

Лекция 9

Решающие деревья. Ансамбли моделей

Габдуллин Р.А., Макаренко В.А.

МГУ им. М.В. Ломоносова

15 марта 2021

Решающее дерево

Пусть X – выборка объектов, которые описываются p признаками:

$$X = (X_1, \dots, X_\ell), \quad X_k \in \mathbb{R}^p.$$

Решающее дерево – бинарное дерево, в котором

- каждой внутренней вершине v приписана функция (предикат) $\beta_v : \mathbb{R}^p \rightarrow \{0, 1\}$.
- каждой листовой вершине приписан прогноз.

Получение предсказания для объекта $x = (x_1, \dots, x_p)$:

- Стартуем обход дерева с корня v_0 .
- Если $\beta_{v_0}(x_1, \dots, x_p) = 1$, то продолжаем обход левого поддерева, иначе – правого.
- Процесс обхода продолжается до тех пор, пока не будет достигнута листовая вершина.
- Прогноз – значение в листовой вершине.

Решающее дерево

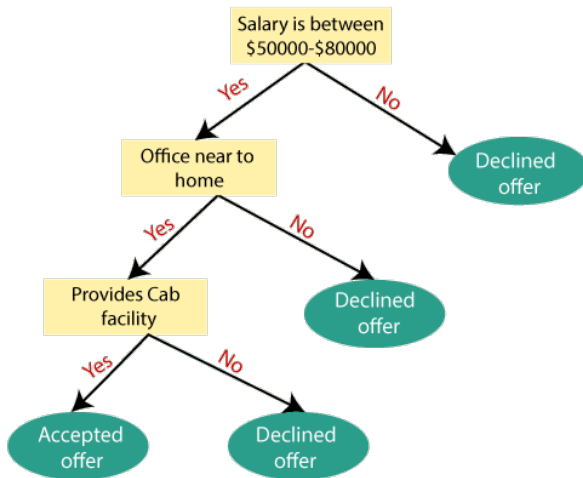


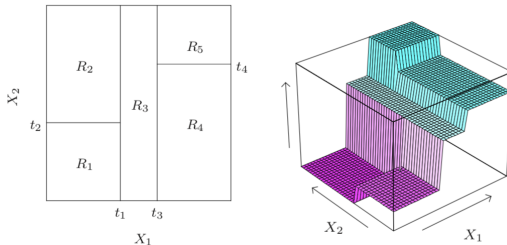
Рис.: Источник: mygreatlearning.com

Вид решающего правила

- Дерево разбивает признаковое пространство на конечное число областей $\{R_m, 1 \leq m \leq M\}$.
- В области R_m выдается константное предсказание c_m :

$$a(x) = \sum_{m=1}^M c_m [x \in R_m].$$

- Решающее правило – линейная модель с M признаками $[x \in R_m]$ и коэффициентами c_m .



Построение решающего дерева

Вид предикатов:

$$\beta_{j,s}(x) = [x_j \leq s], \quad 1 \leq j \leq p, \quad s \in \mathbb{R}.$$

Жадный алгоритм построения дерева:

- Находим наилучшее разбиение выборки на две части

$$R_1(j, s) = \{x \in X \mid x_j \leq s\}, \quad R_2(j, s) = \{x \in X \mid x_j > s\}$$

с точки зрения заданного функционала качества разбиения $Q(X, j, s)$.

- Создадим корневую вершину с предикатом $[x_j \leq s]$, где j, s – лучшие параметры, найденные на предыдущем шаге.
- Объекты с помощью предиката разобьются на две части.
- Рекурсивно строим левое и правое поддеревья.
- Выходим из рекурсии, если выборка пуста или выполнено условие останова.

Функционал качества разбиения

Пусть

- R_v – объекты, попавшие в вершину, разбиваемую на данном шаге.
- $R_l(j, s)$ – объекты, попадающие в левое поддерево при разбиении.
- $R_r(j, s)$ – объекты, попадающие в правое поддерево при разбиении.

Функционал качества разбиения:

$$Q(R_v, j, s) = H(R_v) - \frac{|R_l|}{|R_v|} \cdot H(R_l) - \frac{|R_r|}{|R_v|} \cdot H(R_r) \rightarrow \max$$

где $H(R)$ – критерий информативности (impurity criterion):

$$H(R) = \min_c \frac{1}{|R|} \sum_{(x_i, y_i) \in R} L(y_i, c),$$

$L(y, c)$ – заданная функция потерь.

Критерии информативности в задаче регрессии

- Выборочная дисперсия:

$$H(R) = \min_c \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - c)^2 = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - \bar{y})^2.$$

- Среднее абсолютное отклонение от медианы:

$$H(R) = \min_c \frac{1}{|R|} \sum_{(x_i, y_i) \in R} |y_i - c| = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} |y_i - \text{median}(y)|.$$

Критерии информативности в задаче классификации

- Доля неверно классифицированных объектов:

$$H(R) = \min_c \frac{1}{|R|} \sum_{(x_i, y_i) \in R} [y_i \neq c] = \frac{1}{R} \sum_{(x_i, y_i) \in R} [y_i \neq y^*] = 1 - p_{y^*},$$

где $p_k = \frac{1}{|R|} \sum_{(x_i, y_i)} [y_i = k]$, $y^* = \operatorname{argmax} p_k$.

