Машинное обучение (Machine Learning)

Деревья решений (Decision trees)

Уткин Л.В.

Содержание

Презентация является компиляцией и заимствованием материалов из замечательных курсов и презентаций по машинному обучению:

К.В. Воронцова, А.Г. Дьяконова, Н.Ю. Золотых, С.И. Николенко, Andrew Moore, Lior Rokach, Rong Jin, Luis F. Teixeira, Alexander Statnikov и других.

Общие определения деревьев решений

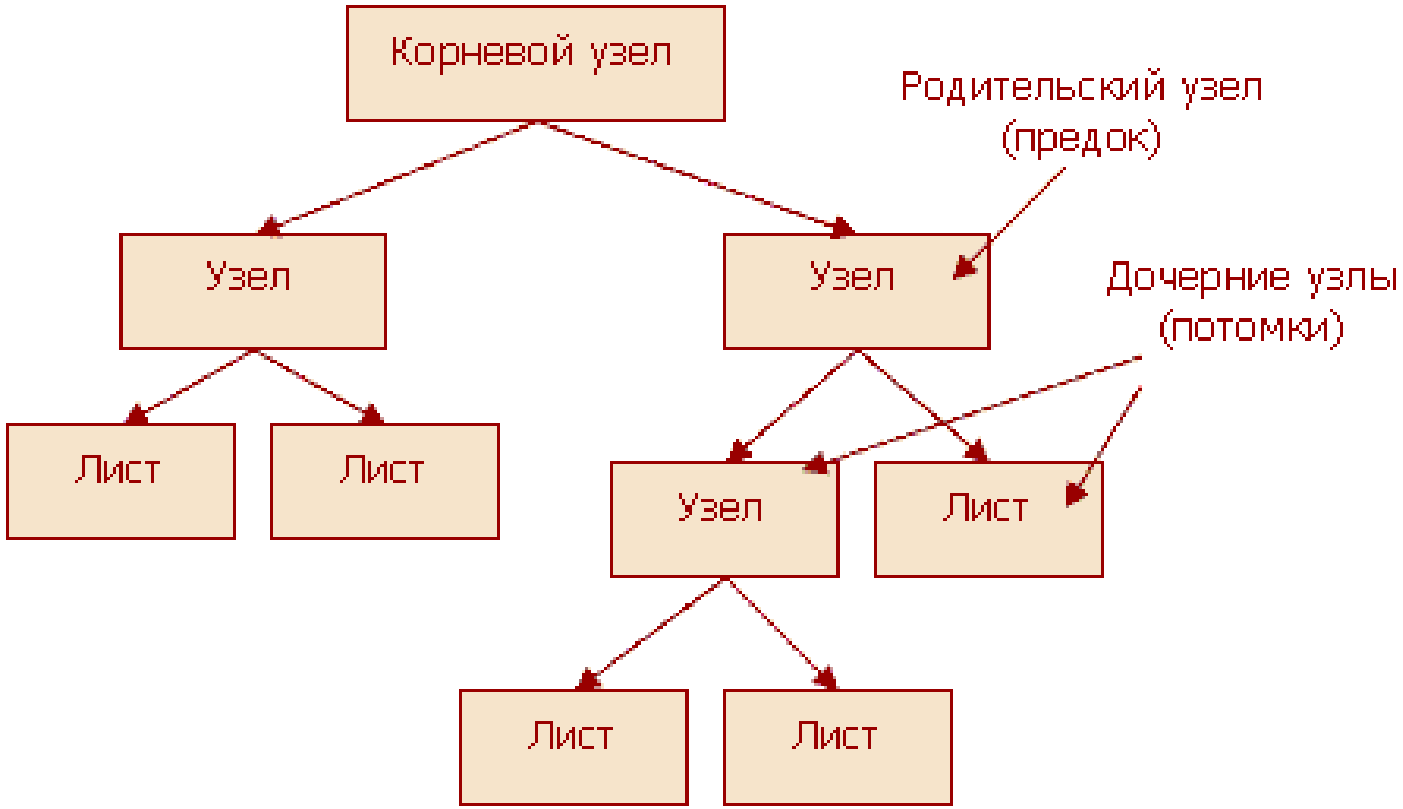
Деревья решений – это способ представления правил в иерархической, последовательной структуре, где каждому объекту соответствует единственный узел, дающий решение

Деревья решений – это логический алгоритм классификации, основанный на поиске конъюнктивных закономерностей.

