Решающие деревья. Случайный лес.

Степанов Иван

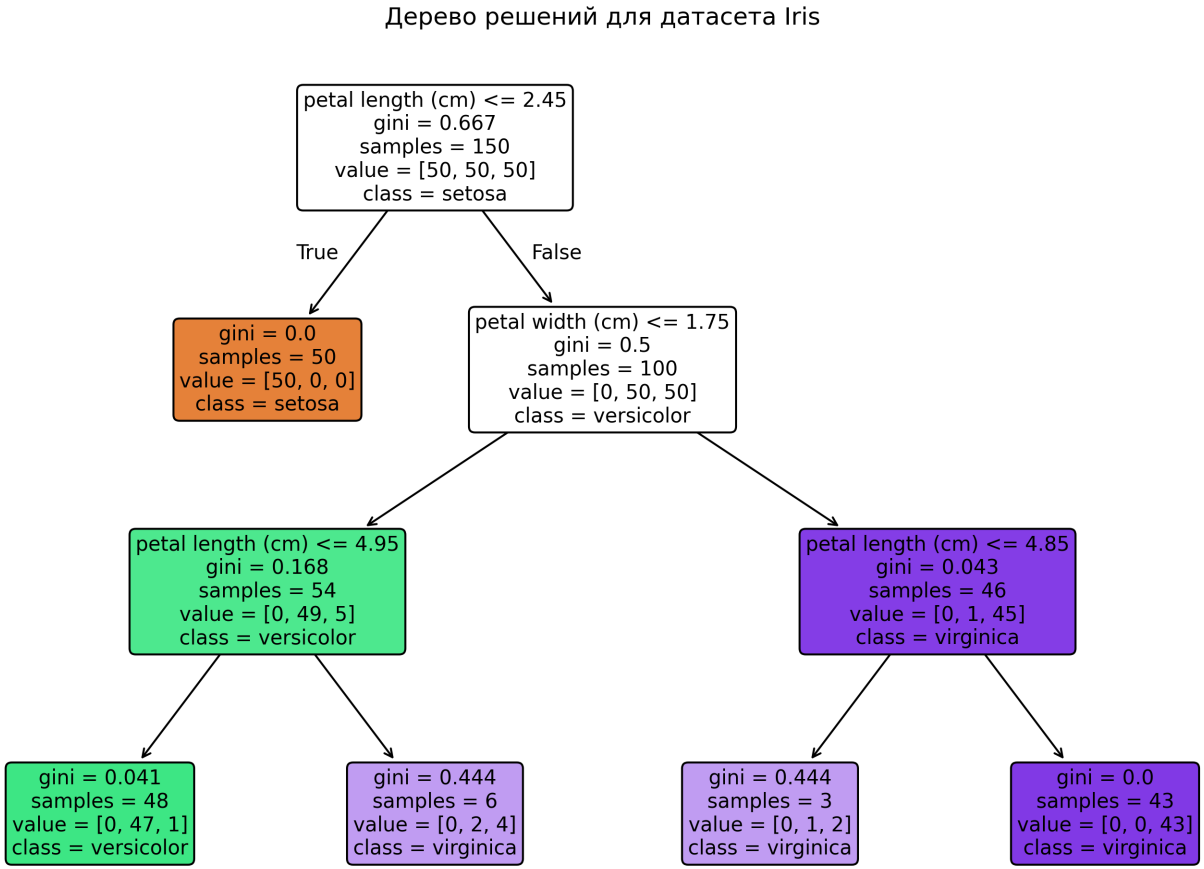
Московский физико-технический институт

Москва, 12 февраля 2025 г.

План лекции

- Определение решающего дерева
- Жадный алгоритм построения решающего дерева
- Критерий разбиения
- Случайный лес
- Особенности случайного леса

Пример решающего дерева



- **узел** - это предикат (правило)
- если предикат верен, переходим в **правого потомка**, если нет — **в левого**
- **листья** - это классы

Решающие дерево - это **бинарное дерево**:

- каждой внутренней вершине v соответствует предикат $B_v : X \to \{0, 1\}$
- каждому листу v соответствует прогноз $c_v \in Y$
- $B(x, j, t) = [x_j \leq t]$

Предсказание решающего дерева

- Для объекта x начинаем движение из корня.
- В вершине v: движемся **вправо**, если $B(x, j, t) = 1$, иначе **влево**.
- Движемся до тех пор, пока не достигнем листа (осуществим прогноз).

