

# Базовые модели машинного обучения: деревья решений

---

Никольская Анастасия Николаевна

Першин Антон Юрьевич, Ph.D.

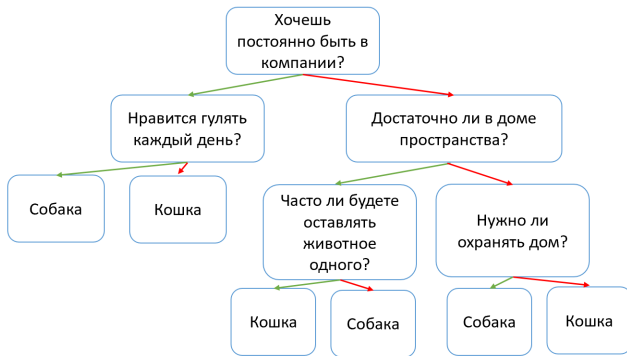
Программа «Большие данные и распределенная цифровая  
платформа»

Санкт-Петербургский государственный университет

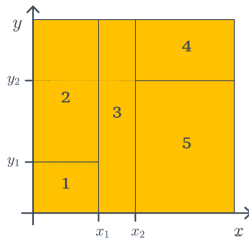
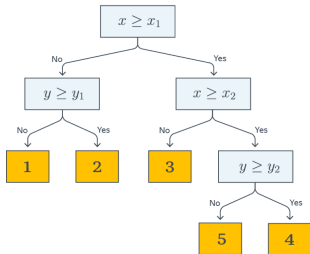
Практика по дисциплине «Технологии ИИ»  
7 апреля 2023 г.

- Дерево решений — это метод представления решающих правил в иерархической структуре, состоящей из элементов двух типов — узлов и листьев.
- В узлах находятся решающие правила.
- В простейшем случае, в результате проверки правила, множество примеров, попавших в узел, разбивается на два подмножества.
- К каждому подмножеству вновь применяется некоторое правило, и процедура рекурсивно повторяется, пока не будет достигнуто некоторое условие остановки алгоритма.
- В последнем узле проверка и разбиение не производится и он объявляется листом.
- Лист определяет решение для каждого попавшего в него примера.

# Пример дерева решений



## Пример дерева решений (2)



Источник картинки

# Определение решающего дерева

Пусть задано бинарное дерево, в котором:

- каждой внутренней вершине  $v$  приписан предикат  $Split(v)$ , каждой листовой вершине приписан прогноз  $Ans(v) \in Y$ , где  $Y$  — область значений целевой переменной
- В ходе предсказания осуществляется проход по этому дереву к некоторому листу. Для каждого примера движение начинается из корня.
- В очередной внутренней вершине роход продолжится вправо, если  $Split(v) = 1$ , и влево, если  $Split(v) = 0$ . Проход продолжается до момента, пока не будет достигнут некоторый лист и будет получен  $Ans(v)$ .

Пусть задано бинарное дерево, в котором:

- Выученная функция является кусочно-постоянной (что из этого следует?)
- Дерево решений не может экстраполировать зависимости за границы области значений обучающей выборки
- Дерево решений способно идеально приблизить обучающую выборку и ничего не выучить, если выродится (в каждый лист попадет один пример)

# Жадный алгоритм построения дерева

- Создаём вершину  $v$
- Если выполнен критерий остановки  $Stop(S_m)$ , останавливаемся, объявляем эту вершину листом и присваиваем ей ответ  $Ans(S_m)$
- Иначе: находим предикат  $Split(S_m)$ , обеспечивающий наилучшее разбиение выборки по некоторому критерию.
- Повторяем процедуру для  $S_{m_i}$  и  $S_{m_j}$  до достижения критерия остановки.
- В разных алгоритмах применяются разные эвристики для "ранней остановки" или "отсечения чтобы избежать построения переобученного дерева.

## Жадный алгоритм построения дерева (2)

- $Ans(S_m)$  в случае задачи классификации — метка самого частого класса или оценка дискретного распределения вероятностей классов для объектов, попавших в этот лист; в случае задачи регрессии — среднее, медиана или другая статистика;
- Критерий остановки — функция, которая решает, нужно ли продолжать ветвление или пора остановиться. Это может быть какое-то тривиальное правило, равенство или близость энтропии нулю или достижение максимального числа итераций.
- Строгой теории, которая бы связывала оптимальность выбора разных вариантов критериев разбиения и разных метрик классификации и регрессии, в общем случае не существует.



- Энтропия Шеннона определяется для системы с  $N$  возможными состояниями как:

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

где  $p_i$  – вероятности нахождения системы в  $i$ -ом состоянии

- Энтропия может рассматриваться как мера неоднородности подмножества по представленным в нём классам
- Энтропия максимальна при равномерном распределении классов и равна 0, если подмножество содержит только один класс

- Лучшим атрибутом разбиения будет тот, который обеспечит максимальное снижение энтропии результирующего подмножества относительно родительского
- Тогда лучшим критерием будет тот, который обеспечит максимальный прирост информации (  $Info = -S$ ):

$$Gain(A) = Info(S) - Info(S_A) = Info(S) - \sum_{i=1}^q \frac{N_i}{N} S_i, \quad (2)$$

где  $S$  - множество до разбиения,  $S_A$  - разбиение множества по атрибуту  $A$  на  $q$  групп

- Т.о. задача выбора атрибута разбиения в узле заключается в максимизации величины  $Gain(A)$

- В основе статистического подхода лежит использование неопределенности Джини
- Он показывает — насколько часто случайно выбранный пример обучающего множества будет распознан неправильно
- Т.о он показывает расстояние между целевым распределением и распределением предсказаний
- Неопределенность Джини можно рассчитать по формуле:

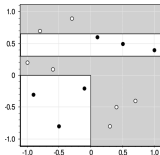
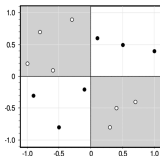
$$Gini(Q) = 1 - \sum_{i=1, N} p_i^2, \quad (3)$$

где  $S$  - множество до разбиения,  $S_A$  - разбиение множества по атрибуту  $A$

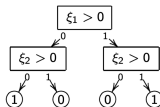
- Т.о. максимизируется число объектов одного класса, попавших в поддерево
- Неопределенность Джини и прирост информации работают почти одинаково

# Неоптимальность критериев разбиения

- Жадный алгоритм не даст оптимального решения задачи XOR.
- Иногда оптимальное с точки зрения выбранной метрики дерево получается с критерием ветвления, построенным по другой метрике



Оптимальное дерево



Дерево, построенное жадным алгоритмом



- Если признак принимает значения  $C \in \{c_1 \dots c_m\}$ , то простой перебор даст  $2^{m-1} - 1$  сплитов
- Перебирать их слишком долго, и часто их пытаются упорядочить
- Для бинарной задачи можно упорядочить категории по доле примеров класса 1 со значением  $c_i$
- Для регрессии - по среднему значению таргета

Чтобы дерево не переобучалось, ветвление обычно останавливают по одному из следующих критериев:

- Ограничение по максимальной глубине дерева
- Ограничение на минимальное количество объектов в листе
- Ограничение на максимальное количество листьев в дереве
- Требование, чтобы функционал качества при делении текущей подвыборки на две улучшался не менее чем на  $S$  процентов