Основные понятия деревьев

- **Деревом** называется конечный связный граф с множеством вершин V, не содержащий циклов и имеющий выделенную вершину $v_0 \in V$, в которую не входит ни одно ребро. Эта вершина называется **корнем дерева**.
- Вершина не имеющая выходящих ребер, называется **терминальной** или **листом**, соответствуют классам
- Остальные вершины называются **внутренними,** они соответствуют признакам.
- Дерево называется **бинарным**, если из любой его внутренней вершины выходит ровно два ребра.
- Выходящие ребра связывают каждую внутреннюю вершину v с левой дочерней вершиной L_v и с правой дочерней вершиной R_v.

Элементы дерева



Определение бинарных деревьев

Бинарное решающее дерево – это алгоритм классификации, задающийся бинарным деревом, в котором каждой внутренней вершине $v \in V$ приписан предикат $\beta_v : X \to \{0, 1\}$, каждой терминальной вершине $v \in V$ приписано имя класса $c_v \in Y$. При классификации объекта $x \in X$ он проходит по дереву путь от корня до некоторого листа.

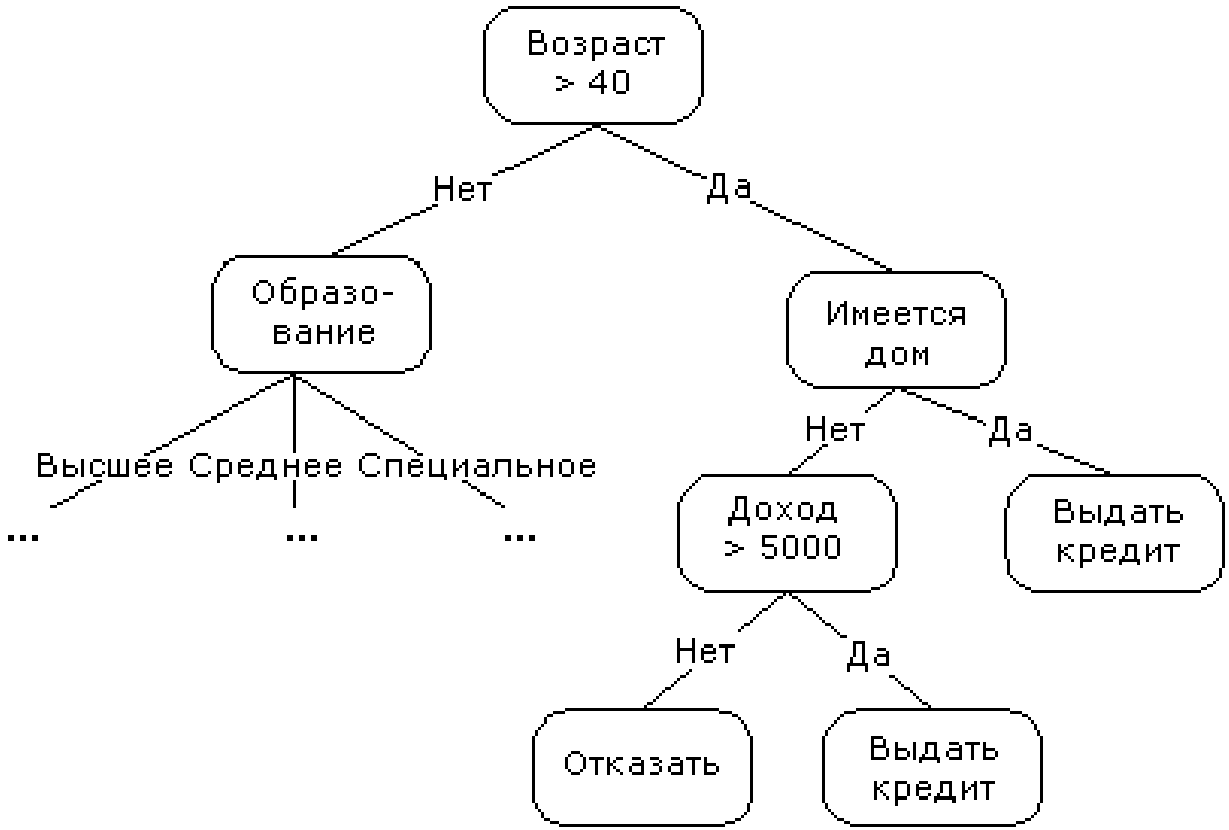
Применение деревьев решений

- **Описание данных**: Деревья решений позволяют хранить информацию о данных в компактной форме, вместо них мы можем хранить дерево решений, которое содержит точное описание объектов.
- **Классификация**: Деревья решений отлично справляются с задачами классификации, т.е. отнесения объектов к одному из заранее известных классов. Целевая переменная должна иметь дискретные значения.
- **Регрессия**: Если целевая переменная имеет непрерывные значения, ДР позволяют установить зависимость целевой переменной от входных переменных

Пример обучающей выборки (выдача кредита)

	возраст	наличие дома	доход	образование	кредит
x_1	32	нет	2000	среднее	нет
x_2	54	да	12000	высшее	да
x_3	73	нет	800	специальное	нет
...	...	...			
x_{50}	18	да	200	среднее	да

Пример дерева классификации (Выдавать ли кредит?)



Этапы конструирования деревьев

1. “Построение” или “создание” дерева (tree building): выбор **критерия расщепления** и **остановки** обучения
