Решающие деревья. Random Forest.Композиция методов. Бустинг

Морозов Никита
Романов Даниил

Санкт-Петербург
2023 г.

Решающие деревья

Решающее дерево — бинарное дерево, в котором

- Каждой внутренней вершине v задан предикат $\beta_v : X \to \{0, 1\}$,
- Каждой листовой вершине v приписан прогноз $c_v \in Y$ или $(c_v \in \mathbb{R}^k, \sum_{i=1}^{k} c_{vi} = 1)$.

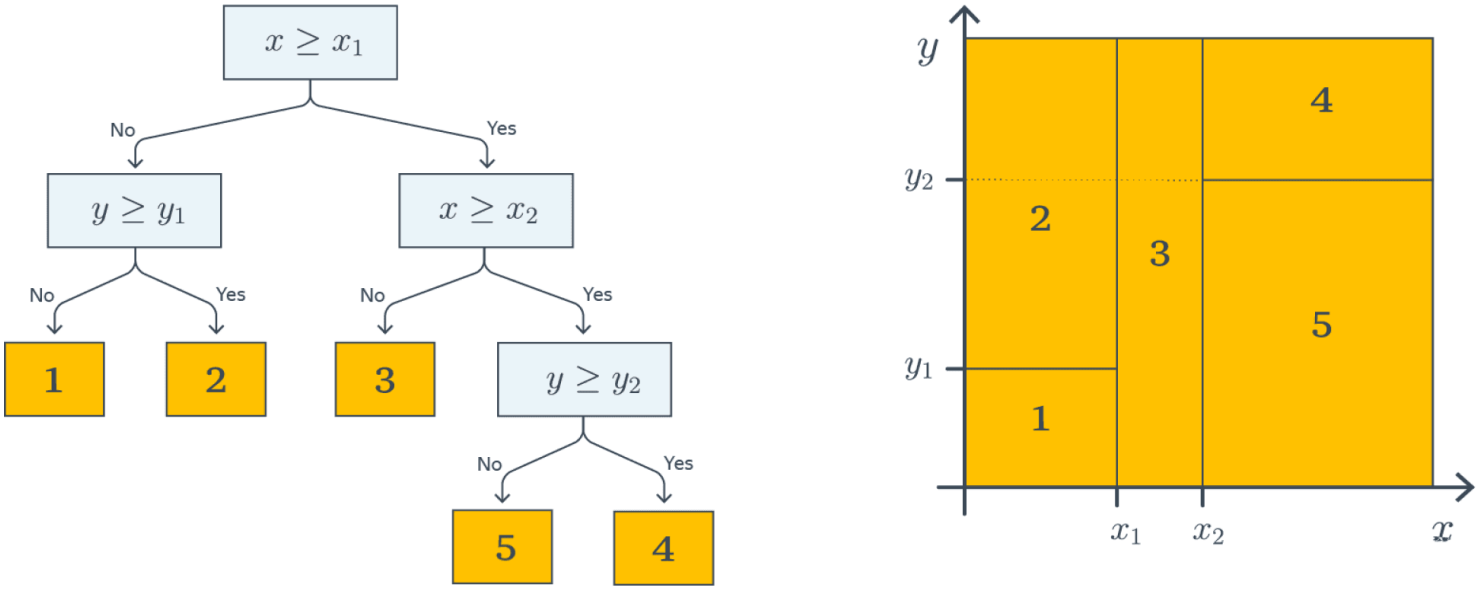

Рис.: Бинарное решающее дерево для классификации на 5 классов и решающие поверхности порождаемых деревом

Задачи решающих деревьев

Решающие деревья можно применять как для задач регрессии, так и для задач классификации.

Есть обучающая выборка $D = \{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}$,где x_i—входные данные, y_i — известные ответы(метки классов, значение регрессии).

- $y_i \in \{1, \dots, K\} \Rightarrow$ задача классификации.
- $y_i \in \mathbb{R} \Rightarrow$ задача регрессии.

Общий подход

Для каждого объекта выборки x начинаем из корня. Если $\beta_v(x) = 1$ двигаемся вправо, если $\beta_v(x) = 0$ то влево. Двигаемся пока не дойдем до листа и получим прогноз c_v для объекта x.