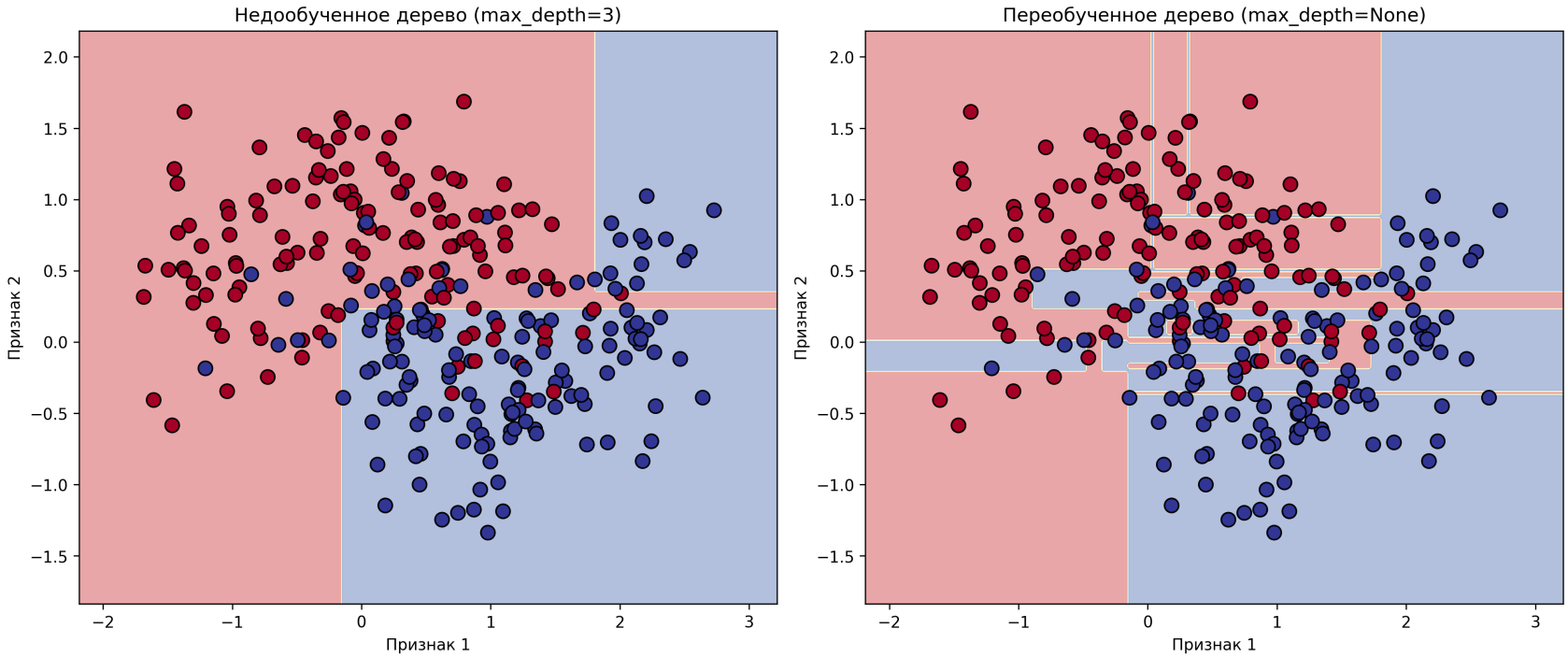
Особенности решающего дерева

Решающие дерево - это **кусочно-постоянная** функция, градиент которой 0. Градиентные методы поиска оптимально решения неприменимы.

Дерево решений **не сможет экстраполировать зависимости** за границы области значений обучающей выборки.

Дерево решений **легко переобучается**.

Переобучение решающего дерева



Для каждого узла дерева нужно выбрать один из множества возможных признаков для разбиения и определить соответствующее пороговое значение. Если в датасете есть d признаков, а каждый признак может быть разделён на множество способов (в зависимости от количества уникальных значений), то число возможных вариантов **растёт экспоненциально**.

Поиск оптимального решающего дерева относится к классу **NP-трудных задач**. Это означает, что найти точное решение за полиномиальное время практически невозможно для **больших датасетов**.

Вместо этого используются эвристические методы, такие как **жадный алгоритм**, который строит дерево последовательно, выбирая **лучшее разбиение** на каждом шаге, но **не гарантирует глобальный оптимум**.

