



北京邮电大学

Beijing University of Posts and Telecommunications

分类与决策树模型

戚 琦

网络与交换技术国家重点实验室 网络智能研究中心 科研楼511

qiqi8266@bupt.edu.cn

13466759972



根据训练数据是否拥有标签

学习任务

监督学习(supervised learning)

无监督学习(unsupervised learning)

半监督学习(semi-supervised learning)

强化学习(reinforcement learning)

弱监督学习 (weak supervised learning)

- 分类：对于给定一个对象，从一个事先定好的分类体系中挑出一个 (或者多个) 最适合该对象的类别。
 - 对象：可以是任何东西
 - 事先定好的分类体系：可能有结构
 - 最适合：判断标准
- 便于今后查找是其最直接、最普遍的应用

- 分类体系一般人工构造
 - 政治、体育、军事
- 分类系统可以是层次结构
- 分类模式
 - 2类问题，属于或不属于 (binary)
 - 多类问题，多个类别 (multi-class)，可拆分成2类问题
 - 一个对象可以属于多类 (multi-label)

分类基本步骤

- 定义分类体系
- 将预先分类过的数据作为训练集
- 从训练集中得出分类模型（需要测试过程，不断细化）
- 用训练获得出的分类模型对其它数据加以分类

自动分类的优点

- 减小人工分类的繁杂工作
- 提高信息处理的效率
- 减小人工分类的主观性



分类问题的一些评价方式

- 准确率: (P, precision)
- 召回率: (R, recall)
- F-Measure

$$F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}}$$

$$F_1 = \frac{2PR}{P + R}$$

正确答案



		Human	
		True	False
Classifier	Yes	a	b
	No	c	d

准确率 $P = a / (a + b)$

召回率 $R = a / (a + c)$ 查全率





正确答案
↓

		Human	
		True	False
Classifier	Yes	a	b
	No	c	d

准确率 $P = a / (a + b)$



召回率 $R = a / (a + c)$ 查全率

假设异常检测任务中，异常样本为正样本数据集：**100**条数据，其中**10**条异常样本（正样本），**90**条正常样本（负样本）

模型A：找到了20条正样本，包括10条真正异常的样本，以及真正正常而误判为异常的样本

模型A的准确率：50%，召回率：100%

模型B：找到了5条正样本，即5条真正异常的样本，并且把5条真正异常的误判为正常

模型B的准确率：100%，召回率：50%



- 决策树基本算法
- 决策树的剪枝
- 回归决策树
- 属性缺失
- 多变量决策树



根据训练数据**是否拥有标签**

学习任务 {

- 监督学习(supervised learning) 分类、回归 (x_i, y_i)
- 无监督学习(unsupervised learning) 聚类
- 半监督学习: 输入数据部分有标注, 部分没有标注, 介于监督学习与非监督学习之间。
- 若监督学习: 输入数据有标注, 但是与要解决的问题相比, 信息不够。
- 强化学习(reinforcement learning)

决策树 (decision tree) 模型常常用来**解决分类和回归问题**。常见的算法包括 CART (Classification And Regression Tree)、ID3、C4.5等。

17个样本

表 4.1 西瓜数据集 2.0

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

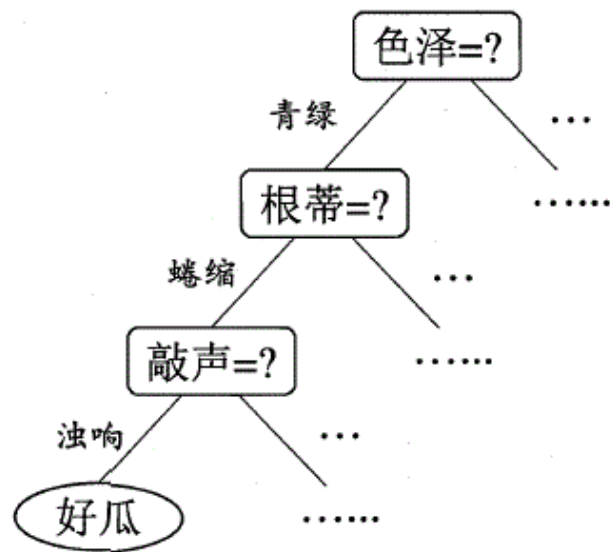
标签：
二分类



二分类学习任务
属性
属性值

- 根结点：包含全部样本
- 叶结点：对应决策结果 “好瓜” “坏瓜”
- 内部结点：对应属性测试

参考周志华老师《机器学习》



决策树学习的目的：为了产生一颗泛化能力强的决策树，即处理未见示例能力强。



输入：训练集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$;

属性集 $A = \{a_1, a_2, \dots, a_d\}$.

过程：函数 TreeGenerate(D, A)

1: 生成结点 node;

2: if D 中样本全属于同一类别 C then

3: 将 node 标记为 C 类叶结点; return

4: end if

5: if $A = \emptyset$ OR D 中样本在 A 上取值相同 then

6: 将 node 标记为叶结点, 其类别标记为 D 中样本数最多的类; return

7: end if

8: 从 A 中选择最优划分属性 a_* ;

9: for a_* 的每一个值 a_*^v do

10: 为 node 生成一个分支; 令 D_v 表示 D 中在 a_* 上取值为 a_*^v 的样本子集;

11: if D_v 为空 then

12: 将分支结点标记为叶结点, 其类别标记为 D 中样本最多的类; return

13: else

14: 以 TreeGenerate($D_v, A \setminus \{a_*\}$) 为分支结点

15: end if

16: end for

输出：以 node 为根结点的一棵决策树

(1) 无需划分

(2) 无法划分

(3) 不能划分

递归过程

第 (1) 种情形：样本都属于同一类
(标签一样)

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否

例子：

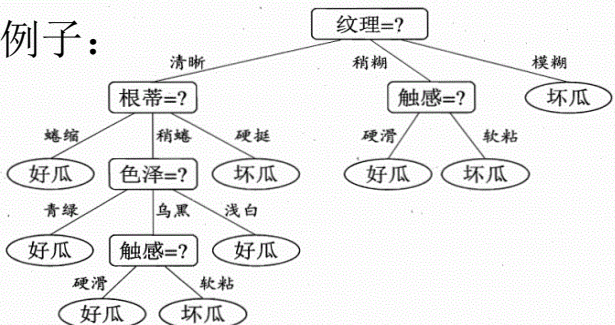


图 4.4 在西瓜数据集 2.0 上基于信息增益生成的决策树

图 4.2 决策树学习基本算法





输入: 训练集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$;

属性集 $A = \{a_1, a_2, \dots, a_d\}$.

