



Деревья решений и ансамбли моделей

Занятие №7

**Журавлёв Вадим
Ярошенко Ангелина**

Показывать

Ближайшие две недели Весь семестр

Дисциплина

Основы машинного обучения

Тип события

Все типы

Группа

Все группы

30 сентября 18:00 — 21:00 среда Основы машинного обучения Смешанное занятие 1 Уточняется ML-11

7 октября 18:00 — 21:00 среда Основы машинного обучения Смешанное занятие 2 Уточняется ML-11

14 октября 18:00 — 21:00 среда Основы машинного обучения Смешанное занятие 3 Уточняется ML-11

21 октября 18:00 — 21:00 среда Основы машинного обучения Смешанное занятие 4 Уточняется ML-11

28 октября 18:00 — 21:00 среда Основы машинного обучения Смешанное занятие 5 Уточняется ML-11

3 ноября 18:00 — 21:00 вторник Основы машинного обучения Смешанное занятие 6 Уточняется ML-11

11 ноября 18:00 — 21:00 среда Основы машинного обучения Смешанное занятие 7 Уточняется ML-11

сентябрь Пн Вт Ср Чт Пт Сб Вс
1 2 3 4 5 6
7 8 9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30

октябрь Пн Вт Ср Чт Пт Сб Вс
1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

ноябрь Пн Вт Ср Чт Пт Сб Вс
1

Приходя на лекцию

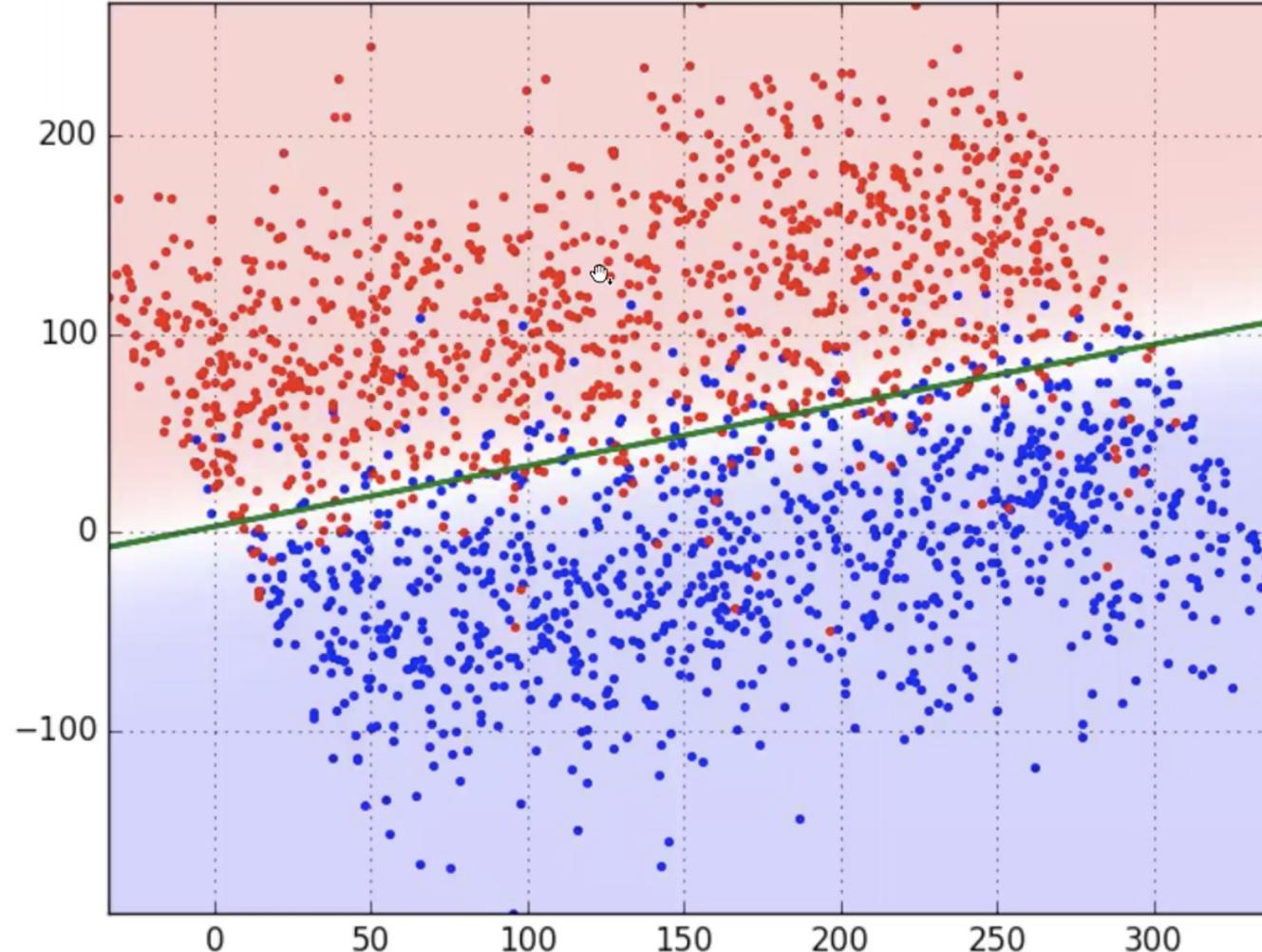


отметиться не
забудь ты

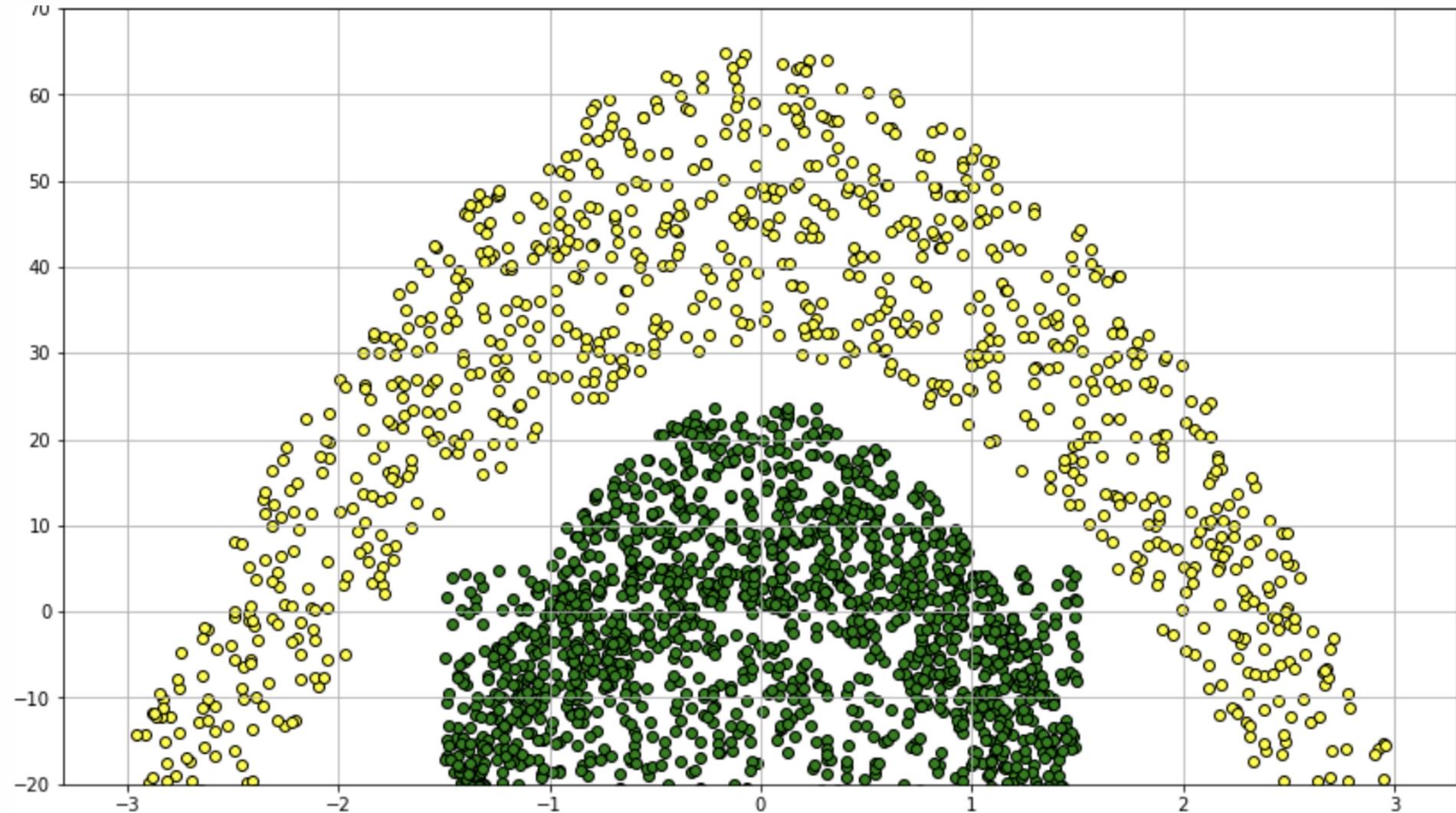
Содержание занятия

1. Деревья, что это такое и с чем их едят
2. Ансамбли моделей и всё-всё-всё

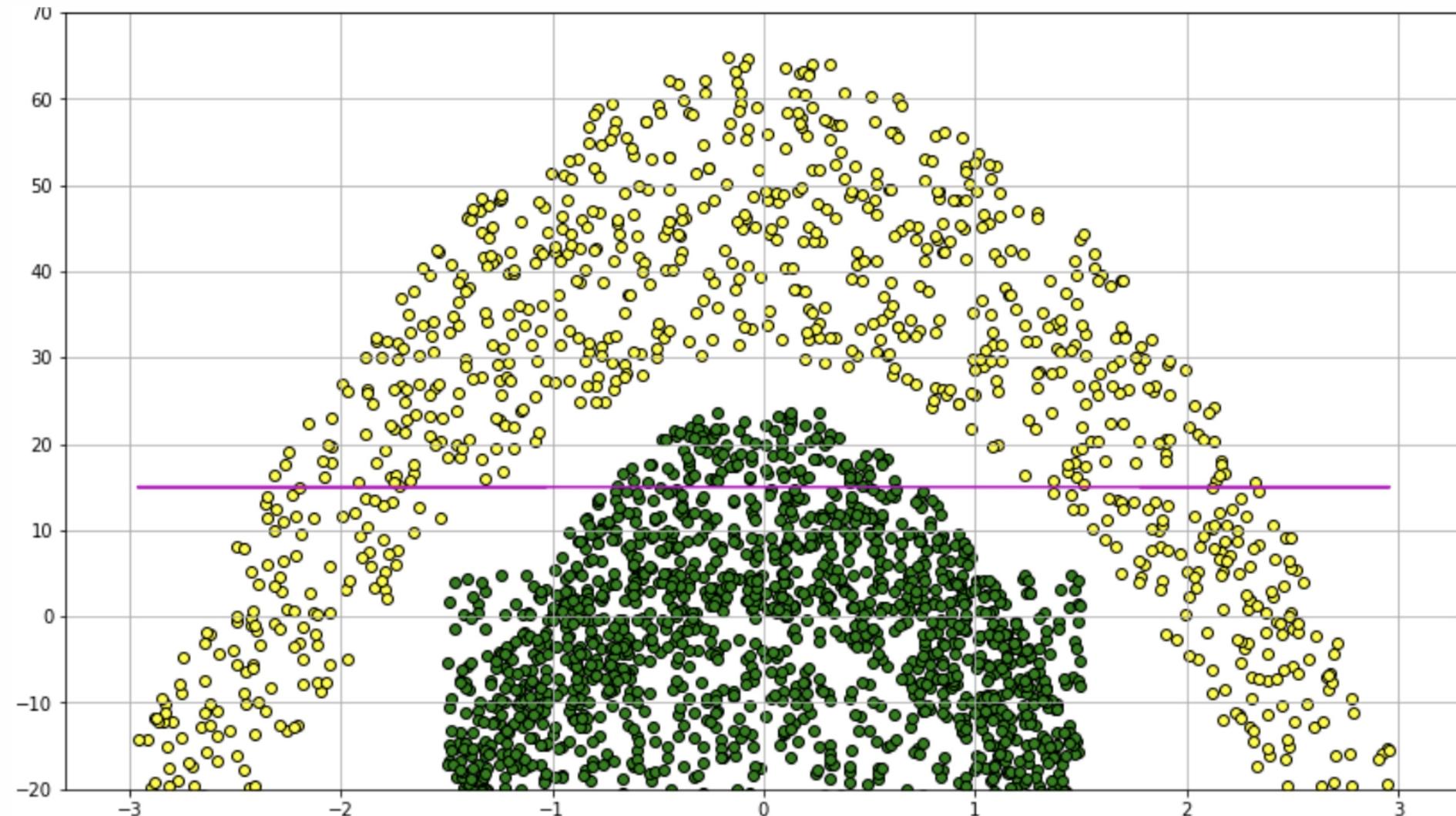
Всегда ли регрессия нас спасает?



