Деревья решений. Классификация

Цели занятия

1. Изучить принципы построения деревьев решений
2. Применять деревья решений для задач машинного обучения
3. Оценивать важность фичей с помощью деревьев решений
4. Понимать основу продвинутых алгоритмов, таких как Random Forest, XG Boost, LGBM, etc..

О чём поговорим и что сделаем

О чём поговорим и что сделаем

1. Дерево решений: что это такое?
2. Дерево решений: как его построить?
3. Построим дерево решений
4. Обсудим достоинства и недостатки деревьев решений.
5. Визуализируем принятие решений и предсказания алгоритма

Дерево решений

1

Дерево решений

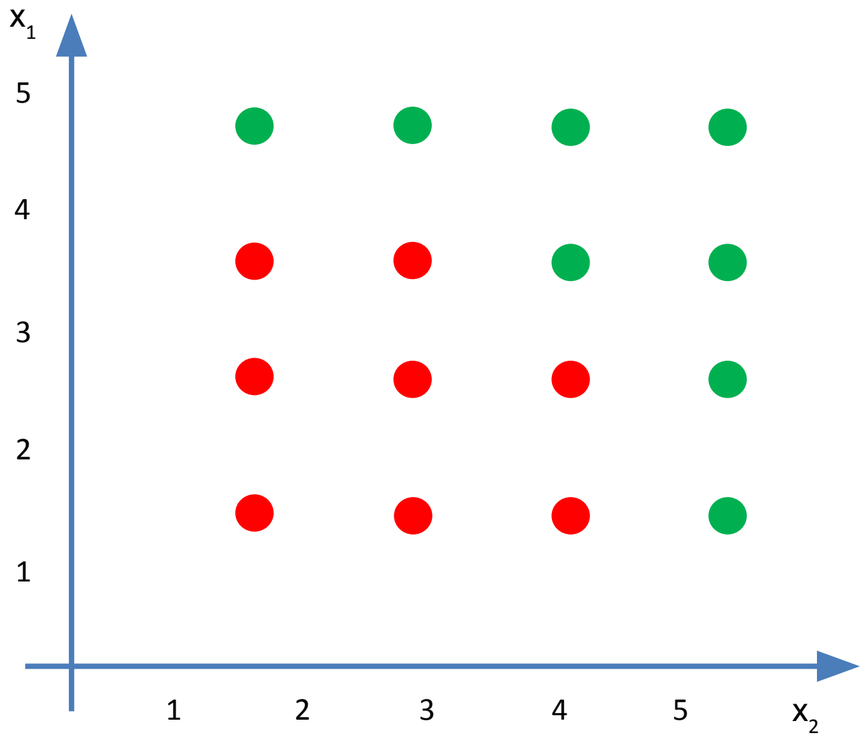
Дерево решений представляет собой древовидную структуру (древовидный граф), состоящую из логических закономерностей, на основе которых решается задачи классификации, регрессии и др.

Логическая закономерность (в задачах классификации) — легко интерпретируемое правило (rule), выделяющее из обучающей выборки достаточно много объектов какого-то одного класса и мало объектов остальных классов.

В процессе построения дерева решений эти закономерности выявляются за счет обобщения (индукции) множества отдельных наблюдений (обучающих примеров). Поэтому их называют индуктивными правилами (rule induction), а сам процесс построения дерева — индукцией деревьев решений.

Дерево решений

Как построить?

