# 四、决策树



主讲教师：周志华

# 决策树模型

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

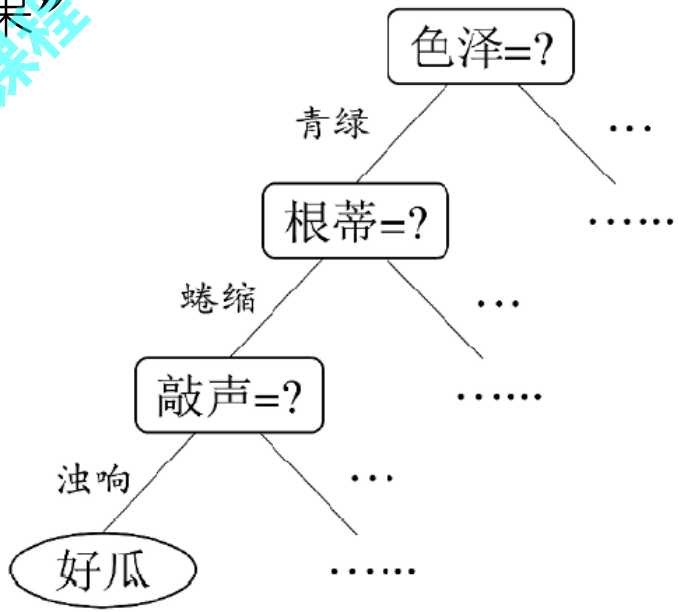



图 4.1 西瓜问题的一棵决策树

# 决策树简史

- 第一个决策树算法： CLS (Concept Learning System)

  [E. B. Hunt, J. Marin, and P. T. Stone's book "*Experiments in Induction*" published by Academic Press in 1966]

- 使决策树受到关注、成为机器学习主流技术的算法： ID3

  [J. R. Quinlan's paper in a book "*Expert Systems in the Micro Electronic Age*" edited by D. Michie, published by Edinburgh University Press in 1979]

- 最常用的决策树算法： C4.5

  [J. R. Quinlan's book "*C4.5: Programs for Machine Learning*" published by Morgan Kaufmann in 1993]

# 决策树简史(con't)

- 可以用于回归任务的决策树算法： CART (Classification and Regression Tree)

  [L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone's book "Classification and Regression Trees" published by Wadsworth in 1984]

- 基于决策树的最强大算法： RF (Random Forest)

  [L. Breiman's MLJ'01 paper "Random Forest"]

  这是一种"集成学习"方法 → 第 8 章

# 基本流程

策略：“分而治之”(divide-and-conquer)

自根至叶的递归过程

在每个中间结点寻找一个“划分”(split or test)属性

三种停止条件：

(1) 当前结点包含的样本全属于同一类别，无需划分；

(2) 当前属性集为空，或是所有样本在所有属性上取值相同，无法划分；

(3) 当前结点包含的样本集合为空，不能划分.

# 基本算法

**输入:** 训练集 $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$;
属性集 $A = \{a_1, a_2, \ldots, a_d\}$.

**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 = ∅ 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 \ {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

# 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

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

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

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

$\mathrm{Ent}(D)$ 的值越小，则 $D$ 的纯度越高

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

# 信息增益

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

$D^v$：$D$ 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：

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

划分前的信息熵

划分后的信息熵

第 $v$ 个分支的权重，
样本越多越重要



ID3算法中使用

# 一个例子

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 |

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

根结点的信息熵为

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

# 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，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：

$$\begin{aligned}
\mathrm{Ent}(D^1) &= -(\tfrac{3}{6}\log_2\tfrac{3}{6} + \tfrac{3}{6}\log_2\tfrac{3}{6}) = 1.000 \\
\mathrm{Ent}(D^2) &= -(\tfrac{4}{6}\log_2\tfrac{4}{6} + \tfrac{2}{6}\log_2\tfrac{2}{6}) = 0.918 \\
\mathrm{Ent}(D^3) &= -(\tfrac{1}{5}\log_2\tfrac{1}{5} + \tfrac{4}{5}\log_2\tfrac{4}{5}) = 0.722
\end{aligned}$$

- 属性 “色泽” 的信息增益为

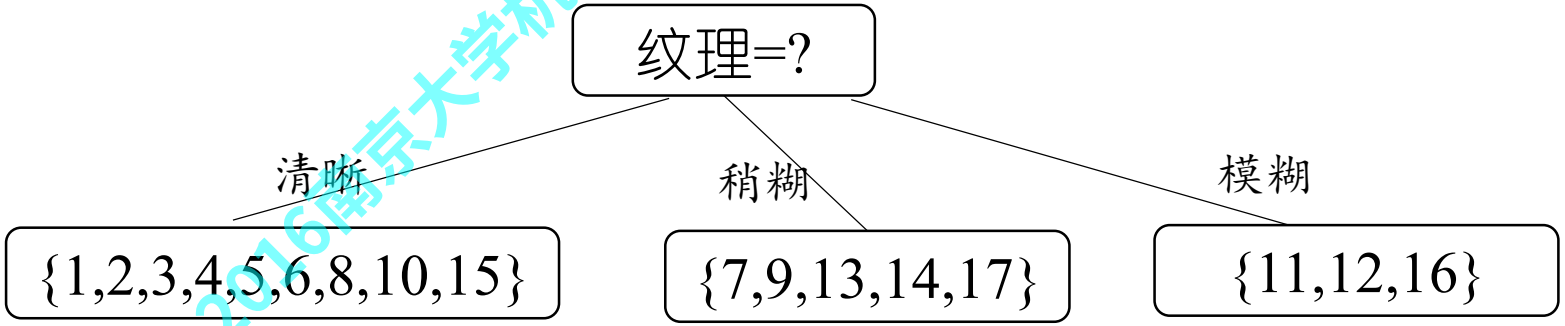
$$\begin{aligned}
\mathrm{Gain}(D, \text{色泽}) &= \mathrm{Ent}(D) - \sum_{v=1}^{3} \frac{|D^v|}{|D|}\mathrm{Ent}(D^v) \\
&= 0.998 - (\tfrac{6}{17} \times 1.000 + \tfrac{6}{17} \times 0.918 + \tfrac{5}{17} \times 0.722) \\
&= 0.109
\end{aligned}$$

