МАШИННОЕ ОБУЧЕНИЕ И АНАЛИЗ ДАННЫХ

(Machine Learning and Data Mining)

Н. Ю. Золотых

http://www.uic.unn.ru/~zny/ml

Лекция 13

Деревья решений

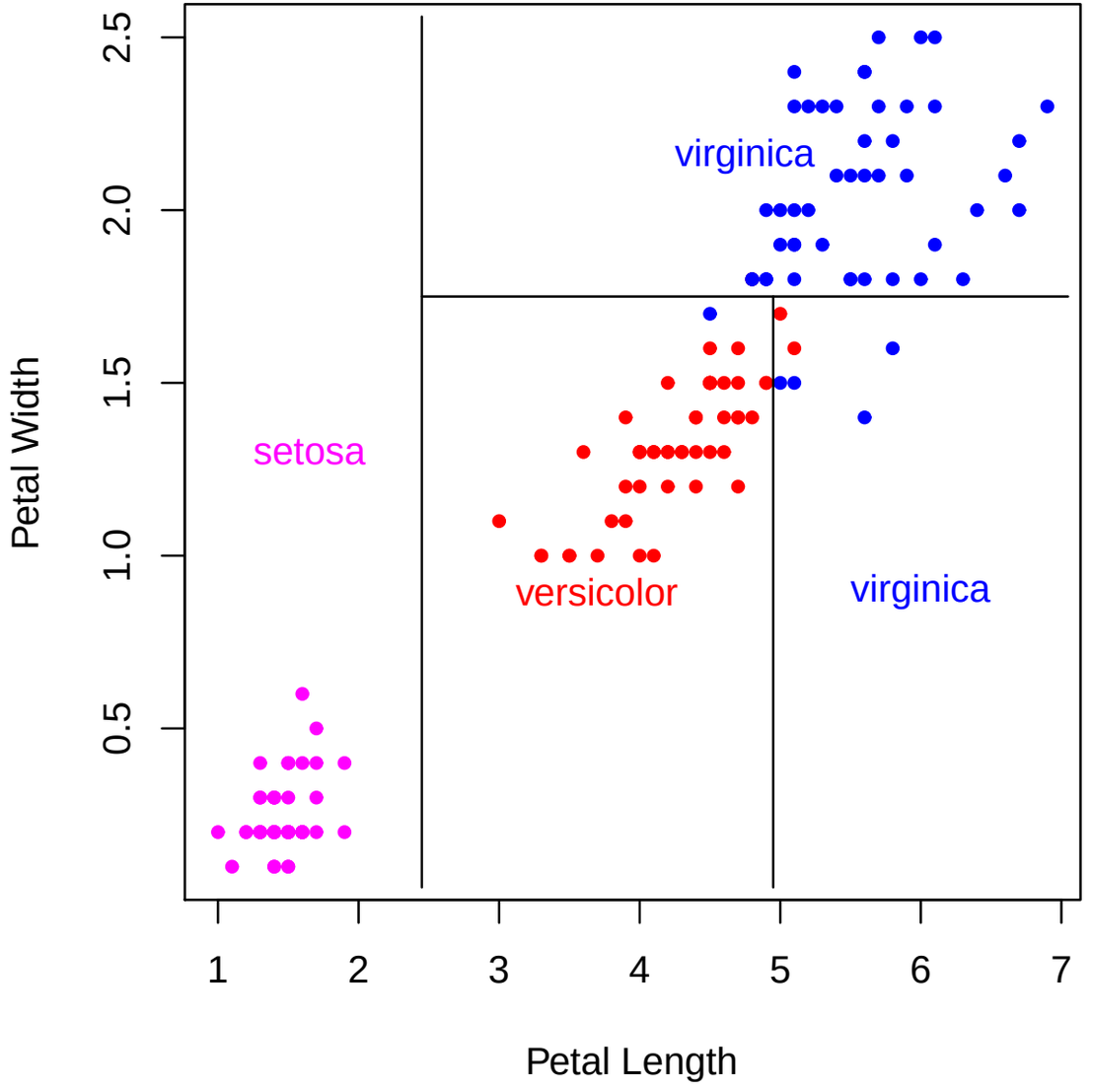
Пространство признаков разбивается на параллелепипеды со сторонами, параллельными осям координат (ящики).

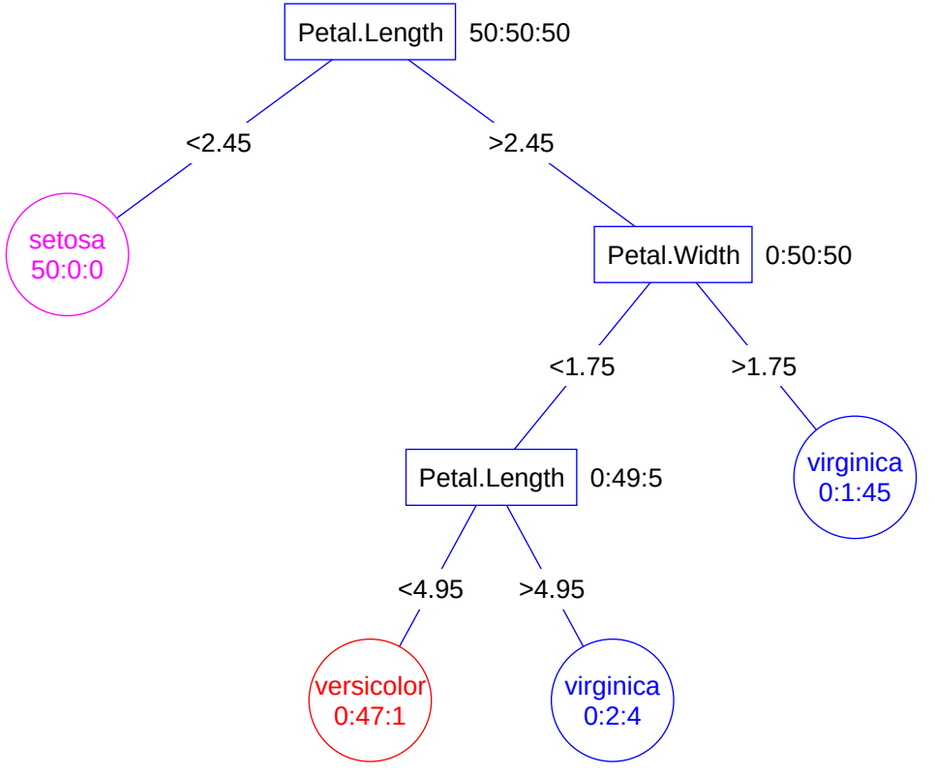
В каждом ящике ответ аппроксимируется с помощью некоторой простой модели, обычно константой (как для задачи классификации, так и для задачи восстановления регрессии).

Используются только рекурсивные гильотинные разбиения.

Задача классификации цветов ириса (Fisher, 1936).

x_1, x_2 — длина и ширина чашелистика.



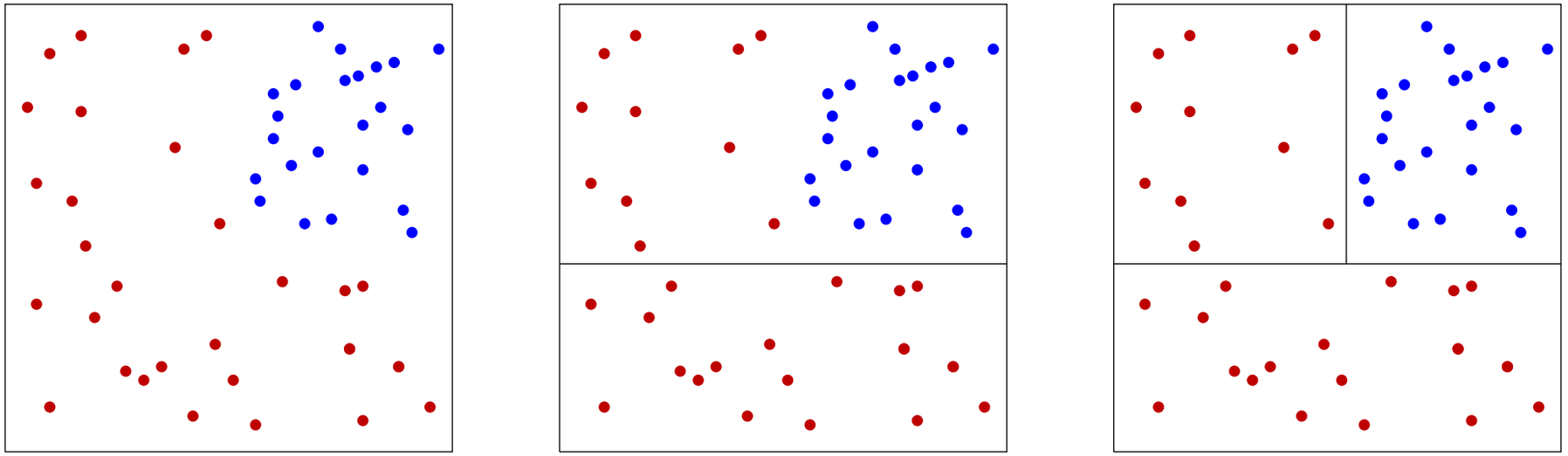


Каждому узлу дерева соответствует «ящик» в пространстве признаков.
Этот ящик может разбиваться далее на следующих ярусах.

13.1. 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] ← CHAID [Kass, 1980] ← THAID [Morgan & Messenger 1973] ← AID [Morgan & Sonquist, 1963]

— это *жадные рекурсивные* алгоритмы, на каждом шаге разбивающие очередной ящик, чтобы добиться максимального уменьшения взвешенной *неоднородности*:



13.2. 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 взвешенной *неоднородности* (impurity) (загрязненности, примесности, хаоса) Q_m ($m = 1, 2, \ldots, M$) (т. е. максимального увеличения «*прироста информации*»).
- Строим разбиение (split) и повторяем действия.

Способы измерить «неоднородности»:

- Для задачи восстановления регрессии:

$$Q_m = \frac{1}{N_m} \sum_{x^{(i)} \in R_m} \left(y^{(i)} - f(x^{(i)}) \right)^2 = \frac{1}{N_m} \sum_{x^{(i)} \in R_m} \left(y^{(i)} - c_m \right)^2,$$

где $N_m = \sum I(x^{(i)} \in R_m)$ — количество $x^{(i)} \in R_m$.

Взвешенная неоднородность:

$$\widehat{Q}_m = \frac{N_{m1}}{N_m} Q_{m1} + \frac{N_{m2}}{N_m} Q_{m2} = \frac{1}{N_m} \left(\sum_{x^{(i)} \in R_{m1}} \left(y^{(i)} - c_{m1} \right)^2 + \sum_{x^{(i)} \in R_{m2}} \left(y^{(i)} - c_{m2} \right)^2 \right) \to \min,$$

где $R_m = R_{m1} \cup R_{m2}$, $N_{m1} = \sum I(x^{(i)} \in R_{m1})$, $N_{m2} = \sum I(x^{(i)} \in R_{m2})$.

