Деревья решений

Интеллектуальный анализ данных, 20!7

Малютин Евгений Алексеевич

А зачем?

Надо:

- бывают категориальные данные
- бывают сложности с метриками
- обратимся, например, к регрессии
  - легко обучается
  - восстанавливает только простые зависимости
  - усложнение - через спрямляющие пространства (и не только)

Рассмотрим достаточно популярный алгоритм анализа данных:

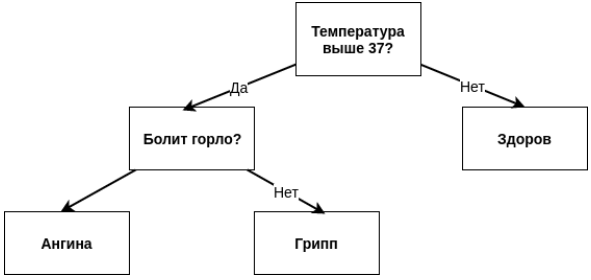

Рис.: Схема работы врача в Николаевской больнице

Решающие деревья

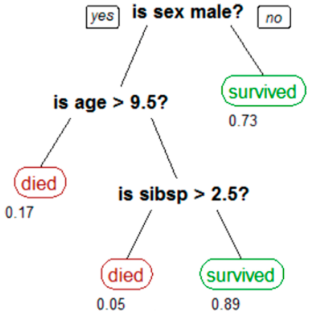

Рис.: Дети и женщины – на Титаник!

Решающие деревья

- Бинарное дерево (не обязательно)
- В каждой внутренней вершине - условие
- В каждом листе записан прогноз

Решающие деревья

Условия:

- Самый популярный вариант: $[x_j < t]$

Прогноз в листе

- Регрессия
  - вещественное число
- Классификация
  - класс
  - распределение вероятностей над классами

Решающие деревья

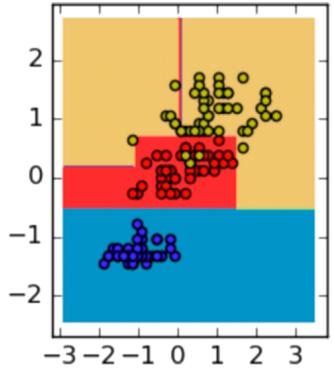

Рис.: Классификация здорового человека

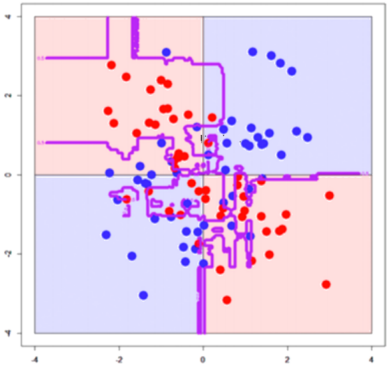

Рис.: Классификация курильщика

Решающие деревья

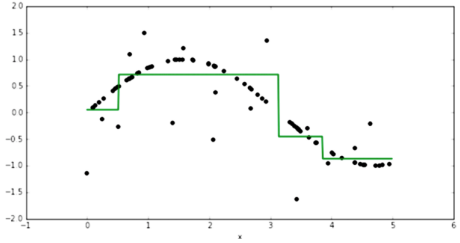

Рис.: Регрессия здорового человека

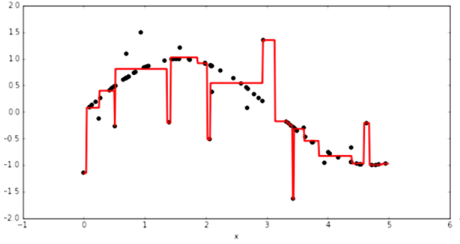

Рис.: Регрессия курильщика

Решающие деревья

Преимущества

- Интуитивность
- Легкость интерпретации результатов
- Не требует выбора входных атрибутов (сам выберет значимые)
- Точность модели сопоставима с другими методами (напр., НС (#антихайп))
- Быстрый процесс обучения
- Возможность обработки пропущенных значений
- Хорошо работают с категориальными типами данных
- Легко переобучаются

Обучение деревьев

Как мы заметили – деревья достаточно легко могут переобучаться. Как с этим жить?

