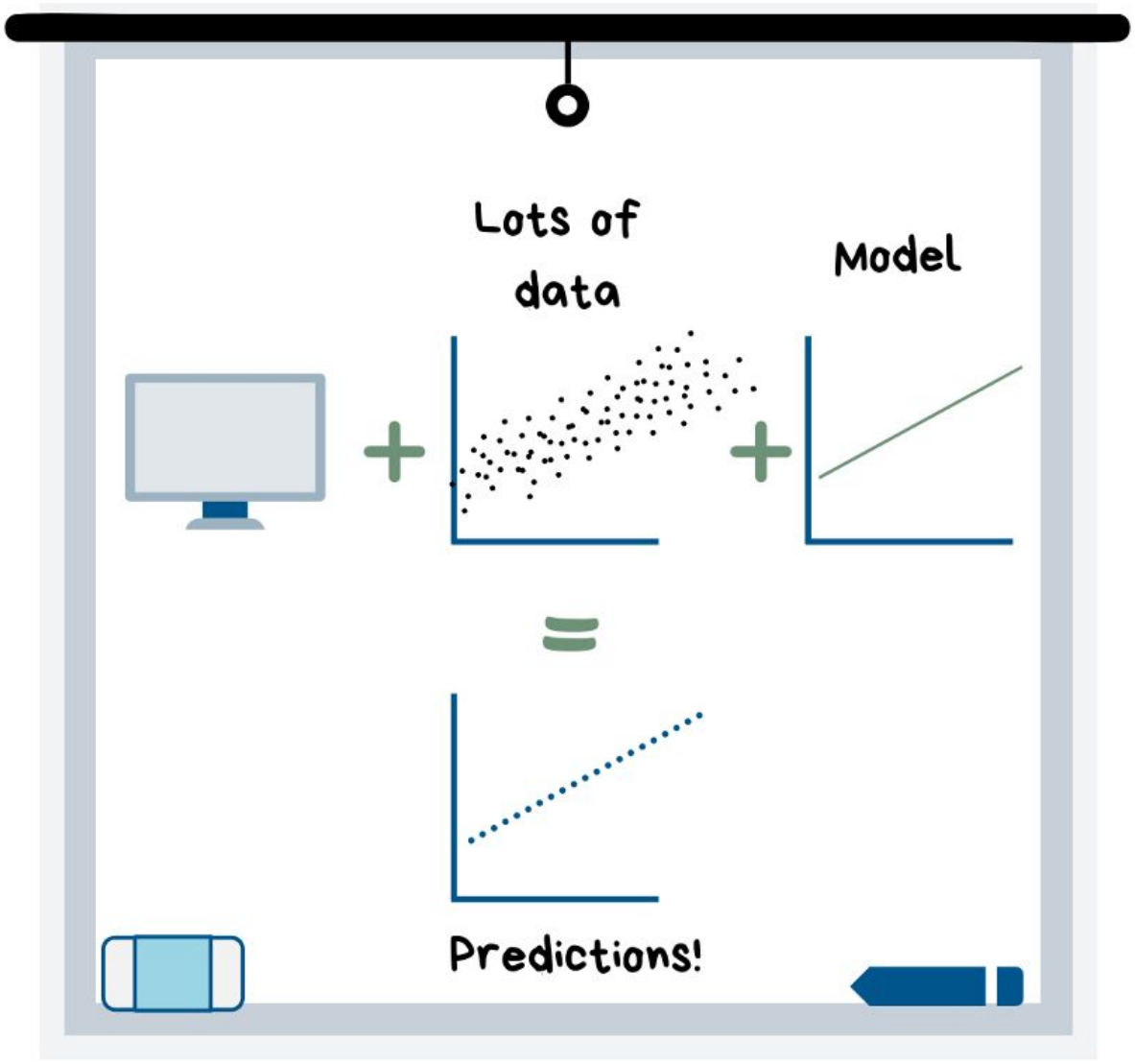
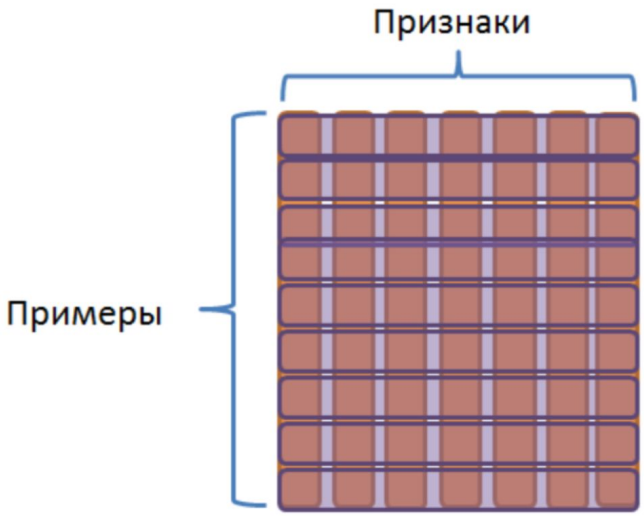
Курсы по машинному обучению

