



Решающие деревья.

Елена Кантонистова

elena.kantonistova@yandex.ru

ВШЭ, 2021

НА ДАННОМ ЭТАПЕ СТУДЕНТЫ УЖЕ ЗНАЮТ СЛЕДУЮЩИЕ АЛГОРИТМЫ:

Линейные:

- ✓ Линейная регрессия (регрессия)
- ✓ Логистическая регрессия (классификация)
- ✓ Метод опорных векторов (классификация)

НА ДАННОМ ЭТАПЕ СТУДЕНТЫ УЖЕ ЗНАЮТ СЛЕДУЮЩИЕ АЛГОРИТМЫ:

Линейные:

- ✓ Линейная регрессия (регрессия)
- ✓ Логистическая регрессия (классификация)
- ✓ Метод опорных векторов (классификация)

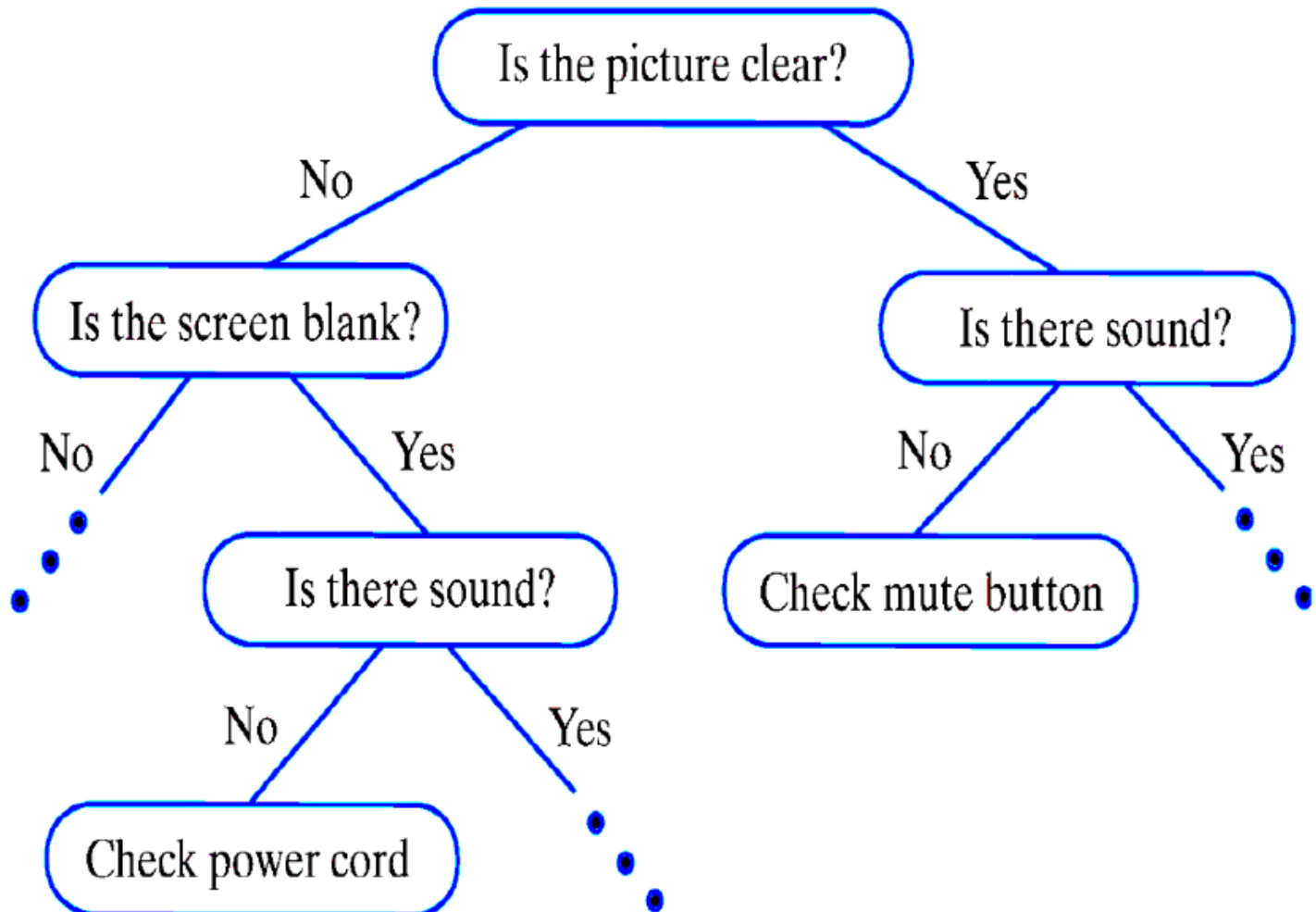
Нелинейные:

- ✓ Наивный байесовский алгоритм (классификация)
- ✓ Метод ближайших соседей (регрессия/классификация)

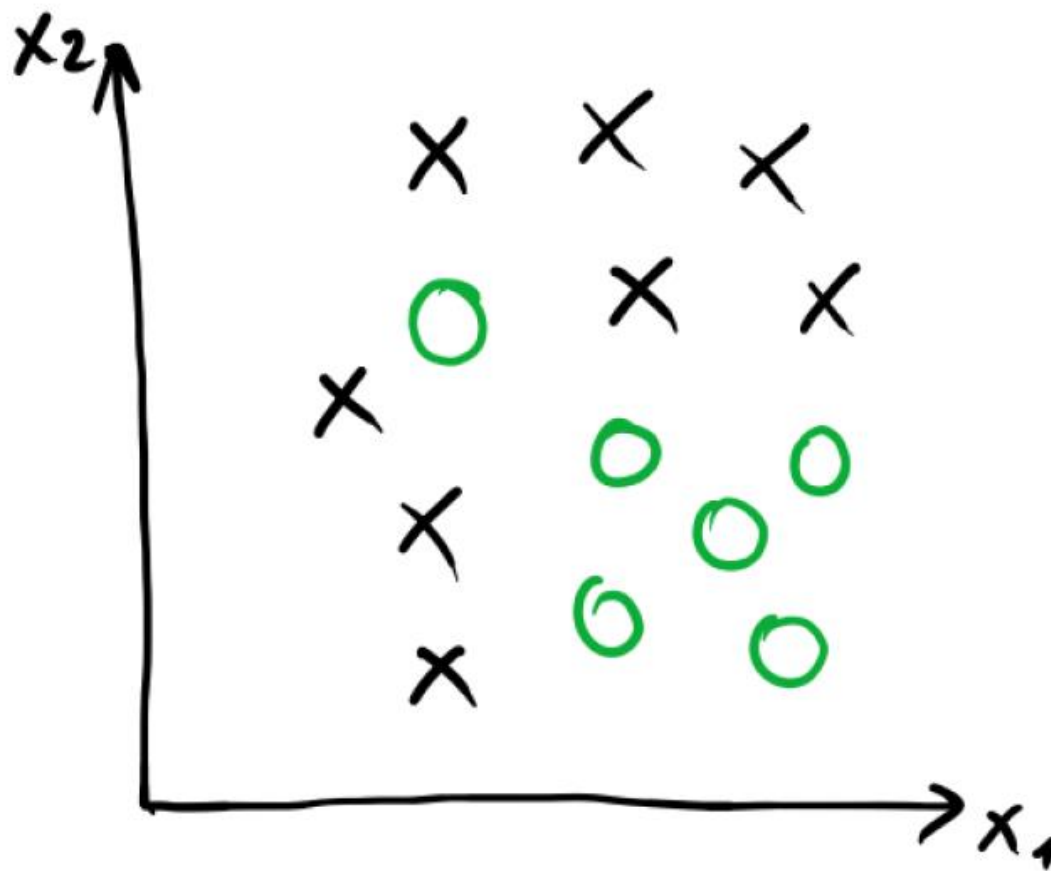
The background features a series of concentric circles in a light gray color, centered on the page. In the four corners, there are decorative elements resembling circuit boards or neural network connections, consisting of thin blue lines and small circles.

