

Лекция 6

Решающие деревья

Чепарухин Сергей
Data Scientist@Mail.Ru

Содержание



1. Знакомимся с решающими деревьями
2. Учимся их строить
3. Разбираем задачу оттока клиентов

Деревья решений



Пример



Вы приходите в банк за кредитом, подаете анкету со всеми необходимыми документами

Сотрудник банка проверяет вашу анкету:

1. Если возраст меньше 18, то отказываем, иначе шаг 2.

2. Если стаж больше года - дать кредит, иначе - шаг 3.

3. Если зарплата меньше 30 тысяч рублей, то отказать, иначе шаг 4.

4. Если есть другие кредиты, то выдаем, иначе отказываем.

Определение бинарного решающего дерева

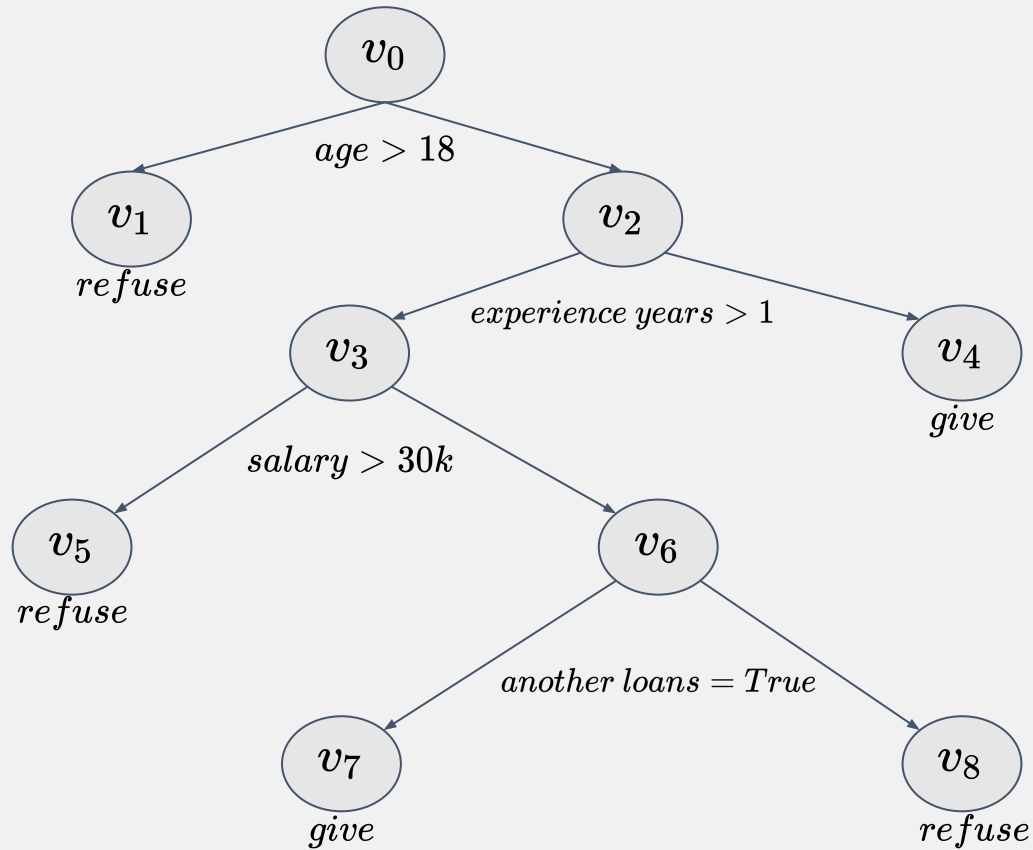


Рассмотрим бинарное дерево, в котором:

- каждой внутренней вершине v приписана функция $\beta_v : \mathbb{X} \rightarrow \{0, 1\}$;
- каждой терминальной(листовой) вершине v приписана метка класса $c_v \in Y$.