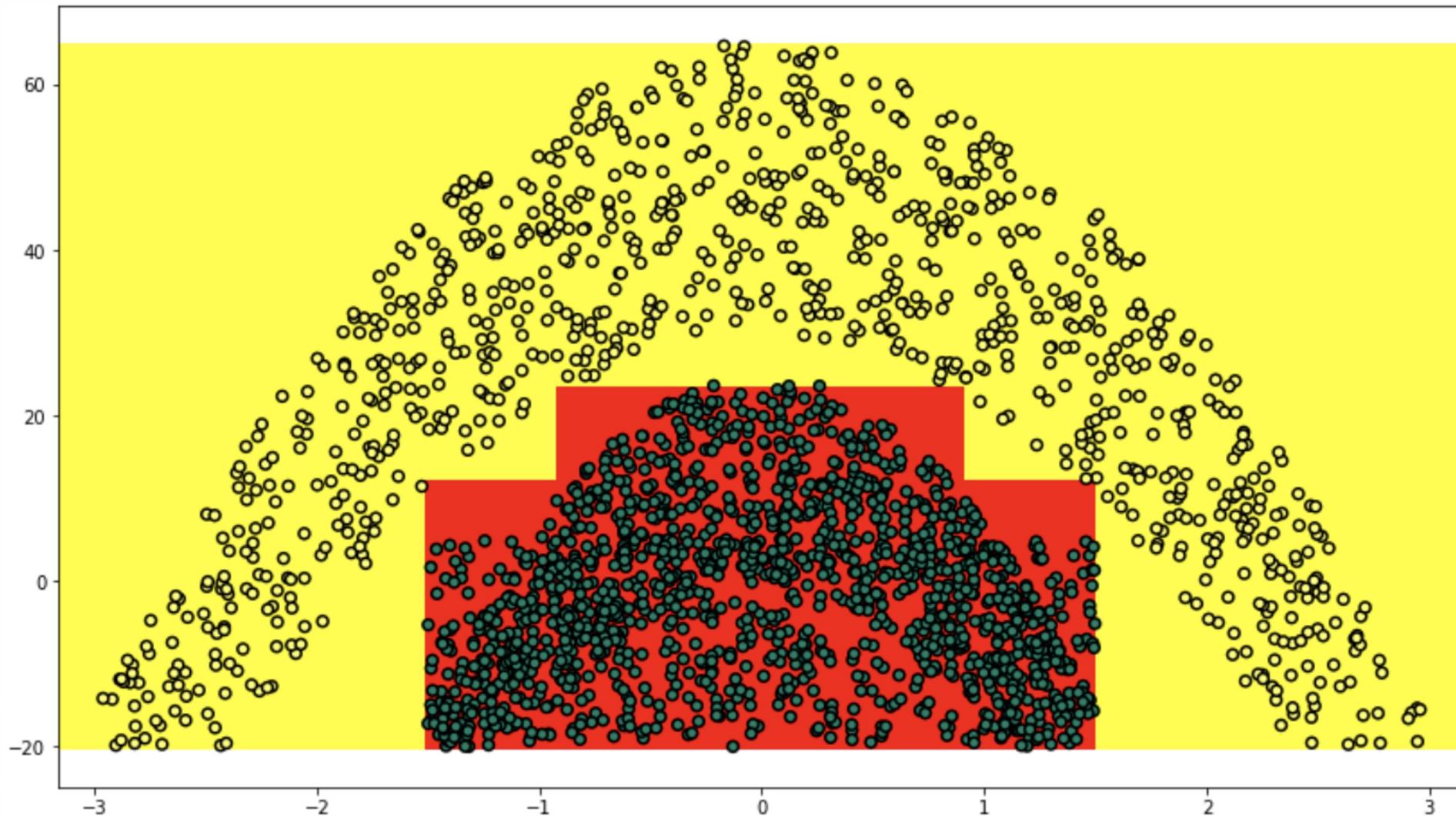
А в таком случае?



А в таком случае?



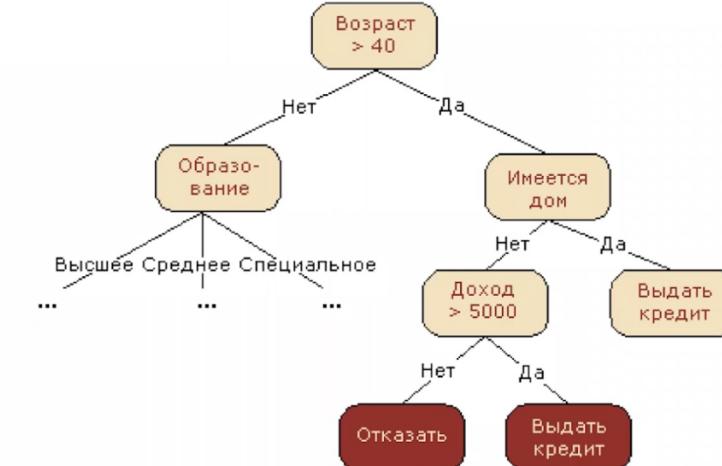
А в таком случае?



Деревья решений

Деревья решений относятся к логическим методам классификации, так как ищут в данных логические закономерности.

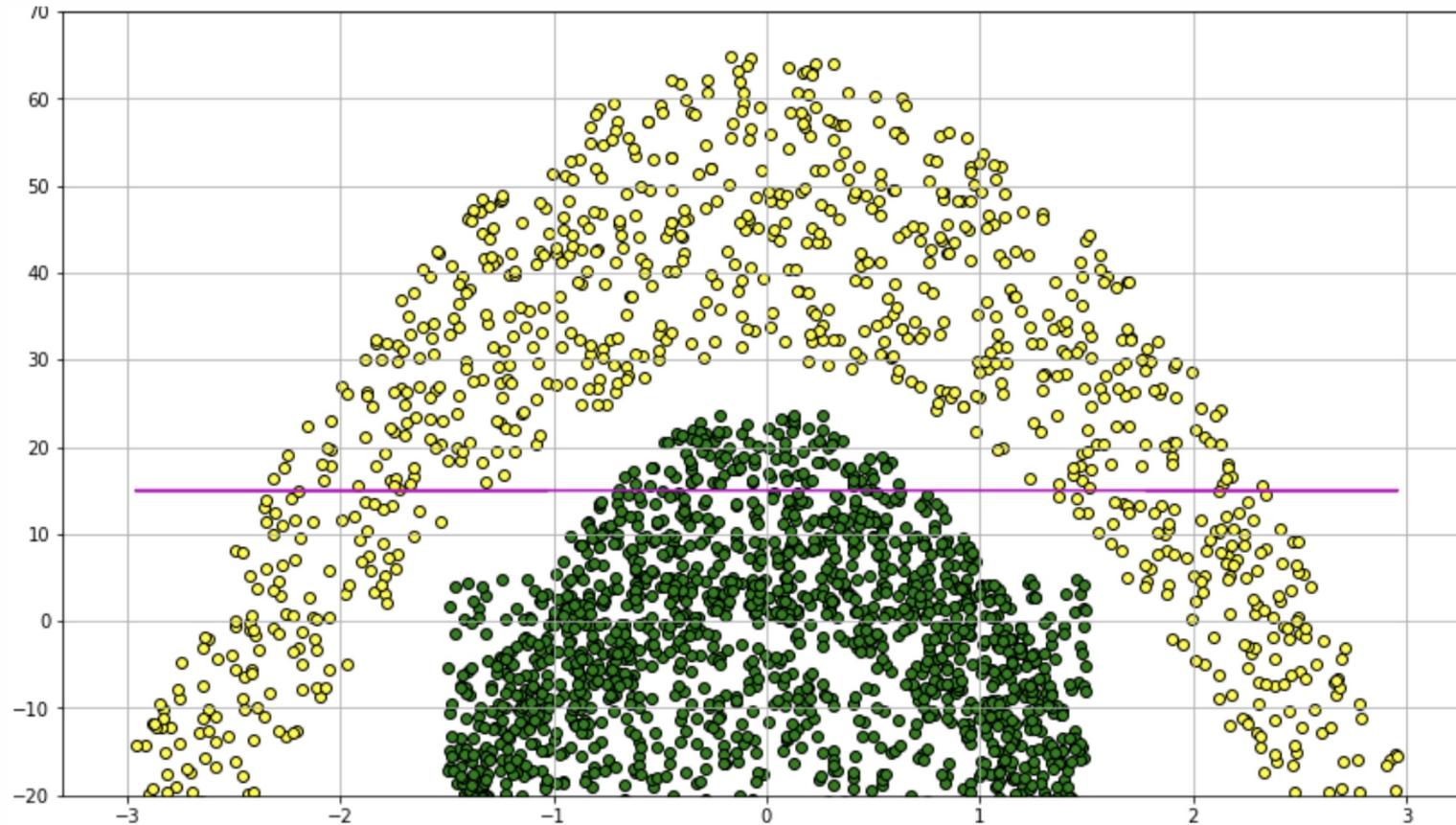
- Температура > 38? Да -> Есть кашель? Да -> Кашель влажный? Да -> Назначать антибиотики
- Возраст > 40? Да -> Имеется дом? Нет -> Доход >5000? Да -> Выдать кредит



В ходе лекции будем рассматривать задачу бинарной классификации, потом обсудим как можно масштабировать

