Лекция 3

Деревья

Руслан Байназаров. email: hocop@yandex.ru, telegram: @nfthl

План

- Структура дерева
- Построение дерева
  - Решающий пень
  - Энтропия
  - “Жадный” алгоритм
  - Критерий Джини
  - Дисперсия
- Регуляризация деревьев
  - Критерии остановки
  - Стрижка (Pruning)

Полезные ссылки

- Про деревья https://towardsdatascience.com/decision-trees-d07e0f420175
- Про деревья еще https://medium.com/greyatom/decision-trees-a-simple-way-to-visualize-a-decision-dc506a403aeb
- Про деревья регрессии отдельно
  https://towardsdatascience.com/tree-based-methods-regression-trees-4ee5d8db9fe9
- Обзор по алгоритмам бустинга (доп. материал)
  https://medium.com/diogo-menezes-borges/boosting-with-adaboost-and-gradient-boosting-9cbab2a1af81
- Подробный разбор параметров LGBM и сравнение с XGBoost (доп. материал)
  https://www.analyticsvidhya.com/blog/2017/06/which-algorithm-takes-the-crown-light-gbm-vs-xgboost/

Ссылки внизу слайдов тоже полезные, но более углубленные по теме слайда.

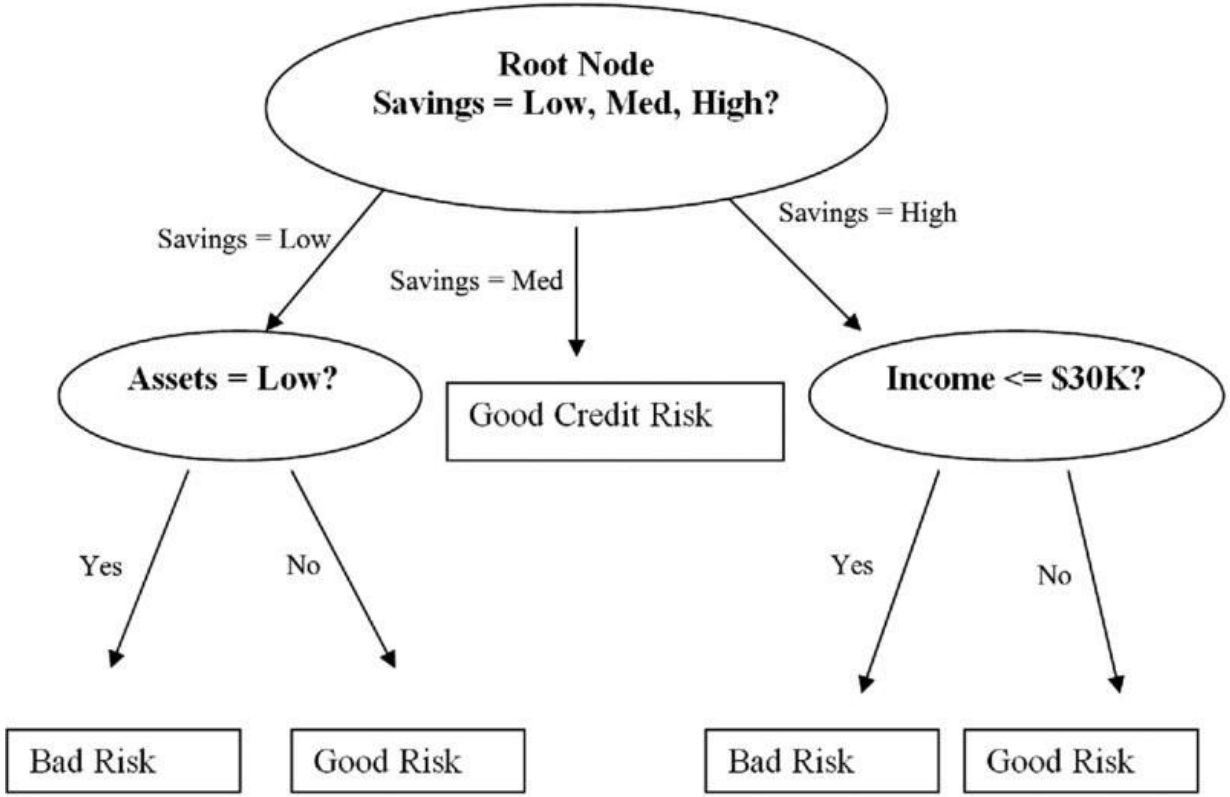
Структура дерева

Дерево решений

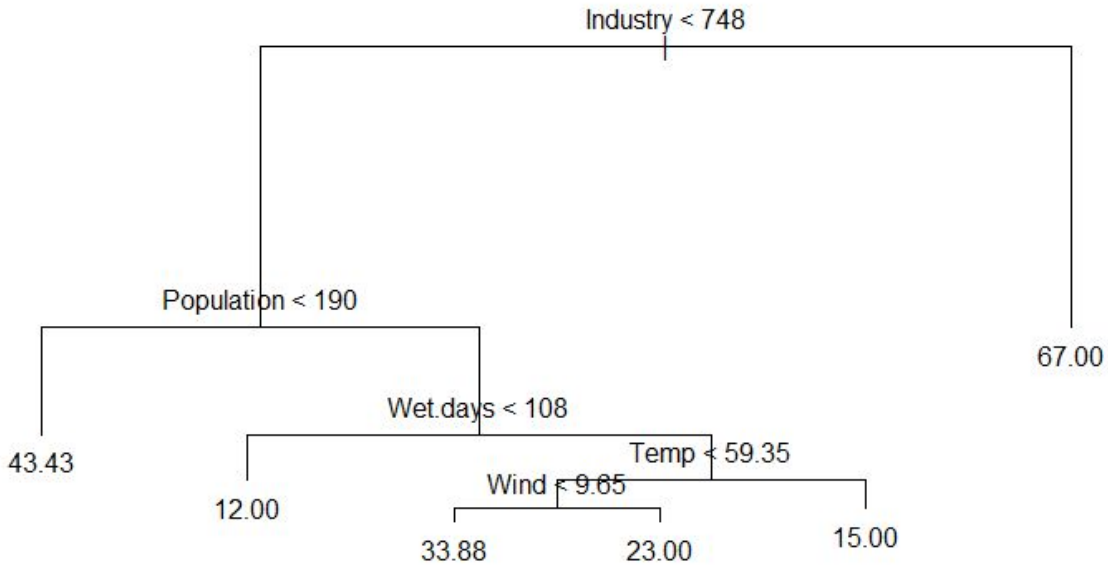
*Decision Trees (**деревья решений**)* или *CART (Classification and Regression Trees)*- алгоритм обучения с учителем, основанный на логических условиях **IF**.

В отличие от написания программы, построение дерева происходит автоматически с использованием тренировочной выборки.

Дерево классификации: предсказать кредитный скоринг



Дерево регрессии: предсказать уровень загрязнения в городе

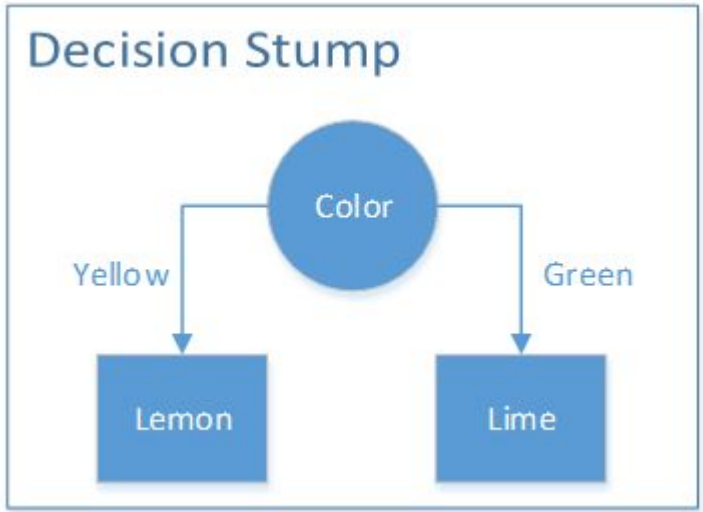


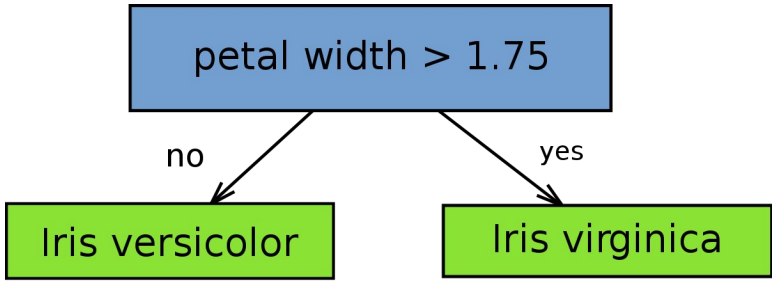
Построение деревьев

Логические условия

Логические операторы делят датасет на несколько частей

- Пороговое условие
  - применяется к числовым признакам
  - делит датасет на две части
- Деление по категориям
  - делит датасет используя один категориальный признак



