

# 第五章 分类

## 5.1 模型评估和性能度量

## 5.2 决策树

### 5.2.1 决策树概况

### 5.2.2 决策树构造

### 5.2.3 剪枝

## 5.3 贝叶斯分类

## 5.4 k最近邻分类

## 5.5 组合分类

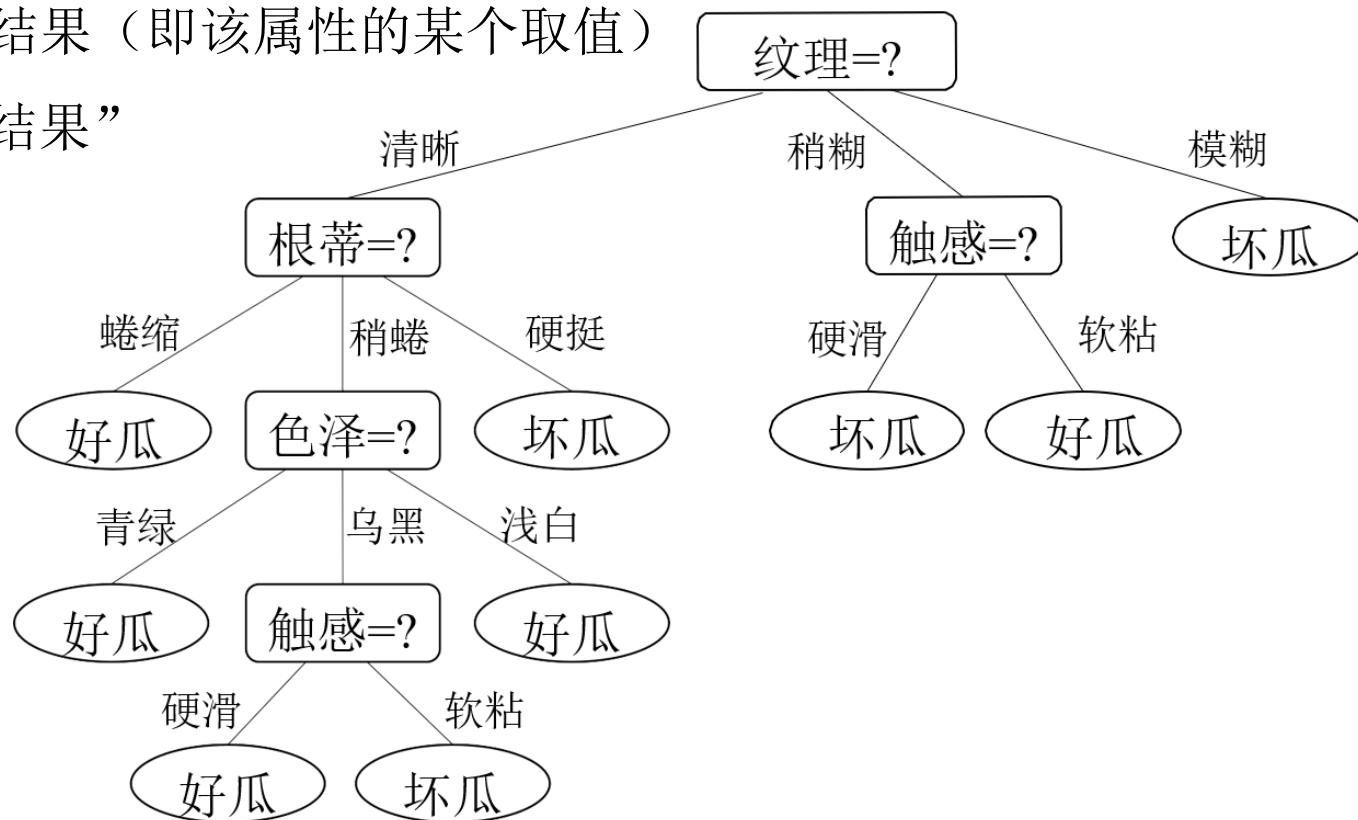
# 决策树模型-概况

决策树基于“树”结构进行决策

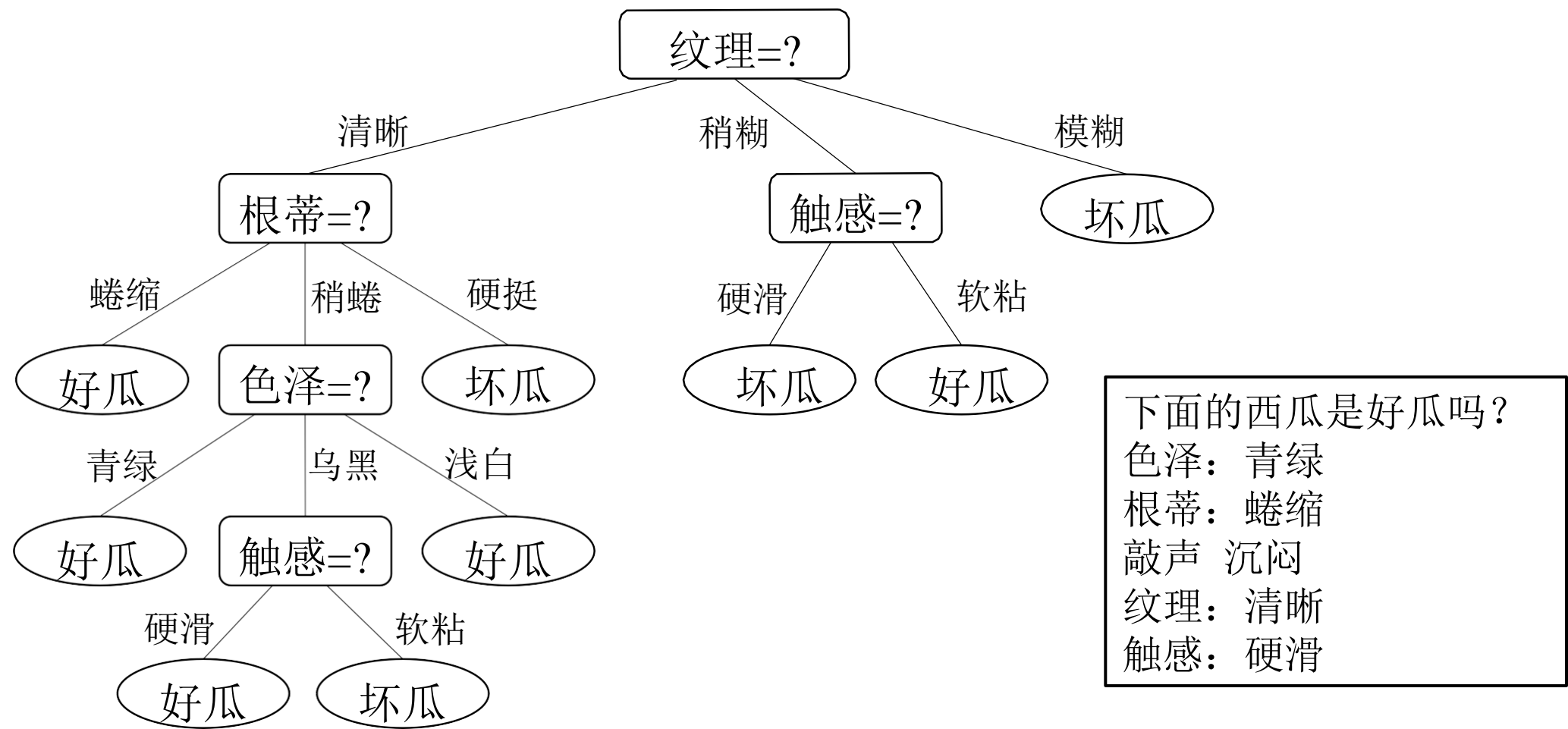
- ❑ 每个“内部结点”对应于某个属性上的“测试”(test)
- ❑ 每个分支对应于该测试的一种可能结果（即该属性的某个取值）
- ❑ 每个“叶结点”对应于一个“预测结果”