# 一个例子 (续)

- 类似的，其他属性的信息增益为

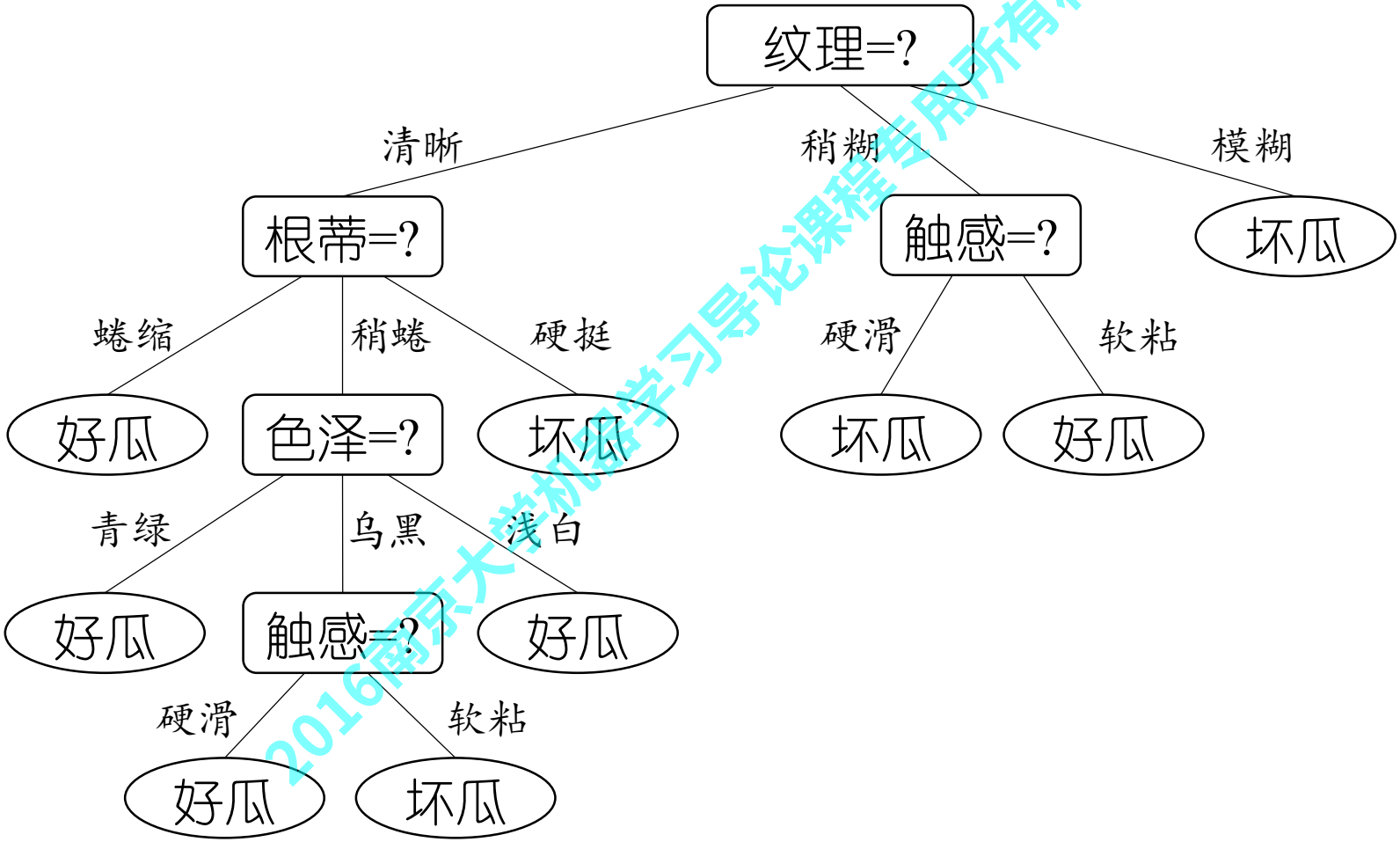
$$\text{Gain}(D, \text{根蒂}) = 0.143 \qquad \text{Gain}(D, \text{敲声}) = 0.141$$
$$\text{Gain}(D, \text{纹理}) = 0.381 \qquad \text{Gain}(D, \text{脐部}) = 0.289$$
$$\text{Gain}(D, \text{触感}) = 0.006$$

- 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



# 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：
$$\text{Gain\_ratio}(D,a)=\frac{\text{Gain}(D,a)}{\text{IV}(a)}$$

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

C4.5算法中使用

# 基尼指数 (gini index)

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

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

$\mathrm{Gini}(D)$ 越小，数据集 $D$ 的纯度越高

属性 $a$ 的基尼指数：$\mathrm{Gini\_index}(D,a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|}\mathrm{Gini}(D^v)$

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

CART算法中使用

# 划分选择 vs. 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%

**Why?**

剪枝 (pruning) 是决策树对付“过拟合”的主要手段！

# 剪枝

为了尽可能正确分类训练样本，有可能造成分支过多 → 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

剪枝过程中需评估剪枝前后决策树的优劣 ⟶ 第 2 章

现在我们假定使用“留出法”

# 数据集

训练集

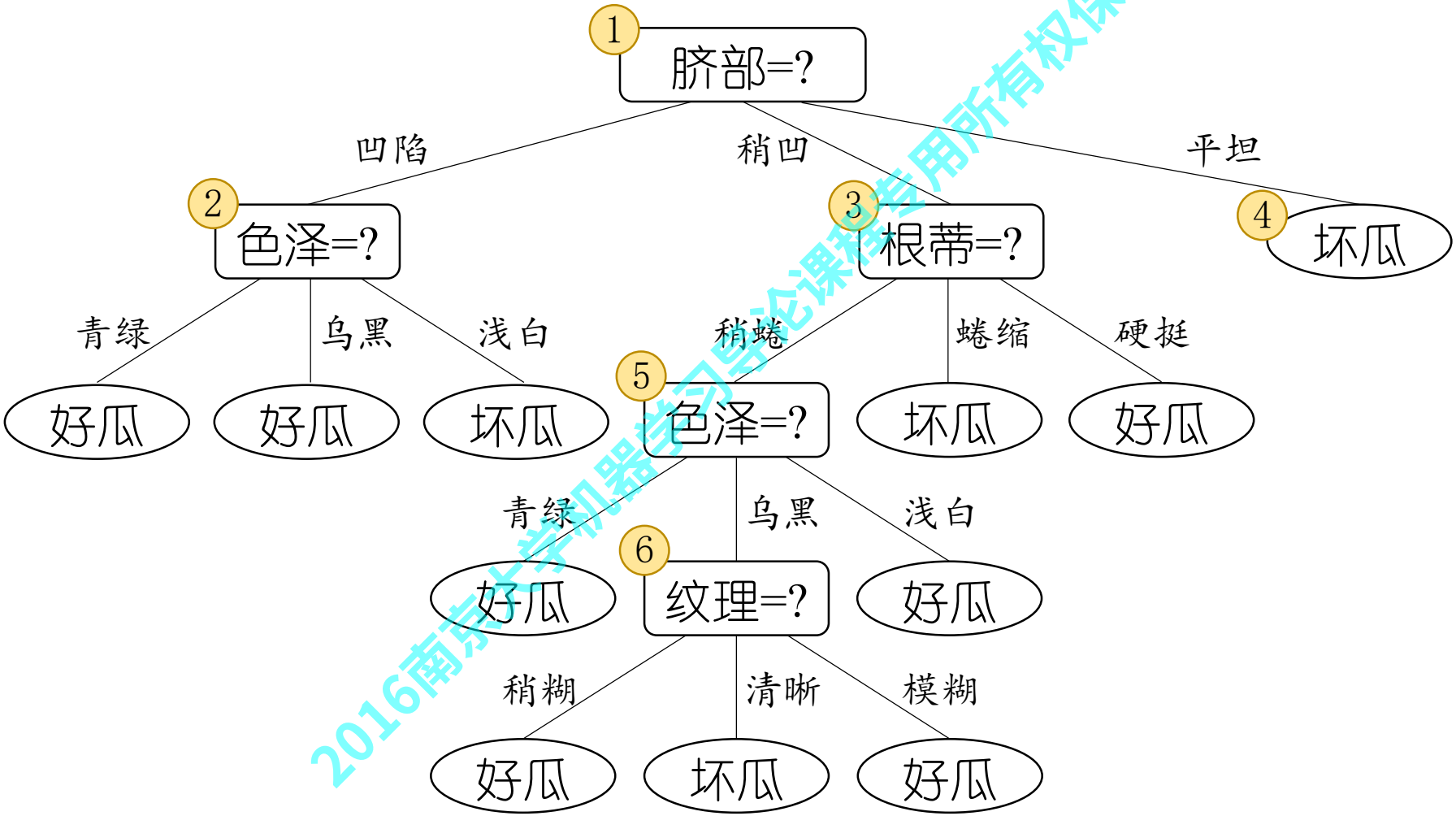
验证集

**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)

| 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 |

| 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 |

# 未剪枝决策树

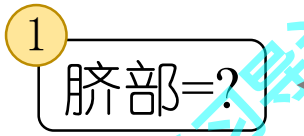


# 预剪枝

| 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 |

验证集

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，得到验证集精度为 $\frac{3}{7} \times 100\% = 42.9\%$