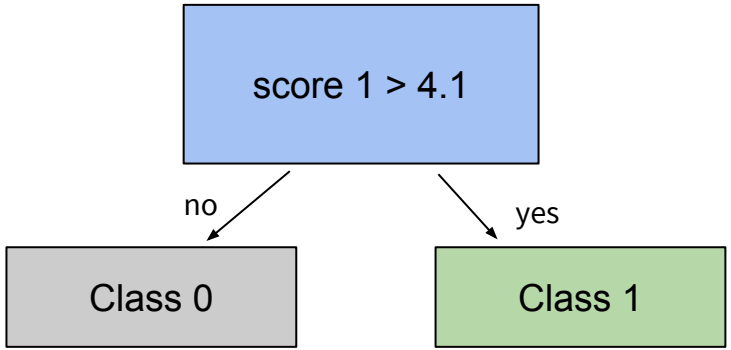


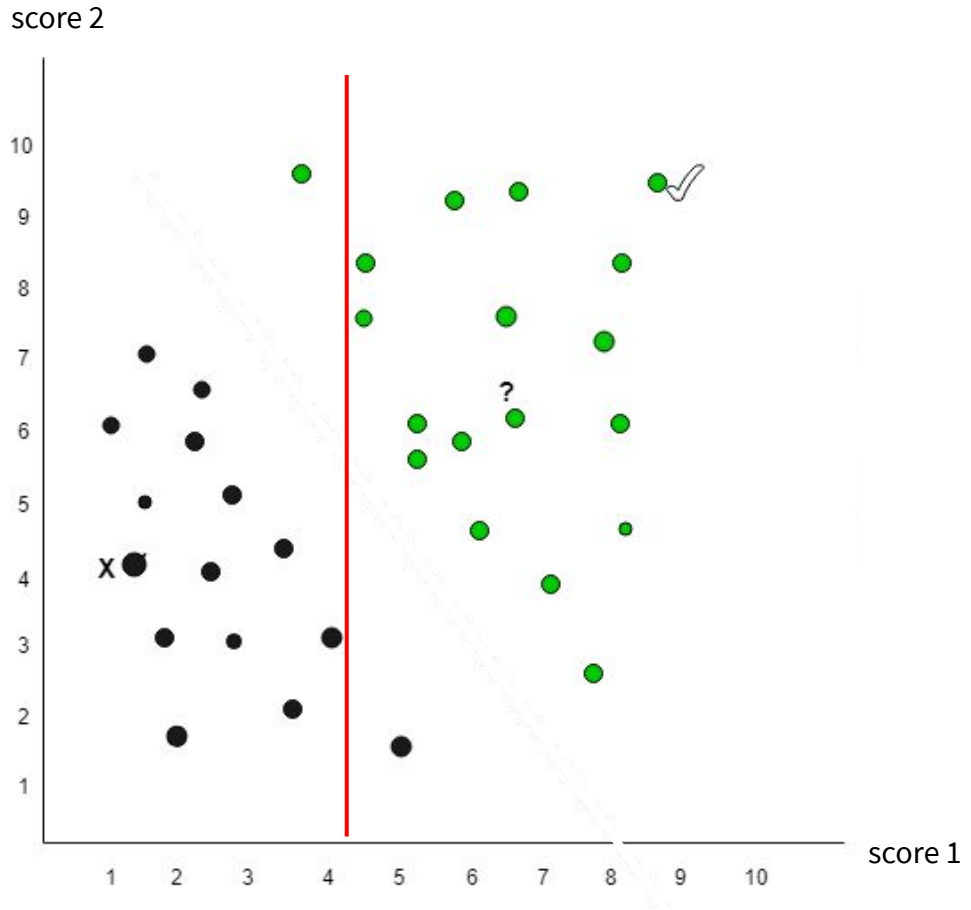
Decision stump

Модель, которая использует одно правило называется *решающий пень*

Пень - это дерево с *глубиной* 1

Зам. Пороговое правило не может разделить датасет по наклонной линии





Выводим критерий качества

Всего: 15 черных точек и 18 зеленых

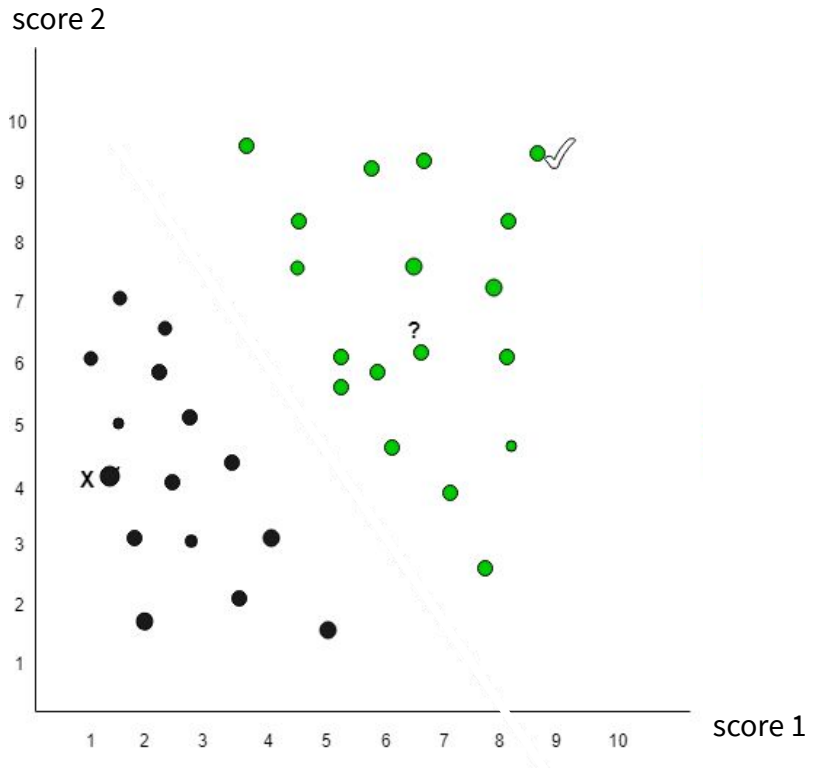
Это описывается частотой (frequency):

$$f = (14, 18)$$

Или вероятностью:

$$P = (0.44, 0.56)$$

В таком распределении большая неопределенность



Выводим критерий качества

Слева от прямой: 14 черных точек и 1 зеленая
Справа: 1 черная и 17 зеленых

Это описывается частотой (frequency):

$$f_L = (1, 14)$$

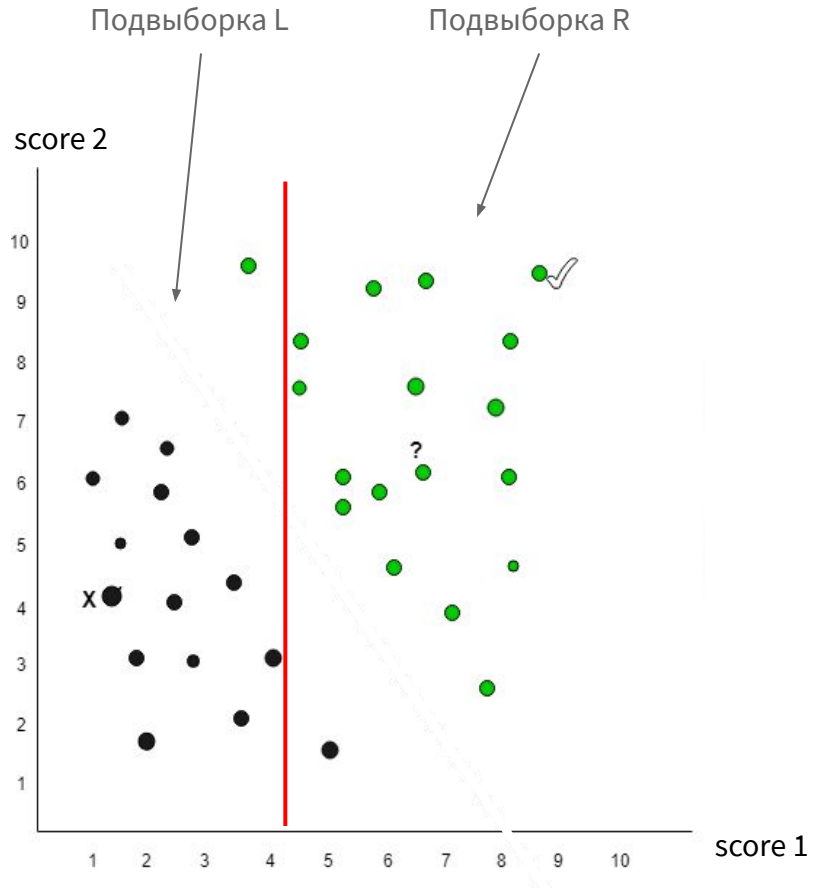
$$f_R = (17, 1)$$

Или вероятностью:

$$P_L = (0.07, 0.93)$$

$$P_R = (0.95, 0.05)$$

А какие вероятности мы хотим видеть?