**学习过程：**通过对训练样本的分析来确定“划分属性”（即内部结点所对应的属性）。

**预测过程：**将测试示例从根结点开始，沿着划分属性所构成的“判定测试序列”下行，直到叶结点。



# 基于决策树的预测



基于西瓜数据集构建的决策树

# 决策树的优点以及相关算法

优点：

- 直观，具有很好的可解释性；
- 不需要任何先验假设。

发展历史及主要算法：

- 使决策树受到关注、成为机器学习主流技术的算法：ID3
- 最常用的决策树算法：C4.5
- 用于分类和回归任务的决策树算法：CART
- 基于决策树的最强大算法：RF (Random Forest) — 集成学习

## 决策树构造：最佳属性度量

策略：“分而治之” (divide-and-conquer) 从根至叶的递归过程

在每个中间结点寻找一个“划分” (split or test) 属性

**关键：** 寻找当前最具有辨别力 (discriminative) 的特征。

编号	色泽	敲声	好瓜
1	青绿	清脆	是
2	青绿	沉闷	是
3	乌黑	沉闷	否
4	乌黑	清脆	否

哪个特征更有利于分类？

一个特征的每个取值对应的类别越纯则该辨别力越大

## 度量一：信息增益 (information gain)

信息熵 (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  $p_k$ ，则  $D$  的信息熵定义为：

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

计算信息熵时约定：若  $p = 0$ ，则  $p \log_2 p = 0$ 。

$\text{Ent}(D)$  的值越小，则  $D$  的纯度越高

$\text{Ent}(D)$  的最小值为 0，最大值为  $\log_2 |\mathcal{Y}|$ 。

信息增益直接以信息熵为基础，计算当前划分对信息熵所造成的变化

## 度量一：信息增益 (information gain)

离散属性  $a$  的取值：  $\{a^1, a^2, \dots, a^V\}$

$D^v$ :  $D$  中在  $a$  上取值  $= a^v$  的样本集合

以属性  $a$  对数据集  $D$  进行划分所获得的信息增益为：

$$\text{Gain}(D, a) = \underbrace{\text{Ent}(D)}_{\text{划分前的信息熵}} - \sum_{v=1}^V \underbrace{\frac{|D^v|}{|D|}}_{\text{第 } v \text{ 个分支的权重, 样本越多越重要}} \underbrace{\text{Ent}(D^v)}_{\text{划分后的信息熵 (加权求和)}}$$

ID3中用信息增益进行属性选择

一个例子

该数据集包含17个训练样例，属于2个类别，即  $|\mathcal{Y}| = 2$ ，其中正例占  $p_1 = \frac{8}{17}$ ，反例占  $p_2 = \frac{9}{17}$ ，

则根结点的信息熵为

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

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



## 一个例子 (续)

□ 以属性“色泽”为例，其对应的3个数据子集分别为 $D^1$  (色泽=青绿)， $D^2$  (色泽=乌黑)， $D^3$  (色泽=浅白)。

□ 子集 $D^1$  包含编号为  $\{1, 4, 6, 10, 13, 17\}$  的6个样例，即 $|D^1| = 6$ ，其中正例占 $p_1 = \frac{3}{6}$ ，反例占 $p_2 = \frac{3}{6}$ ， $D^2$ 、 $D^3$  同理。3个结点的信息熵为：