Мотивация 1. Пример нелинейного датасета

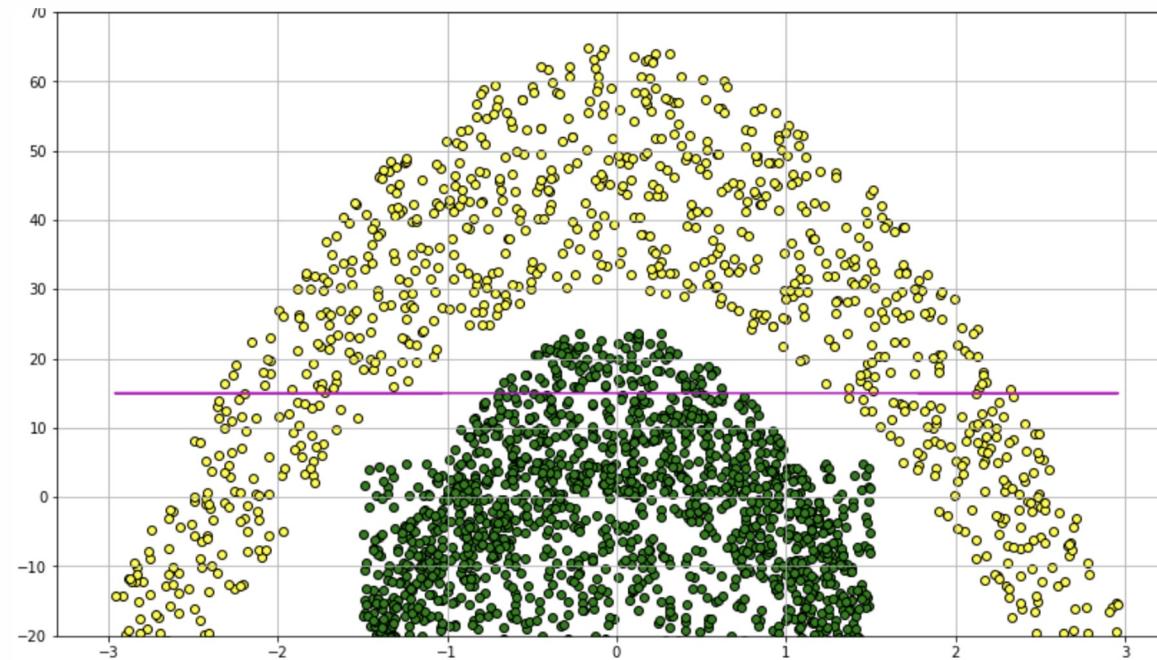
Logistic Regression, f-measure ~0.755



Целевая переменная **нелинейно** зависит от признаков

Мотивация 2. Лог. рег. и наш пример

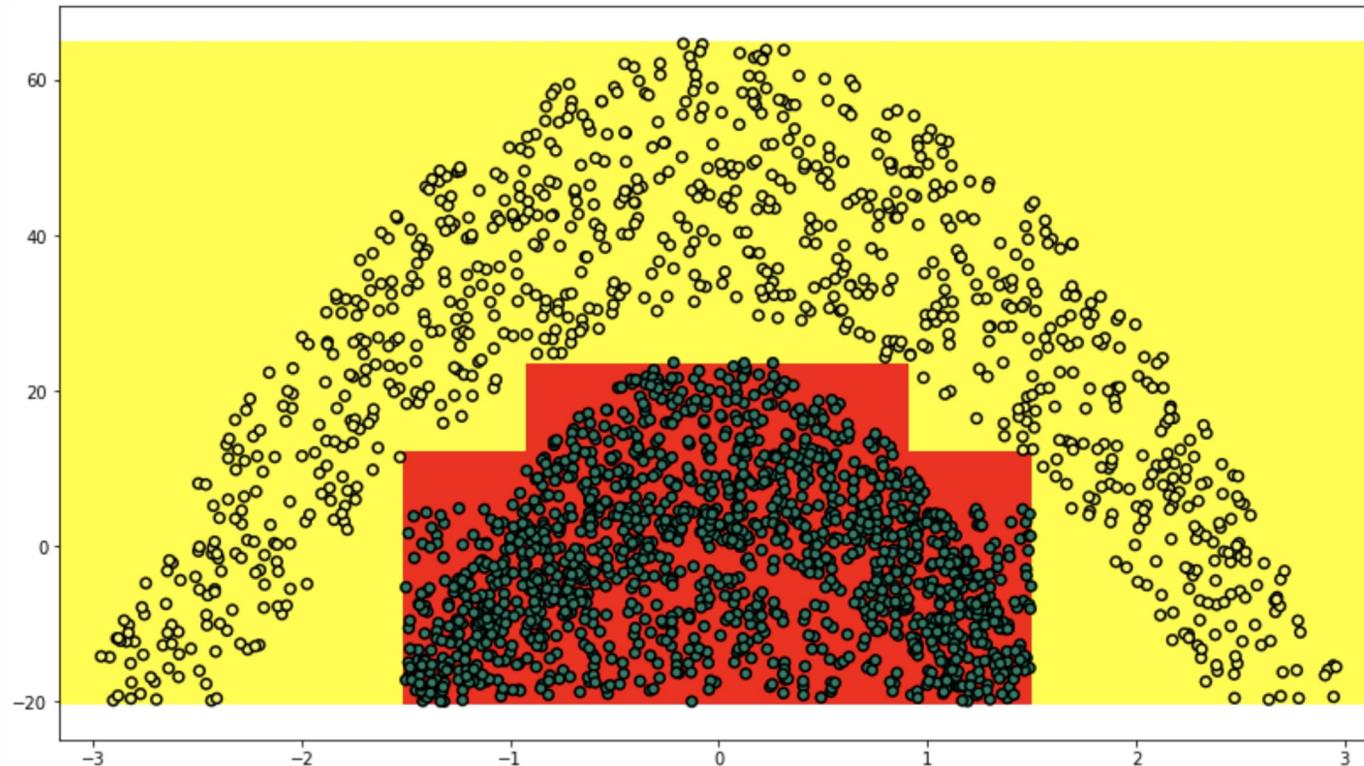
Logistic Regression, f-measure ~0.755



- Лог. рег. хорошо работает при линейной зависимости признаков и целевой переменной
- Экспериментировать с преобразованием признаков и добиться более хорошего качества, но такой подход является эвристиком и вы можете потратить много времени и при этом не получить желаемый результат

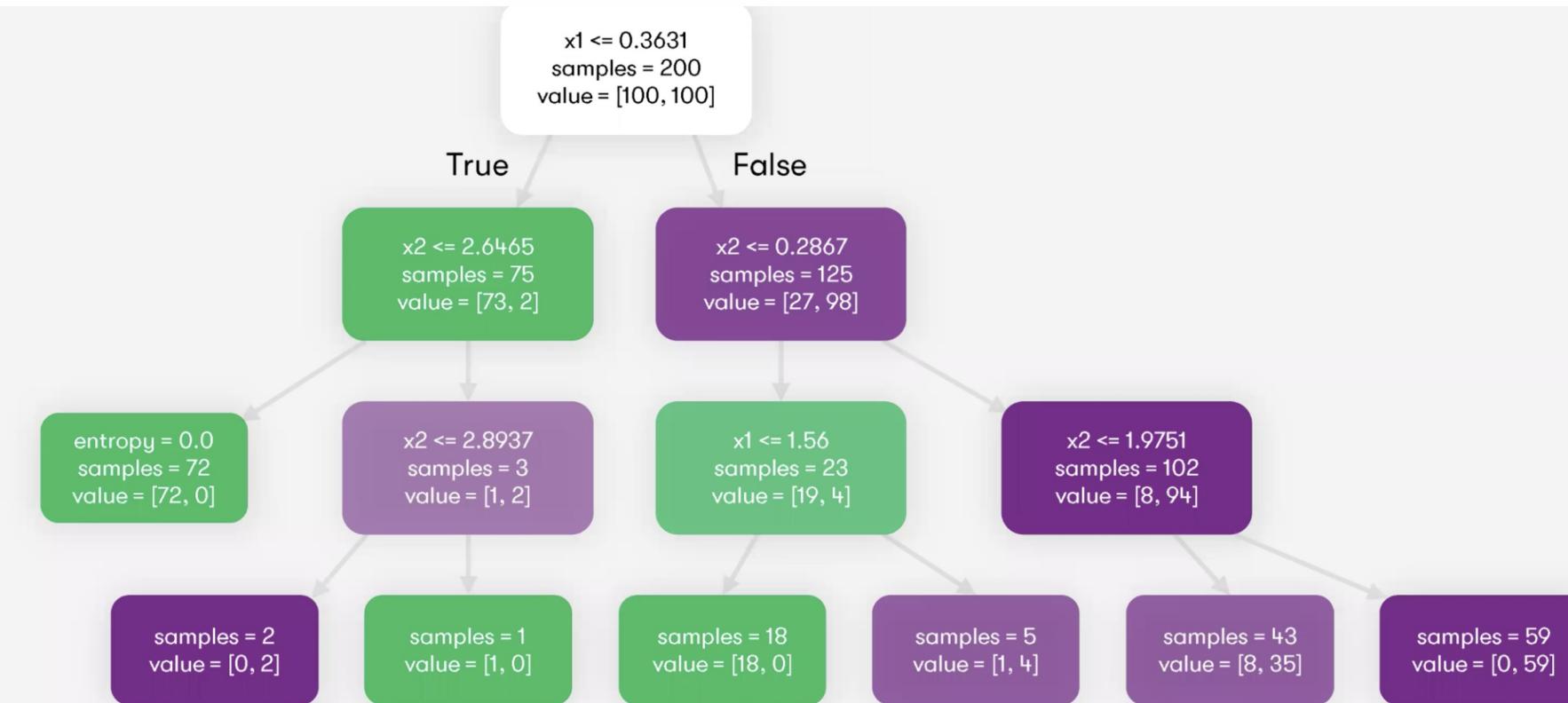
Мотивация 3. Дерево решений и наш пример

Decision Tree, f-measure = 1



- разделяет пространство на многомерные прямоугольники (подпространства)
- в подпространстве формируется ответ на основе обучающей выборки

Мотивация 4. Представление дерева



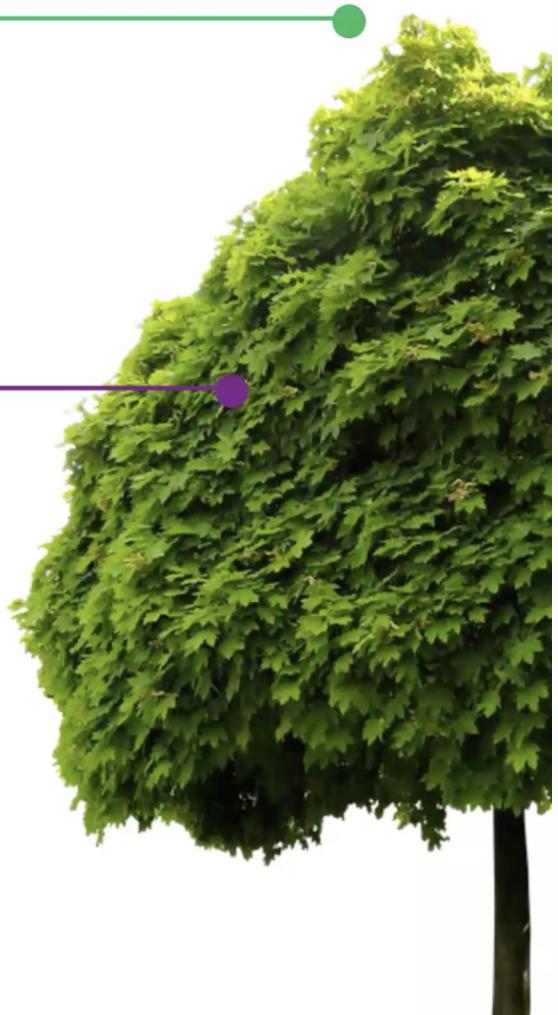
- Последовательность логических правил
- Константа в листьях

Бинарное дерево решений

Вершины - логические правила

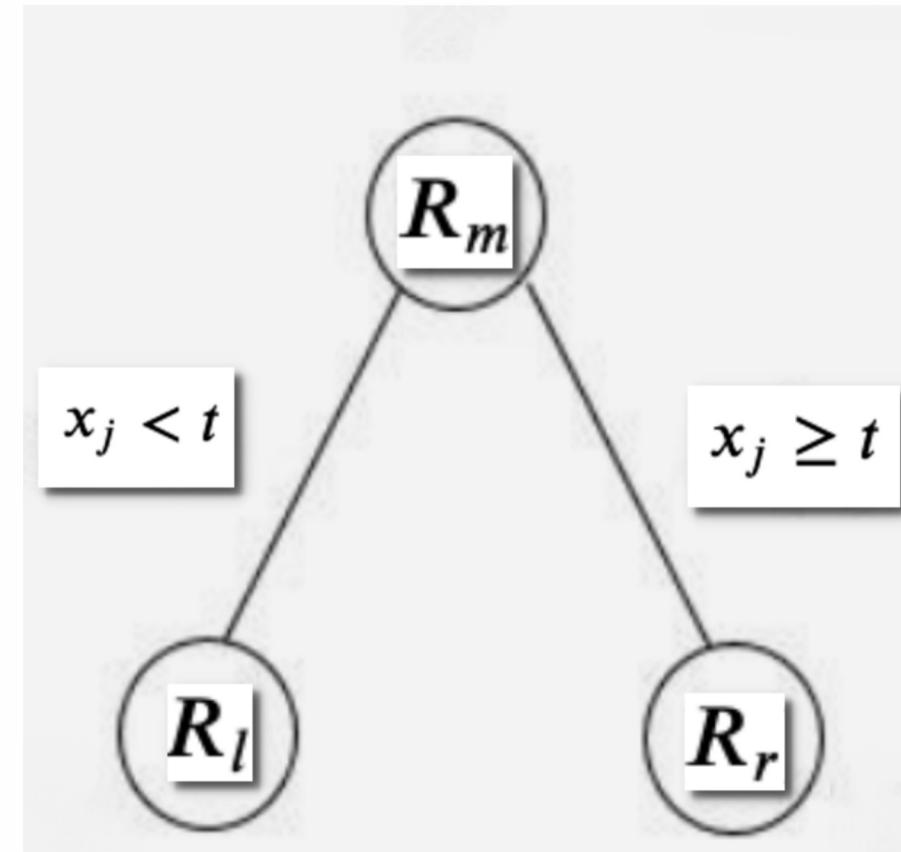
1. Кол-во этажей в доме ≥ 5 ?
2. Квартира студия?