- Для задачи классификации:
  - Ошибка классификации:

$$Q_m = \frac{1}{N_m} \sum_{x^{(i)} \in R_m} I\left(y^{(i)} \neq k(m)\right) = 1 - \max_k p_{km} = 1 - p_{k(m),\, m},$$

    p_{km} — доля объектов k-го класса в R_m, $k(m)$ — класс, преобладающий в R_m.
  - Индекс К. Джини (вероятность, что два наугад взятых элемента из R_m принадлежат разным классам):

$$Q_m = \sum_{k \neq k'} p_{mk} p_{mk'} = \sum_{k=1}^{K} p_{mk}(1 - p_{mk}) = 1 - \sum_{k=1}^{K} p_{mk}^2.$$

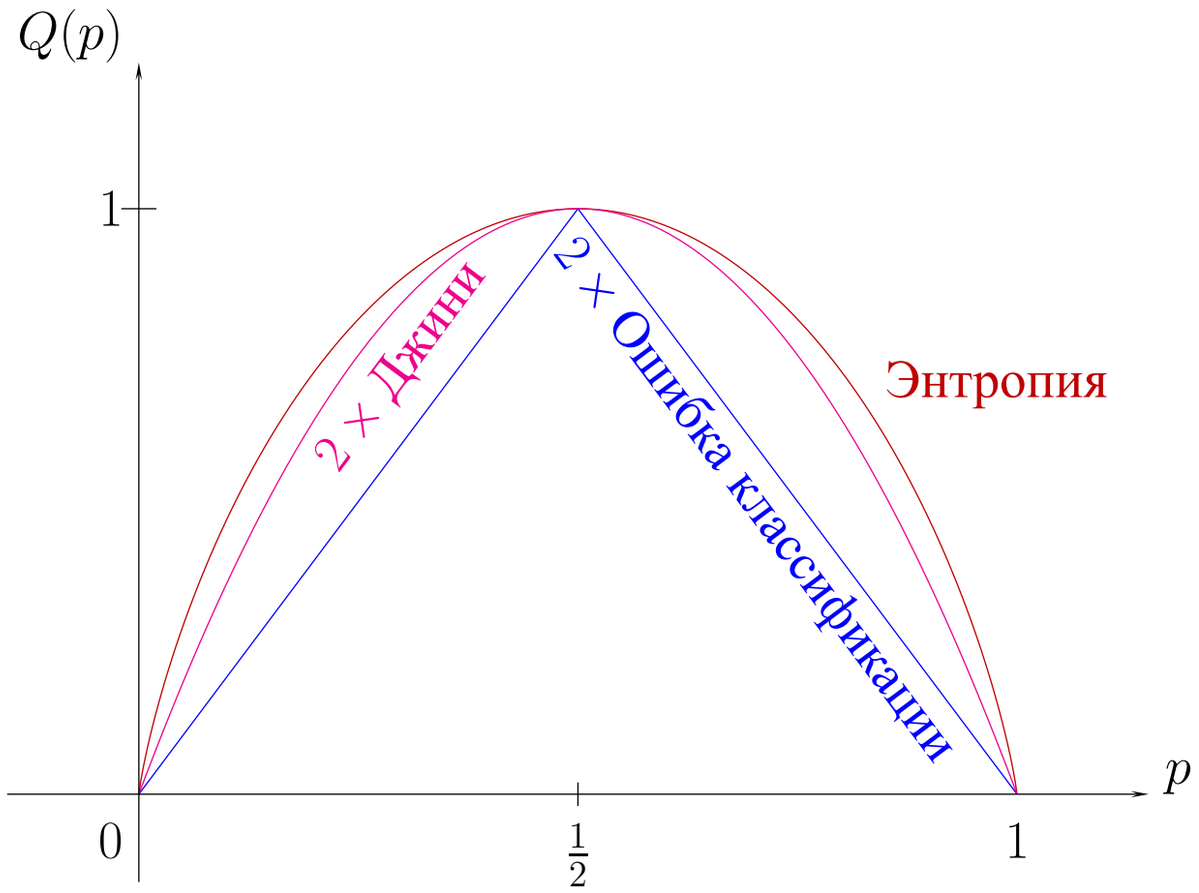
  - Энтропия (количество информации):

$$Q_m = -\sum_{k=1}^{K} p_{mk} \log p_{mk}.$$

Если $K = 2$, то эти функции равны соответственно

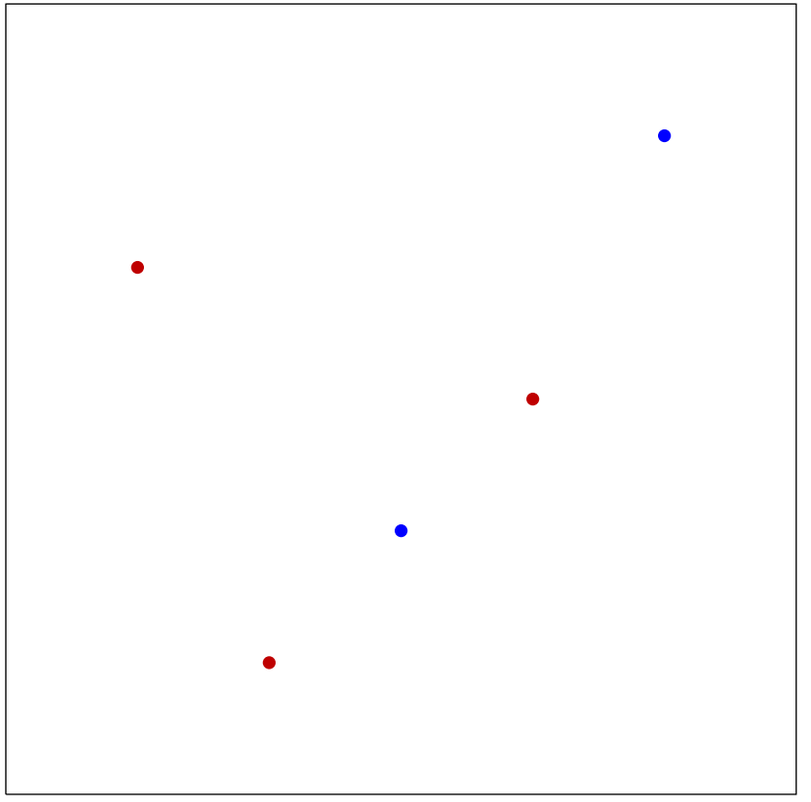
$$1 - \max\{p, 1-p\}, \quad 2p(1-p), \quad -p\log p - (1-p)\log(1-p),$$

где $p = p_{1m}$ — доля объектов 1-го класса, попавших в ящик R_m.



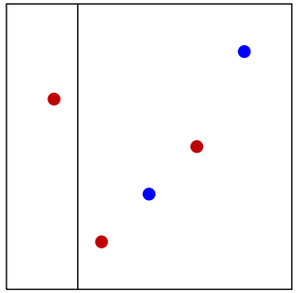
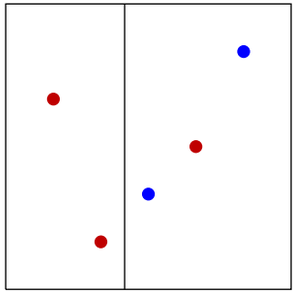
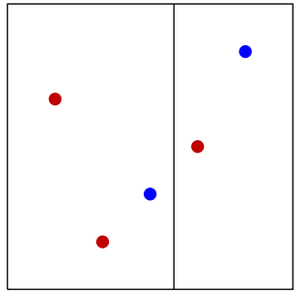
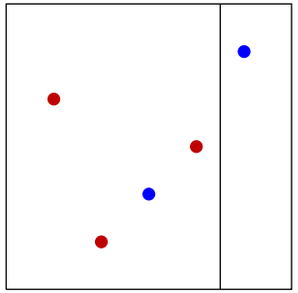
Функции похожи друг на друга. Индекс Джини и энтропия являются гладкими функциями и поэтому более податливы для численной оптимизации.

Каждая из трех приведенных функций равна нулю тогда и только тогда, когда в узле присутствуют объекты только одного класса.



$$Q_{\text{misclass}} = \frac{2}{5} = 0.4, \qquad Q_{\text{Gini}} = 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2 = \frac{12}{25} = 0.4800,$$

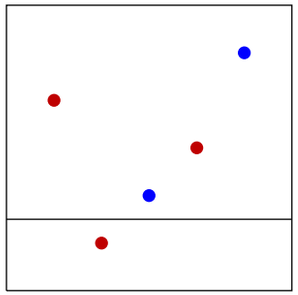
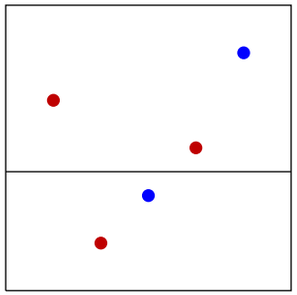
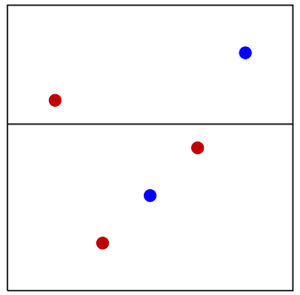
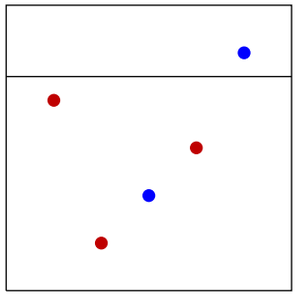
$$Q_{\text{entropy}} = -\frac{2}{5}\log\frac{2}{5} - \frac{3}{5}\log\frac{3}{5} = 0.9710.$$

$N_m \widehat{Q}_{\text{misclass}} =$ $1 \cdot 0 + 4 \cdot \frac{1}{2} = 2$ $2 \cdot 0 + 3 \cdot \frac{1}{3} = 1$ $3 \cdot \frac{1}{3} + 2 \cdot \frac{1}{2} = 2$ $4 \cdot \frac{1}{4} + 1 \cdot 0 = 1$

$N_m \widehat{Q}_{\text{Gini}} =$ $1 \cdot 0 + 4 \cdot \frac{1}{2} = 2$ $2 \cdot 0 + 3 \cdot \frac{4}{9} = \frac{4}{3}$ $3 \cdot \frac{4}{9} + 2 \cdot \frac{1}{2} = \frac{7}{3}$ $4 \cdot \frac{3}{8} + 1 \cdot 0 = \frac{3}{2}$

$N_m \widehat{Q}_{\text{entropy}} =$ $1 \cdot 0 + 4 \cdot 1 = 4$ $2 \cdot 0 + 3 \cdot 0.9183 = 2.7549$ $3 \cdot 0.9183 + 2 \cdot 1 = 4.7549$ $4 \cdot 0.8113 + 1 \cdot 0 = 3.2452$

$N_m \widehat{Q}_{\text{misclass}} =$ $1 \cdot 0 + 4 \cdot \frac{1}{2} = 2$ $2 \cdot \frac{1}{2} + 3 \cdot \frac{1}{3} = 2$ $3 \cdot \frac{1}{3} + 2 \cdot \frac{1}{2} = 2$ $4 \cdot \frac{1}{4} + 1 \cdot 0 = 1$

$N_m \widehat{Q}_{\text{Gini}} =$ $1 \cdot 0 + 4 \cdot \frac{1}{2} = 4$ $2 \cdot \frac{1}{2} + 3 \cdot \frac{4}{9} = \frac{7}{3}$ $3 \cdot \frac{4}{9} + 2 \cdot \frac{1}{2} = \frac{3}{2}$ $4 \cdot \frac{3}{8} + 1 \cdot 0 = \frac{3}{4}$

$N_m \widehat{Q}_{\text{entropy}} =$ $1 \cdot 0 + 4 \cdot 1 = 4$ $2 \cdot 1 + 3 \cdot 0.9183 = 4.7549$ $3 \cdot 0.9183 + 2 \cdot 1 = 4.7549$ $4 \cdot 0.8113 + 1 \cdot 0 = 3.2452$

Замечание 13.1 Дадим еще две интерпретации индексу Джини:

- Вместо того, чтобы в листе m классифицировать объект по большинству голосов, мы можем относить объект к классу k с вероятностью p_{mk}.