过程: 函数 TreeGenerate(D, A)

- 1: 生成结点 node;
- 2: if D 中样本全属于同一类别 C then (1) 无需划分
- 3: 将 node 标记为 C 类叶结点; return
- 4: end if
- 5: if $A = \emptyset$ OR D 中样本在 A 上取值相同 then (2) 无法划分
- 6: 将 node 标记为叶结点, 其类别标记为 D 中样本数最多的类; return
- 7: end if
- 8: 从 A 中选择最优划分属性 a_* ;
- 9: for a_* 的每一个值 a_*^v do
- 10: 为 node 生成一个分支; 令 D_v 表示 D 中在 a_* 上取值为 a_*^v 的样本子集;
- 11: if D_v 为空 then
- 12: 将分支结点标记为叶结点, 其类别标记为 D 中样本数最多的类; return
- 13: else
- 14: 以 TreeGenerate($D_v, A \setminus \{a_*\}$) 为分支结点 (3) 不能划分
- 15: end if
- 16: end for

输出: 以 node 为根结点的一棵决策树

图 4.2 决策树学习基本算法

递归过程

第 (2) 种情形: 所有属性取值相同, 即叶子节点, 设定为该结点所含样本最多的类别, 利用当前结点的后验分布

例子:

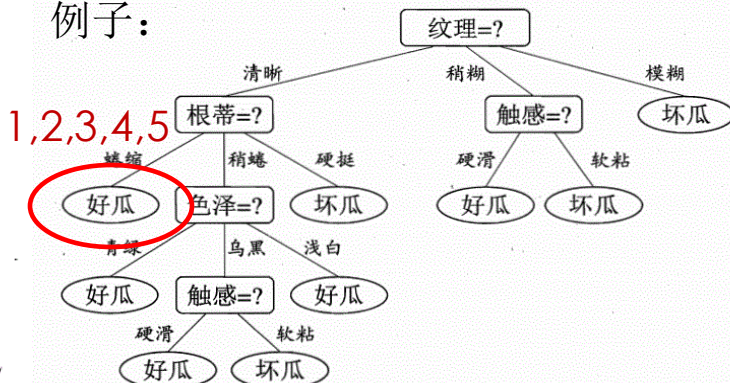


图 4.4 在西瓜数据集 2.0 上基于信息增益生成的决策树





输入: 训练集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$;

属性集 $A = \{a_1, a_2, \dots, a_d\}$.

过程: 函数 TreeGenerate(D, A)

1: 生成结点 node;

2: if D 中样本全属于同一类别 C then

3: 将 node 标记为 C 类叶结点; return

4: end if

5: if $A = \emptyset$ OR D 中样本在 A 上取值相同 then

6: 将 node 标记为叶结点, 其类别标记为 D 中样本数最多的类; return

7: end if

8: 从 A 中选择最优划分属性 a_* ;

9: for a_* 的每一个值 a_*^v do

10: 为 node 生成一个分支; 令 D_v 表示 D 中在 a_* 上取值为 a_*^v 的样本子集;

11: if D_v 为空 then

12: 将分支结点标记为叶结点, 其类别标记为 D 中样本最多的类; return

13: else

14: 以 TreeGenerate($D_v, A \setminus \{a_*\}$) 为分支结点

15: end if

16: end for

输出: 以 node 为根结点的一棵决策树

(1) 无需划分

(2) 无法划分

(3) 不能划分

递归过程

第 (3) 种情形: 分支为空, 即当前节点包含的样本集合为空, 设定为其父结点所含样本最多的类别, 把父结点的样本分布作为当前结点的先验分布

例子:

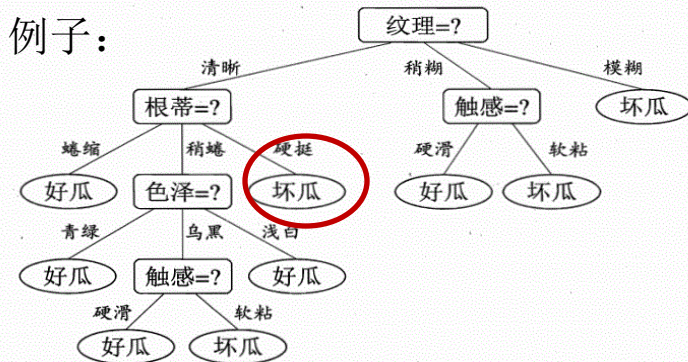
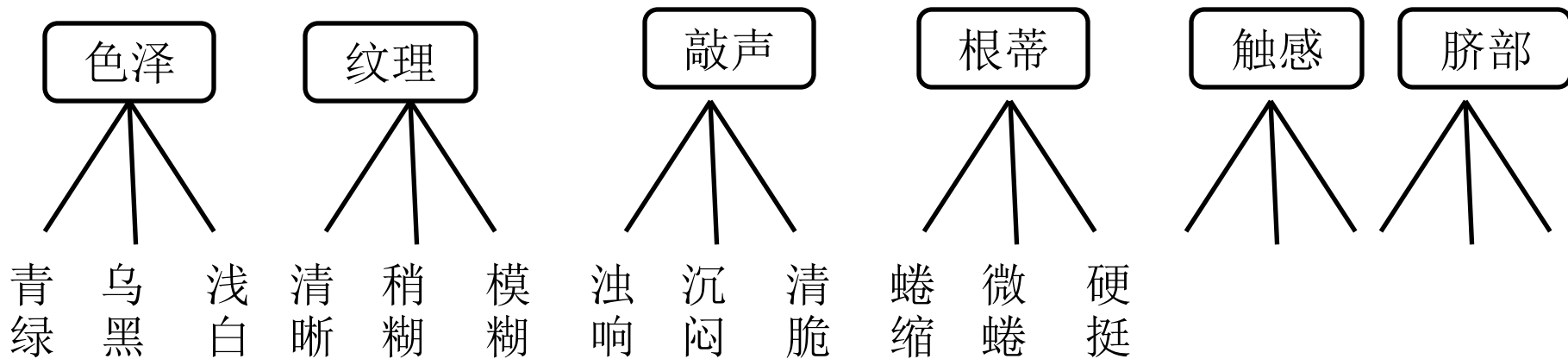


