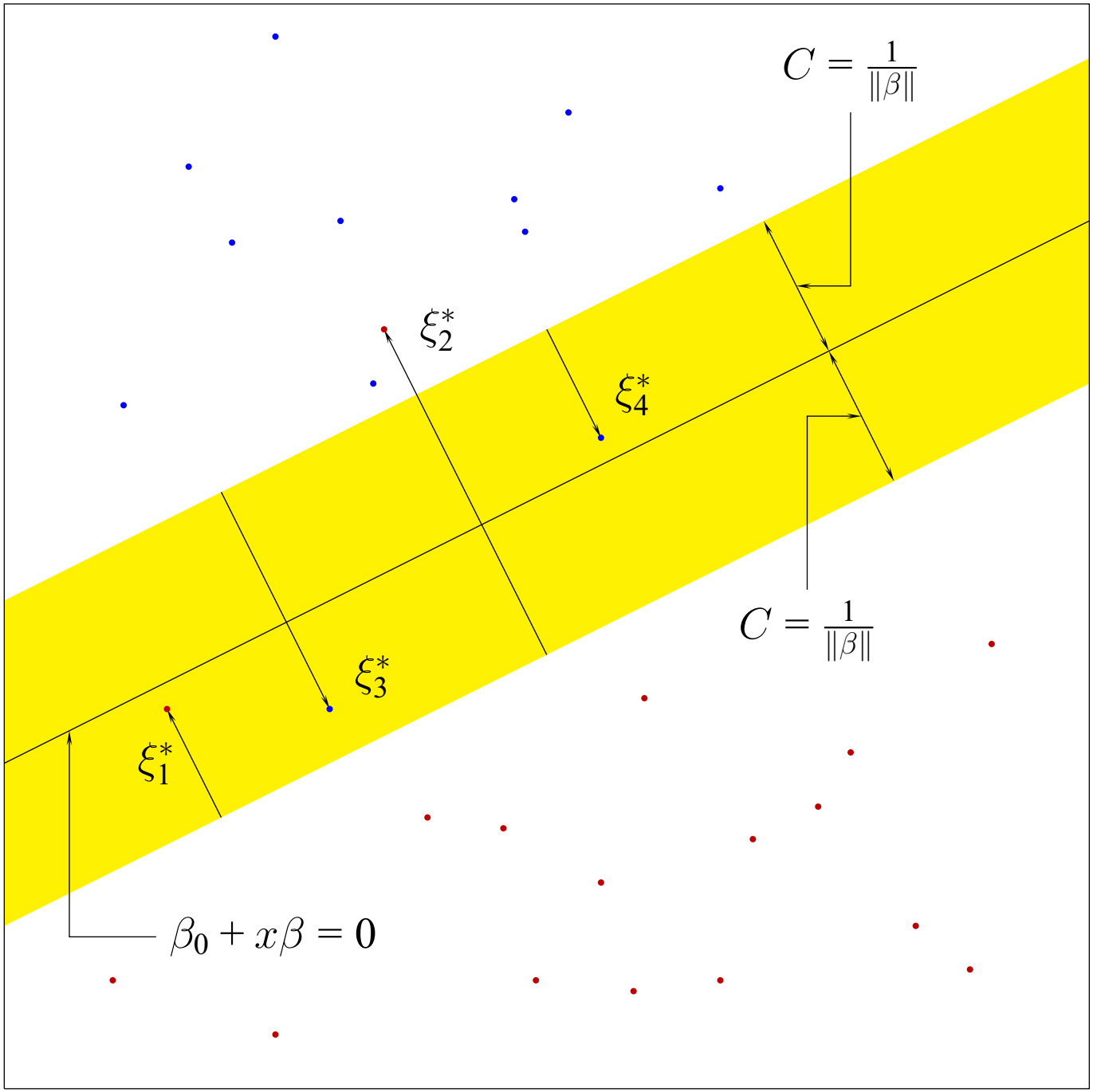
Случаю линейно-отделимых областей соответствует $K = 0$.

ξ_i пропорционально величине, на которую $x^{(i)}$ заходит за границу разделяющей полосы.

В частности, i-й объект будет классифицирован не правильно $\Leftrightarrow \xi_i > 1$.

Чем меньше K, тем меньше объектов будет классифицировано неправильно.

С другой стороны, K должно быть достаточно велико, чтобы задача была совместной.


$C = \frac{1}{\|\beta\|}$
ξ_2^*
ξ_4^*
$C = \frac{1}{\|\beta\|}$
ξ_3^*
ξ_1^*
$\beta_0 + x\beta = 0$

2.4.1. Ядра и спрямляющие пространства

Перейдем от исходного пространства $\mathscr{X}$ в другое, называемое *спрямляющее*, $\mathscr{H}$ с помощью некоторого отображения

$$h(x) = \Big(h_1(x), \ldots, h_M(x)\Big),$$

где $h_m(x)$ — базисные функции ($m = 1, 2, \ldots, M$).

Новый классификатор определяется теперь функцией

$$f(x) = \text{sign}\left(h(x)\widehat{\beta} + \widehat{\beta}_0\right).$$

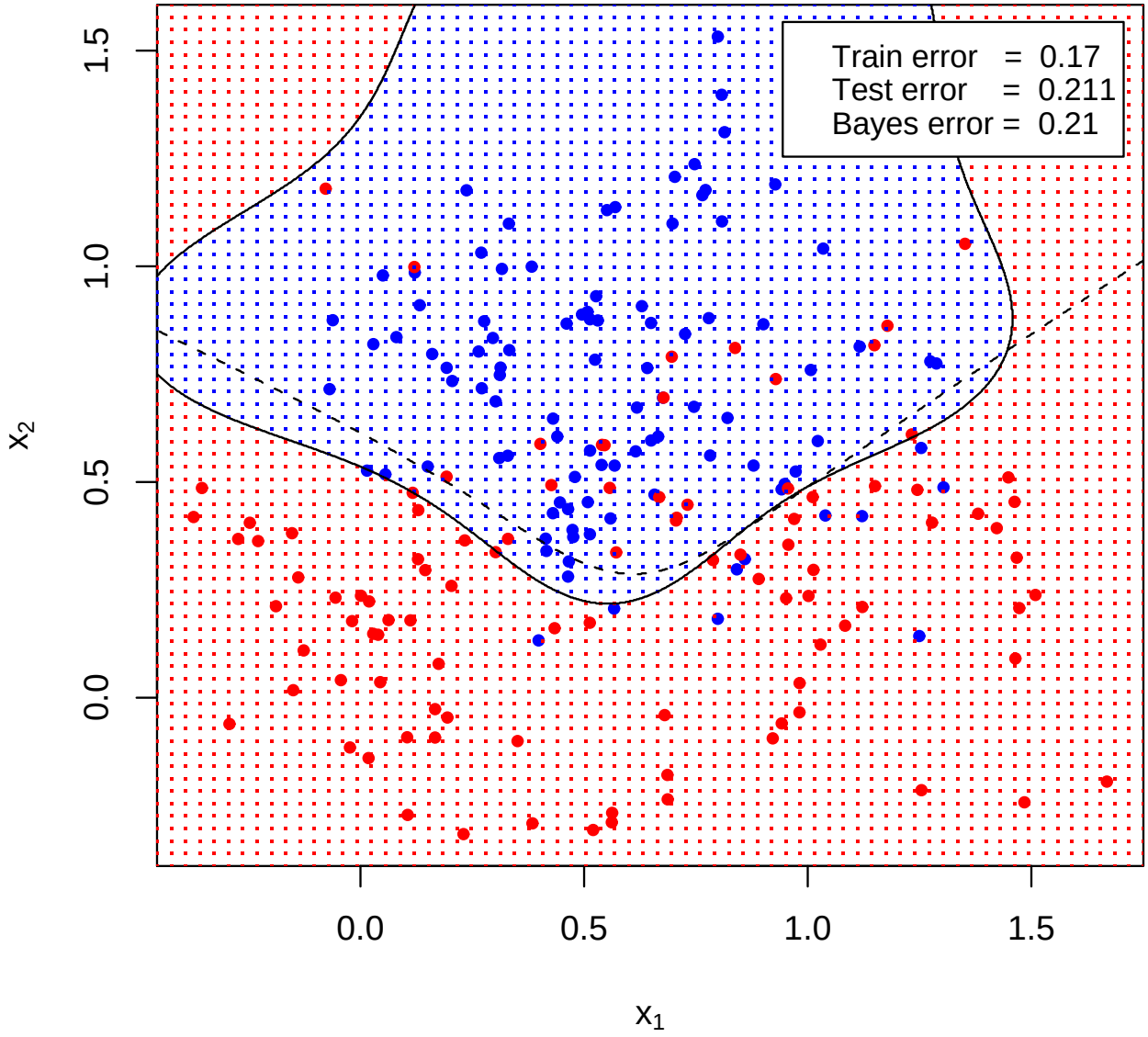
Оказывается, в расчетных формулах $h(x)$ встречается только в скалярном произведении

$$K(x, x') = \langle h(x), h(x')\rangle\,.$$

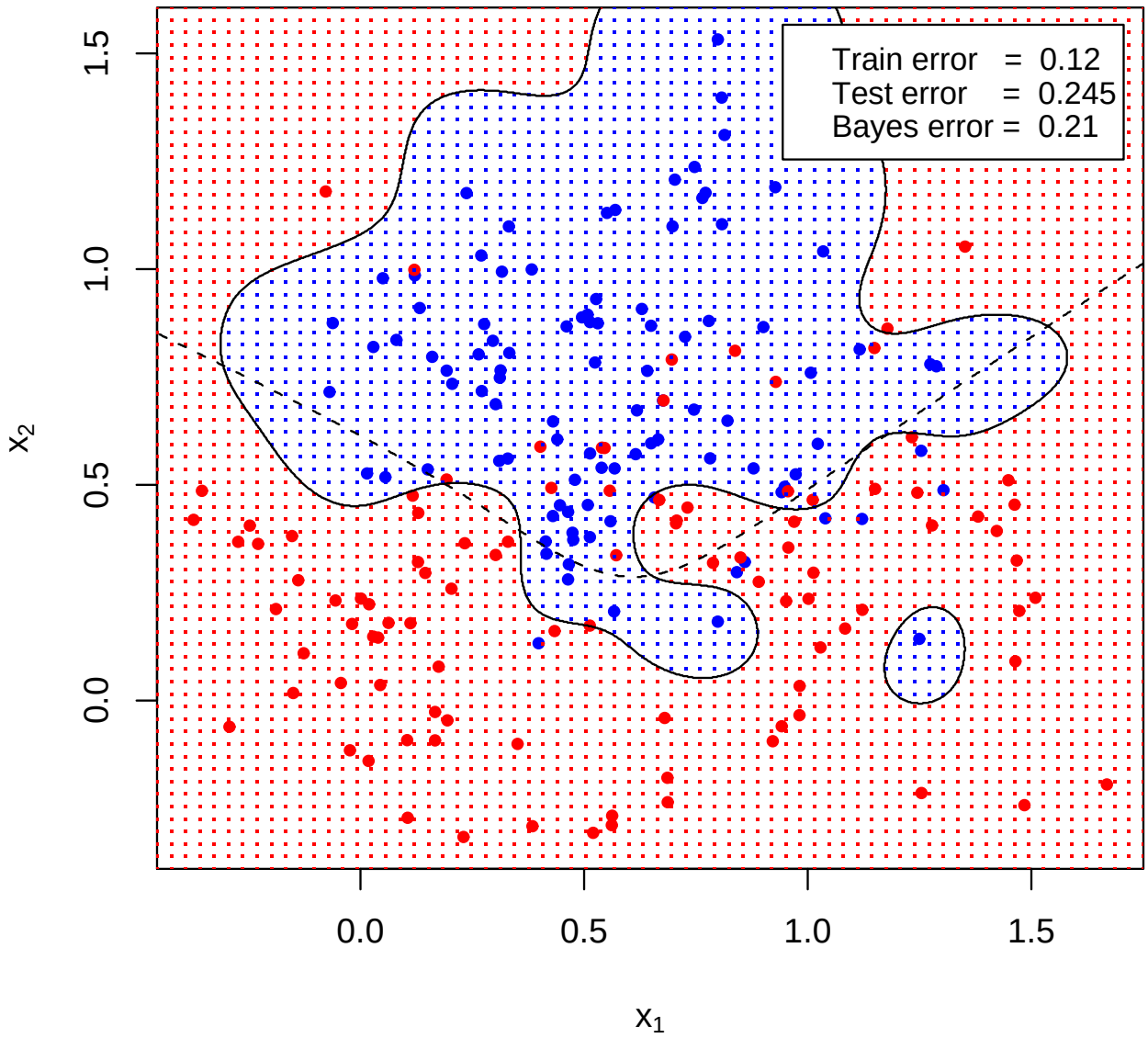
Функция $K(x, x')$ называется *ядром*.