Тема 4. Деревья решений

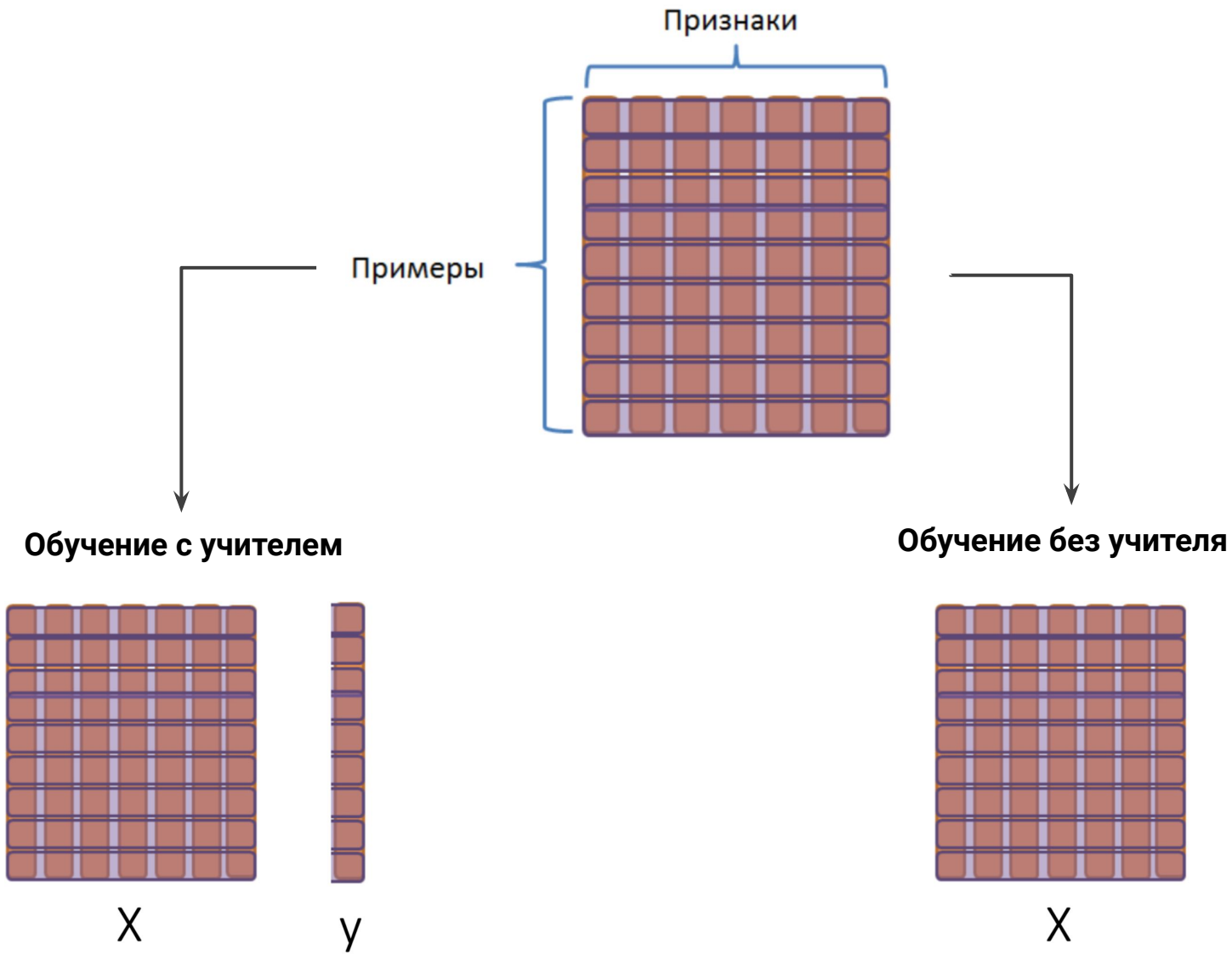
Machine Learning



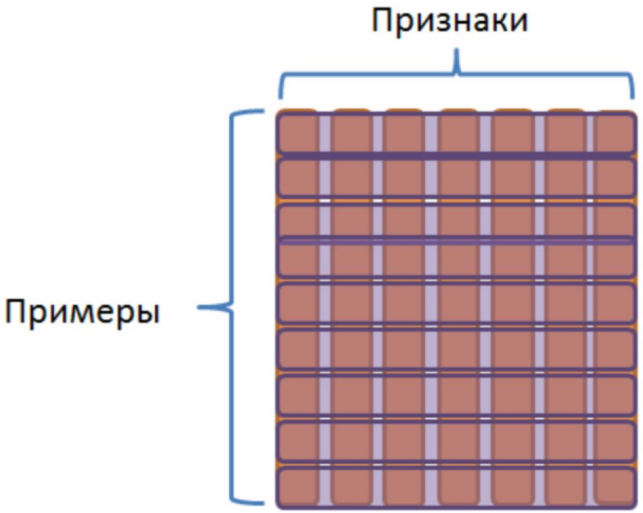
Обучение с учителем и без



Обучение с учителем и без

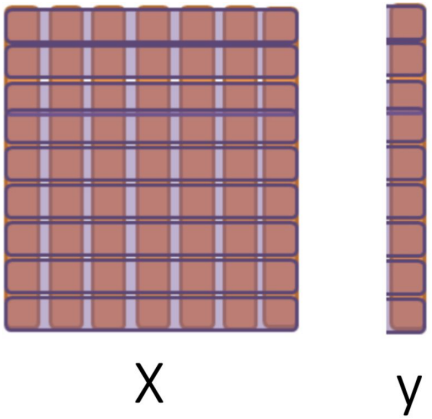


Обучение с учителем



Классификация

Кошка	1	0.13
Собака	0	0.19
Собака	0	0.98
Собака	0	0.14
Рыбка	2	0.57
Собака	0	0.42
Слон	3	0.39
Слон	3	0.02