Процесс предсказания - обход дерева из вершины v_0 до терминальных вершин с последовательным вычислением соответствующих функций β_v

Дерево из примера



Варианты разделяющих функций



Что можно взять в качестве $\beta_v : \mathbb{X} \rightarrow \{0, 1\}; :$

- одномерные предикаты (пороговая функция)
- многомерные предикаты:
 - линейные $\beta_v(x) = [\langle w, x \rangle]$
 - метрические $\beta_v(x) = [\rho(x, x_0) < s]$, где точка x_0 - любая точка признакового пространства

Как строить деревья



Конкретный алгоритм построения задается:

1. видом предикатов;
2. критерием информативности $Q(X, \beta_v)$;
3. критерием останова;
4. методом обработки пропущенных значений

Критерии информативности для классификации



Определим общее число объектов, пришедших в вершину v_m как R_{v_m} . $p_{v_m k}$ - это доля класса C_k в вершине v_m .

- Ошибка классификации:

$$F_E(R_{v_m}) = 1 - \max_k p_{v_m k} \quad Q_E(R_{v_m}, \beta) = F_E(R_{v_m}) - \frac{N_{v_l}}{N_{v_m}} F_E(R_{v_l}) - \frac{N_{v_r}}{N_{v_m}} F_E(R_{v_r})$$

- Индекс Джини(Gini):

$$F_G(R_{v_m}) = 1 - \sum_k (p_{v_m k})^2 = \sum_{k' \neq k} p_{v_m k'} p_{v_m k}$$

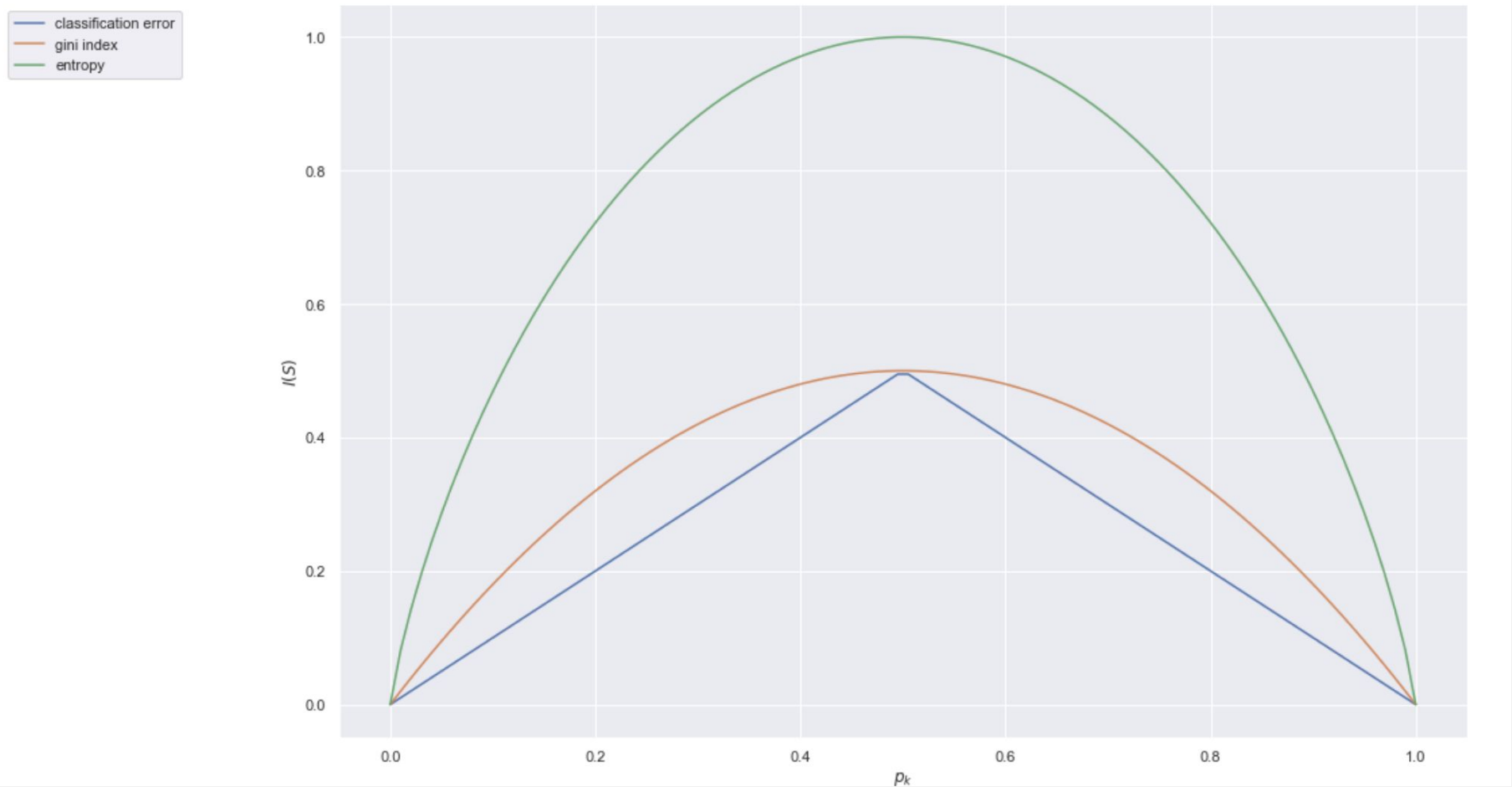
$$Q_G(R_{v_m}, \beta) = F_G(R_{v_m}) - \frac{N_{v_l}}{N_{v_m}} F_G(R_{v_l}) - \frac{N_{v_r}}{N_{v_m}} F_G(R_{v_r})$$