验证集精度

“脐部=?” 划分前：42.9%

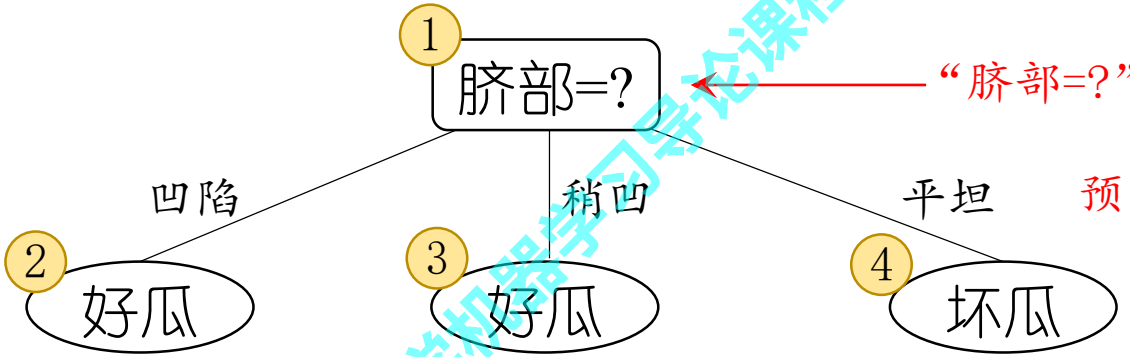
# 预剪枝 (续)

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，验证集精度为$\frac{5}{7}\times 100\% = 71.4\%$

验证集精度




“脐部=?” 划分前：42.9%

划分后：71.4%

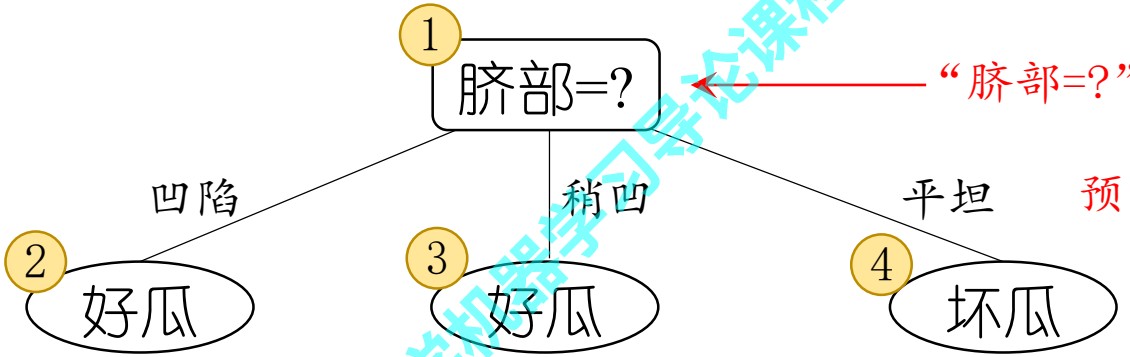
预剪枝决策：划分

# 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②，③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，称为“决策树桩”(decision stump)