2. “Сокращение” дерева (tree pruning): **сокращения** дерева и **отсечение** некоторых его ветвей

Критерий расщепления

- Расщепление должно разбивать исходное множество данных таким образом, чтобы объекты подмножеств, получаемых в результате этого разбиения, являлись представителями одного класса или же были максимально приближены к такому разбиению.
- Количество объектов из других классов, так называемых “примесей”, в каждом классе должно стремиться к минимуму.

Общий жадный алгоритм построения ДР

Жадный алгоритм – алгоритм, заключающийся в принятии локально оптимальных решений на каждом этапе, допуская, что конечное решение также окажется оптимальным.

Алгоритм:

1. На каждой итерации для входного подмножества обучающего множества строится такое разбиение пространства гиперплоскостью (ортогональной одной их осей координат), которое минимизировало бы среднюю меру неоднородности двух полученных подмножеств.
2. Данная процедура выполняется рекурсивно для каждого полученного подмножества до тех пор, пока не будут достигнуты критерии остановки.

Алгоритм CART

Алгоритм CART (Classification and Regression Tree) разработан в 1974-1984 годах L.Breiman (Berkeley), J.Friedman (Stanford), C.Stone (Berkeley) и R.Olshen (Stanford).
Алгоритм CART предназначен для построения *бинарного* дерева решений.
Особенности алгоритма CART:

- функция оценки качества разбиения;
- механизм отсечения дерева;
- алгоритм обработки пропущенных значений;
- построение деревьев регрессии.

Критерий расщепления и алгоритм CART

Критерии расщепления или меры неоднородности множества относительно его меток:

- мера энтропии (cross-entropy): $-\sum_{i=1}^{C} p_i \log(p_i)$
- индекс Gini: $\sum_{i=1}^{C} p_i(1 - p_i)$

p_i - частота или вероятность точек i-го класса в блоке;
Если набор данных разбивается на две части $_1$ и $_2$ с числом примеров в каждом N_1 и N_2 соответственно, тогда показатель качества разбиения будет равен:

$$\text{Gini}_{\text{split}}(T) = \frac{N_1}{N} \cdot \text{Gini}(T_1) + \frac{N_2}{N} \cdot \text{Gini}(T_2)$$

Чем меньше критерий расщепления, тем лучше расщепление.