Энтропия

Энтропия - это мера неопределенности.

$$H(P) = -\sum_i p_i log(p_i)$$

Энтропия также показывает, сколько информации содержится в послании.

Например: есть 8 равновероятных исходов некоторого эксперимента. Сколько бит нужно, чтобы их закодировать?

$$2^3 = 8$$

Ответ: 3 бита дают 8 комбинаций.

$$3 = \log_2(8)$$

3 бита - количество информации, которое содержится в послании с восемью возможными исходами.

$$3 = -\log_2(\frac{1}{8})$$

$$3 = -8 \cdot \frac{1}{8}\log_2(\frac{1}{8})$$

$$3 = -\sum_{i=1}^{8} \frac{1}{8}\log_2(\frac{1}{8})$$

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

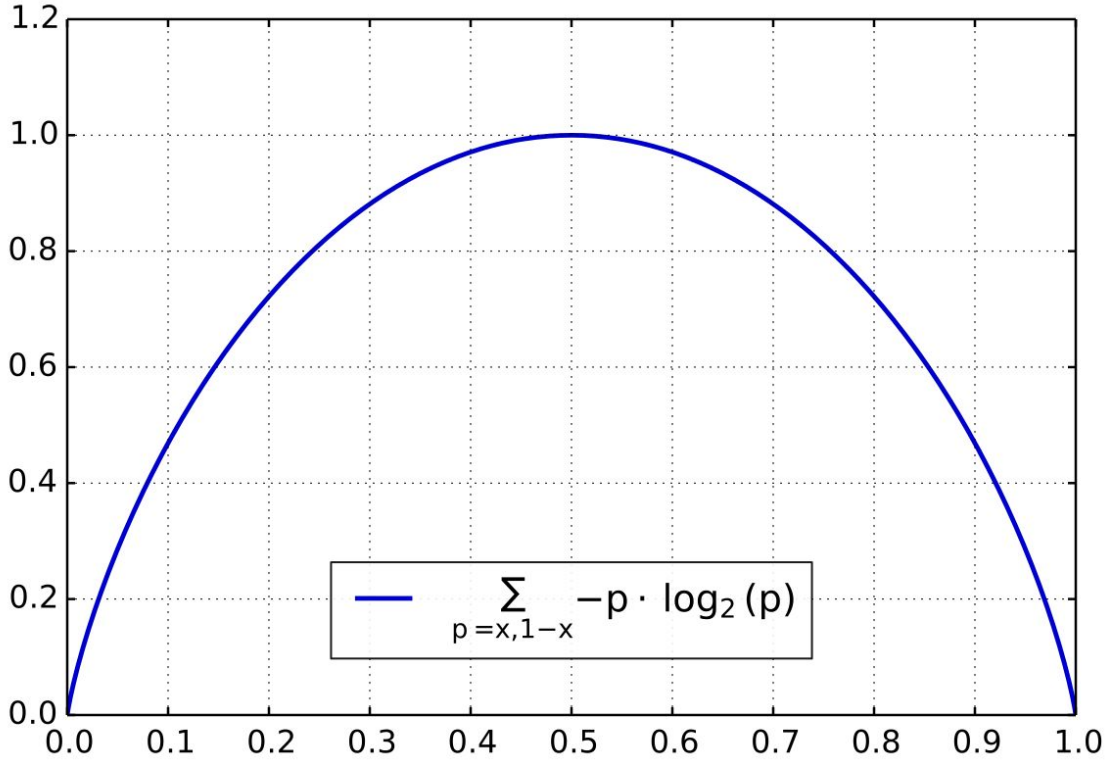
Оно уже не совсем в битах. Энтропия - это общее понятие *информации*

Энтропия

Энтропия максимальна когда вероятности исходов равны, и равна нулю, когда вероятность одного из исходов 1, а остальных - 0.

$$H(P) = -\sum_i p_i log(p_i)$$

На картинке - случай с двумя исходами



Энтропия

$$P = (0.44, 0.56)$$

$$P_L = (0.07, 0.93)$$

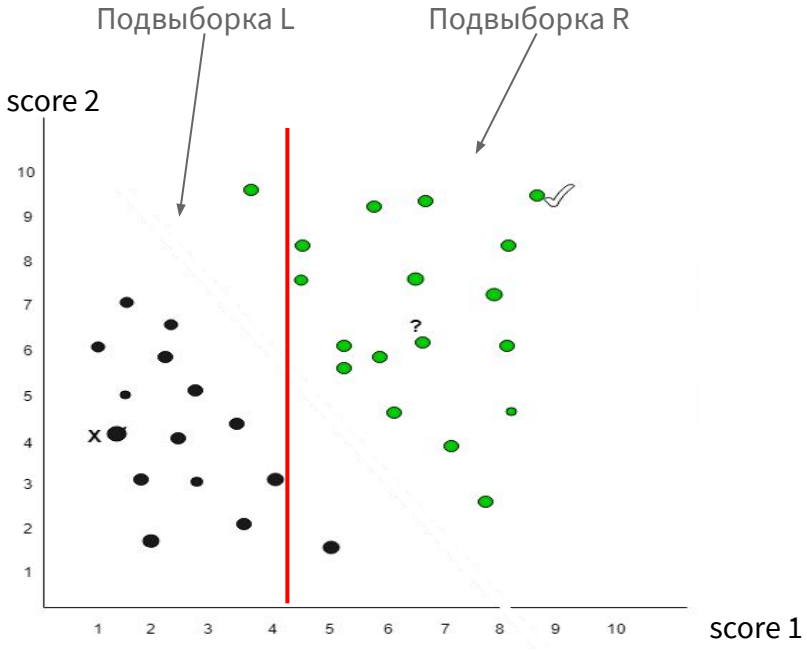
$$P_R = (0.95, 0.05)$$

$$H(P) = -\sum_i p_i log(p_i)$$

$$H(P) = -0.44\log(0.44) - 0.56\log(0.56) \approx 0.3$$

$$H(P_L) = -0.07\log(0.07) - 0.97\log(0.97) \approx 0.094$$

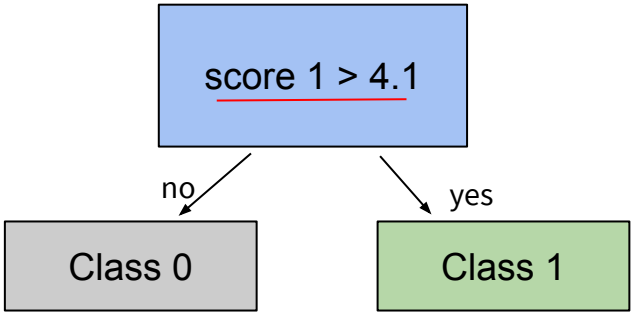
$$H(P_R) = -0.05\log(0.05) - 0.95\log(0.95) \approx 0.086$$

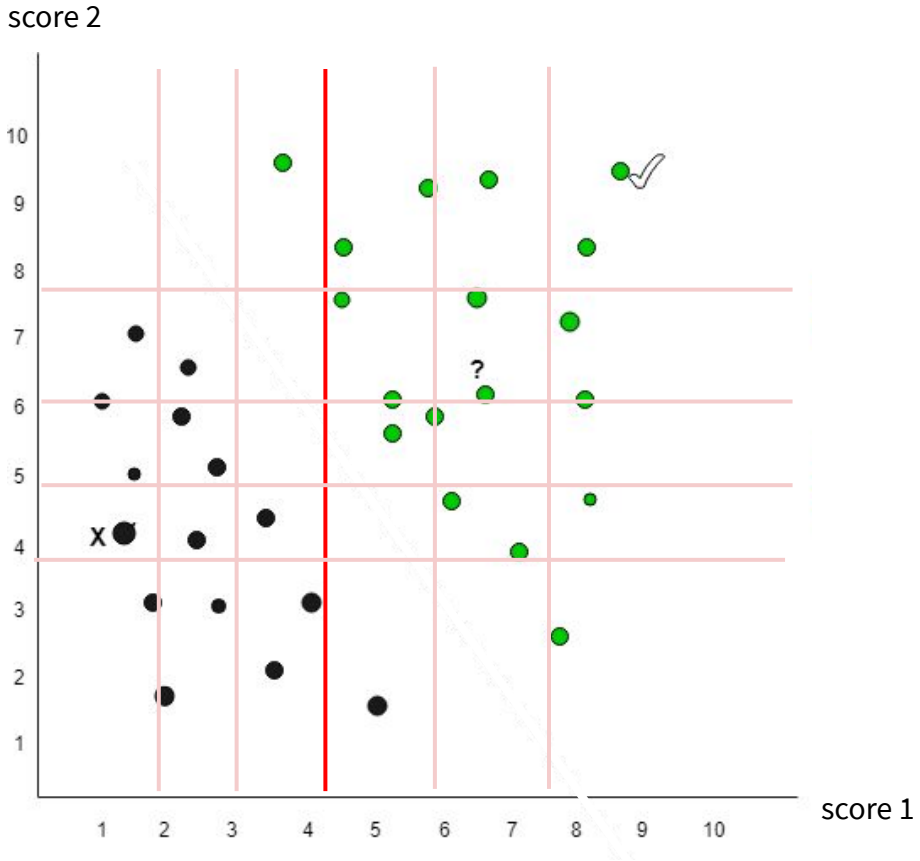


Чему равна H при P = (0, 1)?