РЕШАЮЩИЕ ДЕРЕВЬЯ

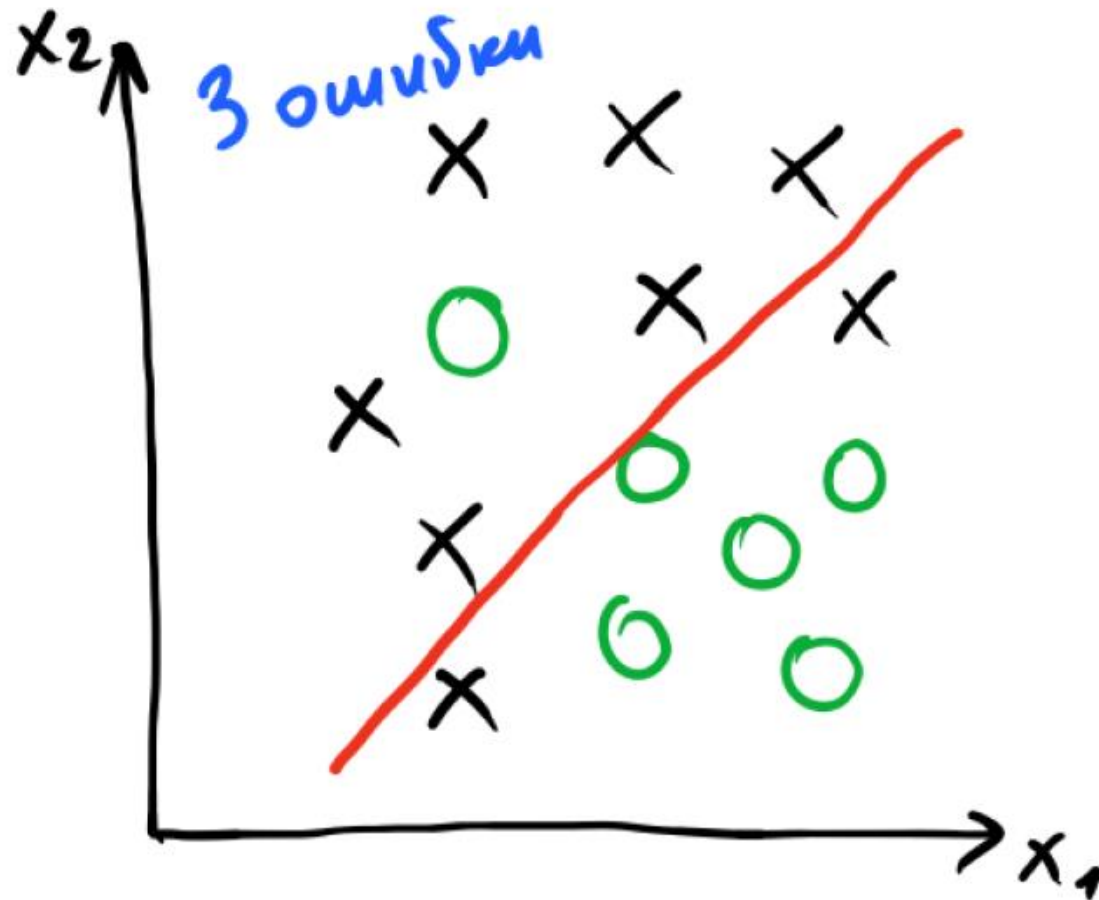
ПРИМЕР РЕШАЮЩЕГО ДЕРЕВА



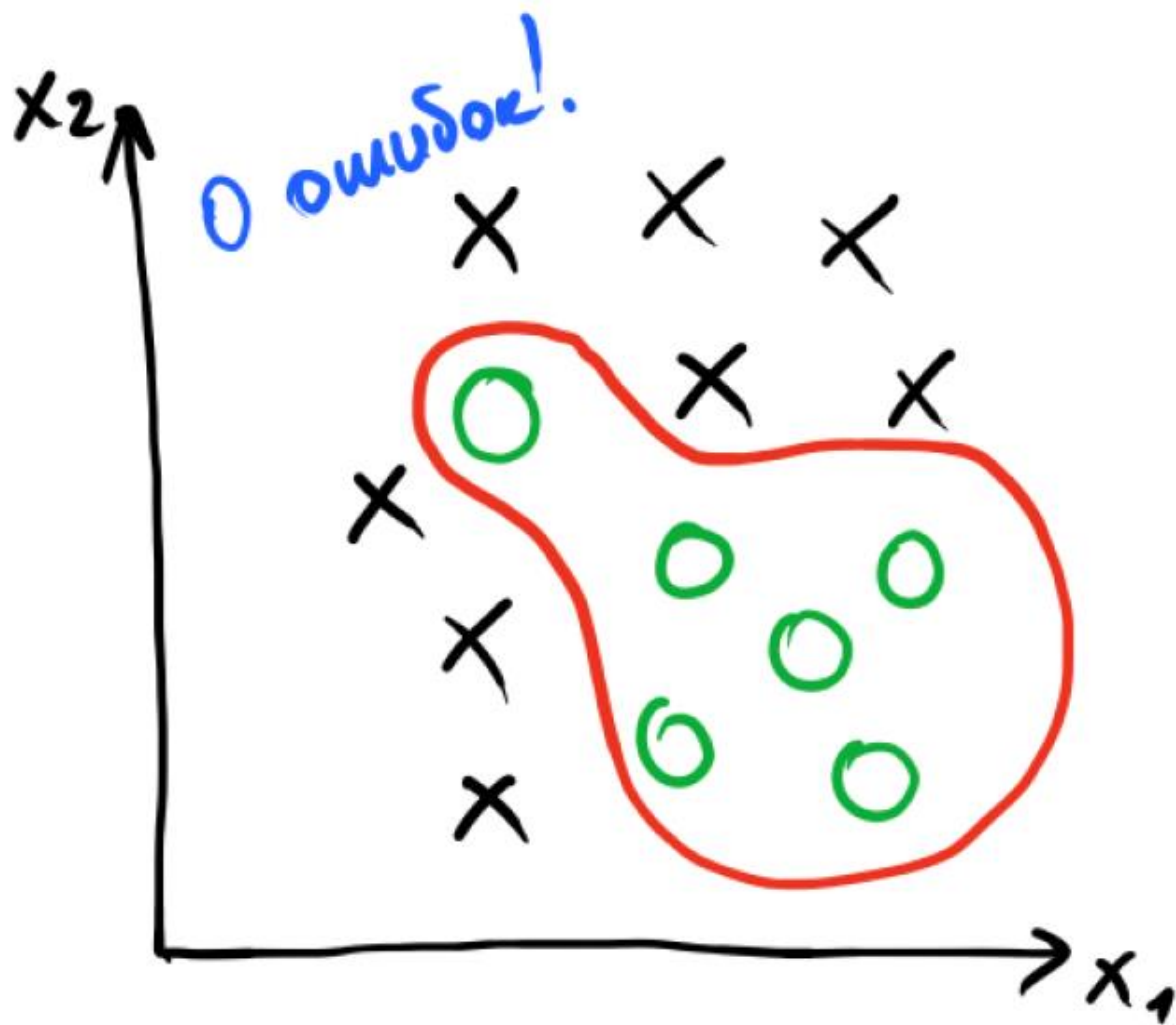
ПРИМЕР



ЛИНЕЙНАЯ МОДЕЛЬ



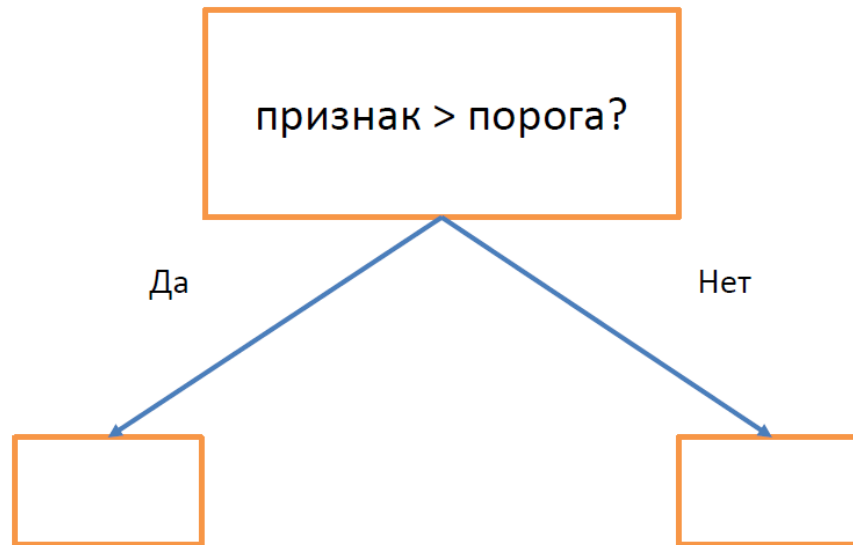
НЕЛИНЕЙНЫЙ АЛГОРИТМ



РЕШАЮЩЕЕ ДЕРЕВО

Решающее дерево – это бинарное дерево, в котором:

1) каждой вершине v приписана функция (предикат) $\beta_v: X \rightarrow \{0,1\}$



РЕШАЮЩЕЕ ДЕРЕВО

Решающее дерево – это бинарное дерево, в котором:

1) каждой вершине v приписана функция (предикат) $\beta_v: X \rightarrow \{0,1\}$



2) каждой листовой вершине v приписан прогноз $c_v \in Y$ (для классификации – класс или вероятность класса, для регрессии – действительное значение целевой переменной)

ЖАДНЫЙ АЛГОРИТМ ПОСТРОЕНИЯ РЕШАЮЩЕГО ДЕРЕВА

1 шаг: найдем наилучшее разбиение всей выборки X на две части: $R_1(j, t) = \{x \mid x_j < t\}$ и $R_2(j, t) = \{x \mid x_j \geq t\}$ с точки зрения некоторого функционала $Q(X, j, t)$:

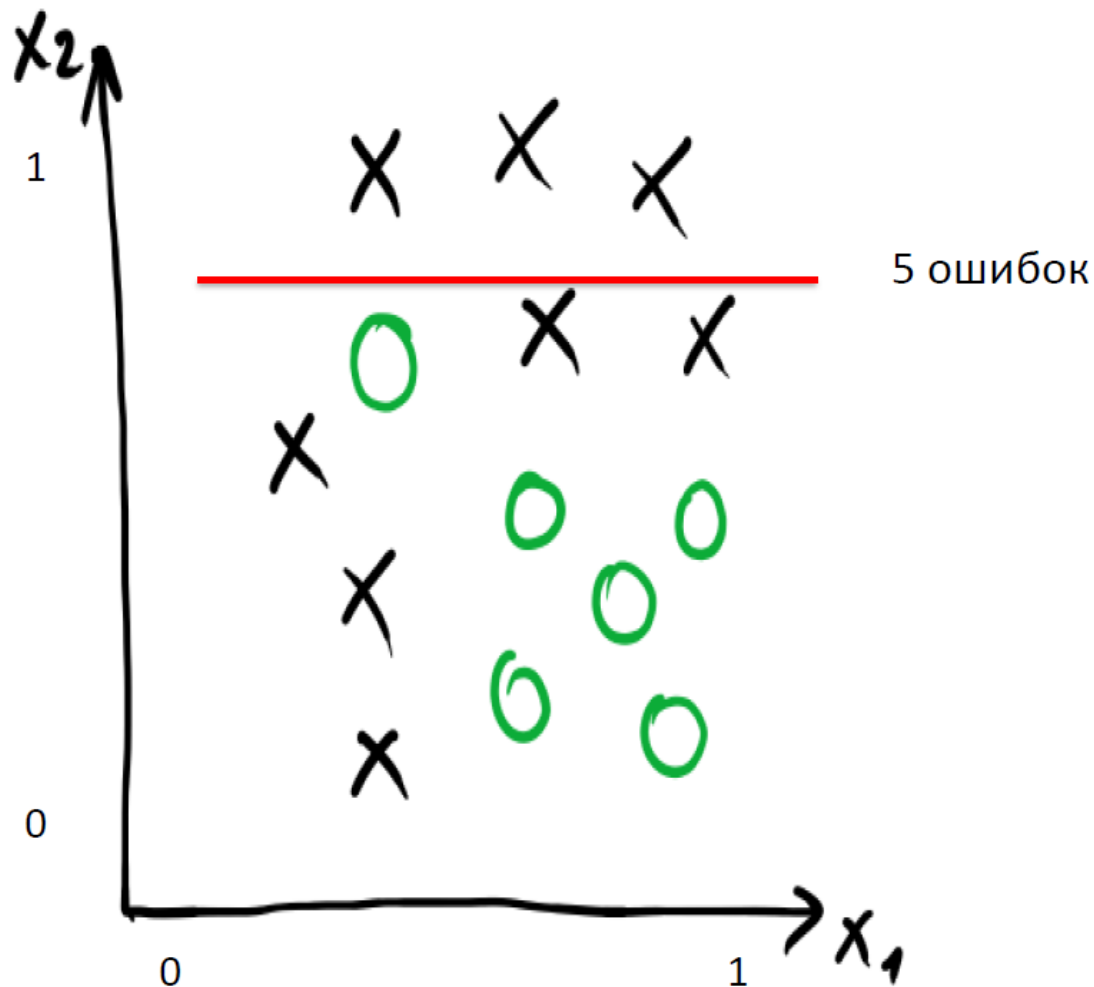
- найдем наилучшие j (признак) и t (порог)
- создадим корень дерева, поставив в него предикат $[x_j < t]$.

2 шаг: Для каждой из полученных подвыборок R_1 и R_2 рекурсивно применим шаг 1. И т.д.

В каждой вершине на каждом шаге проверяем, не выполнилось ли условие останова. Если выполнилось, то объявляем вершину листом и записываем в него предсказание.