  В этом случае средняя ошибка на обучающей выборке равна индексу Джини

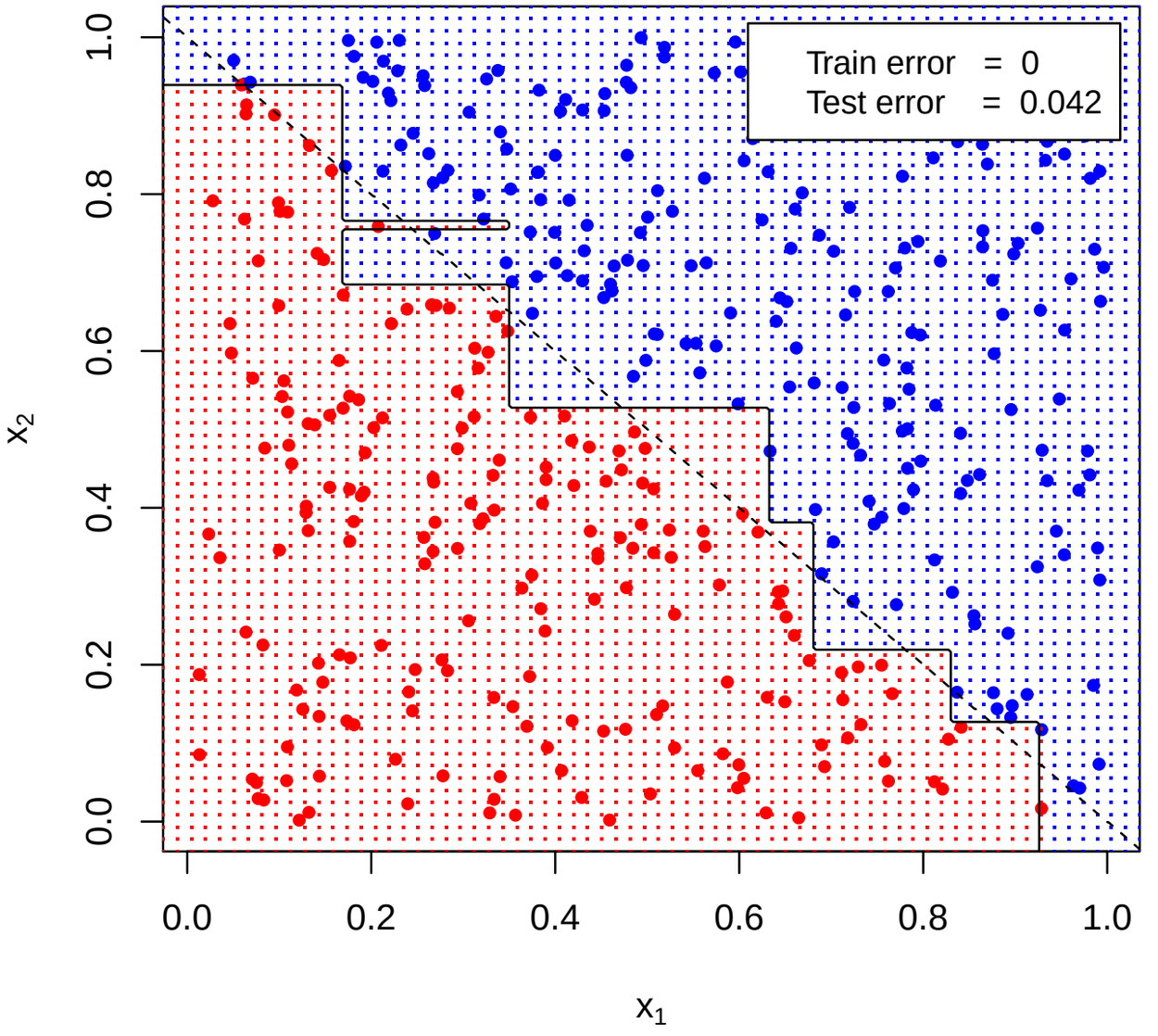
  $$\sum_{k \neq k'} p_{mk} p_{mk'}.$$

  Разумеется, средняя ошибка на обучающей выборке является аппроксимацией средней ошибки на тестовой выборке (для объектов, попадающих в R_m)

- Кодируем единицей объекты k-го класса и нулем — все остальные.
  Тогда выборочная дисперсия этой случайной величины в R_m равна $p_{mk}(1 - p_{mk})$.

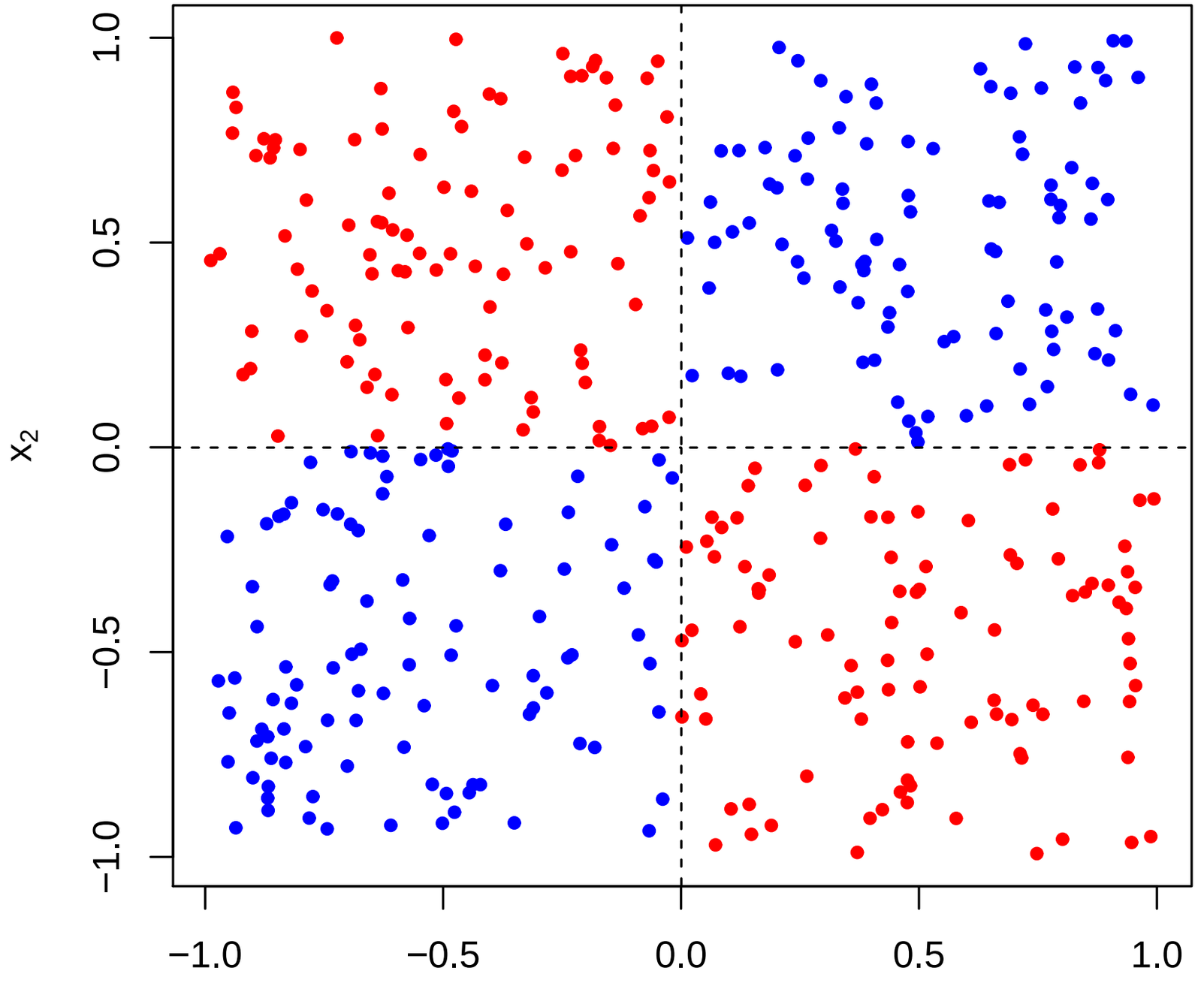
  Сумма по все классам k снова дает индекс Джини.

Высота $= 6$

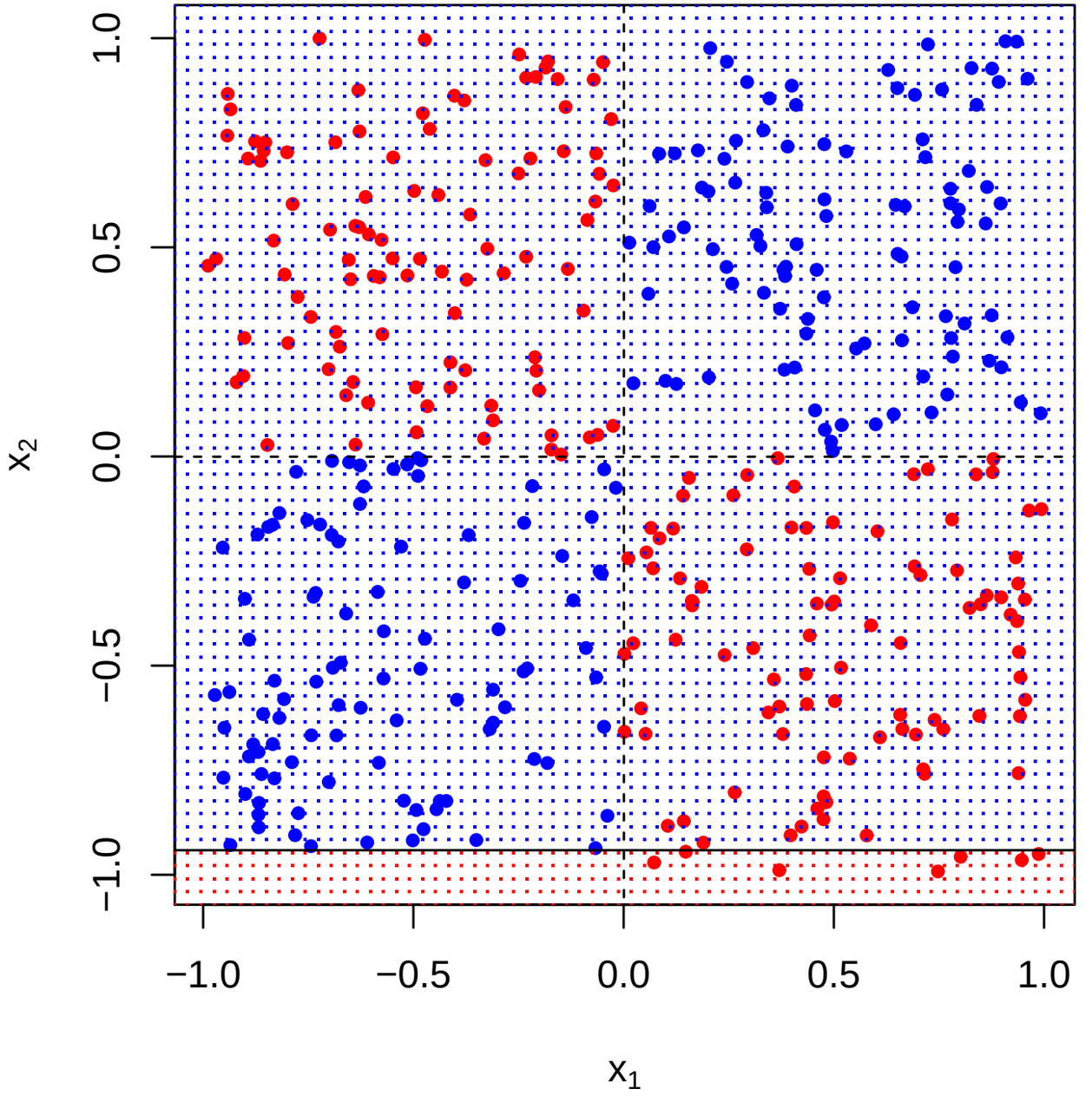


У алгоритм CART проблемы с некоторыми простыми распределениями.