验证集精度

“脐部=?” 划分前：42.9%

划分后：71.4%

预剪枝决策：划分

验证集精度

“色泽=?” 划分前：71.4%

划分后：57.1%

预剪枝决策：禁止划分

验证集精度

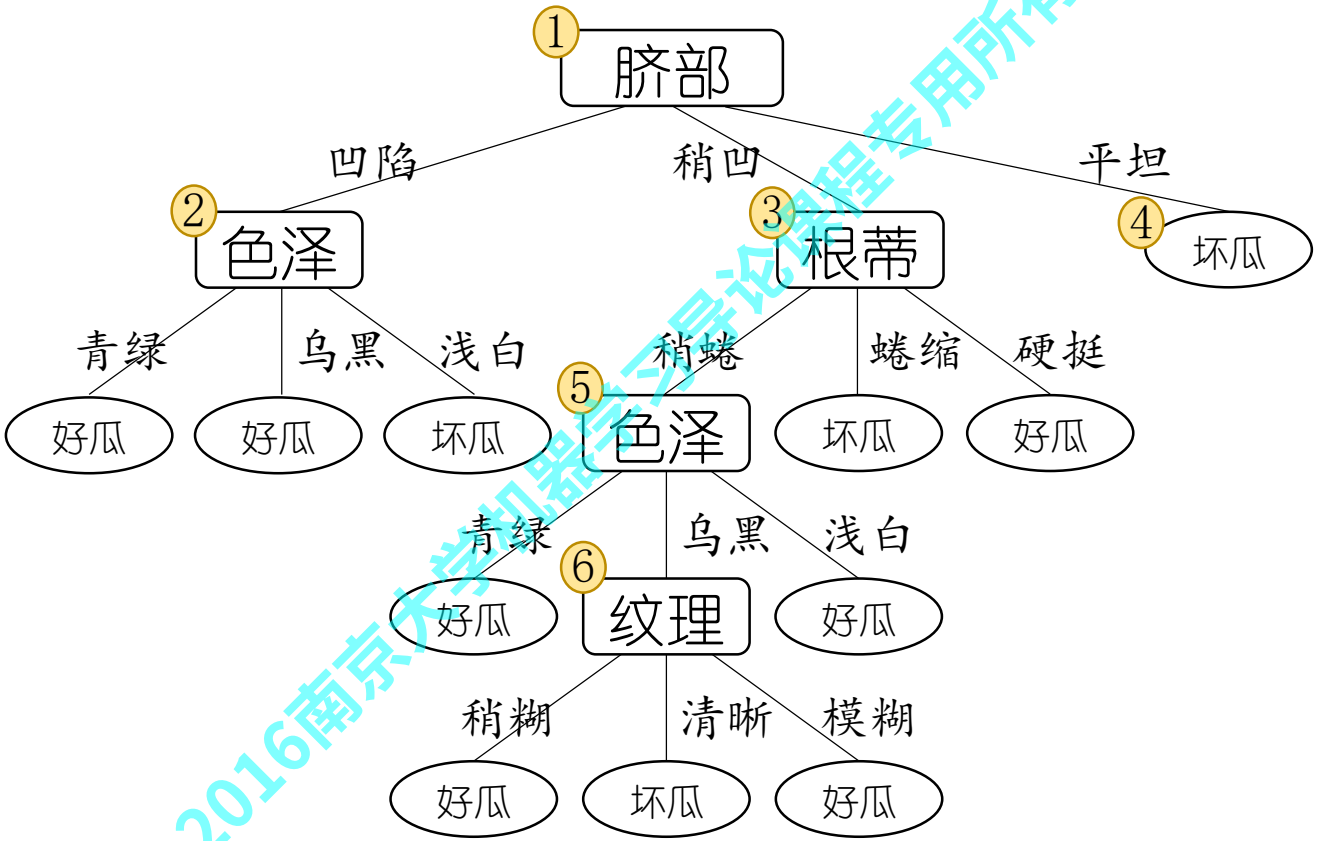
“根蒂=?” 划分前：71.4%

划分后：71.4%

预剪枝决策：禁止划分

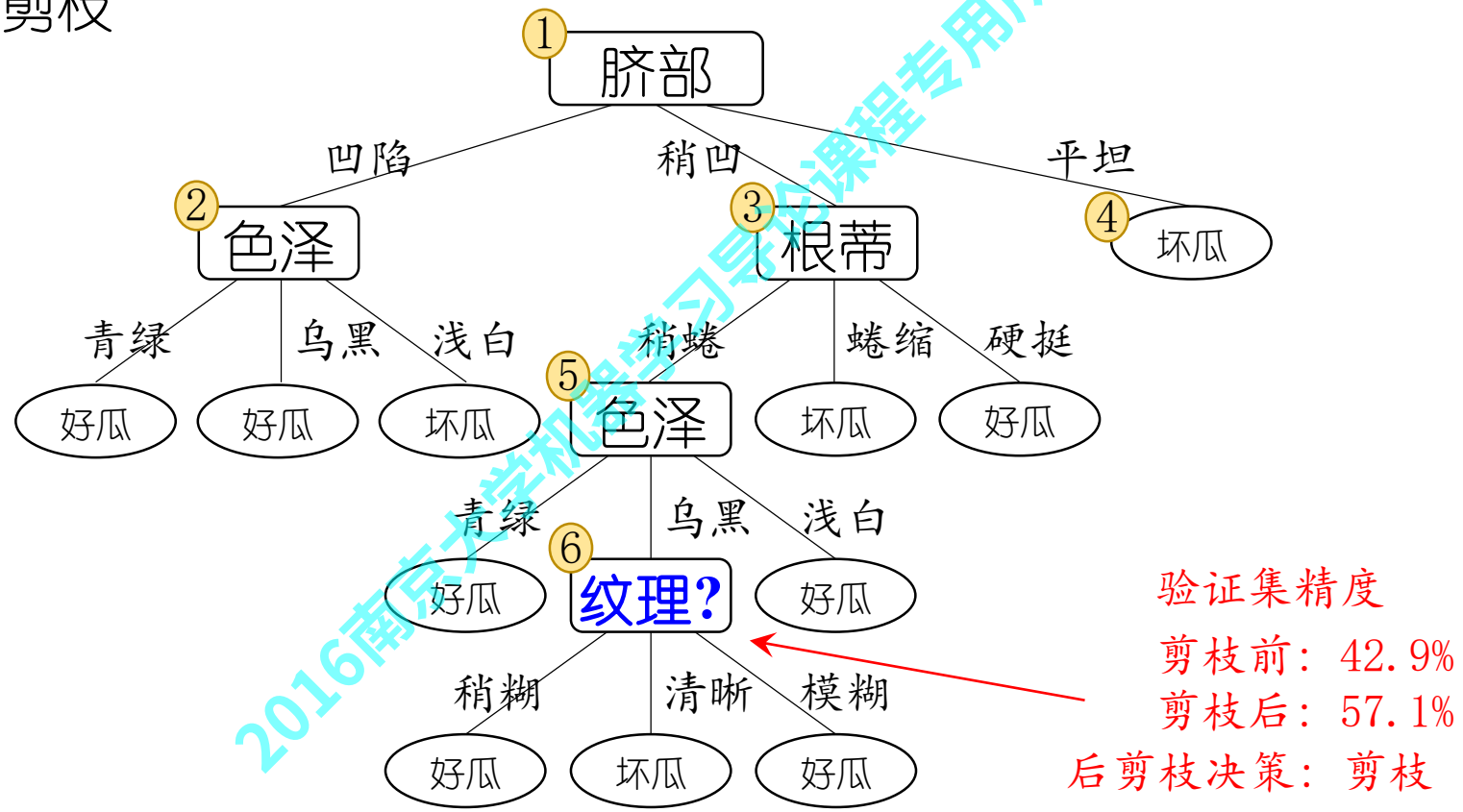
# 后剪枝

先生成一棵完整的决策树，其验证集精度测得为 42.9%



# 后剪枝 (续)

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”，测得验证集精度提高至 57.1%，于是决定剪枝




验证集精度

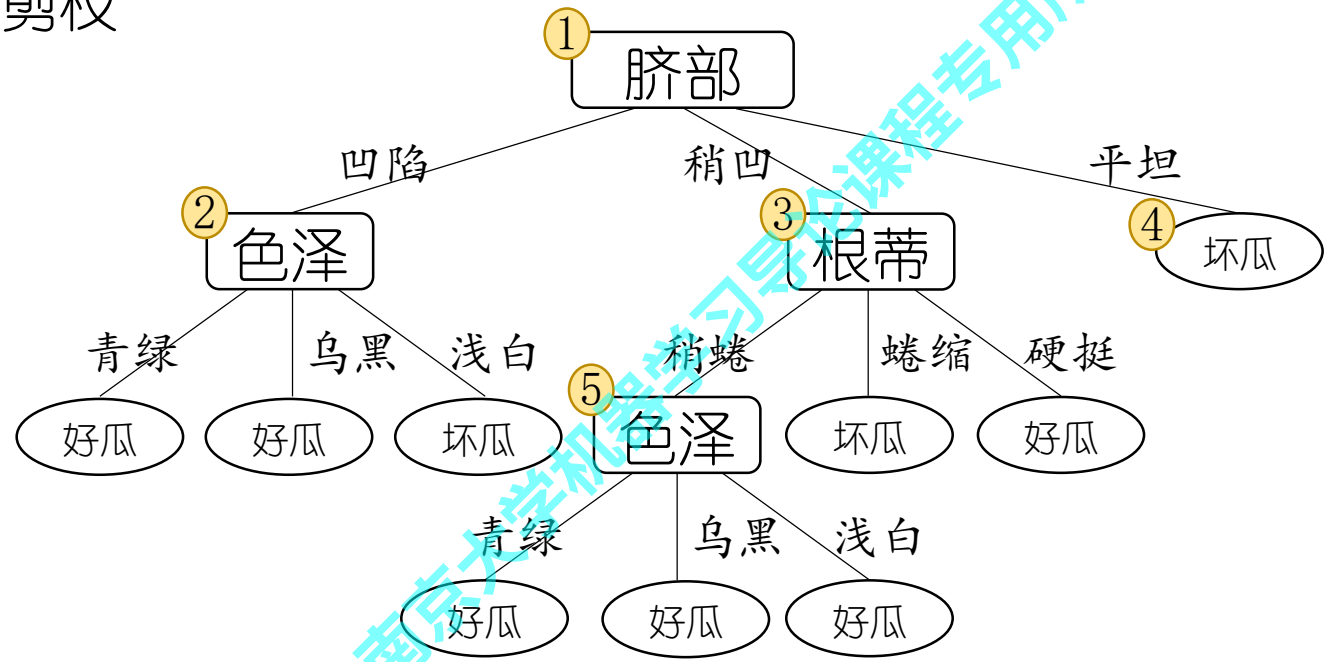
剪枝前：42.9%

剪枝后：57.1%

后剪枝决策：剪枝

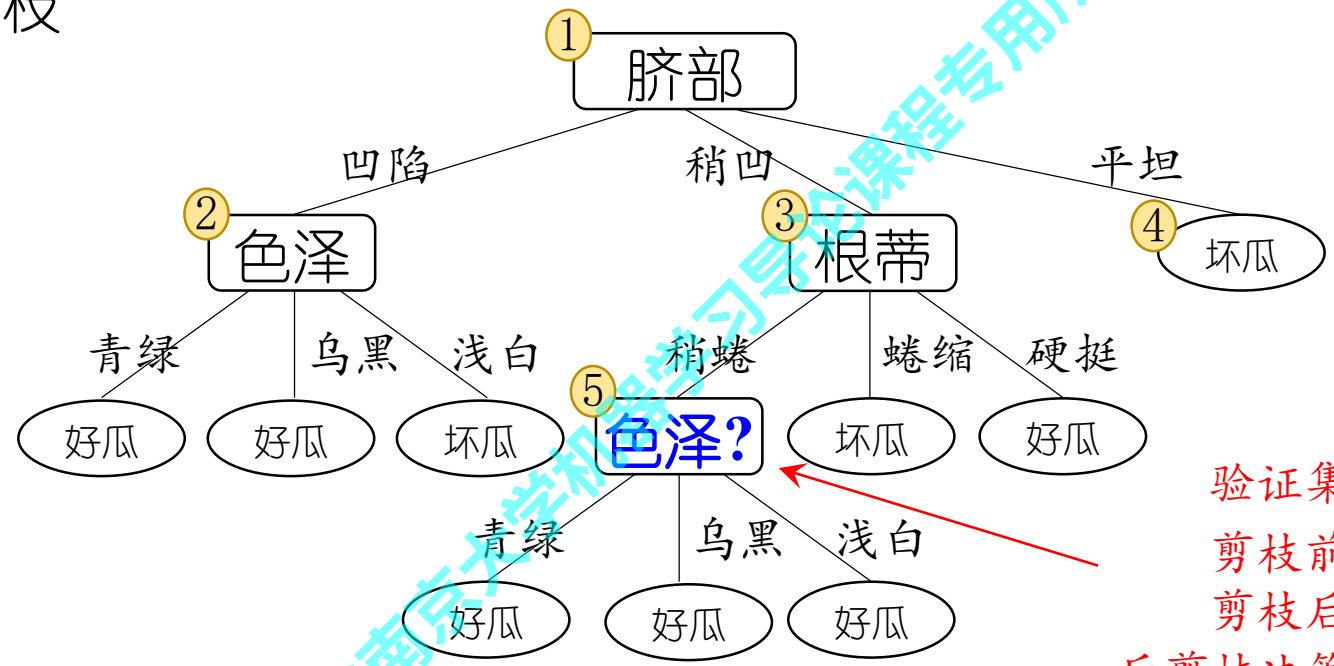
# 后剪枝 (续)