Предикат β_v может иметь любую структуру, но обычно просто сравниваем с порогом $t \in \mathbb{R}$ по какому-то j-му признаку: $\beta_v(x, j, t) = [x_j \leq t]$

Решающие деревья в задаче регрессии

Общая идея состоит в разбиении нашего пространства предикатов(множества возможных значений для $X_1 \dots X_p$) на J различных и непересекающихся областей $R_1, R_2, \dots, R_J$. Для каждого наблюдения попавшего в R_j, мы делаем такой же прогноз, который является средним значением ответа для обучающих наблюдений в R_j

Цель: найти такие прямоугольники $R_1, \dots, R_J$ в которых минимизируется RSS:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2,$$

где $\hat{y}_{R_j}$ среднее ответов на обучающих наблюдения в j-м прямоугольнике

Решающие деревья в задаче регрессии

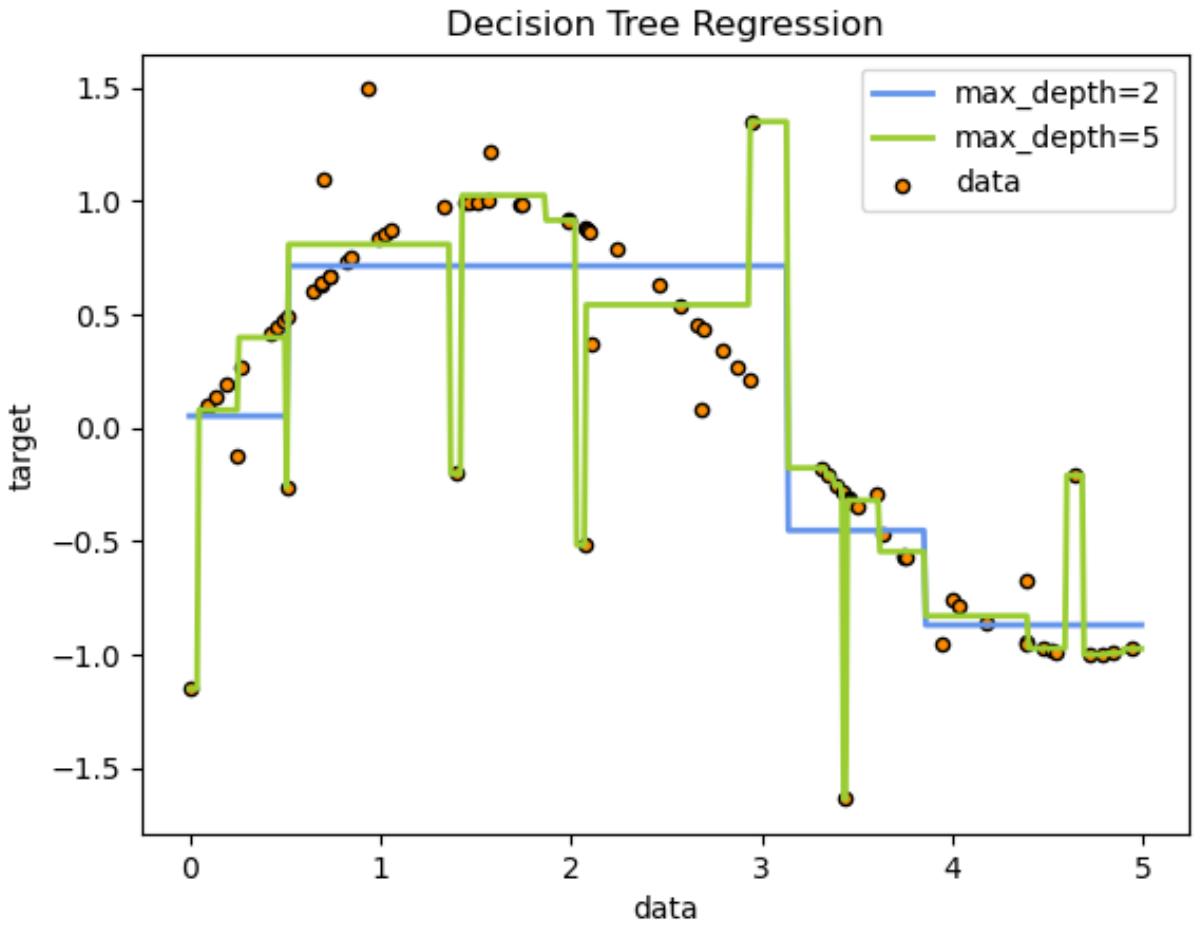

Рис.: Решающее дерево для регрессии

Решающие деревья в задаче классификации

Естественная альтернатива RSS для классификации — коэффициент ошибок классификаций.

$$E = 1 - \max_k(\hat{p}_{mk})$$

где $\hat{p}_{mk}$ — доля объектов обучающей выборки класса k попавших в m-область.