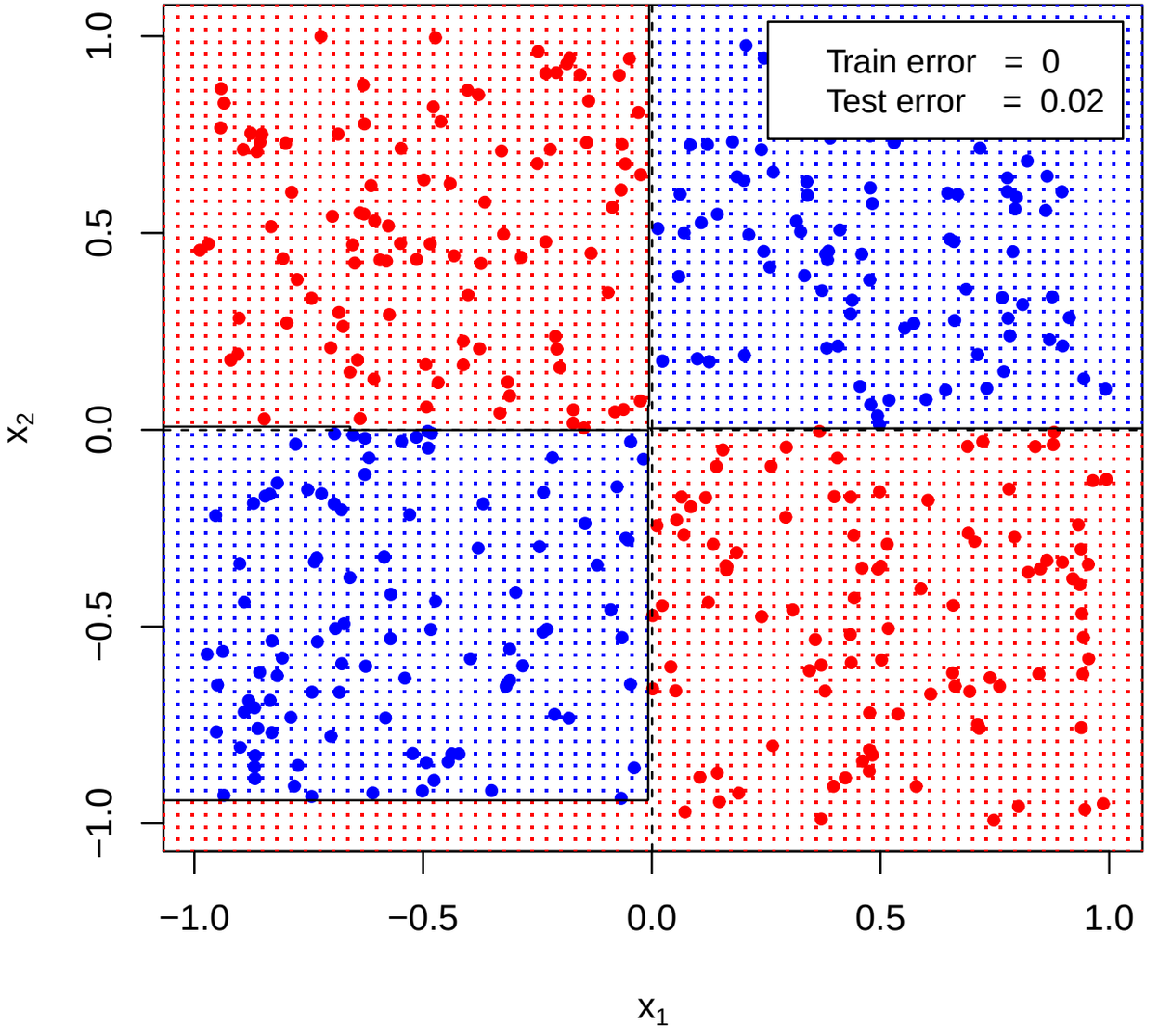
Например, с функцией xor, хотя для нее можно построить хорошее дерево решений.


1.0
0.5
0.0
−0.5
−1.0
X_2
−1.0
−0.5
0.0
0.5
1.0

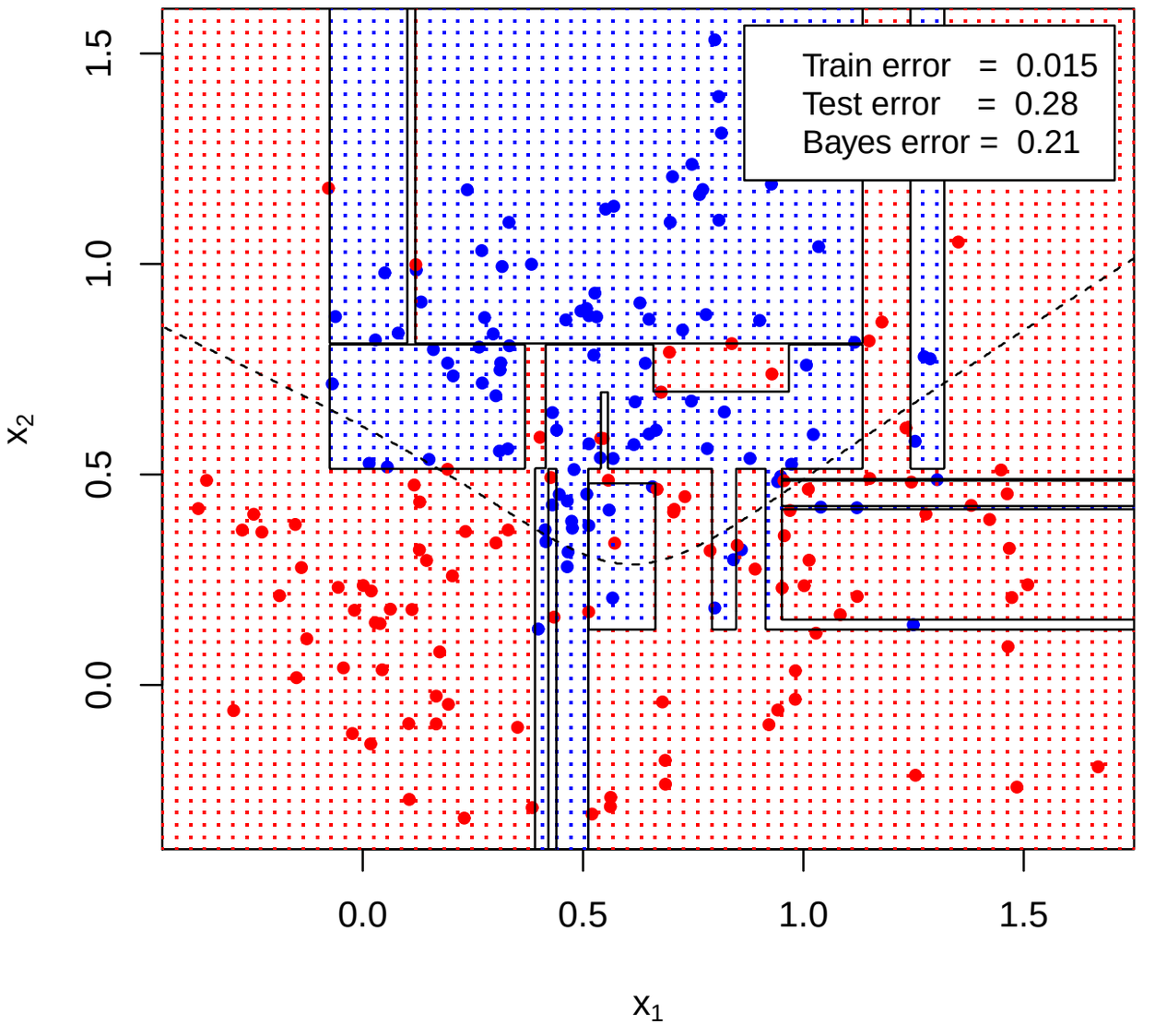
Высота $= 2, 3, 4$


1.0
0.5
0.0
−0.5
−1.0
x_2
−1.0
−0.5
0.0
0.5
1.0
x_1

Высота $= 5$



Высота $= 10$



— переобучение

13.3. Обрезка деревьев

Обрезка, или стрижка, деревьев (pruning) — боремся с переобучением.

$T' \subseteq T \Leftrightarrow$ дерево T' получается из T *отсечениями* (выбираем неконцевую вершину и удаляем оба ее поддерева)

$$Q_\alpha(T) = Q(T) + \alpha \cdot |T|$$

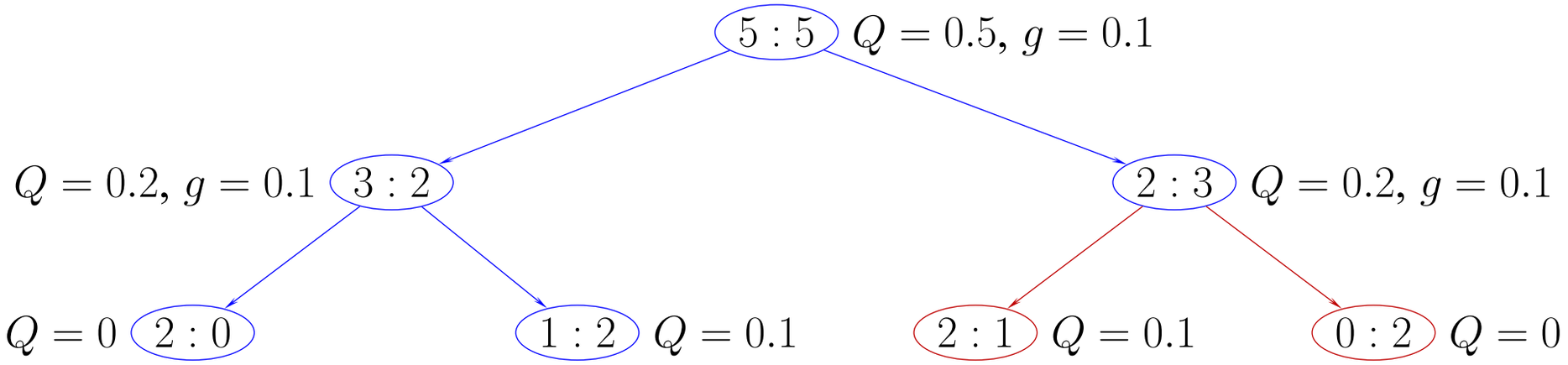
$|T|$ — число листьев в дереве T

Минимизируем $Q_\alpha(T')$ на множестве всех поддеревьев T', получаемых из T отсечениями.