Энтропия

Среди всех возможных разбиений находят такое, которое сильнее всего понижает энтропию





$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Энтропия

А помните, была кросс-энтропия?

$$H(\hat{p}, p) = -\sum_i p_i log(\hat{p}_i)$$

- для одного сэмпла

Как она связана с этой энтропией?

$$H(P) = -\sum_i p_i log(p_i)$$

В этих формулах p означает не одно и то же!

$$Loss = -\frac{1}{N}\sum_{k=1}^{N}\sum_{k=1}^{K}[y_k = k]\log(\hat{p}_k)$$

$$Loss = -\sum_{k=1}^{K}\frac{1}{N}\sum_{k=1}^{N}[y_k = k]\log(\hat{p}_k)$$

$$Loss = -\sum_{k=1}^{K} p_k \log(\hat{p}_k)$$

- для всех сэмплов

$$p_k = \hat{p}_k$$

Критерий Джини

Помимо энтропии есть еще критерий Джини, который выводится из функции потерь Бриера.

Критерий Джини, также, как и энтропия, показывает неопределенность вероятностного распределения.

Критерий Джини вычисляется быстрее, так как не содержит логарифмов.

$$H(R) = \min_{\sum_k c_k = 1} \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \sum_{k=1}^{K} (c_k - [y_i = k])^2.$$

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

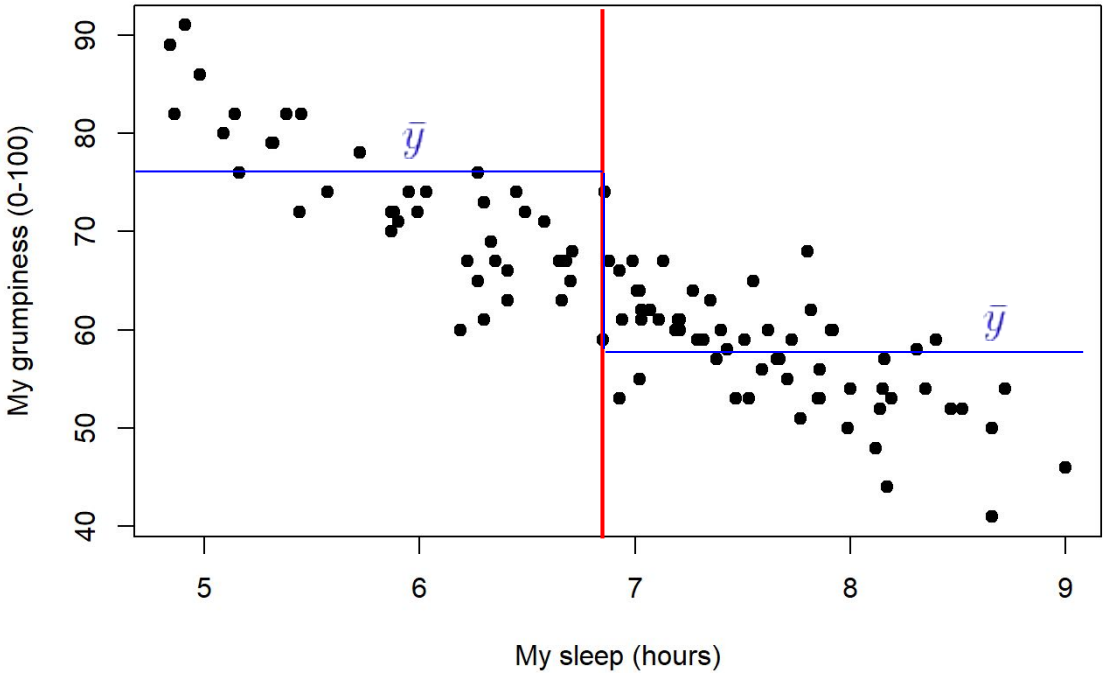
Дисперсия

А если целевая переменная - число?
Какая тогда мера неопределенности?

Дисперсия:

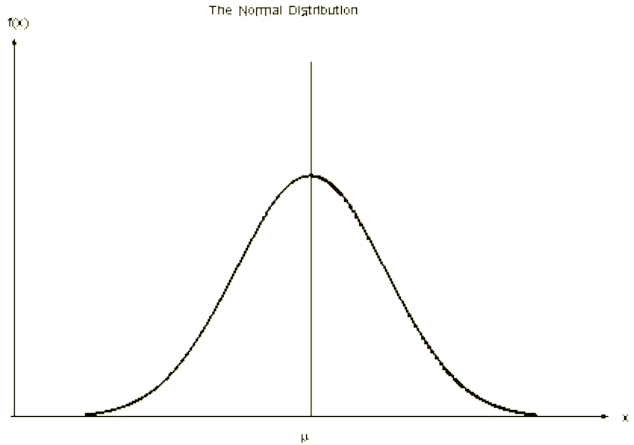
$$H(R) = \sum_i (y_i - \bar{y})^2$$

А как дисперсия связана со среднеквадратичным отклонением предсказания от реального значения?



Дисперсия

А как дисперсия связана с энтропией?



$$H(R) = -\int_{-\inf}^{\inf} p\log(p)dx$$

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{(x-\mu)^2}{2\sigma^2})$$

$$p(x) = \mathcal{N}(x|\mu,\sigma^2)$$

$$\log(p) = -\frac{(x-\mu)^2}{2\sigma^2} - \log\sqrt{2\pi\sigma^2}$$

$$\mathbb{E}(x-\mu)^2 = \sigma^2$$

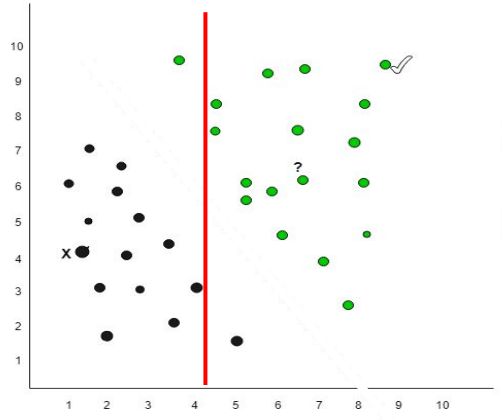
$$H(R) = -\log\sqrt{2\pi\sigma^2} - \frac{1}{2}$$

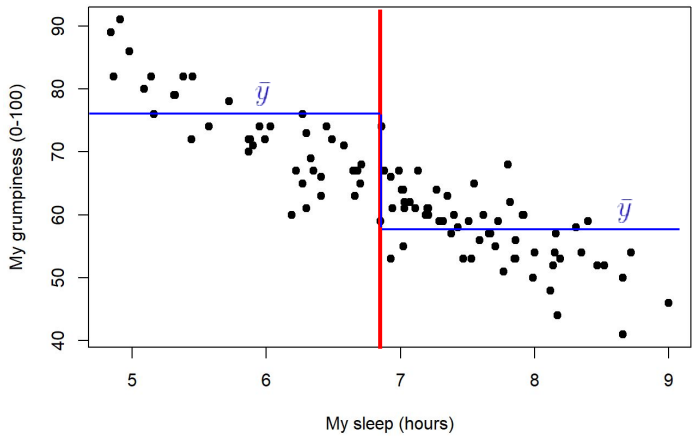
Резюме

Чтобы выбрать наилучшее логическое условие для разделения тренировочной на несколько подвыборок, используют следующие критерии:

- Для классификации:
  - Энтропия
  - Критерий Джини
- Для регрессии:
  - Стандартное отклонение

