

# Введение в анализ данных

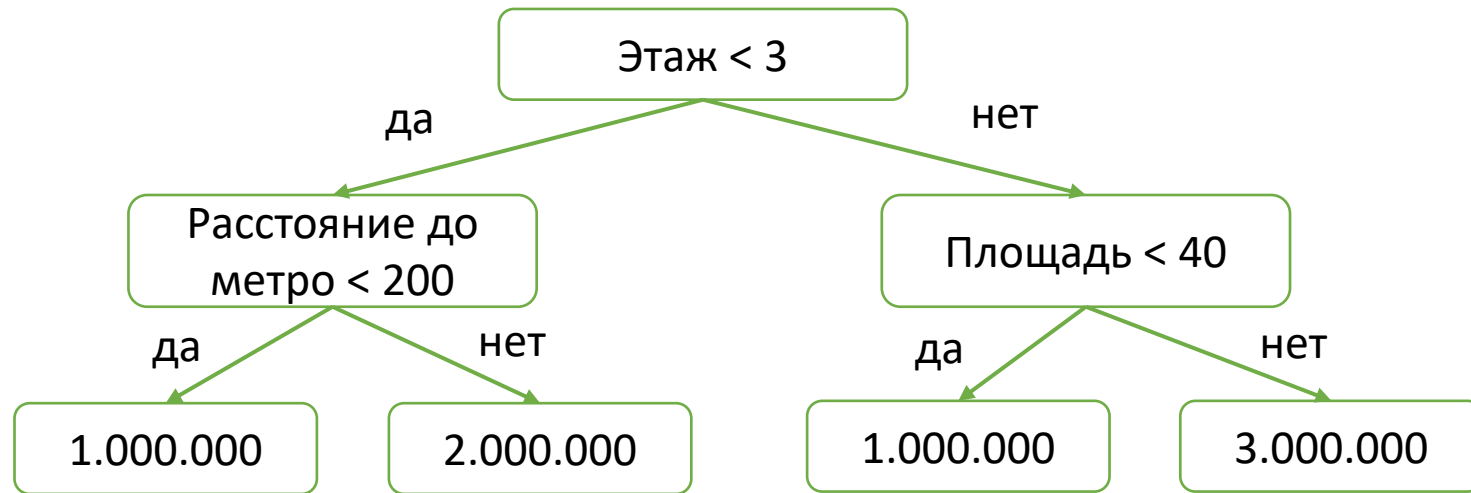
Лекция 7. Решающие деревья

Решающие деревья

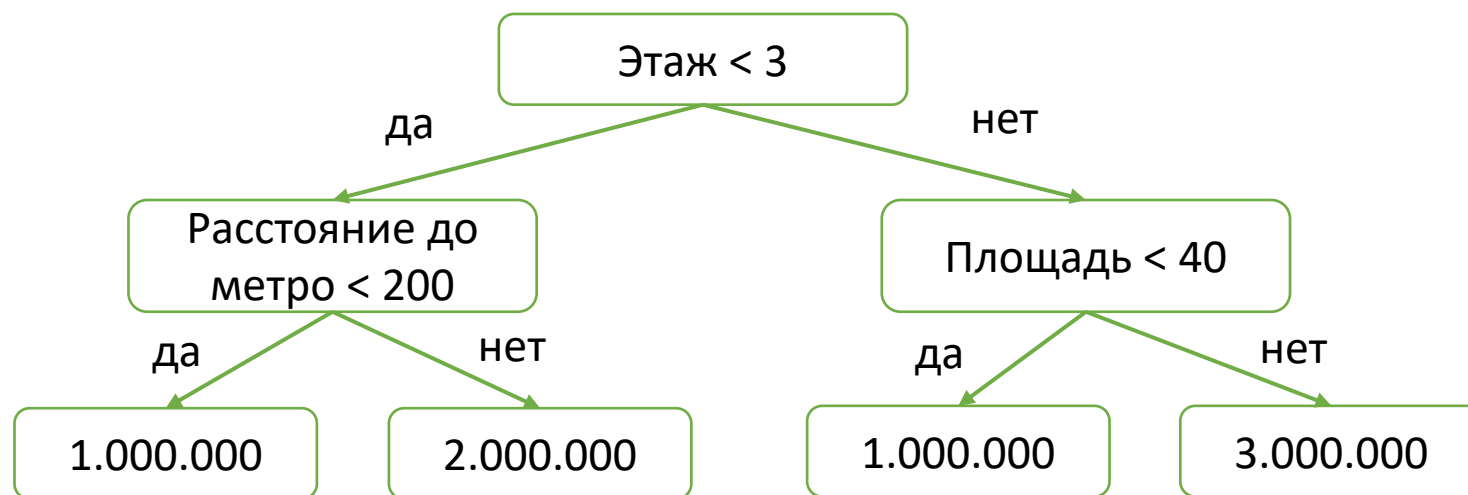
# Логические правила

- $[30 < \text{площадь} < 50][2 < \text{этаж} < 5][500 < \text{расстояние до метро} < 1000]$
- Легко объяснить, как работают
- Находят нелинейные закономерности
- Нужно как-то искать хорошие логические правила
- Нужно уметь составлять модели из логических правил

# Решающее дерево

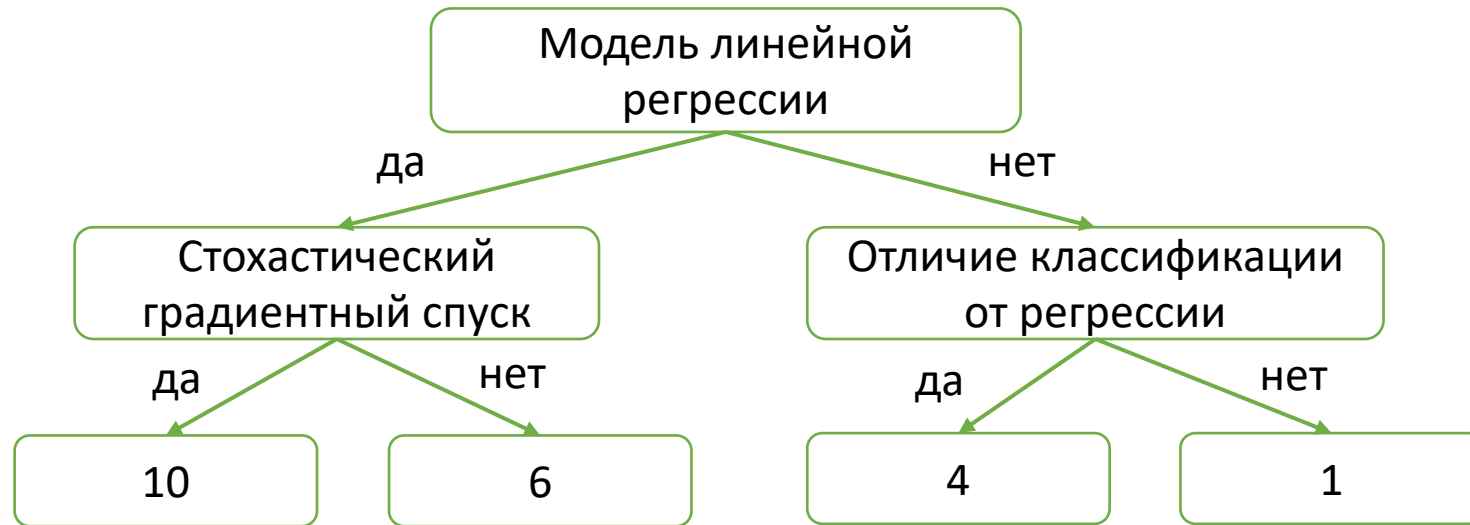


# Решающее дерево

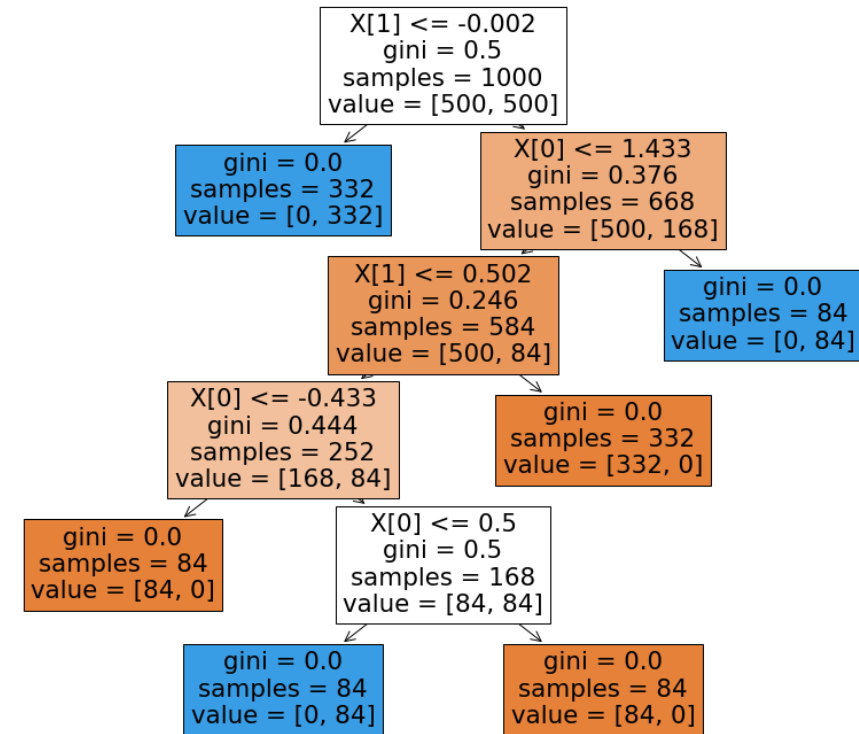
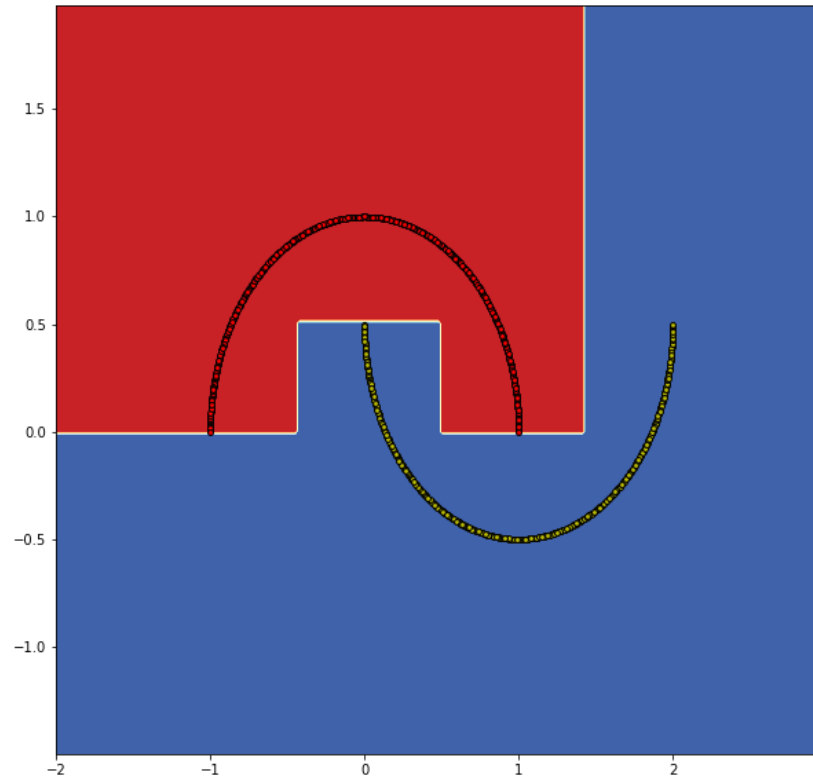


- Внутренние вершины: предикаты  $[x_j < t]$
- Листья: прогнозы  $s \in \mathbb{Y}$

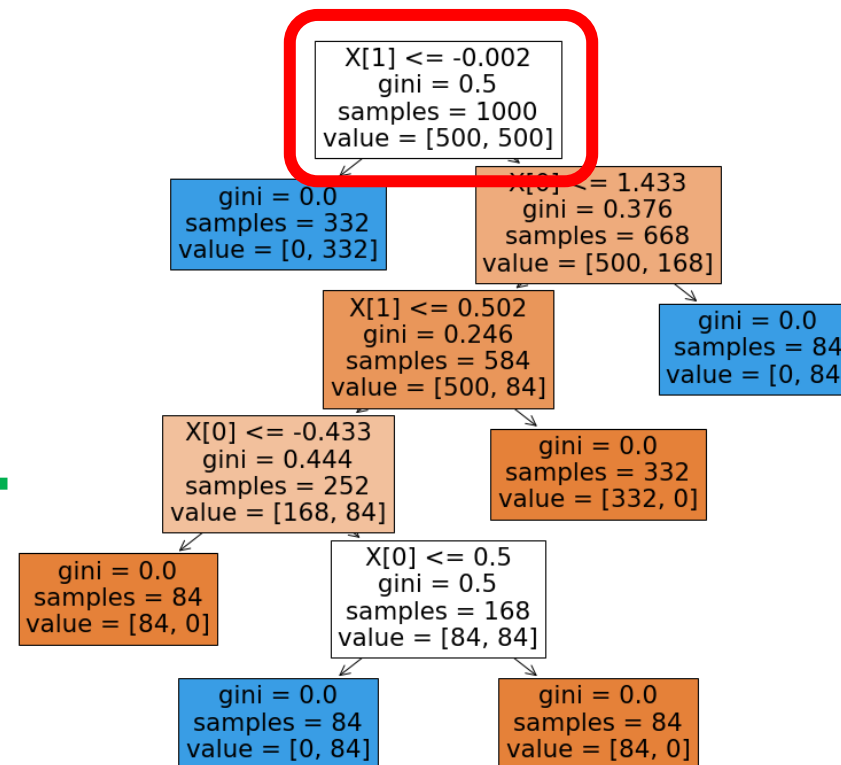
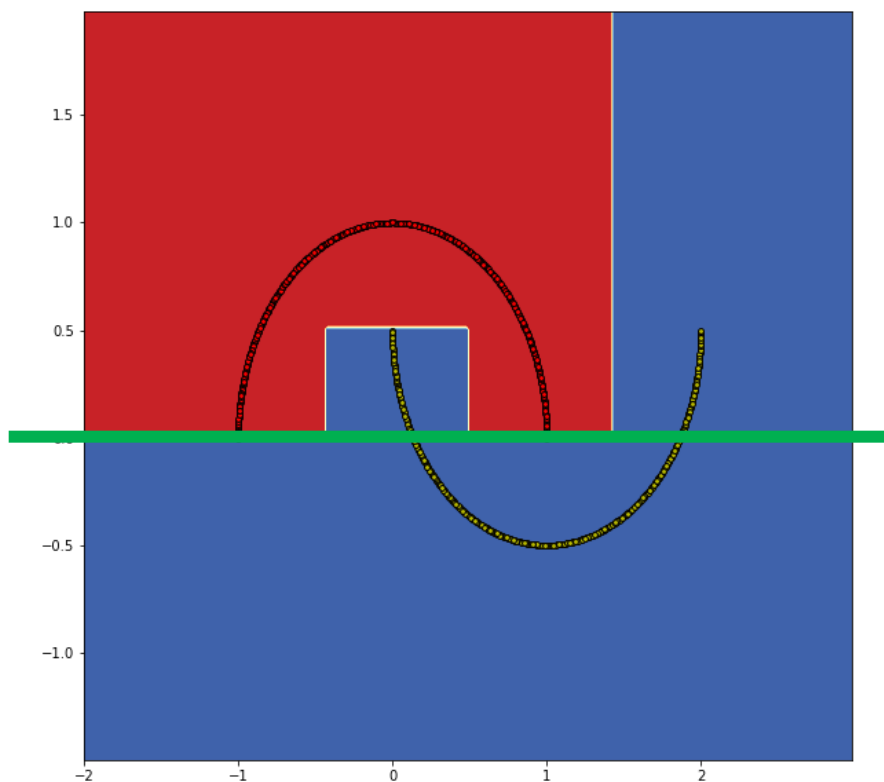
# Решающее дерево



# Решающее дерево

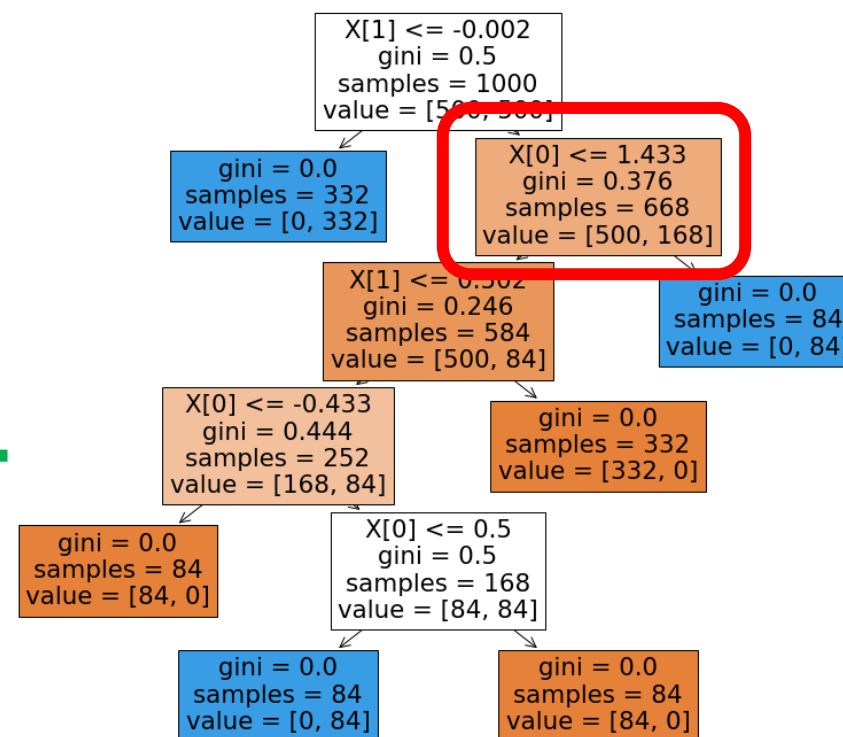
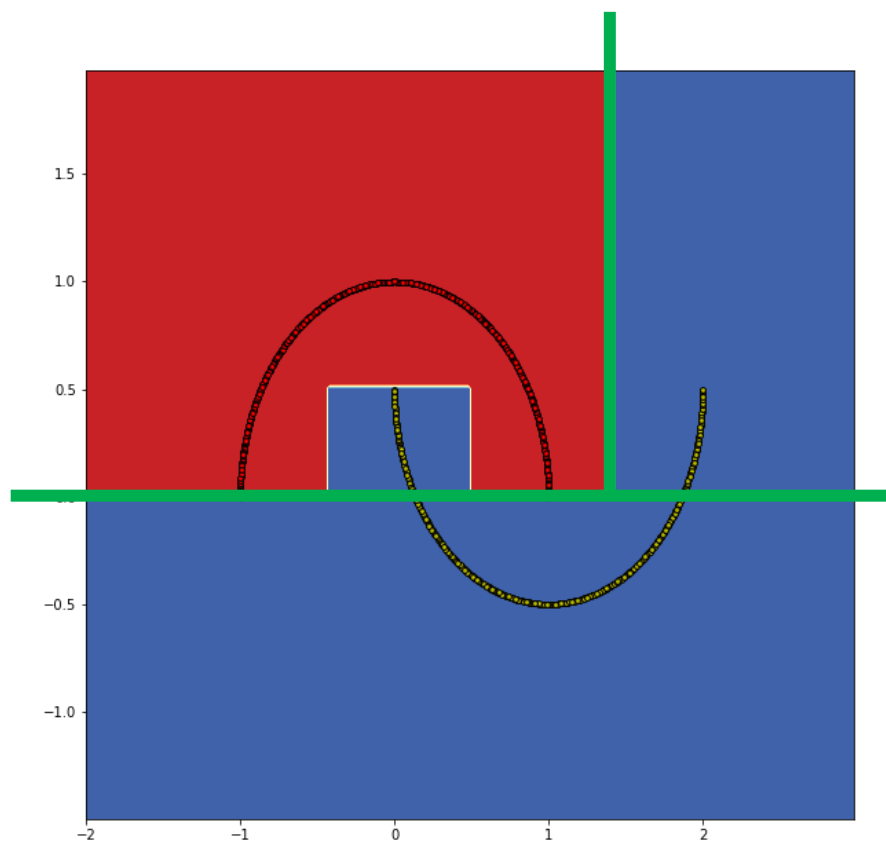