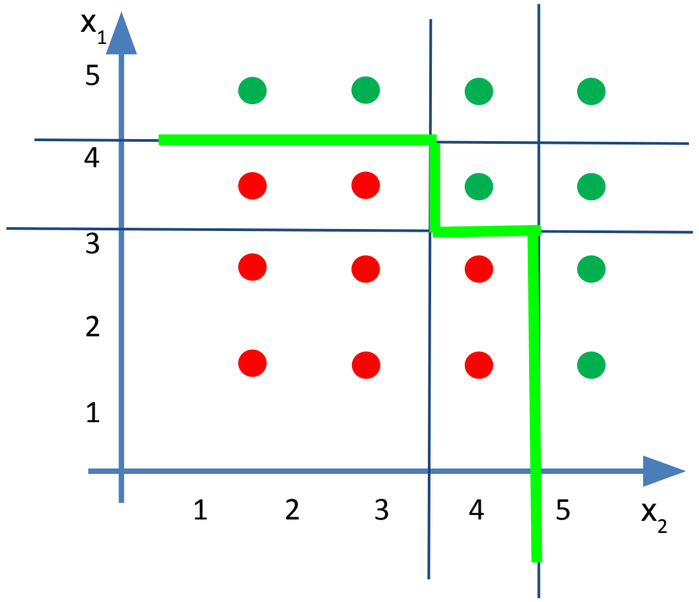


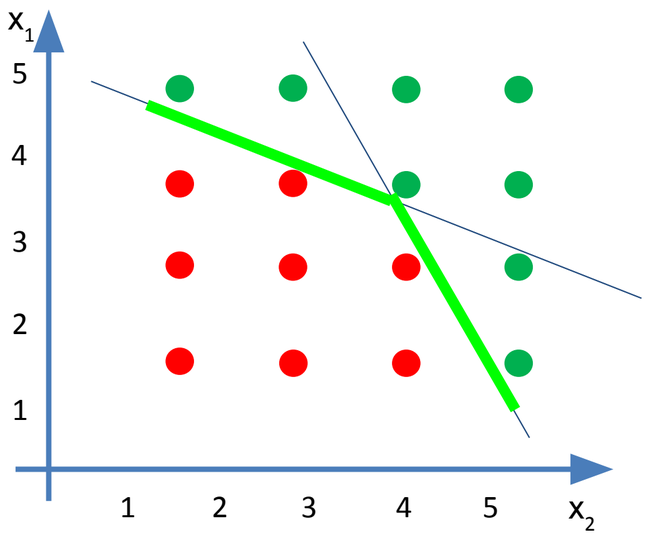
Дерево решений

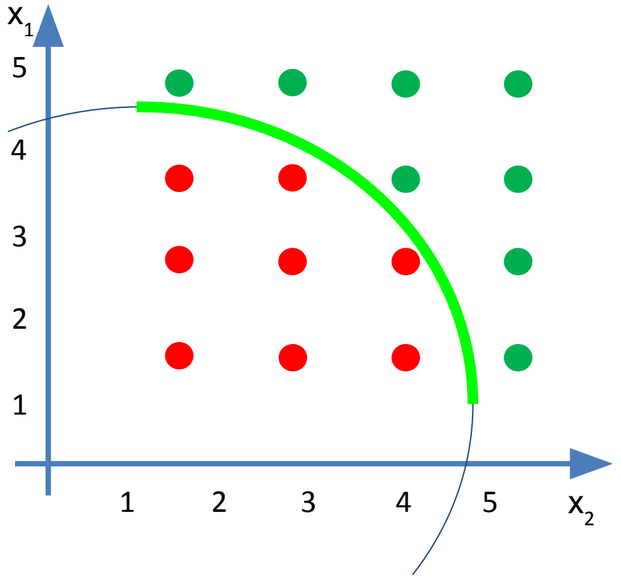
Как построить?

1. Какие виды логических закономерностей (правила) можно использовать?

- **одномерное (пороговое):** сравнивается значение одного признака
- **линейное:** сравнивается линейная комбинация признаков
- **метрическое:** расстояние до точки признакового пространства
- **синдромое**: набор одномерных правил







Дерево решений

Как построить?

2. Как выбрать хорошее правило?

- **логический**: выделяющее из обучающей выборки достаточно много объектов одного класса и мало объектов других.
- **статистический подход:** точный тест Фишера (ГГР), критерий Джини
- **информационный:** информационный критерий как некая мера неопределенности по классам в выборке (критерий Джини, энтропийный критерий)

3. Когда остановиться (переобучение)?

- Ограничения при построении дерева
- Обработка построенного дерева после обучения (pruning)

Построение дерева решений

2

Построение дерева решений

1. Используем пороговое правило для разделения выборки
2. Используем информационный критерий для отбора правил разбиения (информационный критерий как некая мера неопределенности по классам в выборке)

Алгоритм:

- Перебираем признаки
  - сортируем выбраный признак по возрастанию
  - перебираем пороги разделения выборки на две части, считая информационный критерий
- Выбираем лучшее разбиение с точки зрения значения информационного критерия

Выбор лучшего разбиения

1 2 3 4 5 6 7 8 9

Есть 1 группа, в ней 2 класса.