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

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

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

□ 属性“色泽”的信息增益为

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

## 一个例子 (续)

□ 类似的，其他属性的信息增益为

$$\text{Gain}(D, \text{根蒂}) = 0.143$$

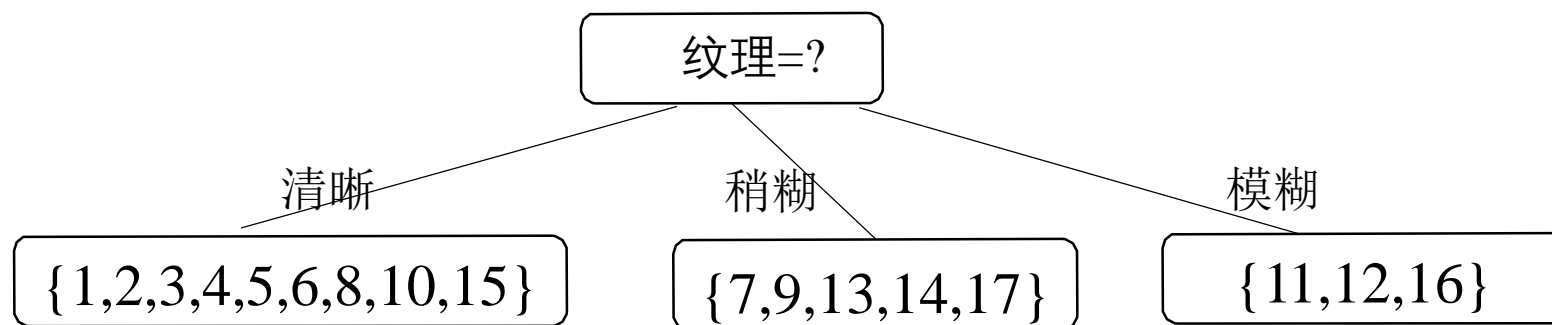
$$\text{Gain}(D, \text{敲声}) = 0.141$$

$$\text{Gain}(D, \text{纹理}) = 0.381$$

$$\text{Gain}(D, \text{脐部}) = 0.289$$

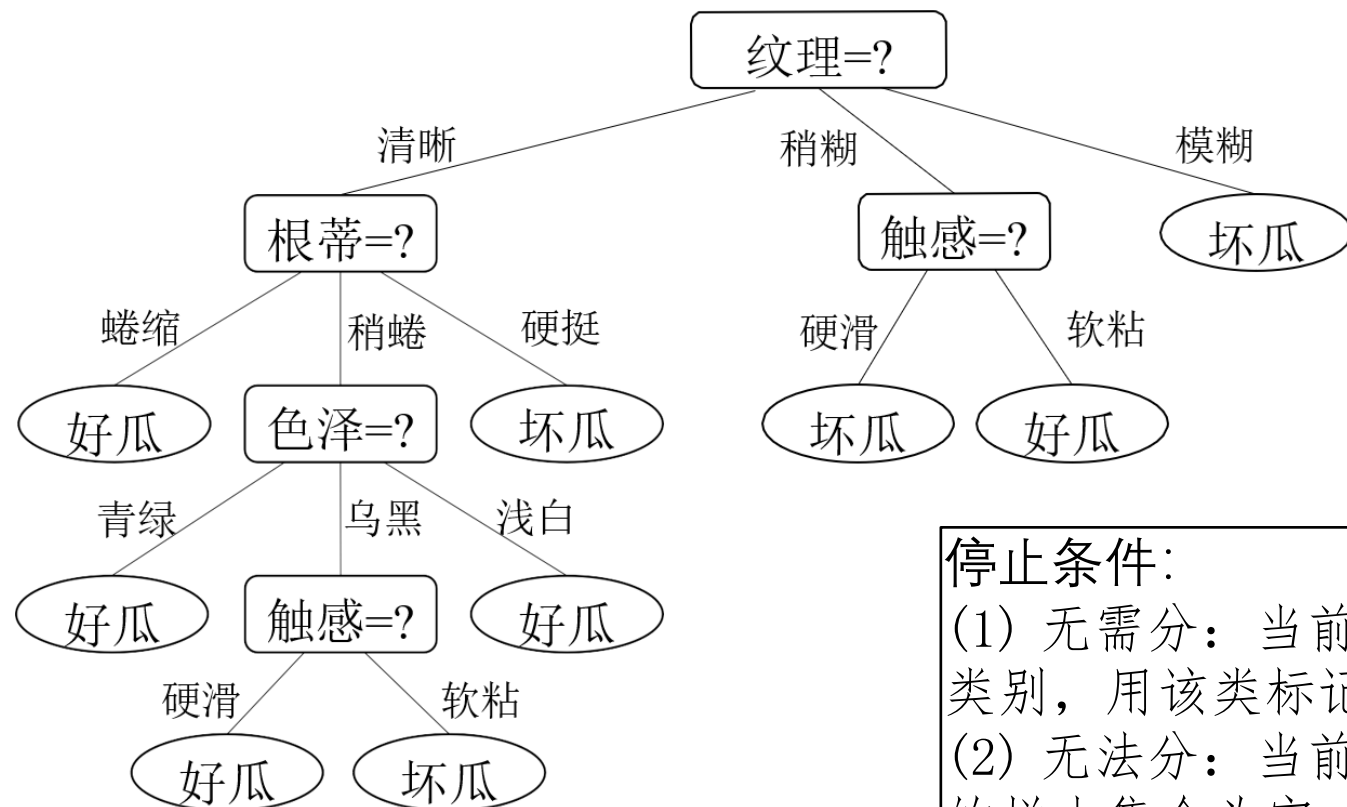
$$\text{Gain}(D, \text{触感}) = 0.006$$

□ 显然，属性“纹理”的信息增益最大，其被选为当前划分属性



## 一个例子 (续)

对每个分支结点做进一步划分，最终得到决策树



### 停止条件:

- (1) 无需分: 当前结点包含的样本全属于同一类别, 用该类标记叶子节点;
- (2) 无法分: 当前属性集为空或当前结点包含的样本集合为空, 用包含最多样本的类标记;

## 度量二： 增益率 (gain ratio)

信息增益：对可取值数目较多的属性有所偏好

有明显弱点，例如：考虑将“编号”作为一个属性

增益率：  $\text{Gain\_ratio}(D, a) = \frac{\text{Gain}(D, a)}{\text{IV}(a)}$  C4.5算法中使用信息增益

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

属性  $a$  的可能取值数目越多 (即  $V$  越大), 则  $\text{IV}(a)$  的值通常就越大

启发式：先从候选划分属性中找出信息增益高于平均水平的，再从中选取增益率最高的。

### 度量三：基尼指数 (gini index)

$$\text{Gini}(D) = \sum_{k=1}^{|\mathcal{Y}|} \sum_{k' \neq k} p_k p_{k'}$$

反映了从  $D$  中随机抽取两个样例，其类别标记不一致的概率

$$= 1 - \sum_{k=1}^{|\mathcal{Y}|} p_k^2$$

$\text{Gini}(D)$  越小，数据集  $D$  的纯度越高

当  $p_k = \frac{1}{|\mathcal{Y}|}$  时， $\text{Gini}(D)$  最大，为  $1 - \frac{1}{|\mathcal{Y}|}$

总体趋势与信息熵类似。

属性  $a$  的基尼指数：

$$\text{Gini\_index}(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Gini}(D^v)$$