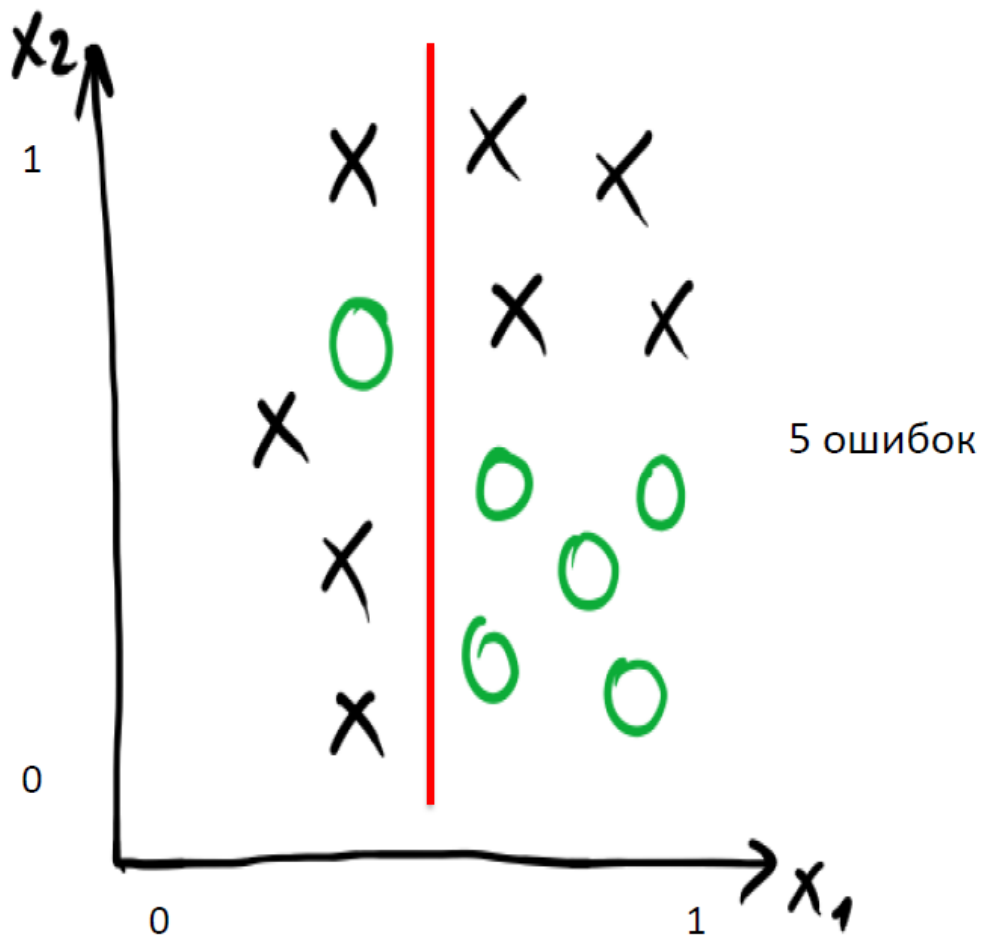
ПРИМЕР: ПОСТРОЕНИЕ РЕШАЮЩЕГО ДЕРЕВА В ЗАДАЧЕ КЛАССИФИКАЦИИ

- Жадно найдем наилучший предикат



ПРИМЕР

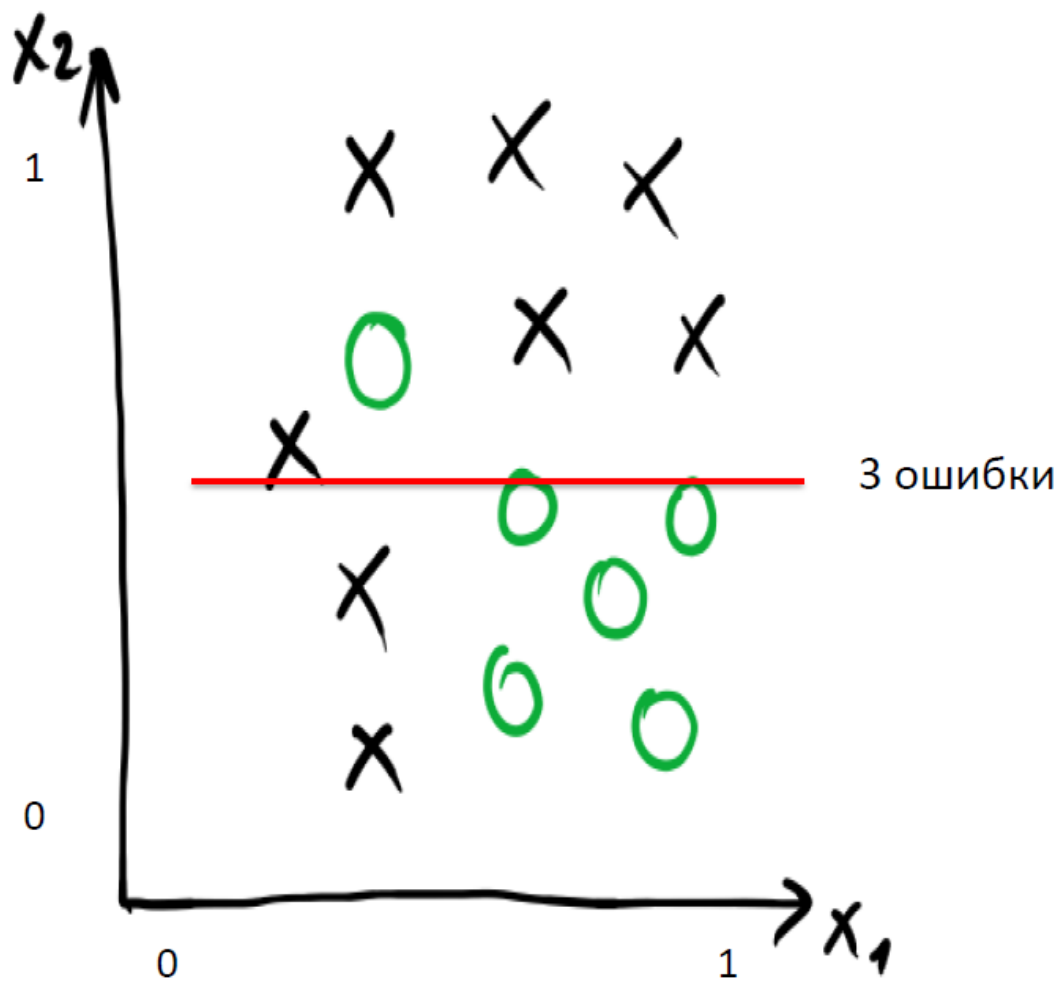
- Жадно найдем наилучший предикат



$$x_1 > 0.5$$

ПРИМЕР

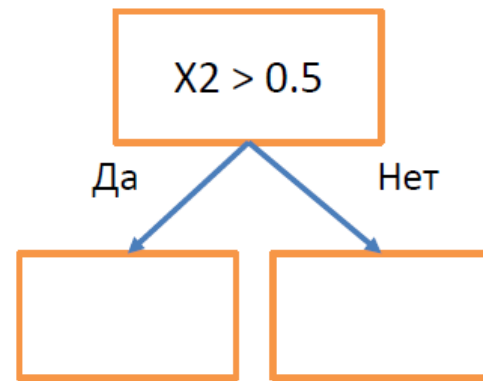
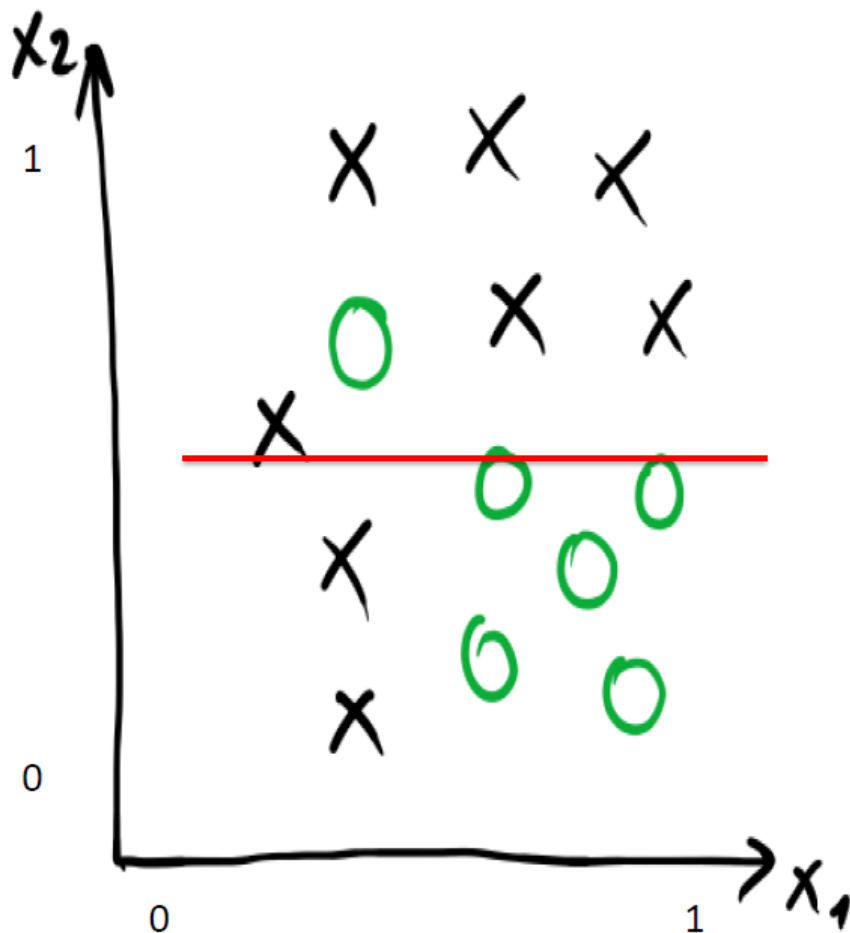
- Жадно найдем наилучший предикат



$$x_2 > 0.5$$

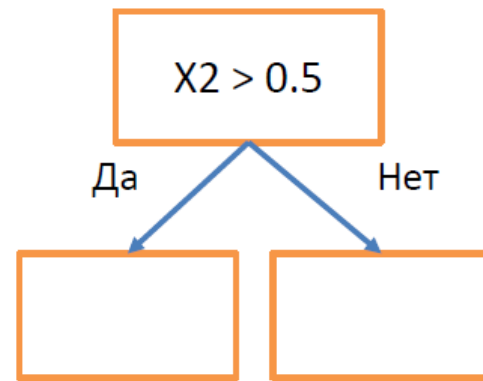
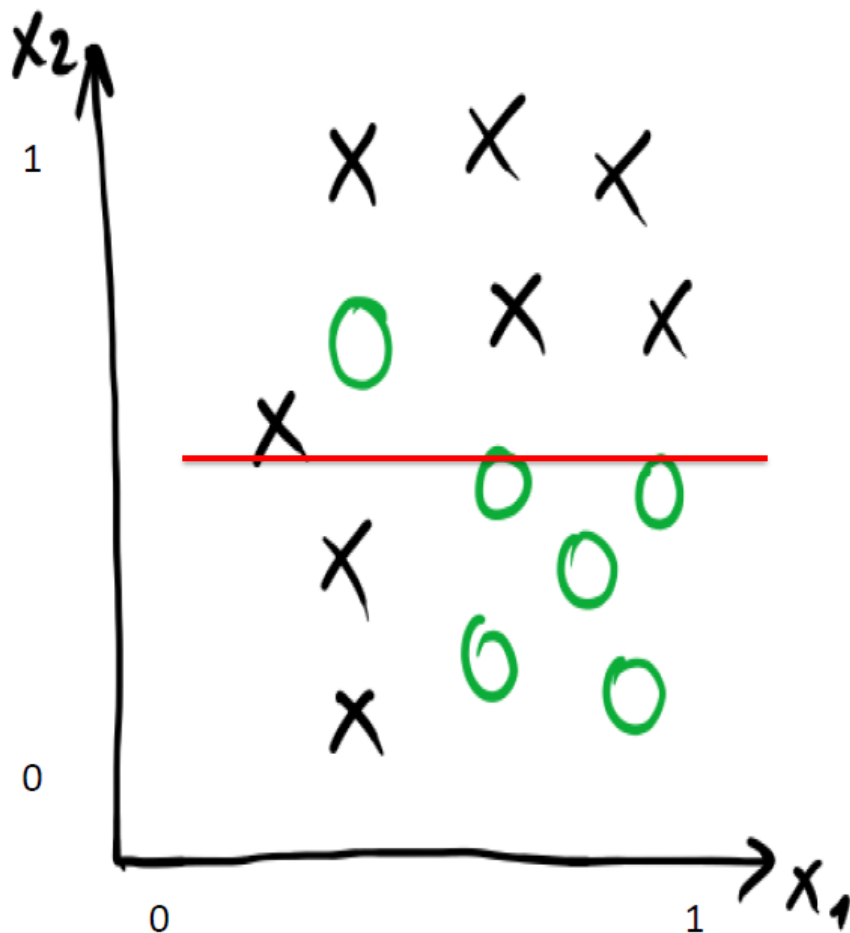
ПРИМЕР