Пусть H(R) -«критерии информативности» группы,
больше разнообразия - больше H(R) -хуже для классификатора

Будем измерять улучшение разбиения по функционалу вида:

$$IG(R) = H(R) - q_{left} * H(R_{left}) - q_{right} * H(R_{right}),$$

где q_{left} и q_{right} - доли объектов, попавших в левый или правый класс соответственно

Выбор лучшего разбиения

$$IG(R) = H(R) - q_{left} * H(R_{left}) - q_{right} * H(R_{right})$$

$$H(R) = x > 0$$

$$H(R_{left}) = 0$$

$$H(R_{right}) = 0$$

$$IG(R) = x - 5/9*0 - 4/9*0 = x > 0$$

Выбор лучшего разбиения

Энтропийный критерий

$$H(R) = -\sum_{k=1}^{K} p_k \log p_k$$

K -количество классов
pk -доля класса в выборке

IG(R) = H(R) - qleft*H(Rleft) - qright*H(Rright)

H(R) = -4/9*log2(4/9) - 5/9*log2(5/9) = 0.991

H(Rleft) = -3/4*log2(3/4) - 1/4*log2(1/4) = 0.81

H(Rright) = -1/5*log2(1/5) - 4/5*log2(4/5) = 0.72

IG(R) = 0.991 - 4/9*0.811 - 5/9*0.722 =**0.22**

Информационная энтропия - мера неопределённости случайной величины.
Мера отличия распределения классов от вырожденного

Выбор лучшего разбиения

Критерий Джини

$$H(R) = \sum_{k=1}^{K} p_k(1 - p_k)$$

K -количество классов
pk -доля класса в выборке

$IG(R) = H(R) - q_{left}*H(R_{left}) - q_{right}*H(R_{right})$

$H(R) = 4/9*(1-4/9) + 5/9*(1-5/9) = 0.494$

$H(R_{left}) = 3/4*(1-3/4) + 1/4*(1-1/4) = 0.375$

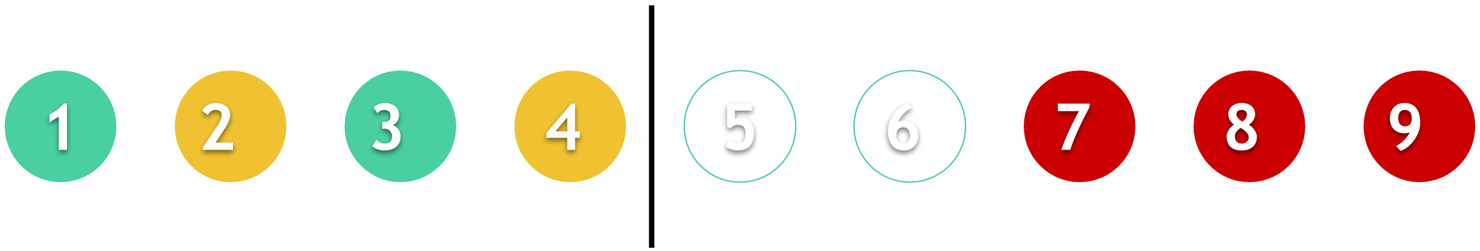
$H(R_{right}) = 1/5*(1-1/5) + 4/5*(1-4/5) = 0.32$

$IG(R) = 0.494 - 4/9*0.375 - 5/9*0.32 =$ **0.15**

Вероятность неправильной классификации, если предсказывать классы с вероятностями их появления в этом узле

Критерий Джини

IG(R) = ?



Выбор лучшего разбиения

Для задачи регрессии

$$H(R) = \min_{c \in \mathbb{Y}} \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - c)^2$$

$$H(R) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \left(y_i - \frac{1}{|R|} \sum_{(x_j, y_j) \in R} y_j \right)^2$$

Квадрат отклонения в качестве функции потерь. Информативность разбиения измеряется дисперсией — чем ниже разброс целевой переменной, тем лучше разбиение.