Популярные ядра:
полином степени d: $K(x, x') = (1 + \langle x, x'\rangle)^d$,
радиальная функция: $K(x, x') = e^{-\|x-x'\|^2/c}$,
сигмоидальная («нейронная») функция: $K(x, x') = \tanh(\kappa_1 \langle x, x'\rangle + \kappa_2)$.

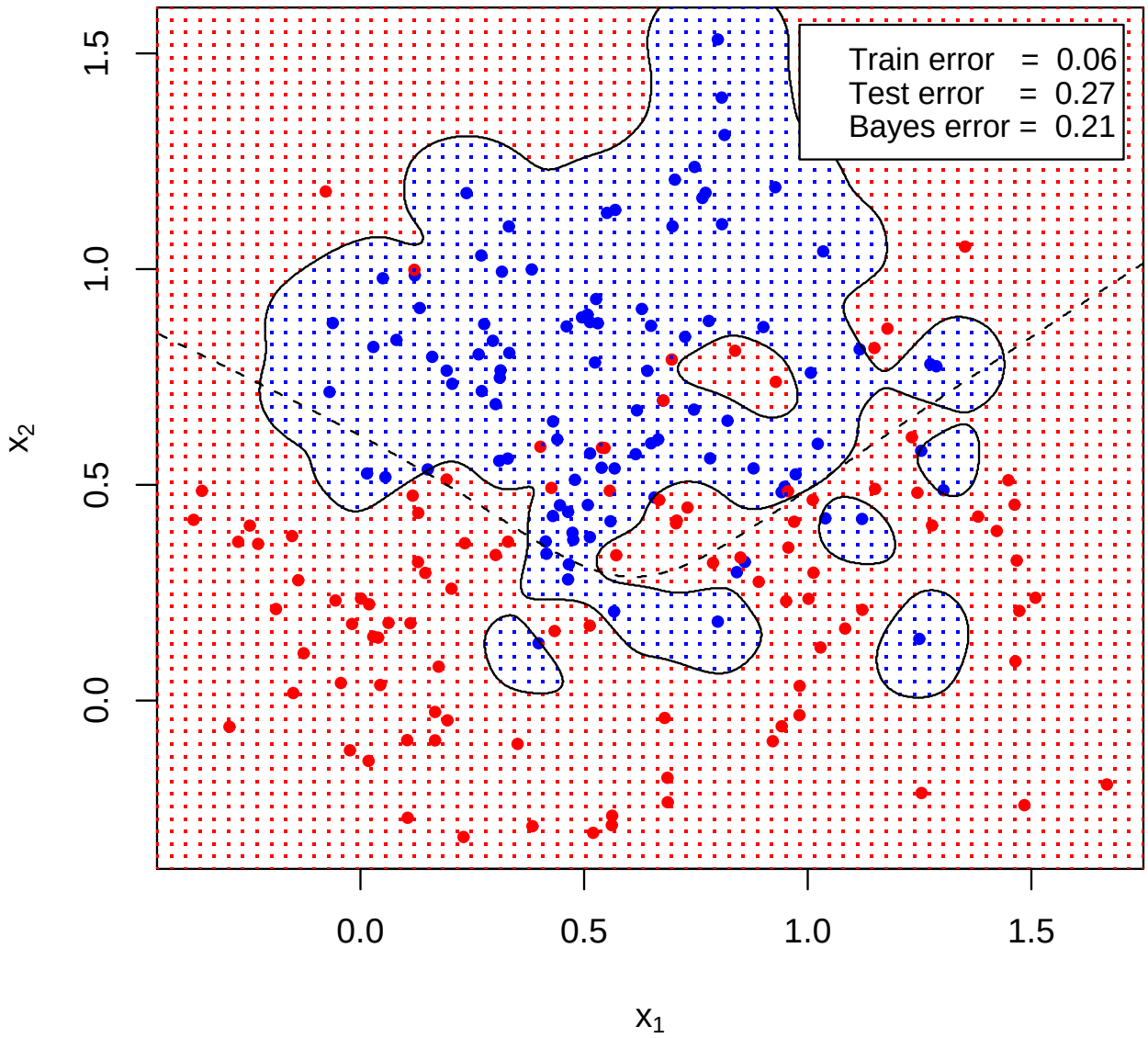
SVM радиальное ядро, $\gamma = 1/2$



SVM радиальное ядро, $\gamma = 5$



SVM радиальное ядро, $\gamma = 20$

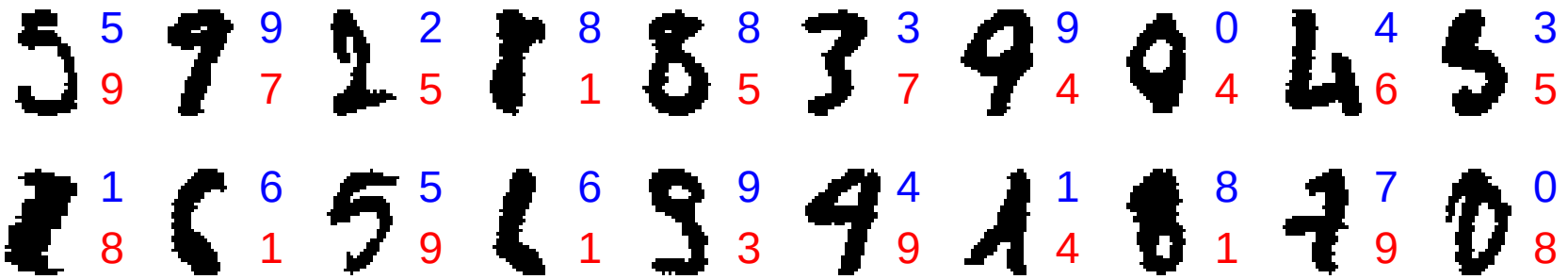


Задача классификации рукописных цифр. Выборка размера 1934 была случайным образом разбита на две группы по 967 объектов в каждой. $\gamma = 1/1024$

Ошибки на обучающей и тестовой выборках приведена в следующей таблице.

Ядро	*Ошибка*	
	на обучающей выборке	*на тестовой выборке*
Линейное	0	0.021
Радиальное	0.011	0.030
Полином 3 степени	0.090	0.094
Сигмоидальное	0.131	0.125

Все случаи неправильной классификации цифр из тестовой выборки в случае линейного ядра.



2.5. Деревья решений

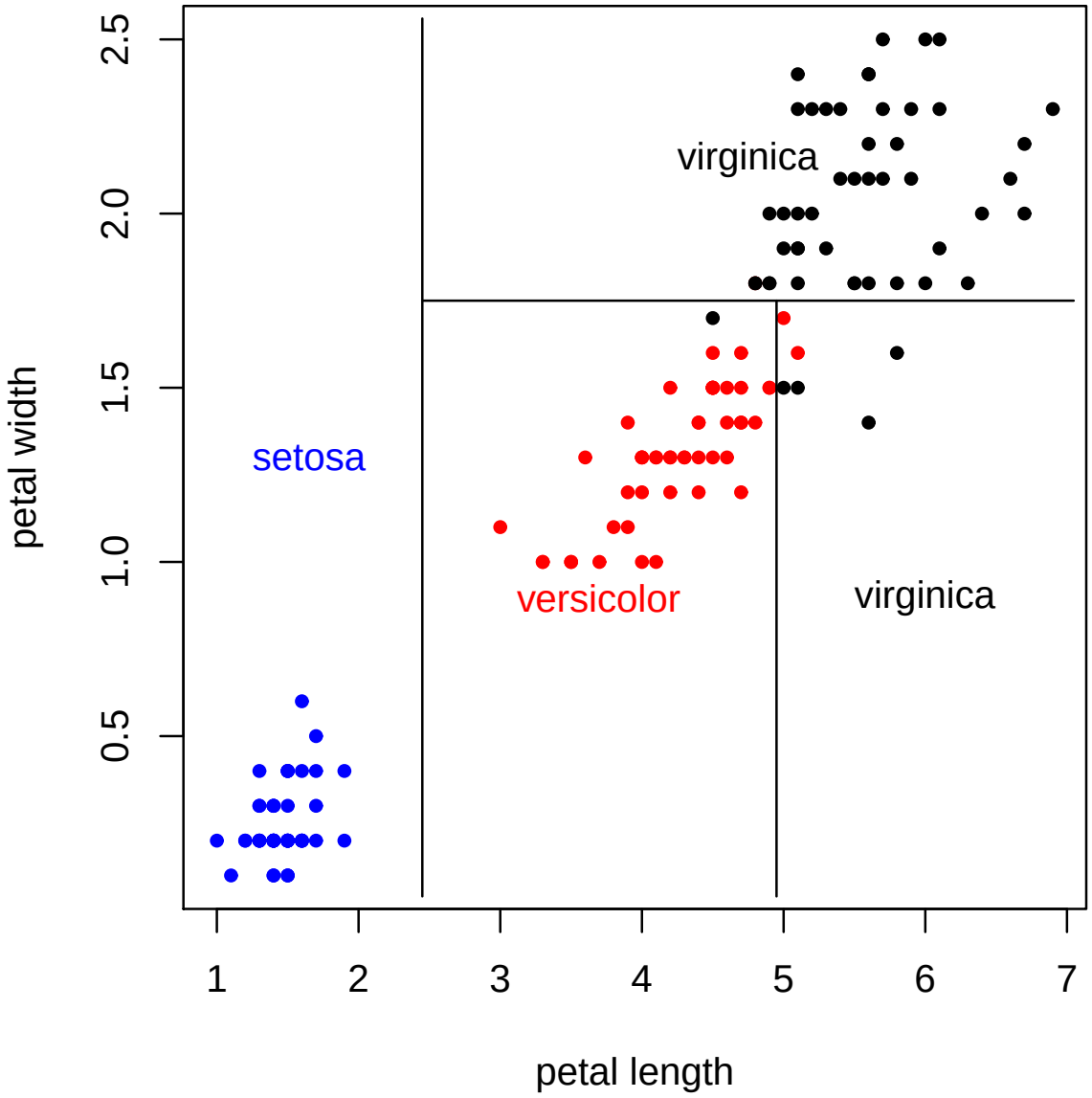
Пространство признаков разбивается на параллелепипеды со сторонами, параллельными осям координат (ящики).