# Решающее дерево

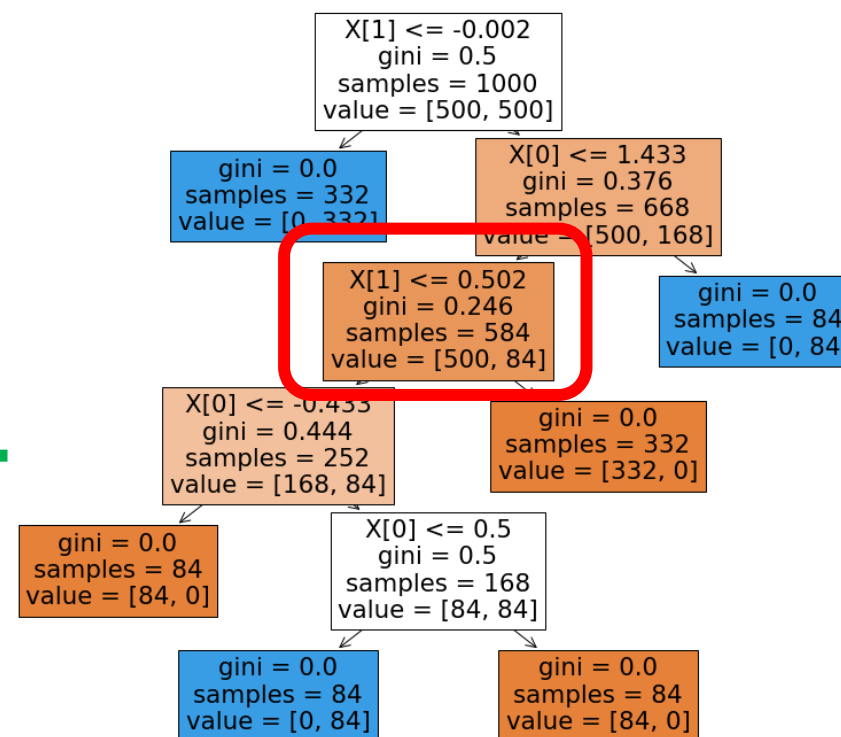
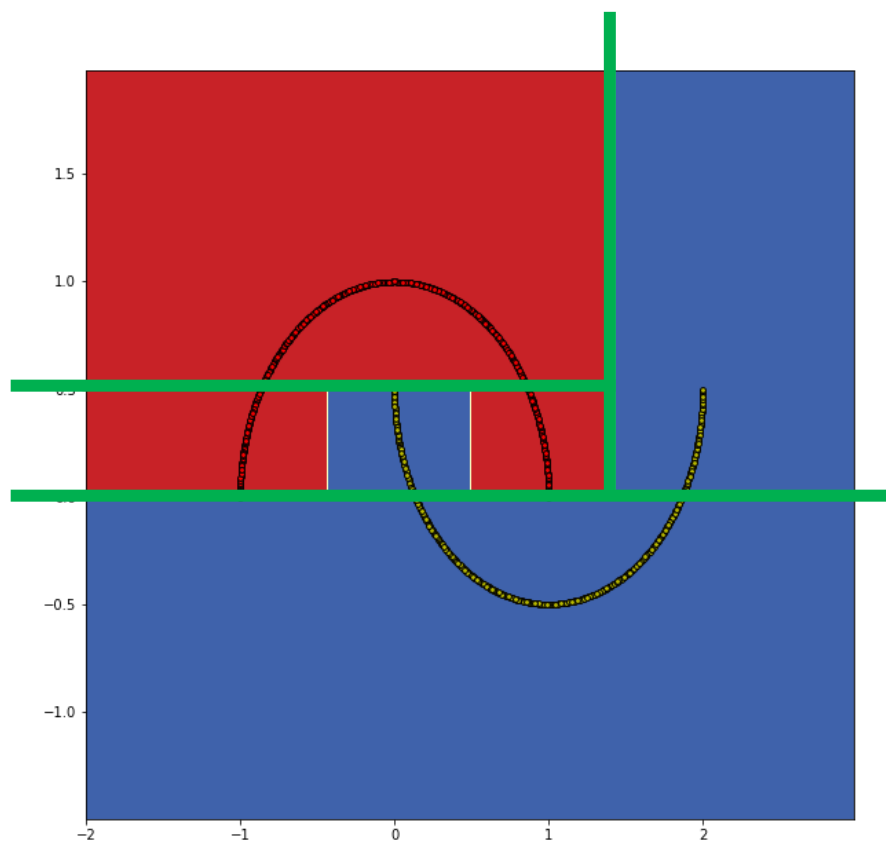




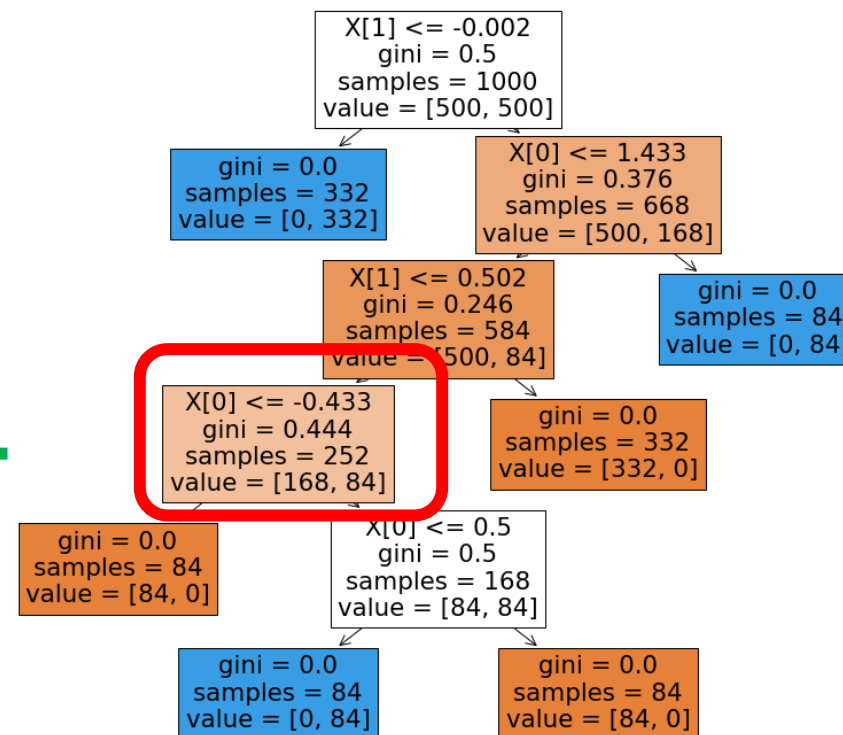
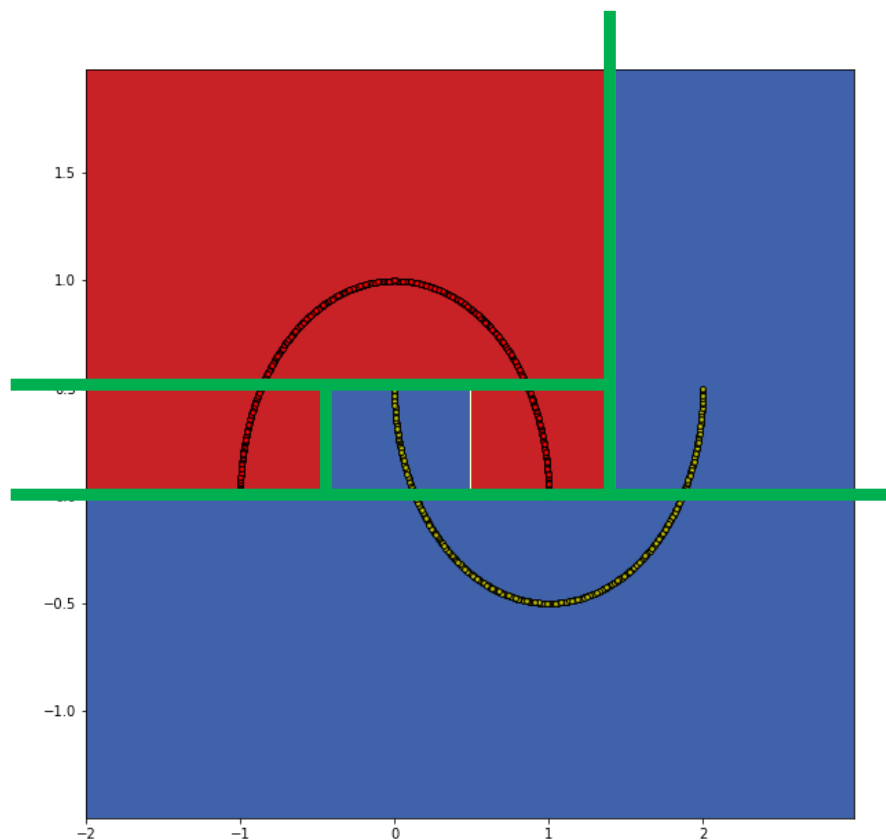
# Решающее дерево



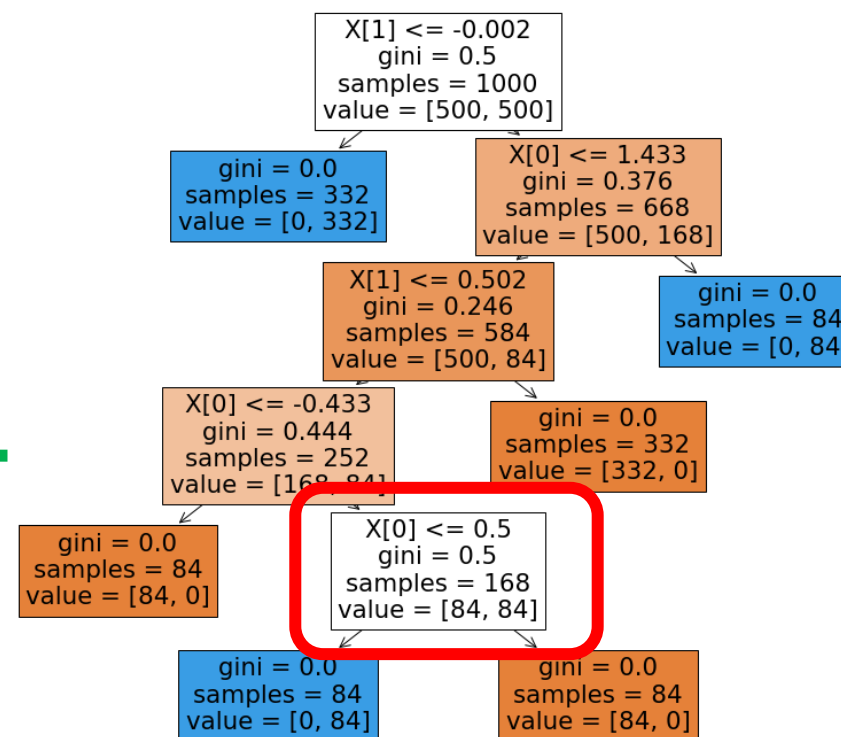
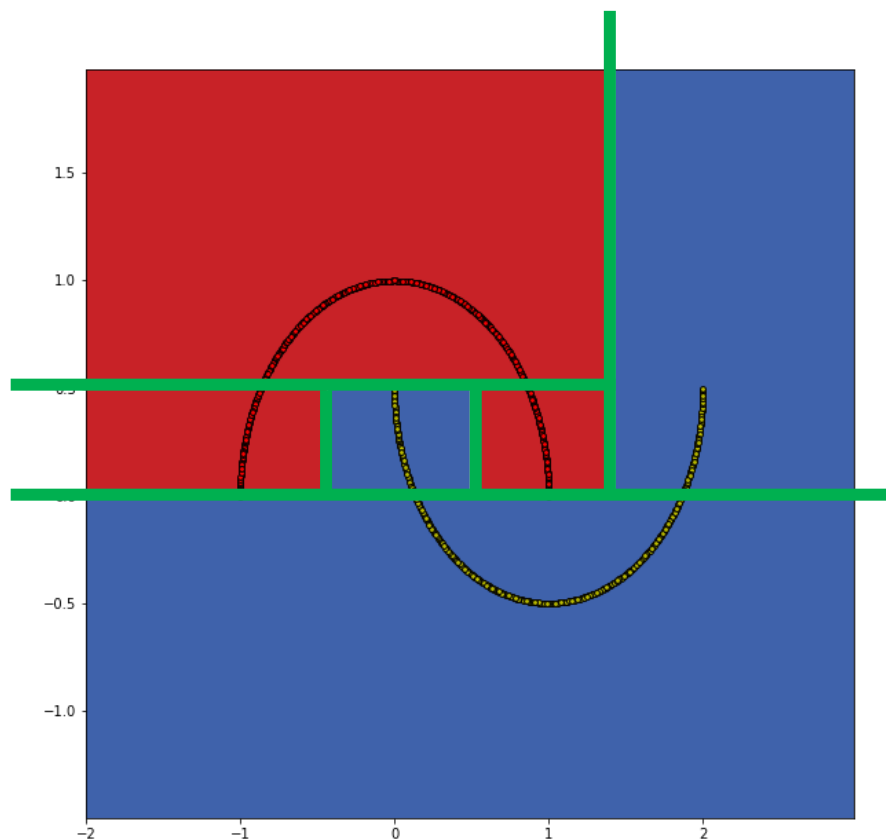
# Решающее дерево



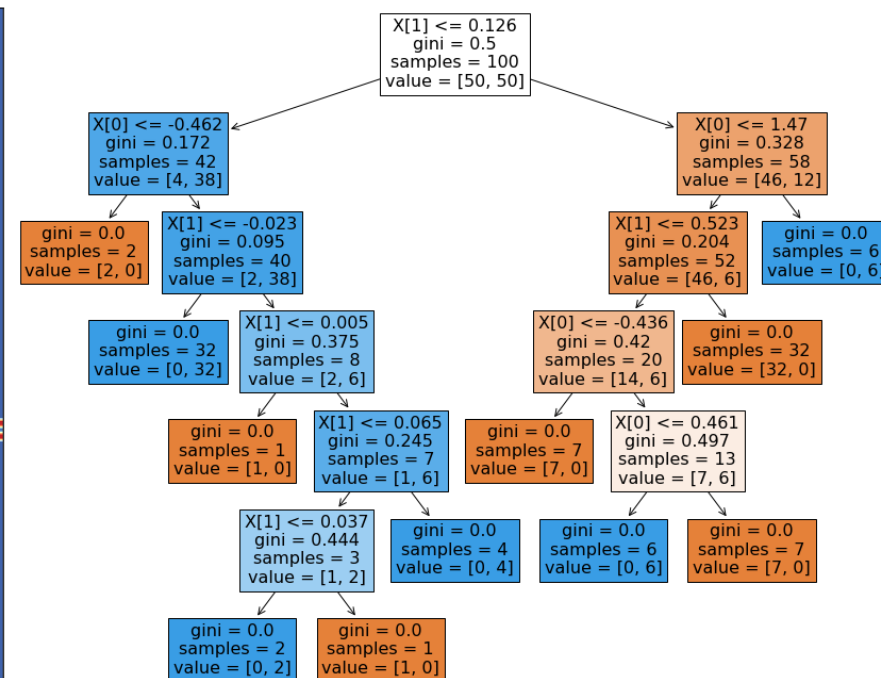
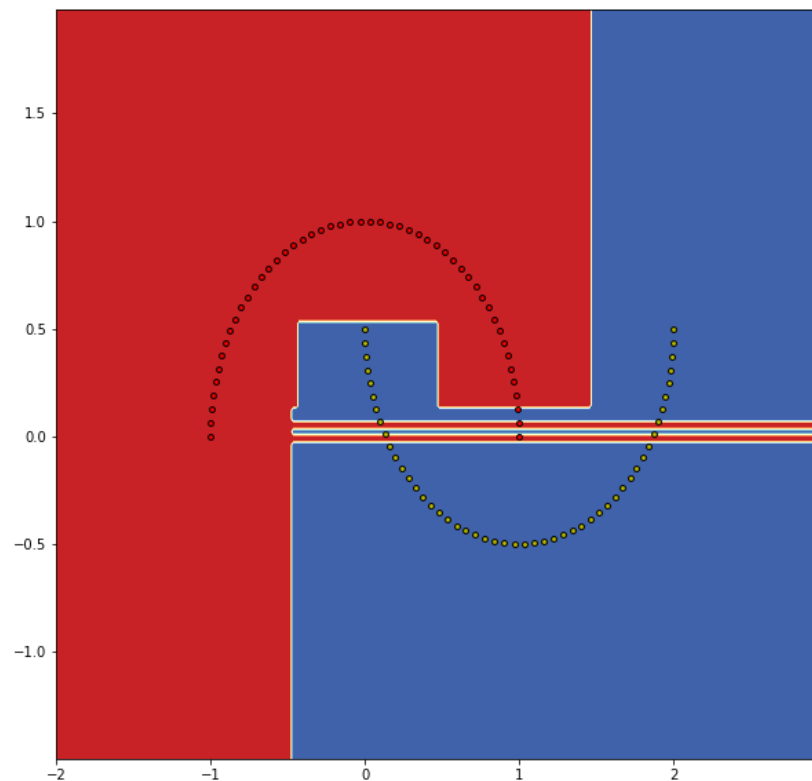
# Решающее дерево