Критерий Останова

- Останов, когда в каждом листе объекты только одного класса
- Ограничение max глубины дерева
- Ограничение min число объектов в листьях
- Требование улучшения функционала качества при дроблении не менее, чем x или на x%

Стрижка деревьев (Pruning)

- Стрижка из полностью построенного дерева убирает наименее информативные листья
- Стрижка работает лучше раннего останова
- Редко используется, т.к. деревья не используются самостоятельно, а в ансамблях она излишняя (там либо нужно переобучение, либо используется ограничение глубины)
- В основе идея регуляризации: в функционале качестве под дерева линейно штрафуется количество листьев

Проблема пропусков

- Выкинуть объекты с пропусками из обучающей (что на тестовой?)
- Замена на значения на средние, медианные и т.д.
- Заменить на значения вне области значений фич
- Модифицировать алгоритм построения и работы дерева:
  - включать элементы с пропусками в обе ветки дерева, но взвешивать качество разбиения по объёму пропусков
  - Суррогатные разбиения: для объектов с пропущенными значениями выбрать разбиение по другому признаку с максимально похожим разбиением

Категориальные признаки

- Для каждой категории свое поддерево (может получиться много листьев)
- Замена на число и обращение как с количественной переменной (LabelEncoding, WOE)
- One Hot Encoding

Возвращаемый результат

Для классификации:

Возвращается самый представленный в итоговом листе класс или вероятность классов пропорциональная их количеству в итоговом листе.

Для регрессии:

Среднее значение целевой переменной примеров обучения попавших в итоговый лист

Популярные методы построения

Деревья в силу дискретности не сводятся к оптимизации в аналитическом виде, поэтому все методы их построения являются эвристическими и жадными

Популярные методы отличаются ранее рассмотренными параметрами построения дерева:

- ID3: энтропийный критерий, максимально жадный, требуется стрижка(1986)
- C4.5, C5.0: нормированный энтропийный критерий
- CART: критерий Джини-используется в sklearn (optimized)

Пример построения дерева решений

3

Цветки ириса: данные

Ирис щетинистый
(*Iris setosa*)

Ирис разноцветный
(*Iris versicolor*)

Ирис виргинский
(*Iris virginica*)

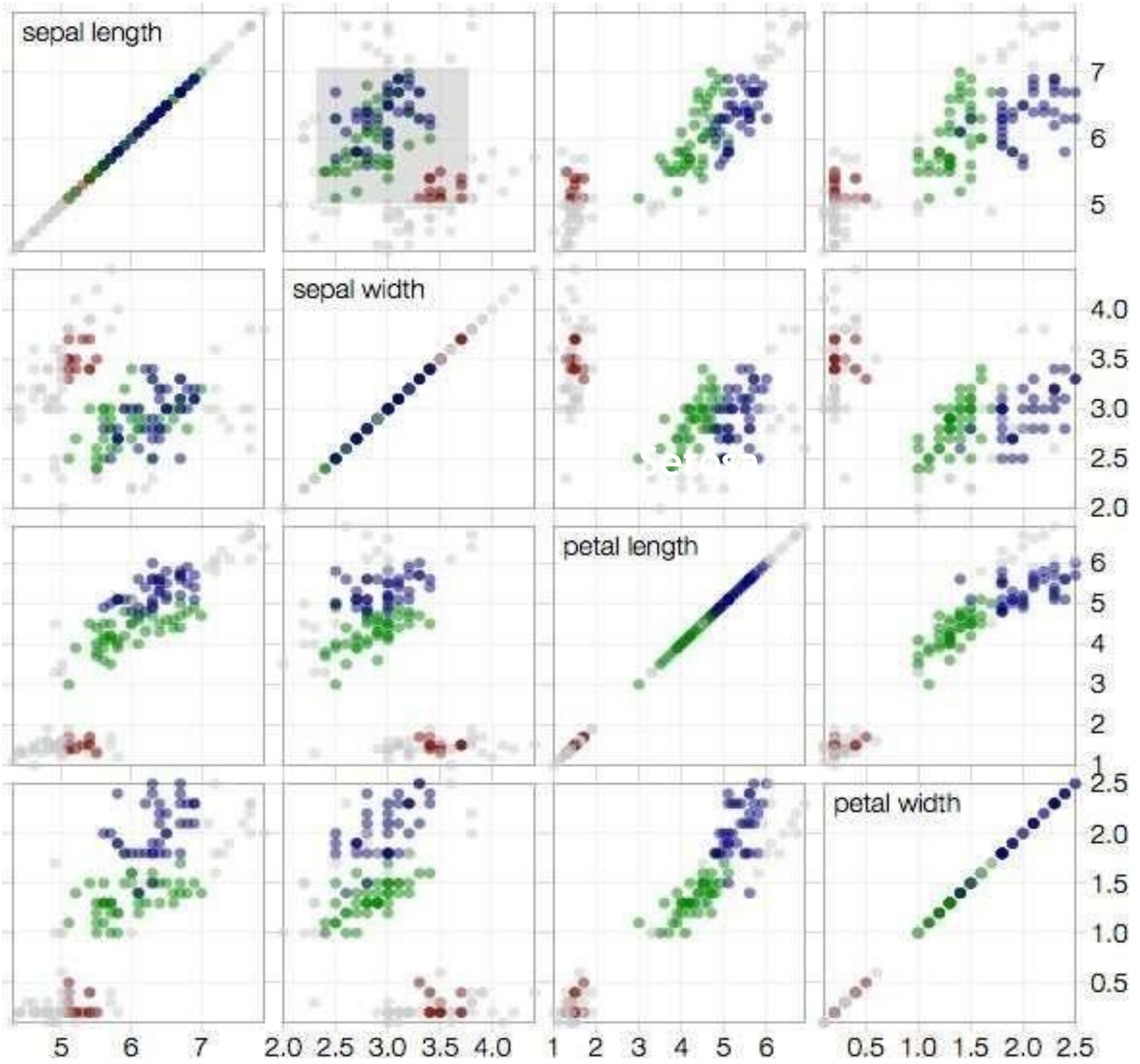
Дано:

- 3 вида цветков ириса
- 4 параметра: 2 длины и 2 ширины листа
- по 50 наборов значений на каждый вид

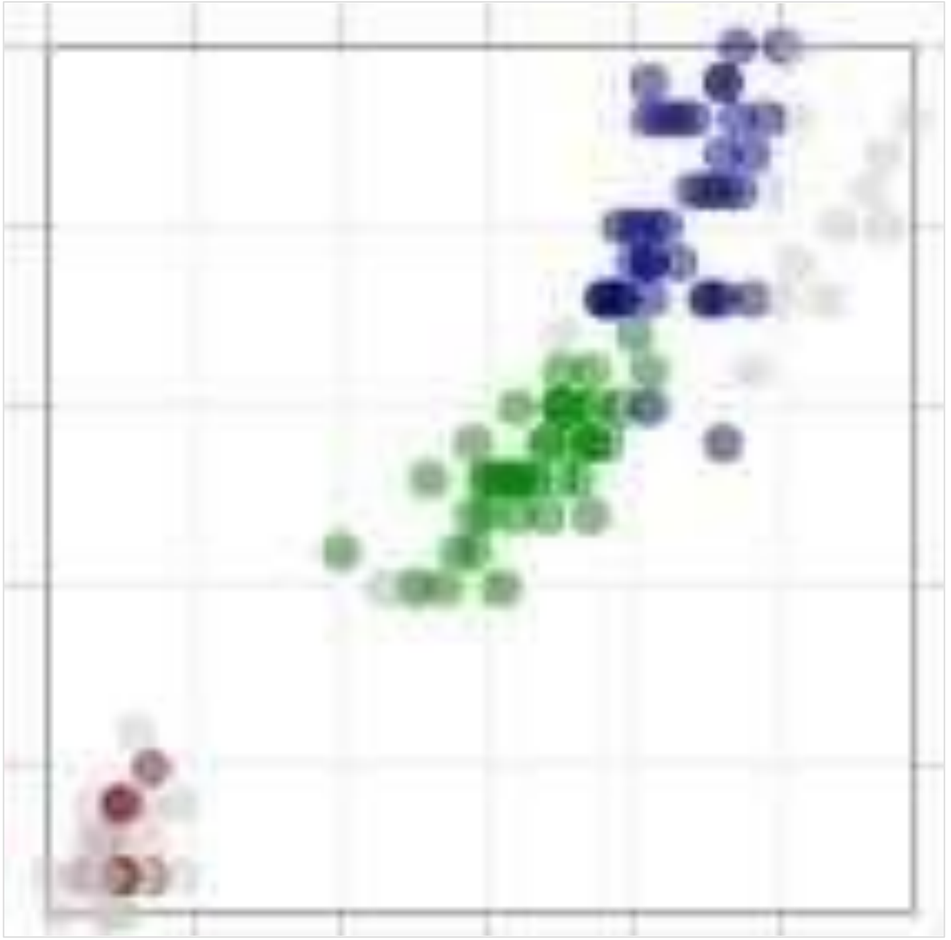
Найти:

- тип цветка по 4 параметрам

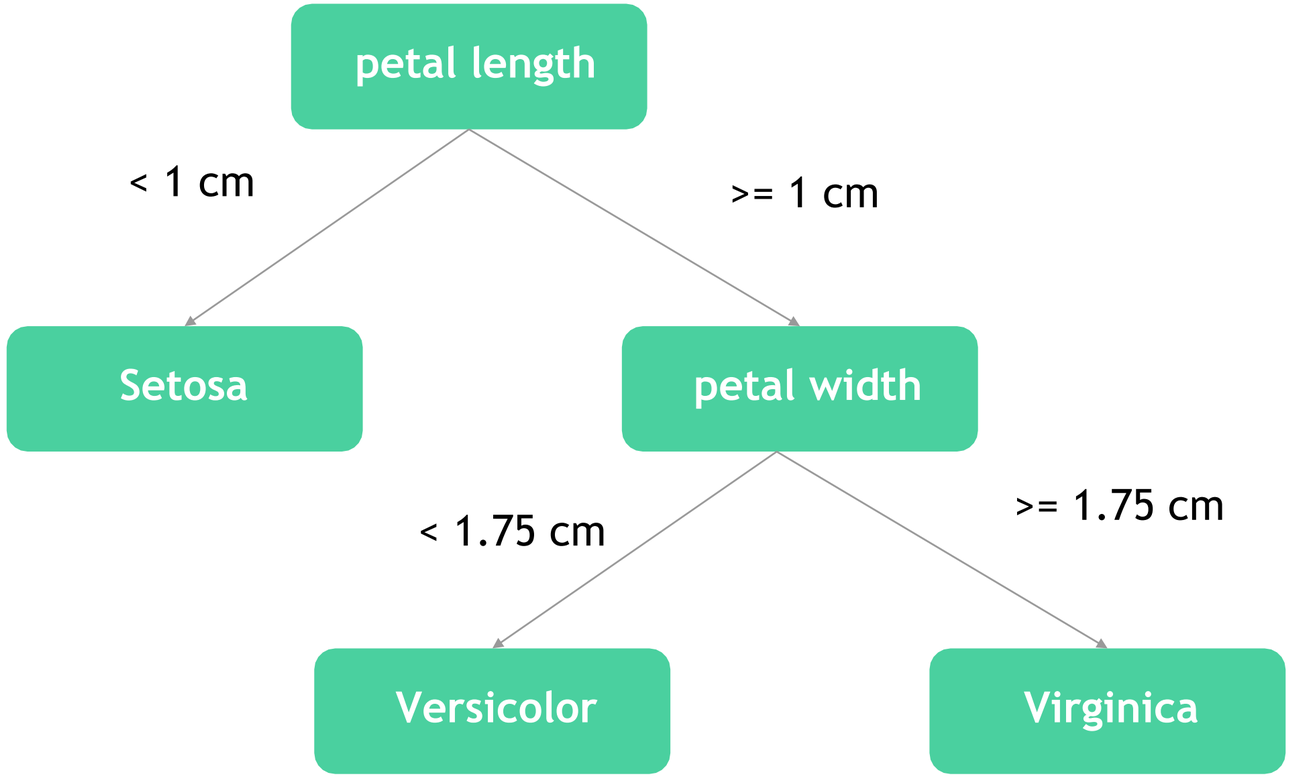
Цветки ириса: связь между признаками



Цветки ириса: связь между признаками



Цветки ириса: дерево решений



Пример реализации

3.1

Реализация в SKLEARN

[sklearn.tree.DecisionTreeClassifier]

*splitter='best'
* max_depth=None
* min_samples_split=2
* min_samples_leaf=1
* min_weight_fraction_leaf=0.0
* max_features=None
* random_state=None
* max_leaf_nodes=None
* min_impurity_split=1e-07
* class_weight=None
* presort=False