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”，测得验证集精度提高至 57.1%，于是决定剪枝



# 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 $\{6, 7, 15\}$ 将其标记为“好瓜”，测得验证集精度仍为 57.1%，可以不剪枝




验证集精度

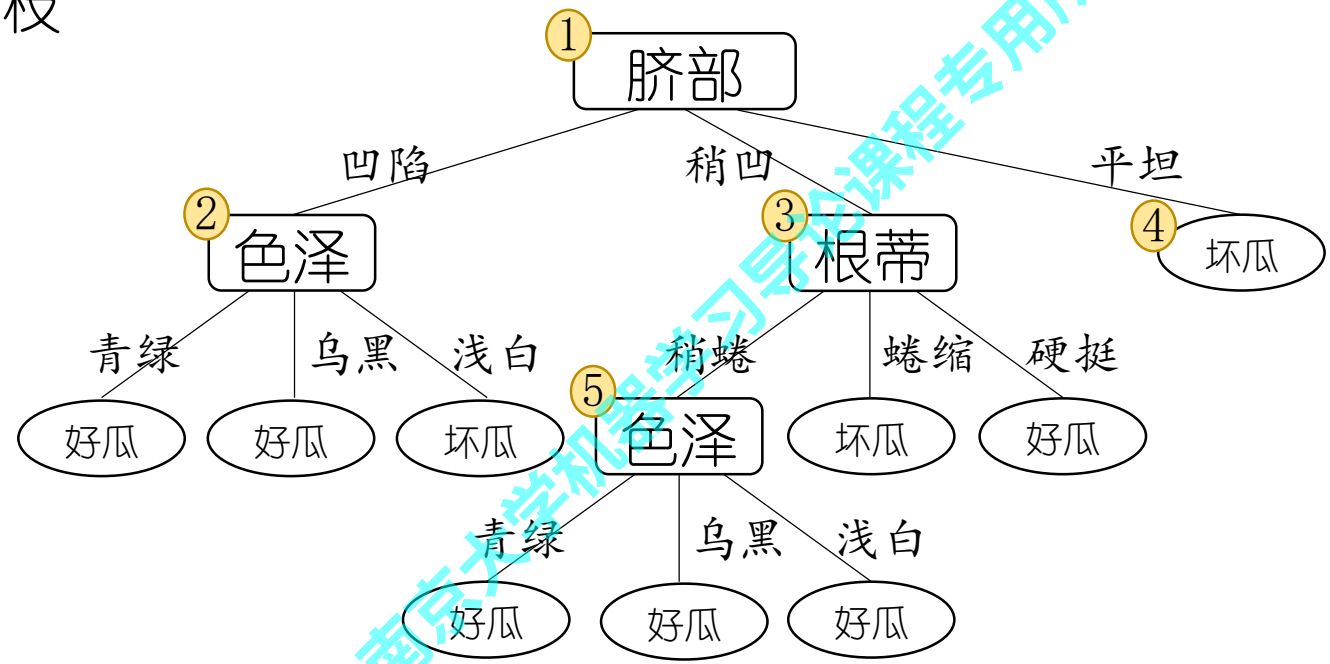
剪枝前：57.1%

剪枝后：57.1%

后剪枝决策：不剪枝

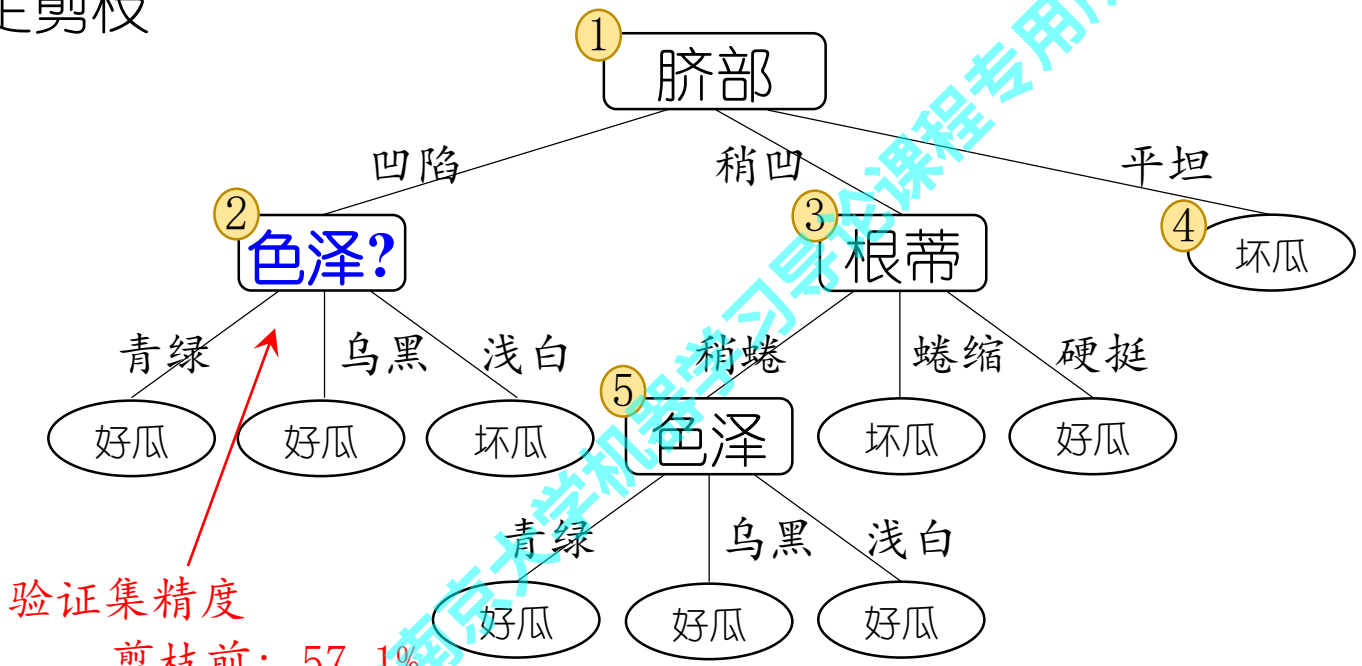
# 后剪枝 (续)

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 $\{6, 7, 15\}$ 将其标记为“好瓜”，测得验证集精度仍为 57.1%，可以不剪枝