- Нашли лучшее первое ветвление



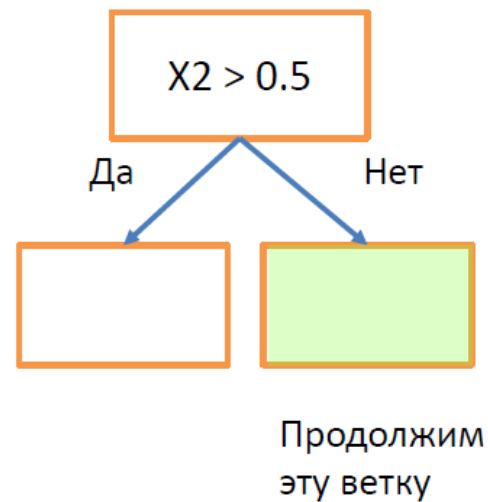
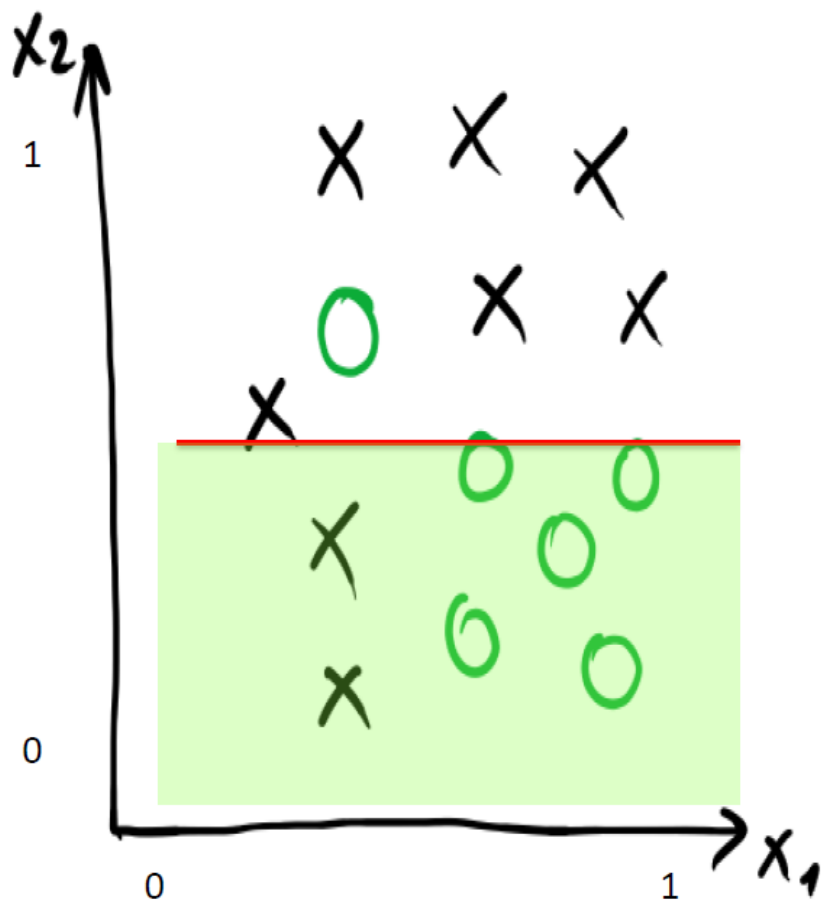
ПРИМЕР

- Нашли лучшее первое ветвление



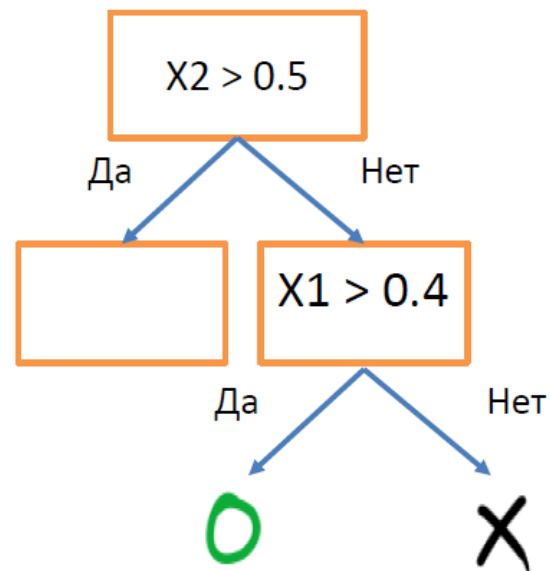
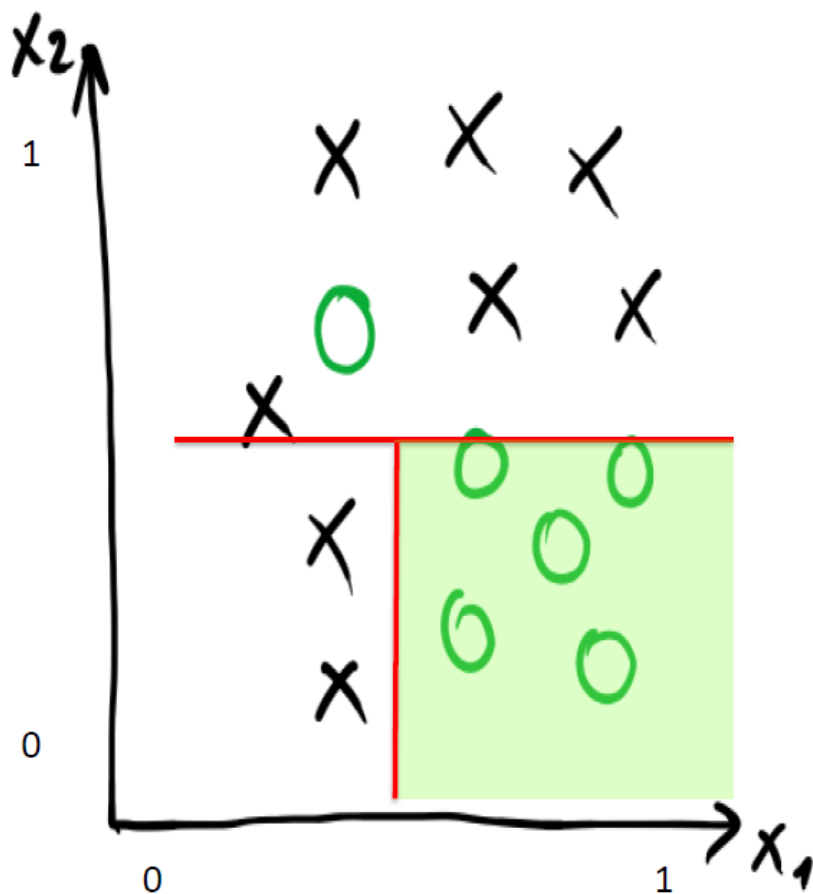
ПРИМЕР

- Нашли лучшее первое ветвление



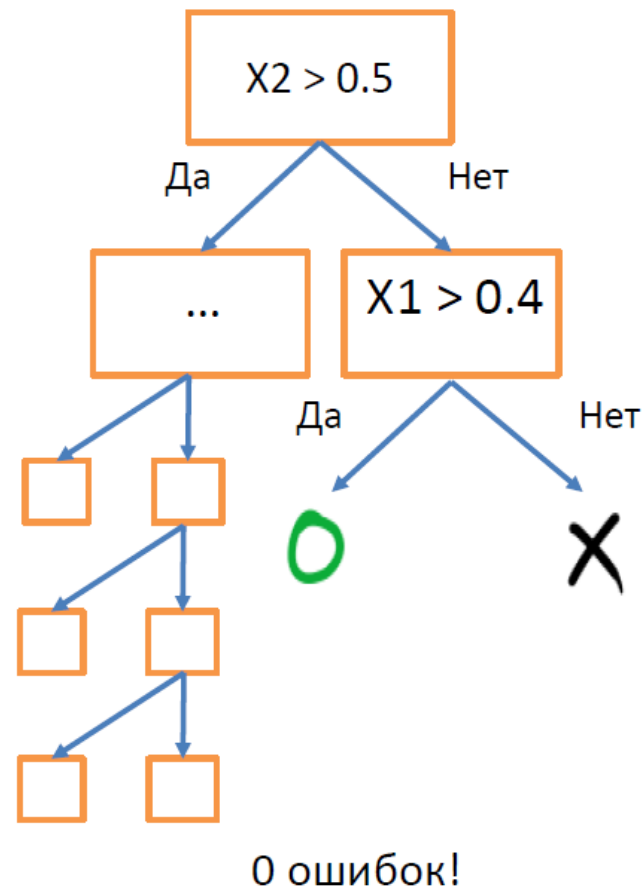
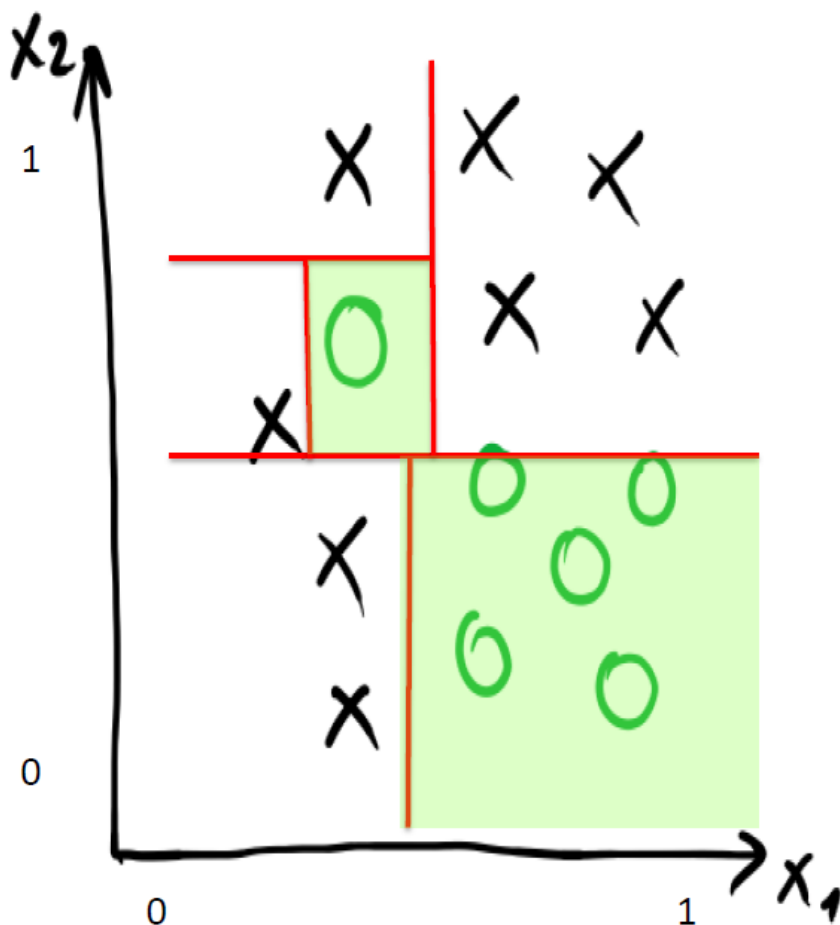
ПРИМЕР