На практике чаще используют две других метрики:

- $G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$ — индекс Джини,
- $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$ — коэффициент перекрёстной энтропии,

Предсказание для объекта x:

$$f(x) = \operatorname*{argmax}_{k \in Y} \hat{p}_{jk}.$$

Решающие деревья в задаче классификации

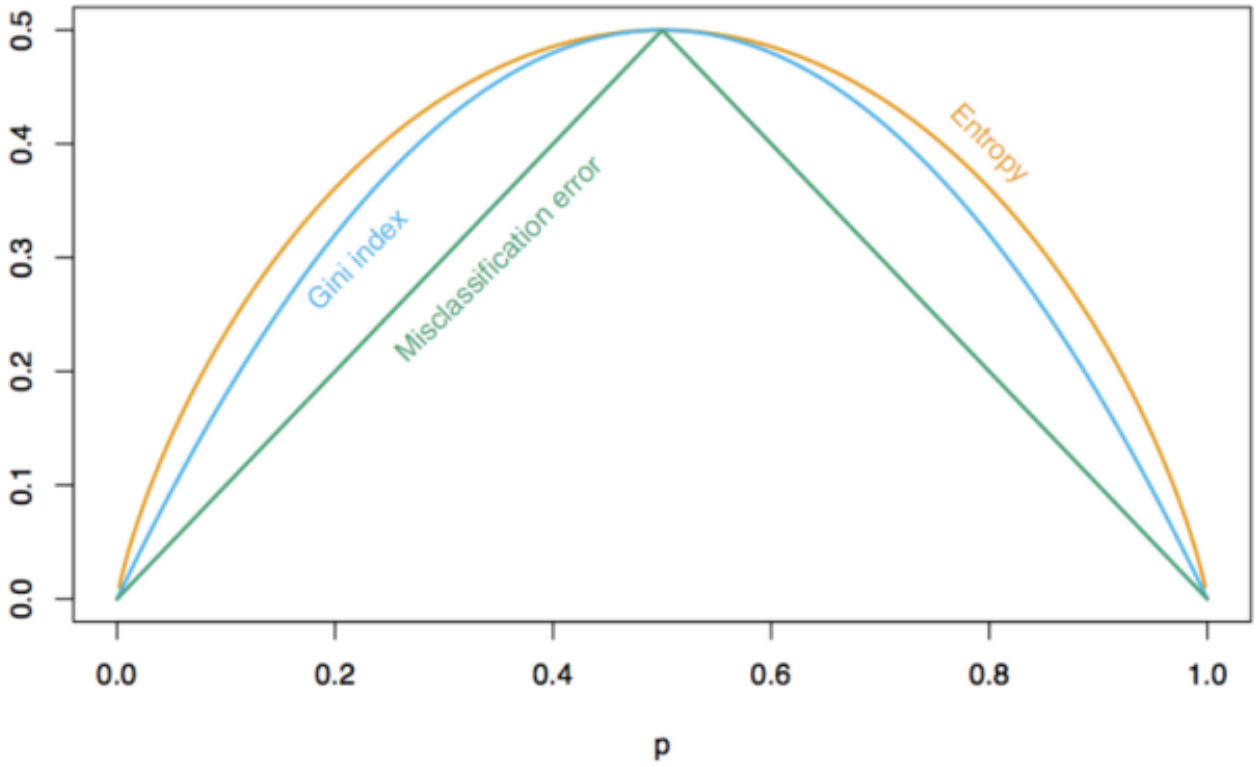

Рис.: Информационные индексы для двухклассовой классовой классификации, как функция от пропорции p для класса 2.