Листья - предсказания в виде константы



Метод построения решающего дерева определяется:

- Предикаты
- Критерий информативности
- Критерий останова
- Обработка пропущенных данных
- Стрижка



Критерии информативности

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

Критерии информативности

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} L(y, c)$$

R_m - выборка в текущей вершине

j - индекс признака

t - порог для признака

$L(y, c)$ - некоторая функция потерь

Критерии информативности

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

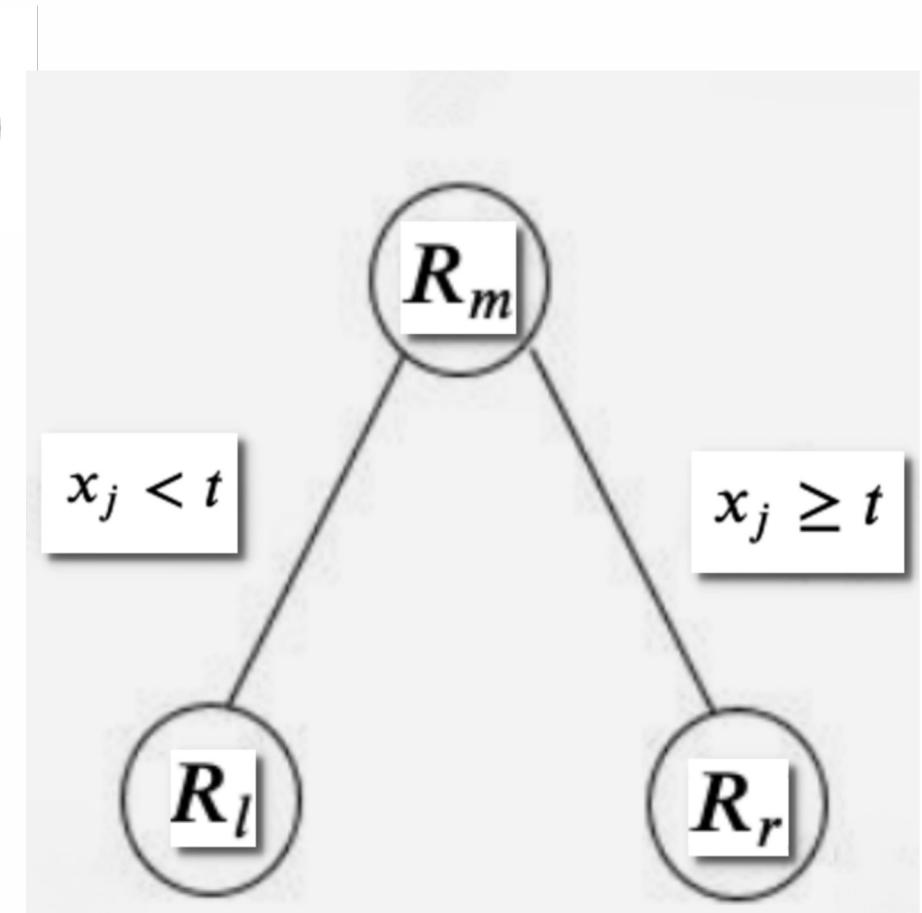
$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} L(y, c)$$

R_m - выборка в текущей вершине

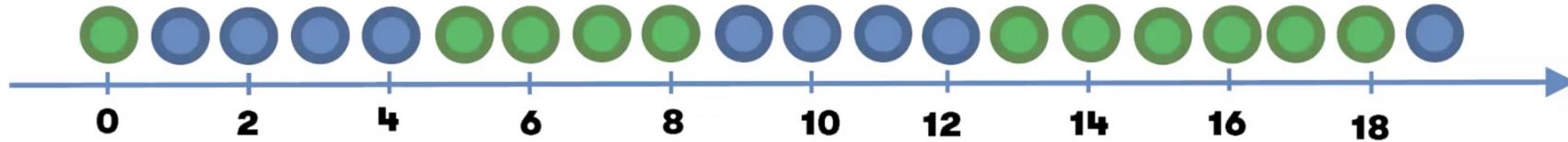
j - индекс признака

t - порог для признака

$L(y, c)$ - некоторая функция потерь



Как выгоднее всего строить дерево?



Как же определить меру беспорядка?

Энтропия Шеннона – мера беспорядка системы:

$$S = - \sum_{i=1}^N p_i \log_2 p_i,$$

В случае бинарной классификации:

$$S = -p_+ \log_2 p_+ - p_- \log_2 p_- = -p_+ \log_2 p_+ - (1 - p_+) \log_2 (1 - p_+);$$

Прирост информации:

$$IG(Q) = S_O - \sum_{i=1}^q \frac{N_i}{N} S_i,$$

Энтропия в наших терминах

$$S_0 = H(R_m) \quad N = |R_m|$$

$$S_1 = H(R_l) \quad N_1 = |R_l|$$

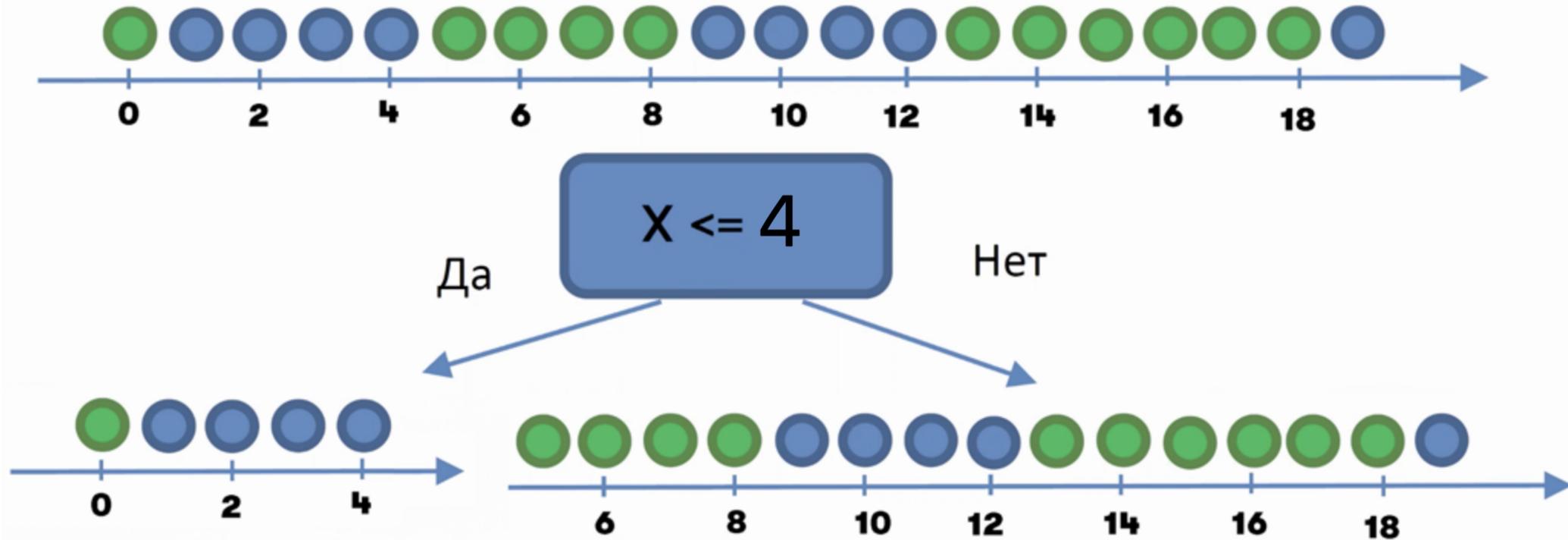
$$S_2 = H(R_r) \quad N_2 = |R_r|$$

$$\mathbf{S} = H(R) = - \sum_{i=1}^K p_i \log_2 p_i$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

#020

Как выгоднее всего строить дерево?

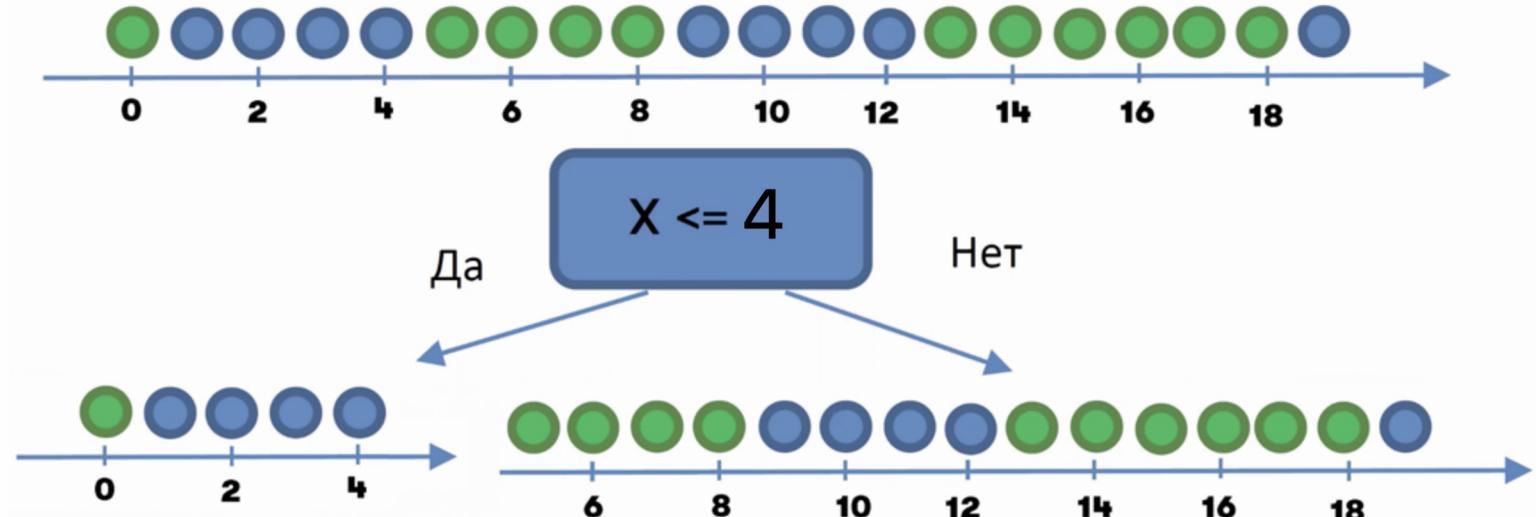


Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = ?, p_c = ?$$



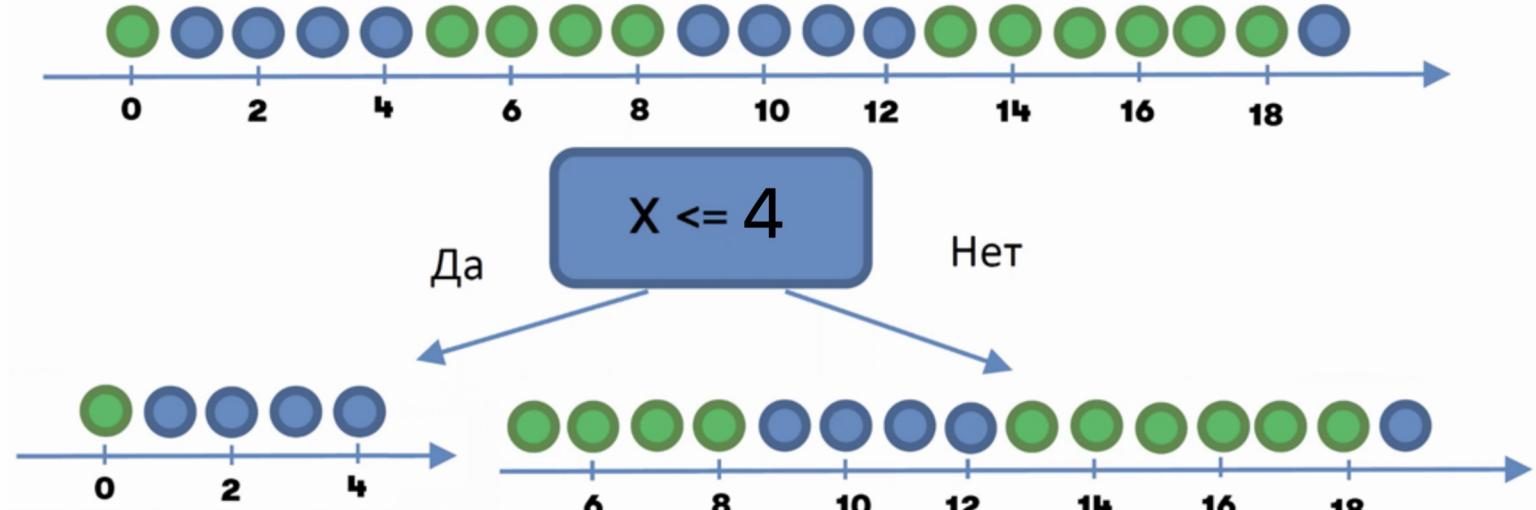
Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i$$