- Энтропийный критерий:

$$F_H(R_{v_m}) = H(p_{v_m}) = - \sum_k p_{v_m k} \log(p_{v_m k})$$

$$Q_H(R_{v_m}, \beta) = F_H(R_{v_m}) - \frac{N_{v_l}}{N_{v_m}} F_H(R_{v_l}) - \frac{N_{v_r}}{N_{v_m}} F_H(R_{v_r})$$

Как выглядят меры неопределенности



Критерии информативности для регрессии



- Дисперсия ответов:

$$F(R_{v_m}) = \frac{1}{N_{v_m}} \sum_{x_i \in R_{v_m}} \left(y_i - \text{mean}(y_i) \right)^2$$

- Среднее абсолютное отклонение от медианы:

$$F(R_{v_m}) = \frac{1}{N_{v_m}} \sum_{x_i \in R_{v_m}} \left| y_i - \text{median}(y_i) \right|$$



- Пока не закончится не разделенная выборка
- Ограничение максимальной глубины
- Ограничение минимального числа объектов в вершине
- Ограничение максимального количества терминальных вершин(листьев)
- В листе находятся объекты одного класса
- Ограничение на относительное изменение критерия информативности

Обработка пропущенных значений



1. Удалить объекты/признаки с пропусками
2. Пропуск - отдельное значение
3. Вычисление критерия информативности без учета объектов с пропусками
4. Суррогатные предикаты
5. Заполнение средними значениями/нулями

Как предсказывать



$c = \max_k p_{v_m k}$ - классификация

$p_k = p_{v_m k}$ - если необходима вероятность

$y_k = \text{mean}_{v_m}(y_i)$ - регрессия



- Разбиение на все возможные значения категориального признака
- Разбиение значений на два подмножества, подбираем подмножества по критерию информативности
- Подбираем по встречаемости:

$$\frac{1}{N_{vm}(k_{(1)})} \sum_{x_i \in R_{vm}(k_{(1)})} [y_i = +1] \leq \dots \leq \frac{1}{N_{vm}(k_{(n)})} \sum_{x_i \in R_{vm}(k_{(n)})} [y_i = +1]$$

Если ищем по Джини или энтропийному, эквивалентно разбиению на два подмножества

$$\frac{1}{N_{vm}(k_{(1)})} \sum_{x_i \in R_{vm}(k_{(1)})} y_i \leq \dots \leq \frac{1}{N_{vm}(k_{(n)})} \sum_{x_i \in R_{vm}(k_{(n)})} y_i$$