Регрессия

30000	- 0.13	7
45000	0.19	57
45000	0.98	19
37500	0.14	24
18000	- 0.57	21
30000	0.42	35
180000	0.39	39
95000	0.02	27

Постановка задачи классификации

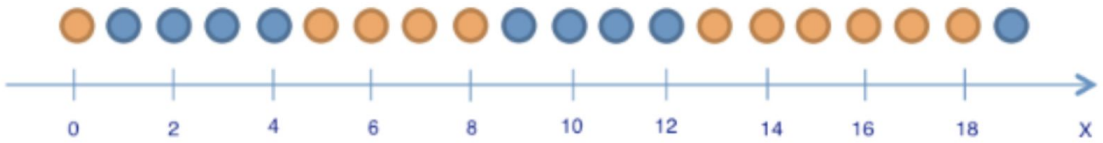
Пусть дан набор объектов $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$, $\mathbf{x}_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, $i \in 1, \ldots, N$, полученный из неизвестной закономерности $y = f(\mathbf{x})$. Необходимо построить такую $h(\mathbf{x})$, которая наиболее точно апроксимирует $f(\mathbf{x})$.

Будем искать неизвестную

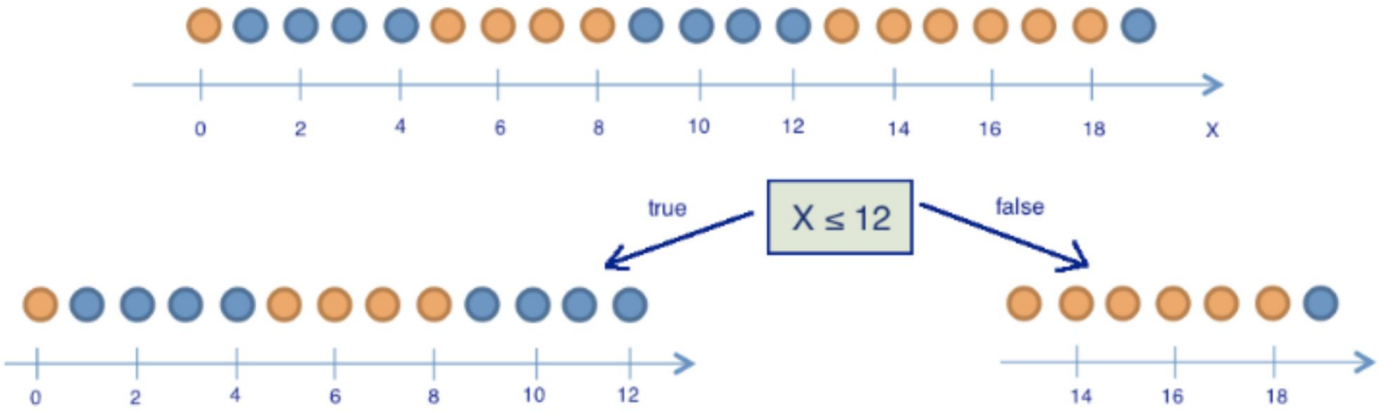
$$h(\mathbf{x}) = h(a_1, \ldots, a_T)$$

Деревья решений

Деревья решений