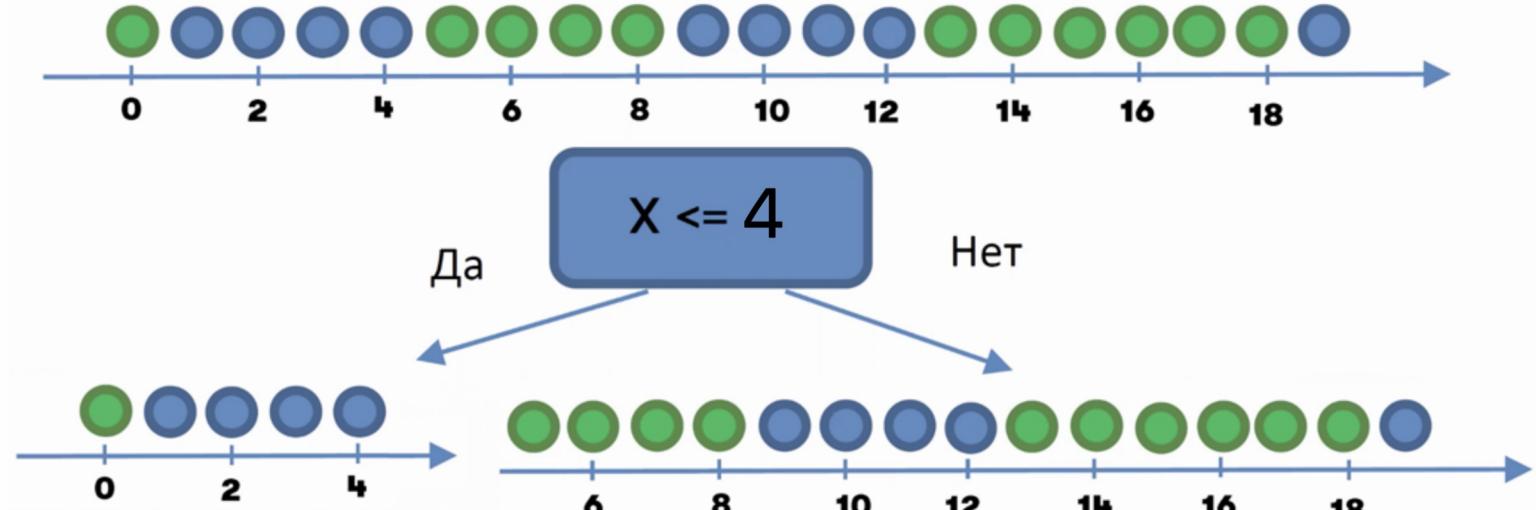
Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{11}{20} \log_2 \frac{11}{20} - \frac{9}{20} \log_2 \frac{9}{20} \approx 0.99277$$



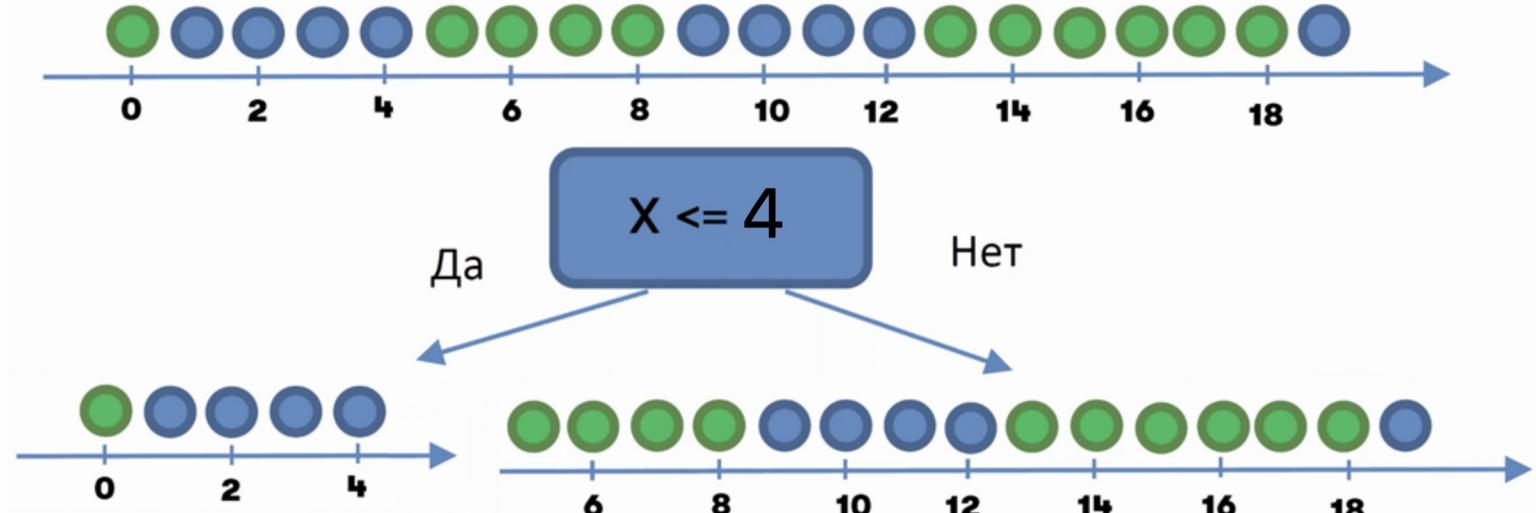
Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{11}{20} \log_2 \frac{11}{20} - \frac{9}{20} \log_2 \frac{9}{20} \approx 0.99277$$



$$S_1 = ?$$

$$S_2 = ?$$

#025

Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

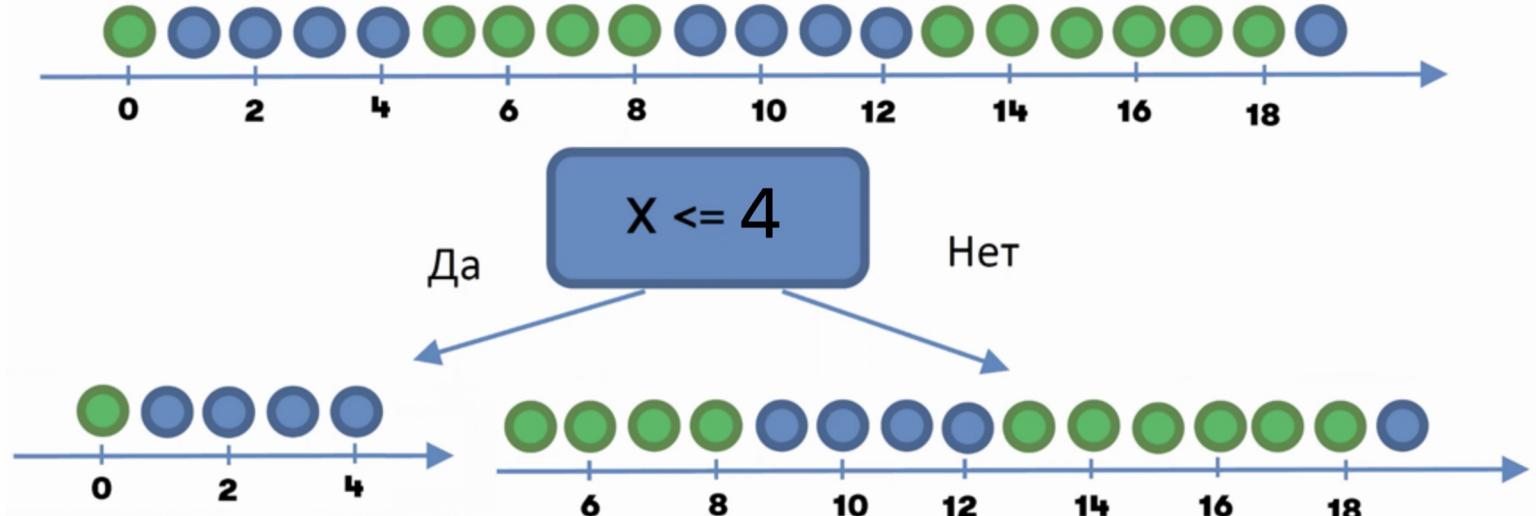
$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{11}{20} \log_2 \frac{11}{20} - \frac{9}{20} \log_2 \frac{9}{20} \approx 0.99277$$

$$S_1 = -\frac{1}{5} \log_2 \frac{1}{5} - \frac{4}{5} \log_2 \frac{4}{5} \approx 0.72193$$

$$S_2 = -\frac{10}{15} \log_2 \frac{10}{15} - \frac{5}{15} \log_2 \frac{5}{15} \approx 0.9183$$



#026

Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

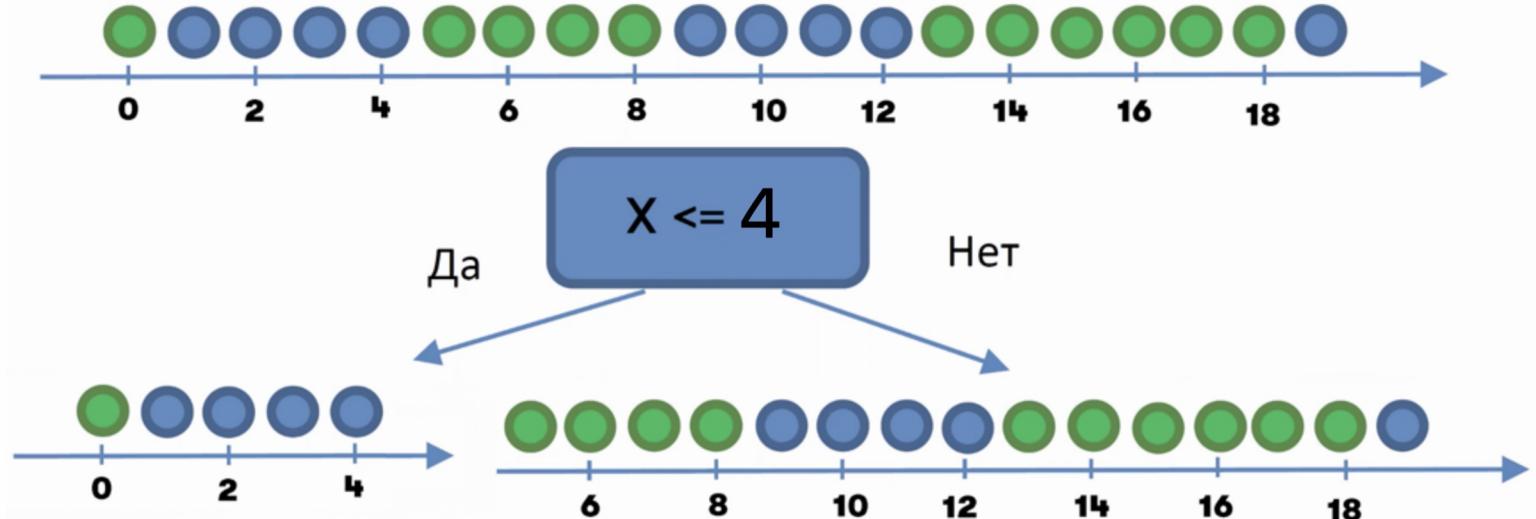
$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{11}{20} \log_2 \frac{11}{20} - \frac{9}{20} \log_2 \frac{9}{20} \approx 0.99277$$

$$S_1 = -\frac{1}{5} \log_2 \frac{1}{5} - \frac{4}{5} \log_2 \frac{4}{5} \approx 0.72193$$

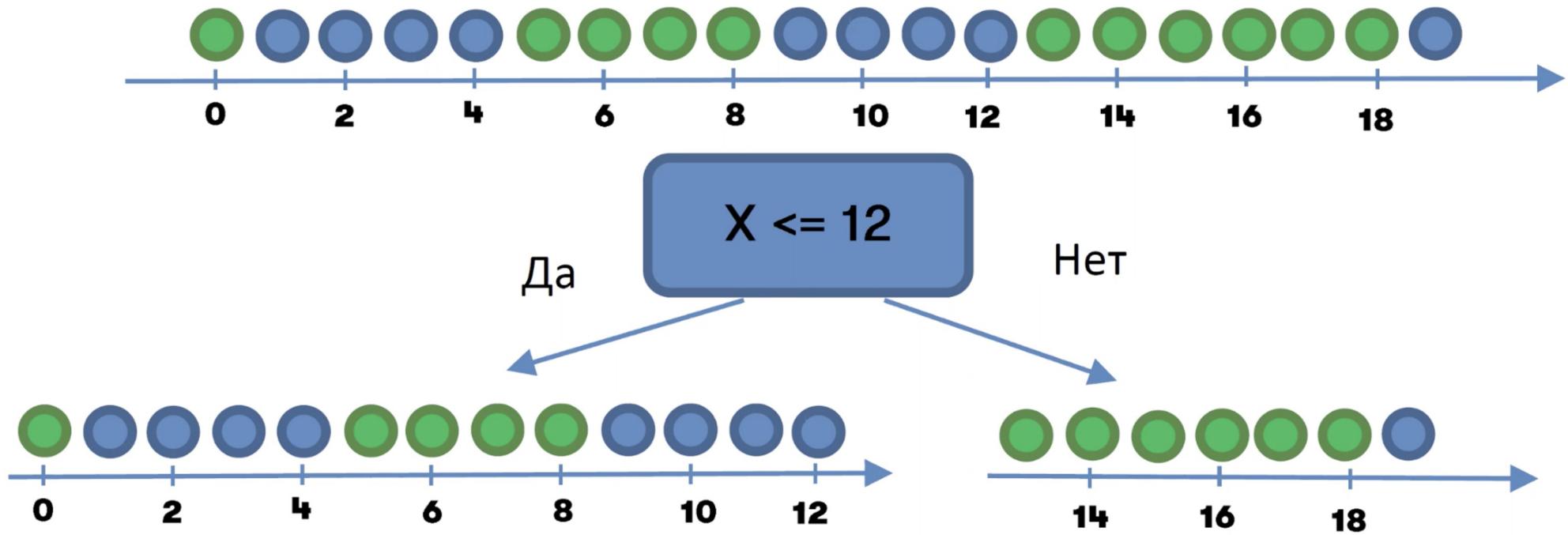
$$S_2 = -\frac{10}{15} \log_2 \frac{10}{15} - \frac{5}{15} \log_2 \frac{5}{15} \approx 0.9183$$



$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i = S_0 - \frac{5}{20} S_1 - \frac{15}{20} S_2 \approx 0.1236$$

#027

Как выгоднее всего строить дерево?