在候选属性集合中，选取那个使划分后基尼指数最小的属性

# 决策树构造的基本算法

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  
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
```

递归返回,  
情形(1)

递归返回,  
情形(2)

利用当前结点的后验分布

递归返回,  
情形(3)

将父结点的样本分布作为  
当前结点的先验分布

决策树算法的  
核心

# 剪枝

为了尽可能正确分类训练样本，有可能造成分支过多  
叶子节点对应的样本数目可能太少，不具有统计意义

导致过拟合！

可通过主动去掉一些分支来降低过拟合的风险

基本策略：

- 预剪枝 (pre-pruning): 提前终止某些分支的生长
- 后剪枝 (post-pruning): 生成一棵完全树，再“回头”剪枝

# 剪枝

研究表明：划分选择的各种准则虽然对决策树的尺寸有较大影响，但对泛化性能的影响很有限。

例如信息增益与基尼指数产生的结果，仅在约 2% 的情况下不同

剪枝方法和程度对决策树泛化性能的影响更为显著

在数据带噪时甚至可能将泛化性能提升 25%

剪枝 (pruning) 是决策树对付“过拟合”的主要手段！

剪枝的基本思路：比较剪枝前后决策树的优劣-验证集的准确率。  
剪枝后错误率是否降低/准确率是否提高来决定是否剪枝



# 数据集

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

剪枝过程中需评估剪枝前后决策树的优劣，现在我们假定使用“留出法”

方法一：预剪枝（自顶向下）