- Нашли лучшее второе ветвление



ПРИМЕР

- Построили всё дерево



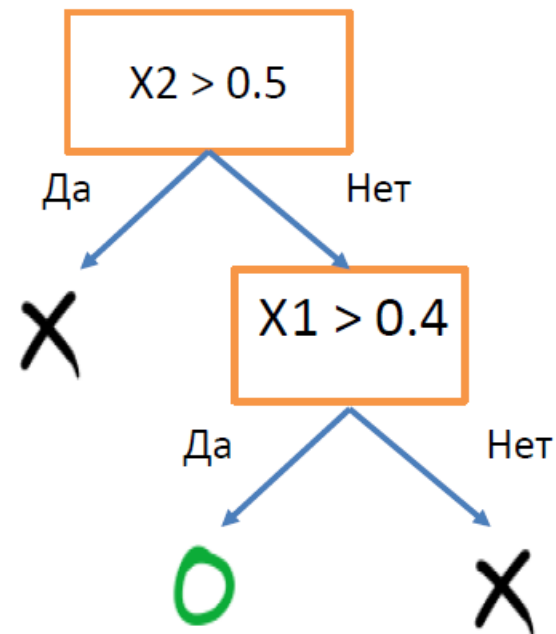
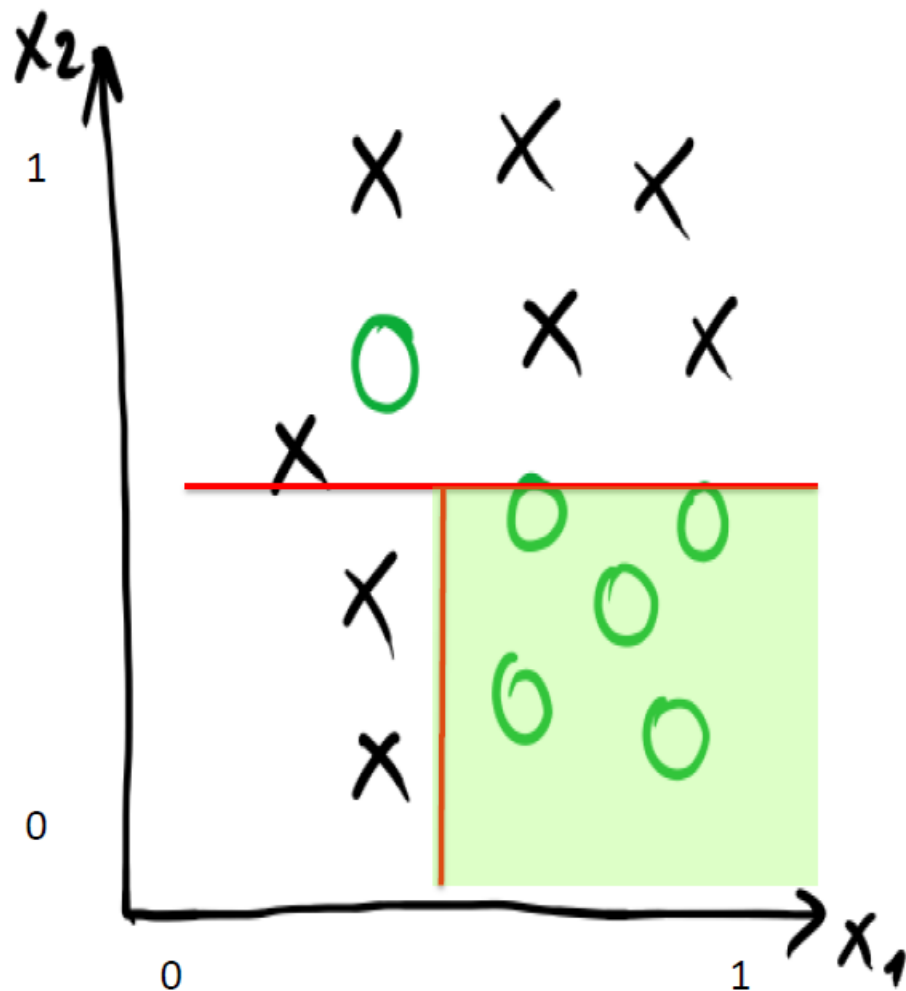
ПЕРЕОБУЧЕНИЕ

Для любой выборки можно построить решающее дерево, не допускающее на ней ни одной ошибки. Такое дерево скорее всего будет переобученным.

ЧТО ВЛИЯЕТ НА ПОСТРОЕНИЕ РЕШАЮЩЕГО ДЕРЕВА

- вид предикатов в вершинах
- функционал качества $Q(X, j, t)$
- критерий останова
- метод обработки пропущенных значений

ПРИМЕР



1 ошибка, но
скорее всего будет
лучше на тесте!

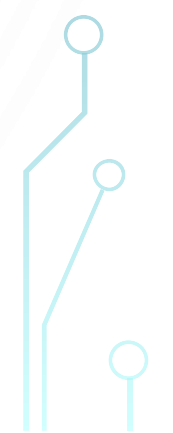



ЭТАПЫ ПОСТРОЕНИЯ РЕШАЮЩЕГО ДЕРЕВА

1. Нахождение структуры дерева:

- выбор предикатов в узлах
- выбор структуры дерева (глубина, условия на продолжение ветвления и т.д.)

2. Получение предсказаний (значений целевой переменной) в листьях дерева.

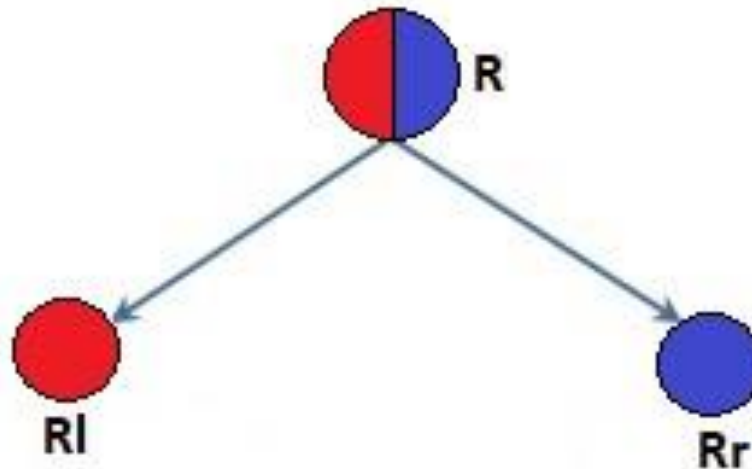


1 ЭТАП: НАХОЖДЕНИЕ СТРУКТУРЫ ДЕРЕВА

В каждой вершине оптимизируем функционал $Q(X, j, t)$.

- Пусть R – множество объектов, попадающих в вершину на данном шаге, а R_l и R_r - объекты, попадающие в левую и правую ветки после разбиения.

Цель: хотим, чтобы после разбиения объектов на две группы внутри каждой группы как можно больше объектов было одного класса.



КРИТЕРИИ ИНФОРМАТИВНОСТИ

- Пусть R – множество объектов, попадающих в вершину на данном шаге, а R_l и R_r - объекты, попадающие в левую и правую ветки после разбиения.

Цель: хотим, чтобы после разбиения объектов на две группы внутри каждой группы как можно больше объектов было одного класса.

- Функция $H(R)$ - критерий информативности - оценивает меру неоднородности целевых переменных внутри группы R .
- Чем меньше разнообразие целевой переменной внутри группы, тем меньше значение $H(R)$. То есть хотим

$$H(R_l) \rightarrow \min, H(R_r) \rightarrow \min$$

КРИТЕРИИ ИНФОРМАТИВНОСТИ

- Пусть R – множество объектов, попадающих в вершину на данном шаге, а R_l и R_r - объекты, попадающие в левую и правую ветки после разбиения.

Цель: хотим, чтобы после разбиения объектов на две группы внутри каждой группы как можно больше объектов было одного класса.

- Чем меньше разнообразие целевой переменной внутри группы, тем меньше значение $H(R)$. То есть

$$H(R_l) \rightarrow \min, H(R_r) \rightarrow \min$$

- Определим функционал Q по формуле:

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

КРИТЕРИИ ИНФОРМАТИВНОСТИ

- Пусть R – множество объектов, попадающих в вершину на данном шаге, а R_l и R_r - объекты, попадающие в левую и правую ветки после разбиения.

Цель: хотим, чтобы после разбиения объектов на две группы внутри каждой группы как можно больше объектов было одного класса.