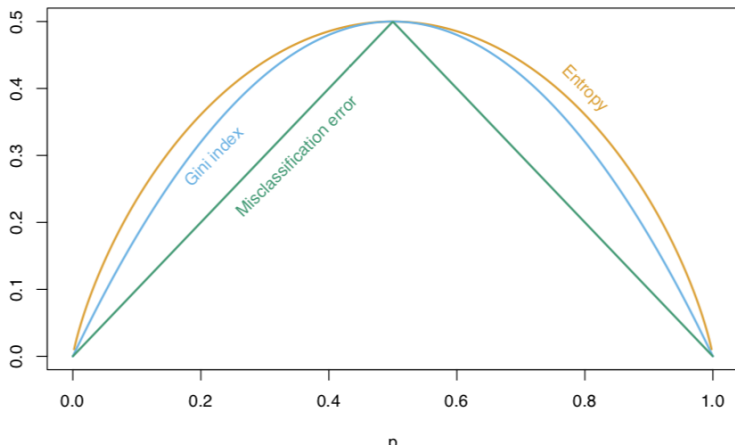
- Критерий Джини:

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

- Энтропийный критерий:

$$H(R) = \min_{\sum c_k=1} \left(-\frac{1}{|R|} \sum_{(x_i, y_i) \in R} \sum_{k=1}^K [y_i = k] \log c_k \right) = -\sum_{k=1}^K p_k \log p_k.$$

Критерии информативности в задаче классификации



- Ограничение максимальной глубины дерева.
- Ограничение максимального количества листьев в дереве.
- Ограничение максимального числа объектов в дереве.
- Требование, что функционал качества должен увеличиться не некоторое количество процентов.

Стрижка дерева (pruning)

Cost-complexity pruning:

- Строим дерево T_0 максимальной глубины.
- Выбираем поддерево T , минимизирующее следующий функционал:

$$R_\alpha(T) = R(T) + \alpha L(T),$$

где $R(T)$ – эмпирический риск, $L(T)$ – количество листьев в дереве, $\alpha \geq 0$.

Обработка пропущенных значений

- Удаление объекта с пропущенными значениями.
- Заполнение пропущенных значений (мода, среднее значение, ...).
- Создание отдельного значения для пропуска.
- Суррогатные предикаты:
 - Рассматривая признак для разбиения при построении очередной вершины, используем только те наблюдения, у которых значение признака известно.
 - Для данной вершины вместе с выбранным предикатом формируем список суррогатных предикатов – предикаты по другим признакам, которые дают похожие разбиения с выбранным предикатом.
 - Во время предсказания используем суррогатные предикаты, если значение признака пропущено.

Работа с категориальными признаками

Проблема:

- Признак с q значениями можно разбить на две группы $2^{q-1} - 1$ способами (экспоненциальный рост числа вариантов).

Решение для бинарной классификации (критерий Джини или энтропийный критерий):

- Упорядиваем значения признака по доле положительного класса среди объектов с таким значением признака.
- Используем обычные пороговые предикаты.

Решение для регрессии (функция потерь MSE):

- Упорядочиваем значение признака по среднему значению целевой переменной среди объектов с таким значением признака.
- Используем обычные пороговые предикаты.

Методы построения деревьев

- ID3
 - Использует энтропийный критерий.
 - Только категориальные признаки.
 - Строится до тех пор, пока в каждом листе не окажутся объекты одного класса, либо пока разбиение дает уменьшение критерия.
- C4.5
 - Использует нормированный энтропийный критерий.
 - Поддержка вещественных признаков.
 - Критерий останова – ограничение числа объектов в листе.
 - Обработка пропущенных значений осуществляется с помощью метода, который игнорирует объекты с пропущенными значениями при вычислении критерия ветвления, а затем переносит такие объекты в оба поддерева с определенными весами.
- CART
 - Критерий Джини.
 - Стрижка cost-complexity pruning.
 - Обработка пропусков с помощью суррогатных предикторов.

Достоинства и недостатки деревьев решений

Достоинства:

- Интерпретируемость
- Минимальная предобработка признаков.
- Гибкость.

Недостатки:

- Склонность к переобучению.
- Негладкое решение.
- Сложность построения модели в случае разделяющей полосы, не параллельной осям координат.

Ансамбли моделей

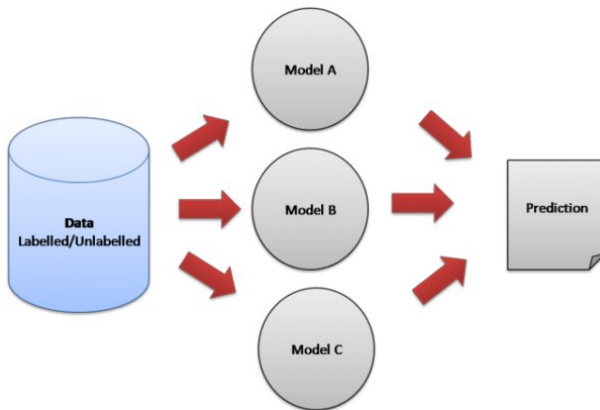


Рис.: Источник: [kdnuggets.com](https://www.kdnuggets.com)

Деревья сильно переобучаются поодиночке. Но можно обучить несколько моделей и усреднить результат.

Композиция алгоритмов. Бэггинг (Bootstrap aggregation)

Пусть $(x_1, y_1), \dots, (x_\ell, y_\ell)$ – обучающая выборка.

- Построим M выборок длины ℓ , выбирая случайно ℓ объектов с возвращением из исходной выборки (bootstrap-подвыборка).
- На каждой выборке обучим модель.
- Прогноз композиции – среднее значение прогнозов построенных моделей (самый частый ответ в задаче классификации).

Случайный лес (Random forest)

Строим композицию из N решающих деревьев. Каждое дерево строится следующим образом:

- Генерируем bootstrap-подвыборку.
- При построении очередной вершины дерева, рассматриваем не все признаки, а m случайных.
- Строим дерево до выполнения критерия останова (обычно строят очень глубокие деревья).