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

结点1：若不划分，则将其标记为叶结点，类别标记为训练样例中最多的类别，若选“好瓜”。验证集中，{4, 5, 8}被分类正确，得到验证集准确率为  $\frac{3}{7} \times 100\% = 42.9\%$

① 脐部=? ← 验证集准确率 “脐部=?” 划分前：42.9%

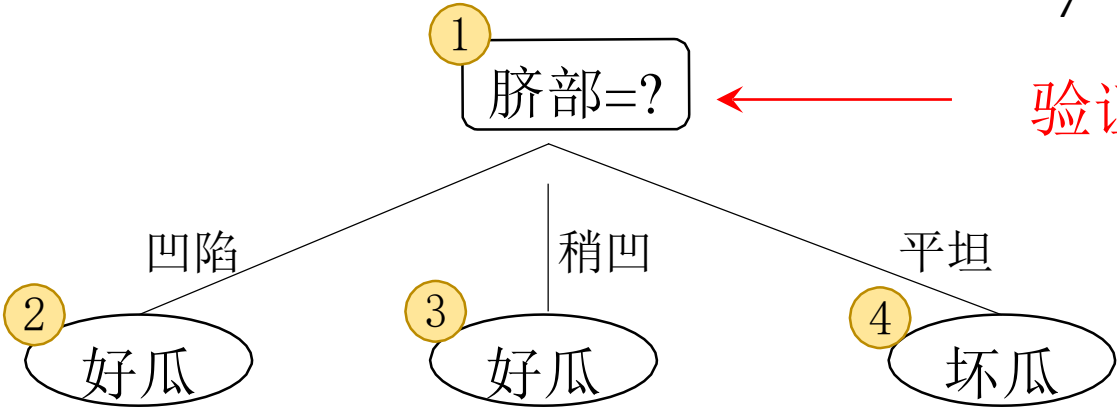
预剪枝 (续)

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

结点1：若划分，根据结点②,③,④ 的训练样例，将这3个结点分别标记为“好瓜”、“好瓜”、“坏瓜”。

此时，验证集中编号为{4, 5, 8, 11, 12}的样例被划分正确，验证集准确率为：  
 $\frac{5}{7} \times 100\% = 71.4\%$



验证集准确率 “脐部=?” 划分前：42.9%  
划分后：71.4%

预剪枝决策：划分

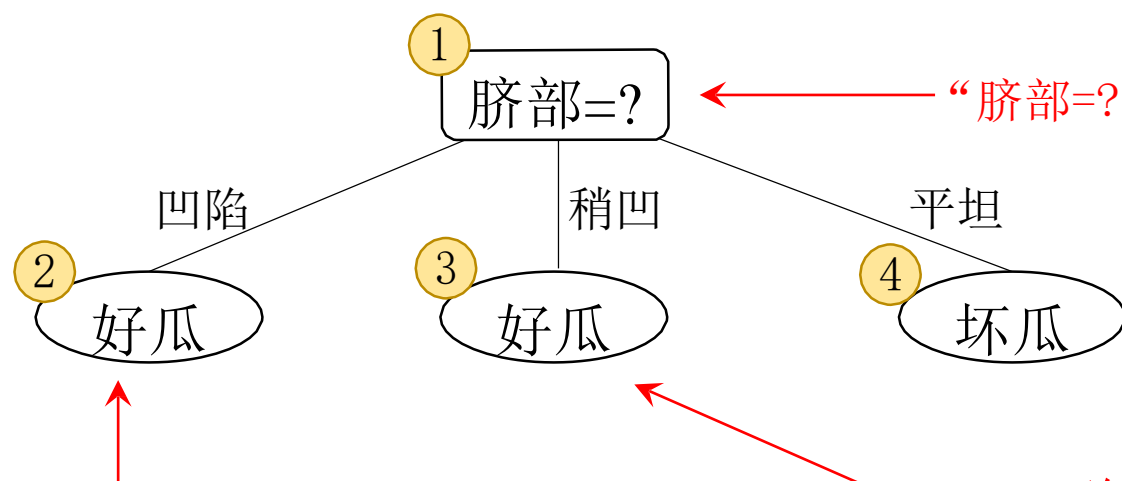
## 预剪枝 (续)

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

对结点②, ③, ④ 分别进行剪枝判断, 结点②和③都禁止划分, 结点④本身为叶子结点。最终得到仅有一层划分的决策树。

验证集准确率



“脐部=?” 划分前: 42.9%

划分后: 71.4%

预剪枝决策: 划分

验证集准确率

“色泽=?” 划分前: 71.4%