Борьба с переобучением

1. дерево может достичь нулевой ошибки на любой выборке
2. борьба с переобучением: минимальное дерево с нулевой ошибкой
3. NP-полная задача

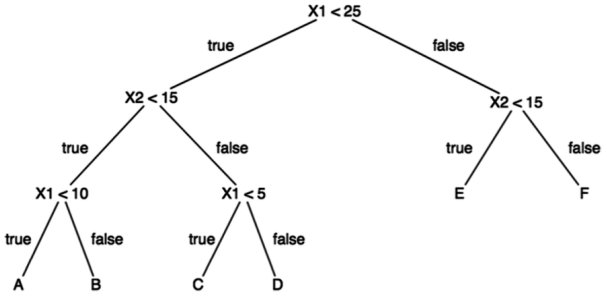

Рис.: Ещё дерево

Обучение деревьев

Поиск разбиения

- пусть в вершине m оказалась выборка X_m
- $Q(X_m, j, t)$ – критерий ошибки условия $[x^j < t]$
- ищем лучшие параметры перебором j и t:
$$Q(X_m, j, t) \to \min_{j,t}$$
- разбиваем X_m на две части
$$X_l = \{x \in X_m \mid [x^j \leq t]\}$$
$$X_r = \{x \in X_m \mid [x^j \leq t]\}$$
- смыть – повторить

Критерий останова:

- В какой момент прекращать разбиение?
- В вершине один объект?
- В вершине объекты одного класса?
- Глубина превысила порог?

Обучение деревьев

Какой прогноз выбрать?

- Регрессия:
$$a_m = \frac{1}{X_m} \sum y_i$$
- Классификация:
$$a_m = arg \max_{y \in Y} \sum_{i \in X_m} [y = y_i]$$

Got new problems

- "Жадность"алгоритма: оптимальное решение выбирается локально
- Пропуски данных

Критерии информативности

Обобщённый критерий ошибки:

$$Q(X_m,\ j,\ t)\ =\ \frac{|X_l|}{X_m}H(X_l) + \frac{|X_r|}{X_m}H(X_r)$$

Критерий информативности:

- $H(x)$
- Зависит от ответов на выборке X_m
- Чем меньше разброс ответов, тем меньше значение $H(x)$

Критерии информативности:

Регрессия

- $\bar{y}(X) = \frac{1}{|X_m|} \sum y_i$ – среднее
- $H(X) = \frac{1}{|X_m|} \sum (y_i - \bar{y}(X))^2$ – банальная дисперсия

Критерии информативности

Классификация

Тут все немного сложнее:

- Введём вспомогательную величину:
$$p_k = \frac{1}{|X|} \sum_{i \in X} [y_i = k]$$
- Критерий Джини:
$$H(X) = \sum_{k=1}^{K} p_k(1 - p_k);$$

если $p_1 = 1; p_2 = p_3 = ... = p_K = 0,$ то $H(X) = 0$

Критерии информативности

Классификация

Тут все немного сложнее:

- Введём вспомогательную величину:
$$p_k = \frac{1}{|X|} \sum_{i \in X} [y_i = k]$$
- Критерий Джини:
$$H(X) = \sum_{k=1}^{K} p_k(1 - p_k);$$
если $p_1 = 1; p_2 = p_3 = ... = p_K = 0,$ то $H(X) = 0$
- Энтропийный критерий:
$$H(X) = \sum_{k=1}^{K} p_k \ ln(p_k);$$
полагаем $0 * ln0 = 0$

Критерий останова

Вершины в листе

- все вершины в листе в одном классе

Критерий останова

Вершины в листе

- $\leq n$ вершин попало в лист
- при $n = 1$ – максимально переобученное дерево
- n – должно быть достаточно, чтобы построить надёжный прогноз
- люди говорят, что $n = 5$ хватит всем

Критерии останова

Ограничение на глубину:

- обрезаем по уровню
- простой
- грубый критерии
- нелпохо работает в композициях

Критерии останова

Ограничение на глубину:

- обрезаем по уровню
- простой
- грубый критерии
- нелпохо работает в композициях

Барбершоп aka стрижка деревьев

- строим максимально переобученное дерево
- удаляем листья по некоторому критерию
- пример: удаляем, пока улучшается ошибка на валидации
- считается, что работает лучше критерия останова

Барбершоп aka стрижка деревьев

- строим максимально переобученное дерево
- удаляем листья по некоторому критерию
- пример: удаляем, пока улучшается ошибка на валидации
- считается, что работает лучше критерия останова

Барбершопы – не нужны!

- трудоёмкая процедура
- имеет смысл только для одного дерева
- в композициях хватает только одного дерева

Категориальные признаки

$$[x_j \leq t]$$

–

только для вещественных и бинарных признаков!