Все эти критерии говорят о том, насколько **уменьшилась неопределенность** при разделении тренировочного набора на подвыборки с помощью данного правила



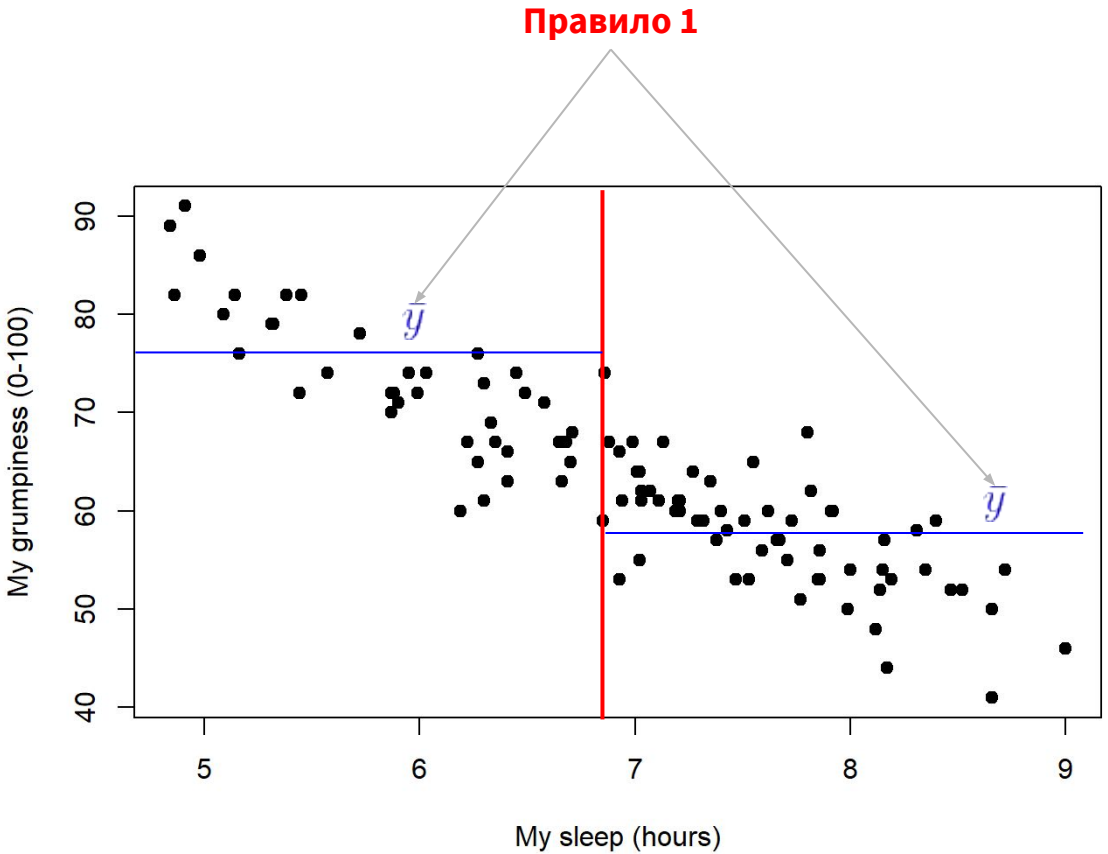


Дерево решений

Дерево

Итак, мы разбили датасет на две подвыборки, в каждой из которых уменьшилась (допустим) энтропия.

Что дальше?

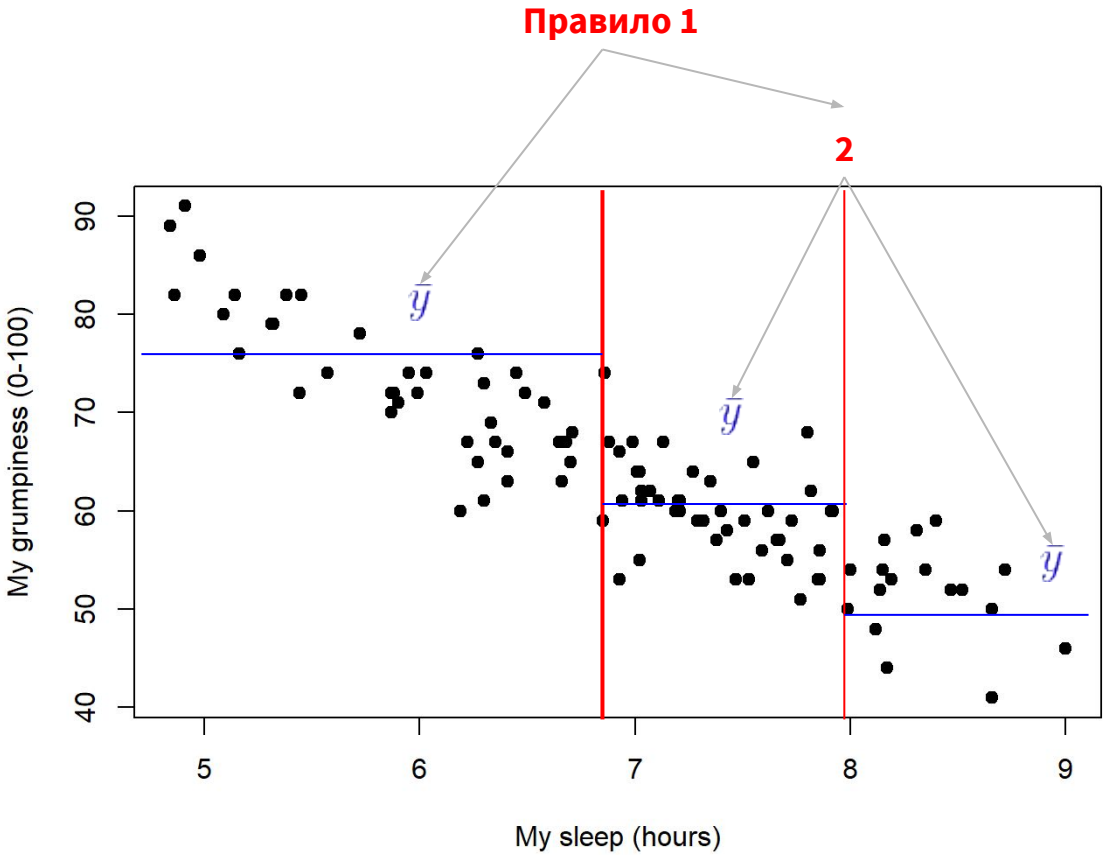


Дерево

Итак, мы разбили датасет на две подвыборки, в каждой из которых уменьшилась (допустим) энтропия.

Что дальше?

Делим еще раз!



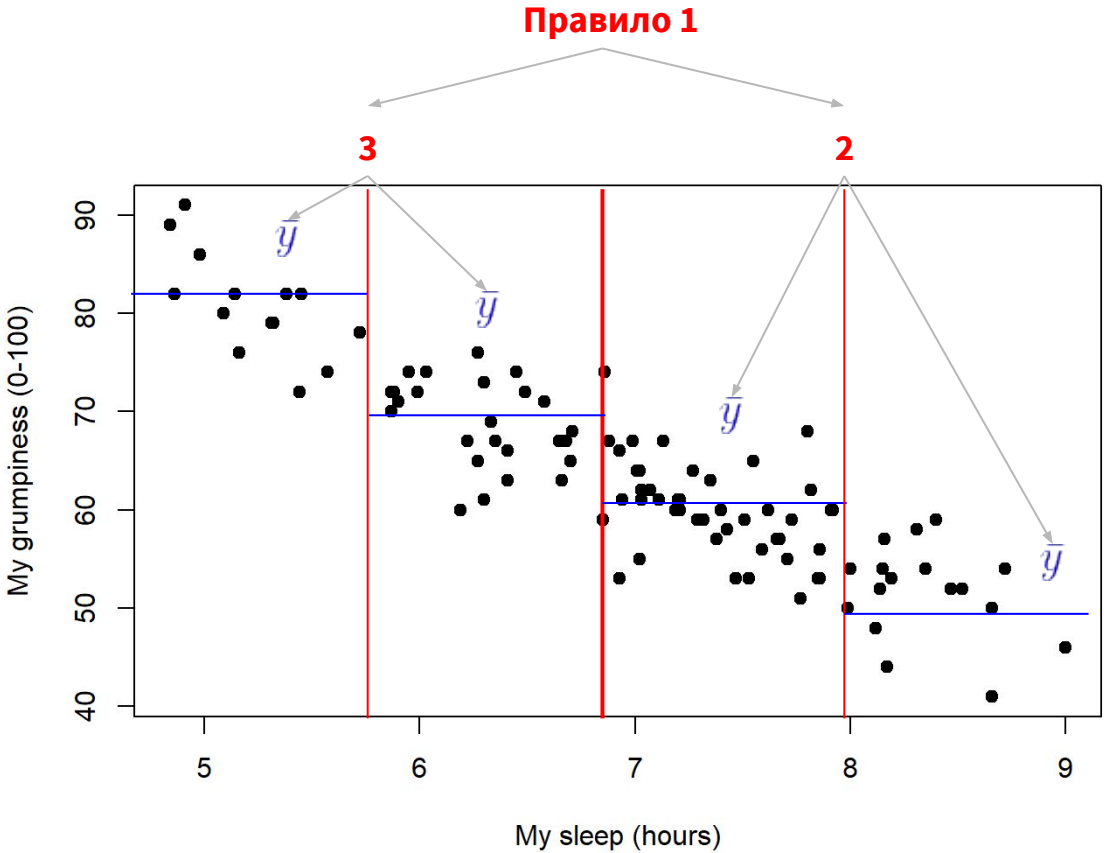
Дерево

Итак, мы разбили датасет на две подвыборки, в каждой из которых уменьшилась (допустим) энтропия.