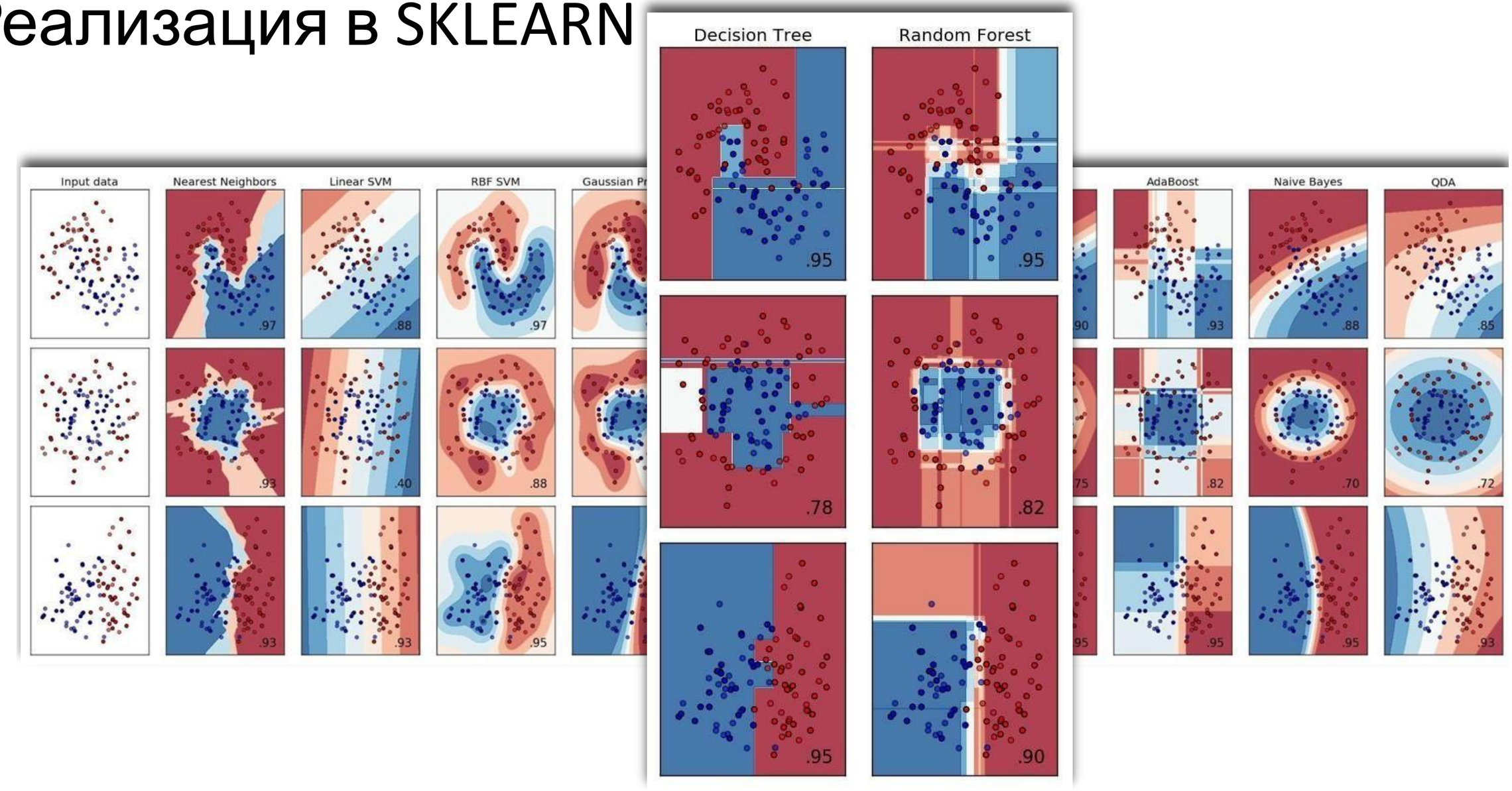
Основные характеристики

- 12 параметров
- Функционал качества: Джини /энтропия
- Реализованы различные простые критерии останова: кол-во объектов, улучшение качества..
- Не реализована стрижка дерева