# Решающее дерево



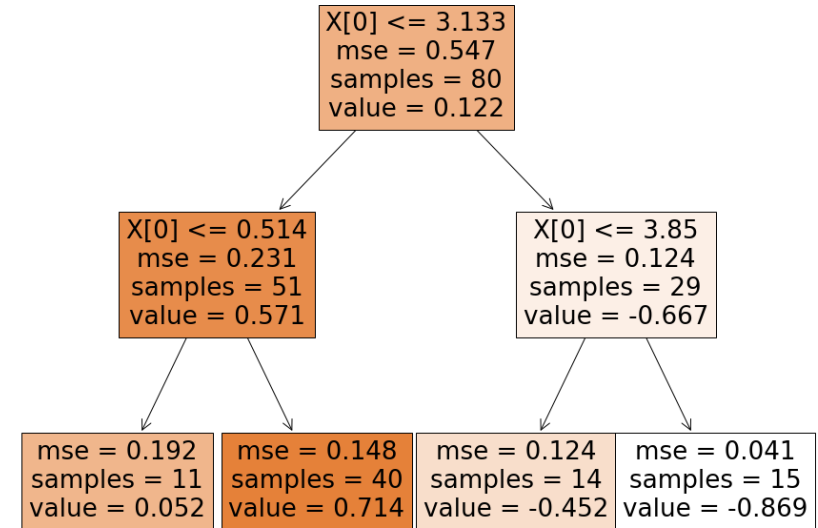
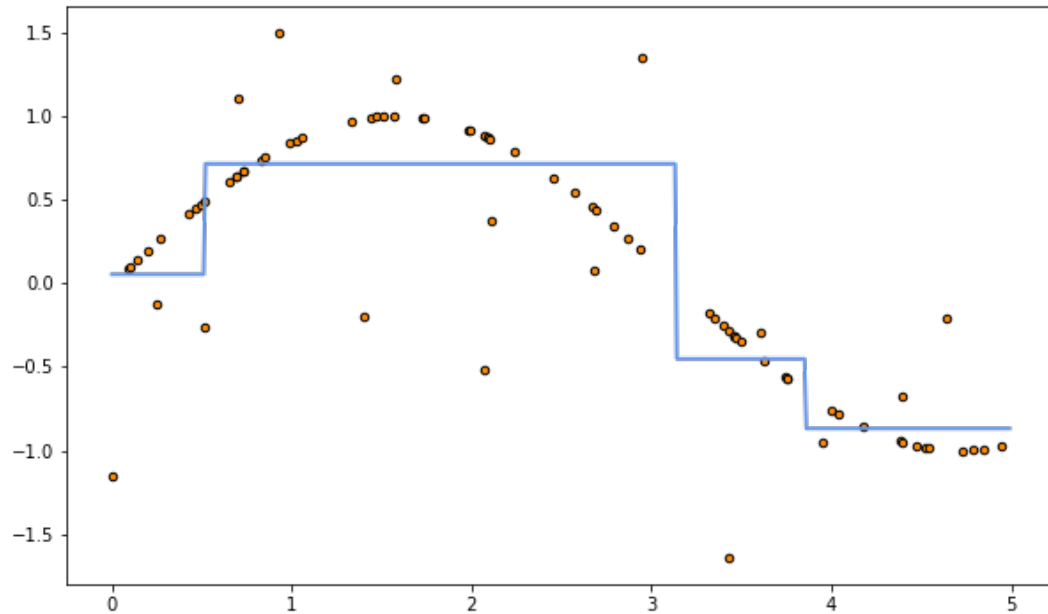
# Решающее дерево



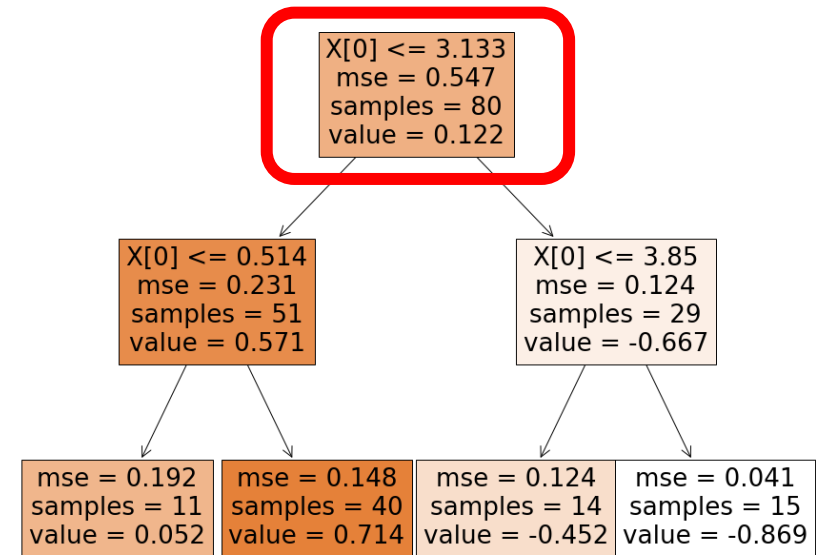
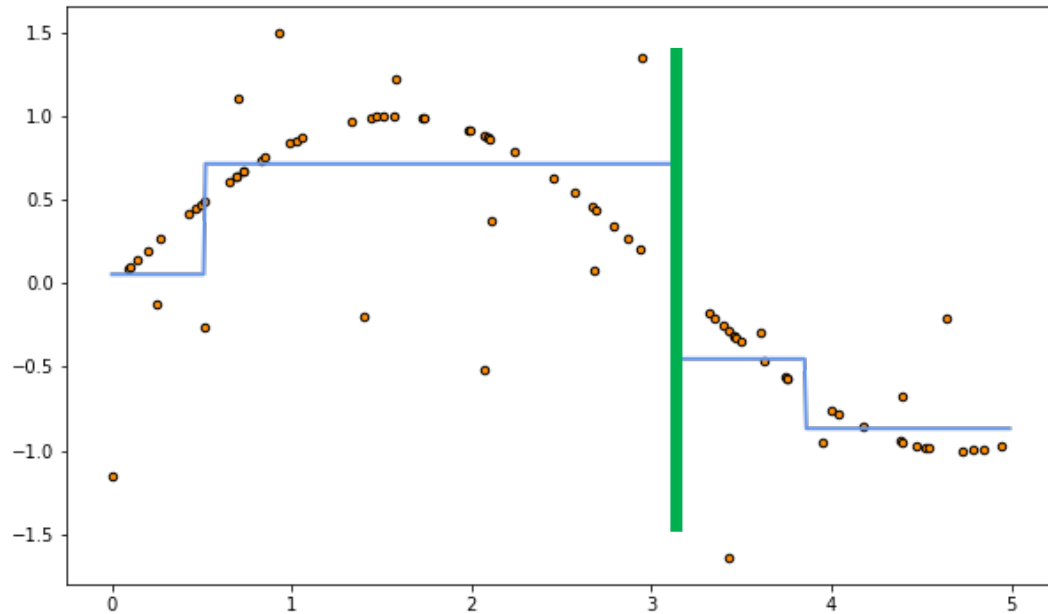
# Сложность дерева

- Решающее дерево можно строить до тех пор, пока каждый лист не будет соответствовать ровно одному объекту
- Деревом можно идеально разделить любую выборку!
- Если только нет объектов с одинаковыми признаками, но разными ответами

# Решающее дерево для регрессии

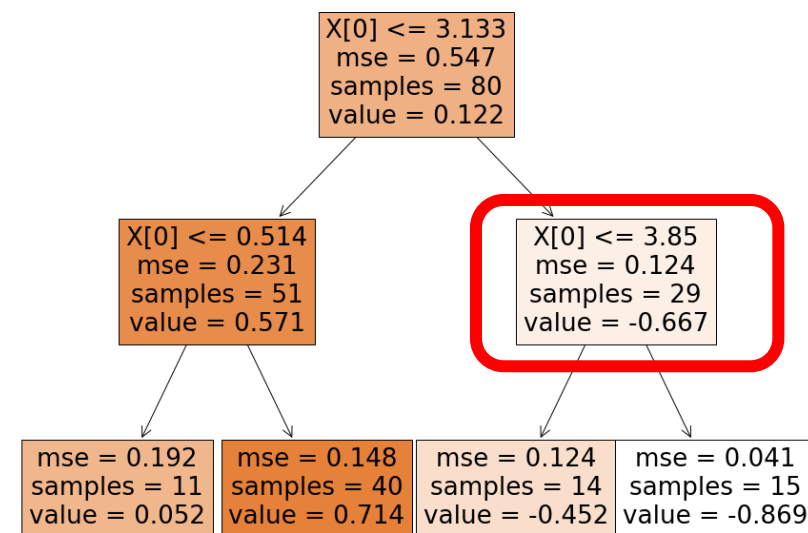
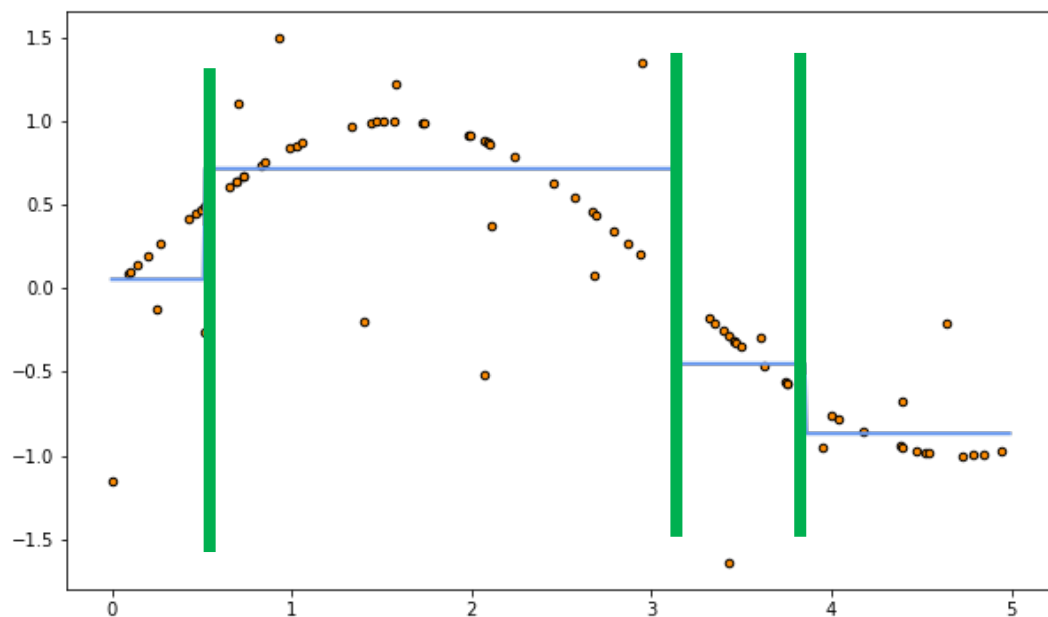


# Решающее дерево для регрессии

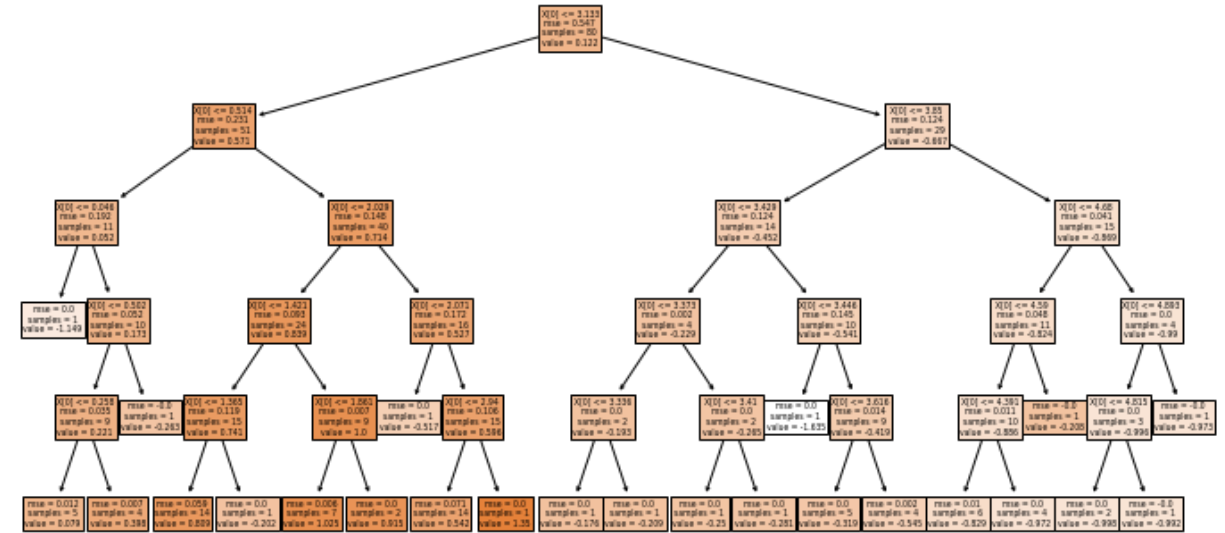
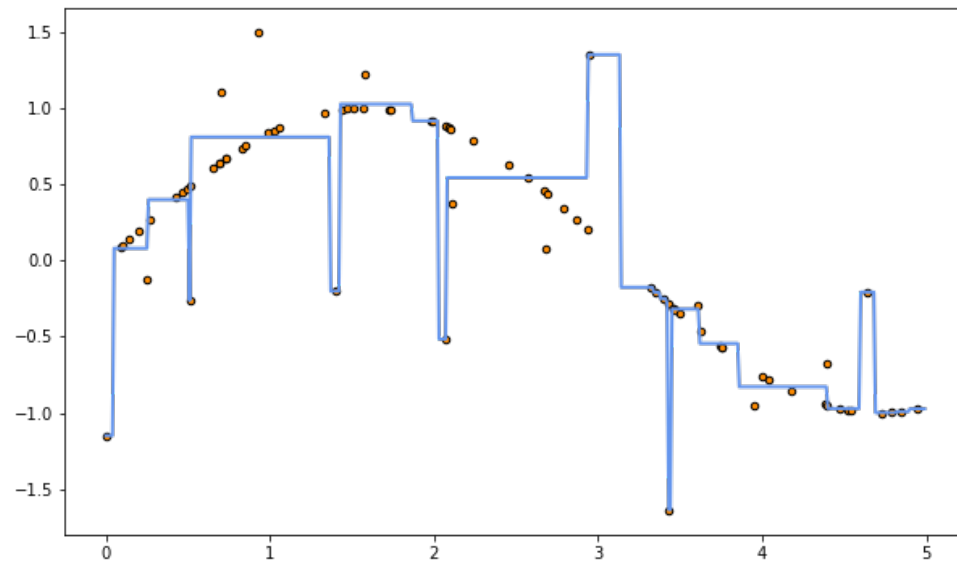




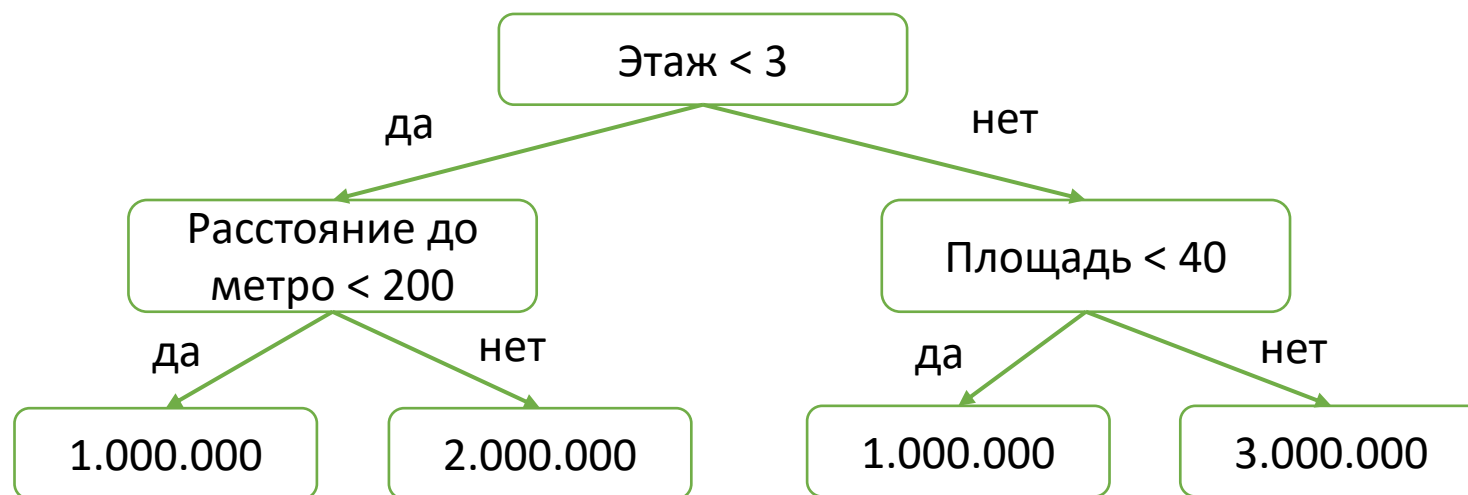
# Решающее дерево для регрессии



# Решающее дерево для регрессии



# Решающее дерево



- Внутренние вершины: предикаты  $[x_j < t]$
- Листья: прогнозы  $s \in \mathbb{Y}$

# Предикаты

- Порог на признак  $[x_j < t]$  — не единственный вариант
- Предикат с линейной моделью:  $[\langle w, x \rangle < t]$
- Предикат с метрикой:  $[\rho(x, x_0) < t]$
- И много других вариантов
- Но даже с простейшим предикатом можно строить очень сложные модели

# Прогнозы в листьях

- Наш выбор: константные прогнозы  $c_v \in \mathbb{Y}$
- Регрессия:

$$c_v = \frac{1}{|R_v|} \sum_{(x_i, y_i) \in R_v} y_i$$

- Классификация:

$$c_v = \arg \max_{k \in \mathbb{Y}} \sum_{(x_i, y_i) \in R_v} [y_i = k]$$

# Прогнозы в листьях

- Наш выбор: константные прогнозы  $c_v \in \mathbb{Y}$
- Классификация и вероятности классов:

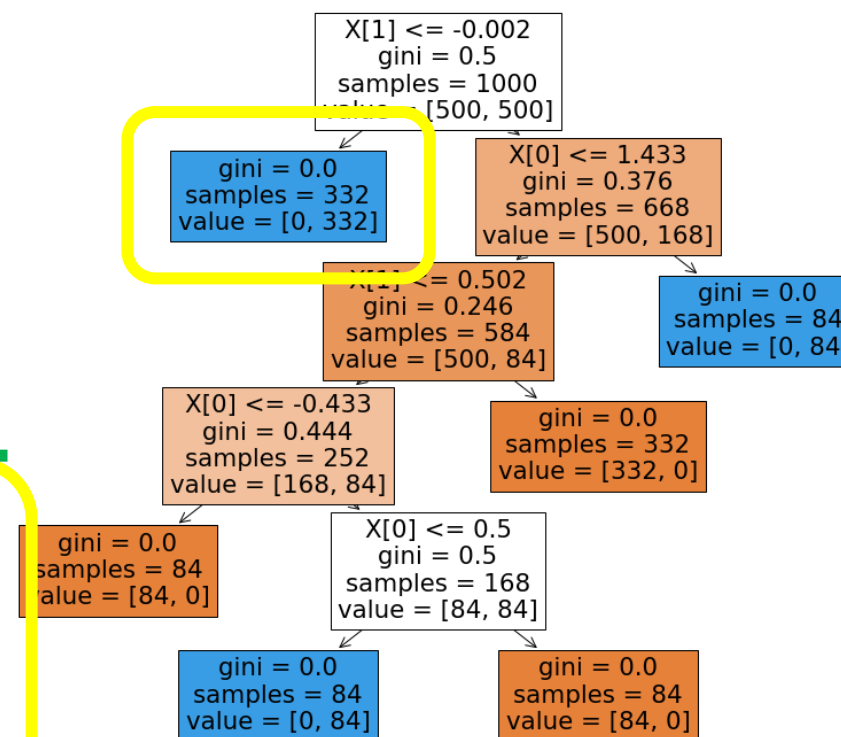
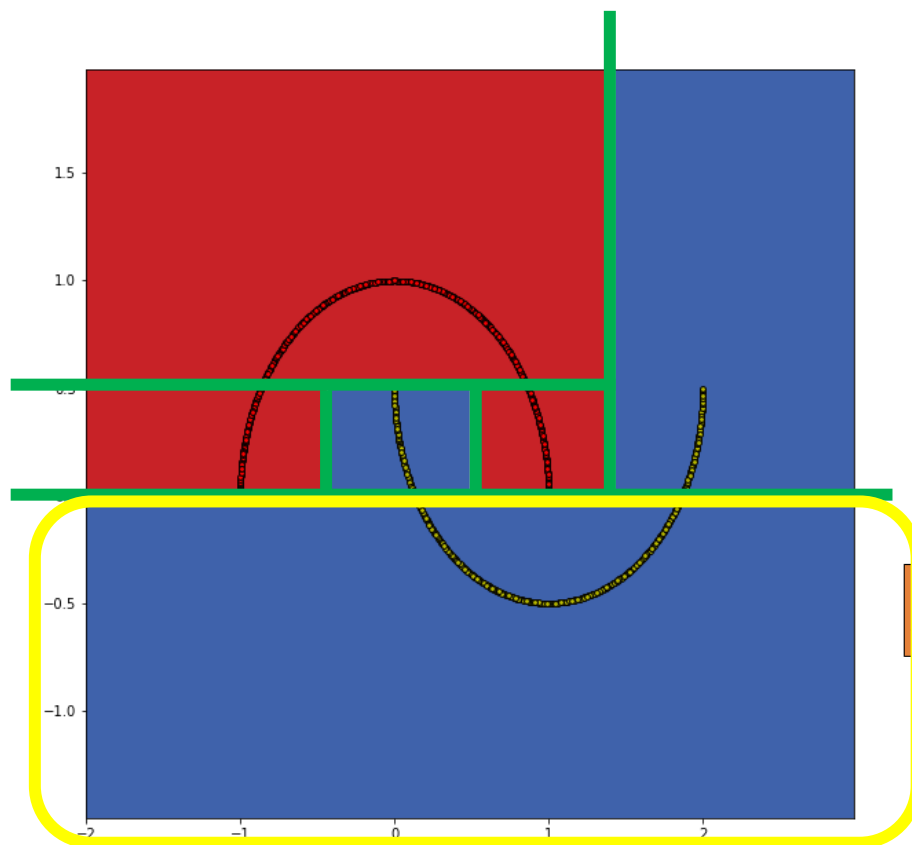
$$c_{vk} = \frac{1}{|R_v|} \sum_{(x_i, y_i) \in R_v} [y_i = k]$$