Решающие деревья в задаче классификации

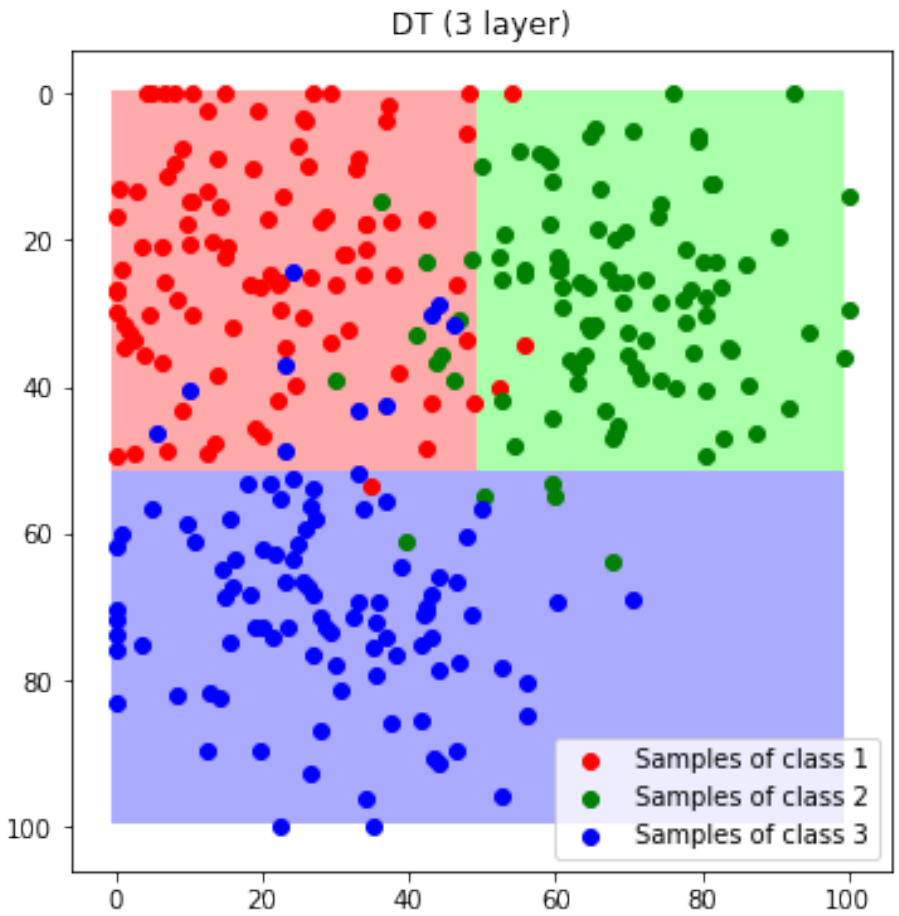

Рис.: Использование решающих деревьев в задачах классификации

Решающие деревья в задаче классификации

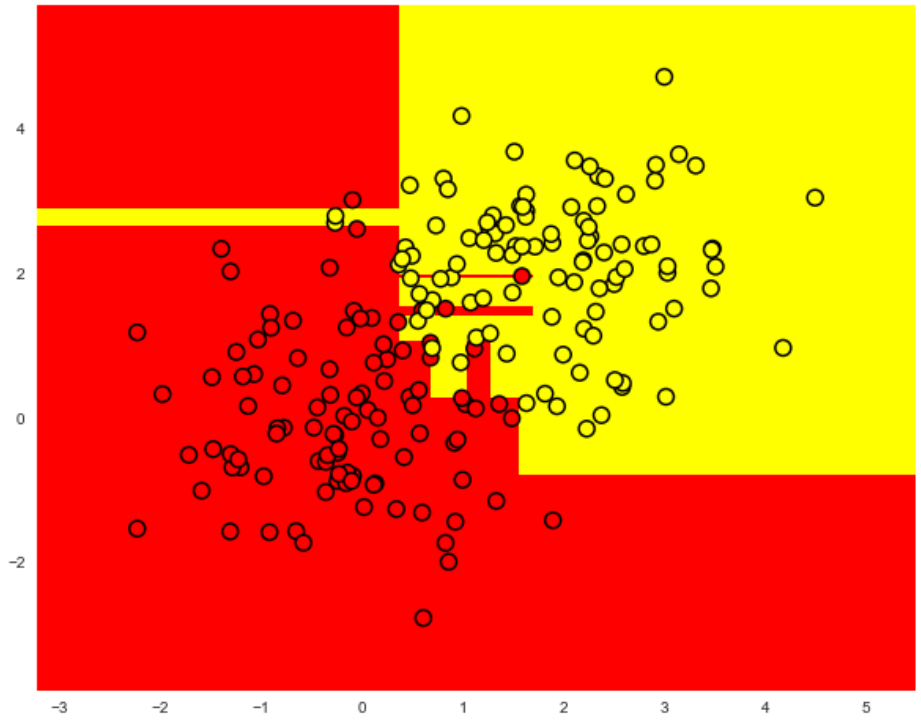

Рис.: Переобученное решающее дерево для классификации

Обучение деревьев(CART)

1. Выбираем признак j и порог s так, чтобы разбиение $X^{(n)}$ на $R_1(j,s) = \{x \in X^{(n)} | X_j < s\}$ и $R_2(j,s) = \{x \in X^{(n)} | X_j \geq s\}$ решало задачу:

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \to \min_{j,s},$$

где $\hat{y}_{R_l} = \frac{1}{|R_l|} \sum_{i:x_i \in R_l(j,s)} y_i, \quad l = 1, 2.$

2. Разбиваем выборку на области R_1 и R_2, образуя две дочерние вершины.
3. Повторяем процедуру в пределах каждой получаемой области, пока не выполнится критерий остановки.

На выходе получаем дерево, в каждом из листов которого содержится по крайней мере 1 объект исходной выборки X^n.

Обучение деревьев(IDE3)

1. $\mathbf{X}$ — обучающая выборка, $\mathbf{y} \in \{1, \cdots, k\}$.
2. Если все $\mathbf{x}_i$ имеют класс k, ставим метку 1 в корень и выходим из цикла.
3. Если ни один $\mathbf{x}_i$ не имеет класс k, ставим метку 0 в корень и выходим из цикла.
4. Предикат $R(\mathbf{x}_i) := \{\mathbf{x}_i | X_j \leqslant s_j\}$ для которого информационная выгода наибольшая.
5. Разбиваем $\mathbf{X}$ на $\mathbf{X}_0$ и $\mathbf{X}_1$ по предикату R