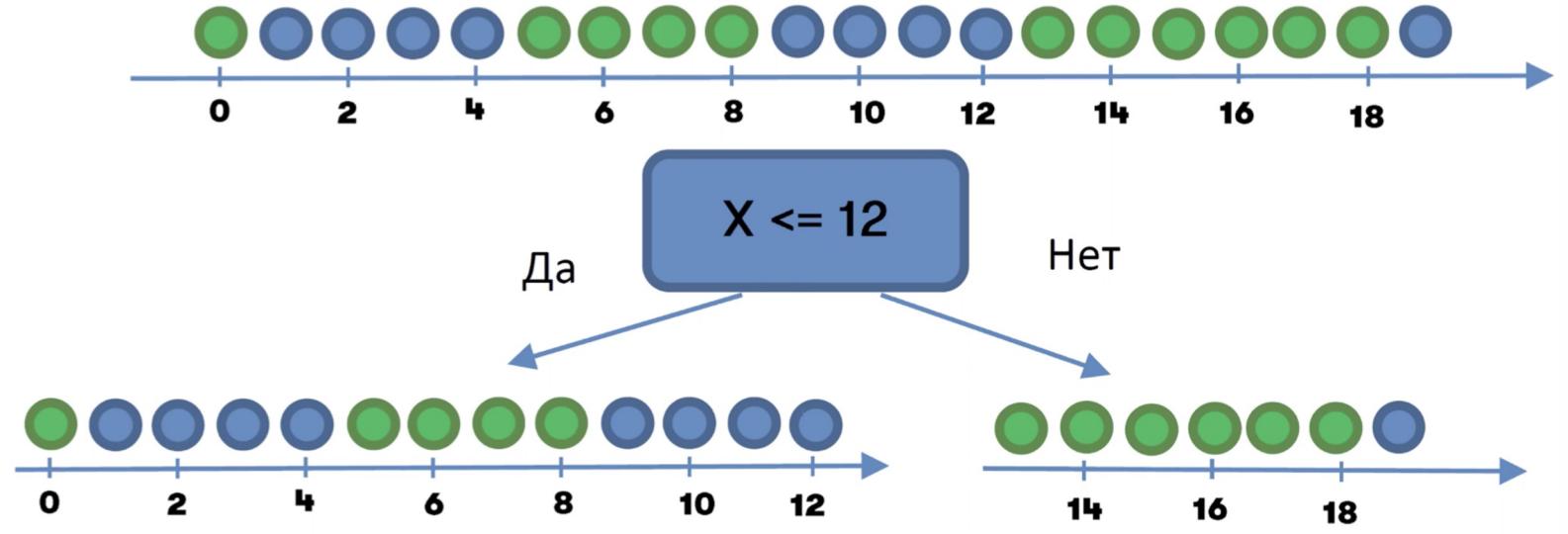
Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{11}{20} \log_2 \frac{11}{20} - \frac{9}{20} \log_2 \frac{9}{20} \approx 0.99277$$



Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

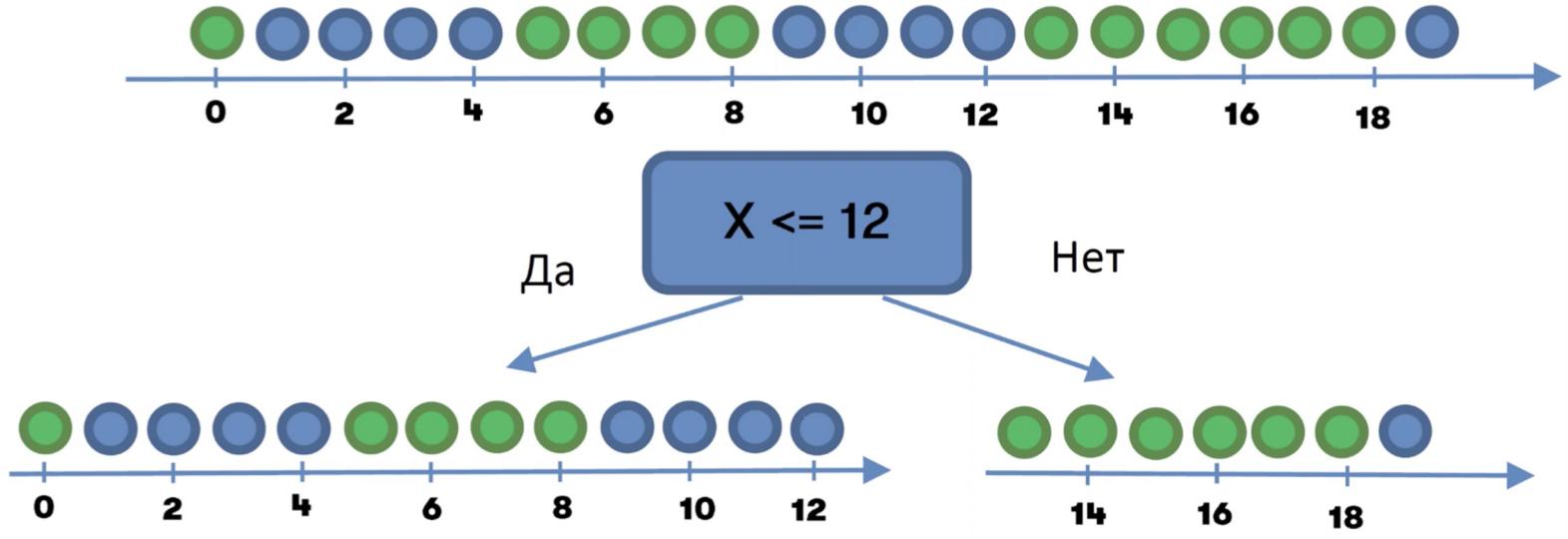
$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$

$$p_3 = \frac{11}{20}, p_c = \frac{9}{20}$$

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{11}{20} \log_2 \frac{11}{20} - \frac{9}{20} \log_2 \frac{9}{20} \approx 0.99277$$

$$S_1 = -\frac{5}{13} \log_2 \frac{5}{13} - \frac{8}{13} \log_2 \frac{8}{13} \approx 0.9612$$

$$S_2 = -\frac{6}{7} \log_2 \frac{6}{7} - \frac{1}{7} \log_2 \frac{1}{7} \approx 0.5917$$

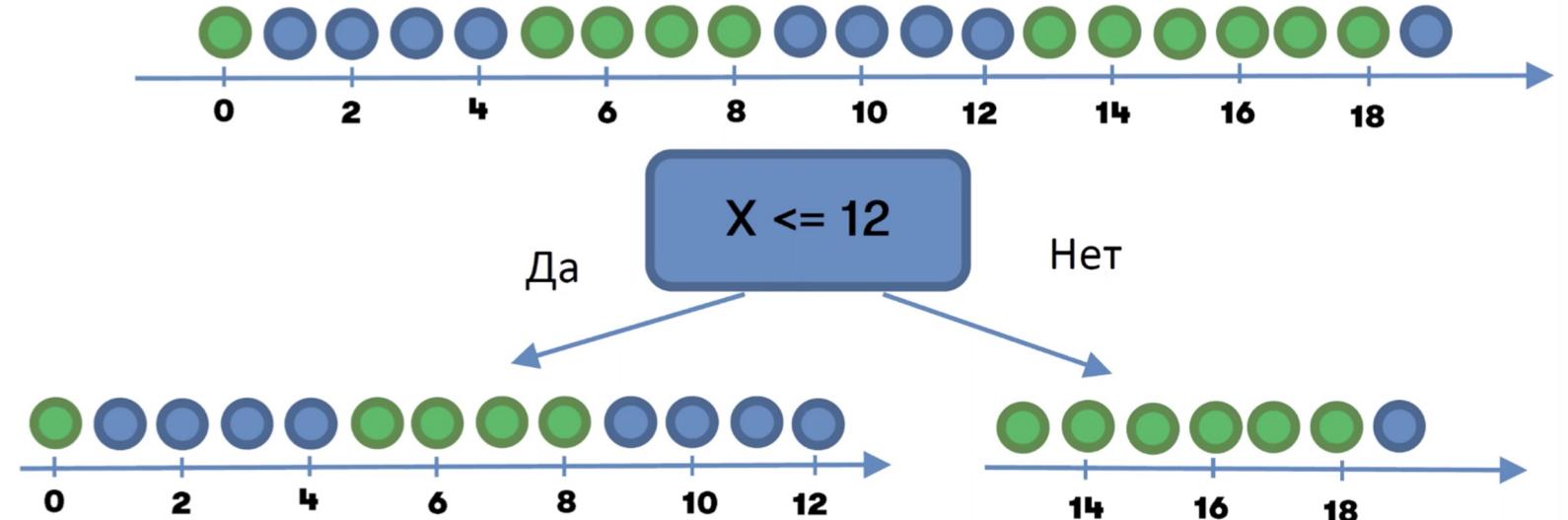


$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i = S_0 - \frac{13}{20} S_1 - \frac{7}{20} S_2 \approx 0.1609$$

Как выгоднее всего строить дерево?

$$S = - \sum_{i=1}^N p_i \log_2 p_i$$

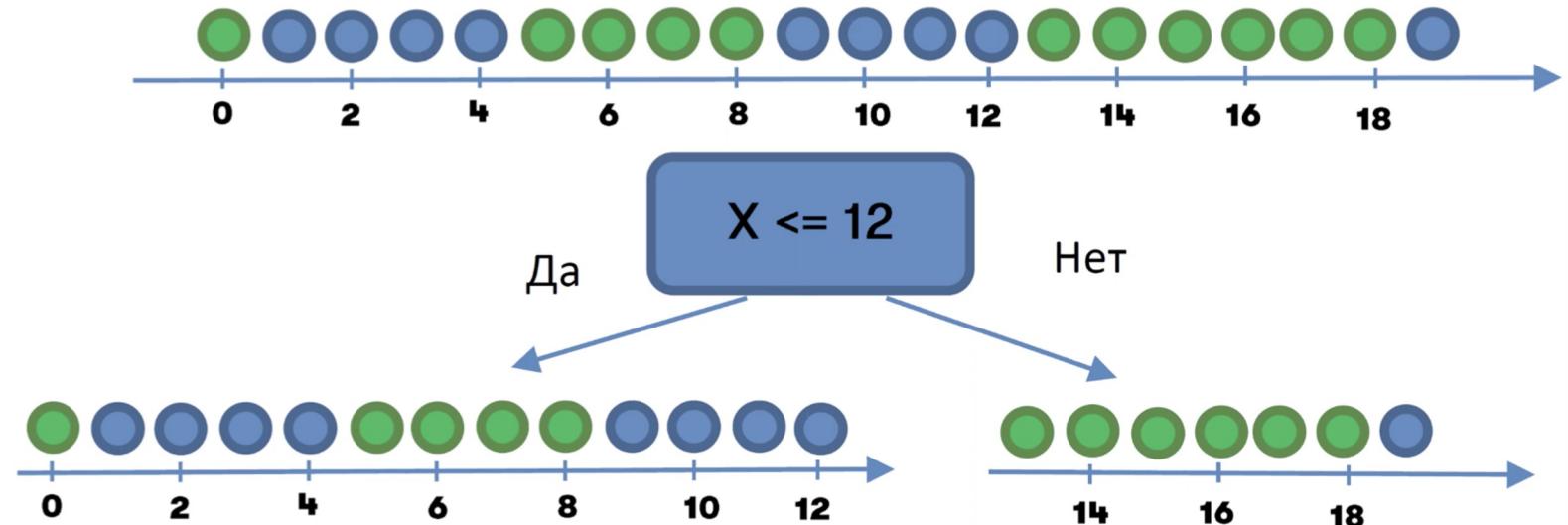
$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i$$