Специальные алгоритмы построения деревьев



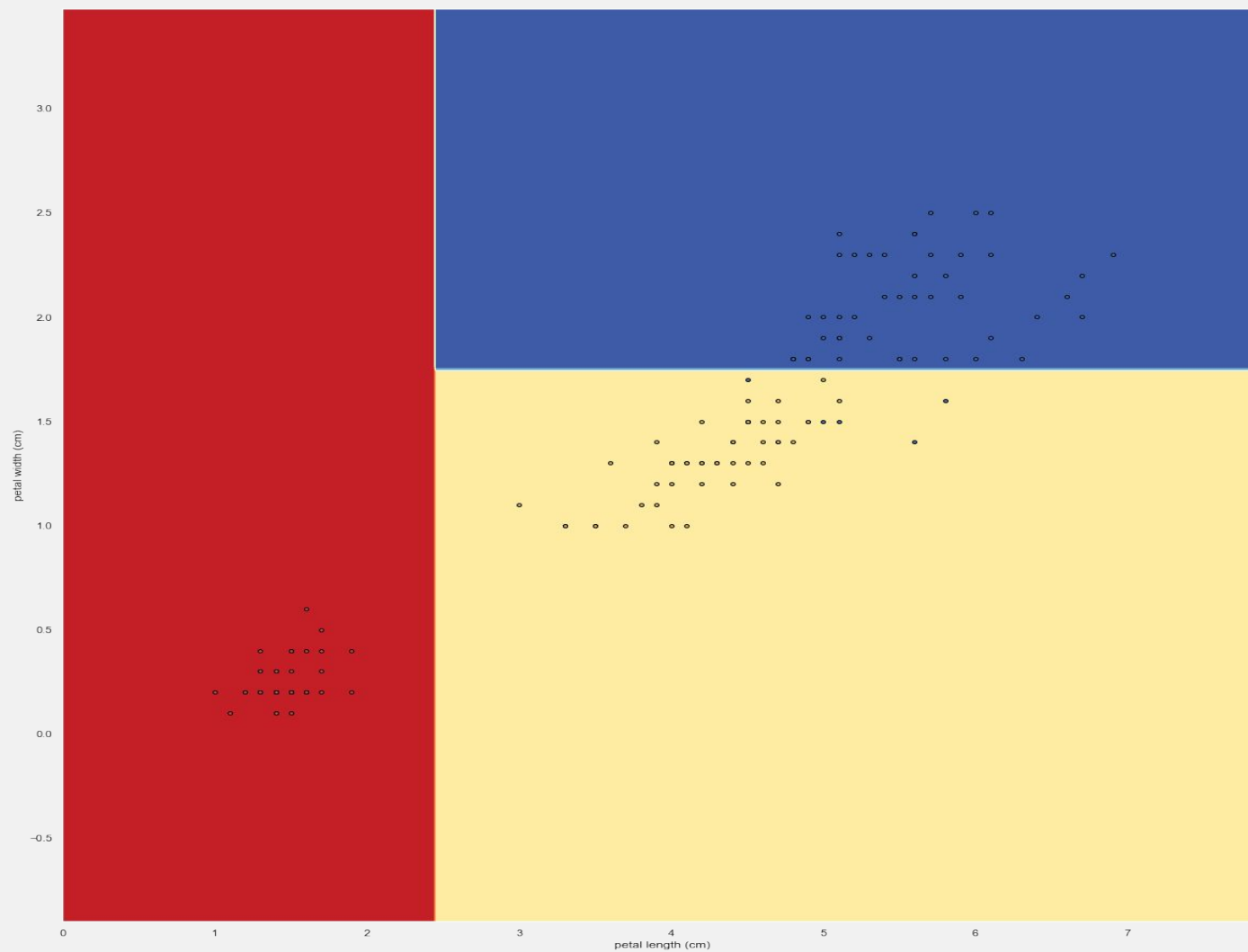
- ID 3
 - Использует энтропийный критерий
 - Только категориальные признаки
 - Количество потомков = количеству значений признака
 - Строится до тех пор пока в каждом листе не окажутся объекты одного класса или пока разбиение дает уменьшение критерия
- C 4.5
 - Использует нормированный энтропийный критерий
 - Поддержка вещественных признаков
 - Категориальные как в ID3
 - Критерия останова - ограничение на число объектов в листе
 - При пропуске значения переход по всем потомкам
- CART(реализован в scikit-learn)
 - Использует критерий Джини
 - Поддержка разных типов признаков
 - При пропусках значений строит суррогатные предикаты