В каждом ящике ответ аппроксимируется с помощью некоторой простой модели, например, константой.

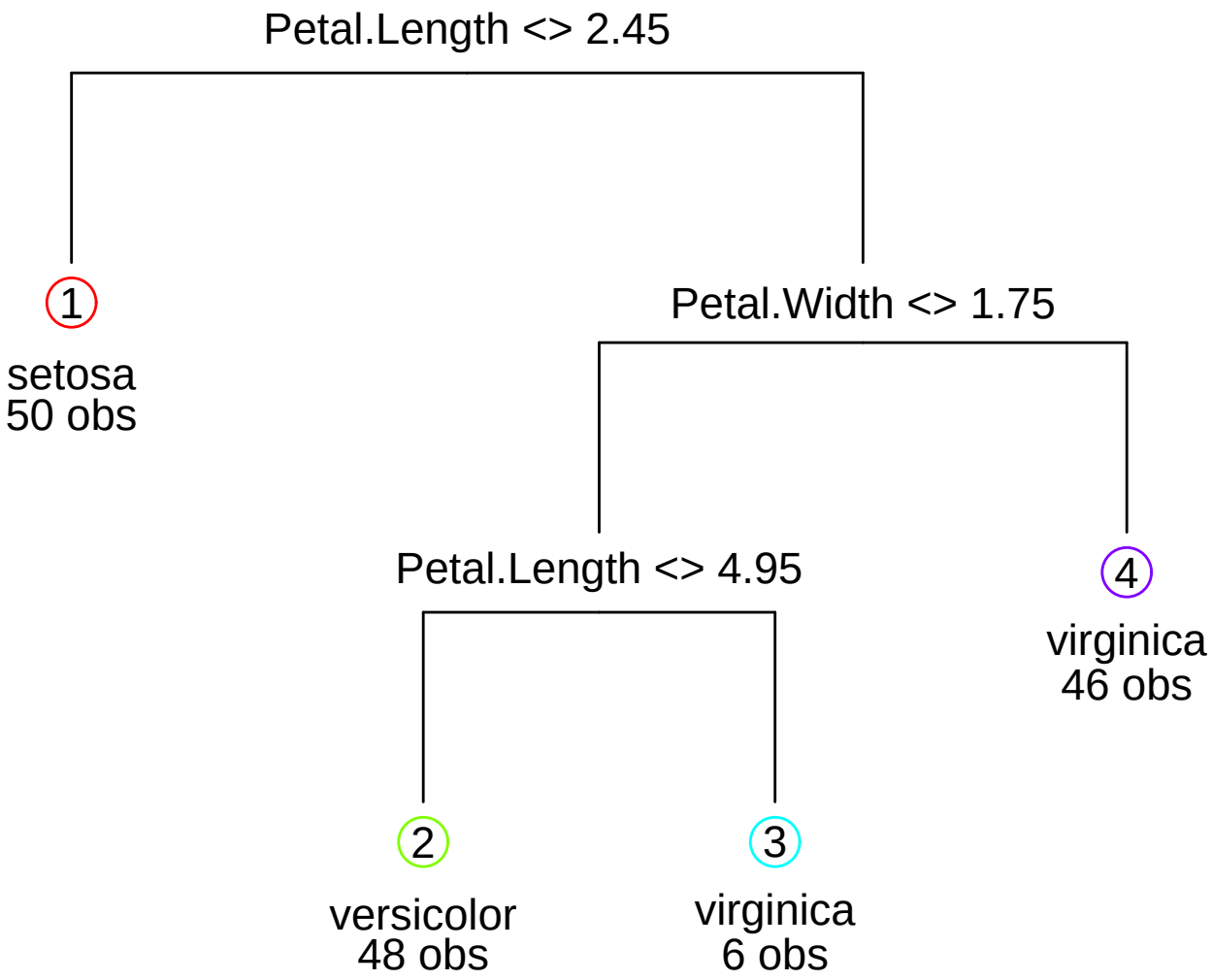
Используются только рекурсивные гильотинные разбиения.

Задача классификации цветов ириса (Fisher, 1936).

x_1, x_2 — длина и ширина чашелистика.



Дерево решений:



2.5.1. Популярные алгоритмы построения деревьев решений

- See5/C5.0 [Quinlan et., 1997] ← C4.5 [Quinlan, 1993] ← ID3 [Quinlan, 1979] ← CLS [Hunt, Marin, Stone, 1966]
- CART — Classification and Regression Trees [Breiman, Friedman, Olshen, Stone, 1984]

2.5.2. Алгоритм CART

Разбиения (splits) имеют вид:

- $x_j \leq c$ для количественных признаков;
- $x_j \in L$, где $L \subset \{1, 2, \ldots, M_j\}$ для качественных признаков.

Дерево строим рекурсивно.

Пусть на текущем шаге имеется разбиение пространства признаков на области $R_1, R_2, \ldots, R_M$.

- Выбираем область R_m.
- Выбираем j и c (или L) так, чтобы добиться максимального уменьшения *неоднородности*, или *загрязненности*, (impurity) Q_m ($m = 1, 2, \ldots, M$).
- Строим разбиение (split) и повторяем действия.

Способы измерить «неоднородность»:

- Для задачи восстановления регрессии:

$$Q_m = \sum_{x_i \in R_m} (y_i - f(x_i))^2.$$

- Для задачи классификации:

$$Q_m = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i \neq k(m)) = 1 - \max_k p_{km} = 1 - p_{k(m),\, m},$$

p_{km} — доля объектов k-го класса в R_m.

2.5.3. Достоинства и недостатки деревьев решений

Достоинства:

- Поддерживают работу с входными переменными разных (смешанных) типов
- Возможность обрабатывать данные с пропущенными значениями
- Устойчивы к выбросам
- Нечувствительность к монотонным преобразованиям входных переменных
- Поддерживают работу с большими выборками
- Возможность интерпретации построенного решающего правила

Основной недостаток — плохая предсказательная (обобщающая) способность.

2.6. Ансамбли решающих правил

Ансамбль, или *комитет*, решающих правил,

или *аркинг* (arcing — adaptive reweighting and combining) — комбинирование решающих правил.

Рассмотрим задачу классификации на K классов.

$$\mathscr{Y} = \{1, 2, \ldots, K\}.$$

Пусть имеется M классификаторов («экспертов») $f_1, f_2, \ldots, f_M$

$$f_m : \ \mathscr{X} \to \mathscr{Y}, \qquad f_m \in \mathscr{F}, \qquad (m = 1, 2, \ldots, M)$$

Построим новый классификатор (простое голосование):

$$f(x) = \max_{k=1,\ldots,K} \sum_{m=1}^{M} I(f_m(x) = k),$$

$$f(x) = \frac{1}{M} \sum_{m=1}^{M} f_m(x),$$

Пример: $K = 2$, $M = 3$.

Решение принимается с использованием простого голосования.

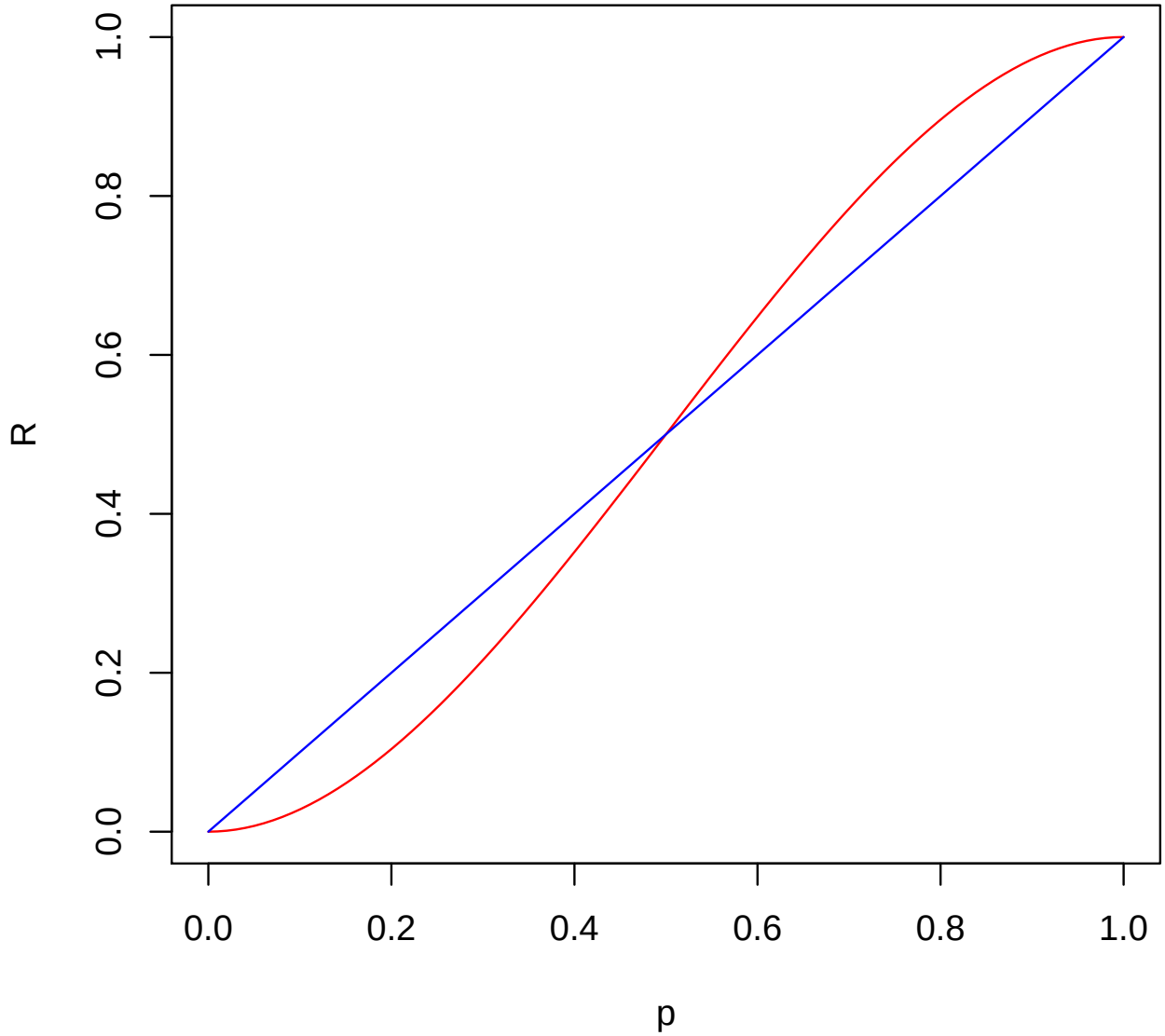
Пусть классификаторы независимы (на практике недостижимое требование!).

p — вероятность ошибки каждого отдельного классификатора.