$$\mathbf{X}_0 := \{\mathbf{x}_i \in \mathbf{X} : R(\mathbf{x}_i) = 0\},$$

$$\mathbf{X}_1 := \{\mathbf{x}_i \in \mathbf{X} : R(\mathbf{x}_i) = 1\}.$$

6. Если $\mathbf{X}_0 = \varnothing$ или $\mathbf{X}_1 = \varnothing$, создаем новый лист v, k_v — класс, в котором находится большинство элементов $\mathbf{x}_i$.
7. Иначе создаем внутреннюю вершину v:
   1. $R_v = R$;
   2. L_v;
   3. R_v.

Сравнение алгоритмов

1. **Тип задач: ID3:** Классификация. **CART:** Классификация и регрессия.
2. **Критерий разбиения: ID3:** Энтропия. **CART:** Индекс Джини (классификация) и среднеквадратичная ошибка (регрессия).
3. **Тип разбиения: ID3:** Многоразовое. **CART:** Бинарное.
4. **Устойчивость к выбросам: ID3:** Чувствителен. **CART:** Более устойчив.
5. **Типы признаков: ID3:** Категориальные. **CART:** Категориальные и числовые.
6. **Критерии останова: ID3:** Глубина, полное разбиение. **CART:** Глубина, минимум объектов в листе.
7. **Сложность модели: ID3:** Более сложные. **CART:** Более простые (бинарная структура).

Критерии остановки

- Ограничение максимальной глубины дерева.
- Ограничение минимального числа объектов в листе .
- Ограничение максимального количества листьев в дереве.
- Остановка в случае, если изменение метрики меньше порога.
- Остановка в случае, если все объекты относятся к 1 классу

Оба метода основаны на жадных подходах(решение лишь оптимально локальное).