Что дальше?

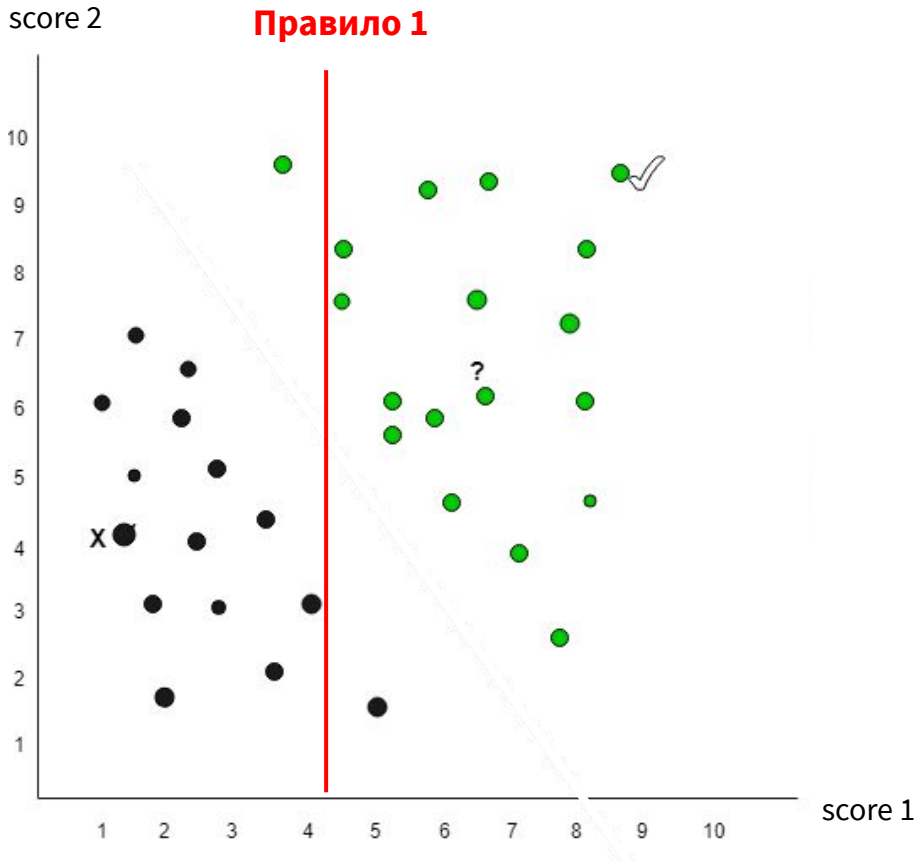
Делим еще раз!

Делим еще раз!



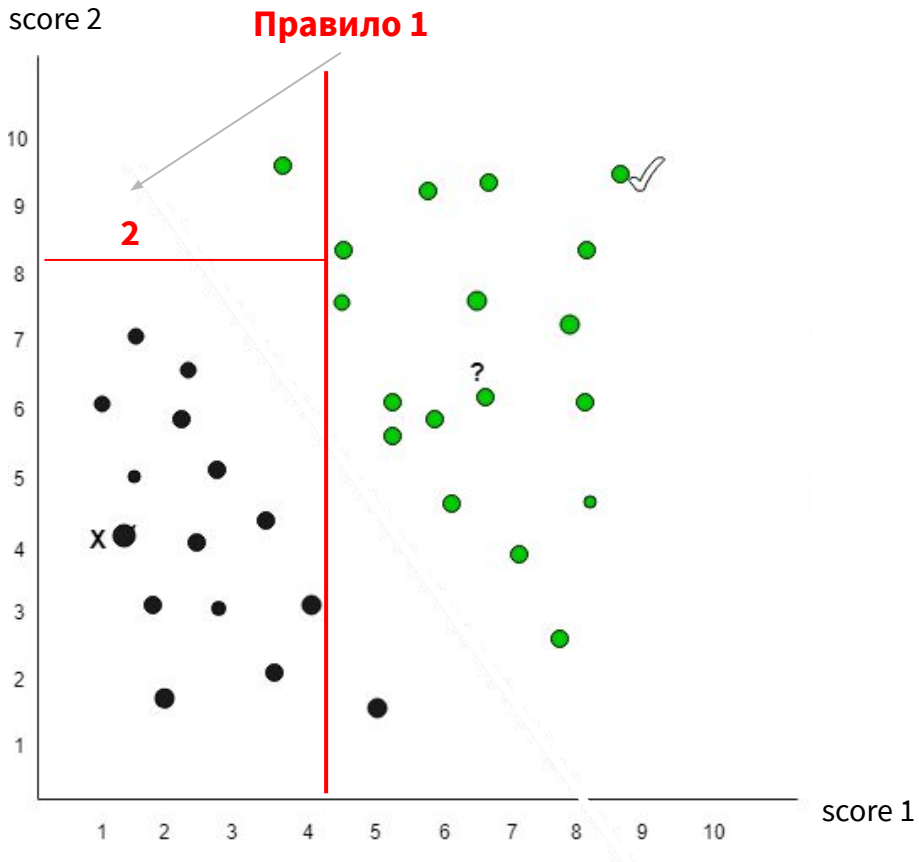
Дерево

А как оно будет выглядеть в случае классификации?



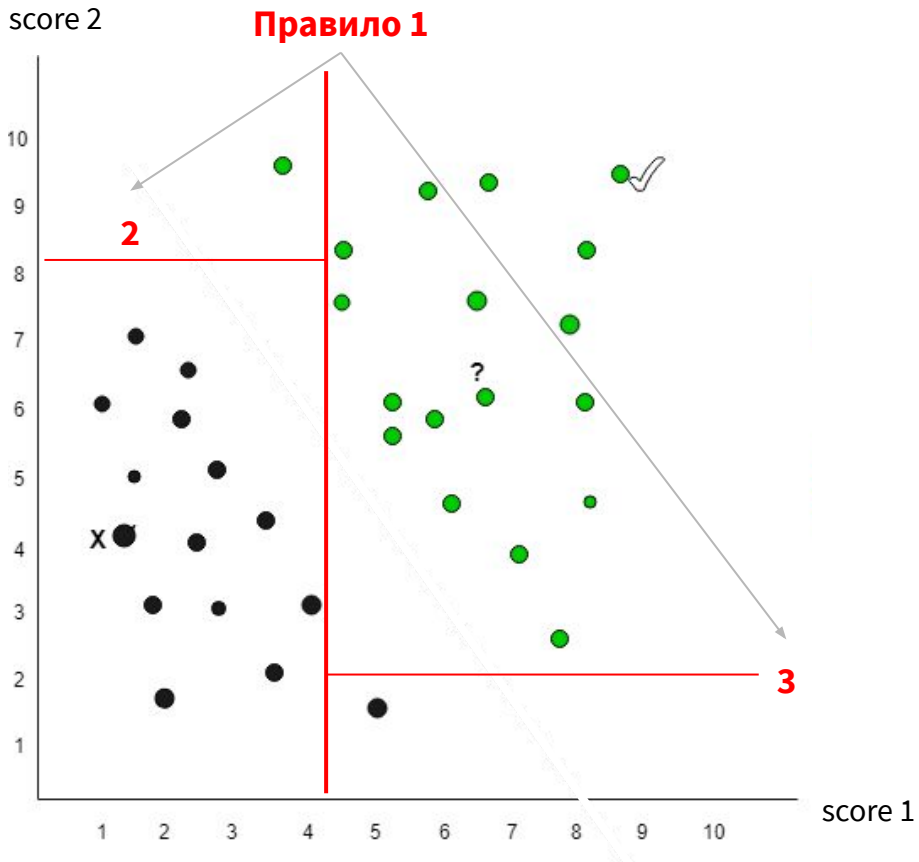
Дерево

А как оно будет выглядеть в случае классификации?