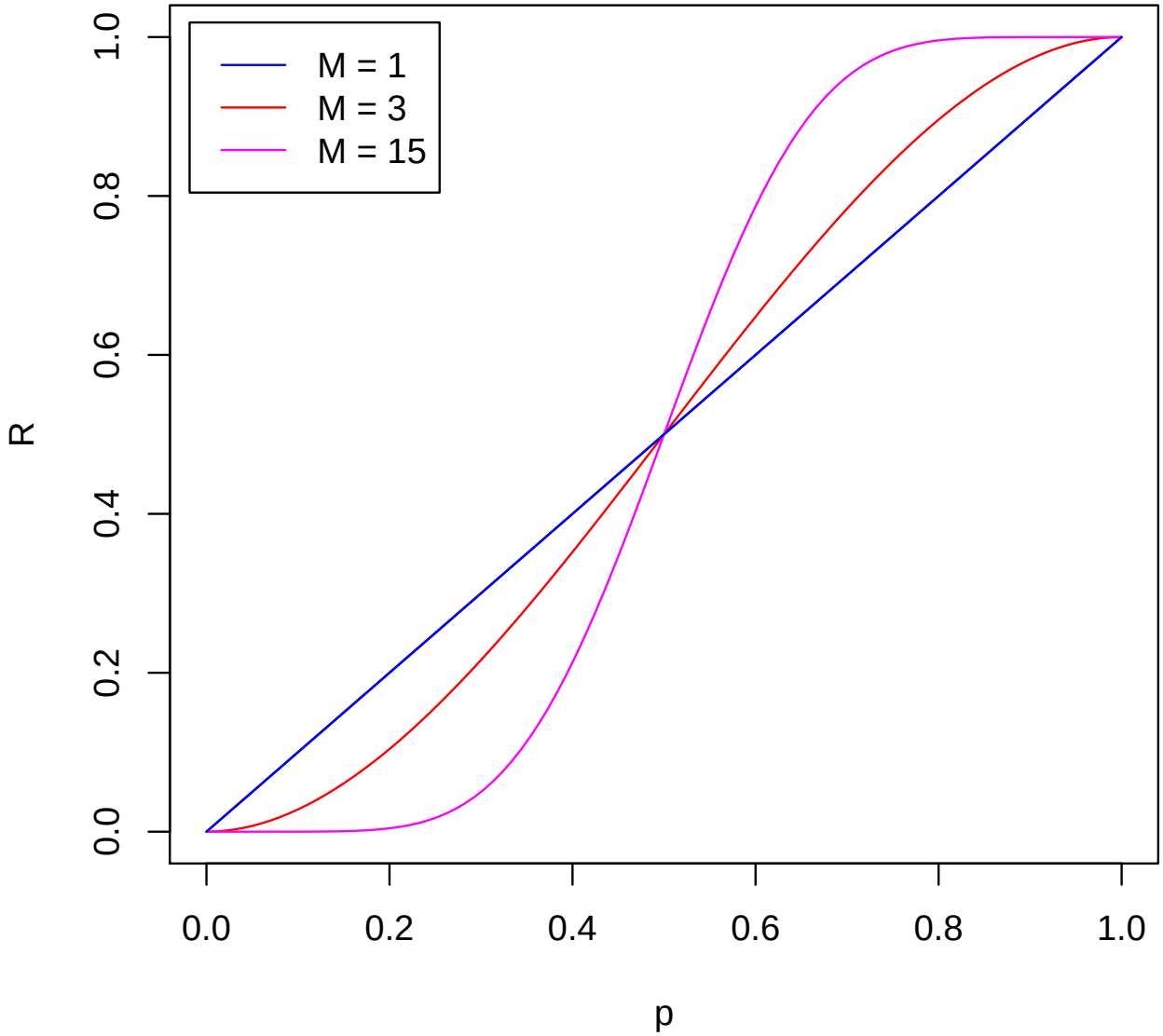
Тогда вероятность ошибки общего решения (ожидаемый риск) равен

$$R = p^3 + 3p^2(1 - p) = 3p^2 - 2p^3.$$

$$R = 3p^2 - 2p^3$$



$M = 1, 3, 15$



Можно ли научиться комбинировать слабые классификаторы, чтобы получить сильный [Kearns, Valiant, 1988]?

Два известных подхода:

- *Баггинг* и т. п.: пытаемся снизить зависимость экспертов друг от друга.
  - Bagging [Breiman, 1994]
  - Random Forests [Breiman, 2001]
  - . . .
- *Бустинг*: эксперты учатся на ошибках других.
  - AdaBoost [Freund, Schapire, 1995]
  - Gradient Boosting Trees [Friedman, 1999]
  - . . .

2.6.1. Баггинг и т. п.

Bagging — от bootstrap aggregation [Breiman, 1994]

Классификатор f_m обучается на bootstrap–выборке ($m = 1, 2, \ldots, M$).

Финальный классификатор f — функция голосования:

$$f(x) = \max_{k=1,\ldots,K} \sum_{m=1}^{M} I\big(f_m(x) = k\big).$$

Случайный лес

Random forests [Breiman, 2001]

Ансамбль параллельно обучаемых «независимых» деревьев решений.

Независимое построение определенного количества M (например, 500) деревьев:

Генерация случайной подвыборки из обучающей выборки (50–70% от размера всей обучающей выборки) и построение дерева решений по данной подвыборке (в каждом новом узле дерева переменная для разбиения выбирается не из всех признаков, а из случайно выбранного их подмножества небольшой мощности).

begin RandomForests

for $m = 1, 2, \ldots, M$

begin

По обучающей выборке построить бутстрэп-выборку

Построить дерево f_m, рекурсивно применяя следующую процедуру,

пока не будет достигнут минимальный размер sz:

begin

Построить случайный набор из p признаков

Выбрать из него лучшую переменную и построить 2 дочерних узла

end

end

Для задачи восстановления регрессии **return** $f = \frac{1}{M} \sum_{m=1}^{M} f_m$

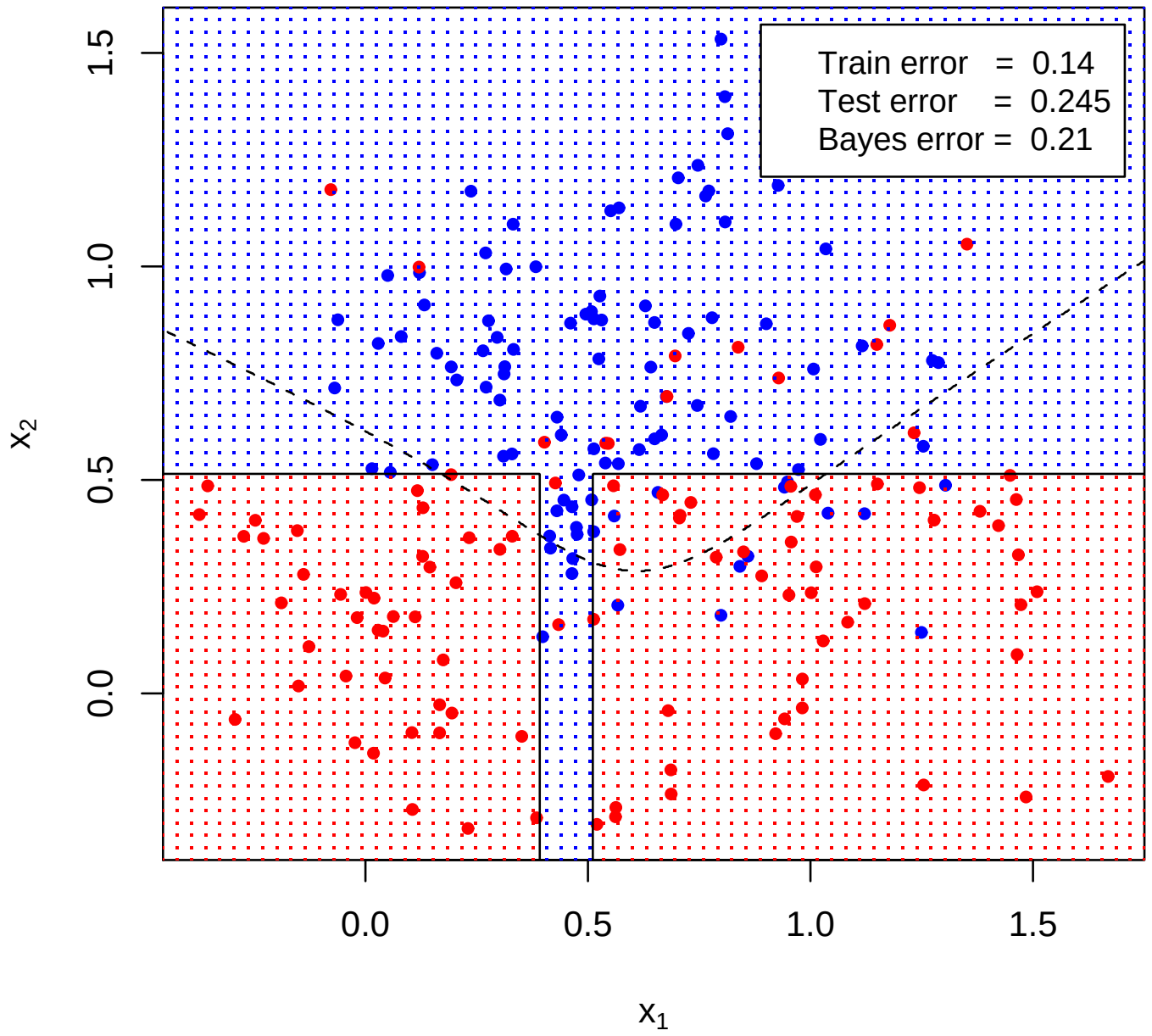
Для задачи классификации **return** $f = \underset{k}{\text{argmax}} \sum_{m=1}^{M} I(f_m = k)$