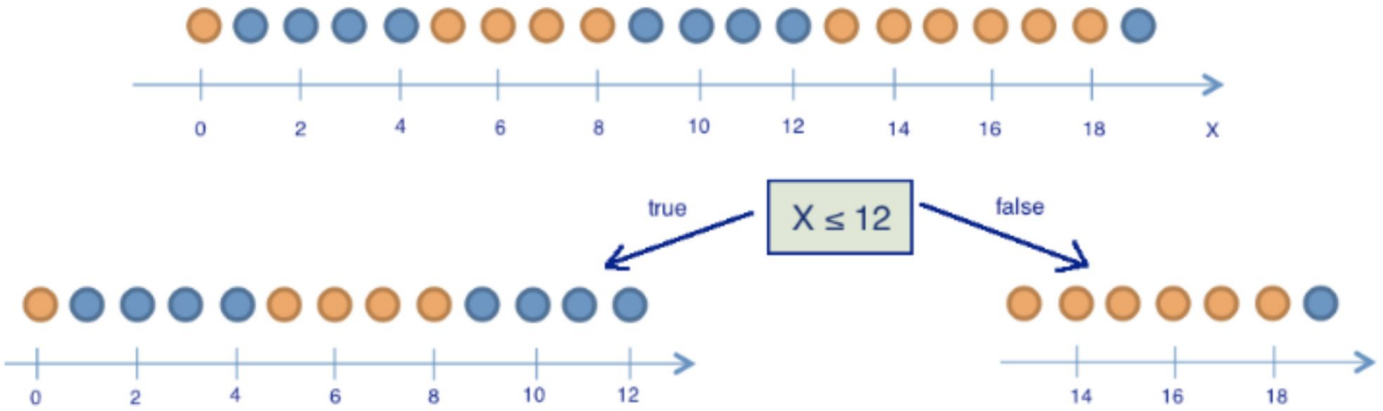


Деревья решений



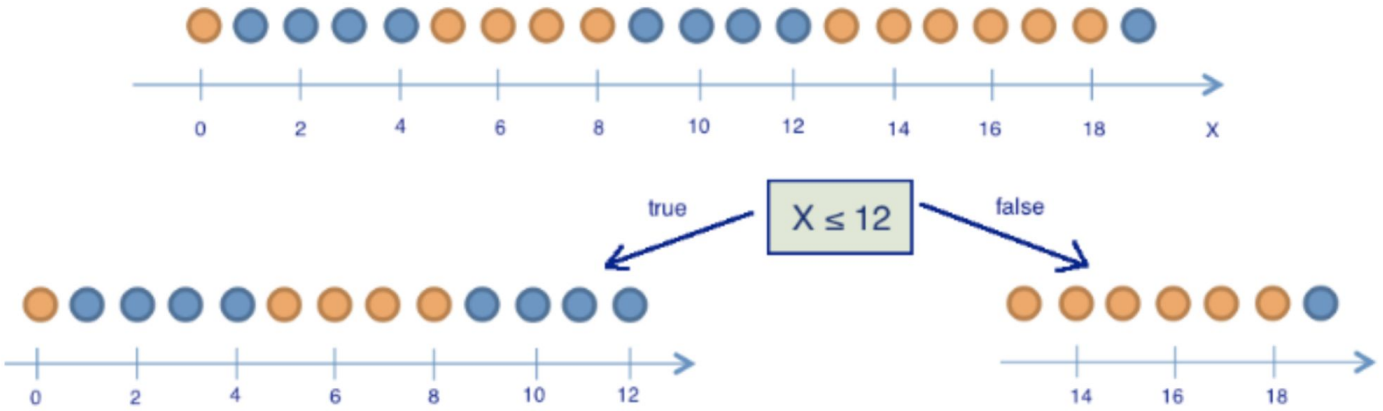
Деревья решений

Оранжевые: $p_1 = \frac{9}{20}$ Синие: $p_2 = \frac{11}{20}$



Деревья решений

Оранжевые: $p_1 = \frac{9}{20}$ Синие: $p_2 = \frac{11}{20}$



Энтропия Шеннона

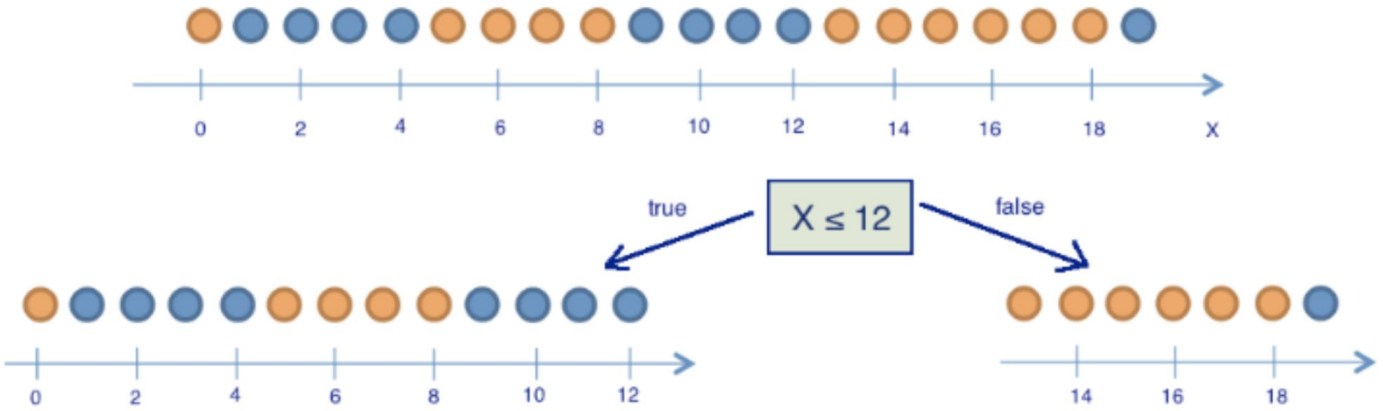
$$-\sum_{i=1}^{N} p_i \log_2(p_i)$$

Индекс Джини

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

Деревья решений

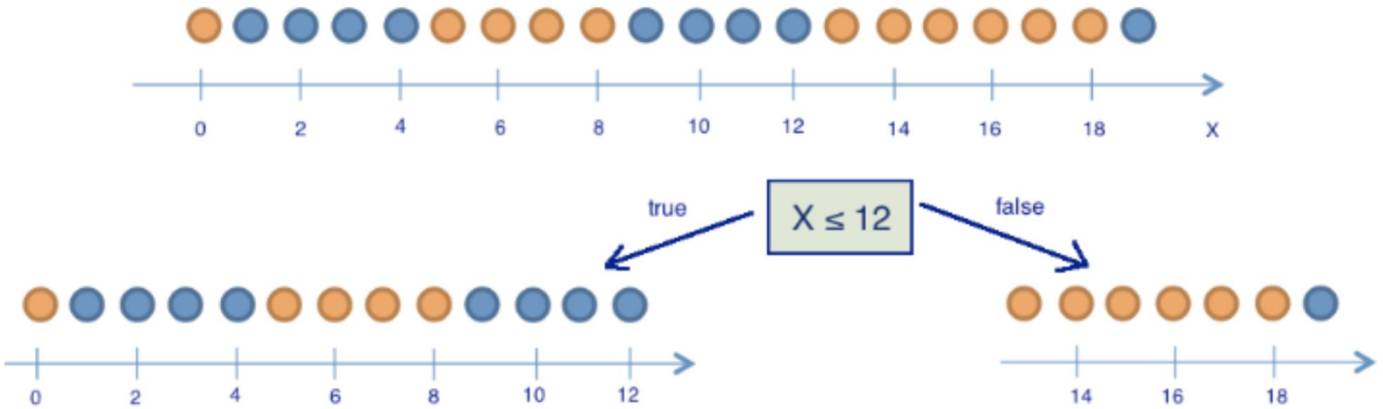
Оранжевые: $p_1 = \frac{9}{20}$ Синие: $p_2 = \frac{11}{20}$ $S_0 = -\frac{9}{20} log_2\left(\frac{9}{20}\right) - \frac{11}{20} log_2\left(\frac{11}{20}\right) = 0.993$