图 4.4 在西瓜数据集 2.0 上基于信息增益生成的决策树

图 4.2 决策树学习基本算法



西瓜的6个属性选择哪个作为根节点？



★ 关键：选择最优划分属性（算法的第8行）

什么样的划分属性是最优的？

希望决策树的分支结点所包含的样本尽可能属于同一类别，即**结点的“纯度”越来越高**，可以高效地从根结点到达叶结点，得到决策结果。

三种度量结点“纯度”的指标：

- 信息增益
- 增益率
- 基尼指数

1. 信息增益

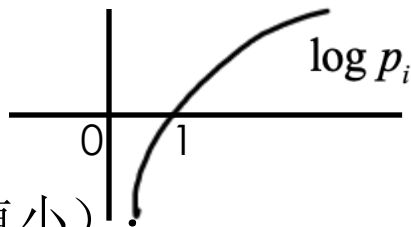
香农提出了“**信息熵**”的概念，解决了对信息的量化度量问题。
香农用“信息熵”的概念来描述信源的**不确定性**。

“信息熵” (information entropy) 是度量样本集合纯度最常用的一种指标。
假定当前样本集合 D 中第 k 类样本所占的比例为 p_k ($k = 1, 2, \dots, |\mathcal{Y}|$), 则 D 的信息熵定义为

$$\text{Ent}(D) = - \sum_{k=1}^{|\mathcal{Y}|} p_k \log_2 p_k . \quad (4.1)$$

Ent(D) 的值越小, 则 D 的纯度越高. 对于二分类任务 $|\mathcal{Y}| = 2$

“信息量”：衡量不确定性的多少



- 不会是负数
- 不确定性函数 f 是概率 p 的单调**递减函数**（概率大 f 值小）；
- 可加性：两个独立符号所产生的不确定性应等于各自不确定性之和，即

$$f(p_1 \times p_2) = f(p_1) + f(p_2)$$

同时满足这三个条件的函数 f 是**负的对数函数**，即 $f(p_i) = \log \frac{1}{p_i} = -\log p_i$

一个事件的信息量就是这个**事件发生的概率的负对数**。

信息熵是跟**事件的所有可能性**有关的，是**平均而言**发生一个事件得到的信息量大小。信息熵其实是信息量的期望。

$$E[-\log p_i] = -\sum_{i=1}^n p_i \log p_i$$

假定离散属性 a 有 V 个可能的取值 $\{a^1, a^2, \dots, a^V\}$, 若使用 a 来对样本集 D 进行划分, 则会产生 V 个分支结点, 其中第 v 个分支结点包含了 D 中所有在属性 a 上取值为 a^v 的样本, 记为 D^v . 我们可根据式(4.1) 计算出 D^v 的信息熵, 再考虑到不同的分支结点所包含的样本数不同, 给分支结点赋予权重 $|D^v|/|D|$, 即样本数越多的分支结点的影响越大, 于是可计算出用属性 a 对样本集 D 进行划分所获得的“信息增益” (information gain)

$$\text{Gain}(D, a) = \text{Ent}(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Ent}(D^v). \quad (4.2)$$

信息熵

属性值出现的比例

信息增益越大, 则意味着使用属性 a 来进行划分所获得的“纯度提升”越大。

决策树算法第8行选择属性 $a_* = \arg \max_{a \in A} \text{Gain}(D, a)$.

著名的ID3决策树算法

举例：求解划分根结点的最优划分属性

数据集包含17个训练样例：

8个正例（好瓜）占 $p_1 = \frac{8}{17}$

9个反例（坏瓜）占 $p_2 = \frac{9}{17}$

对于二分类任务 $|y|=2$

以属性“色泽”为例计算其信息增益

根结点的信息熵：

$$\text{Ent}(D) = - \sum_{k=1}^2 p_k \log_2 p_k = - \left(\frac{8}{17} \log_2 \frac{8}{17} + \frac{9}{17} \log_2 \frac{9}{17} \right) = 0.998 .$$



用“色泽”将根结点划分后获得3个分支结点的信息熵分别为：

$$\text{Ent}(D^1) = - \left(\frac{3}{6} \log_2 \frac{3}{6} + \frac{3}{6} \log_2 \frac{3}{6} \right) = 1.000 ,$$

$$\text{Ent}(D^2) = - \left(\frac{4}{6} \log_2 \frac{4}{6} + \frac{2}{6} \log_2 \frac{2}{6} \right) = 0.918 ,$$

$$\text{Ent}(D^3) = - \left(\frac{1}{5} \log_2 \frac{1}{5} + \frac{4}{5} \log_2 \frac{4}{5} \right) = 0.722 ,$$

属性“色泽”的信息增益为：

$$\begin{aligned} \text{Gain}(D, \text{色泽}) &= \text{Ent}(D) - \sum_{v=1}^3 \frac{|D^v|}{|D|} \text{Ent}(D^v) \\ &= 0.998 - \left(\frac{6}{17} \times 1.000 + \frac{6}{17} \times 0.918 + \frac{5}{17} \times 0.722 \right) \\ &= 0.109 . \end{aligned}$$





类似的, 我们可计算出其他属性的信息增益:

$$\text{Gain}(D, \text{根蒂}) = 0.143; \quad \text{Gain}(D, \text{敲声}) = 0.141;$$

$$\text{Gain}(D, \text{纹理}) = 0.381; \quad \text{Gain}(D, \text{脐部}) = 0.289;$$

$$\text{Gain}(D, \text{触感}) = 0.006.$$

信息熵最大

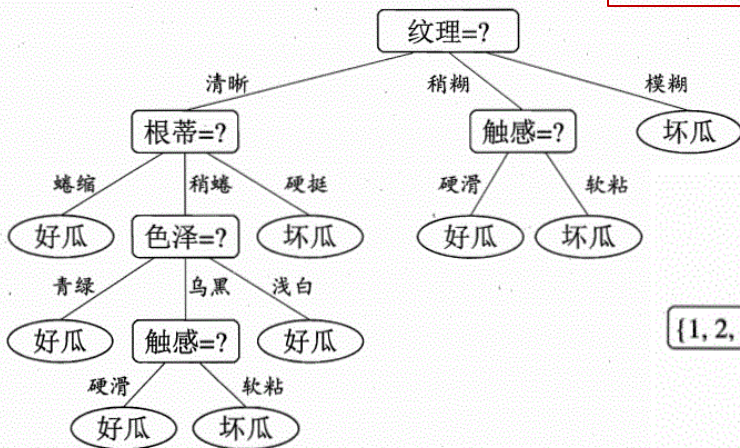


图 4.4 在西瓜数据集 2.0 上基于信息增益生成的决策树

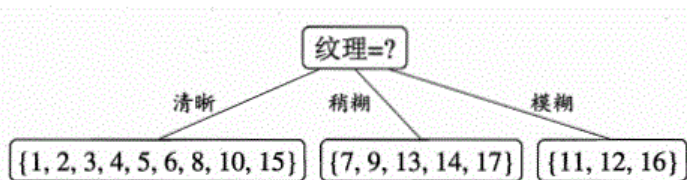


图 4.3 基于“纹理”属性对根结点划分



若把“编号”也作为一个**候选划分属性**，则属性“编号”的信息增益为：

根结点的信息熵仍为： $Ent(D) = 0.998$

用“编号”将根结点划分后获得17个分支结点的信息熵均为：

$$Ent(D^1) = L = Ent(D^{17}) = -\left(\frac{1}{1} \log_2 \frac{1}{1} + \frac{0}{1} \log_2 \frac{0}{1}\right) = 0$$