划分后: 57.1%

预剪枝决策: 禁止划分

验证集准确率

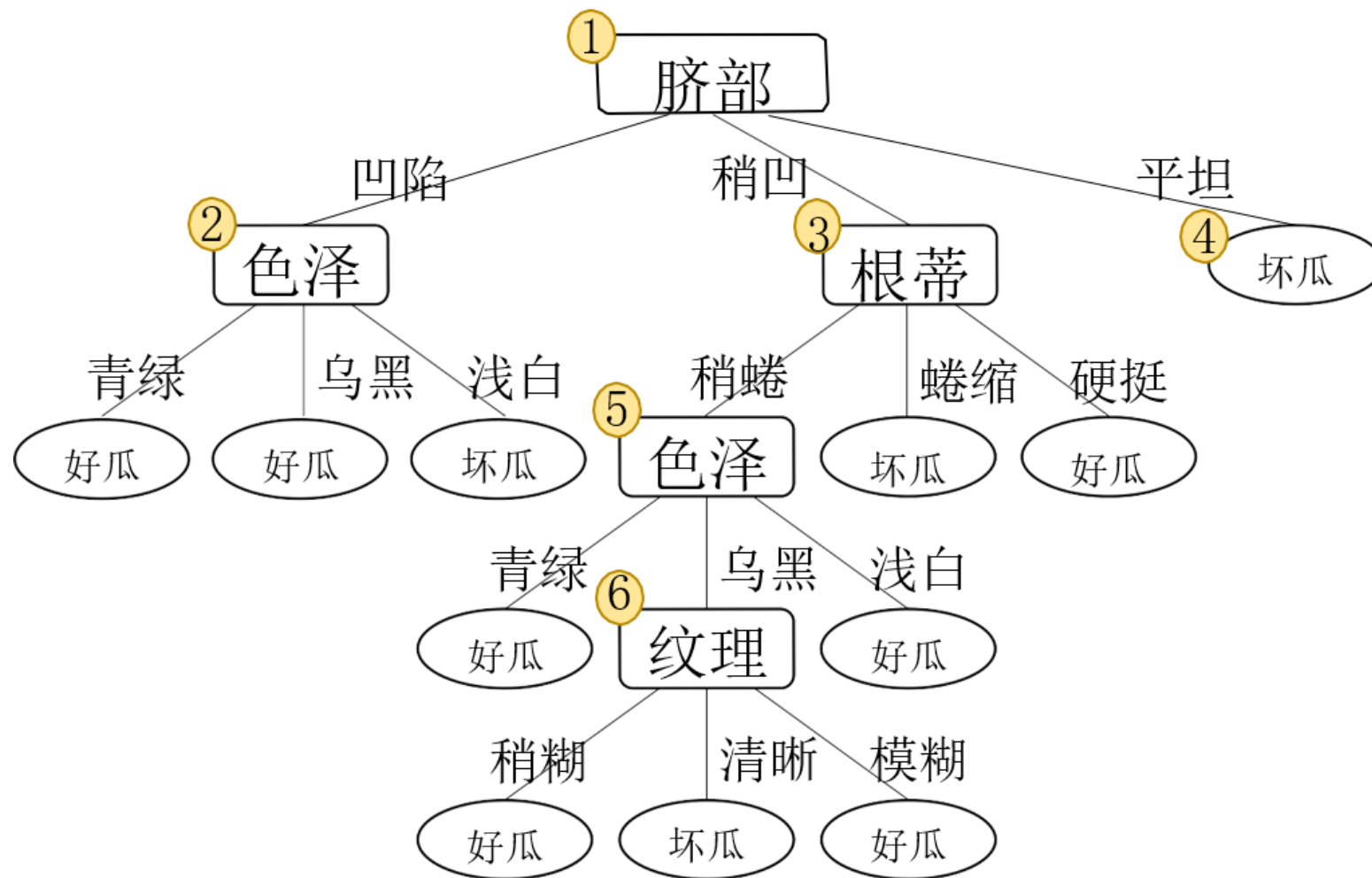
“根蒂=?” 划分前: 71.4%

划分后: 71.4%

预剪枝决策: 禁止划分

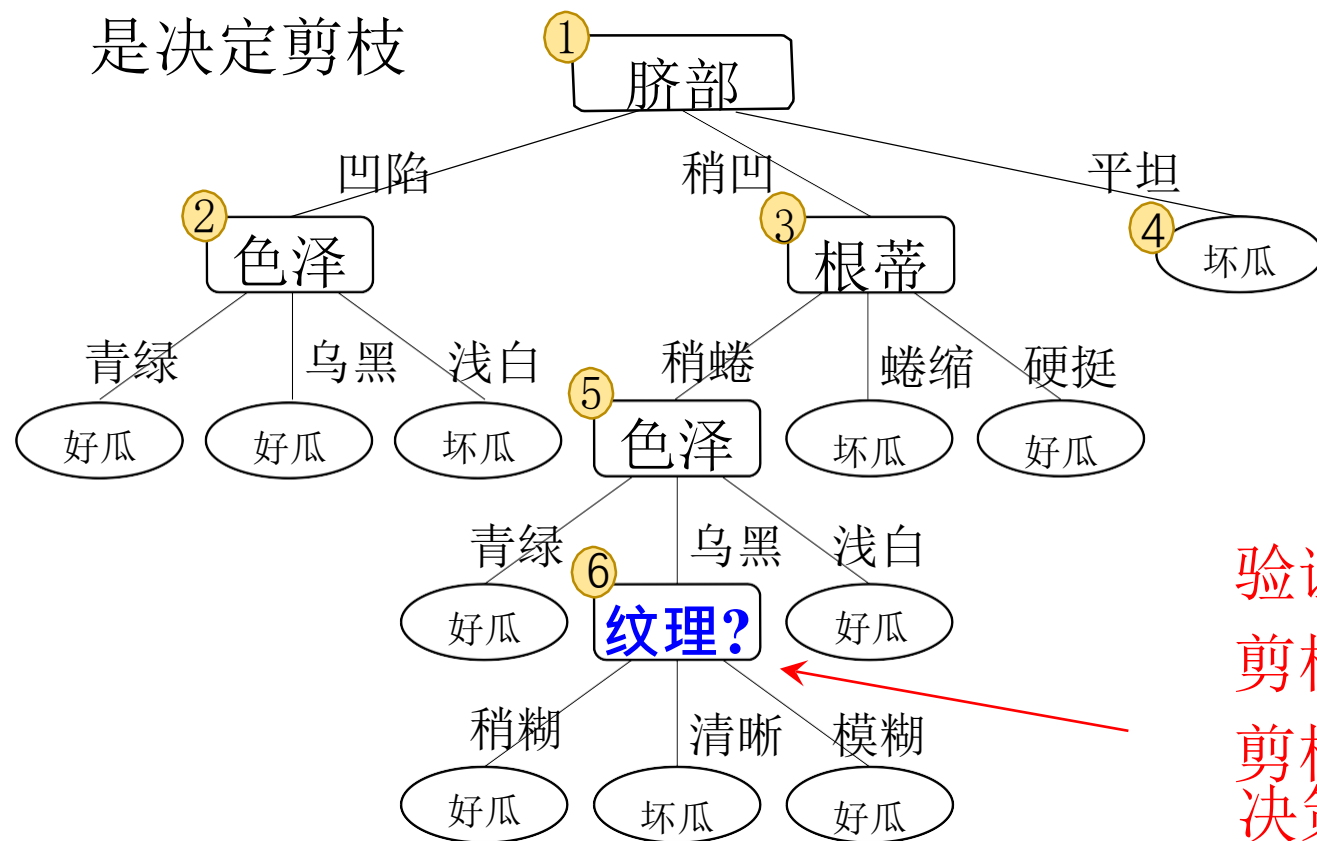
## 方法二：后剪枝

先生成一棵完整的决策树，其验证集准确率测得为 42.9%



## 后剪枝 (续)

自底向上。首先考虑结点 ⑥，若将其替换为叶结点，根据落在其上的训练样例 {7, 15} 将其标记为“好瓜”，测得验证集准确率提高至 57.1%，于是决定剪枝



验证集准确率

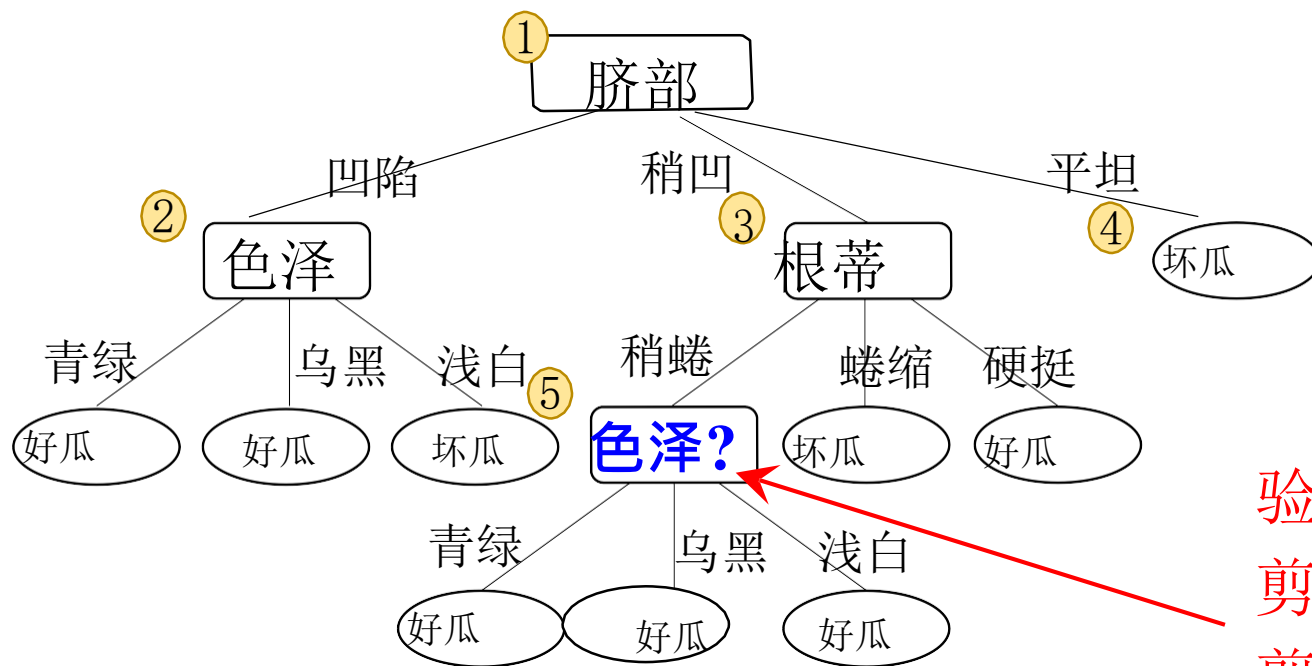
剪枝前: 42.9%

剪枝后: 57.1%

决策: 剪枝, 替换为叶子节点好瓜

## 后剪枝 (续)

然后考虑结点⑤，若将其替换为叶结点，根据落在其上的训练样例将其标记为“好瓜”，测得验证集精度仍为 **57.1%**，可以不剪枝



验证集准确率

剪枝前: 57.1%

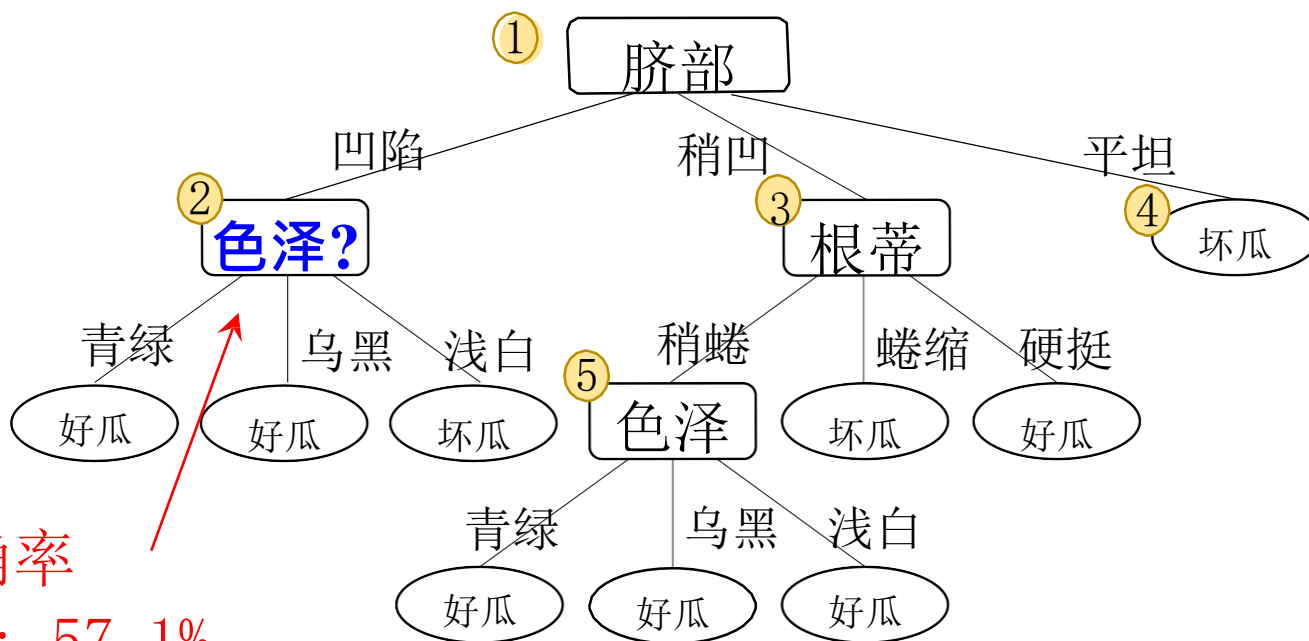
剪枝后: 57.1%

后剪枝决策: 不剪枝



## 后剪枝 (续)

对结点②，若将其替换为叶结点，根据落在其上的训练样例 {1, 2, 3, 14}，将其标记为“好瓜”，测得验证集准确率提升至 **71.4%**，决定剪枝。



验证集准确率

剪枝前: 57.1%

剪枝后: 71.4%

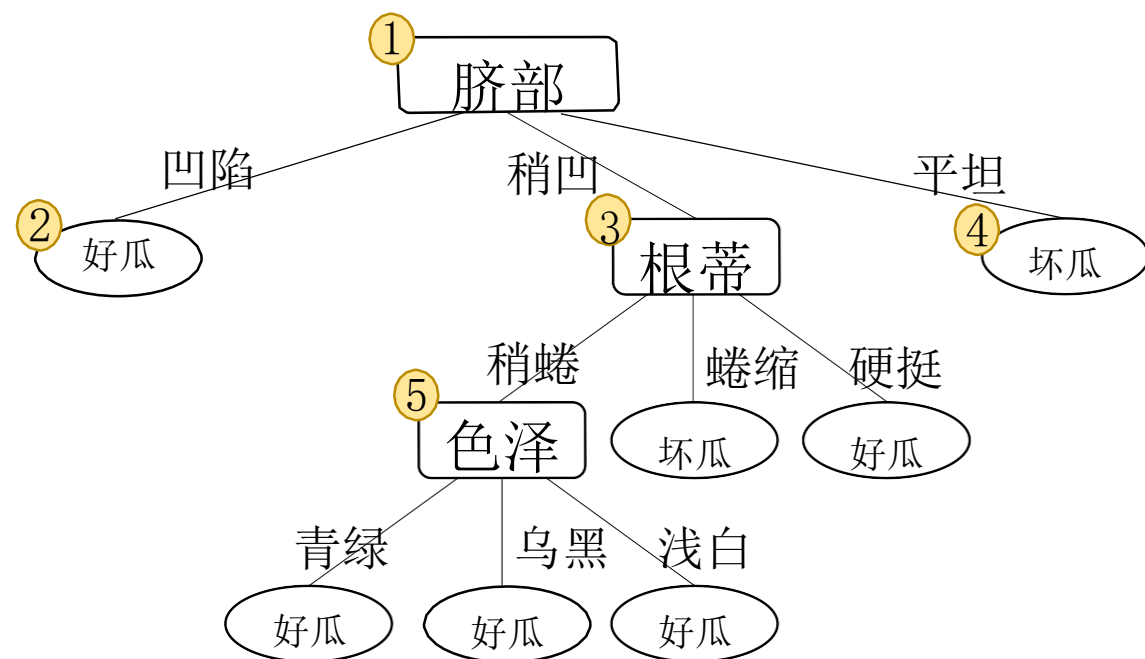
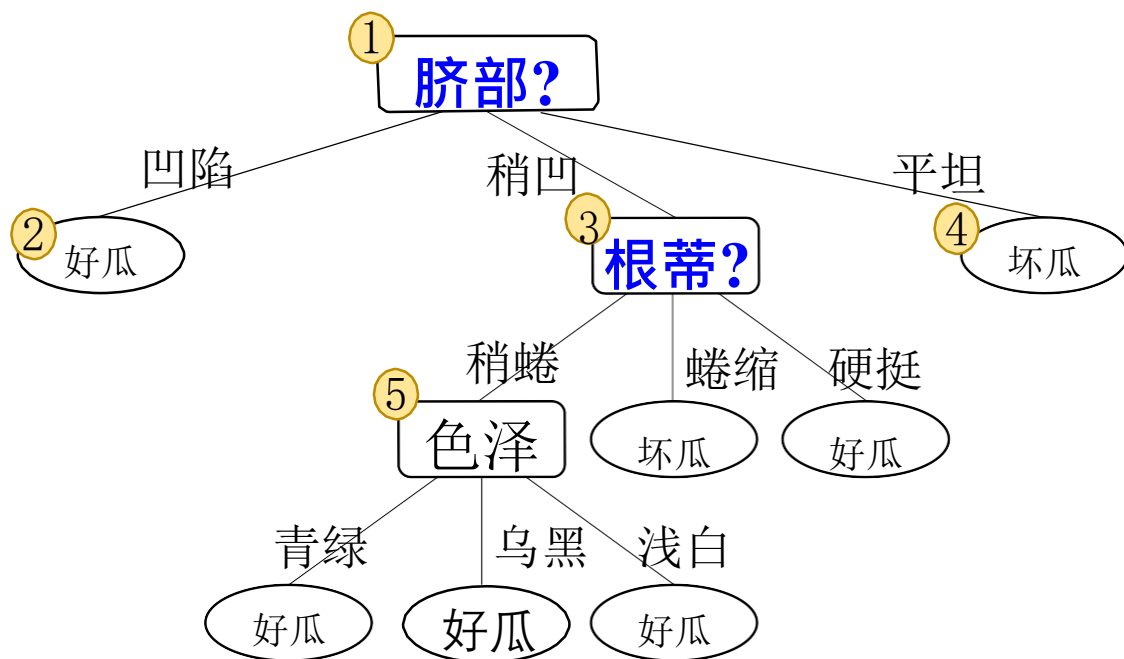