Преимущества и недостатки

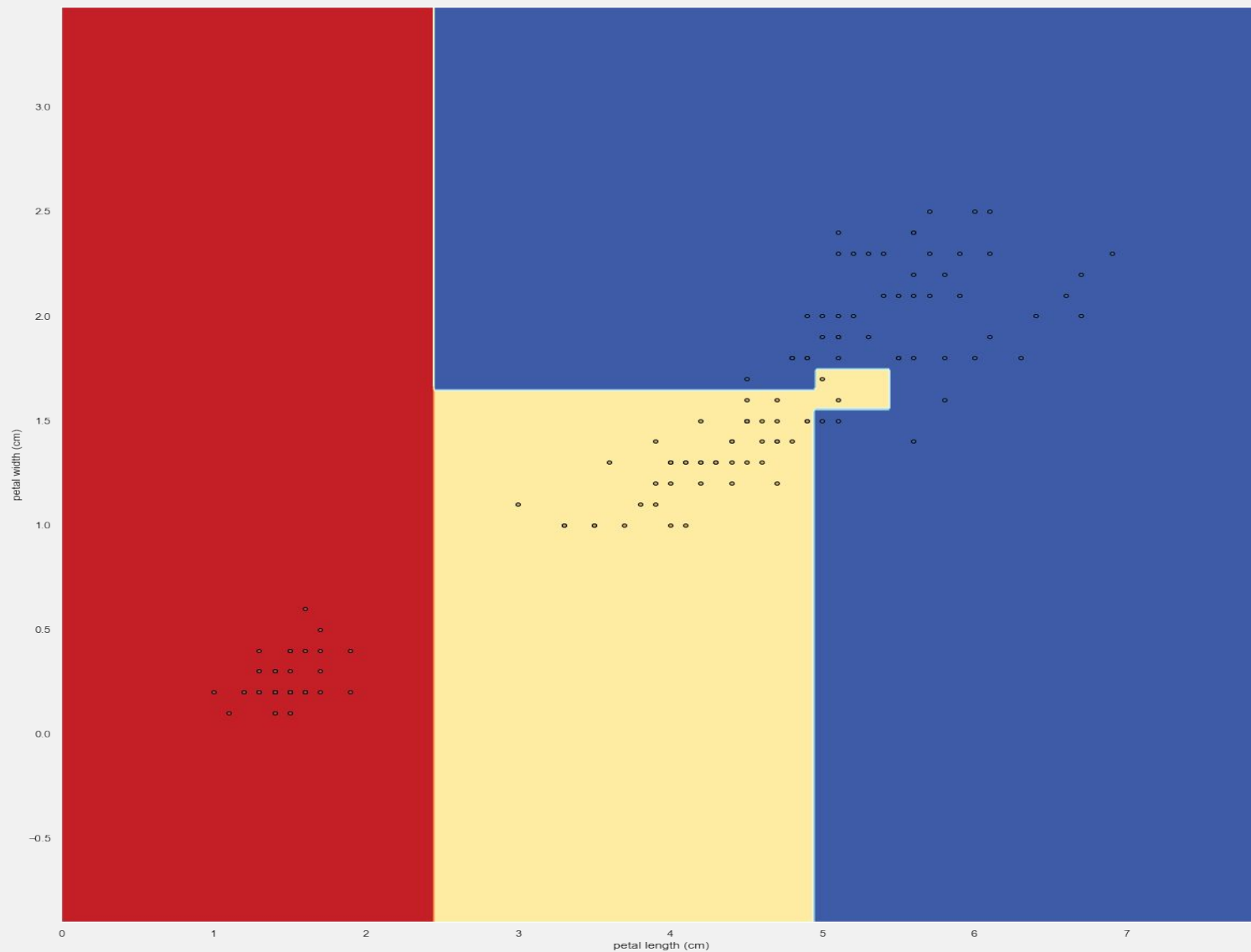


- Преимущества
 - Простота построения
 - Интерпретируемость (при небольшой глубине)
 - Требуется минимальная предобработка признаков
 - Встроенный отбор признаков
- Недостатки
 - Склонность к переобучению
 - При добавлении новых объектов надо полностью перестраивать и результат может получиться совершенно иным
 - Жадность построения
 - Сложность построения модели в случае разделяющей полосы, не параллельной осям координат