# 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”，测得验证集精度提升至 71.4%，决定剪枝




验证集精度

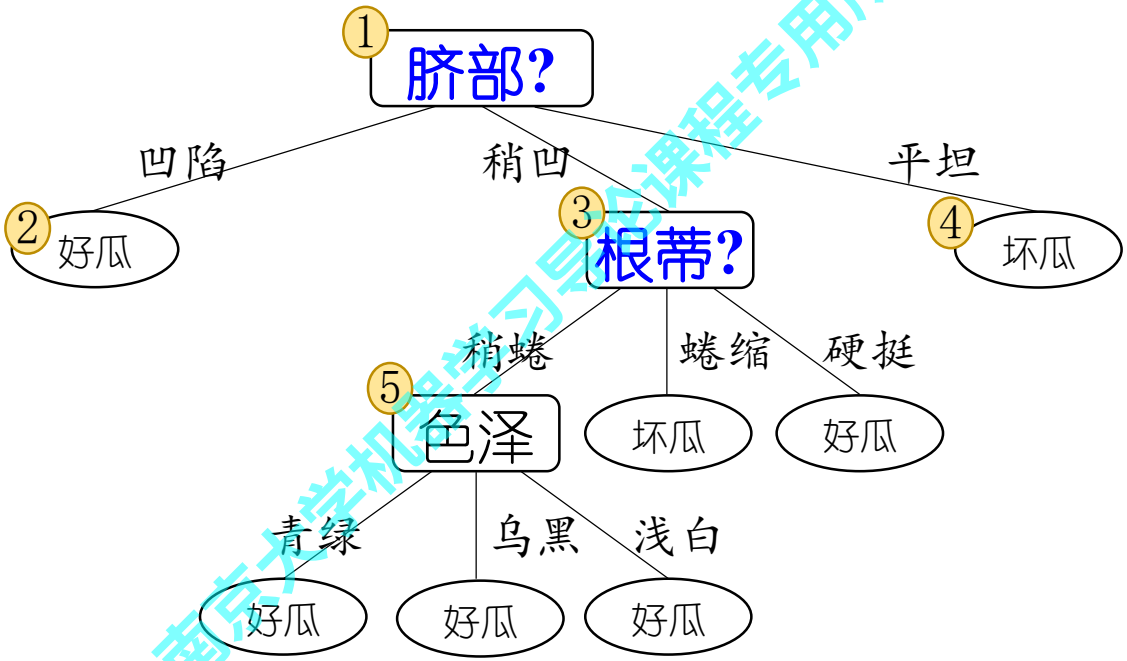
剪枝前：57.1%

剪枝后：71.4%

后剪枝决策：剪枝

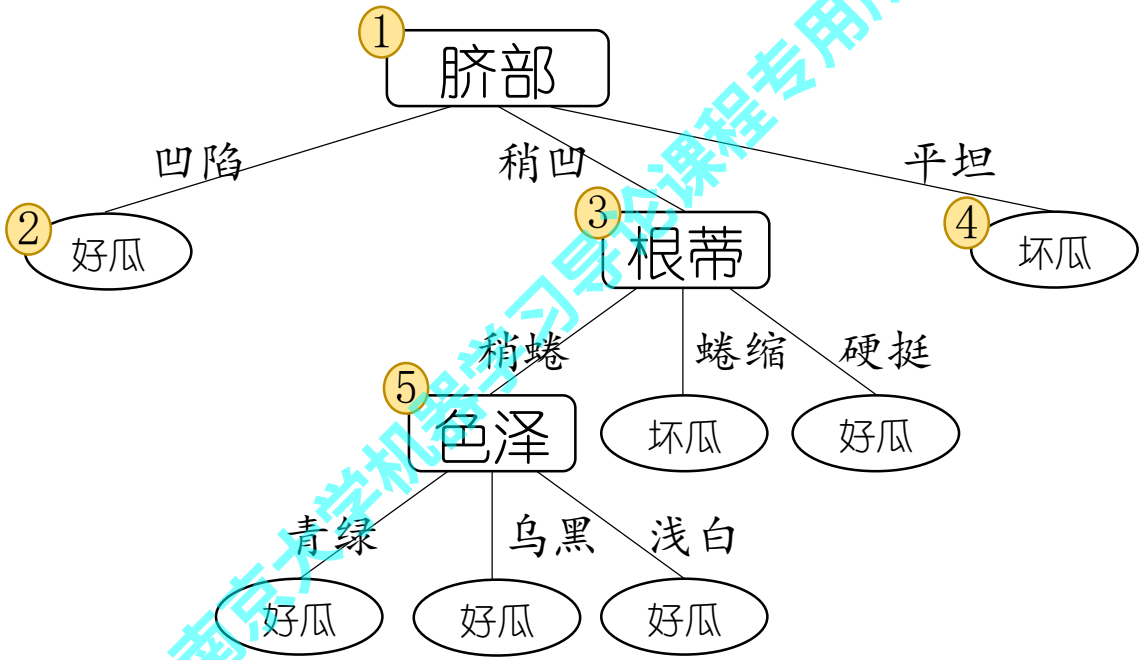
# 后剪枝 (续)

对结点③和①，先后替换为叶结点，均未测得验证集精度提升，于是不剪枝



# 后剪枝 (续)

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

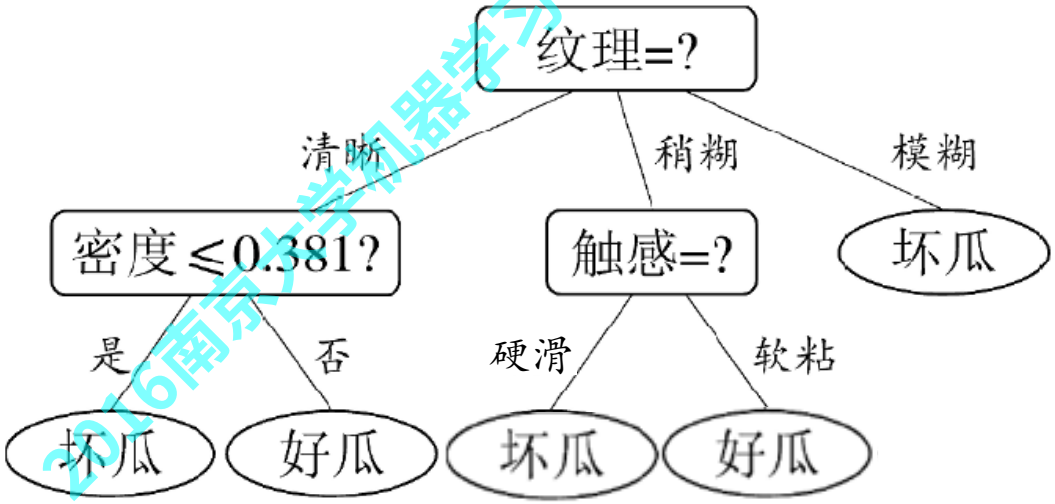
- 泛化性能：后剪枝 通常优于 预剪枝

# 连续值

基本思路：连续属性离散化

常见做法：二分法 (bi-partition)

- $n$ 个属性值可形成 $n$-1 个候选划分
- 然后即可将它们当做 $n$-1 个离散属性值处理



# 缺失值

现实应用中，经常会遇到属性值“缺失”(missing)现象

仅使用无缺失的样例？ → 对数据的极大浪费

使用带缺失值的样例，需解决：

Q1：如何进行划分属性选择？

Q2：给定划分属性，若样本在该属性上的值缺失，如何进行划分？

基本思路：样本赋权，权重划分

# 一个例子

仅通过无缺失值的样例来判断划分属性的优劣

学习开始时，根结点包含样例集 $D$ 中全部17个样例，权重均为 1

表 4.4　西瓜数据集 2.0$\alpha$

| 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 |

以属性“色泽”为例，该属性上无缺失值的样例子集 $\tilde{D}$ 包含 14 个样例，信息熵为

$$\mathrm{Ent}(\tilde{D}) = -\sum_{k=1}^{2} \tilde{p}_k \log_2 \tilde{p}_k = -\left(\frac{6}{14}\log_2\frac{6}{14} + \frac{8}{14}\log_2\frac{8}{14}\right) = 0.985$$