# Прогнозы в листьях

- Можно усложнять листья
- Например:

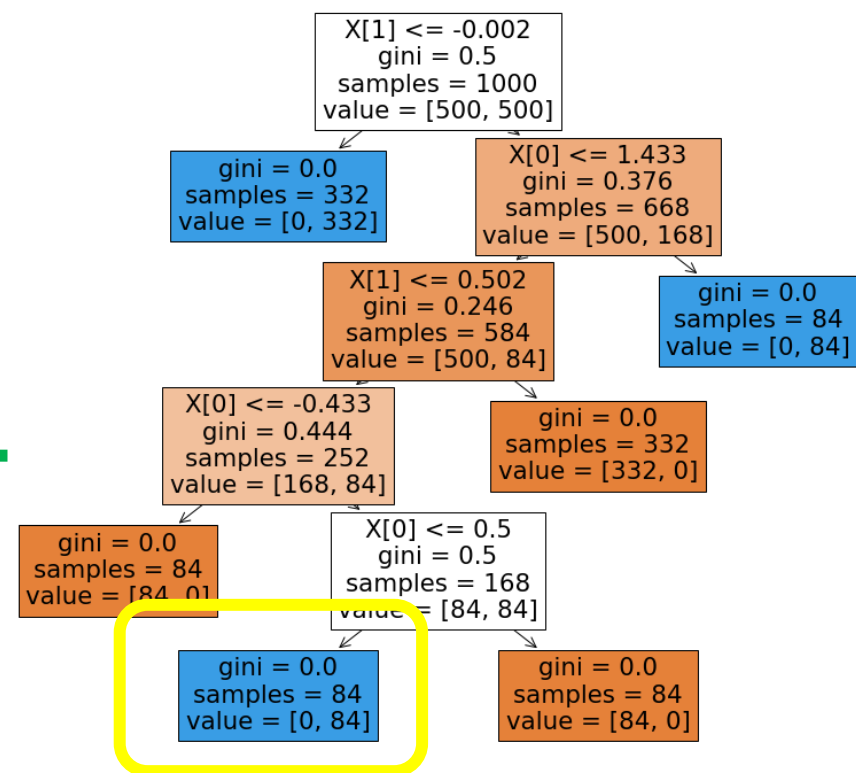
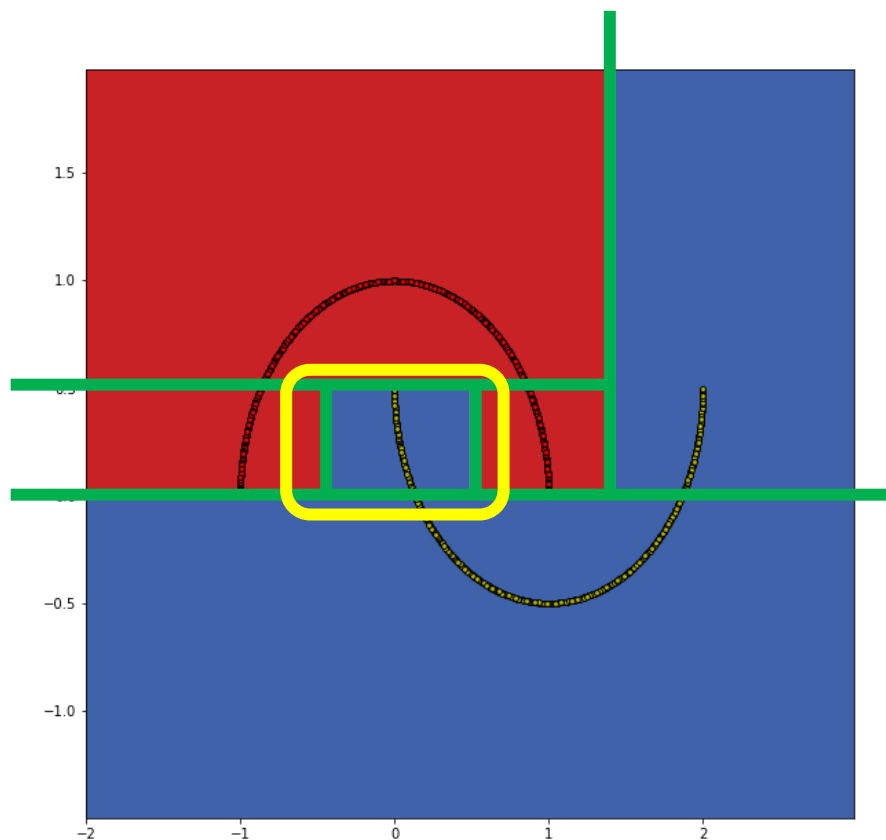
$$c_v(x) = \langle w_v, x \rangle$$

# Решающее дерево





# Решающее дерево



# Формула для дерева

- Дерево разбивает признаковое пространство на области  $R_1, \dots, R_J$
- Каждая область  $R_j$  соответствует листу
- В области  $R_j$  прогноз  $c_j$  константный

$$a(x) = \sum_{j=1}^J c_j [x \in R_j]$$

# Формула для дерева

$$a(x) = \sum_{j=1}^J c_j [x \in R_j]$$

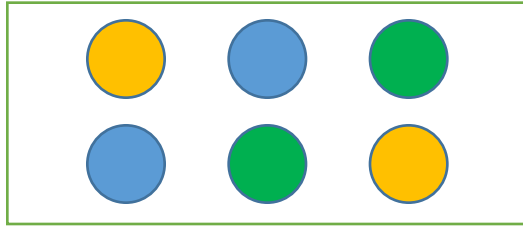
- Решающее дерево находит хорошие новые признаки
- Над этими признаками подбирает линейную модель

Как выбирать предикаты

# Жадное построение

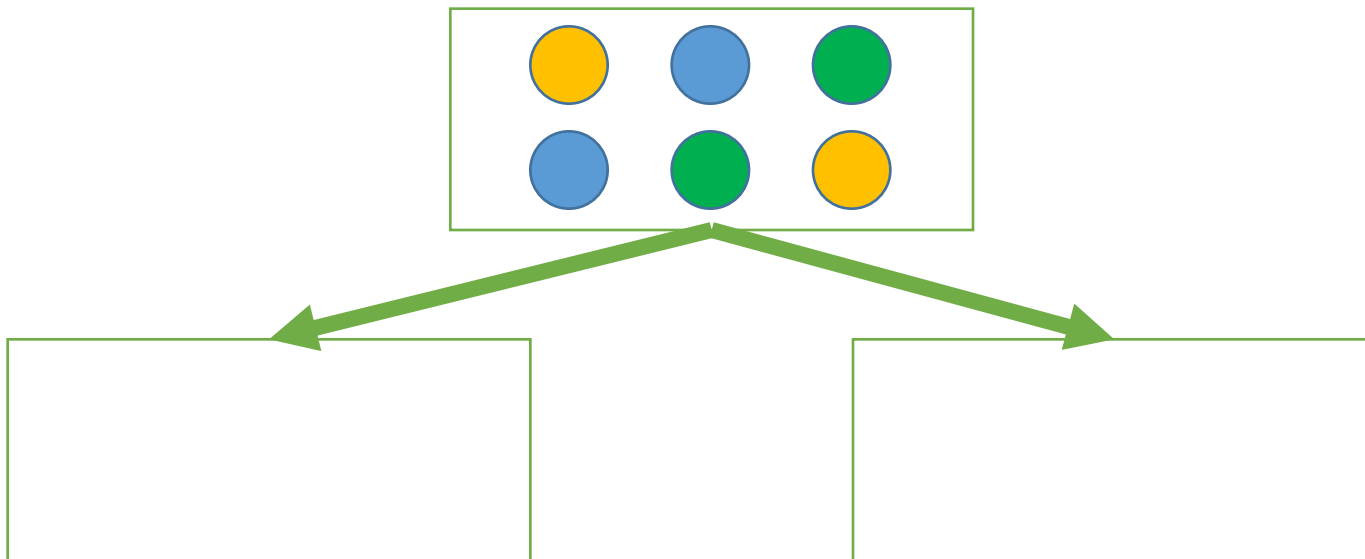
- Разберёмся на примере
- Начнём с задачи классификации

# Жадное построение

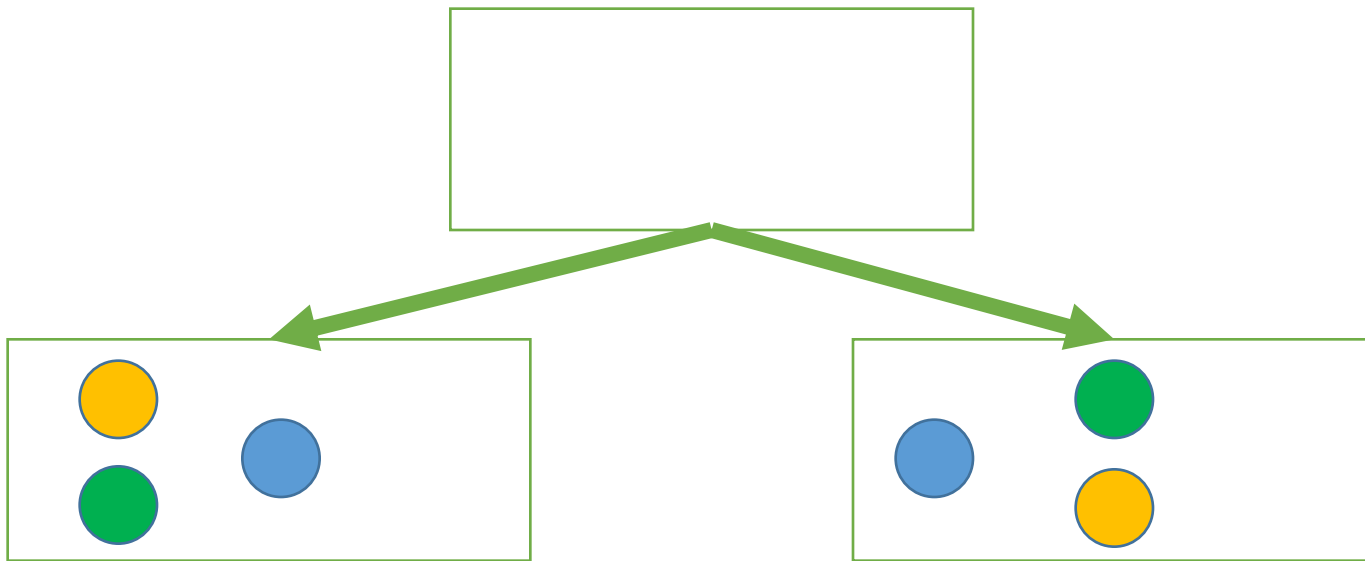


- Как разбить вершину?