end

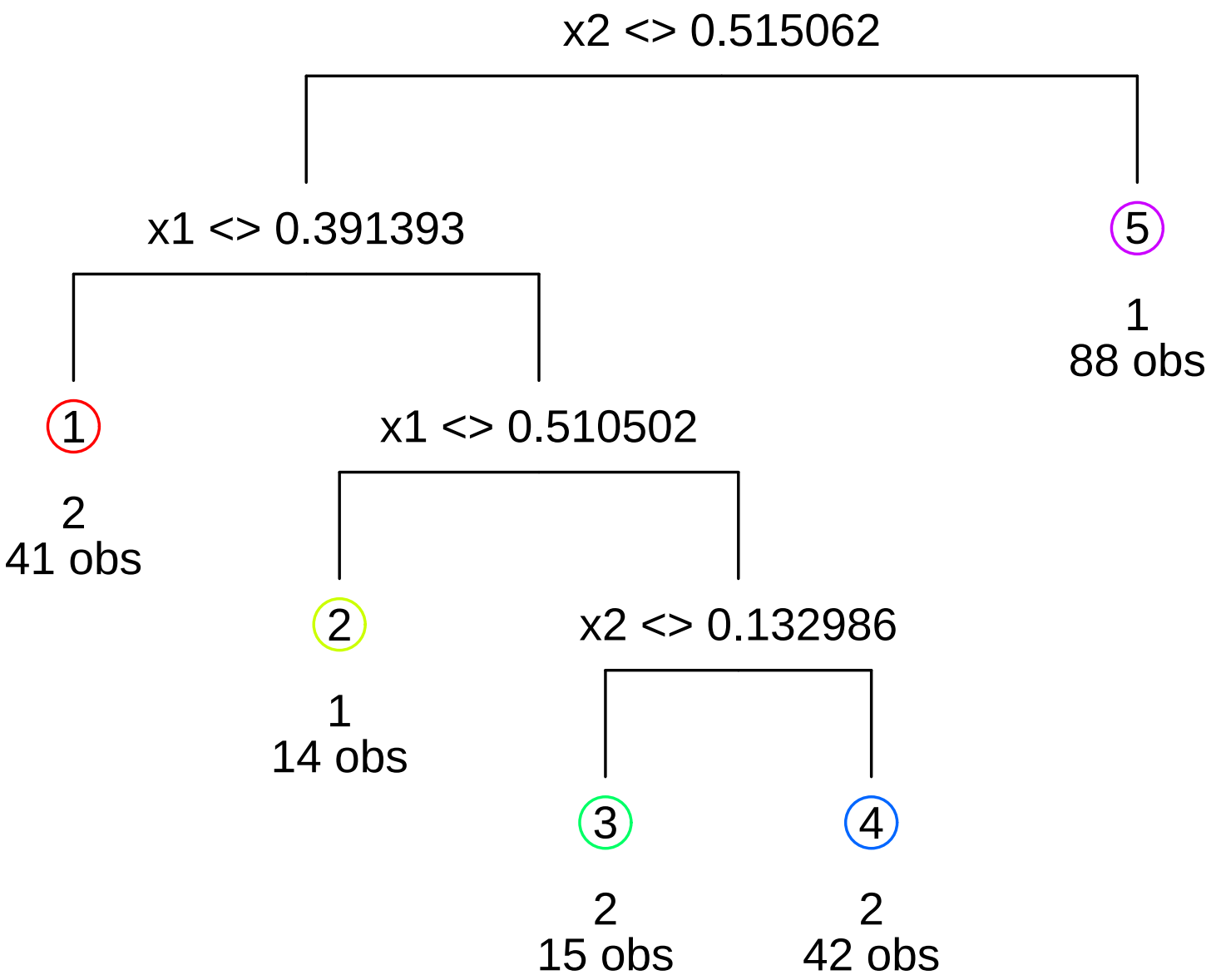
Для задачи восстановления регрессии, например, $p = \sqrt{d}$, sz = 3

Для задачи классификации, например, $p = d/3$, sz = 1

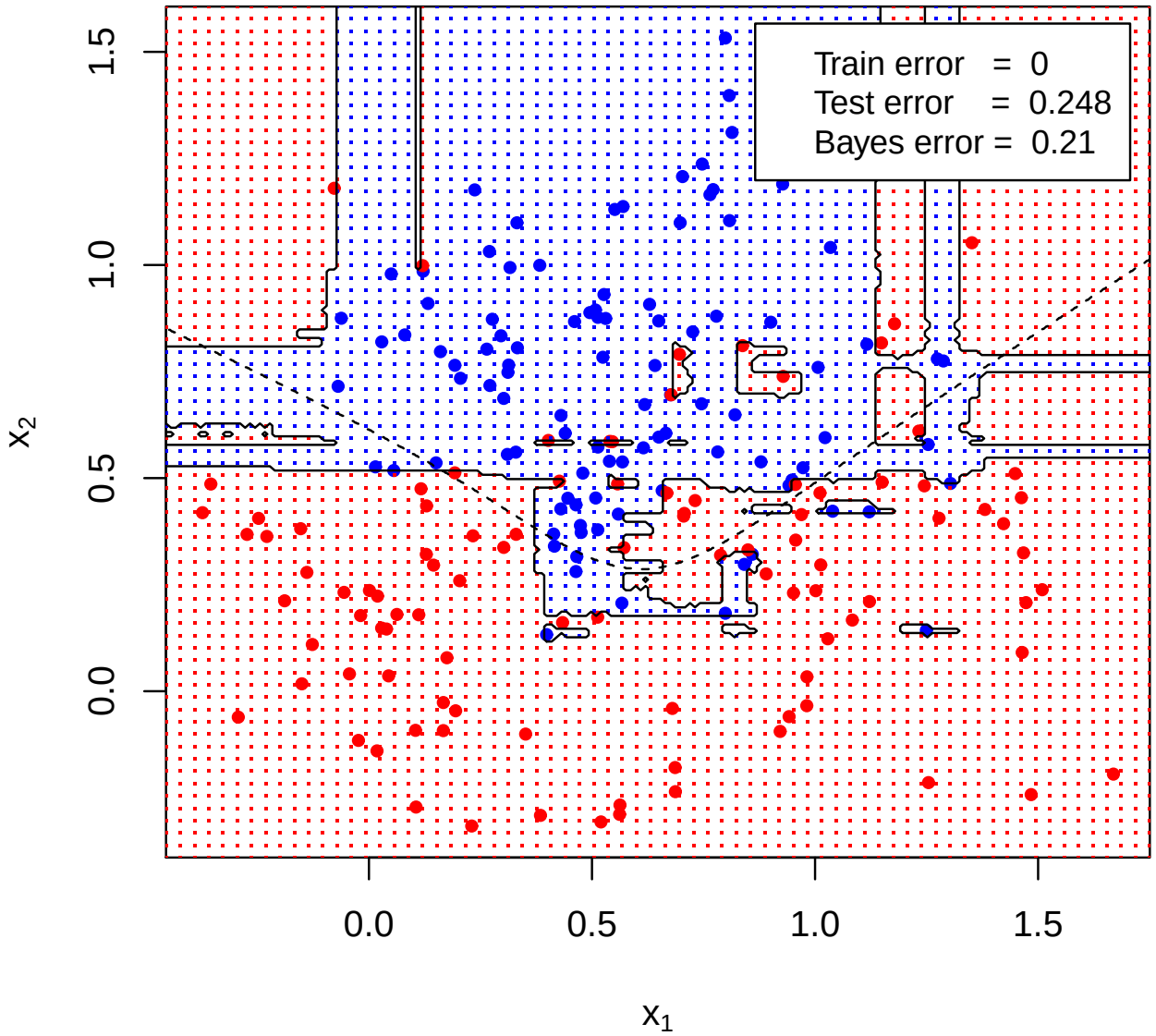
Оптимальное дерево после проведения отсечений — 5 листьев



Оптимальное дерево после проведения отсечений — 5 листьев



Random forest: 500 деревьев



2.7. Эксперименты

Данные: UCI Machine Learning Repository http://archive.ics.uci.edu/ml/

Software: OpenCV Library http://opencv.willowgarage.com
Эксперимент — П.Н. Дружков

Задачи классификации

10-CV ошибка

Задача	N	d (колич.+ном.)	K	GBT	DTree	RF	ExtRF	SVM
Agaricus lepiota	8124	22 (0 + 22)	2	0	0.00	0	0	0
Liver disorders	345	6 (6 + 0)	2	0.25	0.31	0.22	0.25	0.28
Car evaluation	1728	6 (0 + 6)	4	0	0.051	0.036	0.039	0.050

GBT — Gradient Boosting Trees — градиентный бустинг деревьев решений,

DTree — Decision Tree — деревья решений,

RF — Random Forests — случайные леса,

ExtRF — Extremely Random Forests — экстремально случайные леса,

SVM — Support Vector Machine — машина опорных векторов

Задачи восстановления регрессии. Средняя абсолютная 10-CV ошибка

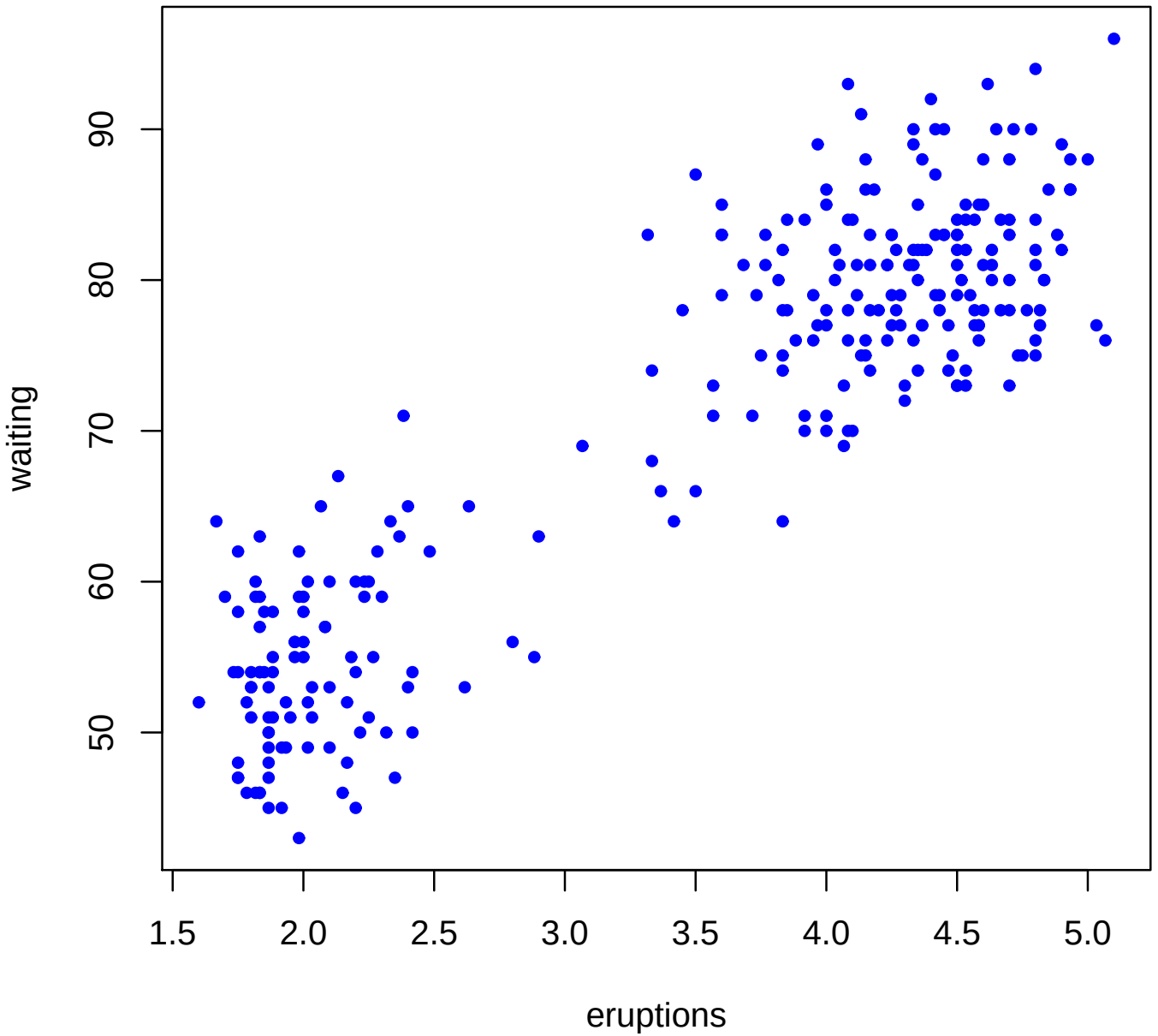
Задача	N	d (колич.+ном.)	GBT	DTree	RF	ExtRF	SVM
Auto-mpg	398	7 (4 + 3)	2.00	2.24	1.88	2.15	2.98
Computer hardware	209	8 (7 + 1)	12.61	15.62	11.62	9.63	37.00
Concrete slump	103	9 (9 + 0)	2.26	2.92	2.60	2.36	1.77
Forestfires	517	12 (10 + 2)	18.74	17.26	17.79	16.64	12.90
Boston housing	506	13 (13 + 0)	2.03	2.60	2.13	2.20	4.05
Import-85	201	25 (14 + 11)	1305	1649	1290	1487	1787
Servo	167	4 (0 + 4)	0.238	0.258	0.247	0.420	0.655
Abalone	4177	8 (7 + 1)	1.470	1.603	1.492	1.498	2.091