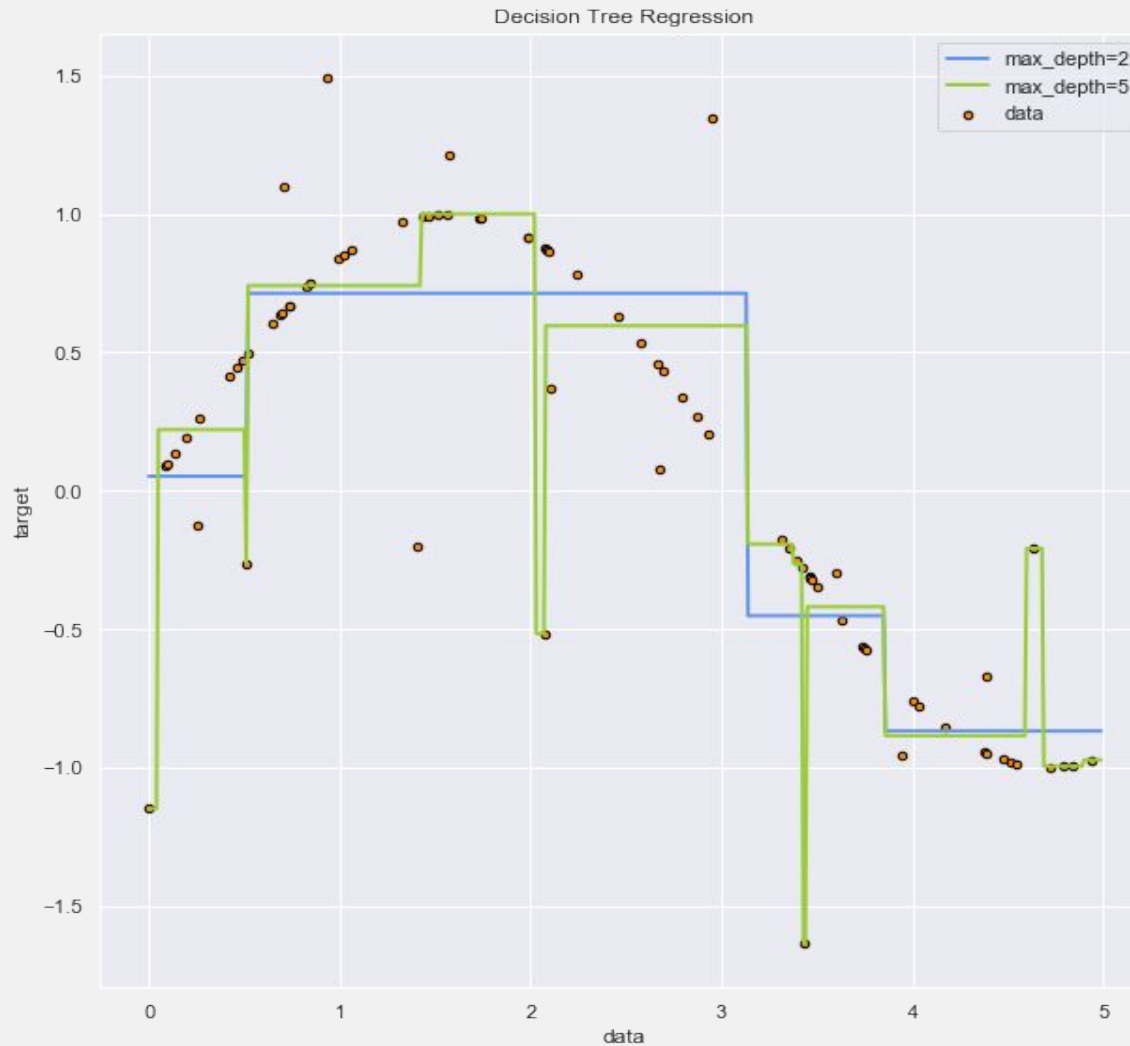
Примеры гиперплоскостей



Примеры гиперплоскостей



Пример построения деревьев для регрессии



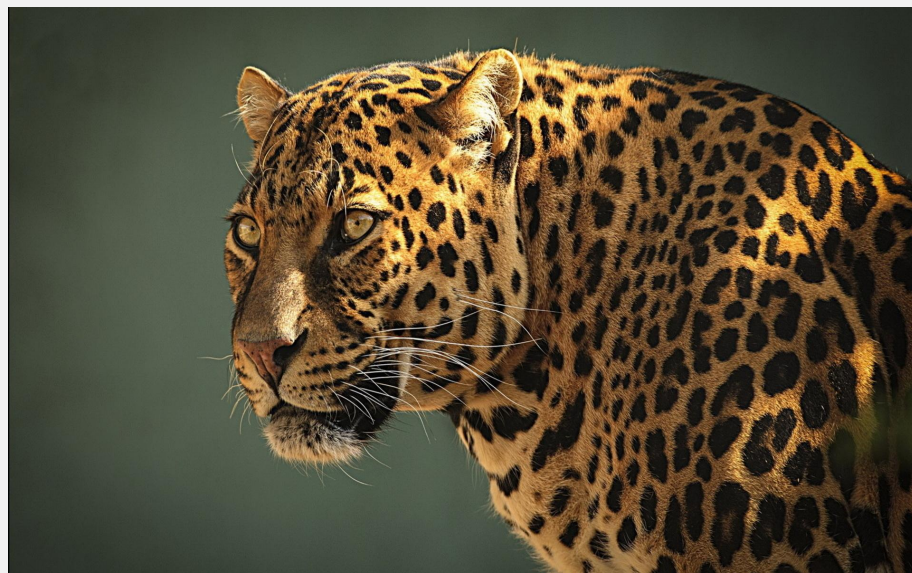
Сыграем в угадайку



Задача регрессии:

Какая высота у Эйфелевой башни?

Задача классификации: На какой картинке леопард?



Композиции деревьев. Бэггинг



Деревья переобучаются, поэтому поодиночке их использовать плохо из-за низкой обобщающей способности.

Можно обучить множество разных деревьев и усреднять их ответ!

Бэггинг(Bagging=Bootstrap Aggregating)



- Обучаем N базовых моделей на n объектах
- Каждая модель обучается на своей подвыборке из I объектов, взятых случайно с возвращением(Bootstrap)
- Ответ композиции равен среднему ответу базовых алгоритмов

Random Forest



Строим N деревьев. Каждое дерево строится следующим образом:

1. Генерируем подвыборку \hat{X}_n ;
2. В каждом узле дерева сперва выбираем m случайных признаков, и ищем оптимальное разбиение только среди них;
3. Дерево строится до тех пор, пока в каждом сплите окажется не более n_{min} объектов

Предсказание осуществляется путем усреднения результатов предсказаний каждого из построенных деревьев.

Extra Trees(Extremely Randomized Trees)



- Все также сэмплируем выборку;
- Не сэмплируем признаки;
- Для каждого признака сэмплируем порог и строим разбиение по нему, без подбора по критерию информативности.

Дополнительные материалы



- [Как работают деревья "на пальцах"](#)
- [Про визуализацию деревьев решений](#)
- [Гайд про энтропию](#)
- [Гайд про энтропию 2](#)
- *Hastie T., Tibshirani R., Friedman J.(2009). The Elements of Statistical Learning. Ch9.2*
- [Random Forests](#)



**Спасибо за
внимание!**

Вопросы?