Основные методы

- fit
- predict,
  predict_proba

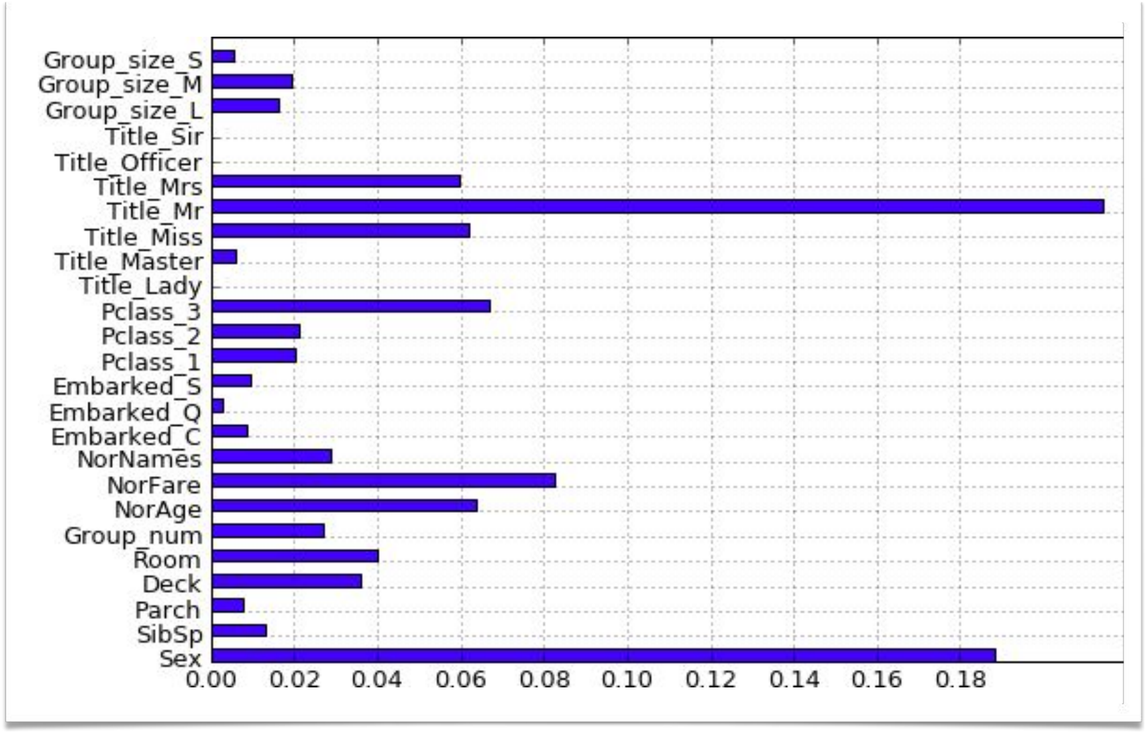
Реализация в SKLEARN



* sklearn, сравнение классификаторов

Реализация в SKLEARN. Бонус

Деревья могут оценивать важность фичей



Например, судя по решению, на выживаемость на Титанике сильнее всего влияли:

- наличие в обращении «Mr.»
- пол
- уровень дохода
- проживание в 3 классе
- возраст
- наличие в обращении «Mrs» / «Miss»

Достоинства и недостатки деревьев решений

4

Достоинства

- Легко интерпретировать, визуализировать, «белый ящик»
- Простота подготовки данных: не требуется нормализация, dummy переменные, возможны пропуски
- Скорость работы
- Формируют четкие и понятные извлекаемые правила (в том числе способны генерировать извлекаемые правила в областях, где специалисту трудно формализовать свои знания).

Недостатки

- Острая проблема переобучения
- Неустойчивость (чувствительны к шумам во входных данных; небольшие изменения обучающей выборки могут привести к глобальным корректировкам модели)
- Не учитывает нелинейные зависимости или даже простые линейные, которые идут не по осям координат
- Чувствителены к несбалансированным классам
- Хорошо интерполирует, плохо экстраполирует. (Дерево решений делает константный прогноз для объектов, находящихся в признаковом пространстве вне параллелепипеда, который охватывает все объекты обучающей выборки.
- Жадный алгоритм построения дерева не гарантирует его оптимальности

Спасибо за внимание!