3. Обучение без учителя: кластеризация

Пример 1. Извержения гейзера

Рассмотрим данные о времени между извержениями и длительностью извержения гейзера Old Faithful geyser in Yellowstone National Park, Wyoming, USA (*A. Azzalini, A.W. Bowman* A look at some data on the Old Faithful geyser // Applied Statistics. 1990, 39. P. 357-–365.)

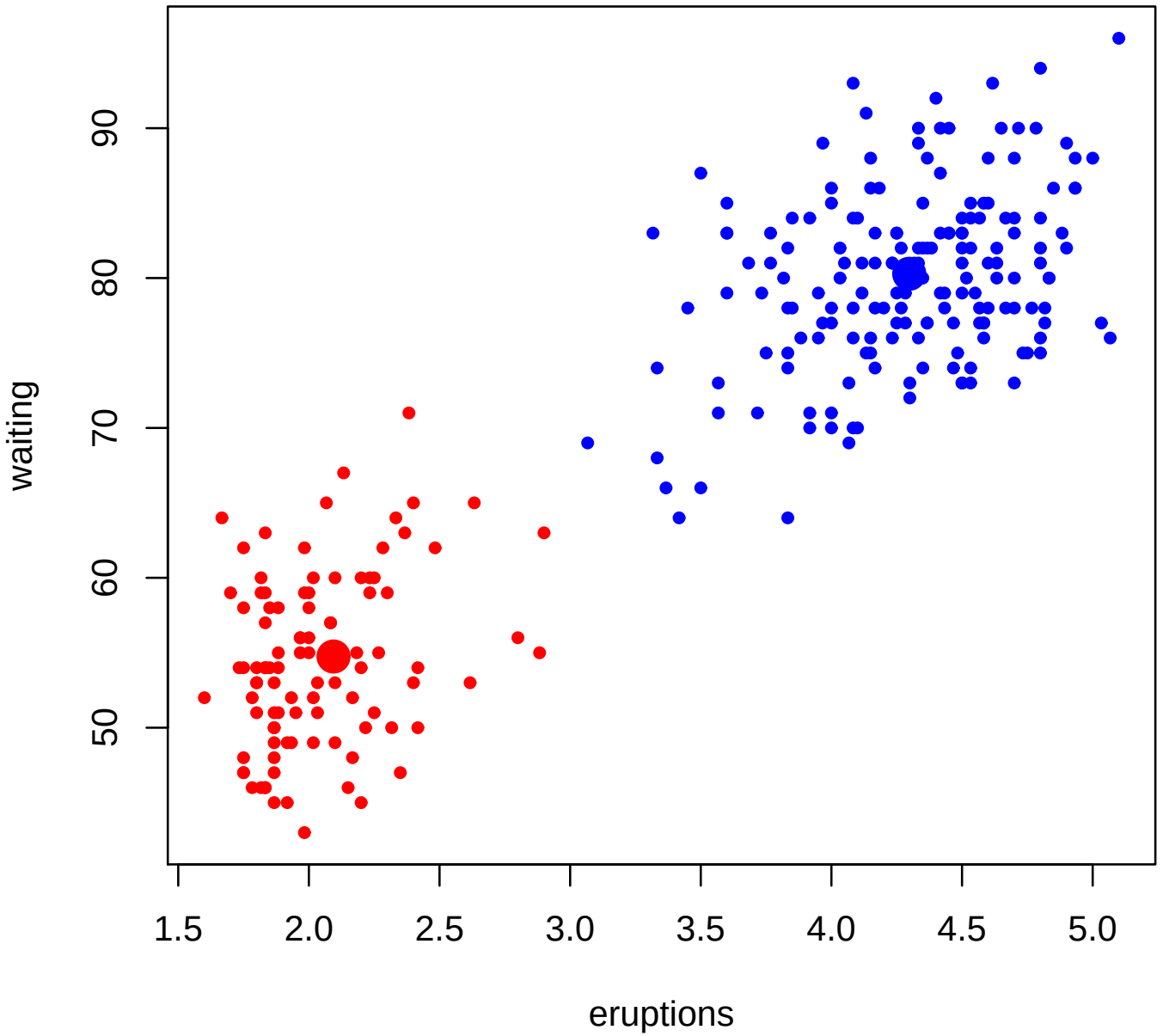
Диаграмма, представляющая данные о времени извержения и промежутках между извержениями гейзера.



Мы видим, что точки группируются в два кластера.

В одном кластере находятся точки, соответствующие извержениям с малой длительностью и малым временем ожидания.

В другом — с большой длительностью и большим временем ожидания.


waiting
50
60
70
80
90
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
eruptions

Пример 2. Анализ данных, полученных с биочипов

Биочип, или *микроэррэй*, (biochip, microarray) — это миниатюрный прибор, измеряющий уровень экспрессии генов в имеющемся материале.

Экспрессия — это процесс перезаписи информации с гена на РНК, а затем на белок.

Количество и даже свойства получаемого белка зависят не только от гена, но также и от различных внешних факторов (например, от введенного лекарства).

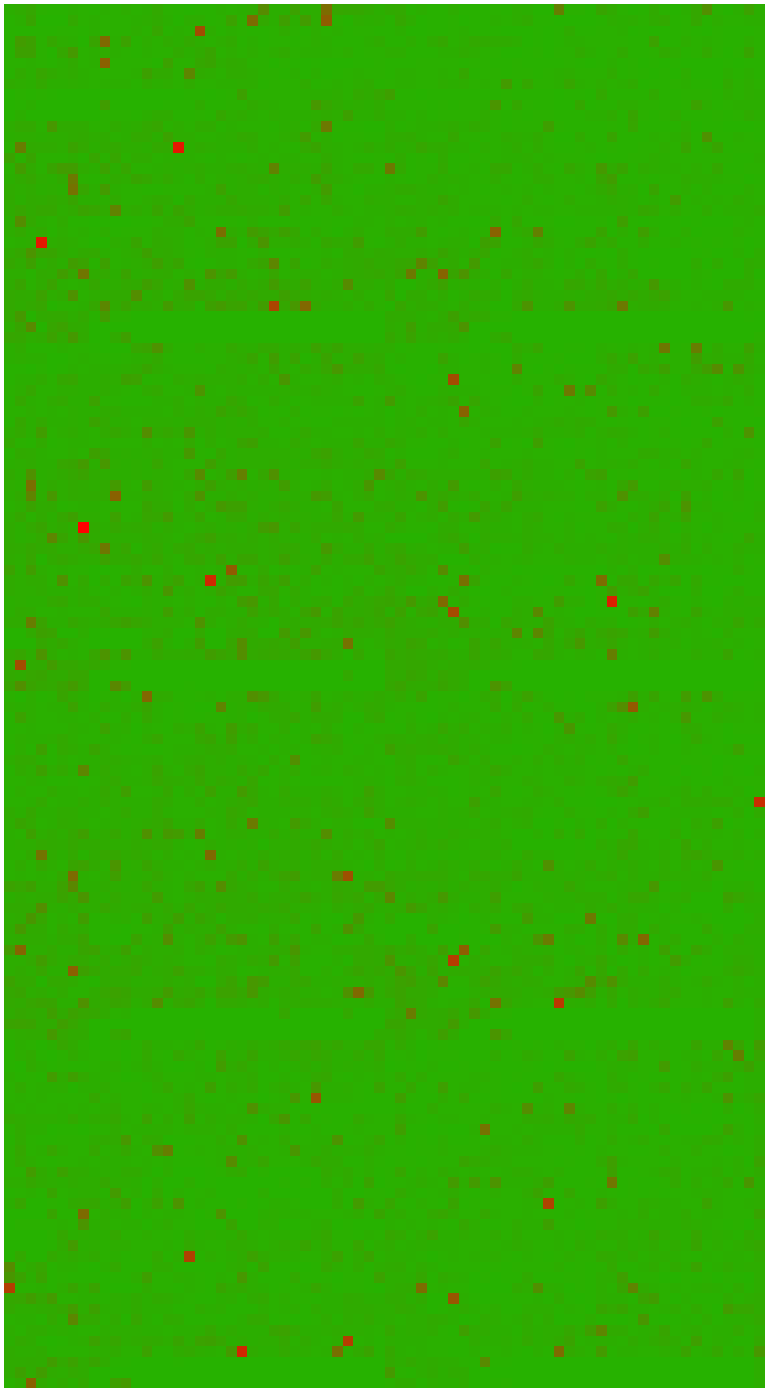
Таким образом, уровень экспрессии — это мера количества генерируемого белка (и скорости его генерирования).

На биочип кроме исследуемого материала помещается также «контрольный» генетический материал.