# Жадное построение

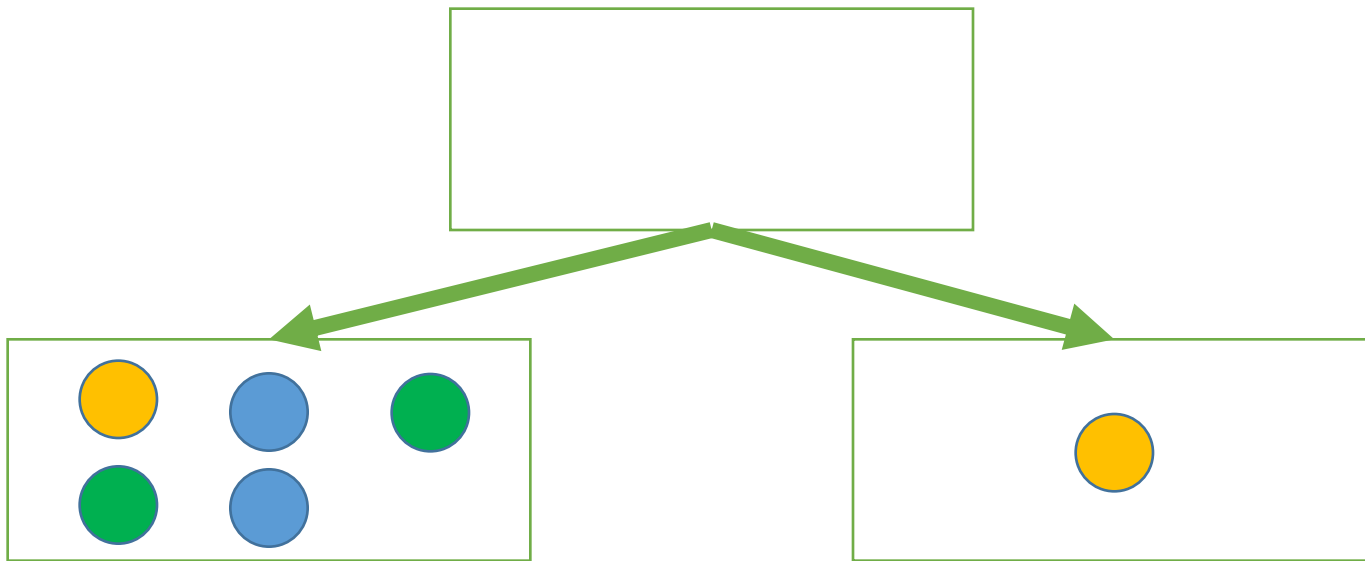


# Жадное построение

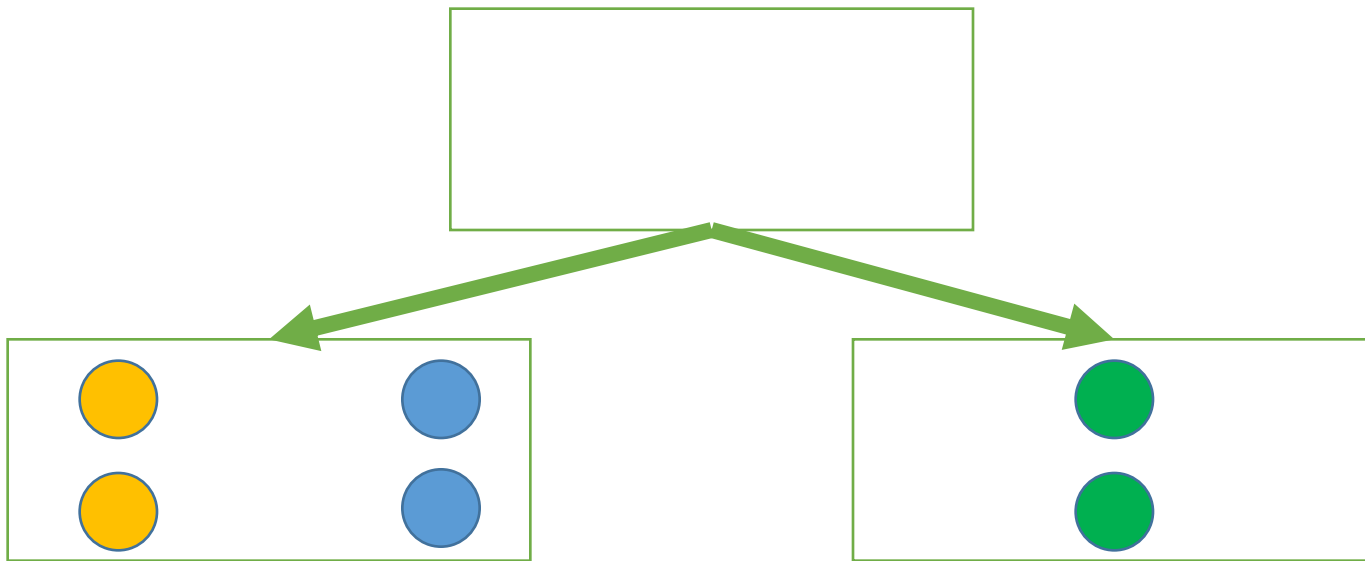




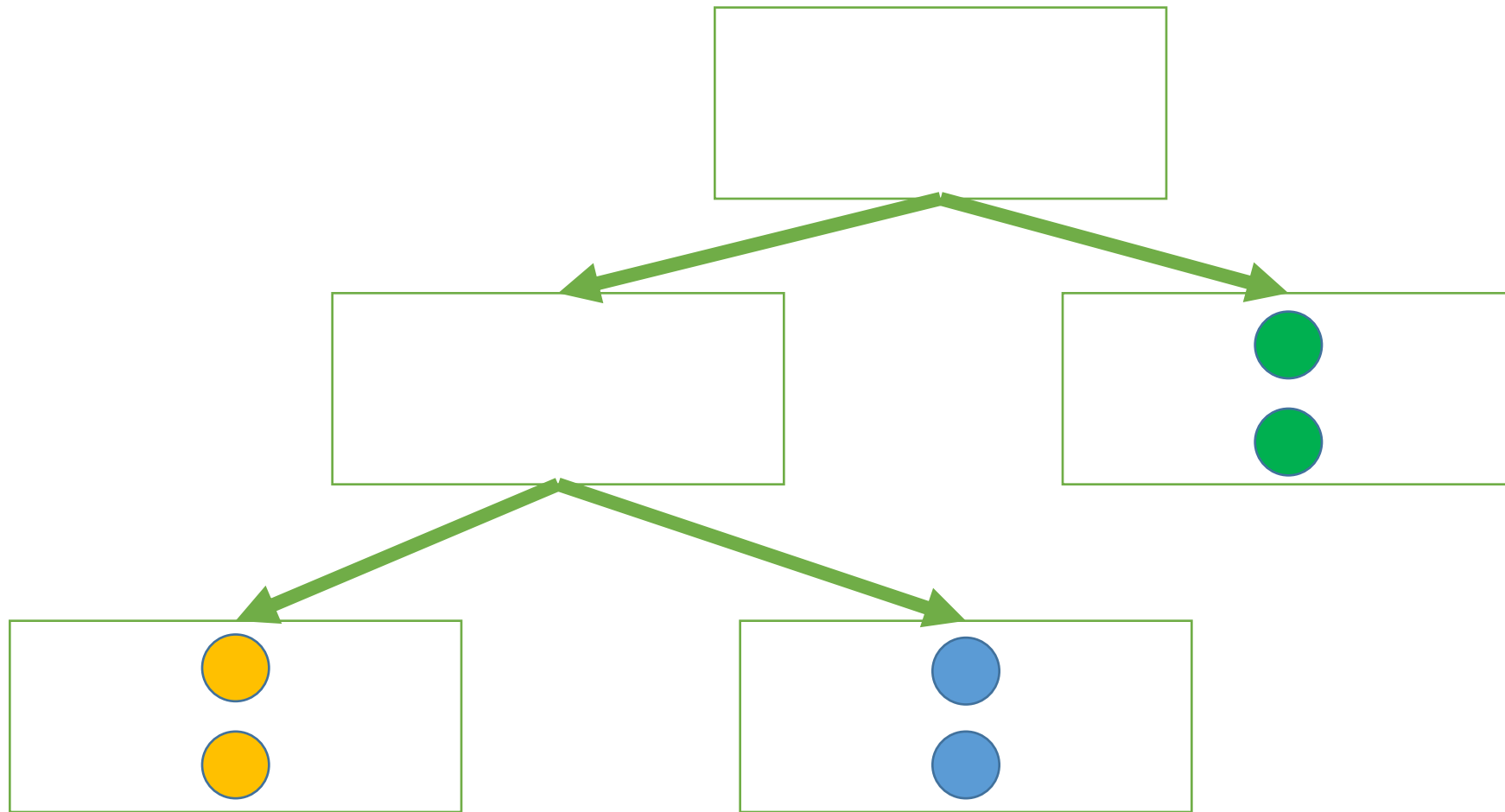
# Жадное построение



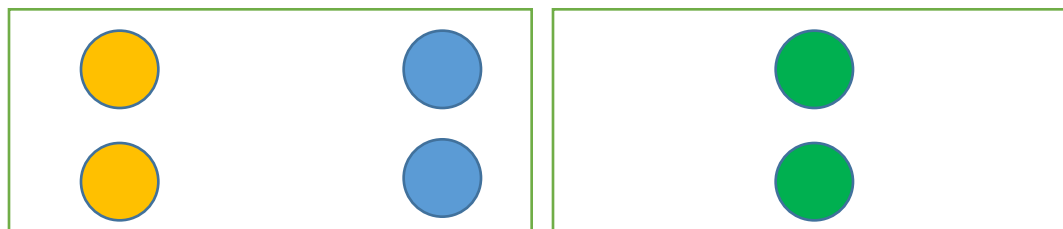
# Жадное построение



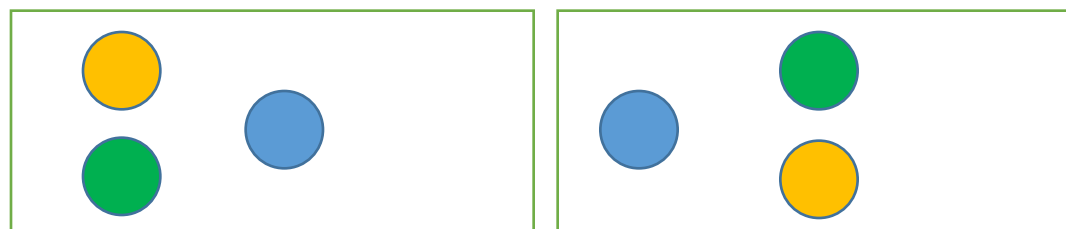
# Жадное построение



# Как сравнить разбиения?

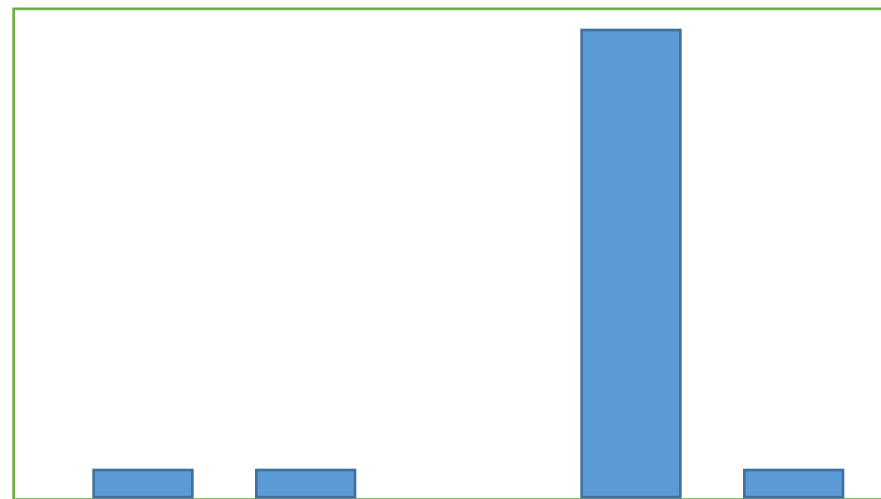
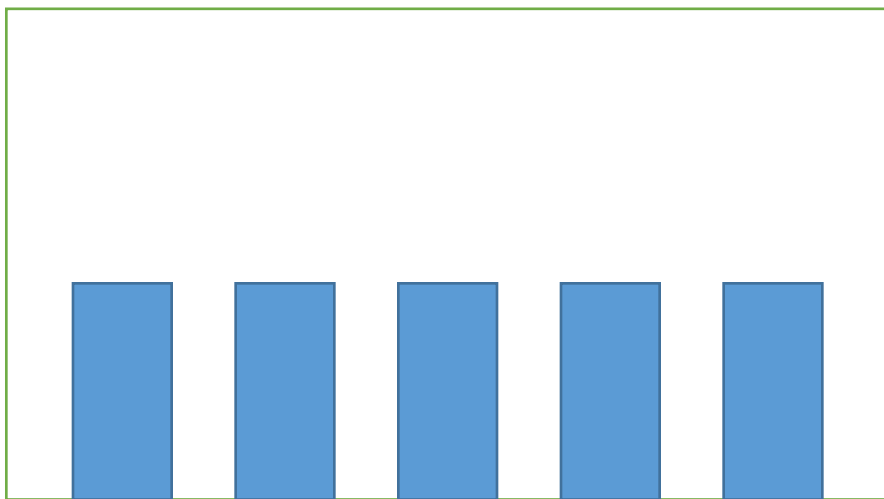


или



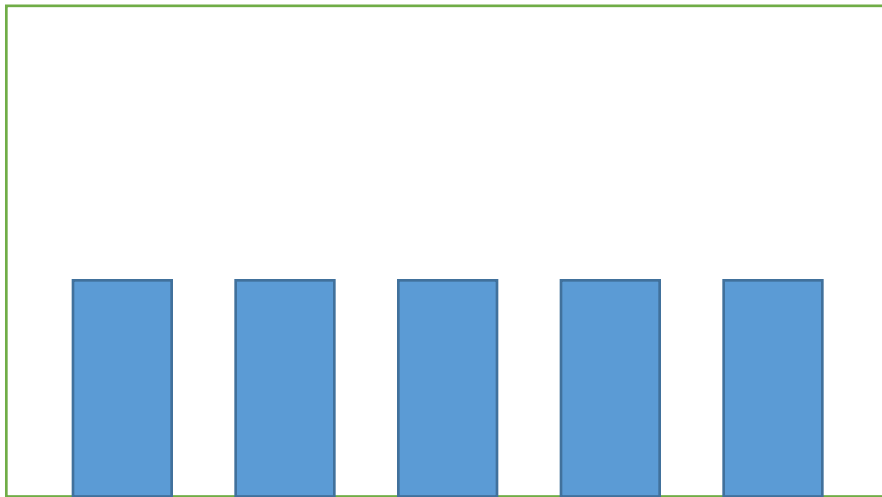
# Энтропия

- Мера неопределённости распределения

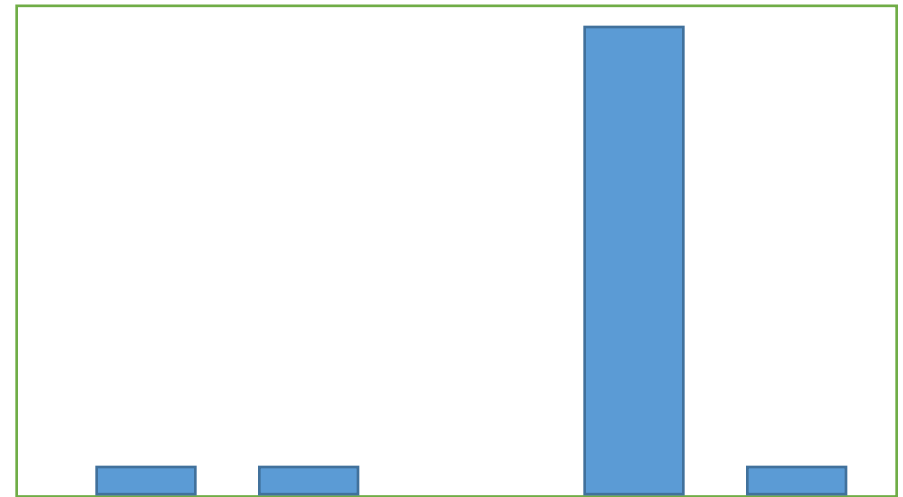


# Энтропия

- Мера неопределённости распределения



Высокая энтропия



Низкая энтропия

# Энтропия

- Дискретное распределение
- Принимает  $n$  значений с вероятностями  $p_1, \dots, p_n$
- Энтропия:

$$H(p_1, \dots, p_n) = - \sum_{i=1}^n p_i \log p_i$$

# Энтропия

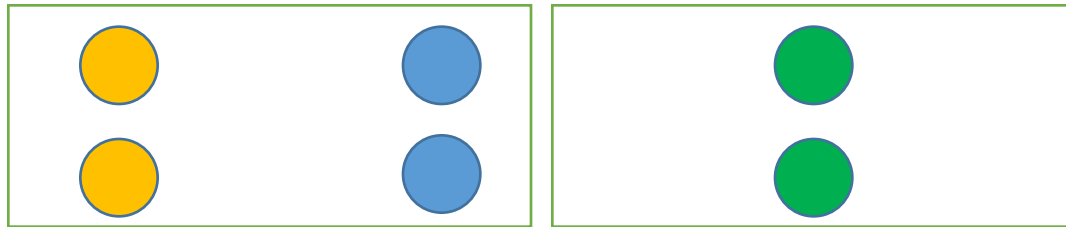
- $(0.2, 0.2, 0.2, 0.2, 0.2)$
- $H = 1.60944 \dots$

- $(0.9, 0.05, 0.05, 0, 0)$
- $H = 0.394398 \dots$

- $(0, 0, 0, 1, 0)$
- $H = 0$



# Как сравнить разбиения?



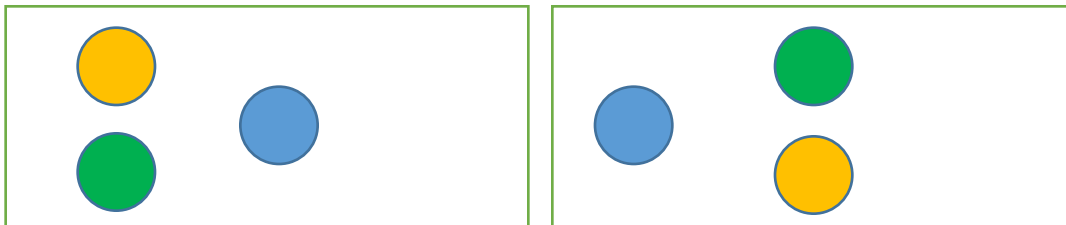
0.693

0

- $(0.5, 0.5, 0)$  и  $(0, 0, 1)$
- $H = 0.693 + 0 = 0.693$

1.09

1.09



- $(0.33, 0.33, 0.33)$  и  $(0.33, 0.33, 0.33)$
- $H = 1.09 + 1.09 = 2.18$

# Энтропия

$$H(p_1, \dots, p_K) = - \sum_{i=1}^K p_i \log_2 p_i$$

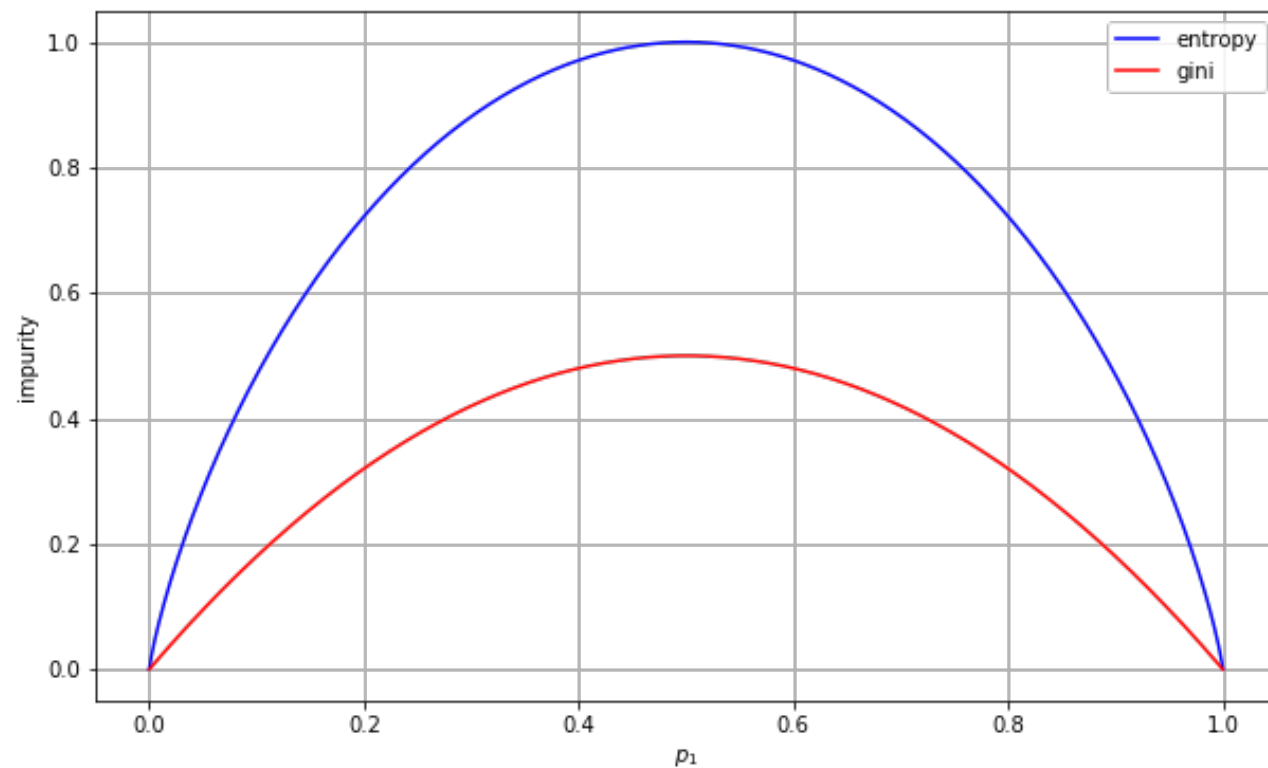
- Характеристика «хаотичности» вершины
- **Impurity**

# Критерий Джини

$$H(p_1, \dots, p_K) = \sum_{i=1}^K p_i (1 - p_i)$$

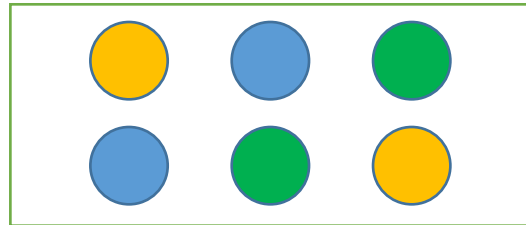
- Вероятность ошибки случайного классификатора, который выдаёт класс  $k$  с вероятностью  $p_k$
- Примерно пропорционально количеству пар объектов, относящихся к разным классам

# Критерии качества вершины

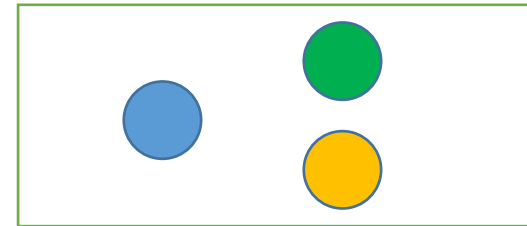
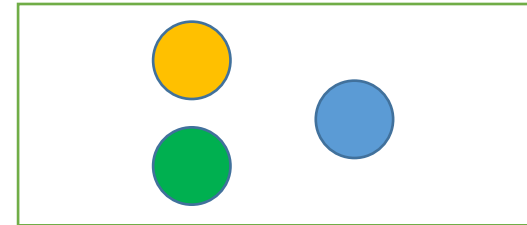


# Критерий информативности

- Как понять, какой предикат лучше?
- Сравнить хаотичность в исходной вершине и в двух дочерних!

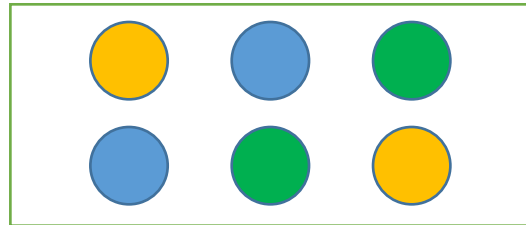


против

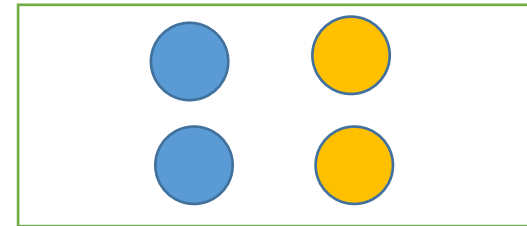
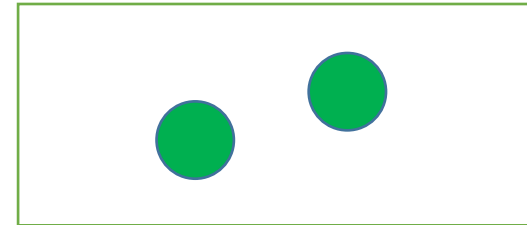


# Критерий информативности

- Как понять, какой предикат лучше?
- Сравнить хаотичность в исходной вершине и в двух дочерних!



против



# Критерий информативности

- Как понять, какой предикат лучше?
- Сравнить хаотичность в исходной вершине и в двух дочерних!

$$Q(R, j, t) = H(R) - H(R_\ell) - H(R_r) \rightarrow \max_{j, t}$$

# Критерий информативности

- Как понять, какой предикат лучше?
- Сравнить хаотичность в исходной вершине и в двух дочерних!

$$Q(R, j, t) = H(R) - H(R_\ell) - H(R_r) \rightarrow \max_{j,t}$$

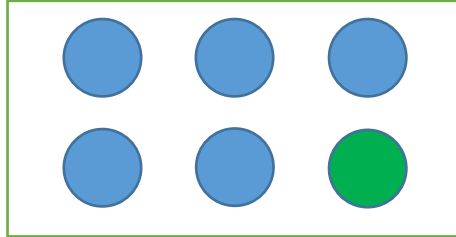
- Или так:

$$Q(R, j, t) = H(R_\ell) + H(R_r) \rightarrow \min_{j,t}$$

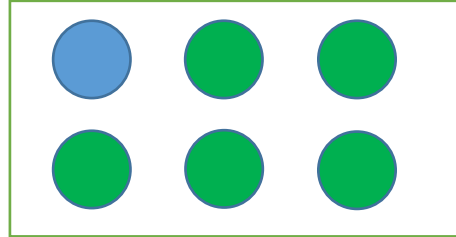
- (у этих формул есть проблемы!)



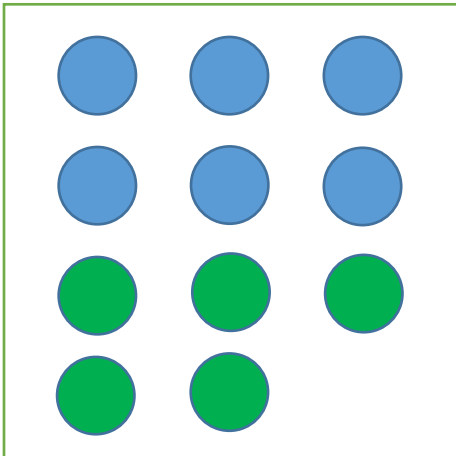
# Как сравнить разбиения?