$S_1 = -\frac{5}{13} log_2\left(\frac{5}{13}\right) - \frac{8}{13} log_2\left(\frac{8}{13}\right) = 0.96$ $S_2 = -\frac{1}{7} log_2\left(\frac{1}{7}\right) - \frac{6}{7} log_2\left(\frac{6}{7}\right) = 0.59$

Деревья решений

Оранжевые: $p_1 = \frac{9}{20}$ Синие: $p_2 = \frac{11}{20}$ $S_0 = -\frac{9}{20} log_2\left(\frac{9}{20}\right) - \frac{11}{20} log_2\left(\frac{11}{20}\right) = 0.993$



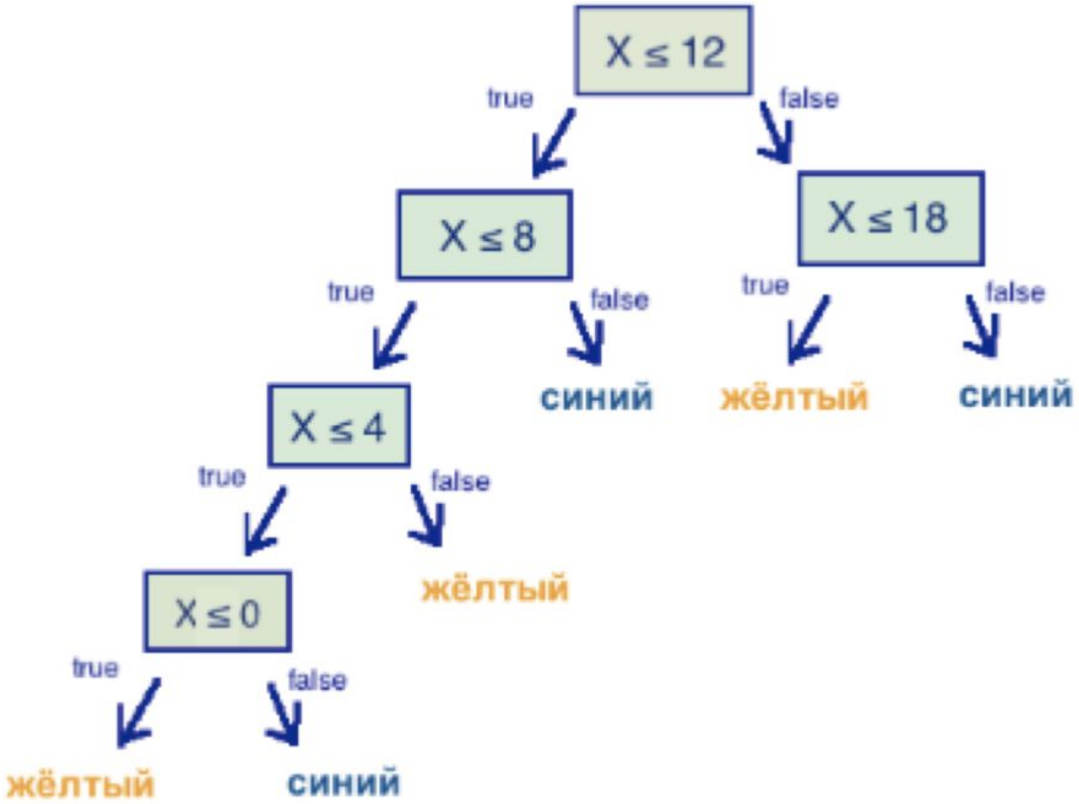
$S_1 = -\frac{5}{13} log_2\left(\frac{5}{13}\right) - \frac{8}{13} log_2\left(\frac{8}{13}\right) = 0.96$ $S_2 = -\frac{1}{7} log_2\left(\frac{1}{7}\right) - \frac{6}{7} log_2\left(\frac{6}{7}\right) = 0.59$

$$\text{IG}(\text{“}X \leq 12\text{”}) = S_0 - \frac{13}{20} S_1 - \frac{7}{20} S_2 = 0.163$$