Для любого α существует единственное тупиковое минимальное дерево $T(\alpha) \subset T$, т. е.

1) на $T(\alpha)$ достигается минимум $Q_\alpha(T)$;
2) из любого другого дерева, на котором достигается минимум $Q_\alpha(T)$, отсечениями можно получить $T(\alpha)$.

Найдем $0 = \alpha_0 < \alpha_1 < \alpha_2 < \cdots < \alpha_s, \qquad T \supset T_1 \supset T_2 \supset \cdots \supset T_s,$
для которых $T_i = T(\alpha)$, где $\alpha_{i-1} < \alpha \leq \alpha_i$.



T — все дерево, T' — синее поддерево.

$$Q_\alpha(T) = 0.2 + 4\alpha, \qquad Q_\alpha(T') = 0.3 + 3\alpha.$$
$$Q_\alpha(T') < Q_\alpha(T) \quad \Leftrightarrow \quad \alpha > 0.1$$

В общем случае (t — вершина, T_t — выходящее из нее поддерево):

$$Q_\alpha(T') < Q_\alpha(T) \quad \Leftrightarrow \quad Q(T') + \alpha|T'| < Q(T) + \alpha|T'| \quad \Leftrightarrow$$
$$\Leftrightarrow \quad Q(t) + \alpha < Q(T_t) + |T_t| \cdot \alpha \quad \Leftrightarrow \quad \alpha > g_T(t) \equiv \frac{Q(t) - Q(T_t)}{|T_t| - 1},$$

так как

$$|T| = |T'| + |T_t| - 1, \qquad Q(T) = Q(T') + Q(T_t) - Q(t).$$

Процедура построения последовательности $T \supset T_1 \supset T_2 \supset \cdots \supset T_s$

begin

$T_0 \leftarrow T$

$\alpha_0 \leftarrow 0$

$k = 0$

while число узлов в T_k больше 1

Для каждого нетерминального узла t дерева T_k вычислить $g_{T_k}(t)$

$\alpha_k \leftarrow \min\limits_t g_{T_k}(t)$ (минимум берется по всем нетерминальным узлам t дерева T_k)

Обойти сверху вниз все узлы t' дерева T_k и обрезать те, в которых $g_{T_k}(t) = \alpha_{k+1}$

Построенное дерево обозначить T_{k+1}

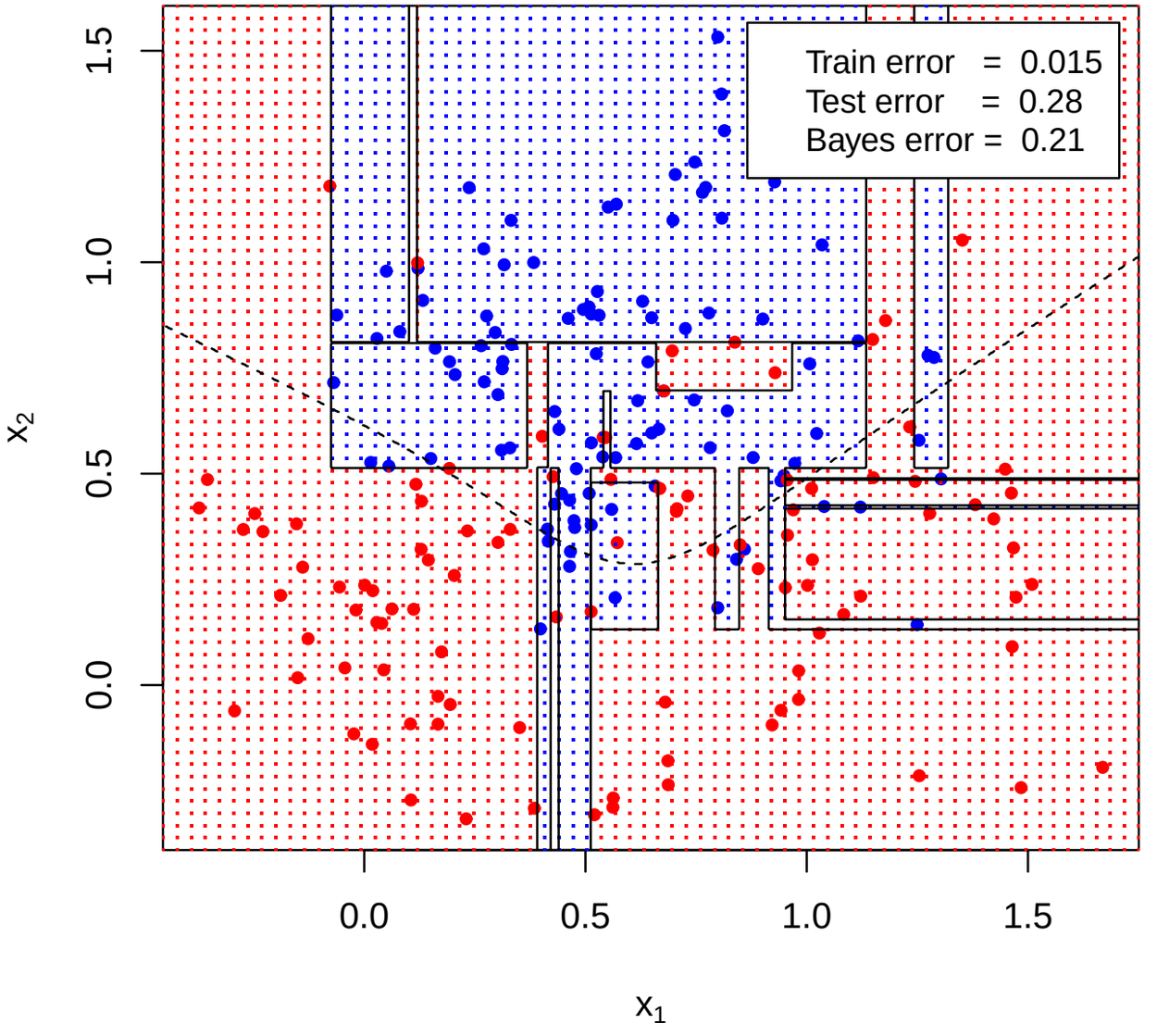
$k \leftarrow k + 1$

end

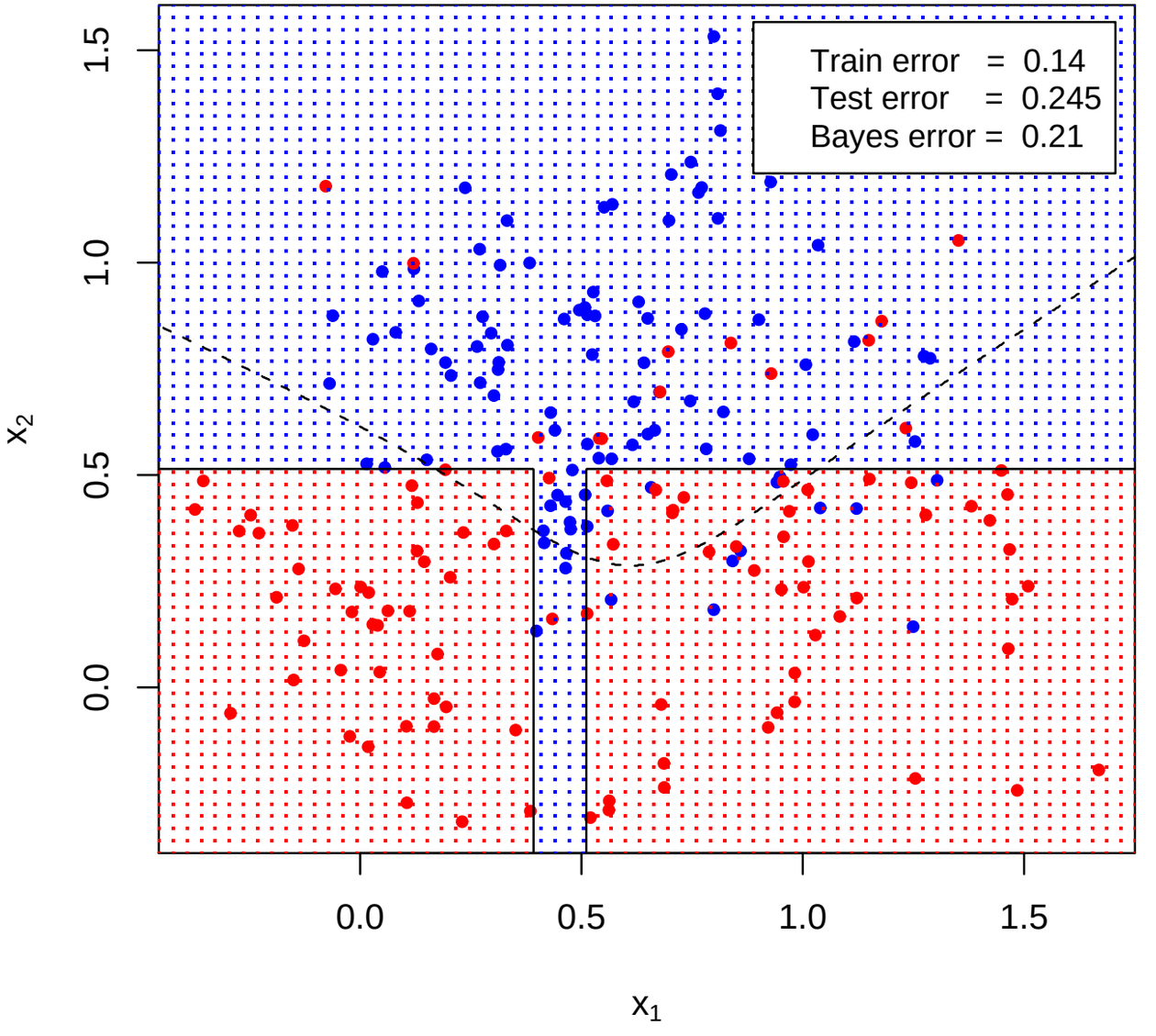
end

Среди всех k выбираем такое, для которого $T(\alpha_k)$ дает наименьшую CV-ошибку.