Стрижка деревьев (pruning tree)

- **Проблема:** переобучение — небольшое смещение, но большая дисперсия.
- **Решение:** пожертвовать смещением, но получить меньшую дисперсию
- Выращиваем дерево только до тех пор, пока уменьшение RSS из-за разбиения превышает некоторый(высокий) порог.
- Однако таким образом можно пропустить хорошее разбиение, остановившись слишком рано. Поэтому можно выращивать большие деревья T_0, а затем обрезать его, для получения поддерева.

Cost complexity pruning

- Получим большое дерево T_0 и обрежем его в узле t, получив поддерево $T^t \subset T_0$.
- Рассмотрим последовательность деревьев проиндексированных положительным параметром α. Каждому α соответствует поддерево $T \subset T_0$, минимизирующее критерий

$$Q_\alpha(T) = Q(T) + \alpha|l(T)|,$$

где $Q(T)$ — training error, $\alpha \geq 0$, $|l(T)|$ — число листьев в поддереве T.

- Выберем α с помощью кросс-валидации и возьмём соответствующее поддерево.

Сравнение деревьев с линейными моделями

Из определения, решающее дерево разбивает все пространство признаков на некоторое количество непересекающихся подмножеств $\{R_1, \ldots, R_n\}$, и в каждом подмножестве выдает константный прогноз c_j.
Модель регрессионного дерева:

$$f(X) = \sum_{j=1}^{n} c_j [X \in R_j].$$

По сути, она является линейной моделью над признаками

$$([X \in R_j])_{j=1}^{n}.$$

Сравнение деревьев с линейными моделями

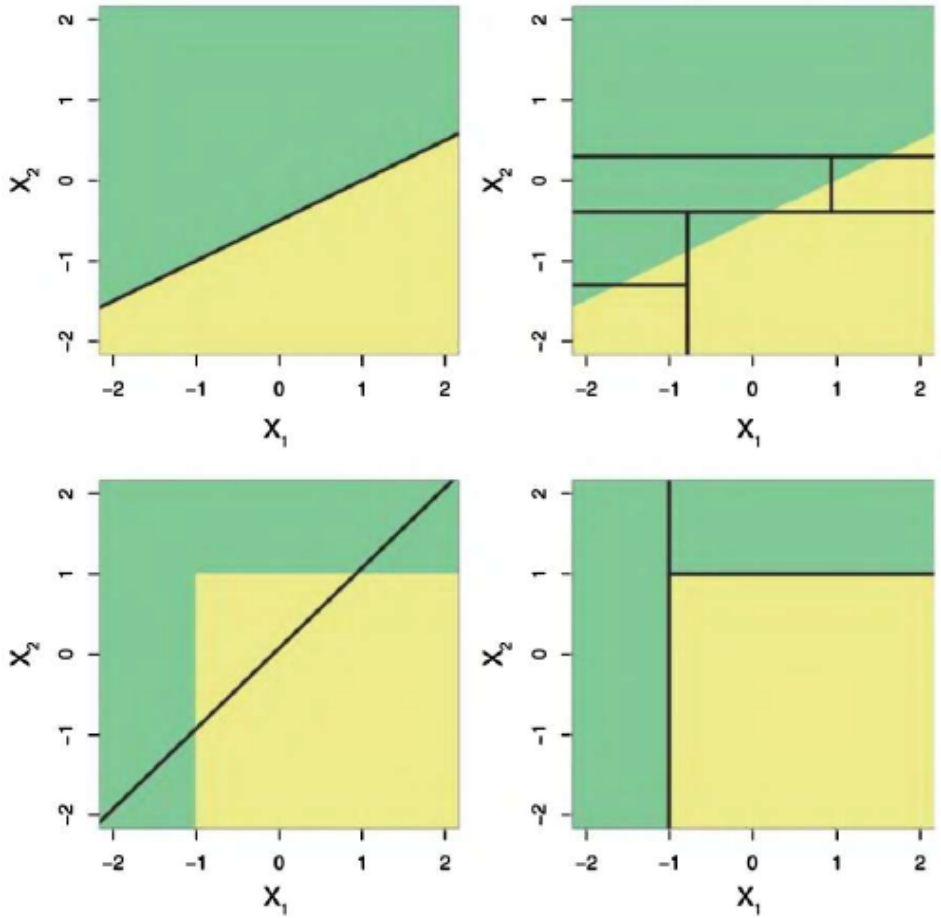

Рис.: Примеры решений задач классификации с линейной (верхний ряд) и нелинейной (нижний ряд) зависимостью. В левой части решение с помощью линейной модели, в правой — с помощью решающего дерева.