则“编号”的信息增益为：

$$Gain(D, \text{编号}) = Ent(D) - \sum_{v=1}^{17} \frac{1}{17} Ent(D^v) = 0.998$$

远大于其他候选属性

信息增益准则对可取值数目较多的属性有所偏好

表 4.1 西瓜数据集 2.0

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否



增益率 $\text{Gain_ratio}(D, a) = \frac{\text{Gain}(D, a)}{\text{IV}(a)}$, (4.3)

其中

$$\text{IV}(a) = - \sum_{v=1}^V \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|} \quad (4.4)$$

称为属性 a 的“固有价值” (intrinsic value) [Quinlan, 1993]. 属性 a 的可能取值数目越多(即 V 越大), 则 $\text{IV}(a)$ 的值通常会越大.

增益率准则对可取值数目较少的属性有所偏好

著名的C4.5决策树算法综合了信息增益准则和信息率准则的特点:
先从候选划分属性中找出信息增益高于平均水平的属性, 再从中选择增益率最高的。



基尼值

$$\begin{aligned} \text{Gini}(D) &= \sum_{k=1}^{|\mathcal{Y}|} \sum_{k' \neq k} p_k p_{k'} \\ &= 1 - \sum_{k=1}^{|\mathcal{Y}|} p_k^2. \end{aligned} \quad (4.5)$$

直观来说, $\text{Gini}(D)$ 反映了从数据集 D 中随机抽取两个样本, 其类别标记不一致的概率. 因此, $\text{Gini}(D)$ 越小, 则数据集 D 的纯度越高.

基尼指数

$$\text{Gini_index}(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Gini}(D^v). \quad (4.6)$$

于是, 我们在候选属性集合 A 中, 选择那个使得划分后基尼指数最小的属性作为最优划分属性, 即 $a_* = \arg \min_{a \in A} \text{Gini_index}(D, a)$.

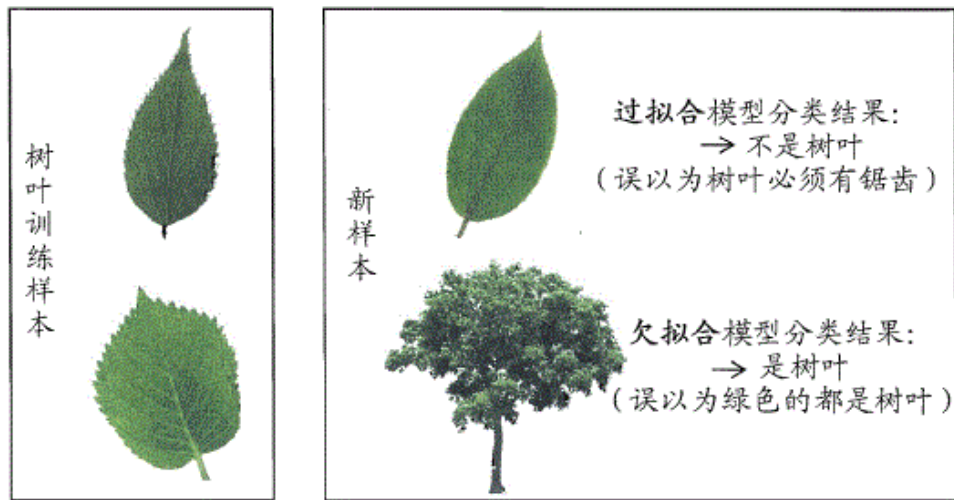
著名的CART决策树算法



- 决策树基本算法
- 决策树的剪枝
- 回归决策树
- 属性缺失
- 多变量决策树



- **过拟合**：学习器学习能力过于强大，把训练样本自身的一些特点当作了所有潜在样本都会具有的一般性质，导致泛化性能下降。
- **欠拟合**：学习器学习能力低下，对训练样本的一般性质尚未学好。



过拟合无法彻底避免，
只能做到“缓解”。

图 2.1 过拟合、欠拟合的直观类比

剪枝：通过主动去掉一些分支来降低过拟合的风险。

决策树的剪枝策略 { 预剪枝
后剪枝

预剪枝：在决策树生成过程中，对每个结点在划分前先进行估计，若当前结点的划分不能带来决策树泛化性能提升，则停止划分并将当前结点标记为叶结点

后剪枝：先从训练集生成一棵完整的决策树，然后自底向上地对非叶结点进行考察，若将该结点对应的子树替换为叶结点能带来决策树泛化性能提升，则将该子树替换为叶结点。

将数据集D划分为两个互斥的集合：训练集S和验证集T

$$D = S \cup T \quad \text{且} \quad S \cap T = \emptyset$$

表 4.2 西瓜数据集 2.0 划分出的训练集(双线上部)与验证集(双线下部)