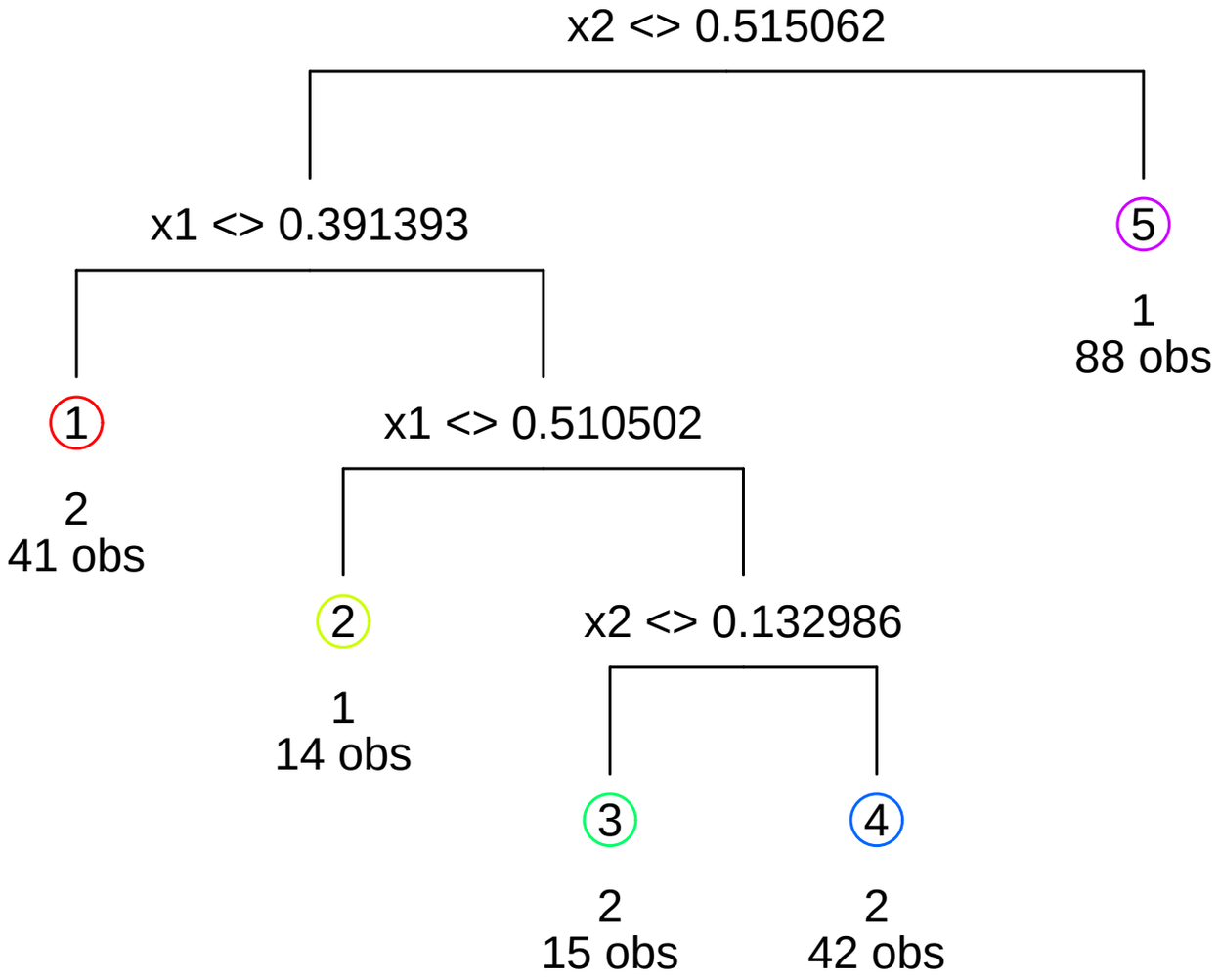
Дерево решений глубины 10 — переобучение



Оптимальное дерево после проведения отсечений — 5 листьев



Оптимальное дерево после проведения отсечений — 5 листьев



Пример `fgl`

По стеклянным осколкам требуется определить их происхождение (B. German).

$N = 214$, $d = 9$, $K = 6$

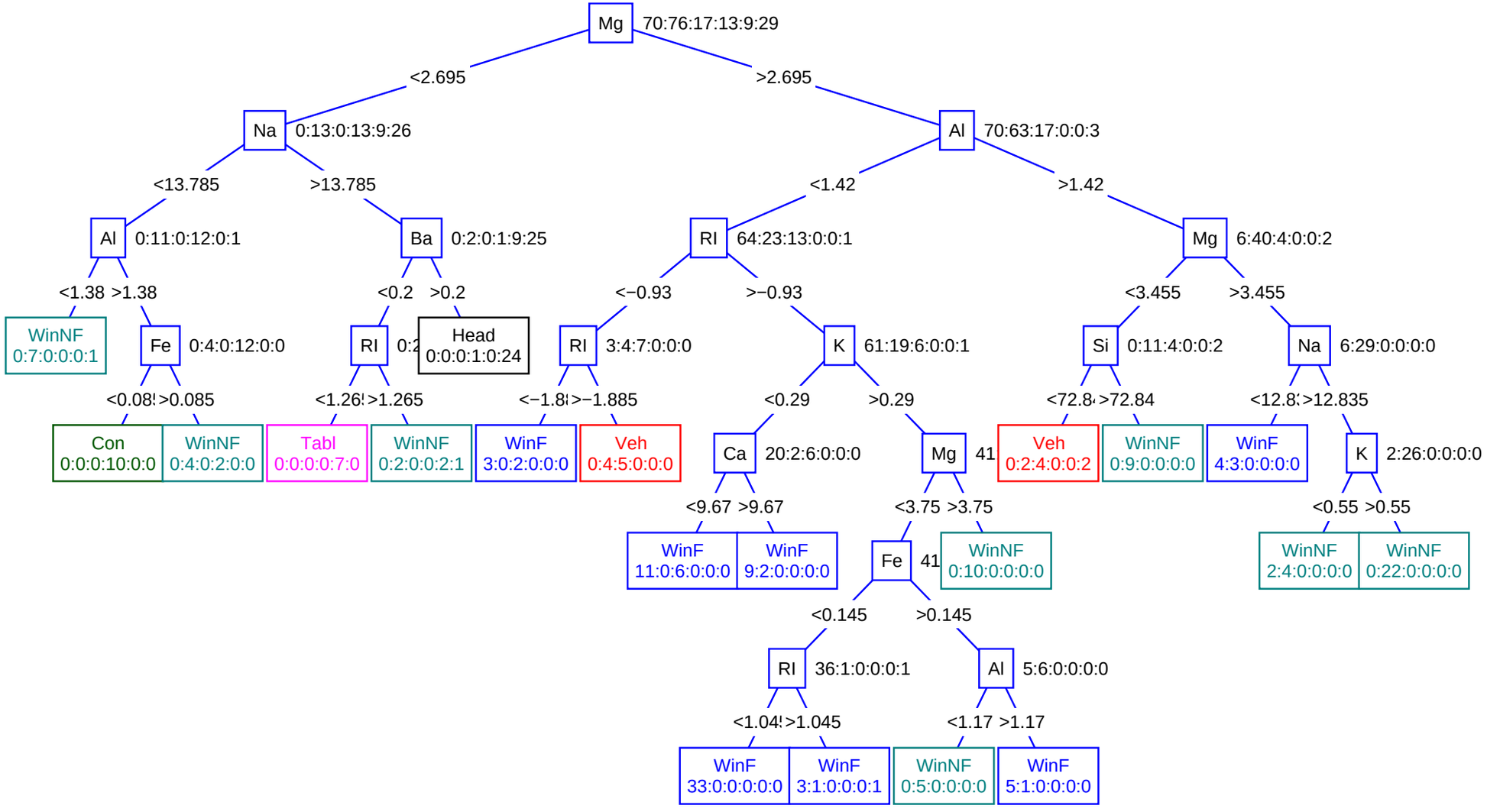
Входные признаки:

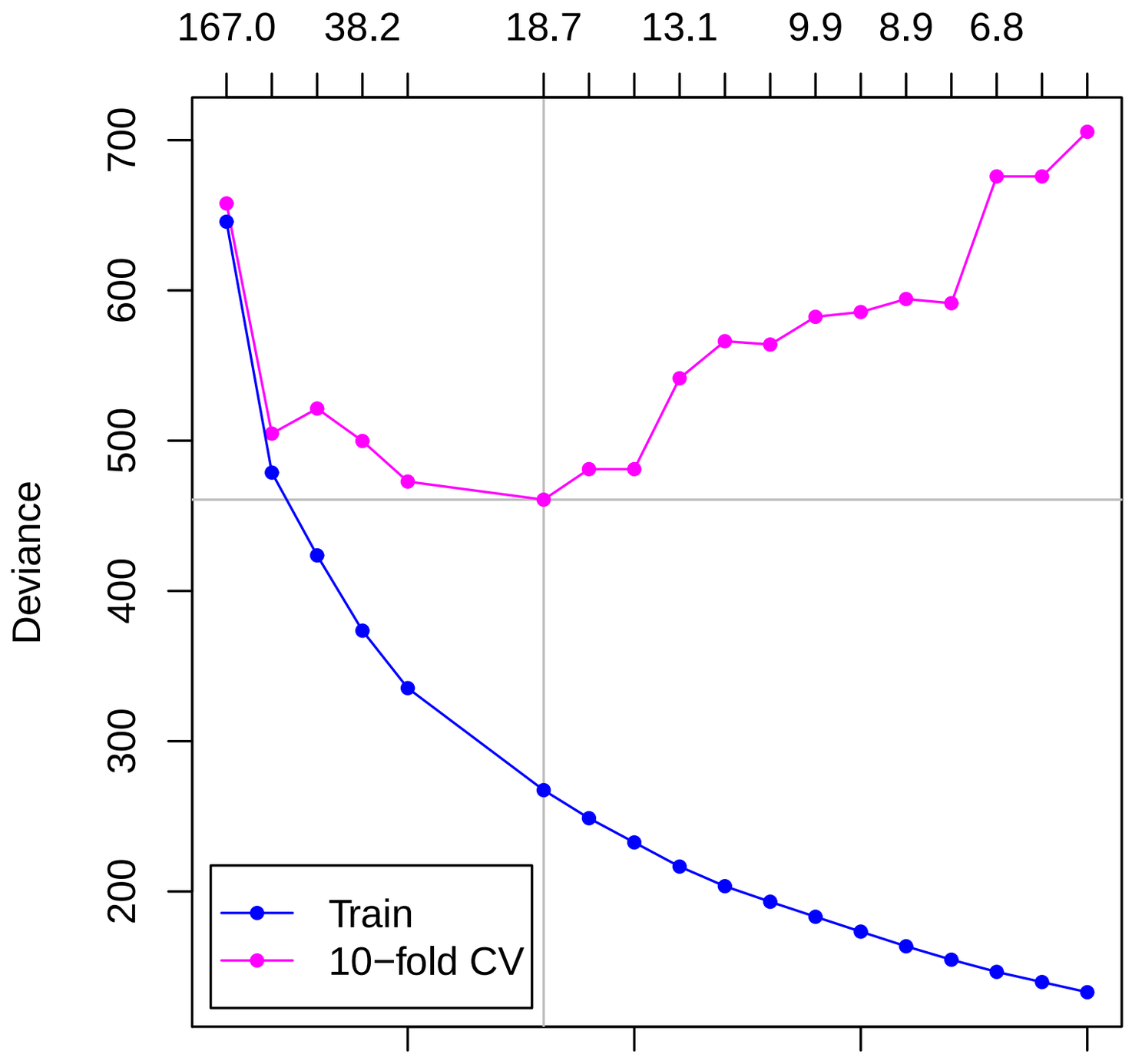
`RI` — показатель преломления
плюс 8 значений процентного содержания оксидов/диоксидов следующих элементов: Na, Mg, Al, Si, K, Ca, Ba, Fe

Классы:

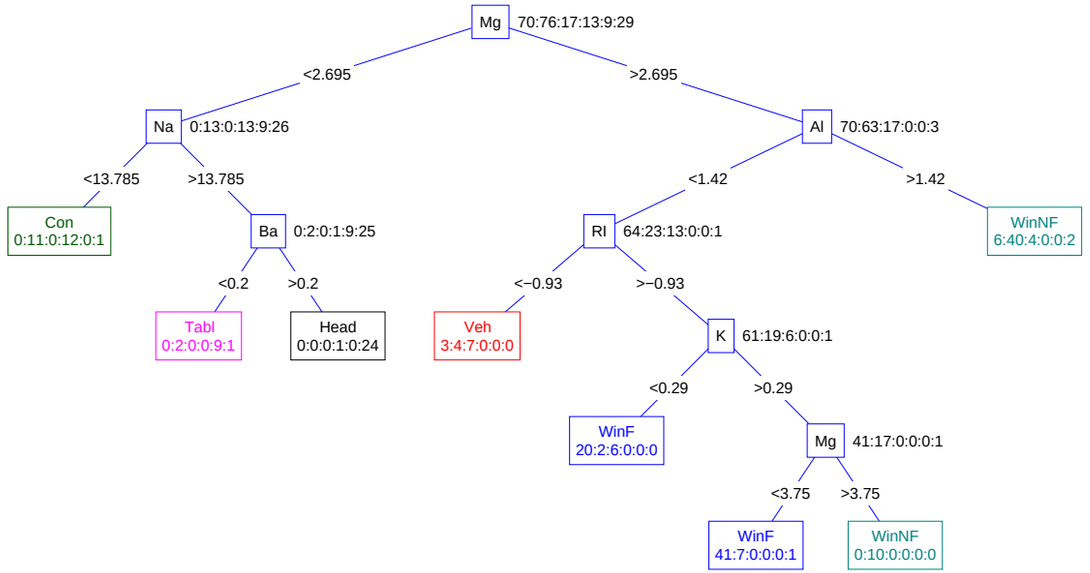
`WinF` — оконное термополированное стекло (флоат-стекло) (70)
`WinNF` — оконное нетермополированное стекло (76)
`Veh` — автомобильные окна (17)
`Con` — сосуд (13)
`Tabl` — посуда (9)
`Head` — автомобильные фары (29)

Мера неоднородности — энтропия.

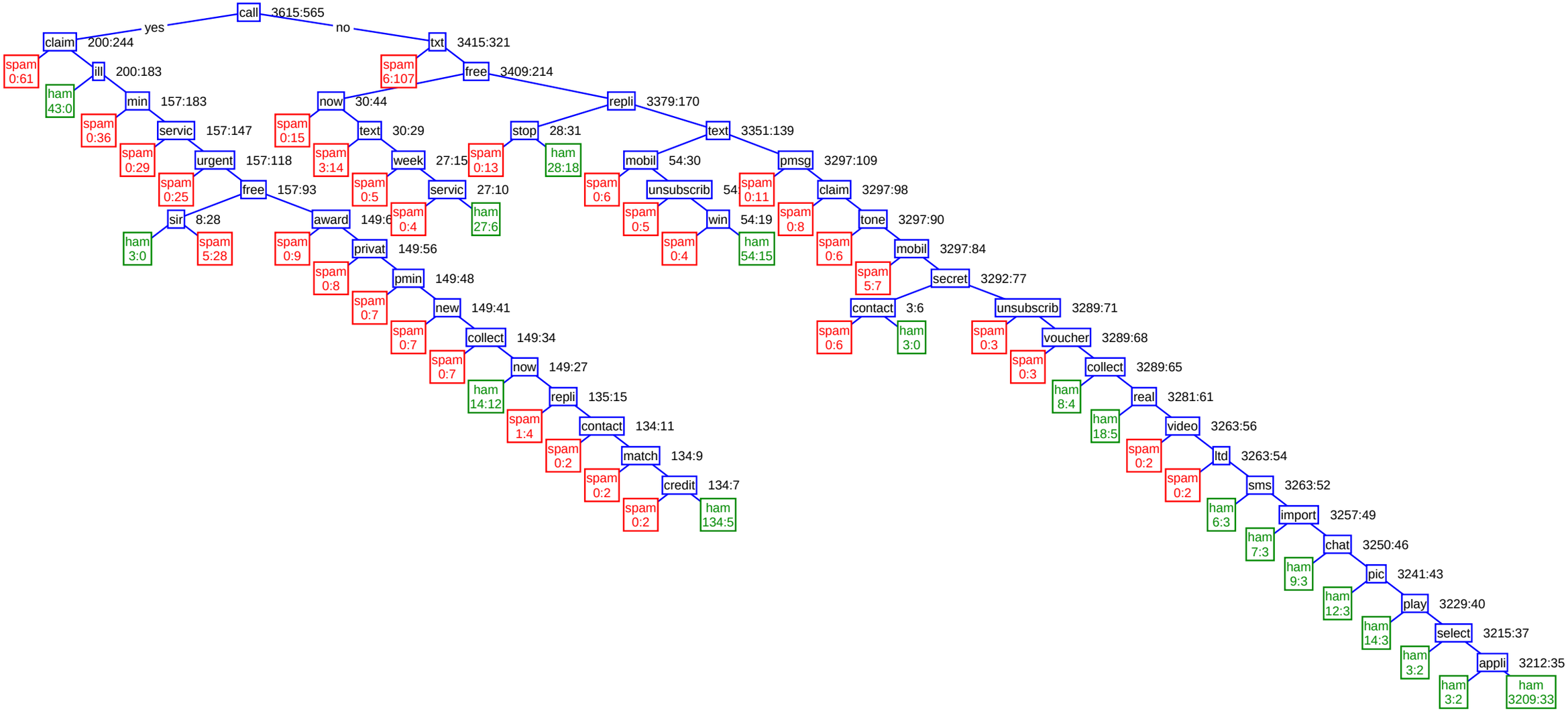

Mg 70:76:17:13:9:29
<2.695
>2.695
Na 0:13:0:13:9:26
Al 70:63:17:0:0:3
<13.785
>13.785
<1.42
>1.42
Al 0:11:0:12:0:1
Ba 0:2:0:1:9:25
RI 64:23:13:0:0:1
Mg 6:40:4:0:0:2
<1.38
>1.38
<0.2
>0.2
<−0.93
>−0.93
<3.455
>3.455
WinNF 0:7:0:0:0:1
Fe 0:4:0:12:0:0
RI 0:2
Head 0:0:0:1:0:24
RI 3:4:7:0:0:0
K 61:19:6:0:0:1
Si 0:11:4:0:0:2
Na 6:29:0:0:0:0
<0.08
>0.085
<1.26
>1.265
<−1.8
>−1.885
<0.29
>0.29
<72.8
>72.84
<12.8
>12.835
Con 0:0:0:10:0:0
WinNF 0:4:0:2:0:0
Tabl 0:0:0:0:7:0
WinNF 0:2:0:0:2:1
WinF 3:0:2:0:0:0
Veh 0:4:5:0:0:0
Ca 20:2:6:0:0:0
Mg 41
Veh 0:2:4:0:0:2
WinNF 0:9:0:0:0:0
WinF 4:3:0:0:0:0
K 2:26:0:0:0:0
<9.67
>9.67
<3.75
>3.75
<0.55
>0.55
WinF 11:0:6:0:0:0
WinF 9:2:0:0:0:0
Fe 41
WinNF 0:10:0:0:0:0
WinNF 2:4:0:0:0:0
WinNF 0:22:0:0:0:0
<0.145
>0.145
RI 36:1:0:0:0:1
Al 5:6:0:0:0:0
<1.04
>1.045
<1.17
>1.17
WinF 33:0:0:0:0:0
WinF 3:1:0:0:0:1
WinNF 0:5:0:0:0:0
WinF 5:1:0:0:0:0

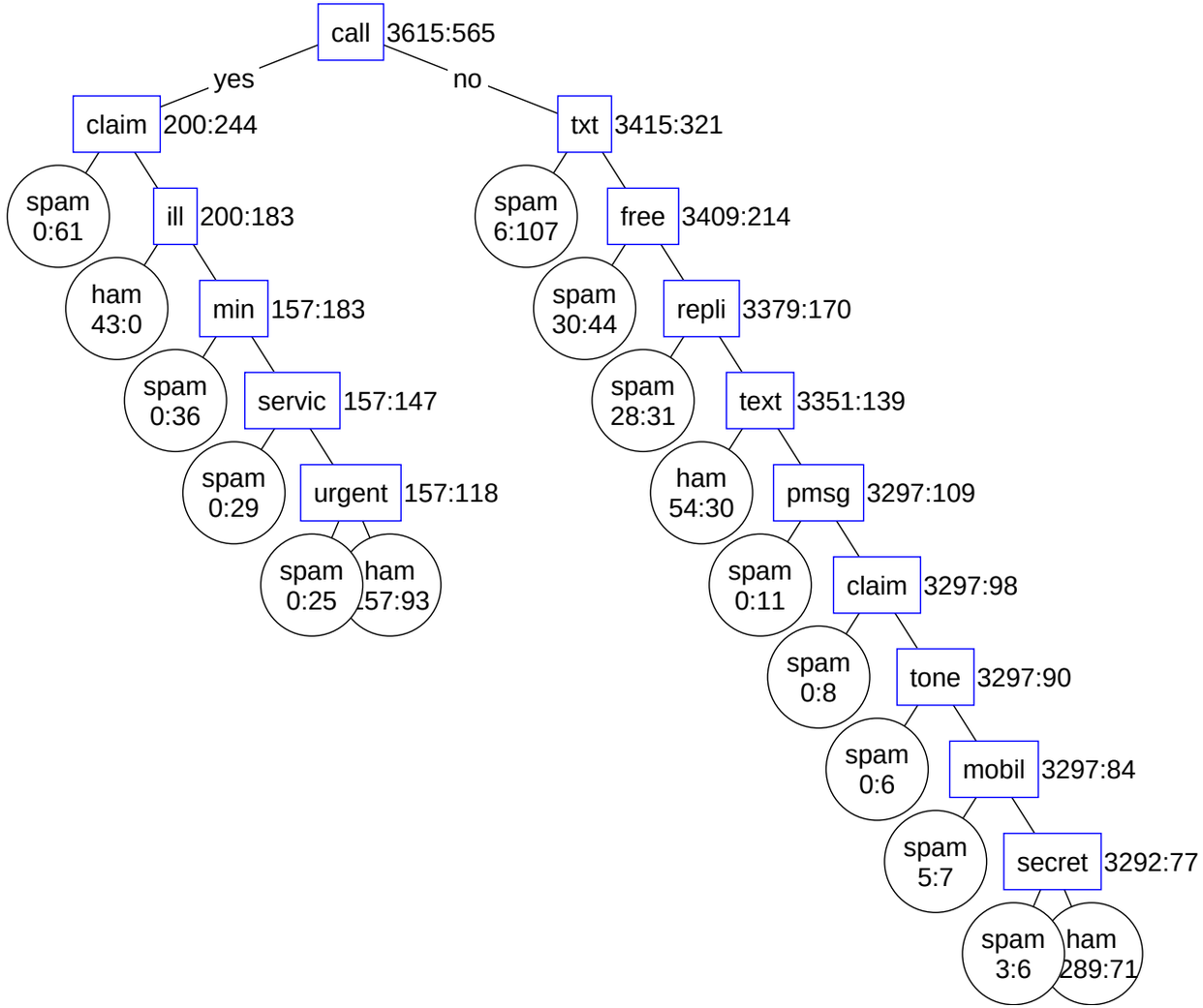

167.0
38.2
18.7
13.1
9.9
8.9
6.8
Deviance
700
600
500
400
300
200
Train
10−fold CV