Вероятностная постановка задачи

Генеральная постановка задачи:

Предполагаем, что η и ξ функционально зависимы:

$$\eta = \varphi(\xi) + \varepsilon,$$

φ — неизвестная функция.

$\eta \in \mathbb{R}$ — случайная величина, зависимая переменная.

$\xi \in \mathbb{R}^p$ — случайный вектор, признаки.

$\varepsilon \in \mathbb{R}$ — случайная величина, ошибка.

Выборочная постановка

$$y_i = \varphi(\mathbf{x}_i) + \varepsilon_i,$$

φ — неизвестная функция.

y_i — реализация случайной величины η, зависимая переменная.

$\mathbf{x}_i$ — реализация случайного вектора ξ, признаки.

$\varepsilon_i \in \mathbb{R}$ — реализация случайной величины ε, ошибка.

Преимущества и недостатки решающих деревьев

Преимущества:

- Простота интерпретации
- Пригодность и для задач регрессии, и для задач классификации
- Возможность работы с пропусками в данных
- Возможность работы с категориальными значениями

Недостатки:

- Основан на «жадном» алгоритме (решение является лишь локально оптимальным)
- Метод явялется неустойчивым и склонным к переобучению

Bootstrap

- Дано $\mathbf{X} \in \mathbb{R}^{n \times p}$ — набор данных, $\mathbf{Y} \in \mathbb{R}^n$ — зависимые переменные, $X = (x_i, y_i)$.
- Возьмем l объектов с возвращениями — X_1
- Повторим N раз — $X_1, \ldots, X_N$
- Обучим по каждой выборке модель линейной регрессии и получим базовые алгоритмы $b_1(x), \ldots, b_N(x)$
- Предположим, что существует модель $y(x) = \sum \beta_i x_i + \varepsilon_i$ и $p(x)$ — распределение $\mathbf{X}$.
- Ошибка регрессии: $\varepsilon_j(x) = b_j(x) - y(x), \quad j = 1, ..., N.$
- $\mathbb{E}_x \varepsilon_j^2(x) = \mathbb{E}_x \left(b_j(x) - y(x)\right)^2$

Среднеквадратичная ошибка

Средняя ошибка построенных функций регрессии:

$$E_1 = \frac{1}{N}\sum_{j=1}^{N}\mathbb{E}_x\epsilon_j^2(x)$$

Пусть

- $\mathbb{E}_x\epsilon_j(x) = 0$ и $\mathbb{E}_x\epsilon_i(x)\epsilon_j(x) = 0,\ i \neq j$
- $a(x) = \frac{1}{N}\sum_{j=1}^{N} b_j(x)$

Тогда

$$E_N = \mathbb{E}_x\left(\frac{1}{N}\sum_{j=1}^{N} b_j(x) - y(x)\right)^2 = \mathbb{E}_x\left(\frac{1}{N}\sum_{j=1}^{N}\epsilon_j(x)\right)^2 =$$

$$= \frac{1}{N^2}\mathbb{E}_x\left(\sum_{j=1}^{N}\epsilon_j^2(x) + \sum_{i\neq j}\epsilon_i(x)\epsilon_j(x)\right) = \frac{1}{N}E_1$$

Bias-Variance decomposition

Пусть задана выборка $X = (x_i, y_i)_{i=1}^{l}$ с ответами $y_i \in \mathbb{R}$ и $\exists p(x, y)$

Рассмотрим $L(y, a) = (y - a(x))^2$ — функция потерь,

и $R(a) = \mathbb{E}_{x,y}\left[(y - a(x))^2\right] \int_{\mathbb{X}} \int_{\mathbb{Y}} p(x, y)(y - a(x))^2 dxdy$ — ее среднеквадратичный риск.