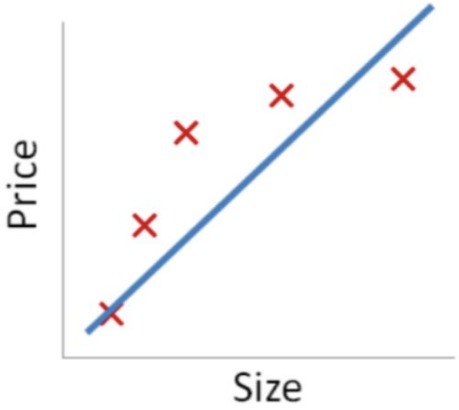
Алгоритм

Обходим все варианты и находим разбиение с наибольшим Information Gain (IG). После того повторяем операцию для каждого из разбиений, пока все объекты из разбиения не будут одного класса.

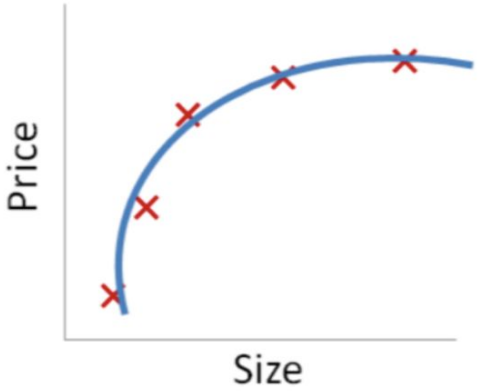
Дерево решений



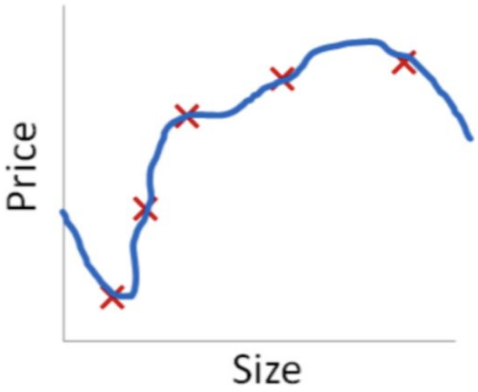
Переобучение



$$\theta_0 + \theta_1 x$$

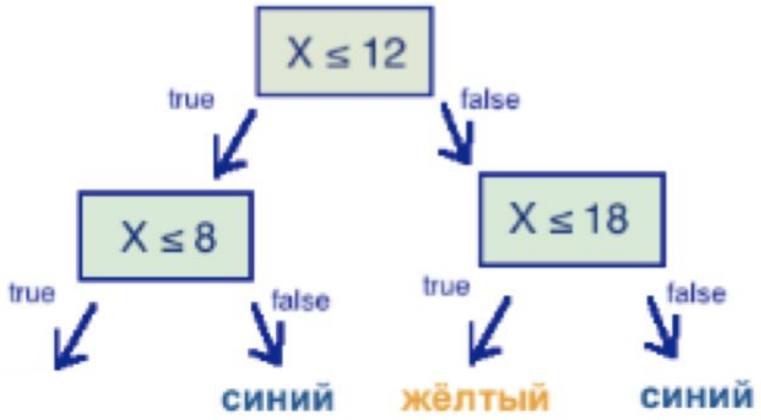


$$\theta_0 + \theta_1 x + \theta_2 x^2$$

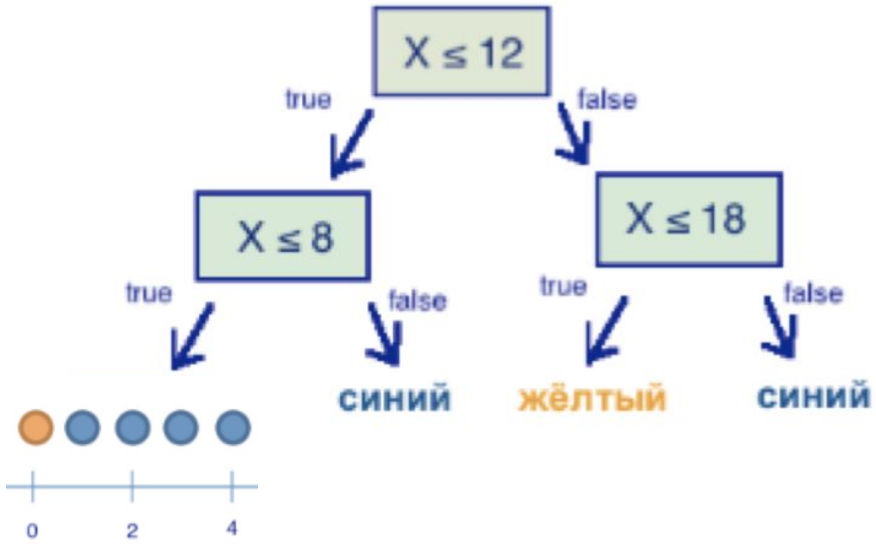