«Оптимальное» дерево — 8 листьев

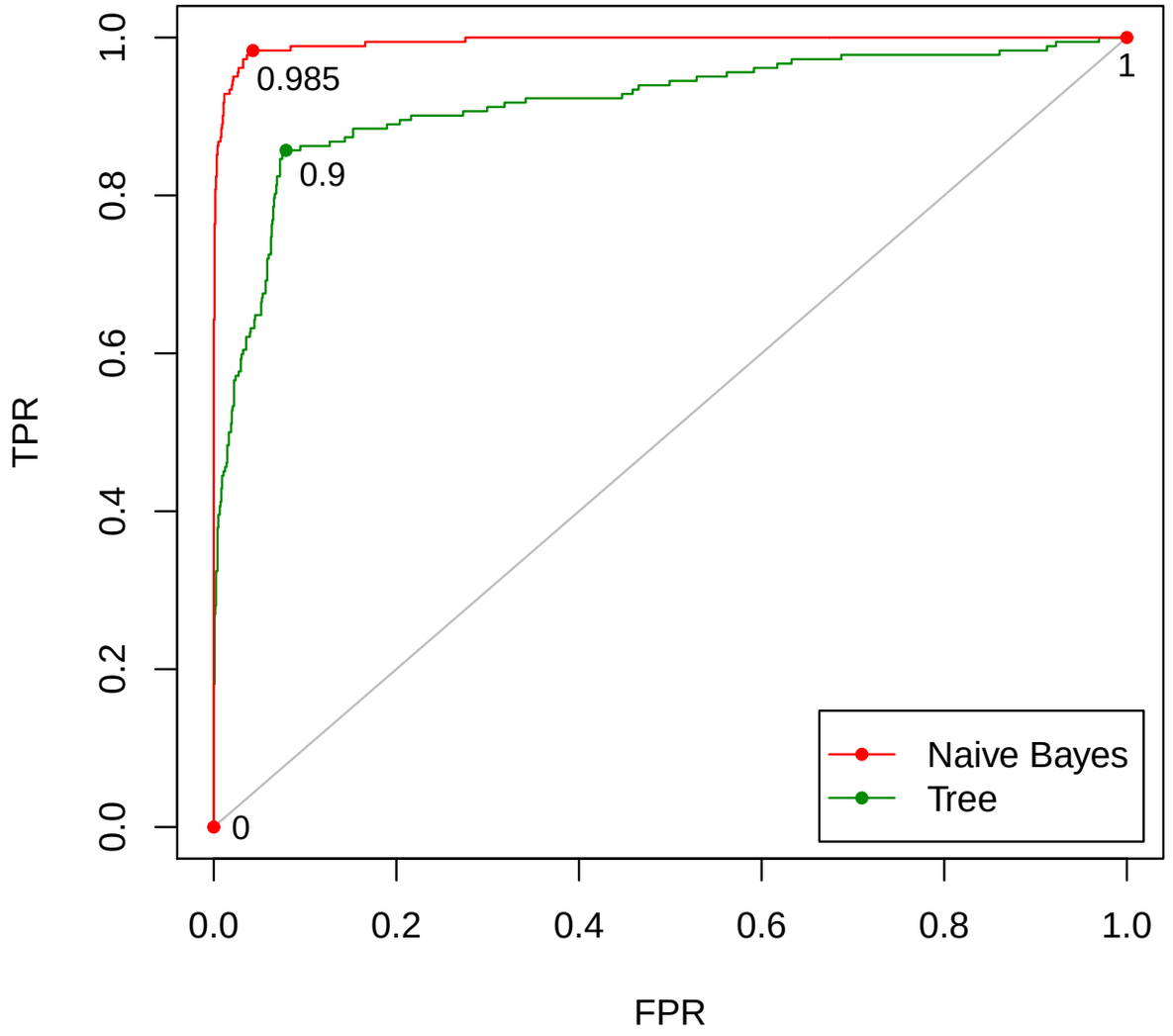


SMS


call 3615:565
yes
no
claim 200:244
txt 3415:321
spam 0:61
ill 200:183
spam 6:107
free 3409:214
ham 43:0
min 157:183
now 30:44
repli 3379:170
spam 0:36
servic 157:147
spam 0:15
text 30:29
stop 28:31
text 3351:139
spam 0:29
urgent 157:118
spam 3:14
week 27:15
spam 0:13
ham 28:18
mobil 54:30
pmsg 3297:109
spam 0:25
free 157:93
spam 0:5
servic 27:10
spam 0:6
unsubscrib 54
spam 0:11
claim 3297:98
sir 8:28
spam 0:4
ham 27:6
spam 0:5
win 54:19
spam 0:8
tone 3297:90
award 149:6
spam 0:4
ham 54:15
ham 3:0
spam 5:28
spam 0:9
privat 149:56
spam 0:6
mobil 3297:84
spam 0:8
pmin 149:48
spam 5:7
secret 3292:77
spam 0:7
new 149:41
contact 3:6
unsubscrib 3289:71
spam 0:7
collect 149:34
spam 0:6
ham 3:0
spam 0:3
voucher 3289:68
spam 0:7
now 149:27
spam 0:3
collect 3289:65
ham 14:12
repli 135:15
ham 8:4
real 3281:61
spam 1:4
contact 134:11
ham 18:5
video 3263:56
spam 0:2
match 134:9
spam 0:2
spam 0:2
ltd 3263:54
spam 0:2
credit 134:7
sms 3263:52
spam 0:2
ham 134:5
ham 6:3
import 3257:49
ham 7:3
chat 3250:46
ham 9:3
pic 3241:43
ham 12:3
play 3229:40
ham 14:3
select 3215:37
ham 3:2
appli 3212:35
ham 3:2
ham 3209:33


call 3615:565
yes
no
claim 200:244
txt 3415:321
spam 0:61
ill 200:183
spam 6:107
free 3409:214
ham 43:0
min 157:183
spam 30:44
repli 3379:170
spam 0:36
servic 157:147
spam 28:31
text 3351:139
spam 0:29
urgent 157:118
ham 54:30
pmsg 3297:109
spam 0:25
ham 57:93
spam 0:11
claim 3297:98
spam 0:8
tone 3297:90
spam 0:6
mobil 3297:84
spam 5:7
secret 3292:77
spam 3:6
ham 289:71

Дерево решений ROC-кривая. tree.test.FPR[tree.threshold.seq == 0.9] = 0.07920792 1 - tree.test.TPR[tree.threshold.seq == 0.9] = 0.1428571 tree.test.Err[tree.threshold.seq == 0.9] = 0.08751793 AUC = 0.916712



13.4. Некоторые отличия C4.5 от CART

- C4.5 для каждого значения номинального признака строит свою ветвь (т. е. деревья *не* бинарные).
- При обрезке используется не перекрестный контроль, а пессимистическая верхняя оценка для биномиального распределения.

13.5. Обрезка в C4.5

Пусть в ящик R_m попало n объектов из обучающей выборки, из них n' — не принадлежат большинству. Таким образом, вероятность ошибки в заданном ящике $\approx n'/n$. Но насколько эта оценка точна? Не будет ли она слишком завышенной («пессимистичной»)?

Пусть p — настоящая вероятность ошибки, тогда

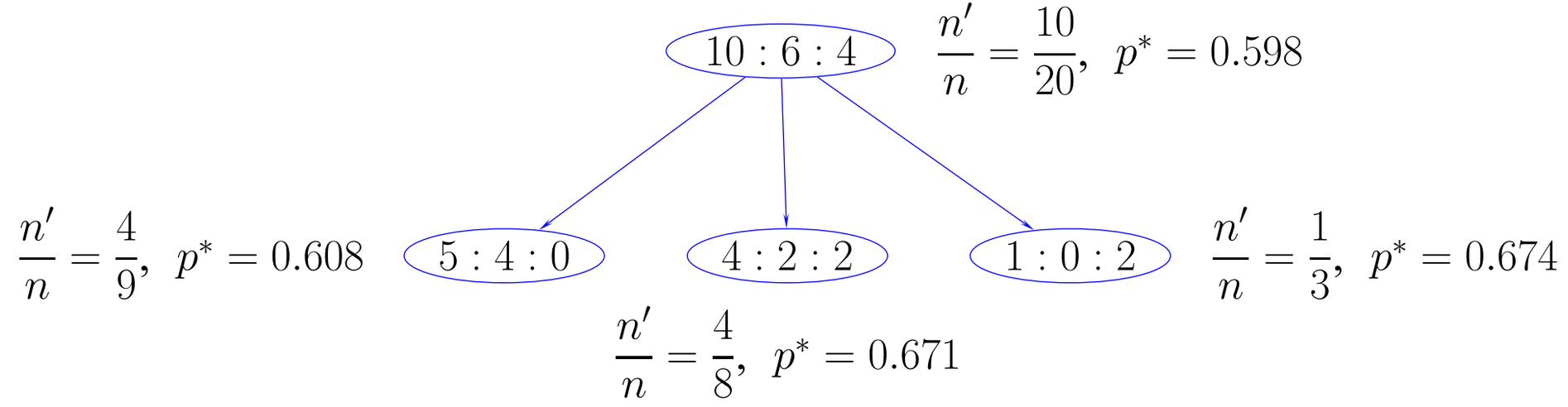
$$\Pr\{\leq n' \text{ объектов не принадлежат большинству}\} = \sum_{i=0}^{n'} \binom{n}{i} p^i (1-p)^{n-i}. \qquad (\star)$$

В качестве верхней оценки p^* для вероятности ошибки берем такое значение p, для которого правая часть равенства $(\star)$ равна выбранному уровню значимости α. Рекомендуется $\alpha = 0.25$.

Т. е. интервал $[0, p^*]$ является доверительным интервалом при оценивании вероятности p с уровнем доверия $1-\alpha$. Иными словами, p^* — верхняя оценка значения вероятности p при уровне доверия $1-\alpha$.

За один проход от листьев к корню удаляются те узлы, взвешенная сумма оценок которых больше оценки для их родителя.

— «Когда не видно разницы, зачем платить больше?»



$$\frac{9}{20} \cdot 0.608 + \frac{8}{20} \cdot 0.671 + \frac{3}{20} \cdot 0.674 > 0.598$$ — все листья отсекаем.

Для вычисления p^* используются аппроксимации (обратной) функции биномиального распределения. Например,

$$p^* = \frac{n' + \frac{1}{2} + \frac{u^2}{2} + u\sqrt{n' + \frac{1}{2} - \frac{1}{n}\left(n' + \frac{1}{2}\right)^2 + \frac{u^2}{4}}}{n + u^2}.$$

[Blyth C.R. Approximate binomial confidence limits // JASA. V. 81, 1986. P. 843–855]

Здесь $u = u_{1-\alpha}$ — квантиль нормального распределения.

Если, например, $\alpha = 0.25$, то $u = u_{0.75} = 0.674$.

13.6. 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

Недостатки:

- *Основной недостаток* — плохая предсказательная (обобщающая) способность.