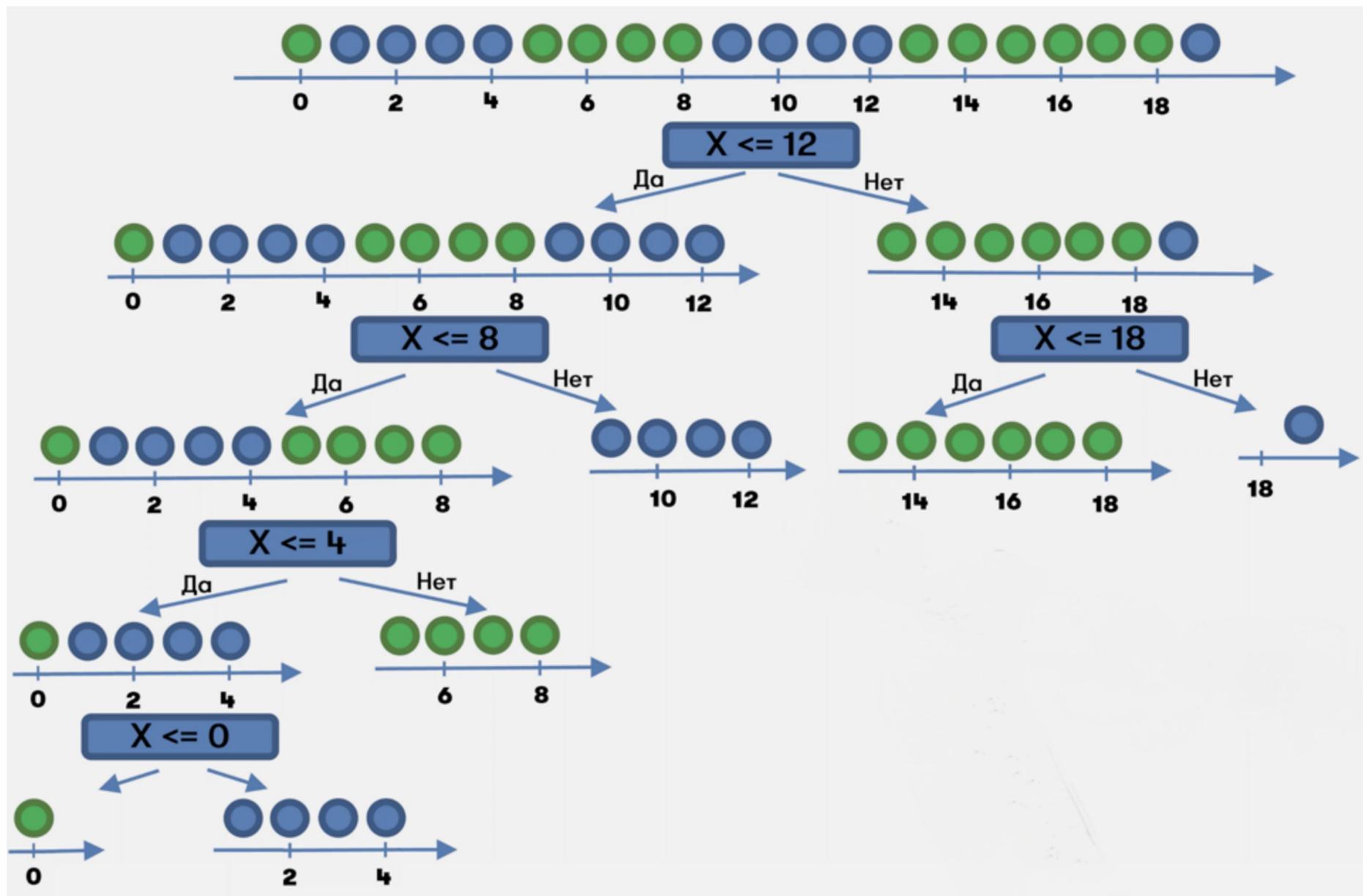
Выбираем разбиение с максимальным $IG(Q)$

Как выгоднее всего строить дерево?

Признак	IG(Q)
0	0.044674
1	0.000806
2	0.024225
3	0.066972
4	0.123571
5	0.059086
6	0.023141
7	0.004853
8	0.000074
9	0.007299
10	0.032825
11	0.080342
12	0.160885
13	0.108108
14	0.064699
15	0.030519
16	0.007153
17	0.000806
18	0.059931



#032



#033

Алгоритм построения дерева

$$S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\} \quad \mathbf{x} = (x_1, \dots, x_d)$$

$$x_j, y \in \{0, 1\}$$

GROWTREE(S)

if ($y = 0$ for all $\langle \mathbf{x}, y \rangle \in S$) **return** new leaf(0)

else if ($y = 1$ for all $\langle \mathbf{x}, y \rangle \in S$) **return** new leaf(1)

else

choose best attribute x_j

$S_0 = \text{all } \langle \mathbf{x}, y \rangle \in S \text{ with } x_j = 0;$

$S_1 = \text{all } \langle \mathbf{x}, y \rangle \in S \text{ with } x_j = 1;$

return new node(x_j , GROWTREE(S_0), GROWTREE(S_1))

Кроме энтропии:

Джини:

$$G = 1 - \sum_k (p_k)^2$$

$$G = 1 - p_+^2 - p_-^2 = 1 - p_+^2 - (1 - p_+)^2 = 2p_+(1 - p_+).$$

Misclassification error:

$$E = 1 - \max_k p_k$$

Рассмотрим индикатор ошибки как функцию потерь

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} L(y, c)$$

$$L(y, c) = [y \neq c]$$

Рассмотрим индикатор ошибки как функцию потерь

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y \neq c]$$

Как выбрать c ?

Рассмотрим индикатор ошибки как функцию потерь

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y \neq c]$$

c^* - самый популярный класс

Рассмотрим индикатор ошибки как функцию потерь

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y \neq c]$$

$c = k_*$ - самый популярный класс

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y \neq k_*] = 1 - \max_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y = k_*] = 1 - p_{k_*} = 1 - \max_k p_k$$

Рассмотрим индикатор ошибки как функцию потерь

$$Q(R_m, j, t) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r)$$

H - критерий информативности (impurity)

$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y \neq c]$$

$c = k_*$ - самый популярный класс

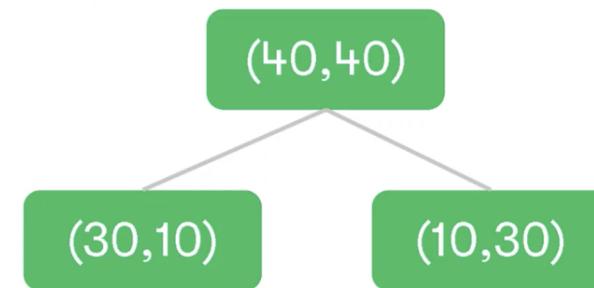
$$H(R) = \min_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y \neq k_*] = 1 - \max_{c \in Y} \frac{1}{|R|} \sum_{(x,y) \in R} [y = k_*] = 1 - p_{k_*} = 1 - \max_k p_k$$

#040

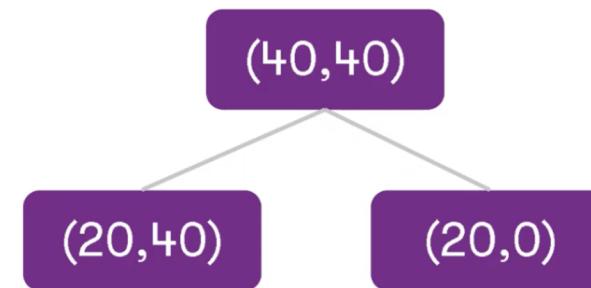
Данный критерий является достаточно грубым, поскольку учитывает частоту лишь одного класса.

Пример. Чувствительность КИ для классификации

A

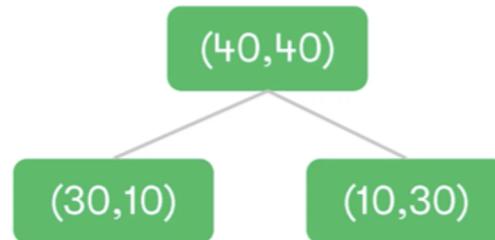


B

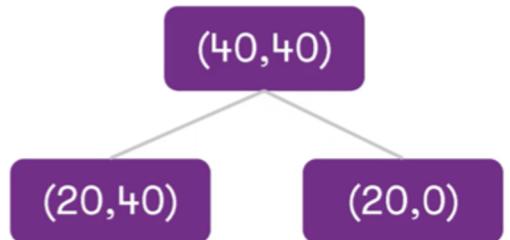


Энтропия:

A



B



A:

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} = 1$$

$$S_1 =$$

$$S_2 =$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i =$$

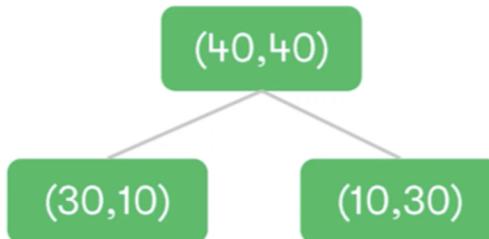
B: $S_1 \approx 0.91$

$$S_2 \approx 0$$

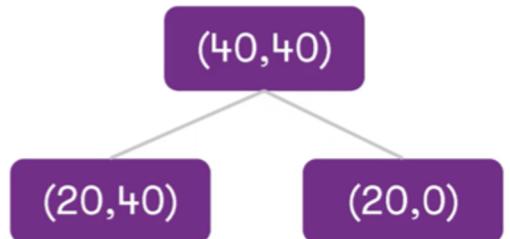
$$IG(Q) \approx 0.31$$

Энтропия:

A



B



A:

$$S_0 = - \sum_{i=1}^N p_i \log_2 p_i = - \sum_{i=1}^2 p_i \log_2 p_i = -\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} = 1$$

$$S_1 = -\frac{3}{4} \log_2 \frac{3}{4} - \frac{1}{4} \log_2 \frac{1}{4} \approx 0.8113$$

$$S_2 = -\frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4} \approx 0.8113$$

$$IG(Q) = S_0 - \sum_{i=1}^q \frac{N_i}{N} S_i = 1 - \frac{1}{2} S_1 - \frac{1}{2} S_2 \approx 0.1887$$