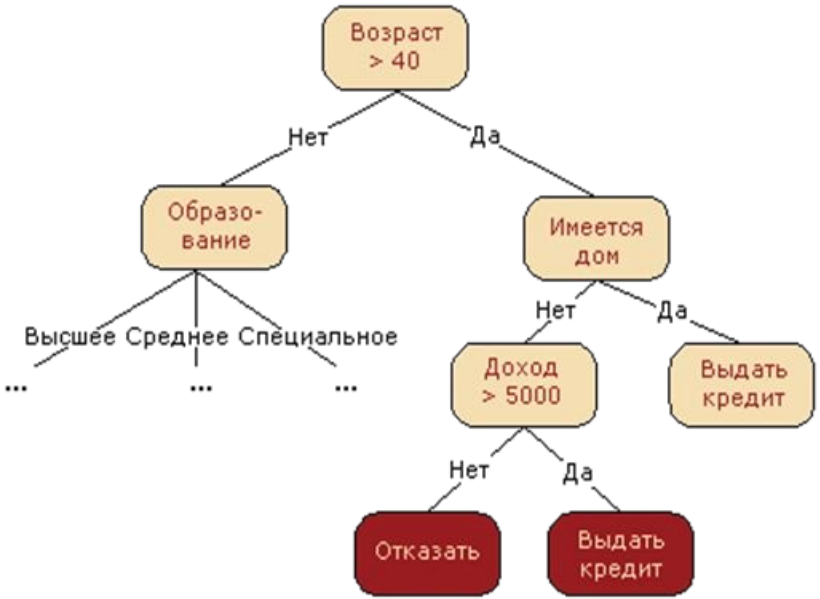


Дерево

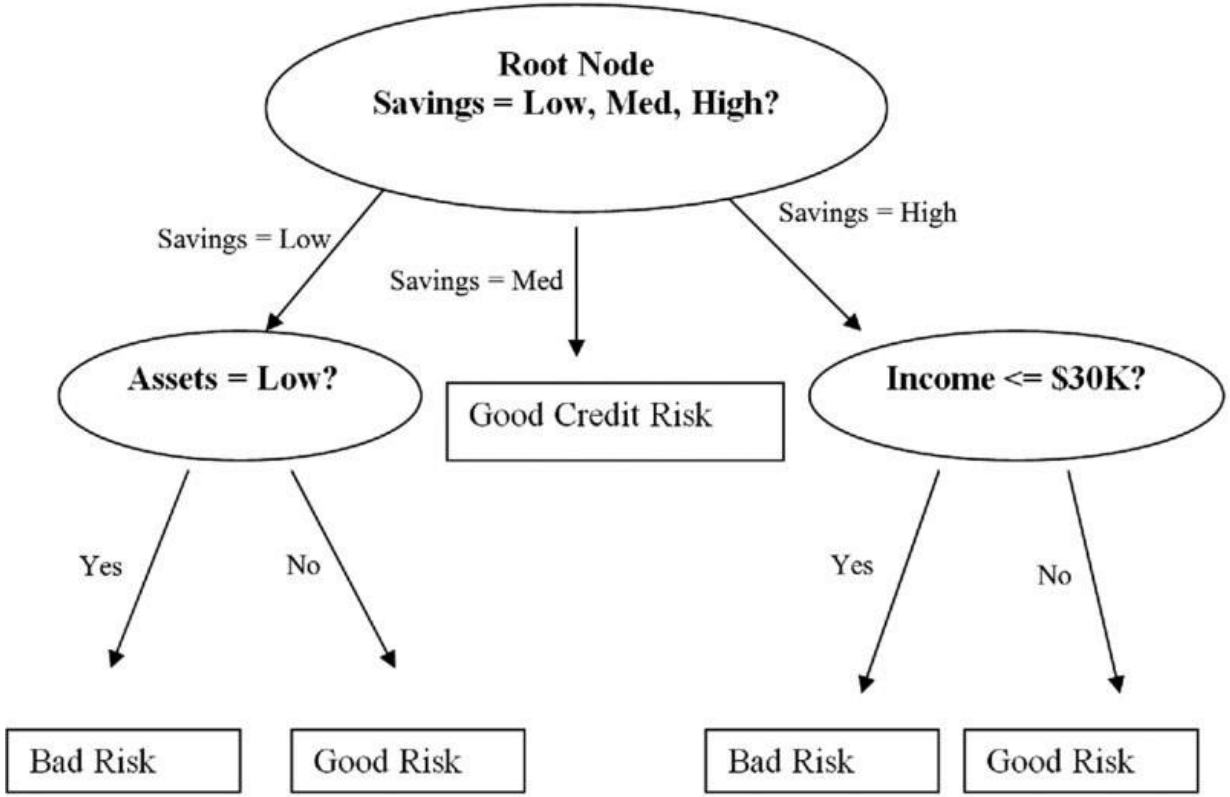
А как оно будет выглядеть в случае классификации?



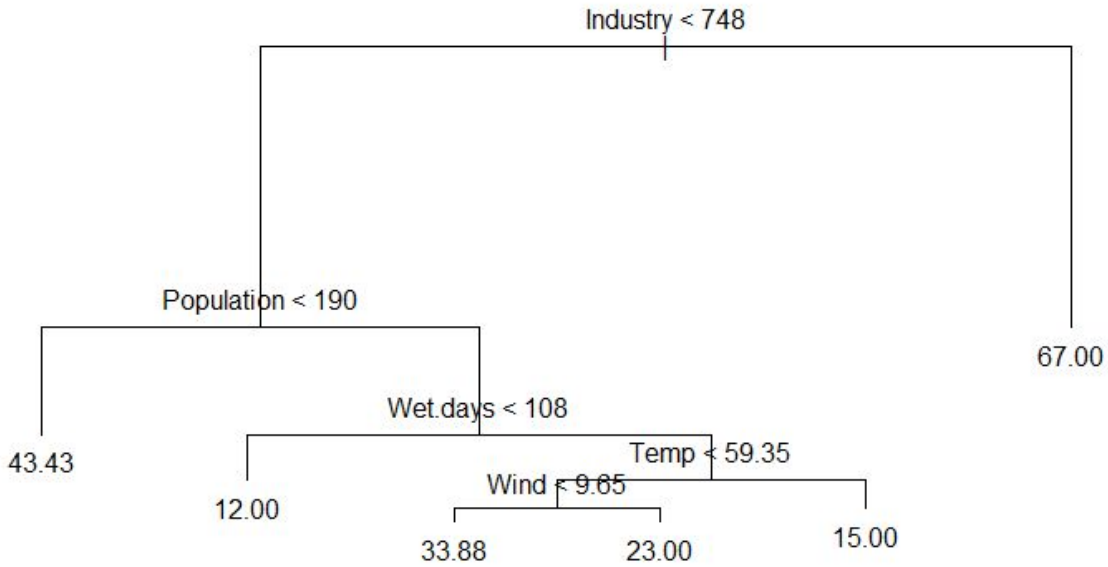
Дерево


Возраст
> 40
Нет
Да
Образо-
вание
Имеется
дом
Высшее
Среднее
Специальное
...
...
...
Нет
Да
Доход
> 5000
Выдать
кредит
Нет
Да
Отказать
Выдать
кредит