- Чем меньше разнообразие целевой переменной внутри группы, тем меньше значение $H(R)$. То есть

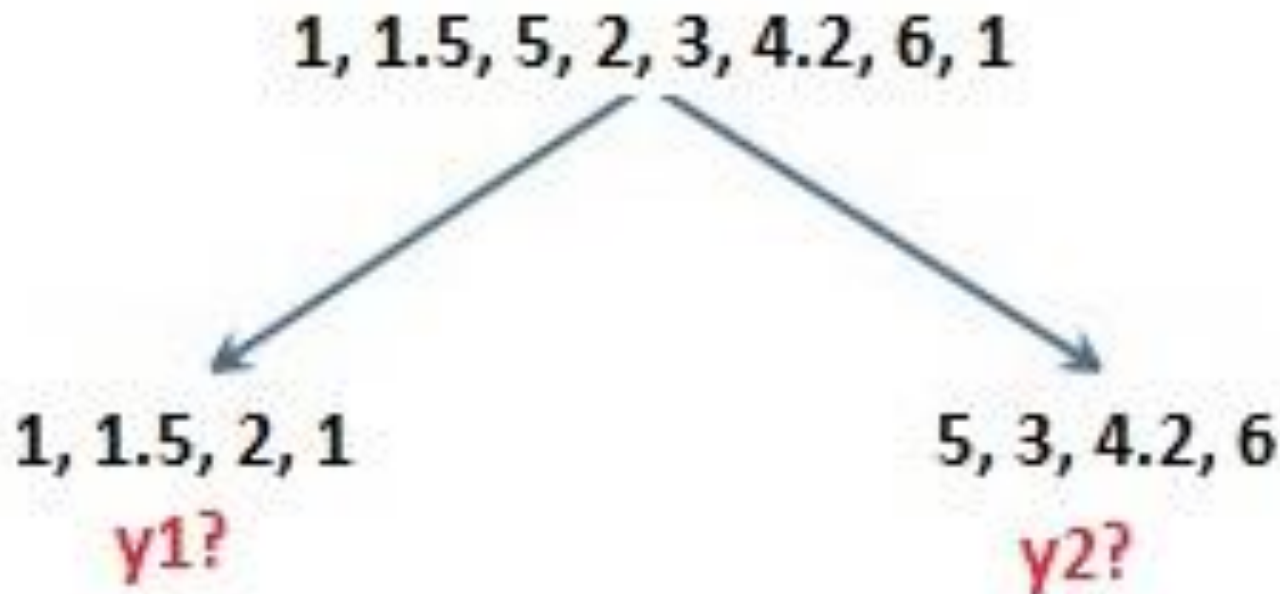
$$H(R_l) \rightarrow \min, H(R_r) \rightarrow \min$$

- Определим функционал Q по формуле:

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

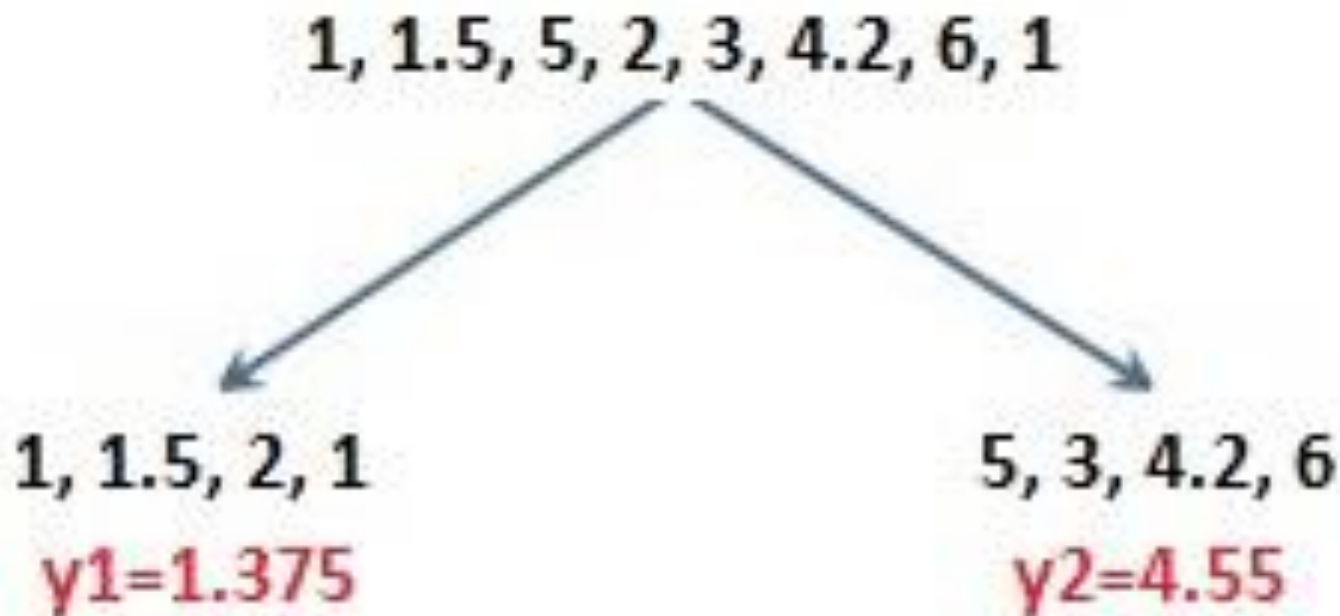
2 ЭТАП: ПОЛУЧЕНИЕ ПРЕДСКАЗАНИЙ В ЛИСТЬЯХ

Предположим, что в лист дерева попало несколько объектов. В каждом листе дерево предсказывает константу. Какую константу выгоднее всего выдать в качестве ответа?



2 ЭТАП: ПОЛУЧЕНИЕ ПРЕДСКАЗАНИЙ В ЛИСТЬЯХ

Если в качестве функционала ошибки в листе использовать среднеквадратичную ошибку, то в качестве ответа надо выдавать среднее значение целевых переменных всех объектов, попавших в лист.



КРИТЕРИЙ ИНФОРМАТИВНОСТИ В ЗАДАЧЕ РЕГРЕССИИ

- В каждом листе дерево выдает константу c (вещественное число – в регрессии, класс или вероятность класса – в классификации).
- Чем лучше объекты в листе предсказываются этой константой, тем меньше средняя ошибка на объектах:

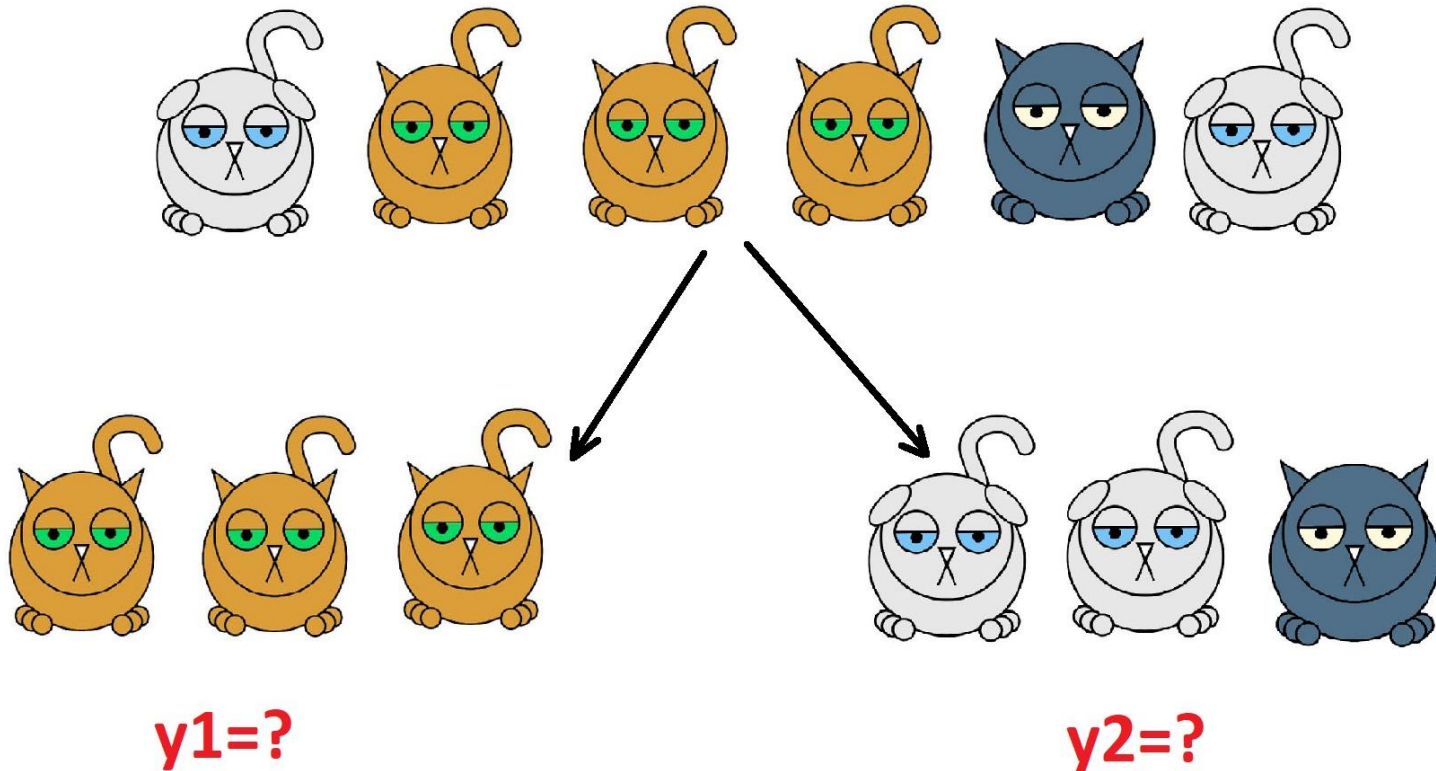
$$H(R) = \min_{c \in \mathbb{R}} \frac{1}{|R|} \sum_{(x_i, y_i) \in R} L(y_i, c),$$

где $L(y, c)$ – некоторая функция потерь.

Информативность в листе – это дисперсия целевой переменной (для объектов, попавших в этот лист). Чем меньше дисперсия, тем меньше разброс целевой переменной объектов, попавших в лист.

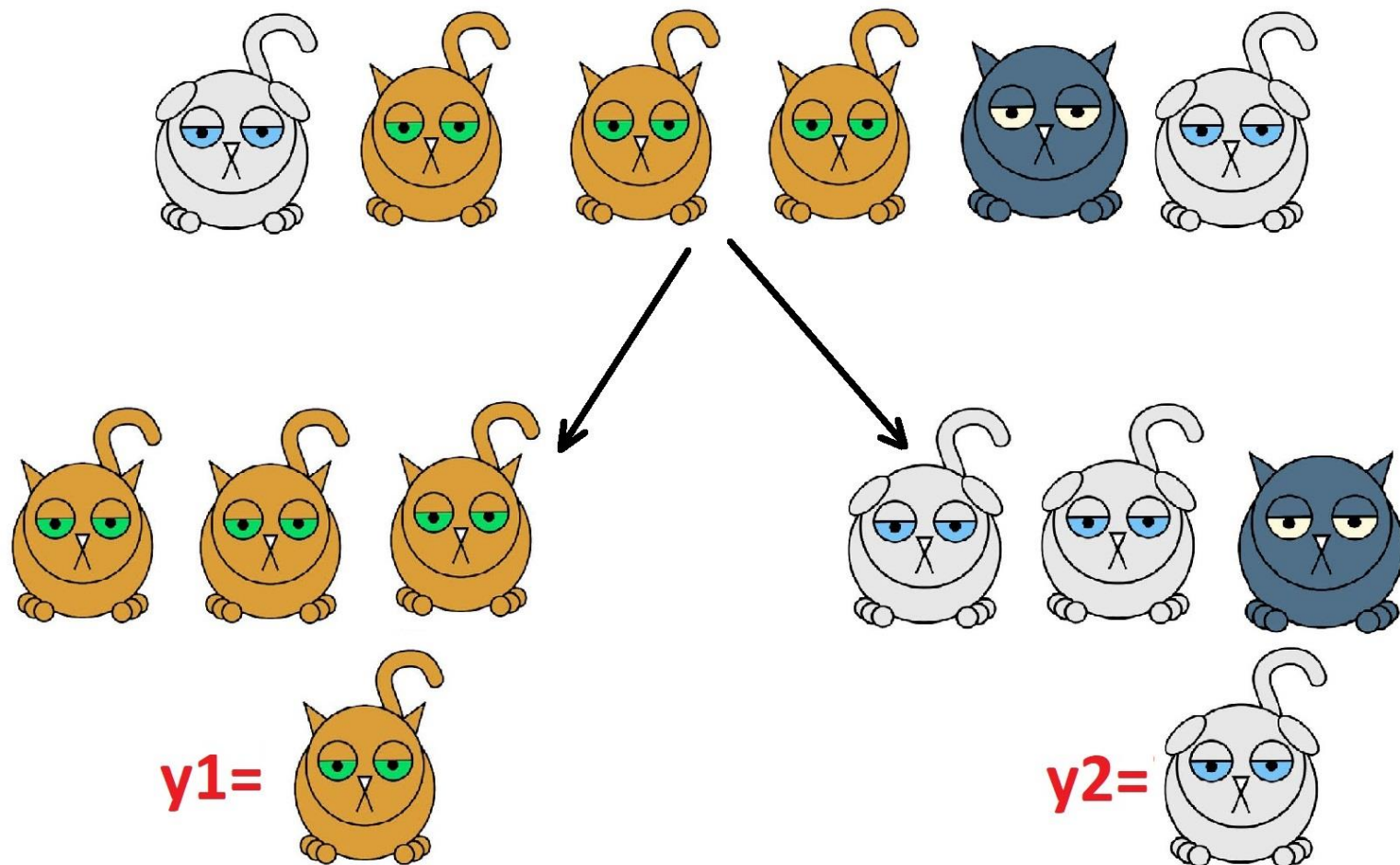
ПОЛУЧЕНИЕ ПРЕДСКАЗАНИЙ В ЗАДАЧЕ КЛАССИФИКАЦИИ