B: $S_1 \approx 0.91$

$$S_2 \approx 0$$

$$IG(Q) \approx 0.31$$

Джини:

$$G = 1 - \sum_k (p_k)^2$$

A: $G = 1 - ((0, 5)^2 + (0.5)^2) = 0.5$

$$G_1 =$$

$$G_2 =$$

$$IG =$$

$$G = 1 - ((0, 5)^2 + (0.5)^2) = 0.5$$

B: $G_1 =$

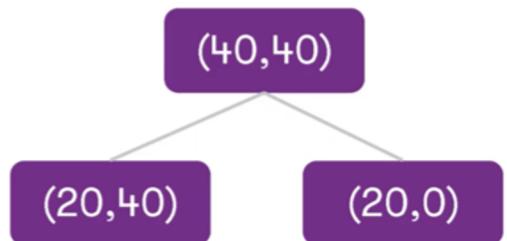
$$G_2 =$$

$$IG =$$

A



B



Джини:

$$G = 1 - \sum_k (p_k)^2$$

A: $G = 1 - ((0, 5)^2 + (0.5)^2) = 0.5$

$$G_1 = 1 - ((0, 25)^2 + (0.75)^2) = 0.375$$

$$G_2 = 1 - ((0, 75)^2 + (0.25)^2) = 0.375$$

$$IG = 0.5 - \frac{1}{2}0.375 - \frac{1}{2}0.375 \approx 0.125$$

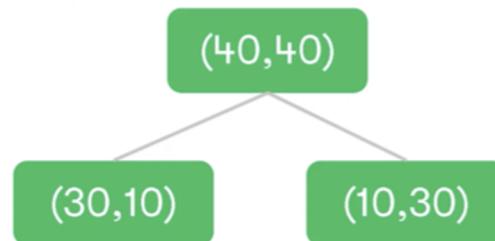
$$G = 1 - ((0, 5)^2 + (0.5)^2) = 0.5$$

B: $G_1 =$

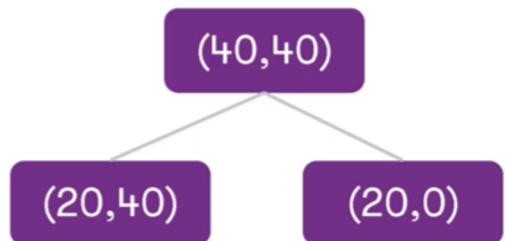
$$G_2 =$$

$$IG =$$

A



B



Джини:

$$G = 1 - \sum_k (p_k)^2$$

A: $G = 1 - ((0, 5)^2 + (0.5)^2) = 0.5$

$$G_1 = 1 - ((0, 25)^2 + (0.75)^2) = 0.375$$

$$G_2 = 1 - ((0, 75)^2 + (0.25)^2) = 0.375$$

$$IG = 0.5 - \frac{1}{2}0.375 - \frac{1}{2}0.375 \approx 0.125$$

$$G = 1 - ((0, 5)^2 + (0.5)^2) = 0.5$$

B: $G_1 = 1 - ((\frac{1}{3})^2 + (\frac{2}{3})^2) \approx 0.44$

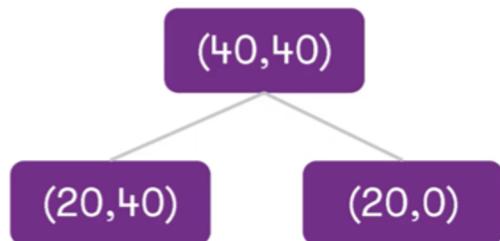
$$G_2 = 1 - ((1)^2 + (0)^2) = 0$$

$$IG = 0.5 - \frac{3}{4}0.44 \approx 0.17$$

A



B



Missclassification error

$$E = 1 - \max_k p_k$$

A: $E = 1 - 0.5 = 0.5$

$$E_1 =$$

$$E_2 =$$

$$IG =$$

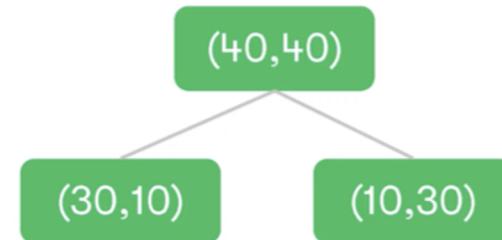
$$E = 1 - 0.5 = 0.5$$

B: $E_1 =$

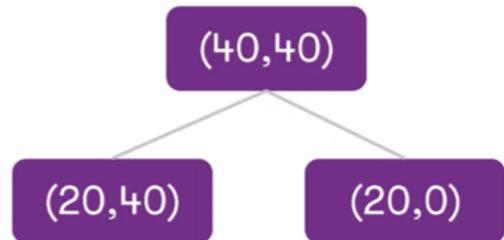
$$E_2 =$$

$$IG =$$

A



B



Missclassification error

$$E = 1 - \max_k p_k$$

A: $E = 1 - 0.5 = 0.5$

$$E_1 = 1 - \frac{3}{4} = 0.25$$

$$E_2 = 1 - \frac{3}{4} = 0.25$$

$$IG = 0.5 - \frac{1}{2}0.25 - \frac{1}{2}0.25 = 0.25$$

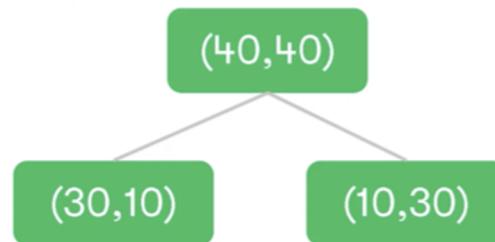
$$E = 1 - 0.5 = 0.5$$

B: $E_1 = 1 - \frac{4}{6} \approx \frac{1}{3}$

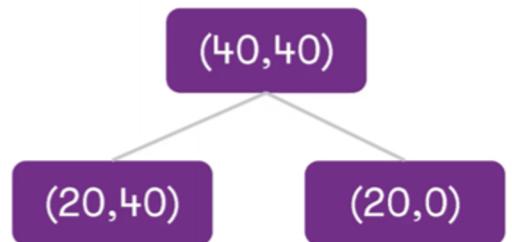
$$E_2 = 1 - 1 = 0$$

$$IG = 0.5 - \frac{3}{4} \frac{1}{3} = 0.25$$

A



B



Регрессия

Как обычно, в регрессии выберем квадрат отклонения в качестве функции потерь. В этом случае критерий информативности будет выглядеть как

$$H(R) = \min_{c \in \mathbb{Y}} \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - c)^2.$$

Как известно, минимум в этом выражении будет достигаться на среднем значении целевой переменной. Значит, критерий можно переписать в следующем виде:

$$H(R) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \left(y_i - \frac{1}{|R|} \sum_{(x_j, y_j) \in R} y_j \right)^2.$$

Регрессия

Как обычно, в регрессии выберем квадрат отклонения в качестве функции потерь. В этом случае критерий информативности будет выглядеть как

$$H(R) = \min_{c \in \mathbb{Y}} \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - c)^2.$$

Как известно, минимум в этом выражении будет достигаться на среднем значении целевой переменной. Значит, критерий можно переписать в следующем виде:

$$H(R) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \left(y_i - \frac{1}{|R|} \sum_{(x_j, y_j) \in R} y_j \right)^2.$$

Мы получили, что информативность вершины измеряется её дисперсией — чем ниже разброс целевой переменной, тем лучше вершина. Разумеется, можно использовать и другие функции ошибки L — например, при выборе абсолютного отклонения мы получим в качестве критерия среднее абсолютное отклонение от медианы.

Работа с категориальными признаками

Пусть категориальный признак x_j имеет множество значений $Q = \{u_1, \dots, u_q\}$, $|Q| = q$.

$$Q = Q_1 \sqcup Q_2$$

$$\beta(x) = [x_j \in Q_1].$$

Проблема: для построения оптимального предиката нужно перебрать $2^{(q-1)} - 1$ вариантов разбиения, что может быть не вполне возможным.

Работа с категориальными признаками

Обозначим через $R_m(u)$ множество объектов, которые попали в вершину m и у которых j -й признак имеет значение u ; через $N_m(u)$ обозначим количество таких объектов.

В случае с бинарной классификацией упорядочим все значения категориального признака на основе того, какая доля объектов с таким значением имеет класс $+1$:

$$\frac{1}{N_m(u_{(1)})} \sum_{x_i \in R_m(u_{(1)})} [y_i = +1] \leq \dots \leq \frac{1}{N_m(u_{(q)})} \sum_{x_i \in R_m(u_{(q)})} [y_i = +1],$$

после чего заменим категорию $u_{(i)}$ на число i , и будем искать разбиение как для вещественного признака. Можно показать, что если искать оптимальное разбиение по критерию Джини или энтропийному критерию, то мы получим такое же разбиение, как и при переборе по всем возможным $2^{q-1} - 1$ вариантам.

Для задачи регрессии с MSE-функционалом это тоже будет верно, если упорядочивать значения признака по среднему ответу объектов с таким значением:

$$\frac{1}{N_m(u_{(1)})} \sum_{x_i \in R_m(u_{(1)})} y_i \leq \dots \leq \frac{1}{N_m(u_{(q)})} \sum_{x_i \in R_m(u_{(q)})} y_i.$$

Работа с пропусками

Пусть у нас есть некоторый признак x^i , значение которого пропущено у некоторых объектов. Как обычно, обозначим через X_m множество объектов, пришедших в рассматриваемую вершину, а через V_m — подмножество X_m , состоящее из объектов с пропущенным значением x^i . В момент выбора сплитов по этому признаку мы будем просто игнорировать объекты из V_m , а когда сплит выбран, мы отправим их в оба поддерева. При этом логично присвоить им веса: $\frac{|X_l|}{|X_m|}$ для левого под дерева и $\frac{|X_r|}{|X_m|}$ для правого. Веса будут учитываться как коэффициенты при $L(y_i, c)$ в формуле информативности.

Критерии останова

Можно придумать большое количество критериев останова. Перечислим некоторые ограничения и критерии:

- Ограничение максимальной глубины дерева.
- Ограничение минимального числа объектов в листе.
- Ограничение максимального количества листьев в дереве.
- Останов в случае, если все объекты в листе относятся к одному классу.
- Требование, что функционал качества при дроблении улучшался как минимум на s процентов.

С помощью грамотного выбора подобных критериев и их параметров можно существенно повлиять на качество дерева. Тем не менее, такой подбор является трудозатратным и требует проведения кросс-валидации.

Популярные методы построения деревьев

- ID3: использует энтропийный критерий. Строит дерево до тех пор, пока в каждом листе не окажутся объекты одного класса, либо пока разбиение вершины дает уменьшение энтропийного критерия.
- C4.5: использует критерий Gain Ratio (нормированный энтропийный критерий). Критерий останова — ограничение на число объектов в листе. Стрижка производится с помощью метода Error-Based Pruning, который использует оценки обобщающей способности для принятия решения об удалении вершины. Обработка пропущенных значений осуществляется с помощью метода, который игнорирует объекты с пропущенными значениями при вычислении критерия ветвления, а затем переносит такие объекты в оба поддерева с определенными весами.
- CART: использует критерий Джини. Стрижка осуществляется с помощью Cost-Complexity Pruning. Для обработки пропусков используется метод суррогатных предикатов.

Класс DecisionTreeClassifier в Scikit-learn

Основные параметры класса `sklearn.tree.DecisionTreeClassifier`:

- `max_depth` – максимальная глубина дерева
- `max_features` — максимальное число признаков, по которым ищется лучшее разбиение в дереве (это нужно потому, что при большом количестве признаков будет "дорого" искать лучшее (по критерию типа прироста информации) разбиение среди всех признаков)
- `min_samples_leaf` – минимальное число объектов в листе. У этого параметра есть понятная интерпретация: скажем, если он равен 5, то дерево будет порождать только те классифицирующие правила, которые верны как минимум для 5 объектов

Параметры дерева надо настраивать в зависимости от входных данных, и делается это обычно с помощью *кросс-валидации*

Плюсы и минусы деревьев решений

Плюсы и минусы деревьев решений

Плюсы:

- Порождение четких правил классификации, понятных человеку, например, "если возраст < 25 и интерес к мотоциклам, то отказать в кредите". Это свойство называют интерпретируемостью модели;
- Деревья решений могут легко визуализироваться, то есть может "интерпретироваться" как сама модель (дерево), так и прогноз для отдельного взятого тестового объекта (путь в дереве);
- Быстрые процессы обучения и прогнозирования;
- Малое число параметров модели;
- Поддержка числовых, и категориальных признаков.

Минусы

- У порождения четких правил классификации есть и другая сторона: деревья очень чувствительны к шумам во входных данных, вся модель может кардинально измениться, если немного изменится обучающая выборка (например, если убрать один из признаков или добавить несколько объектов), поэтому и правила классификации могут сильно изменяться, что ухудшает интерпретируемость модели;
- Необходимость отсекать ветви дерева (pruning) или устанавливать минимальное число элементов в листьях дерева или максимальную глубину дерева для борьбы с переобучением. Впрочем, переобучение — проблема всех методов машинного обучения;
- Нестабильность. Небольшие изменения в данных могут существенно изменять построенное дерево решений. С этой проблемой борются с помощью ансамблей деревьев решений (рассмотрим далее);
- Сложно поддерживаться пропуски в данных. Friedman оценил, что на поддержку пропусков в данных ушло около 50% кода CART
- Модель умеет только интерполировать, но не экстраполировать (это же верно и для леса и бустинга на деревьях). То есть дерево решений делает константный прогноз для объектов, находящихся в признаковом пространстве вне параллелепипеда, охватывающего все объекты обучающей выборки. В нашем примере с зелёными и синими шариками это значит, что модель дает одинаковый прогноз для всех шариков с координатой > 19 или < 0 .

АНСАМБЛЬ КОТИКОВ



#059