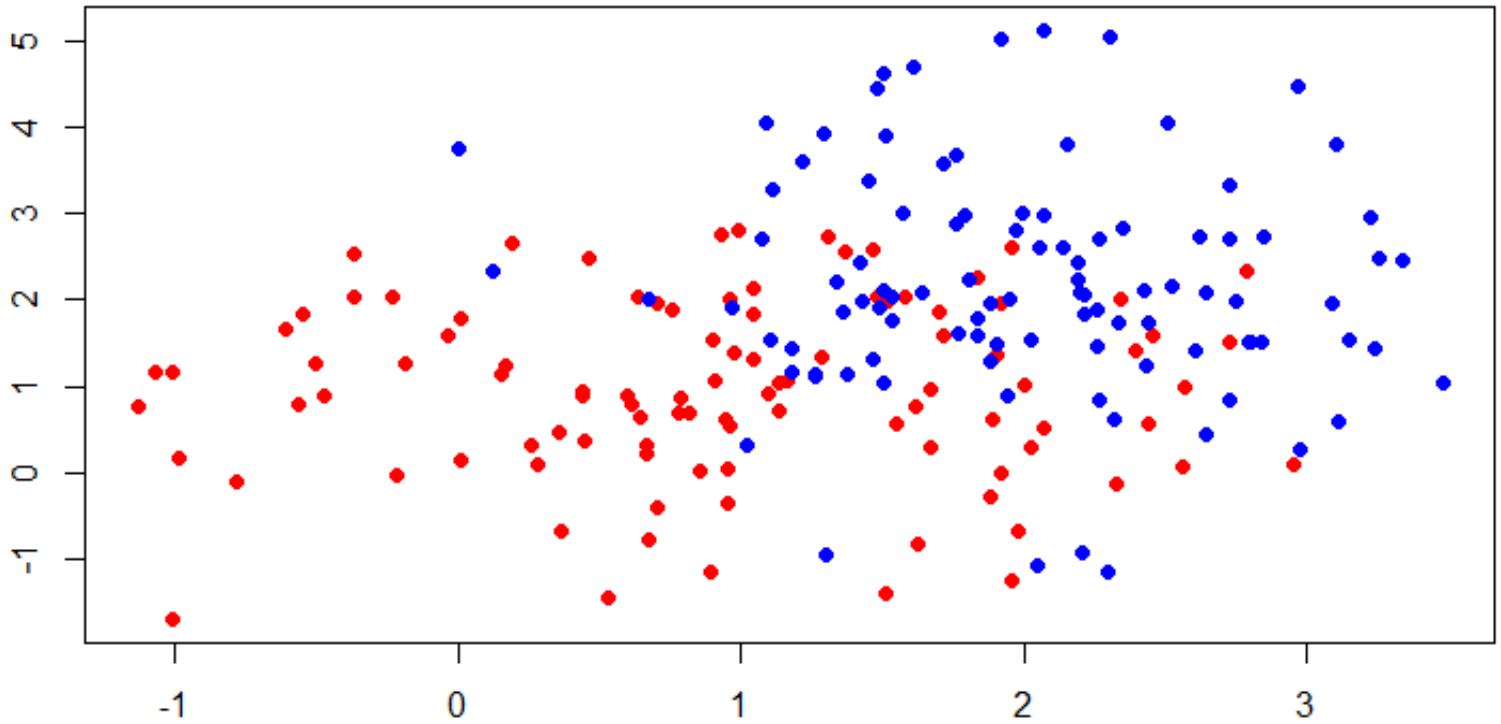
Процедура расщепления в алгоритме CART (1)

1. Выбирается k-ый признак f_k с множеством значений $X^{(k)}$.
2. Определяется такое значение $x_0^{(k)} \in X^{(k)}$ для всех признаков f_k, $k = 1, ..., m$, чтобы мера неоднородности $\text{Gini}_{\text{split}}(T)$ была минимальной, т.е.