Дерево классификации: предсказать кредитный скоринг



Дерево регрессии: предсказать уровень загрязнения в городе



Дерево

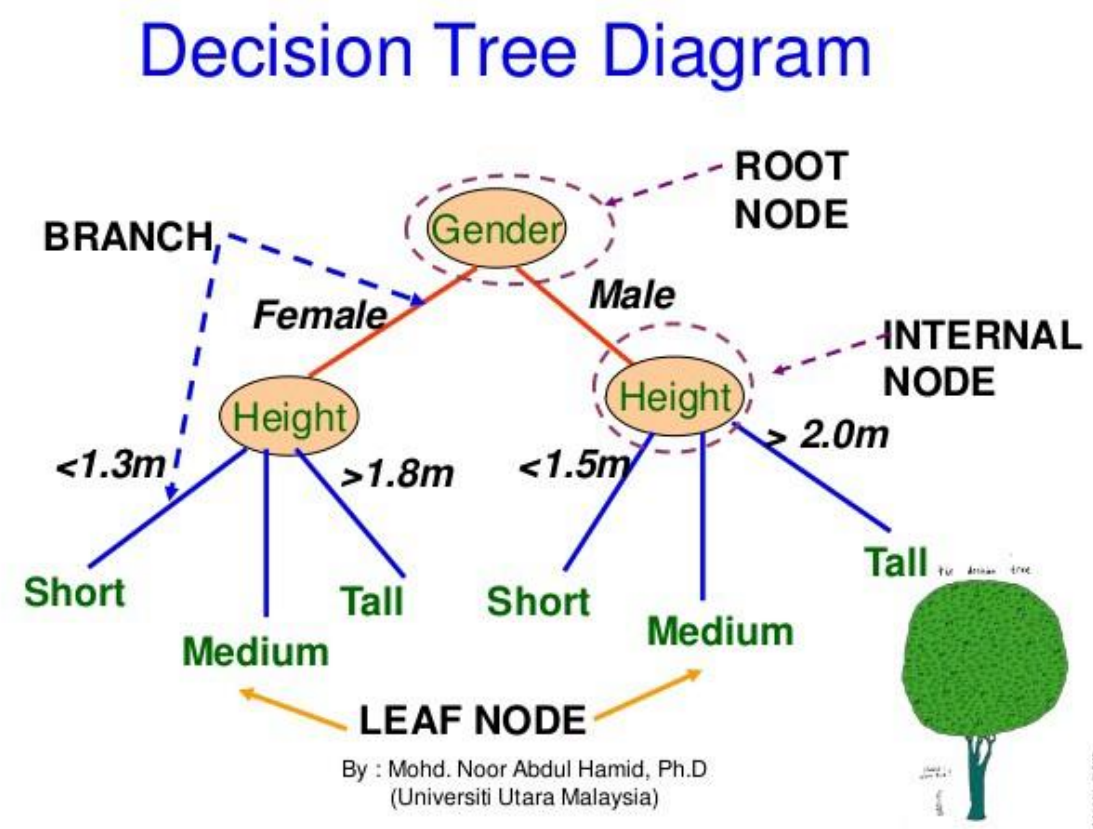
У дерева есть:

- Узлы
  - Корневой узел
  - Внутренние узлы
  - Листья
- Ветки

В каждом узле происходит деление датасета на несколько частей

Из каждого узла исходят несколько веток, которые идут в другие узлы

Узлы, из которых не выходят ветки называются листьями



Регуляризация деревьев

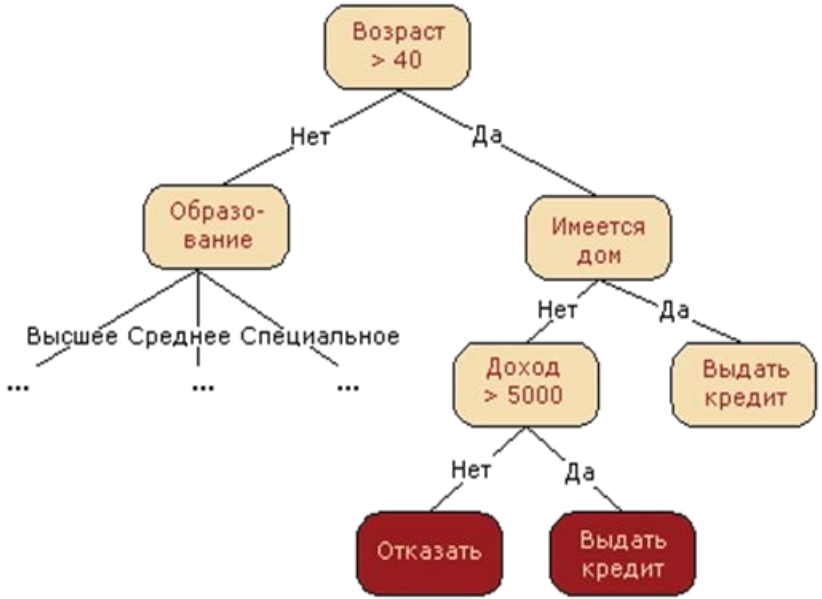
Критерий остановки

Когда останавливать деление датасета?

a) Когда достигнем нужной глубины
b) Когда в узле будет число точек меньше, чем *min_samples_split*
c) ...

Критериев остановки много, давайте лучше посмотрим на параметры конструктора класса *DecisionTreeClassifier:*

https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html



Стрижка деревьев

Стрижка (pruning) - метод регуляризации дерева.

В sklearn реализован алгоритм *Minimal Cost-Complexity Pruning*

Суть метода в том, чтобы найти дерево, которое минимизирует следующую функцию ошибки:

$$R_\alpha(T) = R(T) + \alpha|T|$$

Параметр альфа известен как *complexity parameter*

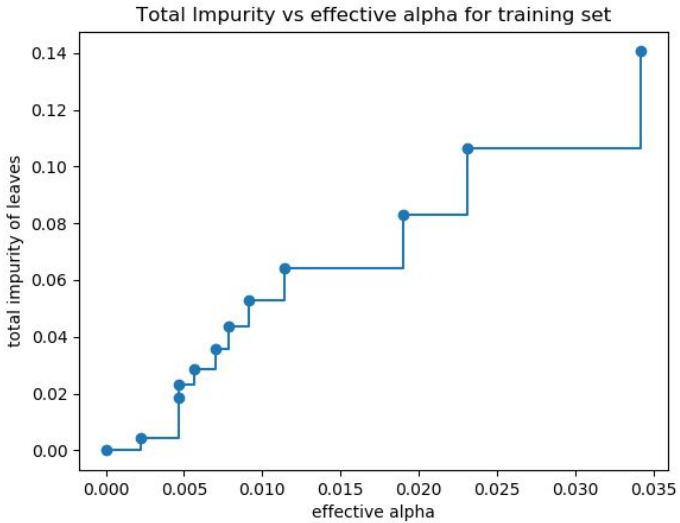
1. Обучают дерево
2. Создают последовательность деревьев $T_0 \dots T_m$, где каждое следующее дерево имеет меньше веток, чем предыдущее.
   T_0 -- исходное дерево
   T_m -- только корневой узел
3. Из всех этих деревьев выбирают то, у которого функция ошибки минимальна.

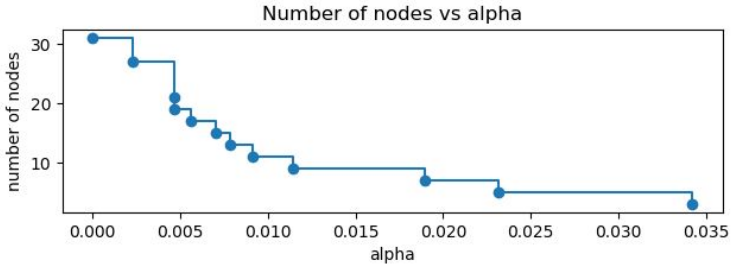
На каждой итерации удаляют ту ветку (subtree), у которой минимален следующий критерий:

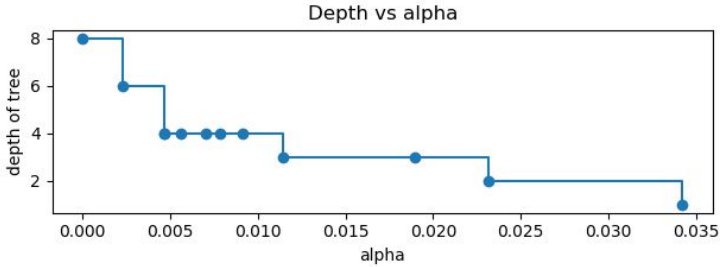
$$\frac{\mathrm{err}(\mathrm{prune}(T,t),S) - \mathrm{err}(T,S)}{|\mathrm{leaves}(T)| - |\mathrm{leaves}(\mathrm{prune}(T,t))|}$$

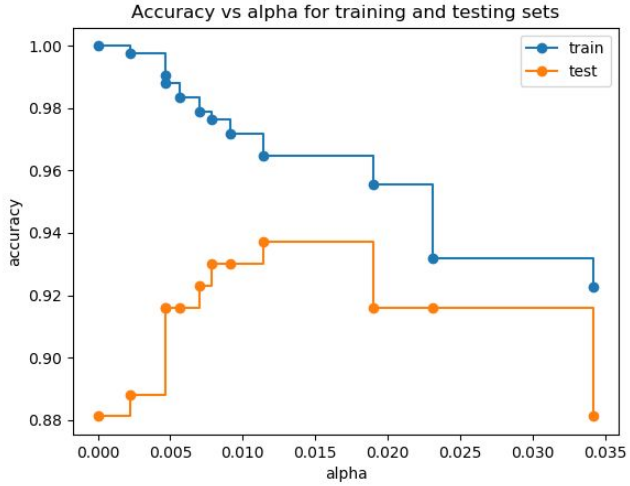
https://scikit-learn.org/stable/modules/tree.html#minimal-cost-complexity-pruning

Стрижка деревьев









https://scikit-learn.org/stable/auto_examples/tree/plot_cost_complexity_pruning.html

Практика