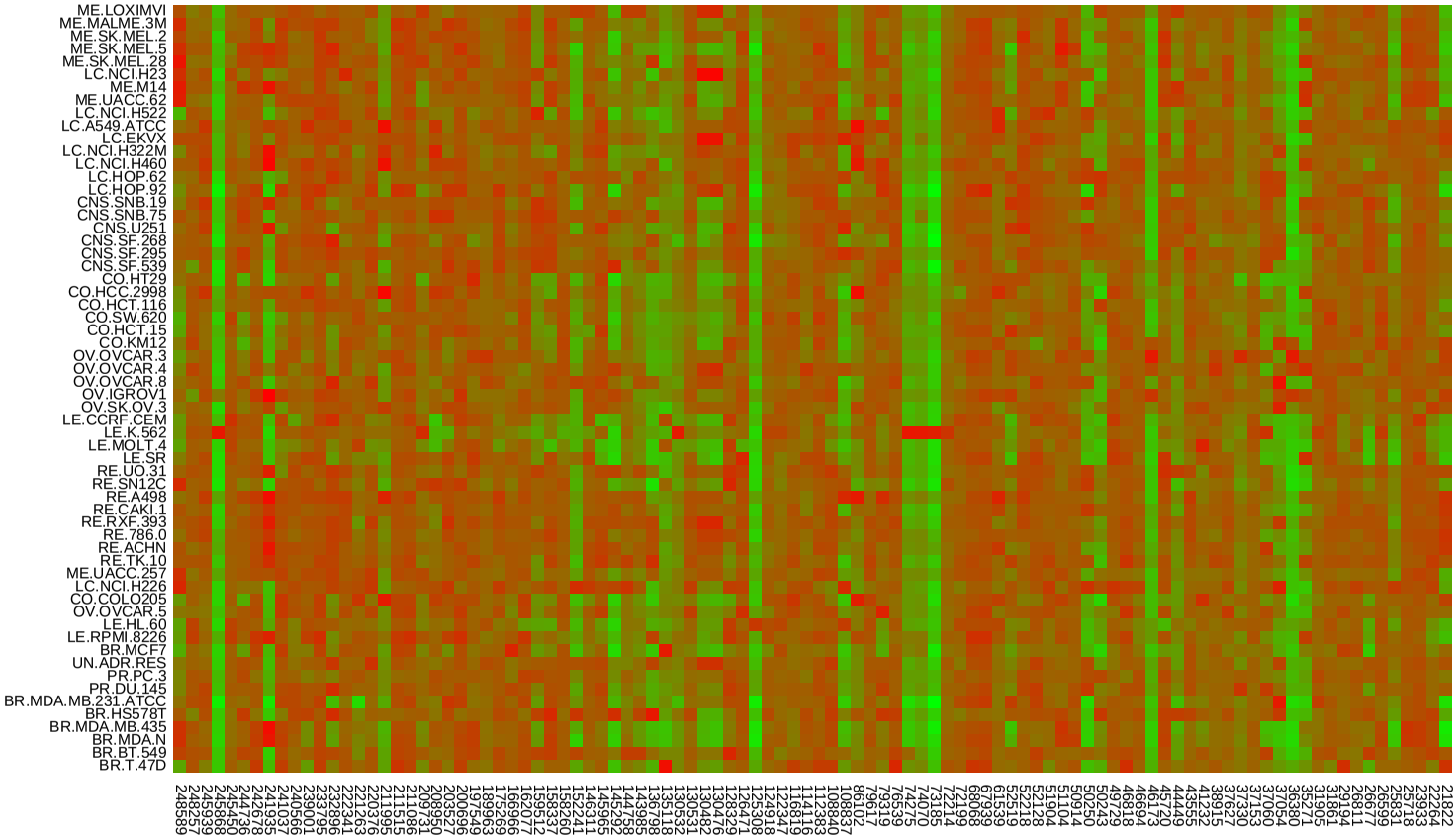
Положительные значения (красный цвет) — увеличение уровня экспрессии по сравнению с контрольным.

Отрицательные значения (зеленый цвет) — уменьшение.

Условное изображение биочипа. Каждая точка на рисунке соответствует определенному гену. Всего анализируется $132 \times 72 = 9504$ гена. Brown, V.M., Ossadtchi, A., Khan, A.H., Yee, S., Lacan, G., Melega, W.P., Cherry, S.R., Leahy, R.M., and Smith, D.J.; Multiplex three dimensional brain gene expression mapping in a mouse model of Parkinson's disease; Genome Research 12(6): 868-884 (2002).



Данные для 60 экспериментов с биочипом
http://discover.nci.nih.gov/datasetsNature2000.jsp Строки соответсвуют генам, столбцы — экспериментам. Приведены только первые 100 строк (из общего числа 1375). Строки, содержащие отсутствующие значения, исключены.


ME.LOXIMVI
ME.MALME.3M
ME.SK.MEL.2
ME.SK.MEL.5
ME.SK.MEL.28
LC.NCI.H23
ME.M14
ME.UACC.62
LC.NCI.H522
LC.A549.ATCC
LC.EKVX
LC.NCI.H322M
LC.NCI.H460
LC.HOP.62
LC.HOP.92
CNS.SNB.19
CNS.SNB.75
CNS.U251
CNS.SF.268
CNS.SF.295
CNS.SF.539
CO.HT29
CO.HCC.2998
CO.HCT.116
CO.SW.620
CO.HCT.15
CO.KM12
OV.OVCAR.3
OV.OVCAR.4
OV.OVCAR.8
OV.IGROV1
OV.SK.OV.3
LE.CCRF.CEM
LE.K.562
LE.MOLT.4
LE.SR
RE.UO.31
RE.SN12C
RE.A498
RE.CAKI.1
RE.RXF.393
RE.786.0
RE.ACHN
RE.TK.10
ME.UACC.257
LC.NCI.H226
CO.COLO205
OV.OVCAR.5
LE.HL.60
LE.RPMI.8226
BR.MCF7
UN.ADR.RES
PR.PC.3
PR.DU.145
BR.MDA.MB.231.ATCC
BR.HS578T
BR.MDA.MB.435
BR.MDA.N
BR.BT.549
BR.T.47D
248589
248257
245939
245868
245450
244736
242678
241935
241037
239001
238795
232896
223341
221263
220376
211995
211515
211086
209731
208950
207649
189963
175269
166966
162077
159512
158337
158260
152241
146311
145965
145255
144385
138798
135118
130532
130531
130482
130476
128329
126471
125308
124918
122347
116816
112383
108840
108837
86102
79617
79319
76539
74275
74070
73185
72214
72198
67889
62869
61539
52519
52218
52128
51904
51104
50914
50250
50243
49729
46918
46603
45720
44449
43555
41232
38915
37627
37330
37153
37060
37054
36580
36501
35905
31861
29194
26811
26677
26599
25831
25718
23933
22264
21822

Поставим следующие задачи:

(а) Найти гены, показавшие высокую экспрессию, в заданных экспериментах.

т.е. найти наиболее красные клетки в заданных столбцах.

(б) Разбить гены на группы в зависимости от влияния на них экспериментов. Гены, реагирующие «почти одинаковым» образом в «большом» числе эсперименов, должны попасть в одну группу. Гены, реагирующие по-разному, должны находиться в разных группах.

т.е. разбить строки на группы (кластеры) «похожих» между собой строк

(в) Разбить эксперименты на группы в зависимости от их влияния на гены. Эксперименты, в которых одинаковые гены реагировали «сходным» образом должны оказаться в одной группе. Эксперименты, в которых гены реагировали «различно», должны находиться в разных группах.

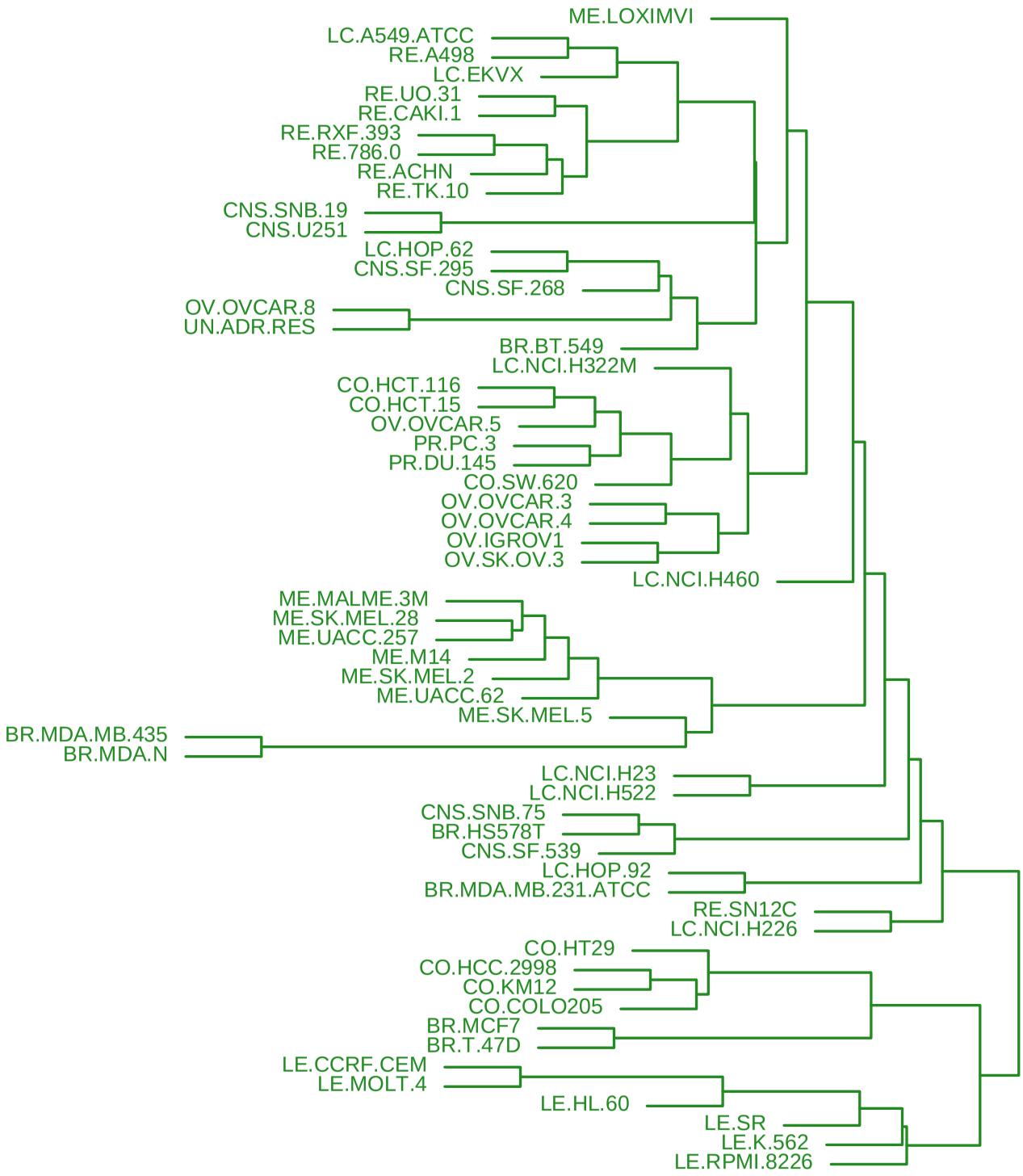
т.е. разбить столбцы на группы (кластеры) «похожих» между собой строк

Задачи (б) и (в) — это задачи кластерного анализа.