训练集

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

验证集

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否

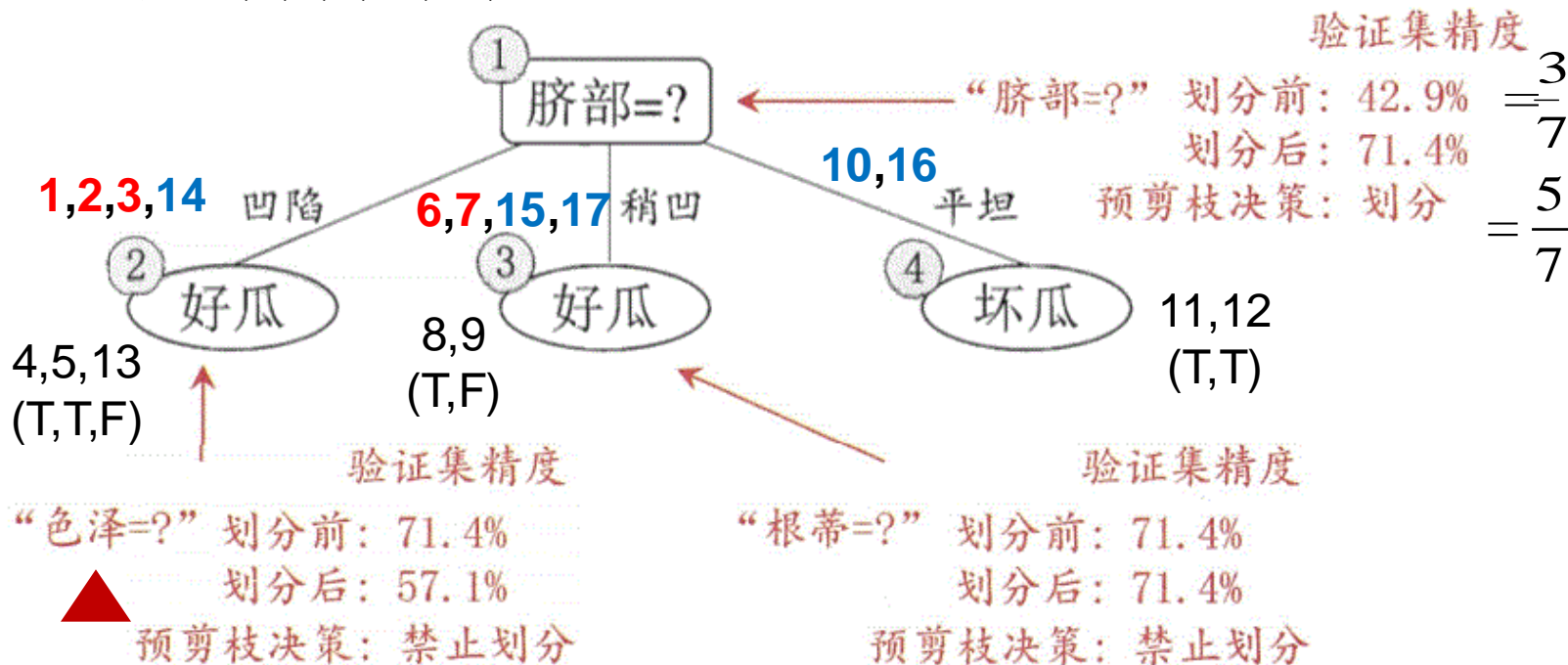
编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否

预剪枝

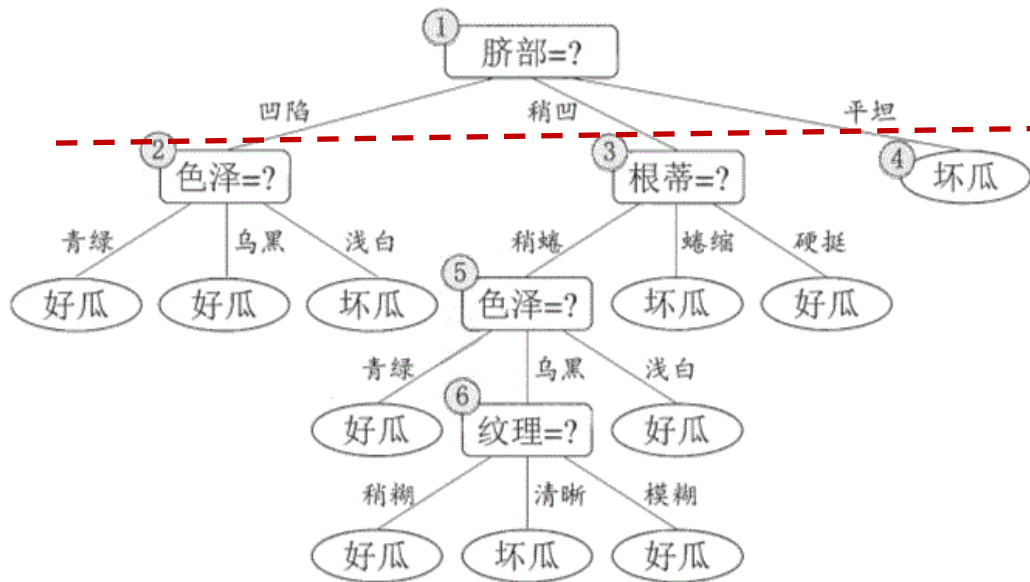
训练集：好瓜 坏瓜 1,2,3,6,7,10,14,15,16,17

验证集：4,5,8,9,11,12,13

评估精度：正确分类的
样本占有所有样本的比例



未剪枝的决策树



预剪枝使得决策树的很多分支都没有“展开”

优点:

- 降低过拟合的风险
- 减少了训练时间开销和测试时间开销

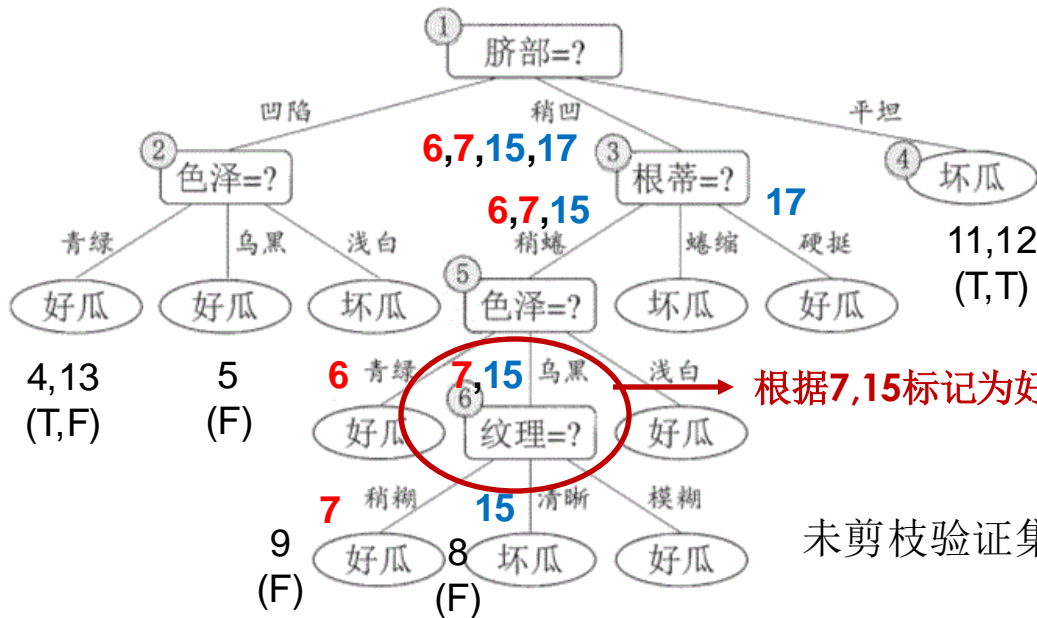
不足: 基于“贪心”本质禁止某些分支展开, 带来了欠拟合的风险

后剪枝：先从训练集**生成一棵完整的决策树**，然后**自底向上**地对非叶结点进行考察，若将该结点对应的**子树替换为叶结点**能带来决策树泛化性能提升，则将该子树替换为叶结点。