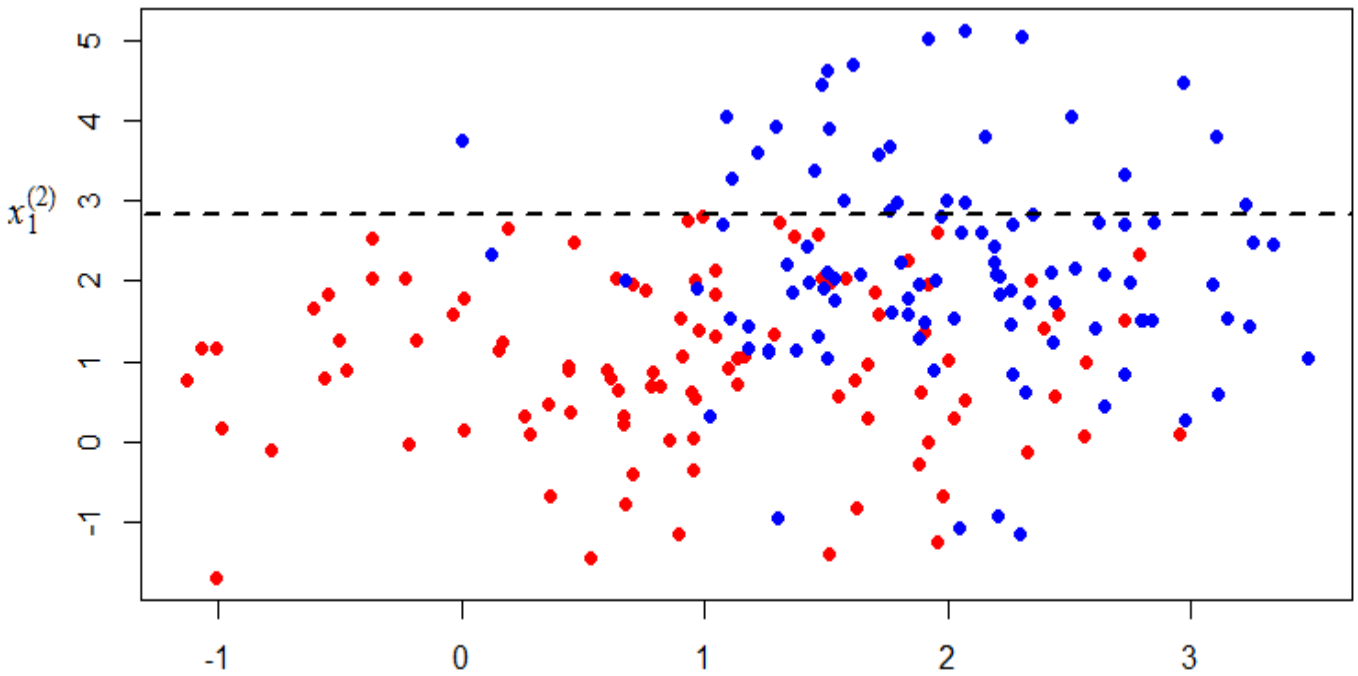
$$x_0^{(k)} = \arg \min_{f_k, x^{(k)} \in X^{(k)}} \text{Gini}_{\text{split}}(T, x^{(k)})$$

3. Данная процедура выполняется рекурсивно для каждого полученного подмножества до тех пор, пока не будут достигнуты критерии остановки.