Таксономия 60 клеток на основе анализа уровня экспресии их генов.

http://discover.nci.nih.gov/datasetsNature2000.jsp

60 прецедентов, 1375 признаков

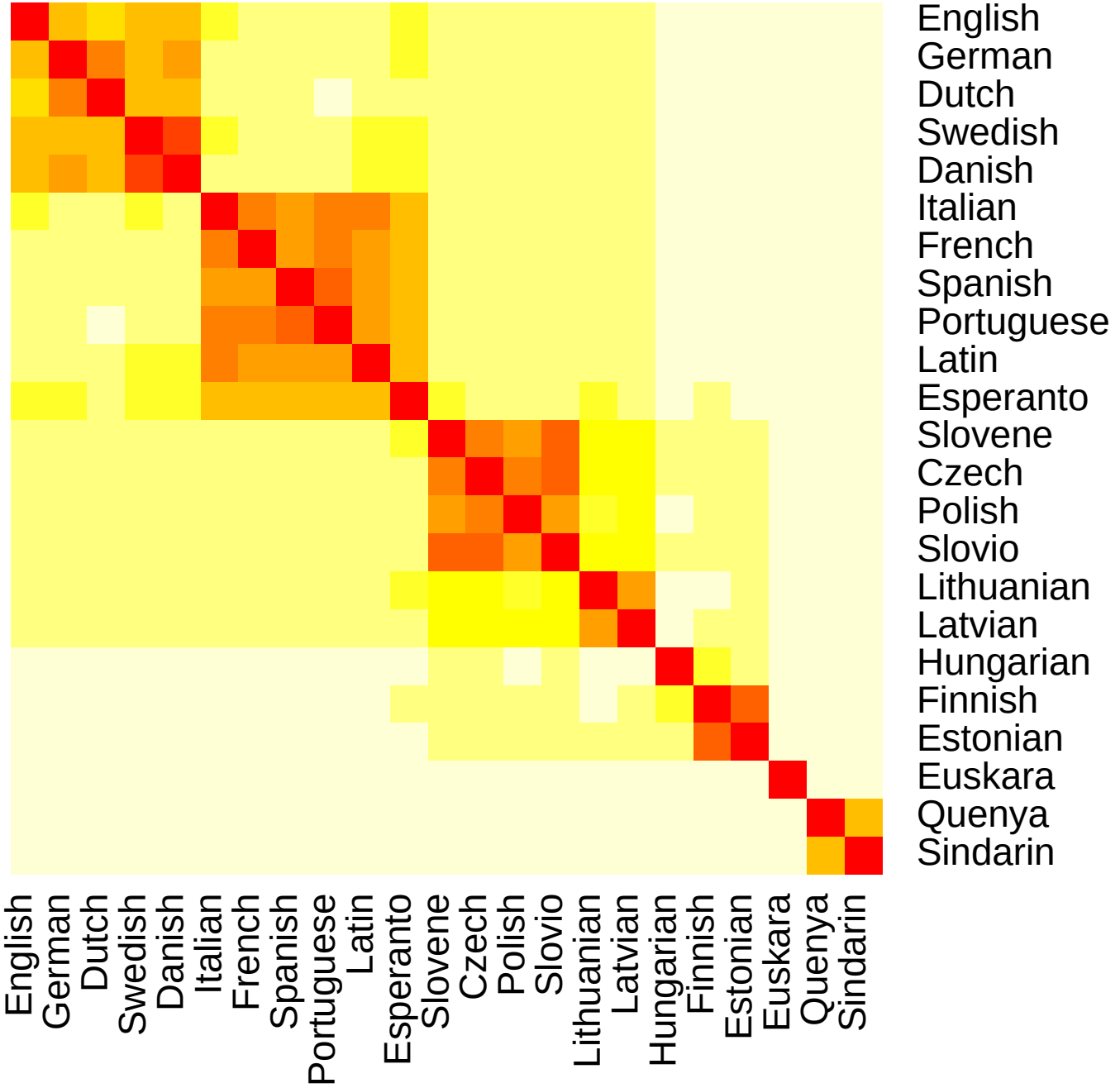

ME.LOXIMVI
LC.A549.ATCC
RE.A498
LC.EKVX
RE.UO.31
RE.CAKI.1
RE.RXF.393
RE.786.0
RE.ACHN
RE.TK.10
CNS.SNB.19
CNS.U251
LC.HOP.62
CNS.SF.295
CNS.SF.268
OV.OVCAR.8
UN.ADR.RES
BR.BT.549
LC.NCI.H322M
CO.HCT.116
CO.HCT.15
OV.OVCAR.5
PR.PC.3
PR.DU.145
CO.SW.620
OV.OVCAR.3
OV.OVCAR.4
OV.IGROV1
OV.SK.OV.3
LC.NCI.H460
ME.MALME.3M
ME.SK.MEL.28
ME.UACC.257
ME.M14
ME.SK.MEL.2
ME.UACC.62
ME.SK.MEL.5
BR.MDA.MB.435
BR.MDA.N
LC.NCI.H23
LC.NCI.H522
CNS.SNB.75
BR.HS578T
CNS.SF.539
LC.HOP.92
BR.MDA.MB.231.ATCC
RE.SN12C
LC.NCI.H226
CO.HT29
CO.HCC.2998
CO.KM12
CO.COLO205
BR.MCF7
BR.T.47D
LE.CCRF.CEM
LE.MOLT.4
LE.HL.60
LE.SR
LE.K.562
LE.RPMI.8226

Пример 3. Списки Сводеша и таксономия языков

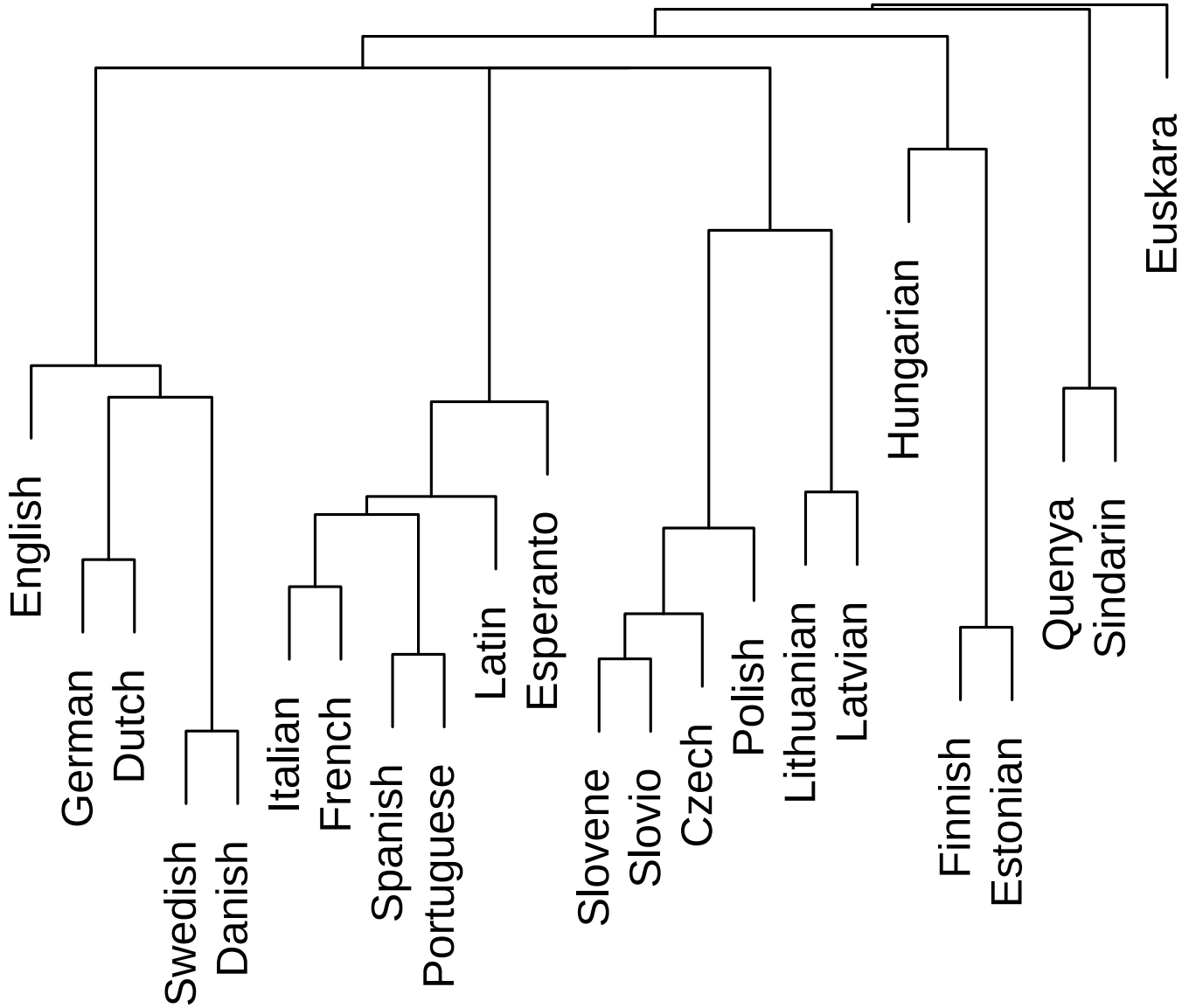
Список Сводеша (Swadesh) — список из 207 слов «базового» словаря (местоимения, числительные 1–5, глаголы, обозначающие простые действия и т. п.)

№	Русский	Английский	Немецкий	Итальянский	Французский	Чешский
1	я	I	ich	io	je	já
2	ты	you	du	tu	tu	ty
3	он	he	er	lui	il	on
4	мы	we	wir	noi	nous	my
5	вы	you	ihr	voi	vous	vy
6	они	they	sie	loro	ils	oni
7	этот	this	dieses	questo	ceci	tento
8	тот	that	jenes	quello	cela	tamten
9	здесь	here	hier	qui	ici	zde
10	там	there	dort	lá	lá	tam
11	кто	who	wer	chi	qui	kdo
12	что	what	was	che	quoi	co
13	где	where	wo	dove	où	kde
14	когда	when	wann	quando	quand	kdy
15	как	how	wie	come	comment	jak
16	не	not	nicht	non	ne. . . pas	ne
. . . . .						
205	если	if	wenn	se	si	jestlize
206	потому что	because	weil	perché	parce que	protoze
207	имя	name	Name	nome	nom	jméno

Матрица сходства между некоторыми языками, построенная на основе списков Сводеша.


English
German
Dutch
Swedish
Danish
Italian
French
Spanish
Portuguese
Latin
Esperanto
Slovene
Czech
Polish
Slovio
Lithuanian
Latvian
Hungarian
Finnish
Estonian
Euskara
Quenya
Sindarin

Дерево иерархической кластеризации для 23 языков, построенное на основе списков Сводеша.



4. Выводы

- Узнали, что такое обучение с учителем и обучение без учителя
- Узнали, что такое метод ближайших соседей
- Познакомились с идеей метода SVM (машина опорных векторов)
- Узнали, что такое деревья решений
- Познакомились с двумя принципами комбинирования слабых классификаторов (бустинг и баггинг)
- Узнали, что такое алгоритм «Random Forests»
- Узнали, что такое переобучение; осознали необходимость борьбы с ним

КОНЕЦ