$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

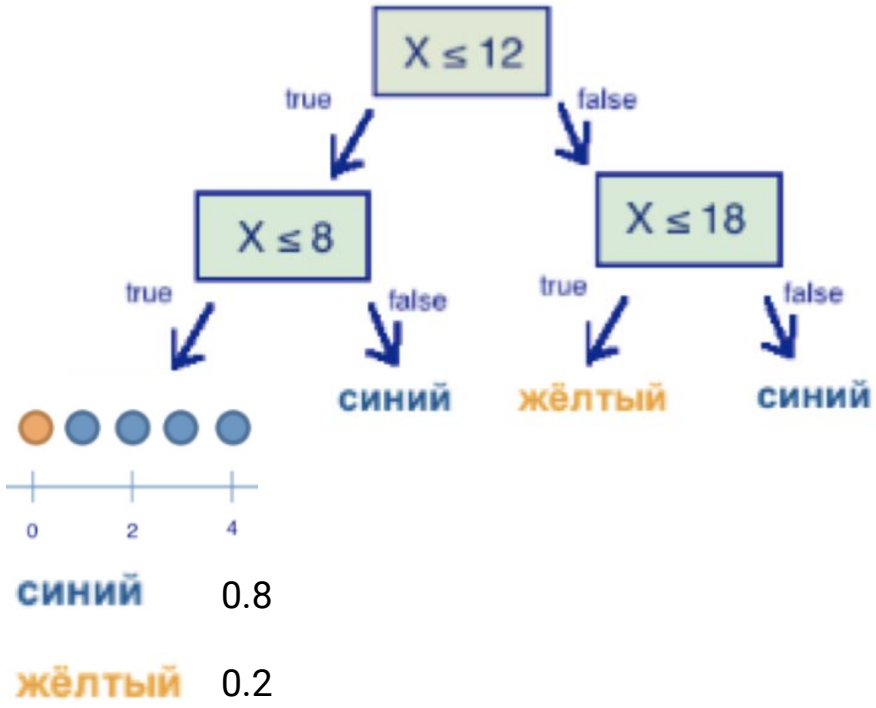
Дерево решений



Дерево решений



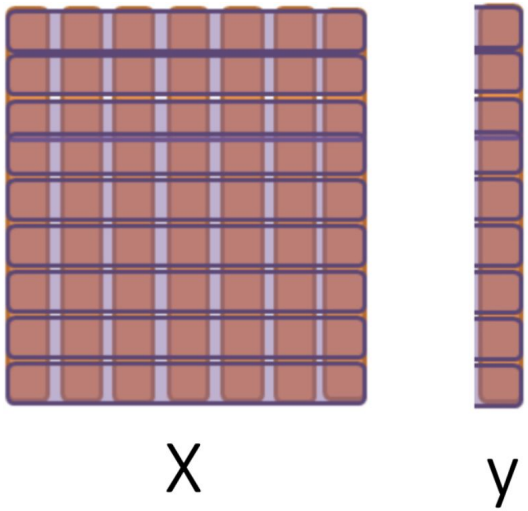
Дерево решений



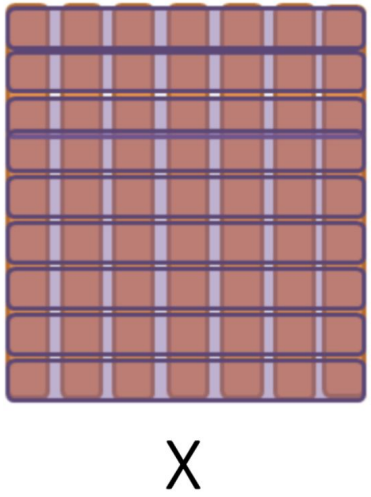
Дерево решений

```
def build(L):
    create node t
    if the stopping criterion is True:
        assign a predictive model to t
    else:
        Find the best binary split L = L_left + L_right
        t.left = build(L_left)
        t.right = build(L_right)
    return t
```