# 一个例子

令 $\tilde{D}^1$, $\tilde{D}^2$, $\tilde{D}^3$ 分别表示在属性“色泽”上取值为“青绿”“乌黑”以及“浅白”的样本子集，有

$$\mathrm{Ent}(\tilde{D}^1) = -\left(\tfrac{2}{4}\log_2\tfrac{2}{4} + \tfrac{2}{4}\log_2\tfrac{2}{4}\right) = 1.000 \quad \mathrm{Ent}(\tilde{D}^2) = -\left(\tfrac{4}{6}\log_2\tfrac{4}{6} + \tfrac{2}{6}\log_2\tfrac{2}{6}\right) = 0.918$$

$$\mathrm{Ent}(\tilde{D}^3) = -\left(\tfrac{0}{4}\log_2\tfrac{0}{4} + \tfrac{4}{4}\log_2\tfrac{4}{4}\right) = 0.000$$

因此，样本子集 $\tilde{D}$ 上属性“色泽”的信息增益为

$$\begin{aligned}\mathrm{Gain}(\tilde{D}, \text{色泽}) &= \mathrm{Ent}(\tilde{D}) - \sum_{v=1}^{3} \tilde{r}_v \mathrm{Ent}(\tilde{D}^v) \\ &= 0.985 - \left(\tfrac{4}{14} \times 1.000 + \tfrac{6}{14} \times 0.918 + \tfrac{4}{14} \times 0.000\right) \\ &= 0.306\end{aligned}$$

无缺失值样例中属性 $a$ 取值为 $v$ 的占比

于是，样本集 $D$ 上属性“色泽”的信息增益为

$$\mathrm{Gain}(D, \text{色泽}) = \rho \times \mathrm{Gain}(\tilde{D}, \text{色泽}) = \tfrac{14}{17} \times 0.306 = 0.252$$

无缺失值样例占比

# 一个例子

类似地可计算出所有属性在数据集上的信息增益

$$\text{Gain}(D, \text{色泽}) = 0.252 \qquad \text{Gain}(D, \text{根蒂}) = 0.171$$
$$\text{Gain}(D, \text{敲声}) = 0.145 \qquad \text{Gain}(D, \text{纹理}) = 0.424$$
$$\text{Gain}(D, \text{脐部}) = 0.289 \qquad \text{Gain}(D, \text{触感}) = 0.006$$

进入“纹理=清晰”分支

进入“纹理=稍糊”分支

进入“纹理=模糊”分支

样本权重在各子结点仍为1

在“纹理”上出现缺失值，样本 8, 10 同时进入三个分支，三分支上的权重分别为 7/15, 5/15, 3/15

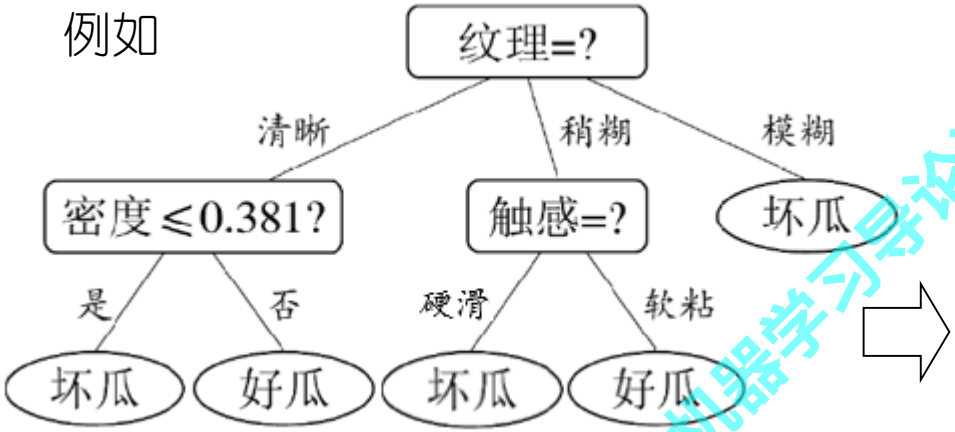
| 编号 | 色泽 | 根蒂 | 敲声 | 纹理 | 脐部 | 触感 | 好瓜 |
|---|---|---|---|---|---|---|---|
| 1 | – | 蜷缩 | 浊响 | 清晰 | 凹陷 | 硬滑 | 是 |
| 2 | 乌黑 | 蜷缩 | 沉闷 | 清晰 | 凹陷 | – | 是 |
| 3 | 乌黑 | 蜷缩 | – | 清晰 | 凹陷 | 硬滑 | 是 |
| 4 | 青绿 | 蜷缩 | 沉闷 | 清晰 | 凹陷 | 硬滑 | 是 |
| 5 | – | 蜷缩 | 浊响 | 清晰 | 凹陷 | 硬滑 | 是 |
| 6 | 青绿 | 稍蜷 | 浊响 | 清晰 | – | 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 |

权重划分

# 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) ∧ (密度≤0.381) THEN 坏瓜
- IF (纹理=清晰) ∧ (密度>0.381) THEN 好瓜
- IF (纹理=稍糊) ∧ (触感=硬滑) THEN 坏瓜
- IF (纹理=稍糊) ∧ (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