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否

训练集：好瓜 坏瓜 1,2,3,6,7,10,14,15,16,17

验证集：4,5,8,9,11,12



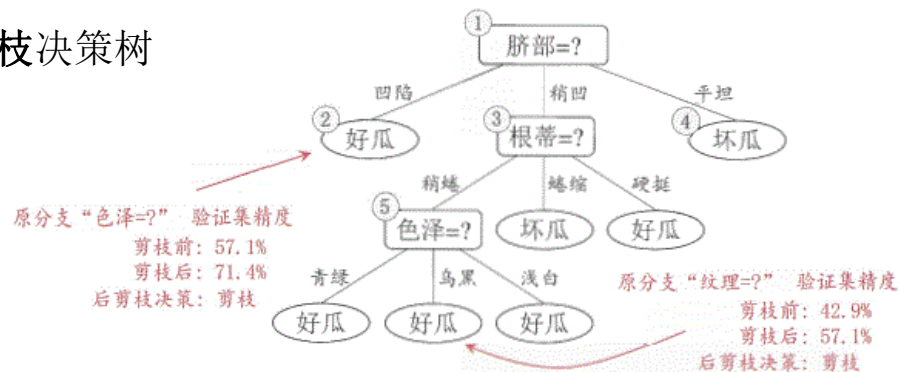
减去结点⑥
验证集变为： $\frac{4}{7}$ 8,9 (T,F)

考察结点顺序：
⑥⑤②③①

根据7,15标记为好瓜

未剪枝验证集精度： $\frac{3}{7}$

后剪枝决策树

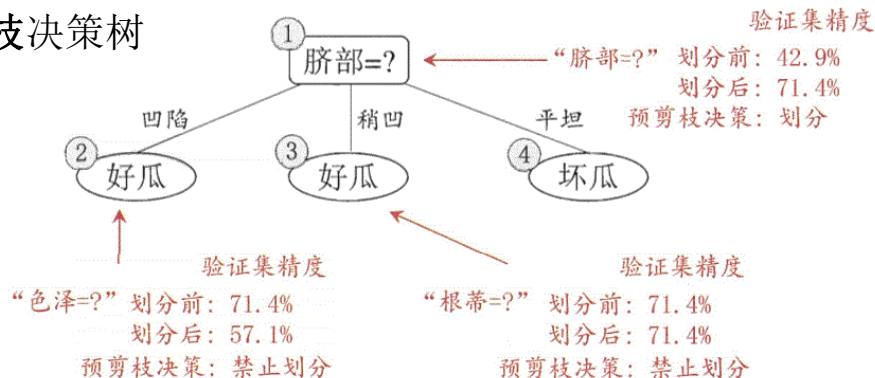


- 保留了更多的分支
- 欠拟合风险很小
- 泛化能力优于预剪枝决策树

- 训练时间开销比未减枝和预剪枝决策树大得多

- 生产完全决策树
- 所有非叶节点逐一考察

预剪枝决策树





- 决策树基本算法
- 决策树的剪枝
- 回归决策树
- 属性缺失
- 多变量决策树



表 4.3 西瓜数据集 3.0

编号	色泽	根蒂	敲声	纹理	脐部	触感	密度	含糖率	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	0.697	0.460	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	0.774	0.376	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	0.634	0.264	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	0.608	0.318	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	0.556	0.215	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	0.403	0.237	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	0.481	0.149	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	0.437	0.211	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	0.666	0.091	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	0.243	0.267	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	0.245	0.057	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	0.343	0.099	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	0.639	0.161	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	0.657	0.198	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	0.360	0.370	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	0.593	0.042	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	0.719	0.103	否

离散属性：脐部 根蒂 色泽...

连续属性：密度 含糖率...

连续属性离散化技术：二分法 C4.5决策树算法

样本集 D

连续属性 a ，有 n 个不同的取值，将 n 个取值从小到大排序：

$$\{a^1, a^2, \dots, a^n\}$$

划分点 t （数值）将 D 划分为两个子集 D_t^- 和 D_t^+

$$\underbrace{\{a^1, a^2, \dots, a^i\}}_{D_t^+} \quad \bigg| \quad \underbrace{\{a^{i+1}, \dots, a^n\}}_{D_t^-}$$

D_t^+

t

D_t^-

小于 t 的一类，大于 t 的一类

显然，对相邻的属性取值 a^i 和 a^{i+1} 来说， t 在区间 $[a^i, a^{i+1})$ 中取任意值所产生的划分结果都相同



可考察包含 $n-1$ 个元素的候选划分点集合

$$T_a = \left\{ \frac{a^i + a^{i+1}}{2} \mid 1 \leq i \leq n-1 \right\}, \quad (4.7)$$

即把区间 $[a^i, a^{i+1})$ 的中位点 $\frac{a^i + a^{i+1}}{2}$ 作为候选划分点. 然后, 我们就可像离散属性值一样来考察这些划分点, 选取最优的划分点进行样本集合的划分. 例如, 可对式(4.2)稍加改造:

$$\begin{aligned} \text{Gain}(D, a) &= \max_{t \in T_a} \text{Gain}(D, a, t) \\ &= \max_{t \in T_a} \text{Ent}(D) - \sum_{\lambda \in \{-, +\}} \frac{|D_t^\lambda|}{|D|} \text{Ent}(D_t^\lambda), \end{aligned} \quad (4.8)$$

其中 $\text{Gain}(D, a, t)$ 是样本集 D 基于划分点 t 二分后的信息增益. 于是, 我们就可选择使 $\text{Gain}(D, a, t)$ 最大化的划分点.