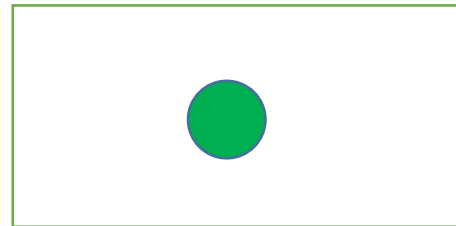
0.65



0.65



0.994



0

- $(5/6, 1/6)$  и  $(1/6, 5/6)$
- $0.65 + 0.65 = 1.3$

- $(6/11, 5/11)$  и  $(0, 1)$
- $0.994 + 0 = 0.994$

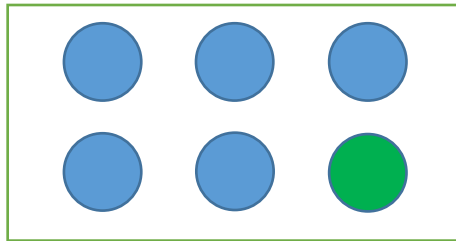
# Критерий информативности

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

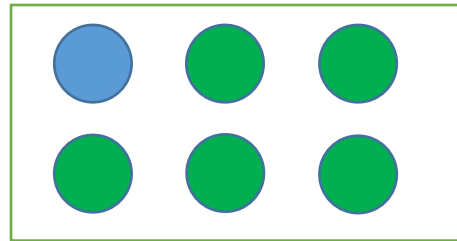
- Или так:

$$Q(R, j, t) = \frac{|R_\ell|}{|R|} H(R_\ell) + \frac{|R_r|}{|R|} H(R_r) \rightarrow \min_{j,t}$$

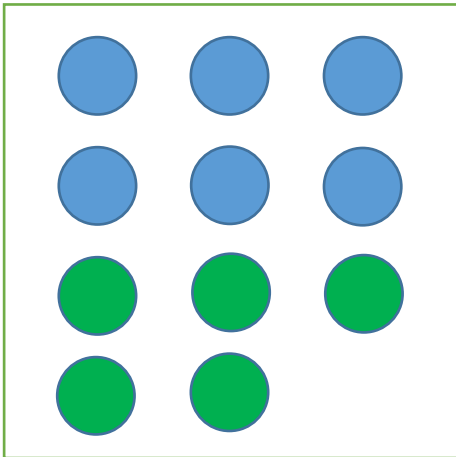
# Как сравнить разбиения?



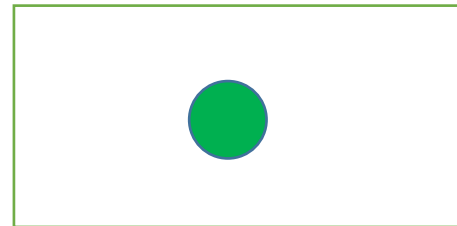
0.65



0.65



0.994



0

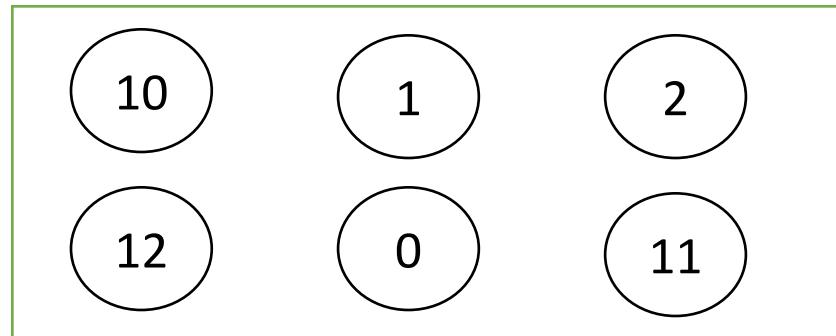
- $(5/6, 1/6)$  и  $(1/6, 5/6)$

- $0.5 * 0.65 + 0.5 * 0.65 = 0.65$

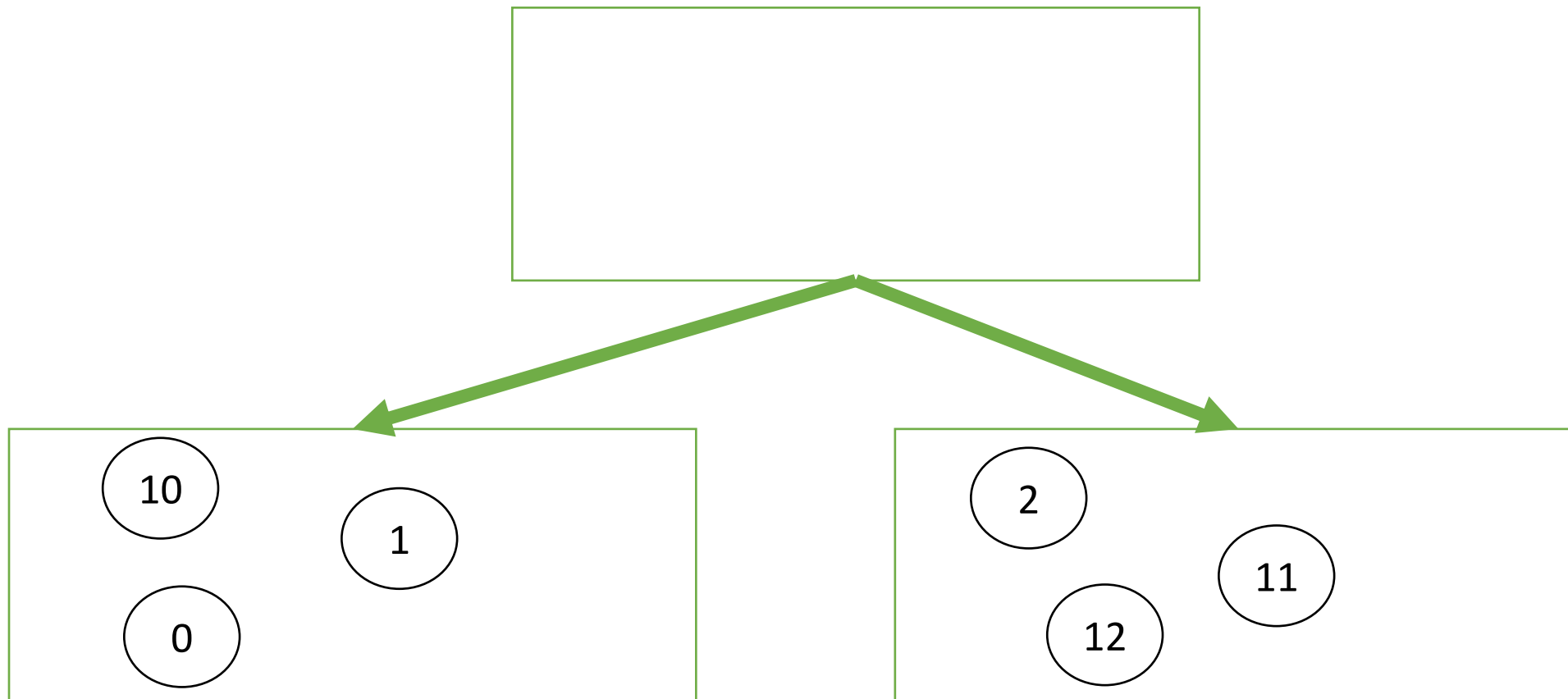
- $(6/11, 5/11)$  и  $(0, 1)$

- $\frac{11}{12} * 0.994 + \frac{1}{12} * 0 = 0.911$

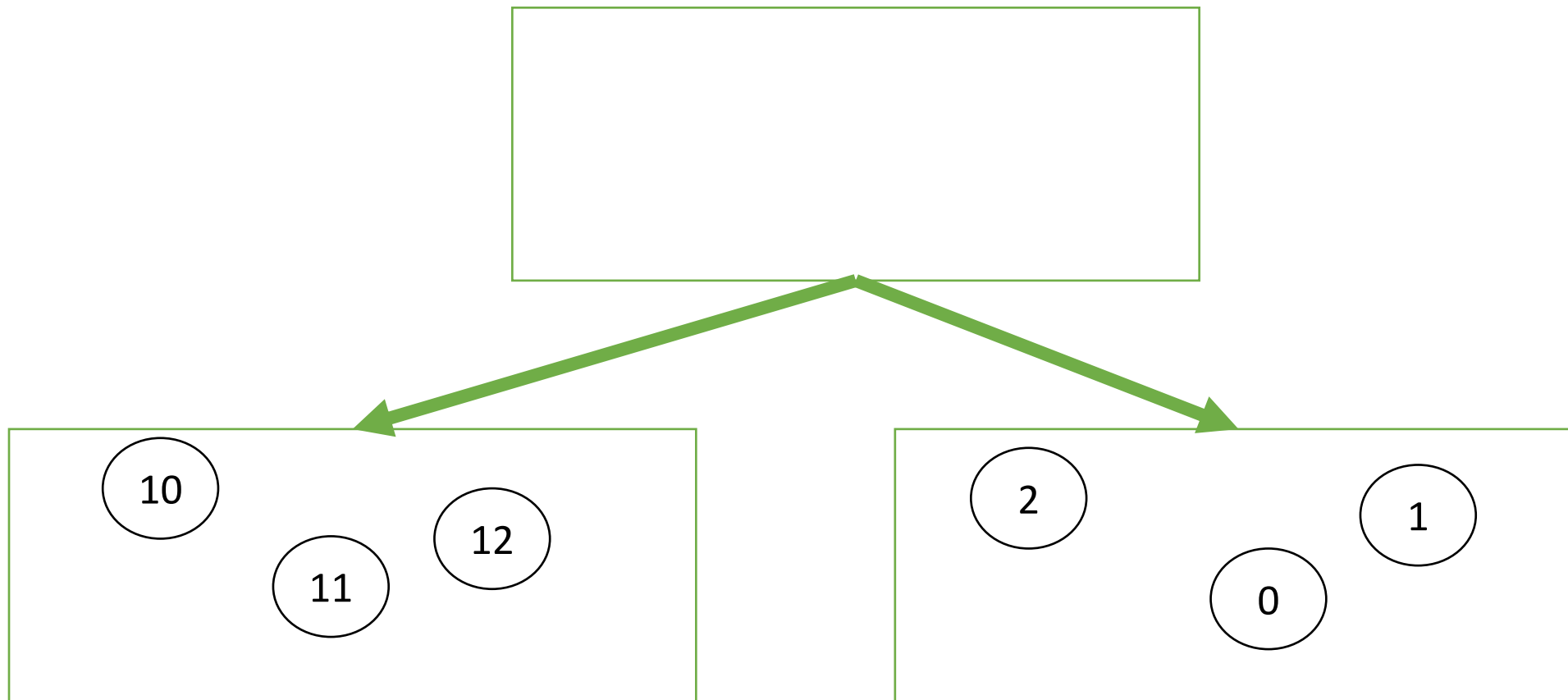
А для регрессии?



А для регрессии?



А для регрессии?



# Задача регрессии

$$H(R) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - y_R)^2$$

$$y_R = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} y_i$$

- То есть «хаотичность» вершины можно измерять дисперсией ответов в ней

Жадное построение дерева



# Как строить дерево?

- Оптимальный вариант: перебрать все возможные деревья, выбрать самое маленькое среди безошибочных
- Слишком долго

# Как строить дерево?

- Мы уже умеем выбрать лучший предикат для разбиения вершины
- Будем строить жадно
- Начнём с корня дерева, будем разбивать последовательно, пока не выполнится некоторый критерий останова

# Критерий останова

- Ограничить глубину
- Ограничить количество листьев
- Задать минимальное число объектов в вершине
- Задать минимальное уменьшение хаотичности при разбиении
- И так далее

# Жадный алгоритм

1. Поместить в корень всю выборку:  $R_1 = X$
2. Запустить построение из корня:  $\text{SplitNode}(1, R_1)$

# Жадный алгоритм

- $\text{SplitNode}(m, R_m)$

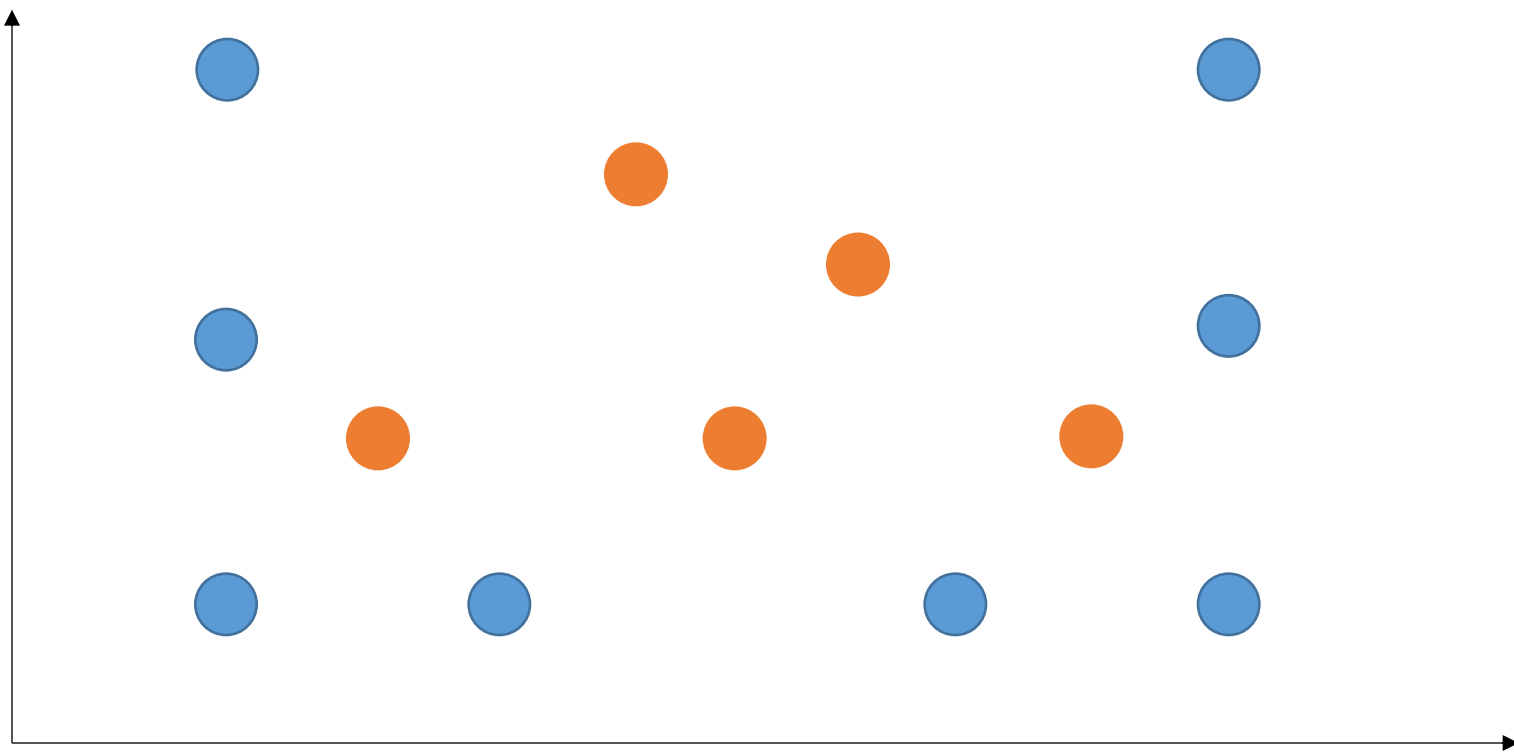
1. Если выполнен критерий останова, то выход

2. Ищем лучший предикат:  $j, t = \arg \min_{j, t} Q(R_m, j, t)$

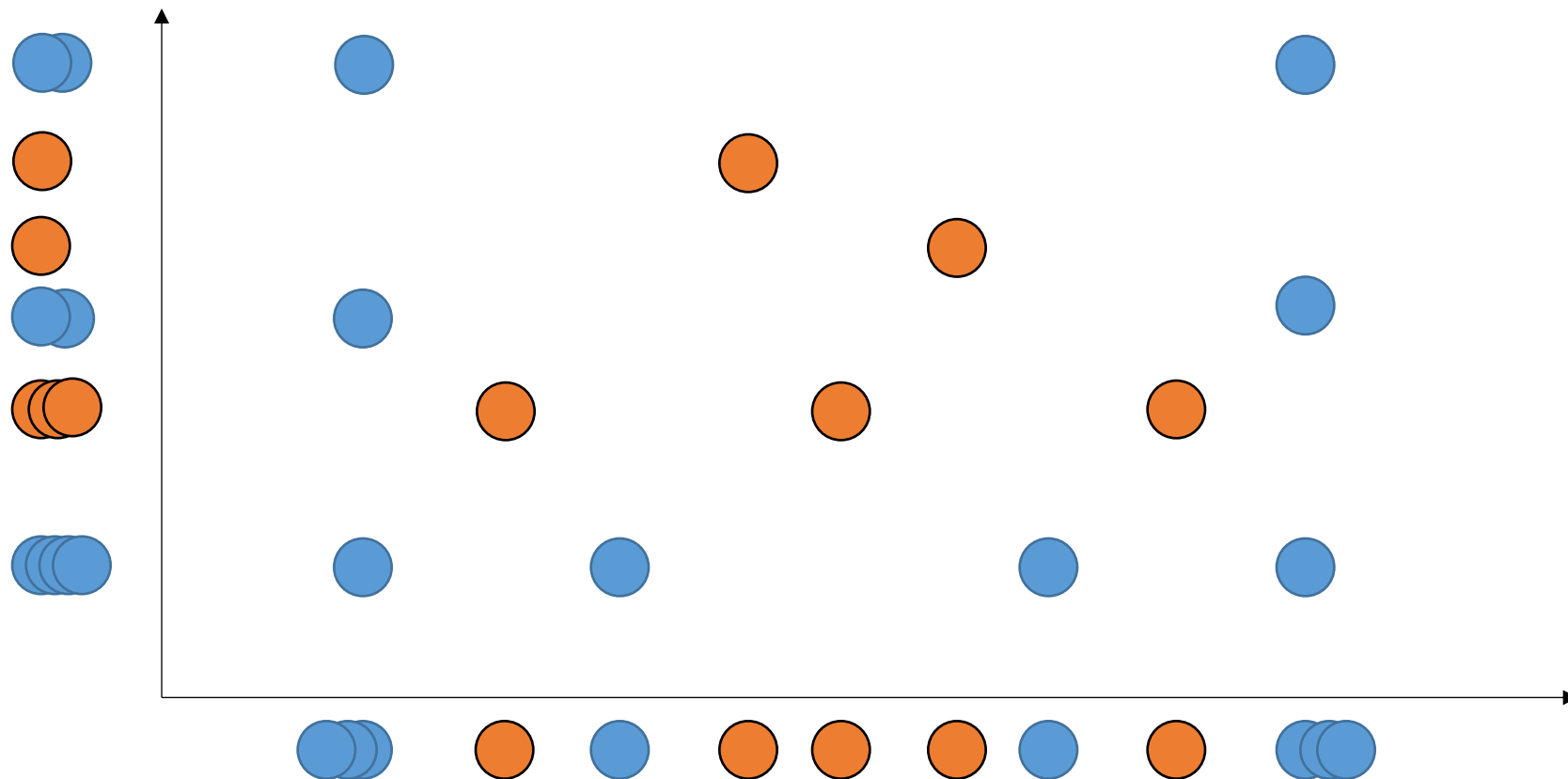
3. Разбиваем с его помощью объекты:  $R_\ell = \{(x, y) \in R_m \mid [x_j < t]\}$ ,  
 $R_r = \{(x, y) \in R_m \mid [x_j \geq t]\}$