Обучающая и тестовая выборки

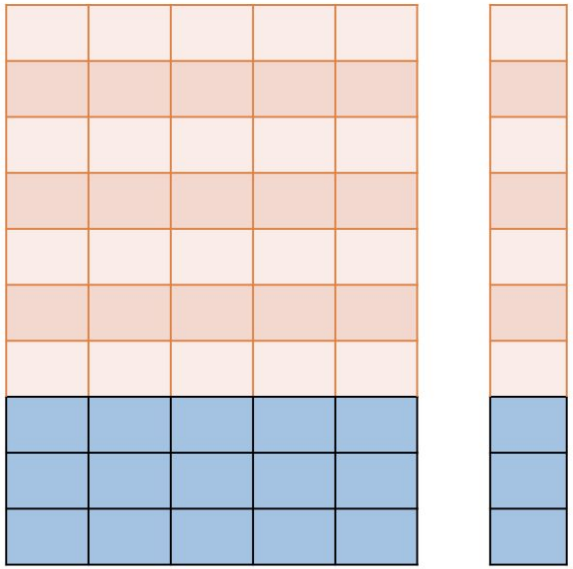


Обучающая выборка



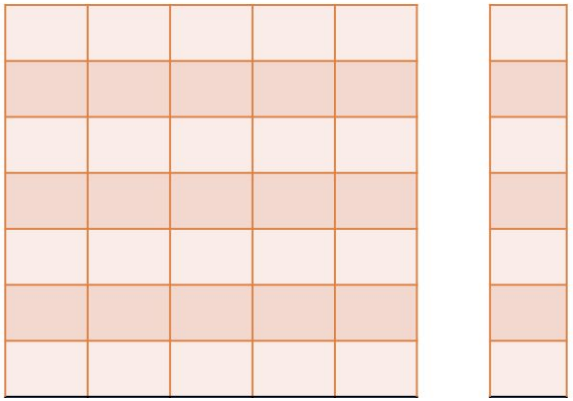
Тестовая выборка

Обучающая и тестовая выборки

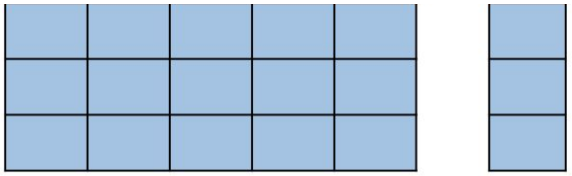


Обучающая выборка

Обучающая и тестовая выборки

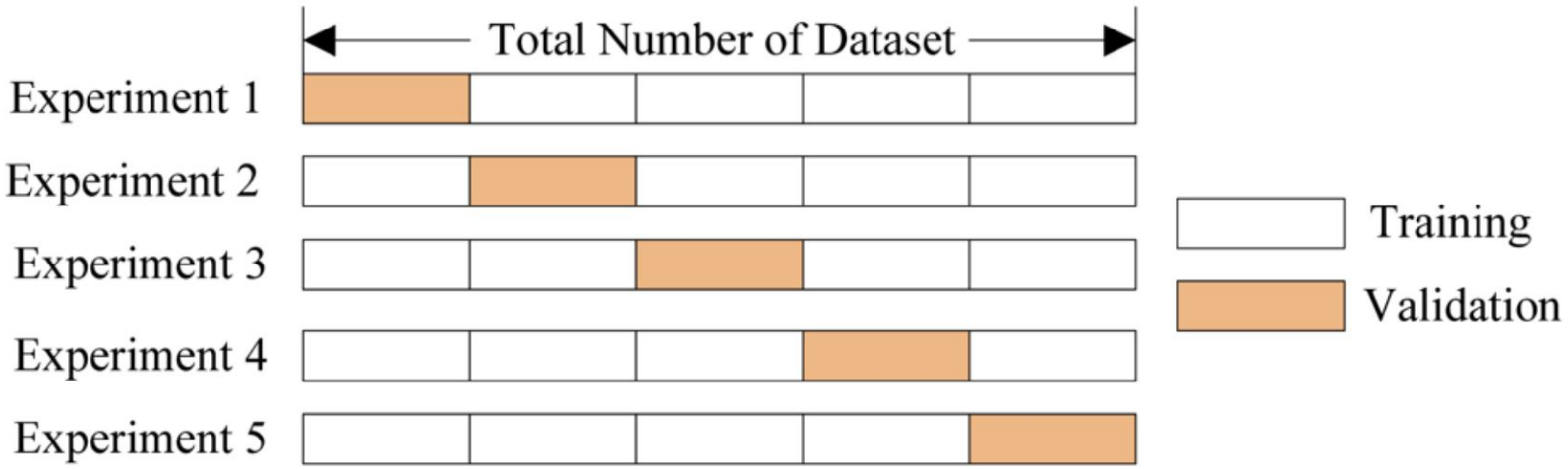


Обучающая выборка
(X_train, y_train)

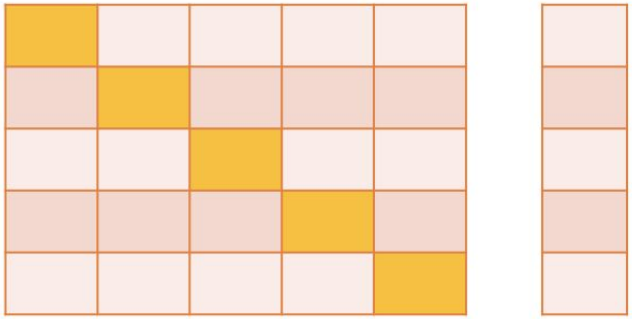


Тестовая выборка
(X_test, y_test)

Кросс-валидация

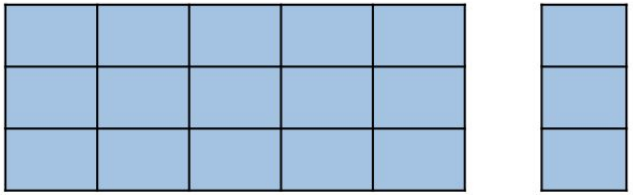


Финальный пайплайн



Кросс-валидация на обучающей выборке

+



Результат работы на отложенной выборке

Дерево решений

Плюсы

- Интерпретируемость
- Отлично подходит как базовый алгоритм для ансамбля моделей
- Мало чувствителен к выбросам
- Высокая скорость работы
- Не требует сложной предобработки данных
- Можно оценить модель с помощью статистических тестов

Минусы

- Очень склонен к переобучению
- Изменяет всю структуру дерева от небольших изменений в выборке
- Алгоритм построен на эвристиках
- Слабый алгоритм