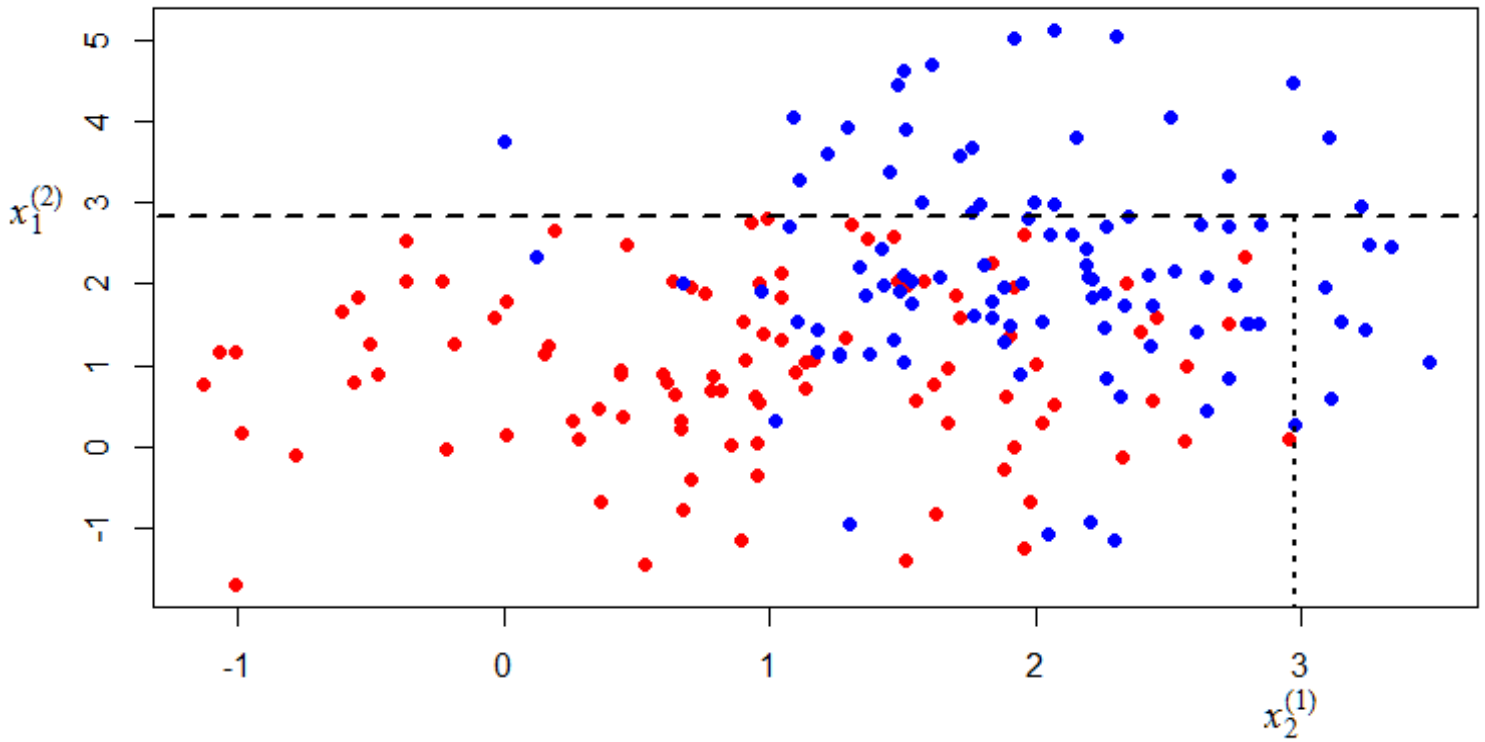
Процедура расщепления в алгоритме CART (2)