4. Повторяем для дочерних вершин:  $\text{SplitNode}(\ell, R_\ell)$  и  $\text{SplitNode}(r, R_r)$

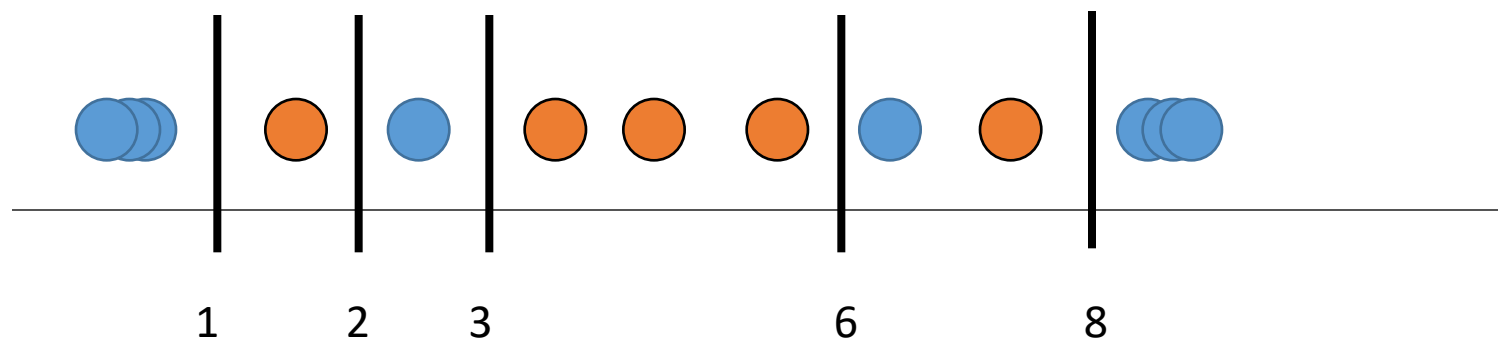
# Обучение деревьев



# Признаки

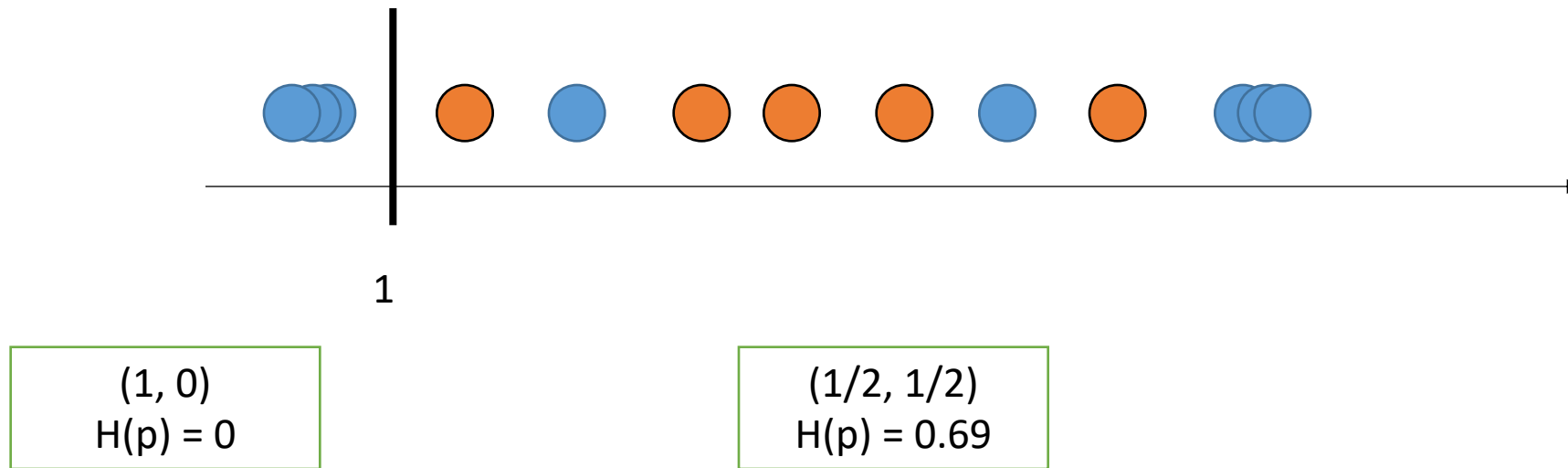


# Разбиения по признаку 1



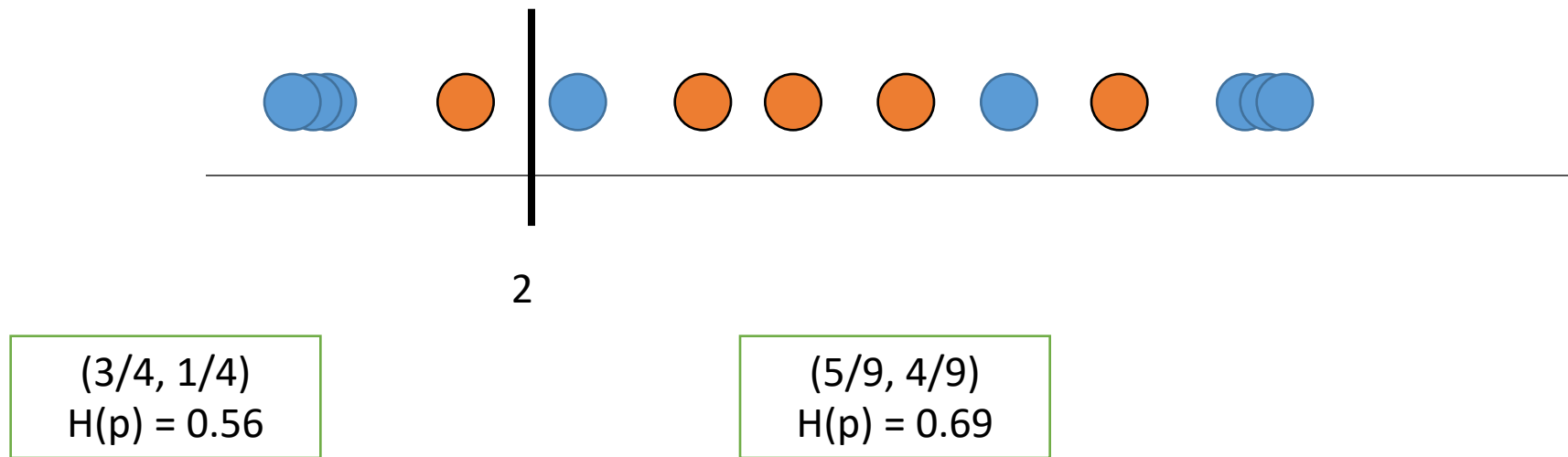


# Разбиения по признаку 1



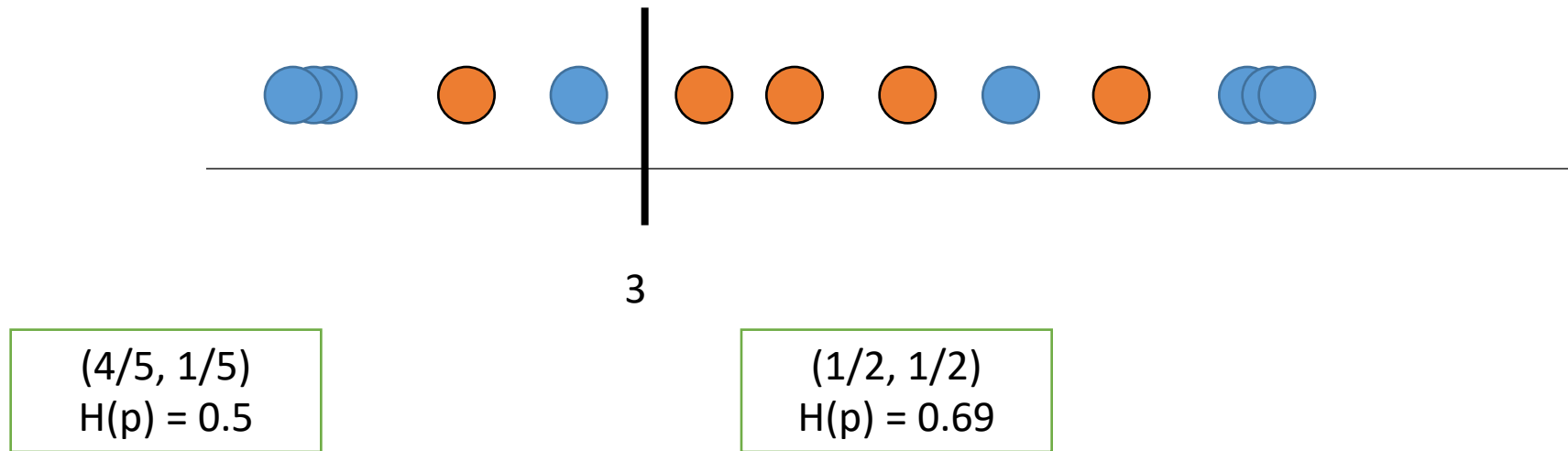
$$\frac{3}{13}H(p_l) + \frac{10}{13}H(p_r) = 0.53$$

# Разбиения по признаку 1



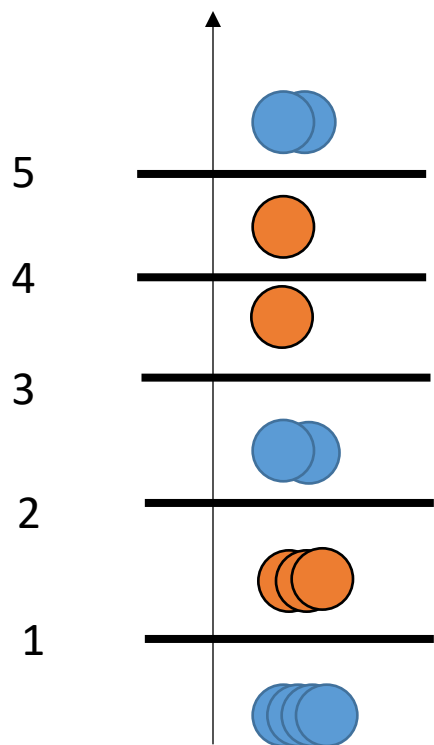
$$\frac{4}{13}H(p_l) + \frac{9}{13}H(p_r) = 0.65$$

# Разбиения по признаку 1

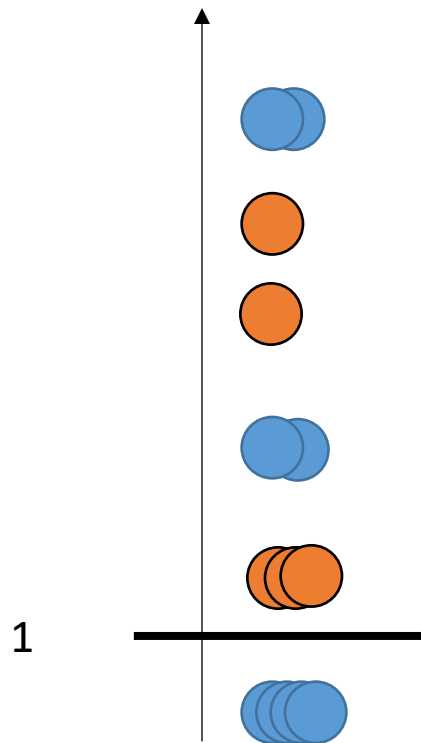


$$\frac{5}{13}H(p_l) + \frac{8}{13}H(p_r) = 0.62$$

# Разбиения по признаку 2



# Разбиения по признаку 2

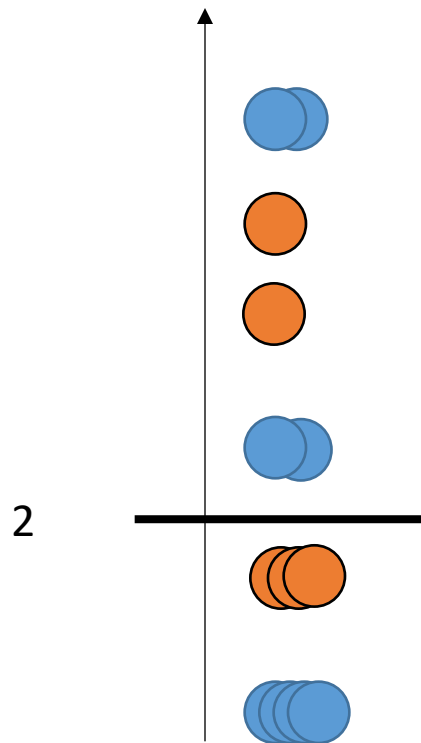


$(4/9, 5/9)$   
 $H(p) = 0.69$

$(1, 0)$   
 $H(p) = 0$

$$\frac{4}{13}H(p_l) + \frac{9}{13}H(p_r) = 0.47$$

# Разбиения по признаку 2

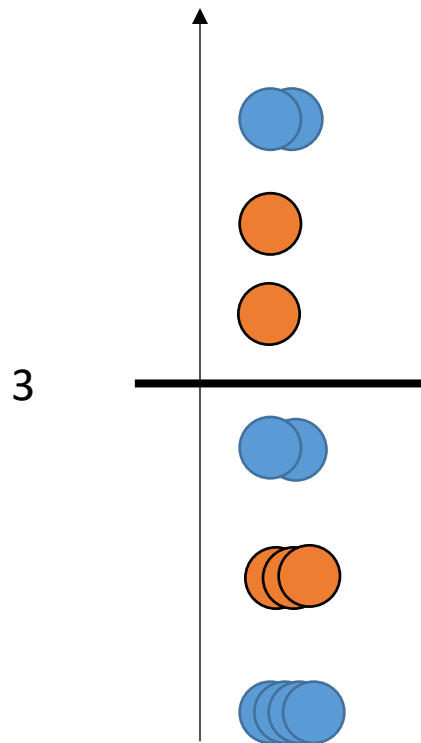


$(4/6, 2/6)$   
 $H(p) = 0.64$

$(4/7, 3/7)$   
 $H(p) = 0.68$

$$\frac{7}{13}H(p_l) + \frac{6}{13}H(p_r) = 0.66$$

# Разбиения по признаку 2

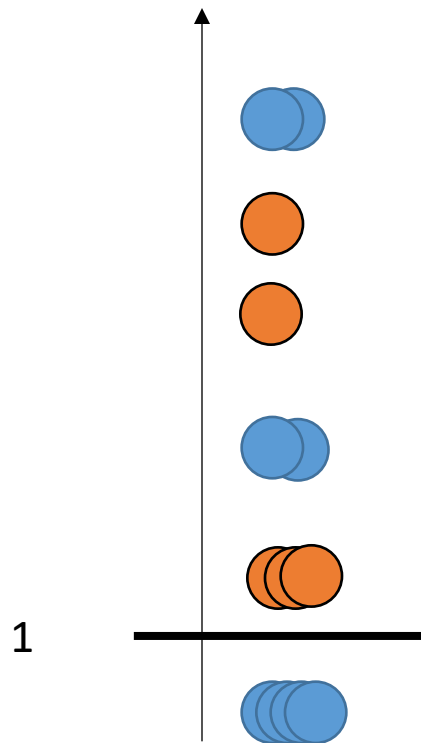


$(1/2, 1/2)$   
 $H(p) = 0.69$

$(6/9, 3/9)$   
 $H(p) = 0.46$

$$\frac{9}{13}H(p_l) + \frac{4}{13}H(p_r) = 0.53$$

# Разбиения по признаку 2



$(4/9, 5/9)$   
 $H(p) = 0.69$

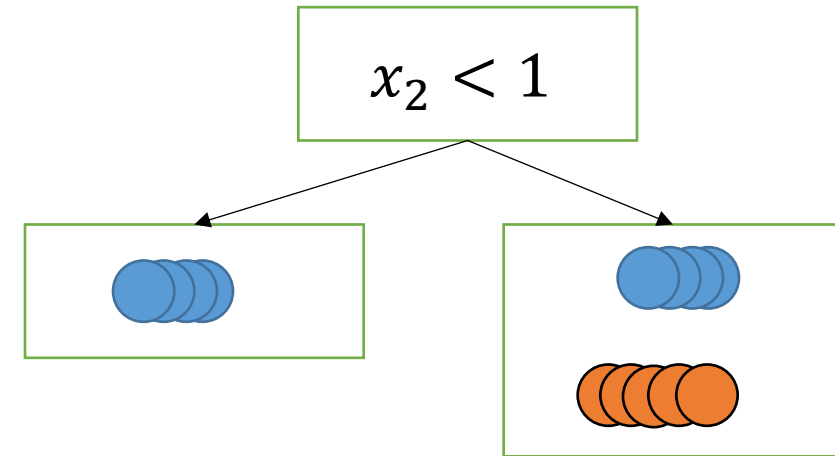
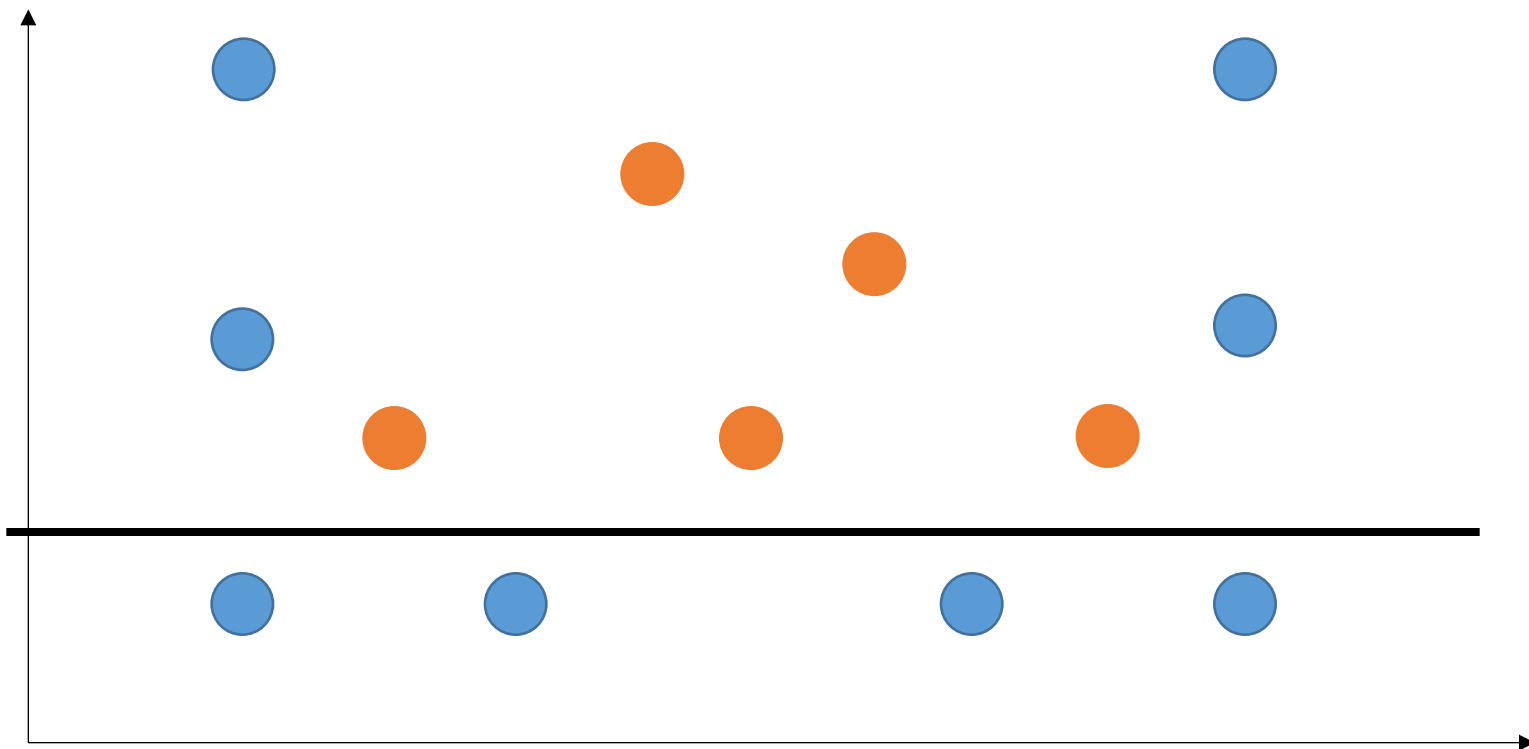
$(1, 0)$   
 $H(p) = 0$