Выбор признака для разбиения

Для **каждого признака** j ($j = 1, 2, \ldots, d$) и **каждого возможного порогового значения** t_j, вычисляется качество разбиения и находится предикат $B(x, j, t)$ **максимизирующий данное** качество.

Разбиение данных

После выбора оптимального признака j^* и порога t_{j^*}, данные **разделяются** на две подвыборки: X_l, X_r.

Повторение процесса

Процесс **повторяется рекурсивно** для каждой подвыборки, пока не будут выполнены условия остановки.

Определение прогнозов

Когда создание новых узлов становится невозможным, то этот узел **становится листом**, и ему присваивается прогноз.

- X_m - выборка до разбиения
- X_l - выборка в левом узле после разбиения
- X_r - выборка в правом узле после разбиения

Критерий (качество) разбиения

$$Branch(X_m, j, t) = |X_m| \cdot H(X_m) - |X_l| \cdot H(X_l) - |X_r| \cdot H(X_r)$$

$$Branch(X_m, j, t) \to max$$

H - обычно называют **информативностью**.

Критерий остановки

- Узел содержит объекты только одного класса.
- Размер узла меньше минимального допустимого значения.
- Глубина дерева достигла заданного предела.

Задача классификации (энтропия)

Энтропия измеряет степень "хаоса"или неопределённости в распределении классов в узле. Для узла S с K классами энтропия вычисляется как:

$$H(S) = -\sum_{k=1}^{K} p_k \log_2(p_k),$$

где:

- p_k — доля объектов класса k в узле S
- $\log_2(p_k)$ — логарифм вероятности класса k

Задача классификации (индекс Джини)

Индекс Джини измеряет вероятность того, что случайно выбранный объект будет неправильно классифицирован, если его класс назначается случайным образом. Для узла S он вычисляется как:

$$G(S) = 1 - \sum_{k=1}^{K} p_k^2,$$

где p_k — доля объектов класса k в узле S.

Стоит понимать, что G и H это одно и то же в формулах выше.

Задача регрессии (MSE)

Дисперсия (Var) измеряет разброс значений целевой переменной y вокруг её среднего значения:

$$\text{Var}(S) = \frac{1}{|S|} \sum_{i \in S} (y_i - \bar{y}_S)^2,$$

где:

- S – множество объектов в узле,
- $|S|$ – количество объектов в узле,
- y_i – значение целевой переменной для i-го объекта,
- $\bar{y}_S = \frac{1}{|S|} \sum_{i \in S} y_i$ – среднее значение целевой переменной в узле.

Стоит понимать, что *Var* и *H* это одно и то же в формулах выше.

Пример

У нас есть данные о том будет ли человек гулять (y) в зависимости от температуры (X):

Температура X	будет гулять y
10	No
15	No
20	Yes
25	Yes
30	Yes
35	Yes

Мы хотим построить решающее дерево, используя энтропийный критерий для выбора оптимального разбиения.

В исходном узле у нас есть 6 объектов, из которых:

- 3 объекта принадлежат классу Yes,
- 3 объекта принадлежат классу No.

Энтропия $H(S)$ вычисляется по формуле:

$$H(S) = -\sum_{k=1}^{K} p_k \log_2(p_k),$$

где:

- $K = 2$ (классы Yes и No),
- $p_{\text{Yes}} = \frac{3}{6} = 0.5$,
- $p_{\text{No}} = \frac{3}{6} = 0.5$.

Подставляем значения:

$$H(S) = -(0.5 \cdot \log_2(0.5) + 0.5 \cdot \log_2(0.5)).$$

Вычисляем логарифмы: - $\log_2(0.5) = -1$.
Подставляем:

$$H(S) = -(0.5 \cdot (-1) + 0.5 \cdot (-1)) = -(0.5 + 0.5) = 1.$$

Энтропия исходного узла: $H(S) = 1$.

Выберем пороговое значение $t = 22.5$ °C для разбиения данных. После разбиения получаем два подузла:

Левый подузел (S_L):

- Температура ≤ 22.5 °C,
- Классы: No, No, Yes.