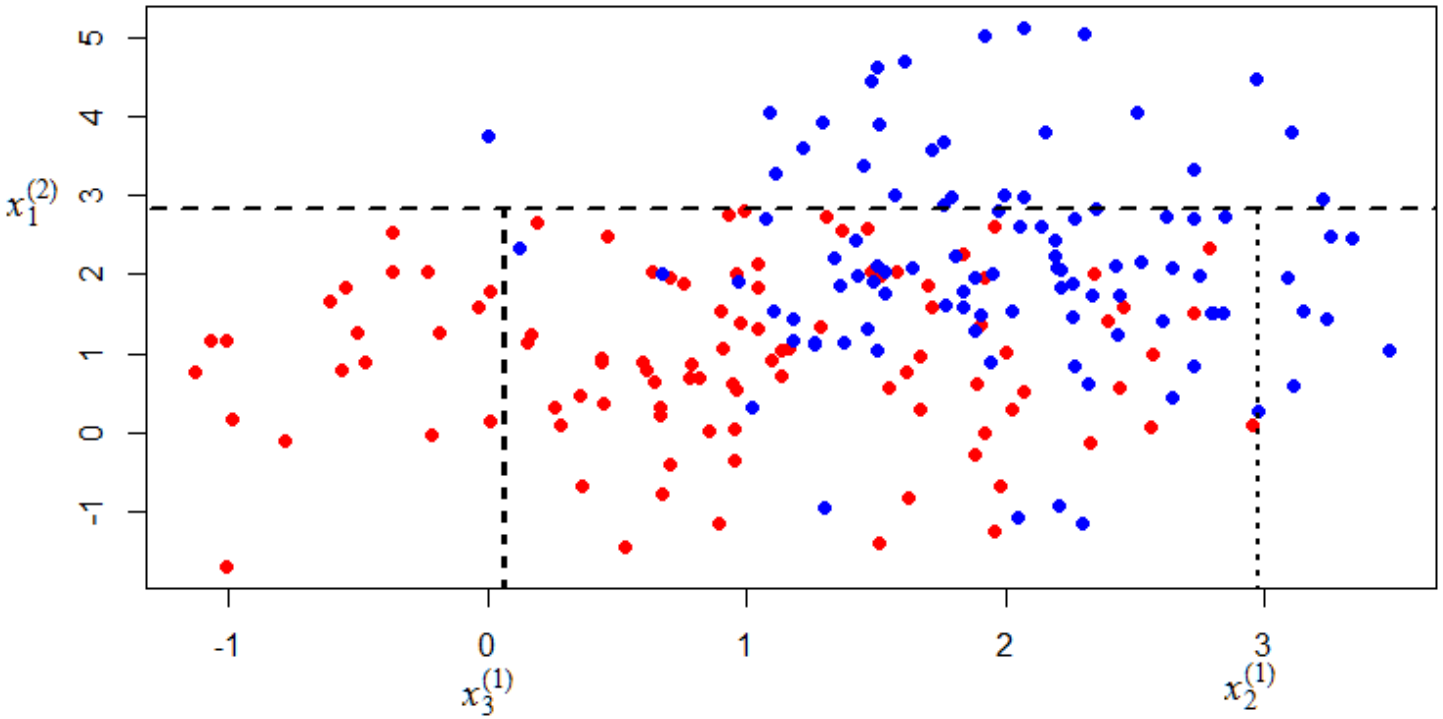
Процедура расщепления в алгоритме CART (3)



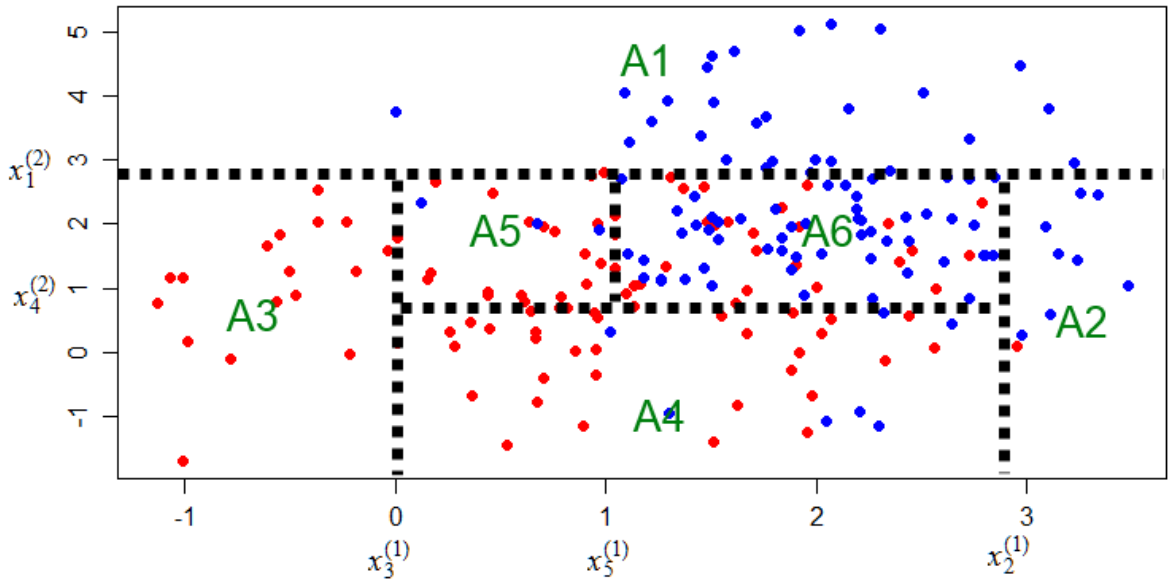
Процедура расщепления в алгоритме CART (4)