Категориальные признаки

N-арные деревья

- нужно сделать разбиение вершины m
- для категориального признака x^j с n значениями $\{c_1, c_2, ..., c_n\}$
- разбиваем на n вершин
- в $i-$ вершину отправились $x^j = c_i$

Категориальные признаки

N-арные деревья

- нужно сделать разбиение вершины m
- для категориального признака x^j с n значениями $\{c_1, c_2, ..., c_n\}$
- разбиваем на n вершин
- в $i-$ вершину отправились $x^j = c_i$

Критерий информативности

- разбиваем X_m на n частей по признакам
- аналогично считаем:

$$H(X) = \sum_{i=1}^{n} \frac{X_m}{X_n} H(X_m)$$

Категориальные признаки

N-арные деревья

- нужно сделать разбиение вершины m
- для категориального признака x^j с n значениями $\{c_1, c_2, ..., c_n\}$
- разбиваем на n вершин
- в $i-$ вершину отправились $x^j = c_i$

Критерий информативности

- разбиваем X_m на n частей по признакам
- аналогично считаем:

$$H(X) = \sum_{i=1}^{n} \frac{X_m}{X_n} H(X_m)$$

Особенности

- будем часто выбирать признаки с большим n
- легко переобучиться
- подходит для очень больших выборок

Категориальные признаки

Бинарные деревья

- нужно сделать разбиение вершины m
- для категориального признака x^j с n значениями $\{c_1, c_2, ..., c_n\}$
- разобьём множество значений: $C = \{C_1 \cup C_2\}$
- разбиение $x_j \in C_1$

Категориальные признаки

Бинарные деревья

- нужно сделать разбиение вершины m
- для категориального признака x^j с n значениями $\{c_1, c_2, ..., c_n\}$
- разобьём множество значений: $C = \{C_1 \cup C_2\}$
- разбиение $x_j \in C_1$

Хитрый трюк:

- Как разбить C?
- отсортируем $c_{(1)}...c_{(n)}$
- заменим $c_{(1)}...c_{(n)}$ на $1, ...n$
- будем работать как с вещественным признаком

Категориальные признаки

Бинарные деревья

- нужно сделать разбиение вершины m
- для категориального признака x^j с n значениями $\{c_1, c_2, ..., c_n\}$
- разобьём множество значений: $C = \{C_1 \cup C_2\}$
- разбиение $x_j \in C_1$

Хитрый трюк:

- Как разбить C?
- отсортируем $c_{(1)}...c_{(n)}$
- заменим $c_{(1)}...c_{(n)}$ на $1, ...n$
- будем работать как с вещественным признаком

ЩИТО?!?!

Категориальные признаки

Бинарная классификация

$$\frac{\sum\limits_{i \in X_m} [x_j = c_{(1)}][y_i = +1]}{\sum\limits_{i \in X_m} [x_j = c_{(1)}]} \leq \ldots \leq \frac{\sum\limits_{i \in X_m} [x_j = c_{(n)}][y_i = +1]}{\sum\limits_{i \in X_m} [x_j = c_{(n)}]}$$

Категориальные признаки

Регрессия

$$\frac{\sum\limits_{i \in X_m} [x_j = c_{(1)}] y_i}{\sum\limits_{i \in X_m} [x_j = c_{(1)}]} \leq \ldots \leq \frac{\sum\limits_{i \in X_m} [x_j = c_{(n)}] y_i}{\sum\limits_{i \in X_m} [x_j = c_{(n)}]}$$

Категориальные признаки

Бинарная классификация

$$\frac{\sum\limits_{i \in X_m} [x_j = c_{(1)}][y_i = +1]}{\sum\limits_{i \in X_m} [x_j = c_{(1)}]} \leq \ldots \leq \frac{\sum\limits_{i \in X_m} [x_j = c_{(n)}][y_i = +1]}{\sum\limits_{i \in X_m} [x_j = c_{(n)}]}$$

Регрессия

$$\frac{\sum\limits_{i \in X_m} [x_j = c_{(1)}] y_i}{\sum\limits_{i \in X_m} [x_j = c_{(1)}]} \leq \ldots \leq \frac{\sum\limits_{i \in X_m} [x_j = c_{(n)}] y_i}{\sum\limits_{i \in X_m} [x_j = c_{(n)}]}$$

Резюме

- аналогично полному перебору
- выполняется для MSE, Джини и энтропийного
- но вместо экспоненты выполняется за линию

Практика

Что делать?

1. идём на кеглю и качаем bike sharing demand
   https://www.kaggle.com/c/bike-sharing-demand
2. там есть признак, который слишком высоко-коррелирует с ответом - удаляем
3. собираем дерево из sklearn
4. меряем F-score
5. рисуем на graphviz, думаем
6. тюним гипер-параметры
7. опять рисуем
8. сравниваем
9. пытаемся что-нибудь ещё улучшить. кто сможет – молодец