Предположим, что в лист дерева попало несколько объектов. В каждом листе дерево предсказывает класс объекта. Какой класс выгоднее всего выдать в качестве ответа?



ПОЛУЧЕНИЕ ПРЕДСКАЗАНИЙ В ЗАДАЧЕ КЛАССИФИКАЦИИ

Разумнее всего в качестве ответа в листе выдавать самый представительный класс.



$H(R)$ В ЗАДАЧАХ КЛАССИФИКАЦИИ

Решаем задачу классификации с K классами: $1, 2, \dots, K$.

- Пусть p_k доля объектов класса k , попавших в вершину:

$$p_k = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} [y_i = k]$$

- Пусть k_* - самый представительный класс в данной вершине:

$$k_* = \operatorname{argmax}_k p_k$$

Ошибка классификации:

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

H(R) В ЗАДАЧАХ КЛАССИФИКАЦИИ

- Будем в каждой вершине в качестве ответа выдавать не класс, а распределение вероятностей классов:

$$c = (c_1, \dots, c_K), \sum_i c_i = 1.$$

$H(R)$ В ЗАДАЧАХ КЛАССИФИКАЦИИ

- Будем в каждой вершине в качестве ответа выдавать не класс, а распределение вероятностей классов:

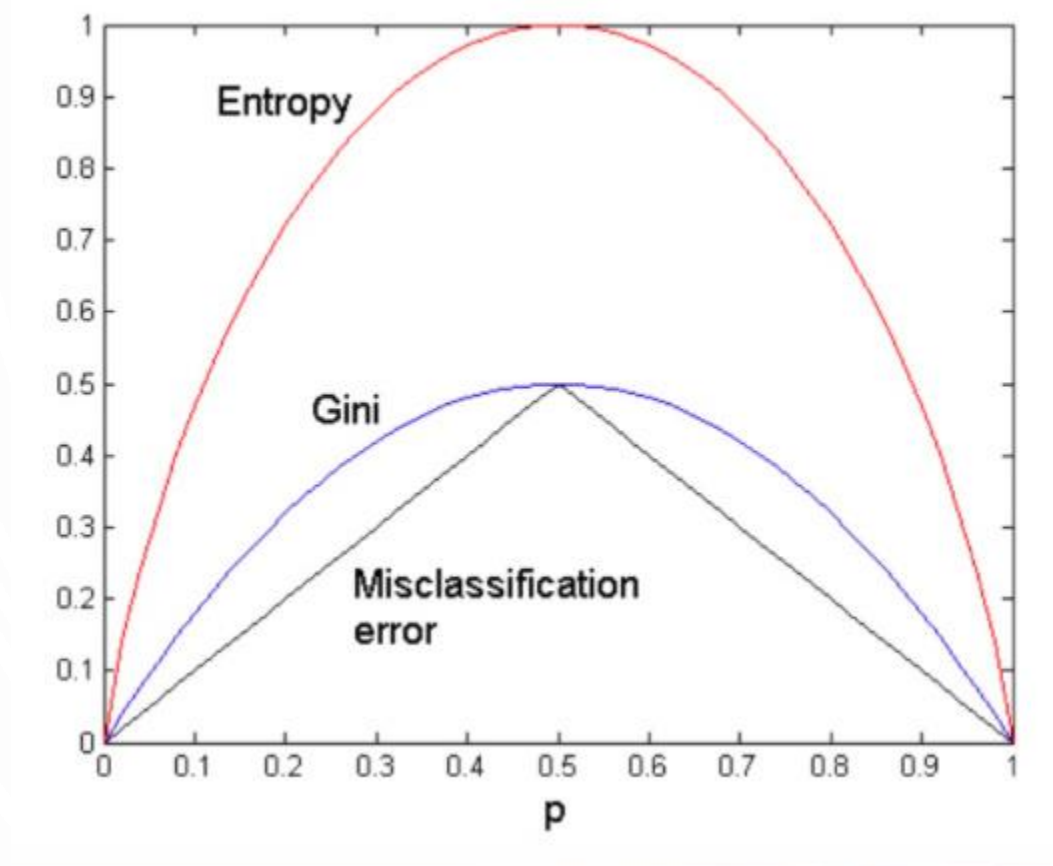
$$c = (c_1, \dots, c_K), \sum_i c_i = 1.$$

В данной задаче для самыми популярными функционалами $H(R)$, которые мы минимизируем при построении решающего дерева, являются:

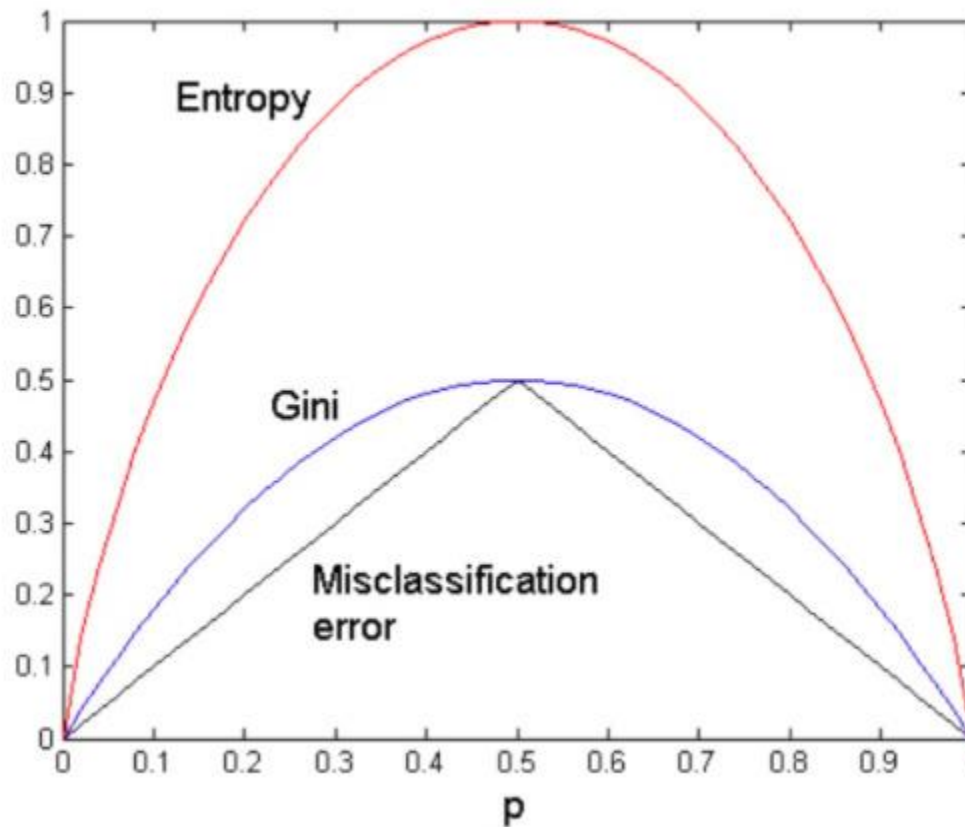
1) $H(R) = \sum_{k=1}^K p_k(1 - p_k)$ (критерий Джини)

2) $H(R) = -\sum_{k=1}^K p_k \log p_k$ (энтропия)

КРИТЕРИЙ ДЖИНИ И ЭНТРОПИЯ

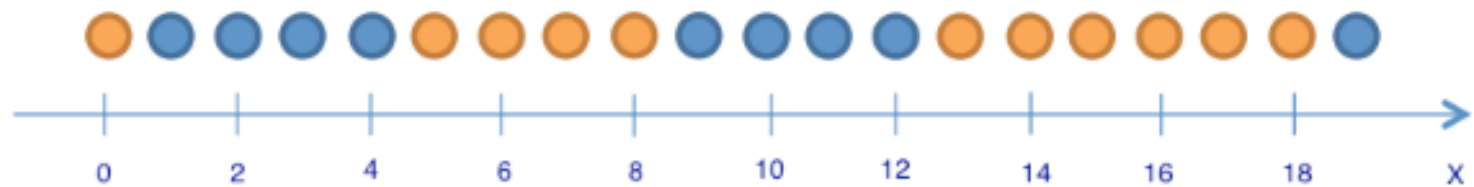


КРИТЕРИЙ ДЖИНИ И ЭНТРОПИЯ



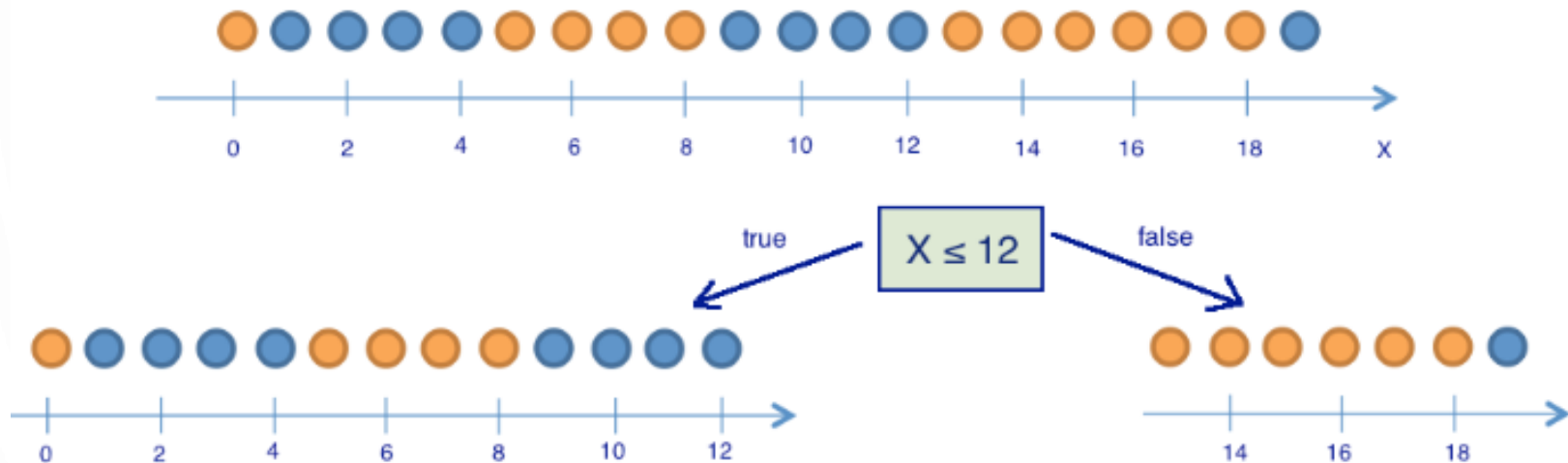
- Минимум $H(R)$ достигается на распределении $p_i = 1, p_j = 0, j \neq i$
- Максимум $H(R)$ достигается на равномерном распределении $p_1 = \dots = p_K = \frac{1}{K}$.