D_t^-	密度	好瓜
	0.243	否
	0.245	否
	0.343	否
	0.360	否
	0.403	是1
	0.437	是2
	0.481	是3
D_t^+	0.556	是4
	0.593	否
	0.680	是5
	0.634	是6
	0.639	否
	0.657	否
	0.666	否
	0.697	是7
	0.719	否
	0.774	是8

$t = 0.381$

根结点包含17个训练样本，密度有17个不同取值

候选划分点集合包含16个候选值

每一个划分点能得到一个对应的信息增益，**取最大值**

$$\begin{aligned} \text{Gain}(D, a) &= \max_{t \in T_a} \text{Gain}(D, a, t) \\ &= \max_{t \in T_a} \text{Ent}(D) - \sum_{\lambda \in \{-, +\}} \frac{|D_t^\lambda|}{|D|} \text{Ent}(D_t^\lambda), \end{aligned} \quad (4.8)$$

根结点的信息熵仍为： $\text{Ent}(D) = 0.998$

$$\text{Ent}(D_t^-) = -\left(\frac{0}{4} \log_2 \frac{0}{4} + \frac{4}{4} \log_2 \frac{4}{4}\right) = 0$$

$$\text{Ent}(D_t^+) = -\left(\frac{8}{13} \log_2 \frac{8}{13} + \frac{5}{13} \log_2 \frac{5}{13}\right) = 0.961$$

$\text{Gain}(D, \text{密度}, 0.381)$

$$= \text{Ent}(D) - \left[\frac{4}{17} \times \text{Ent}(D_t^-) + \frac{13}{17} \times \text{Ent}(D_t^+)\right] = 0.263$$



$\text{Gain}(D, \text{色泽}) = 0.109$; $\text{Gain}(D, \text{根蒂}) = 0.143$;
 $\text{Gain}(D, \text{敲声}) = 0.141$; $\text{Gain}(D, \text{纹理}) = 0.381$;
 $\text{Gain}(D, \text{脐部}) = 0.289$; $\text{Gain}(D, \text{触感}) = 0.006$;
 $\text{Gain}(D, \text{密度}) = 0.262$; $\text{Gain}(D, \text{含糖率}) = 0.349$.

选择“纹理”作为根结点划分属性

与离散属性不同，若当前结点划分属性为连续属性，**该连续属性还可被再次选作后代结点的最优划分属性。**

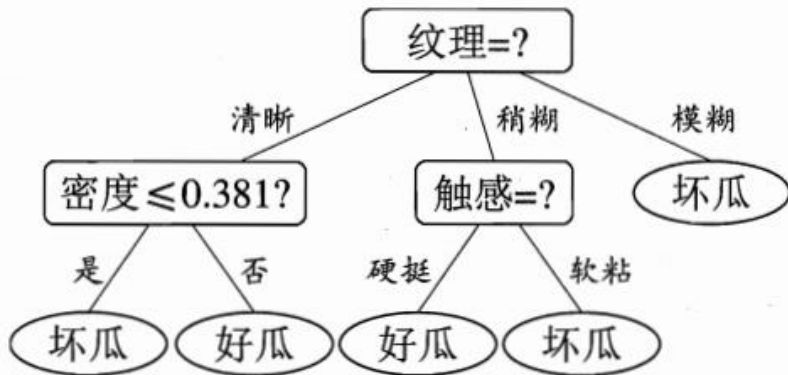


图 4.8 在西瓜数据集 3.0 上基于信息增益生成的决策树



- 决策树基本算法
- 决策树的剪枝
- 回归决策树
- 属性缺失
- 多变量决策树



表 4.4 西瓜数据集 2.0 α

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	-	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	-	是
3	乌黑	蜷缩	-	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	-	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	-	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	-	稍凹	硬滑	是
9	乌黑	-	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	-	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	-	否
12	浅白	蜷缩	-	模糊	平坦	软粘	否
13	-	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	-	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	-	沉闷	稍糊	稍凹	硬滑	否

- 现实任务中，尤其在属性数目较多时，存在大量样本出现缺失值。
- 出于成本和隐私的考虑，难以获取全部属性



- 问题 1. 属性值缺失时，如何进行划分属性选择？如何计算信息增益
- 问题 2. 给定划分属性，若样本在该属性上的值缺失，如何对样本进行划分？
对于缺失属性值的样本如何将它从父结点划分到子结点中

D : 训练集 D° : 训练集中 在属性 a 上没有缺失值的样本子集

D°_k : D° 被属性 a 划分后的样本子集 D°_k : D° 中属于第 k 类的样本子集

假定我们为每个样本 x 赋予一个权重 w_x , 并定义

$$\rho = \frac{\sum_{x \in \tilde{D}} w_x}{\sum_{x \in D} w_x}, \quad \rho: \text{无缺失值样本所占比例}$$

$$\tilde{p}_k = \frac{\sum_{x \in \tilde{D}_k} w_x}{\sum_{x \in \tilde{D}} w_x} \quad (1 \leq k \leq |\mathcal{Y}|), \quad \tilde{p}_k: \text{无缺失值样本中第 } k \text{ 类所占比例}$$

$$\tilde{r}_v = \frac{\sum_{x \in \tilde{D}^v} w_x}{\sum_{x \in \tilde{D}} w_x} \quad (1 \leq v \leq V). \quad (4.11)$$

$$\tilde{p}_v^a: \text{无缺失值样本中在属性 } a \text{ 上取值 } a^v \text{ 的样本所占比例}$$



对于问题1：直观地看，对属性 a , ρ 表示无缺失值样本所占的比例, \tilde{p}_k 表示无缺失值样本中第 k 类所占的比例, \tilde{r}_v 则表示无缺失值样本中在属性 a 上取值 a^v 的样本所占的比例. 显然, $\sum_{k=1}^{|\mathcal{Y}|} \tilde{p}_k = 1$, $\sum_{v=1}^V \tilde{r}_v = 1$.

基于上述定义, 我们可将信息增益的计算式(4.2)推广为

$$\begin{aligned} \text{Gain}(D, a) &= \rho \times \text{Gain}(\tilde{D}, a) \quad \text{无缺失值的样本子集 } \tilde{D} \text{ 上的信息增益} \\ &= \rho \times \left(\text{Ent}(\tilde{D}) - \sum_{v=1}^V \tilde{r}_v \text{Ent}(\tilde{D}^v) \right), \end{aligned} \quad (4.12)$$

其中由式(4.1), 有

$$\text{Ent}(\tilde{D}) = - \sum_{k=1}^{|\mathcal{Y}|} \tilde{p}_k \log_2 \tilde{p}_k .$$



■ 对于问题2：对于有缺失值的样本如何将它从父结点划分到子结点中

若样本 x 在划分属性 a 上的取值已知，则将 x 划入与其取值对应的子结点，且样本权值在子结点中保持为 w_x

若样本 x 在划分属性 a 上的取值未知，则将 x 同时划入所有子结点，且样本权值在子结点中调整为 $\tilde{r}_v w_x$ ，就是让同一个样本以不同的概率划入不同的子结点中