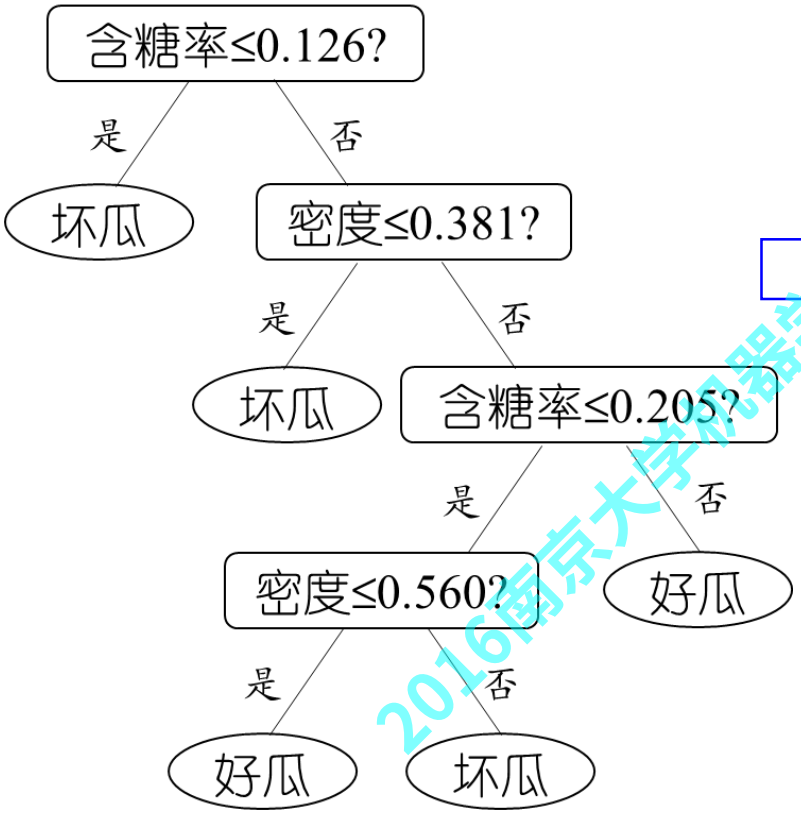
由于转化过程中通常会进行前件合并、泛化等操作

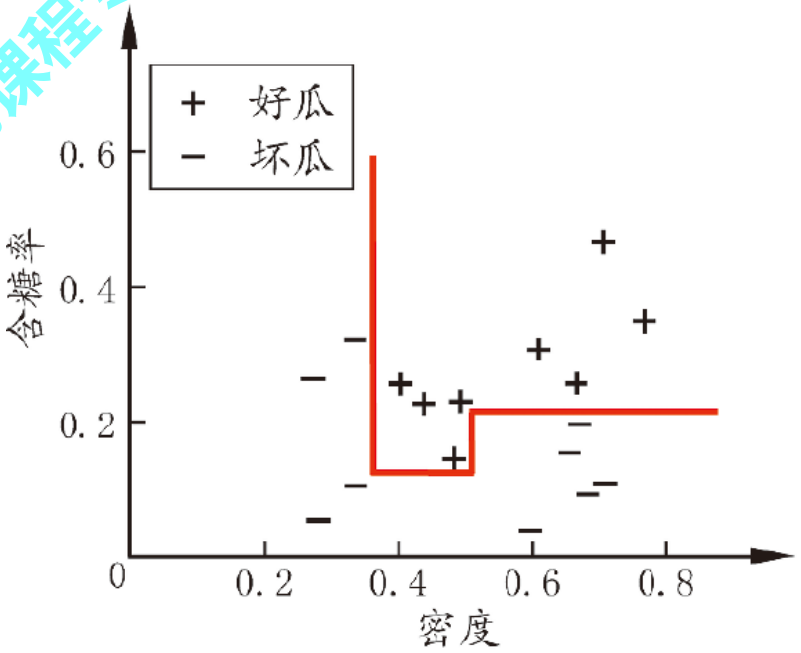
例如 C4.5Rule 的泛化能力通常优于 C4.5决策树

# 轴平行划分

单变量决策树：在每个非叶结点仅考虑一个划分属性

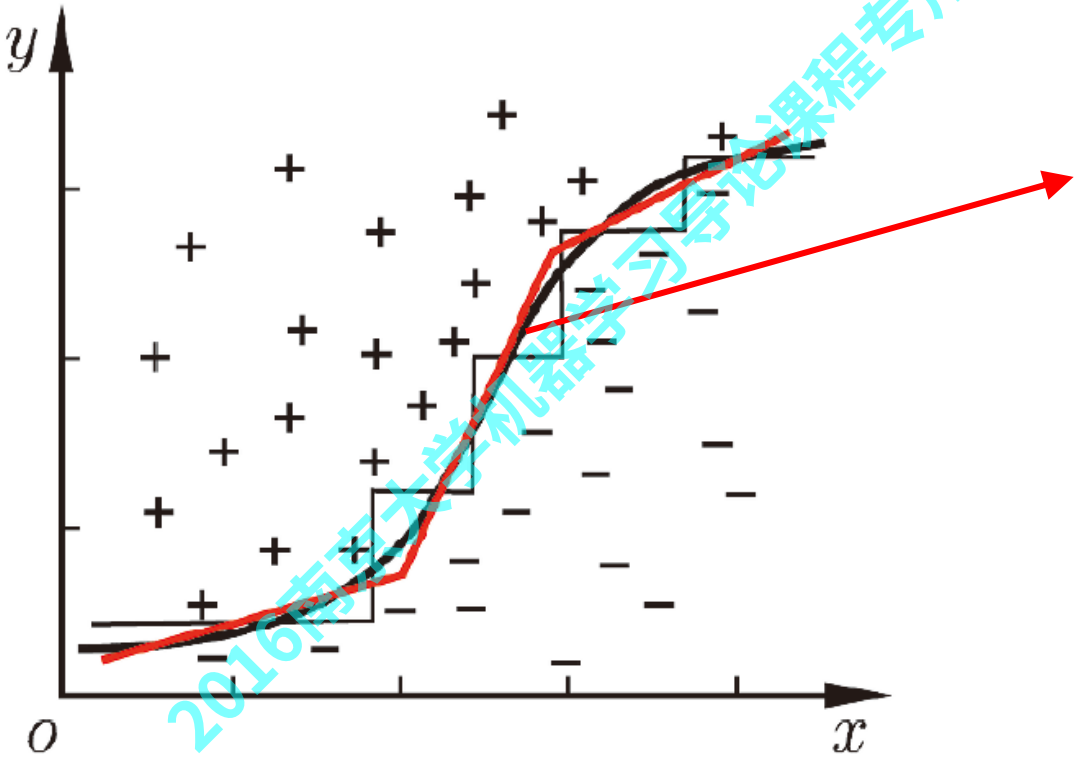
产生“轴平行”分类面



# 轴平行 vs. 倾斜

当学习任务所对应的分类边界很复杂时，需要非常多段划分才能获得较好的近似

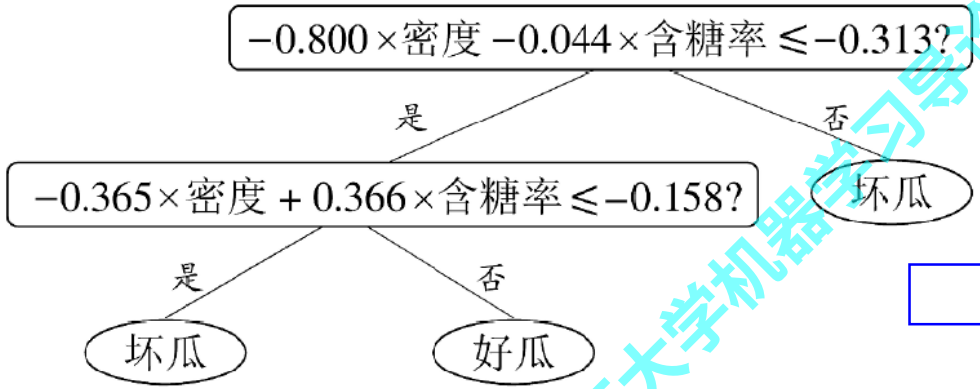


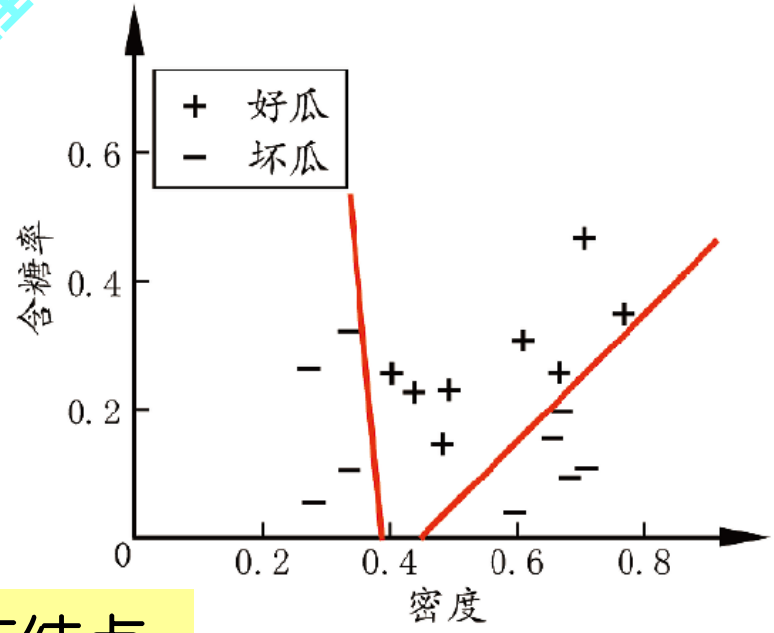

能否产生这样的分类边界？

# 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

# 决策树常用软件包

- ID3, C4.5, C5.0

  http://www.rulequest.com/Personal/

- J4.8

  http://www.cs.waikato.ac.nz/ml/weka/

- … …

# 前往第五站……