后剪枝决策: 剪枝, 替换为叶子节点好瓜



## 后剪枝 (续)

对节点 ③ 和 ① 先后替换为叶结点，均未测得验证集准确率提升，于是不剪枝。

最终，后剪枝得到的决策树：



# 预剪枝 vs. 后剪枝

## □ 时间开销:

- 预剪枝: 训练时间开销降低, 测试时间开销降低
- 后剪枝: 训练时间开销增加, 测试时间开销降低

## □ 过/欠拟合风险:

- 预剪枝: 过拟合风险降低, 欠拟合风险增加
- 后剪枝: 过拟合风险降低, 欠拟合风险基本不变

## □ 泛化性能: 后剪枝通常优于预剪枝

剪枝评判标准:

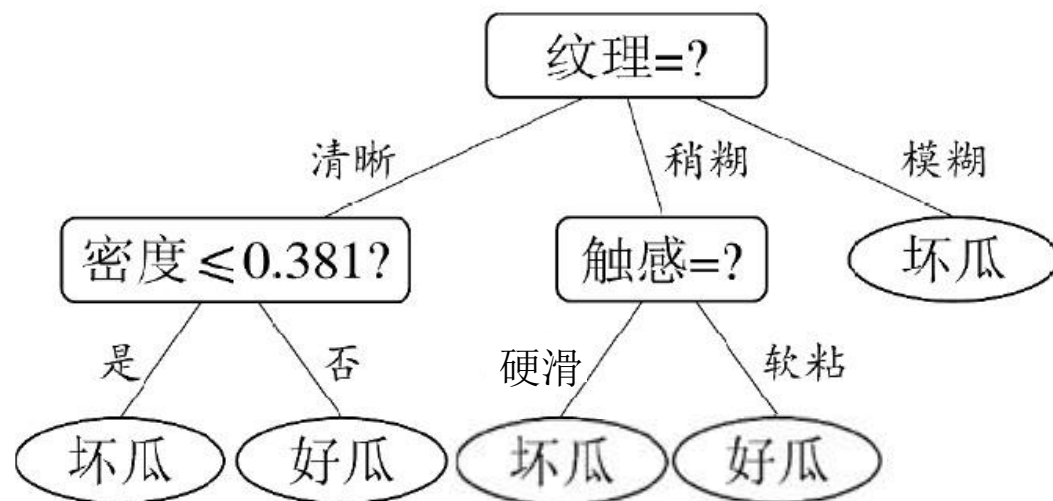
除了准确率, 还可以同时考虑模型的复杂度, 比如树叶节点的个数、对树编码需要的二进制位数等。

# 连续属性

## 基本思路：连续属性离散化

常见做法：二分法 (bi-partition): 一个连续属性转化成多个二值离散属性

- 把 $n$  个属性值排序，取相邻两个属性值的中间值作为候选划分点；
- 然后将每个候选划分点当做一个（二值）离散属性处理；

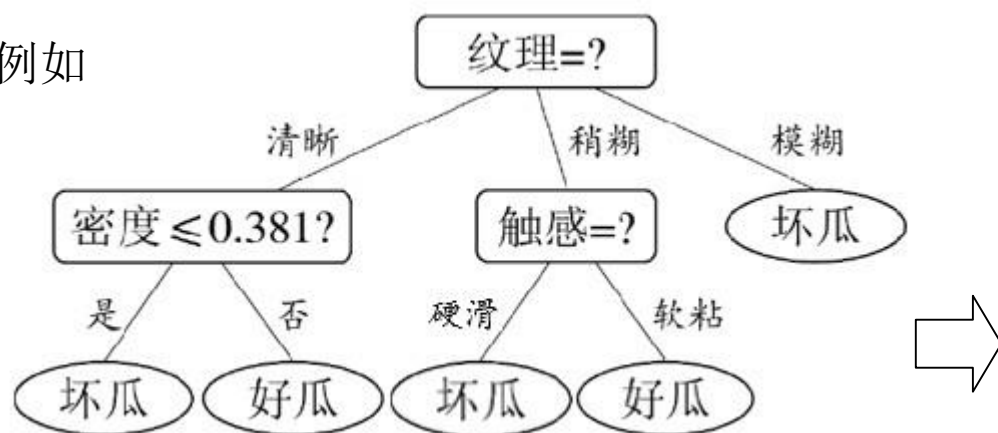


结论：一个连续属性转换成 $t$ 个二值属性， $t$ 为候选划分点的个数。

# 从“树”到“规则”

- 一棵决策树对应于一个“规则集”
- 每个从根结点到叶结点的分支路径对应于一条规则

例如



- IF (纹理=清晰)  $\wedge$  (密度 $\leq 0.381$ ) THEN 坏瓜
- IF (纹理=清晰)  $\wedge$  (密度 $> 0.381$ ) THEN 好瓜