$$\frac{4}{13}H(p_l) + \frac{9}{13}H(p_r) = 0.47$$

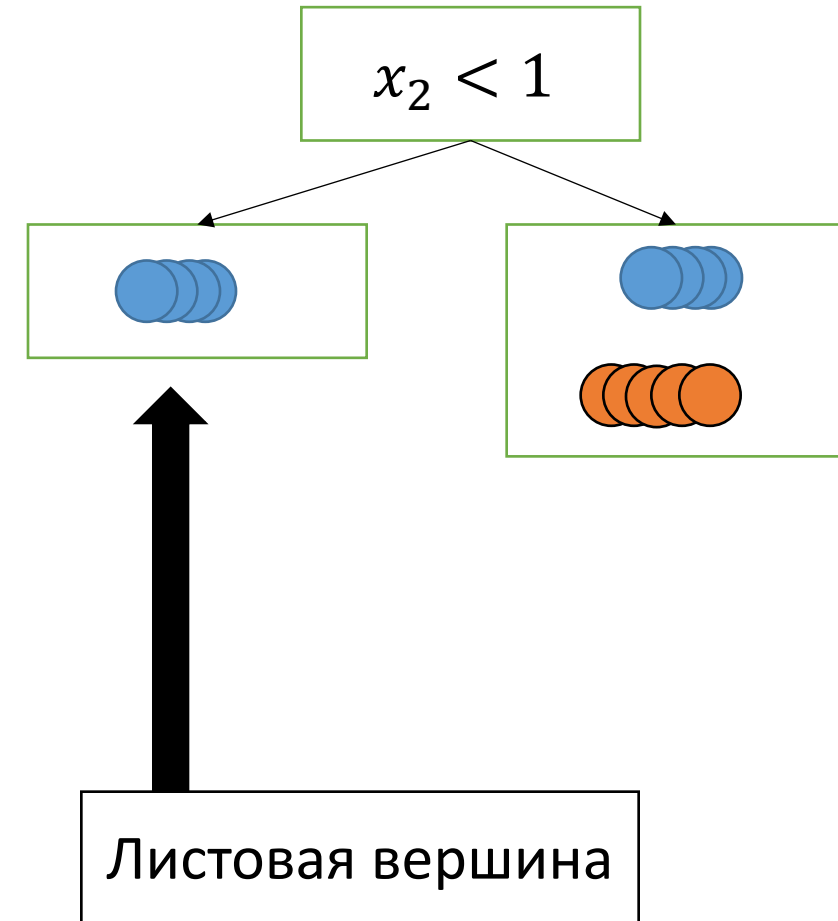
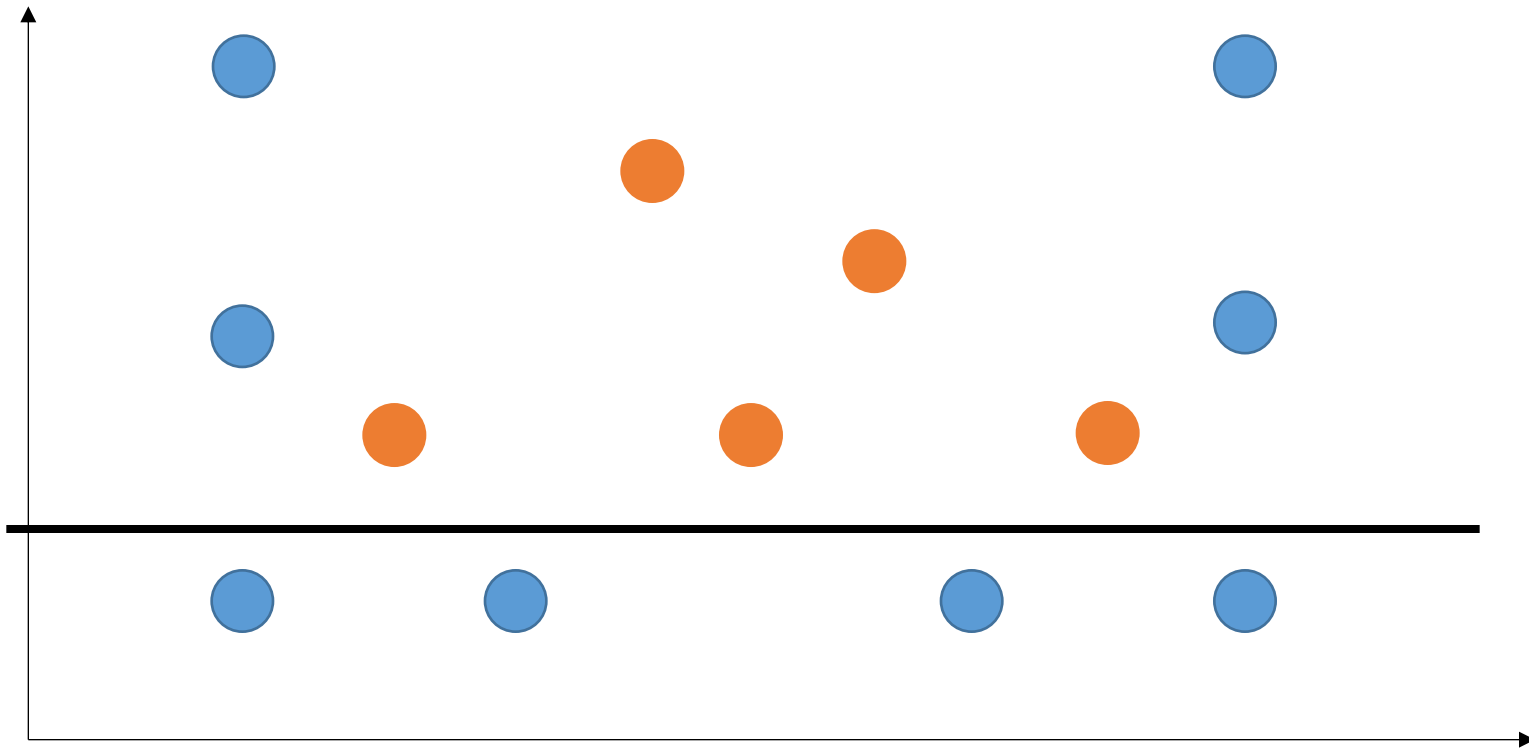
Лучшее разбиение!



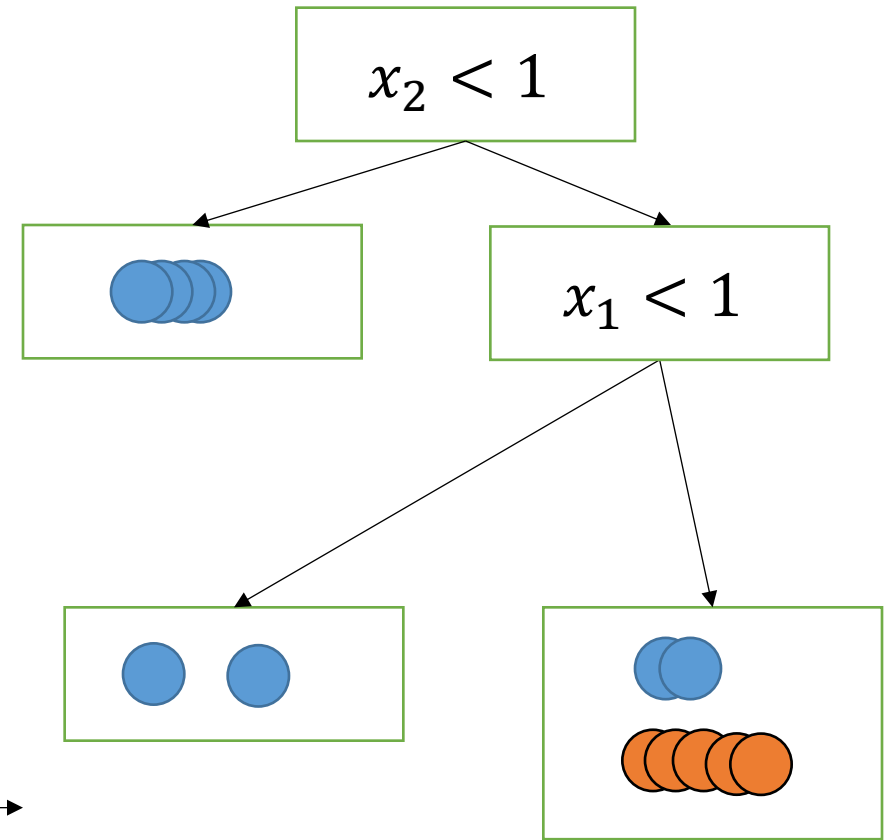
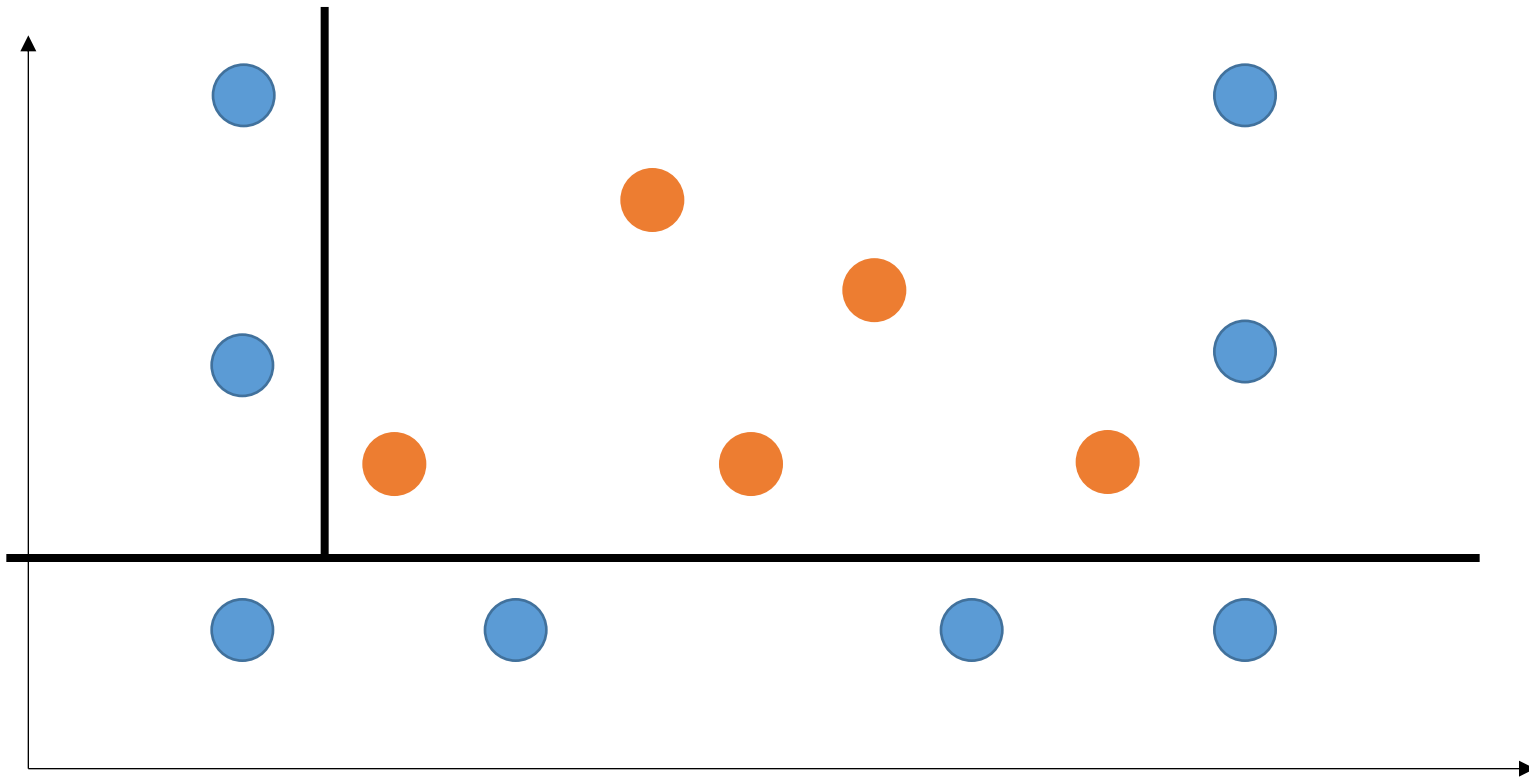
# Обучение деревьев



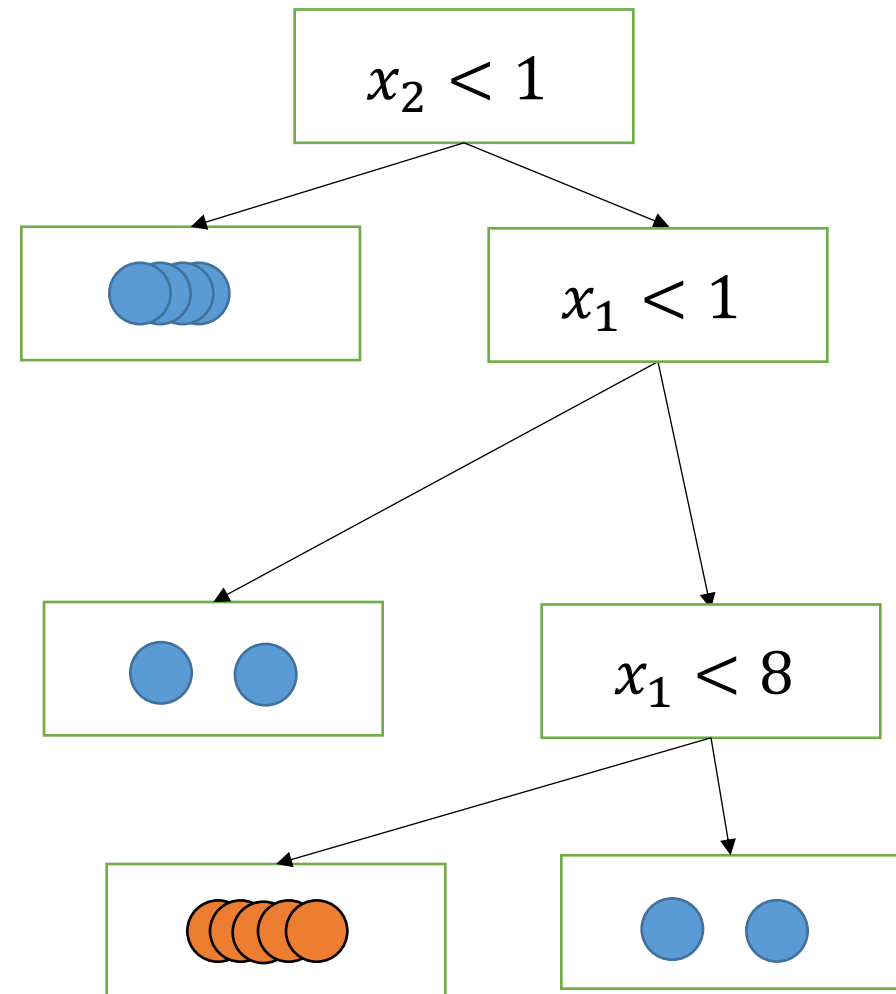
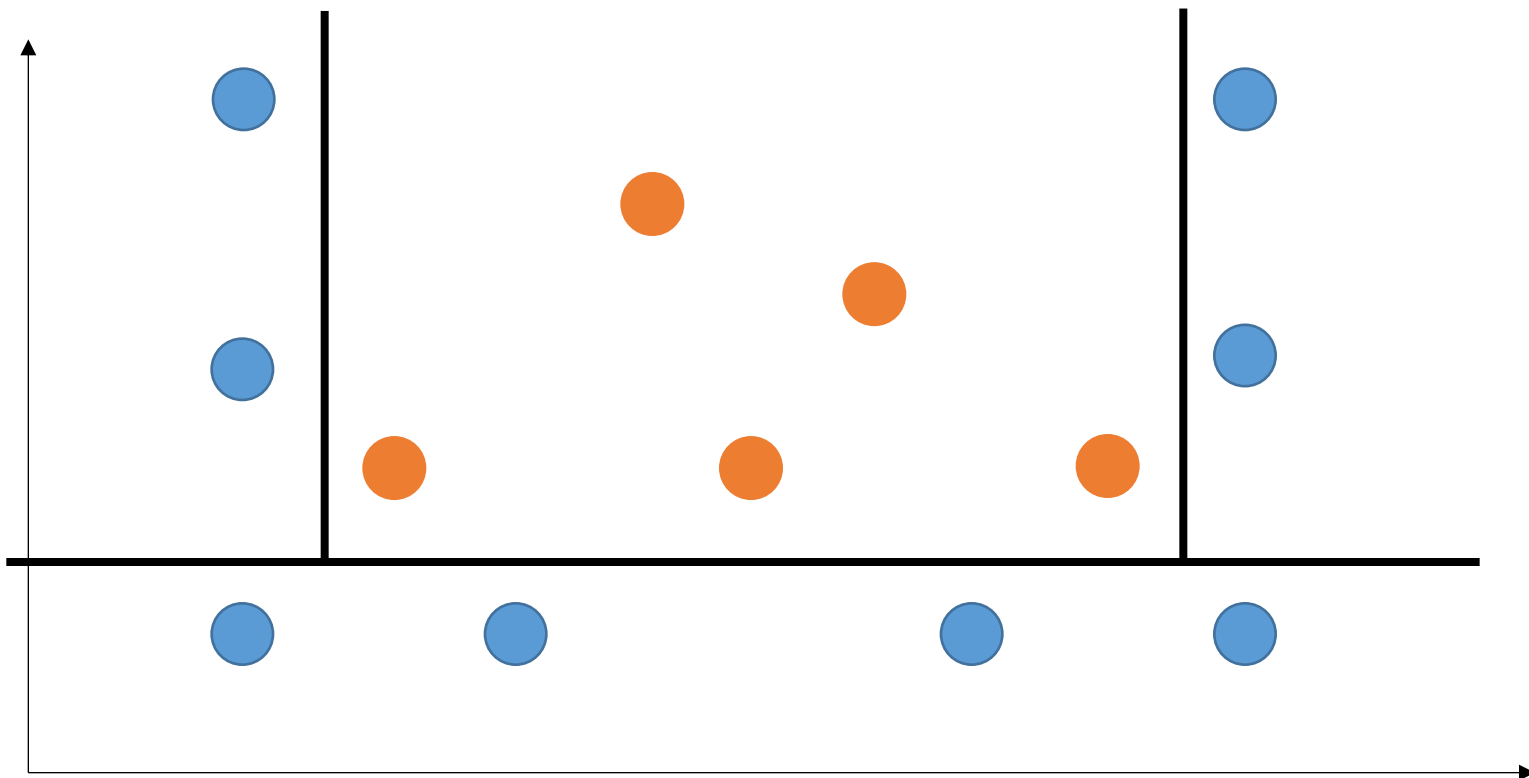
# Обучение деревьев



# Обучение деревьев



# Обучение деревьев



# Резюме

- Решающие деревья позволяют строить сложные модели, но есть риск переобучения
- Деревья строятся жадно, на каждом шаге вершина разбивается на две с помощью лучшего из предиктов
- Алгоритм довольно сложный и требует перебора всех предикатов на каждом шаге