ПРИМЕР ИСПОЛЬЗОВАНИЯ ЭНТРОПИЙНОГО КРИТЕРИЯ



- $p_1 = \frac{9}{20}, p_2 = \frac{11}{20} \Rightarrow$ энтропия $H_0 = -\frac{9}{20} \log \frac{9}{20} - \frac{11}{20} \log \frac{11}{20} \approx 1$

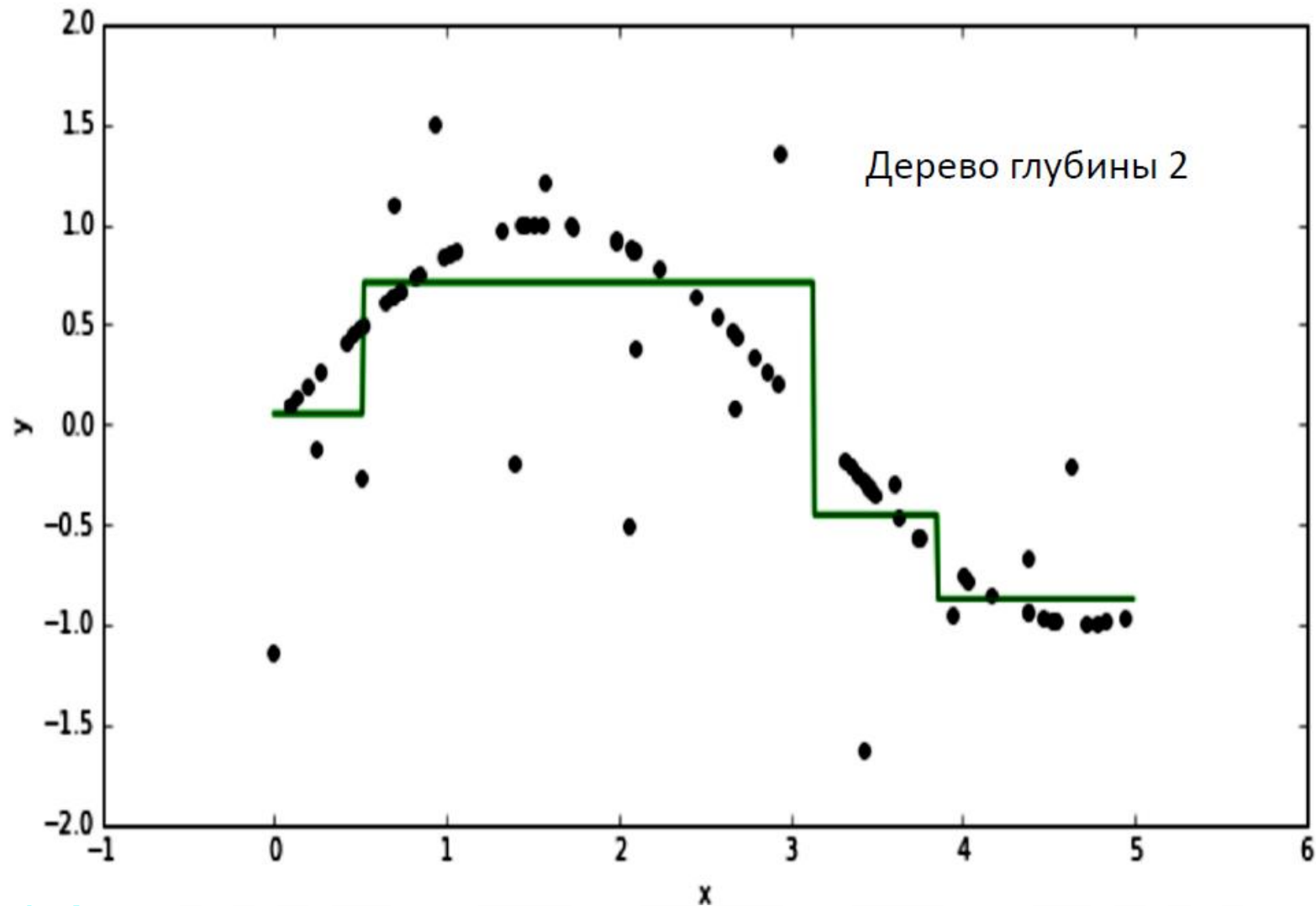
ПРИМЕР ИСПОЛЬЗОВАНИЯ ЭНТРОПИЙНОГО КРИТЕРИЯ



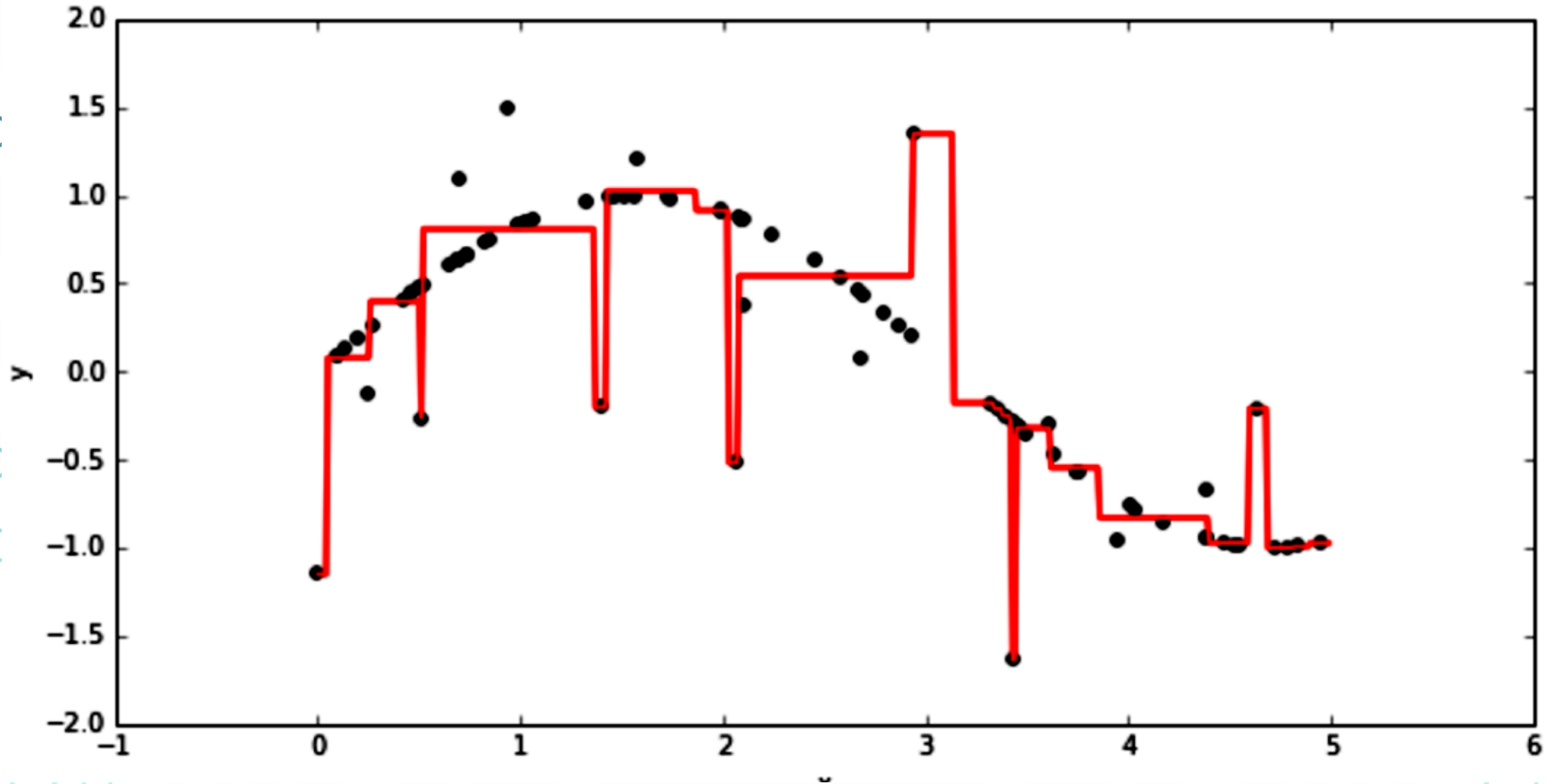
- В левой части $H_l = -\frac{5}{13} \log \frac{5}{13} - \frac{8}{13} \log \frac{8}{13} \approx 0.96$
- В правой части $H_r = -\frac{1}{7} \log \frac{1}{7} - \frac{6}{7} \log \frac{6}{7} \approx 0.6$

То есть $Q = H_0 - \frac{|R_l|}{R} H_l - \frac{|R_r|}{|R|} H_r = 1 - \frac{13}{20} \cdot 0.96 - \frac{7}{20} \cdot 0.6 \approx 0.16$

ПРИМЕР: ДЕРЕВО ГЛУБИНЫ 2 В ЗАДАЧЕ РЕГРЕССИИ



ПРИМЕР: ДЕРЕВО ГЛУБИНЫ 5 В ЗАДАЧЕ РЕГРЕССИИ



ПЛЮСЫ РЕШАЮЩИХ ДЕРЕВЬЕВ

- Четкие правила классификации (интерпретируемые предикаты, например, “возраст > 25 ”)
- Деревья решений легко визуализируются, то есть хорошо интерпретируются
- Быстро обучаются и выдают прогноз
- Малое число параметров

МИНУСЫ РЕШАЮЩИХ ДЕРЕВЬЕВ

- Очень чувствительны к шумам в данных, модель сильно меняется при небольшом изменении обучающей выборки
- Разделяющая граница имеет свои ограничения (состоит из гиперплоскостей)
- Необходимость борьбы с переобучением (стрижка или какой-либо из критериев останова)
- Проблема поиска оптимального дерева (NP-полная задача, поэтому на практике используется жадное построение дерева)