- IF (纹理=稍糊)  $\wedge$  (触感=硬滑) THEN 坏瓜
- IF (纹理=稍糊)  $\wedge$  (触感=软粘) THEN 好瓜
- IF (纹理=模糊) THEN 坏瓜

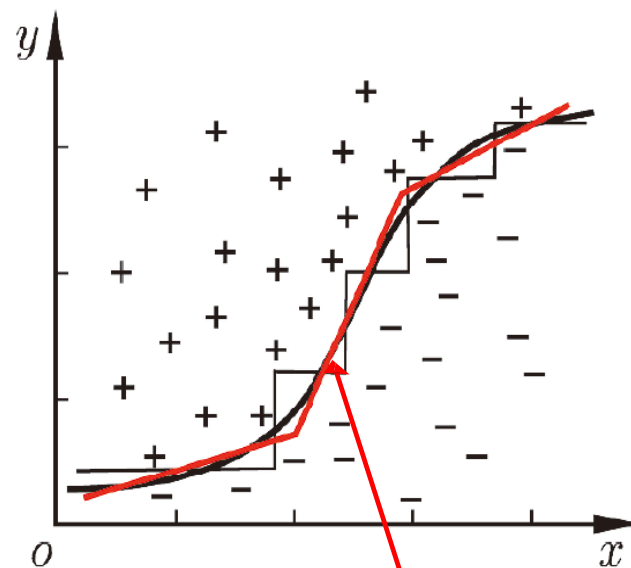
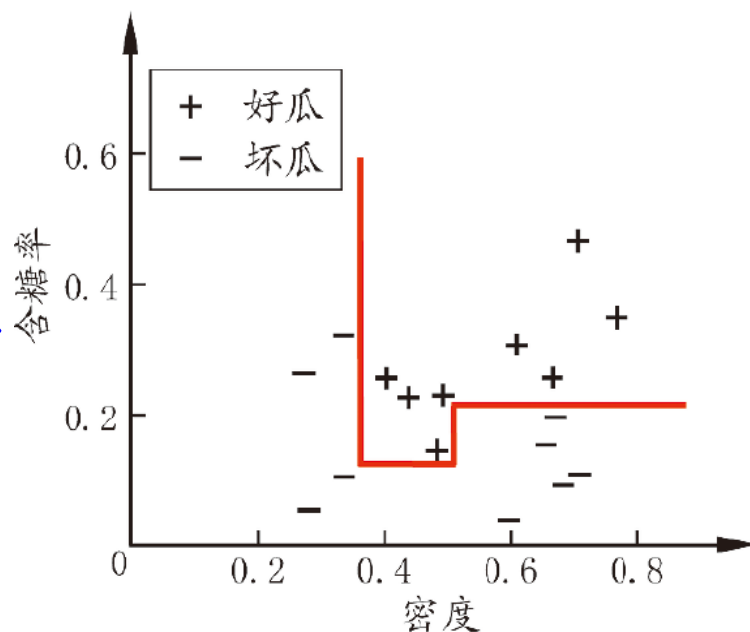
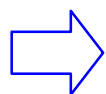
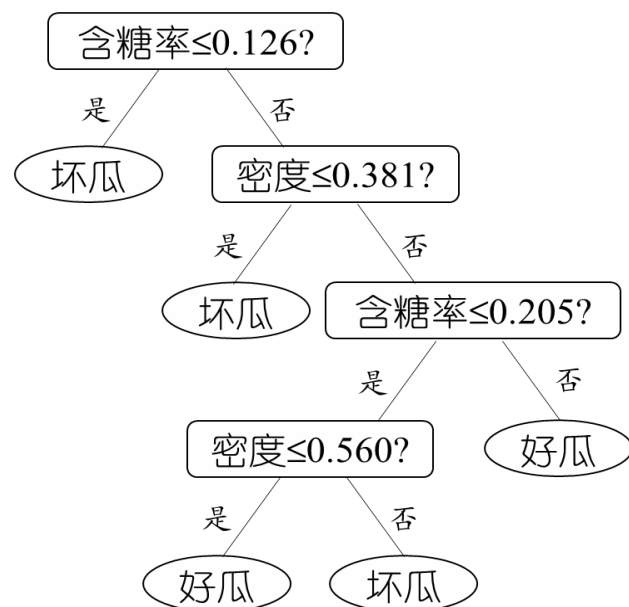
好处:

- 改善可理解性
- 进一步提升泛化能力

由于转化过程中通常会进行前件合并、泛化等操作  
例如 **C4.5Rule** 的泛化能力通常优于 **C4.5**决策树

# 决策边界：轴平行 vs. 倾斜

单变量决策树：在每个非叶结点仅考虑一个划分属性



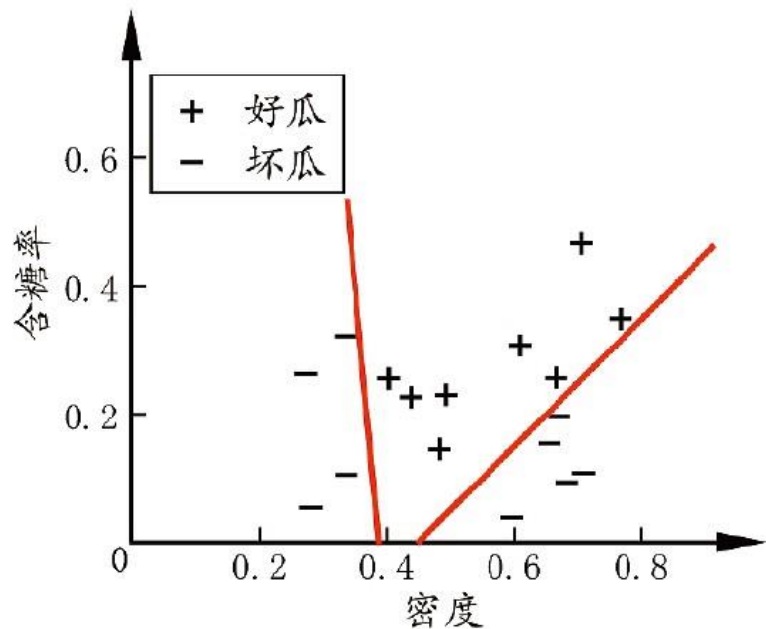
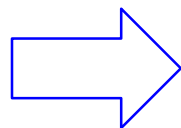
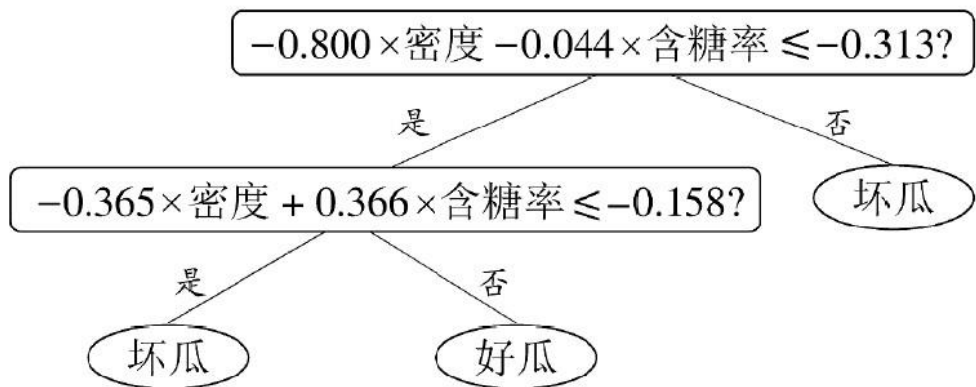
能否产生这样的决策边界?

单变量决策树产生“轴平行”决策边界 (decision boundary)

# 多变量(multivariate)决策树

多变量决策树：每个非叶结点不仅考虑一个属性

例如“**斜决策树**”(oblique decision tree) 不是为每个非叶结点寻找最优划分属性，而是建立一个**线性分类器**



更复杂的“混合决策树”甚至可以在结点嵌入神经网络或其他非线性模型