Ошибка метода обучения

Метод обучения

$$\mu : (\mathbb{X} \times \mathbb{Y})^l \to \mathbf{A}$$

$$L(\mu) = \mathbb{E}_X \left[\mathbb{E}_{x,y} \left[(y - \mu(X)(x)))^2 \right] \right] \tag{1}$$

Среднеквадратичный риск на фиксированной выборке X

$$\mathbb{E}_{x,y} \left[(y - \mu(X))^2 \right] = \mathbb{E}_{x,y} \left[(y - \mathbb{E}[y|x])^2 \right] + \mathbb{E}_{x,y} \left[(\mathbb{E}[y|x] - \mu(X))^2 \right]$$

Подставим это в формулу (1).

Ошибка метода обучения

$$L(\mu) = \mathbb{E}_X \left[\underbrace{\mathbb{E}_{x,y}\left[(y - \mathbb{E}[y|x])^2\right]}_{\text{не зависит от X}} + \mathbb{E}_{x,y}\left[(\mathbb{E}[y|x] - \mu(X))^2\right] \right] =$$
$$= \mathbb{E}_{x,y}\left[(y - \mathbb{E}[y|x])^2\right] + \mathbb{E}_{x,y}\left[\mathbb{E}_X\left[(\mathbb{E}[y|x] - \mu(X))^2\right]\right] \tag{2}$$

Преобразовываем второе слагаемое:

$$\mathbb{E}_{x,y}\left[\mathbb{E}_X\left[(\mathbb{E}[y|x] - \mu(X))^2\right]\right] =$$
$$= \mathbb{E}_{x,y}\left[\mathbb{E}_X\left[(\mathbb{E}[y|x] - \mathbb{E}_X[\mu(X)] + \mathbb{E}_X[\mu(X)] - \mu(X))^2\right]\right] =$$
$$= \mathbb{E}_{x,y}\left[\mathbb{E}_X\left[\underbrace{(\mathbb{E}[y|x] - \mathbb{E}_X\mu(X))^2}_{\text{не зависит от X}}\right]\right] + \mathbb{E}_{x,y}\left[\mathbb{E}_X\left[(\mathbb{E}_X\mu(X) - \mu(X))^2\right]\right.$$
$$+ 2\mathbb{E}_{x,y}\left[\mathbb{E}_X\left[(\mathbb{E}[y|x] - \mathbb{E}_X[\mu(X)])(\mathbb{E}_X[\mu(X)] - \mu(X))\right]\right] \tag{3}$$

Bias-Variance decomposition

Подставим (3) в (2).

$$L(\mu) = \underbrace{\mathbb{E}_{x,y}\left[(y - \mathbb{E}[y|x]^2)\right]}_{\text{шум}} + \tag{4}$$

$$+ \underbrace{\mathbb{E}_x\left[\mathbb{E}_X[\mu(X)] - \mathbb{E}[y|x]\right]}_{\text{смещение}} + \underbrace{\mathbb{E}_x\left[\mathbb{E}_X\left[(\mu(X) - \mathbb{E}_X[\mu(X)])^2\right]\right]}_{\text{разброс}} \tag{5}$$

Bagging

Цель: уменьшение дисперсии модели с сохранением низкого смещения.

Идея: пусть $\xi_1, \ldots, \xi_n$ — н.о.р.с.в., $\mathrm{D}\,\xi_i = \sigma^2$, тогда $\mathrm{D}\,\bar{\xi} = \frac{\sigma^2}{n}$.

Реализация:

$X^n = (x_i, y_i)_{i=1}^n$ — обучающая выборка.

- B бутстреп-выборок (с возвращением) $X_b^{*n}, \quad b = 1, \ldots, B$,
- B решающих деревьев $\{T_b\}_{b=1}^B$,
- находим оценку:
  - в задаче регрессии $\hat{f}_{bag}(x) = \frac{1}{B} \sum\limits_{b=1}^{B} T_b(x)$,
  - в задаче классификации с K классами: записываем класс предсказанный каждым деревом, итоговое предсказание — самый часто встречающийся класс среди предсказаний.

Оценка ошибки out-of-bag

- Дерево, обученное по бутстреп-выборке, использует в среднем $2/3$ наблюдений.
- Оставшуюся $1/3$ выборки называют *out-of-bag* наблюдениями.
- Те деревья, у которых i-ое наблюдение out-of-bag, могут использоваться для предсказания на i. Таким образом можно получить примерно $B/3$ предсказаний.
- В результате получаем способ тестирования bagged модели прямо на обучающей выборке.