Правый подузел (S_R):

- Температура > 22.5 °C,
- Классы: Yes, Yes, No.

$$H(S_L) = -(0.333 \cdot \log_2(0.333) + 0.667 \cdot \log_2(0.667)).$$

$$H(S_L) = -(0.333 \cdot (-1.585) + 0.667 \cdot (-0.585)) = -(0.528 + 0.390) = 0.918.$$

$$H(S_R) = -(0.667 \cdot \log_2(0.667) + 0.333 \cdot \log_2(0.333)).$$

$$H(S_R) = -(0.667 \cdot (-0.585) + 0.333 \cdot (-1.585)) = -(0.390 + 0.528) = 0.918.$$

Рассчитываем критерий

- $|X_m| = 6$
- $|X_r| = 3$
- $|X_l| = 3$

$$Branch(X_m, j, t) = |X_m| \cdot H(X_m) - |X_l| \cdot H(X_l) - |X_r| \cdot H(X_r)$$

$$Branch(X_m, j, t) = 6 \cdot 1 - 3 \cdot 0.918 - 3 \cdot 0.918 = 0.492$$

Задача 1

Какое минимальное и максимальное значение $Branch(X_m, j, t)$ возможно?

Задача 2

Как решающие дерево работает с категориальными признаками?

Задача 3

Как решающие дерево работает с пропусками?

Задача 4

В обучающей выборке n объектов. Какая максимальная глубина построенного дерева может быть возможна?

Случайный лес



Пусть дана обучающая выборка (X, y), $|X| = n$

Генерация подвыборки

Выберем из выборки (X, y) равновероятно n объектов. Получим новую выборку (X^i, y^i).

Генерация подвыборки

Обучим на выборке (X^i, y^i) дерево таким образом, что в процессе обучения каждого дерева в каждой вершине случайно выбираются $d_i < d$ признаков, где d — полное число признаков.

классификация

$$y(x) = \arg\max_k \sum_{i=1}^{B} \mathbf{1}(b_i(x) = k),$$

регрессия

$$y(x) = \frac{1}{B} \sum_{b=1}^{B} b_i(x).$$

Задача 3

- Какое число признаков d_i необходимы для обучения?
- Какая оптимальная глубина дерева?
- Какое число уникальных объектов в X_i?
- Какое число деревьев является оптимальным?

Задача 3

- Какое число признаков d_i необходимы для обучения?
  Рекомендация для классификации: $d_i = \sqrt{d}$
  Рекомендация для регрессии: $d_i = \frac{d}{3}$
- Какая оптимальная глубина дерева?
- Какое число уникальных объектов в X_i?
- Какое число деревьев является оптимальным?

Задача 3

- Какое число признаков d_i необходимы для обучения?
  Рекомендация для классификации: $d_i = \sqrt{d}$
  Рекомендация для регрессии: $d_i = \frac{d}{3}$
- Какая оптимальная глубина дерева?
  Чем **больше**, тем **лучше**. В разумных пределах.
- Какое число уникальных объектов в X_i?
  Уникальное число объектов при больших n стремится к $1 - \frac{1}{e}$
- Какое число деревьев является оптимальным?

Задача 3

- Какое число признаков d_i необходимы для обучения?
  Рекомендация для классификации: $d_i = \sqrt{d}$
  Рекомендация для регрессии: $d_i = \frac{d}{3}$
- Какая оптимальная глубина дерева?
  Чем **больше**, тем **лучше**. В разумных пределах.
- Какое число уникальных объектов в X_i?
  Уникальное число объектов при больших n стремится к $1 - \frac{1}{e}$
- Какое число деревьев является оптимальным?
  Такое число, при котором эффективность перестаёт **значительно** расти.

Сравнение решающих деревьев и случайного леса

Характеристика	Решающее дерево	Случайный лес
Простота модели	Простая	Сложная
Обобщающая способность	Низкая (часто переобучается)	Высокая (меньше риск переобучения)
Стабильность	Нестабильная (чувствительна к данным)	Стабильная (благодаря ансамблевому подходу)
Качество предсказаний	Хорошее на простых задачах	Лучшее на сложных задачах
Вычислительная сложность	Низкая (быстрое обучение)	Высокая (требует больше времени)
Обработка шума	Чувствительна к шуму	Меньше чувствительна к шуму
Интерпретируемость	Высокая (легко понять правила)	Низкая (сложнее анализировать множество деревьев)
Параметры тюнинга	Минимум параметров	Больше параметров
Применение	Подходит для простых задач	Подходит для сложных задач

Литература

- Полезные ссылки
- Полезные ссылки

Вопросы?