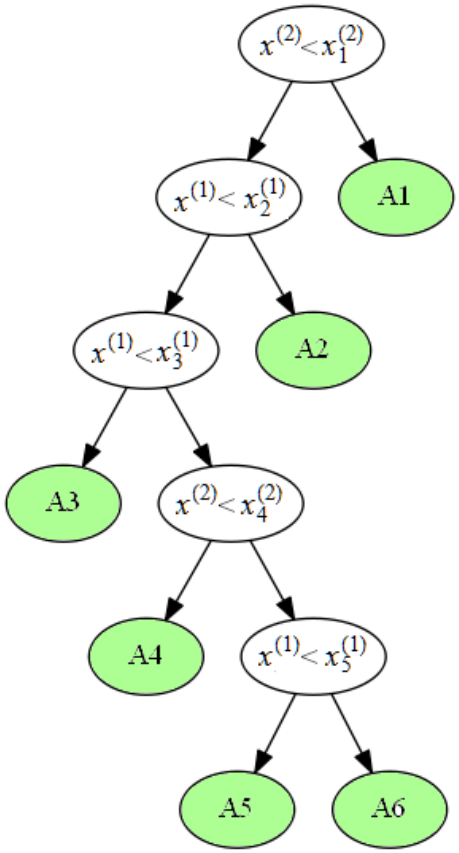


Процедура расщепления в алгоритме CART (5)



Процедура расщепления в алгоритме CART (6)





Процедура расщепления в алгоритме CART (7)

1. Выбирается k-ый признак f_k с множеством значений $X^{(k)}$.
2. Определяется такое значение $x_0^{(k)} \in X^{(k)}$ для всех признаков f_k, $k = 1, ..., m$, чтобы мера неоднородности $\text{Gini}_{\text{split}}(T)$ была минимальной, т.е.

$$x_0^{(k)} = \arg \min_{f_k, x^{(k)} \in X^{(k)}} \text{Gini}_{\text{split}}(T, x^{(k)})$$

3. Данная процедура выполняется рекурсивно для каждого полученного подмножества до тех пор, пока не будут достигнуты критерии остановки.

Критерий остановки

Остановка *- такой момент в процессе построения дерева, когда следует прекратить дальнейшие ветвления*

- достигнута максимальная глубина узла;
- вероятность доминирующего класса в разбиении превышает некоторый порог (например, 0.95);
- количество элементов в подмножестве меньше некоторого порога.

Сокращение дерева или отсечение ветвей

Сокращение *- это компромисс между получением дерева "подходящего размера" и получением наиболее точной оценки классификации.*

Осуществляется путем **отсечения** (pruning) некоторых ветвей.
Отсечение (прореживание) важно не только для упрощения деревьев, но и для избежания переобучения.

Основные характеристики алгоритма CART

- бинарное расщепление с критерием расщепления - индексом Gini,
- специальный механизм отсечения (minimalcost-complexity tree pruning),
- V-fold cross-validation,
- принцип “вырастить дерево, а затем сократить”,
- высокая скорость построения.

Другие известные алгоритмы

- **Алгоритм C4.5** строит дерево решений с неограниченным количеством ветвей у узла, может работать только с дискретным зависимым атрибутом, может решать только задачи классификации
- **Алгоритм ID3**. В основе лежит понятие информационной энтропии. Использует рекурсивное разбиение подмножеств в узлах дерева по одному из выбранных атрибутов.
- **Алгоритм MARS** (Multivariate adaptive regression splines).
- **Алгоритм CHAID** (CHi-squared Automatic Interaction Detection).

Достоинства и недостатки деревьев решений

Преимущества:

- интерпретируемость, допускаются разнотипные данные, возможность обхода пропусков;

Недостатки:

- переобучение, неустойчивость к шуму, составу выборки, критерию;

Способы устранения этих недостатков:

- редукция, композиции (леса) деревьев

Программная реализация в R

- https://cran.r-project.org/web/views/MachineLearning.html
- Пакет **rpart**, функция **rpart**
- Пакет **C50**, функция **C5.0.default**
- Пакет **data.tree**, функция **data.tree**

Вопросы

?