其中， w_x 是为每个样本 x 赋予的一个权重

\tilde{r}_v : 无缺失值样本中在属性 a 上取值 a^v 的样本所占比例



- 决策树基本算法
- 决策树的剪枝
- 回归决策树
- 属性缺失
- 多变量决策树



每个属性 ~ 坐标空间中的一个坐标轴
 d个属性描述的样本 ~ d维空间中的一个数据点
 对样本分类 ~ 在坐标空间中寻找不同类样本之间的分类边界

决策树形成的分类边界的明显特点：轴平行，分类边界由若干个与坐标轴平行的分段组成。

优点： 学习结果解释性强，每个划分都对应一个属性取值

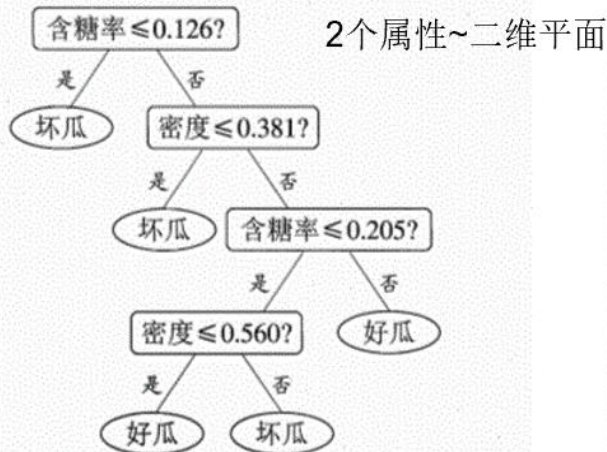


图 4.10 在西瓜数据集 3.0α 上生成的决策树

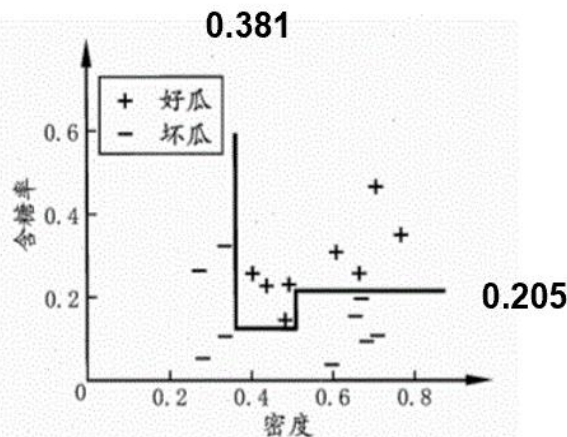
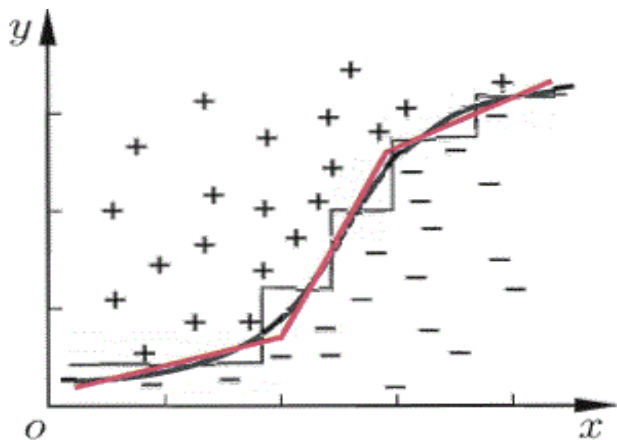


图 4.11 图 4.10 决策树对应的分类边界



决策树对复杂分类使用分段近似

不足：如图 4.12 所示；此时的决策树会相当复杂，由于要进行大量的属性测试，预测时间开销会很大。

若能使用斜的划分边界，如图 4.12 中红色线段所示，则决策树模型将大为简化。“多变量决策树” (multivariate decision tree) 就是能实现这样的“斜划分”甚至更复杂划分的决策树。



	单变量决策树	多变量决策树
非叶结点的属性测试	一个属性 (最优划分属性)	属性的线性组合 (线性分类器)
算法第8行	寻找最优划分属性	建立合适的线性分类器

$$\sum_{i=1}^d w_i a_i = t \quad \leftarrow \text{这条斜线}$$

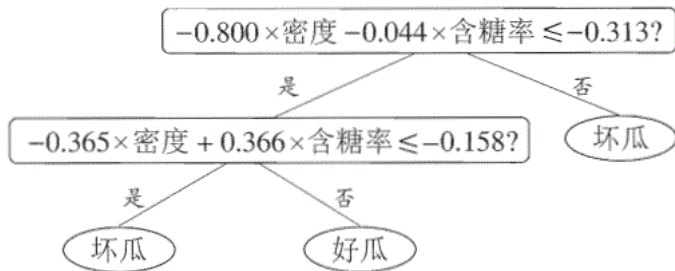


图 4.13 在西瓜数据集 3.0α 上生成的多变量决策树

w_i, t 可以从该结点所含的样本集D
和属性集A上学得

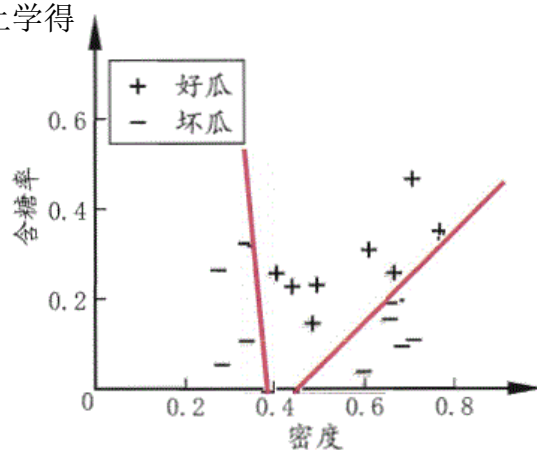


图 4.14 图 4.13 多变量决策树对应的分类边界

- 决策树是一种**基于规则的方法**，嵌套规则进行预测。根据判断结果进入一个分支，反复执行这个操作直到叶子节点，得到预测结果。这些规则通过训练得到，而非人工制定。
- 本质上决策树是通过一系列规则对数据进行分类的过程。首先对数据进行处理，利用归纳算法生成可读的规则和决策树，然后使用决策对新数据进行分析。
- 决策树的优点
 - 推理过程容易理解，决策推理过程可以表示成If Then 形式；
 - 推理过程完全依赖于属性变量的取值特点；
 - 可自动忽略目标变量没有贡献的属性变量，也为判断属性变量的重要性，减少变量的数目提供参考。

- 基本算法3种递归返回情形、第8行
- 3种度量结点“纯度”的指标：
 - 信息增益 ID3
 - 增益率 C4.5
 - 基尼指数 CART
- 决策树剪枝
 - 预剪枝
 - 后剪枝
- 决策树属性缺失