MACHINE LEARNING

机器学习

第四章　决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树

第四章　决策树

重点

- 划分选择
- 剪枝处理

难点

- 划分选择
- 剪枝处理

第四章　决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树

基本流程

决策树 基于树结构来进行预测

根结点

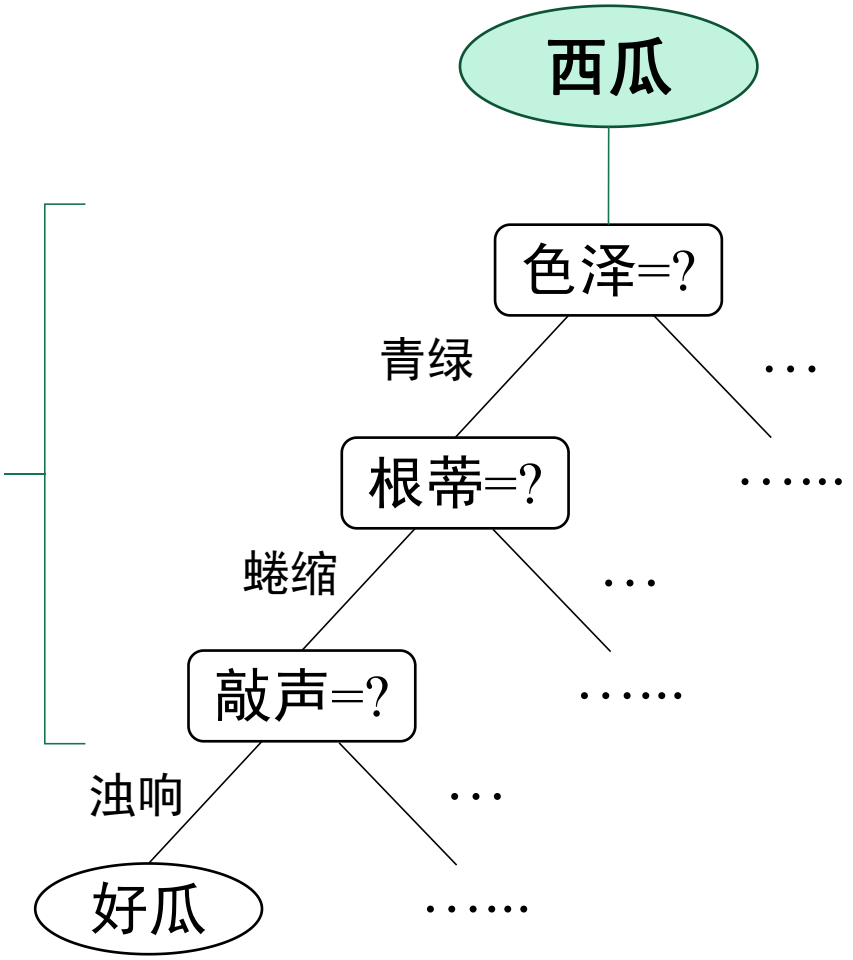
包含　样本全集

内部结点

每个内部节点 对应一个属性测试

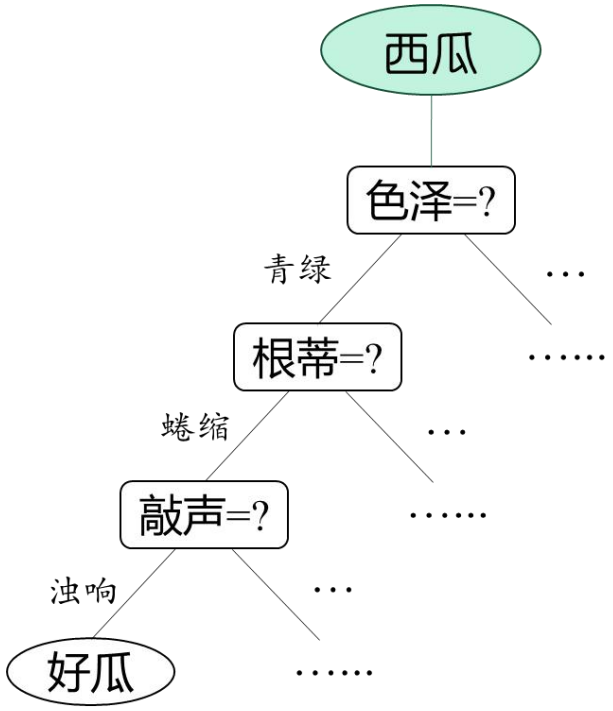
叶结点

每个叶节点对应一个决策结果



基本流程

决策

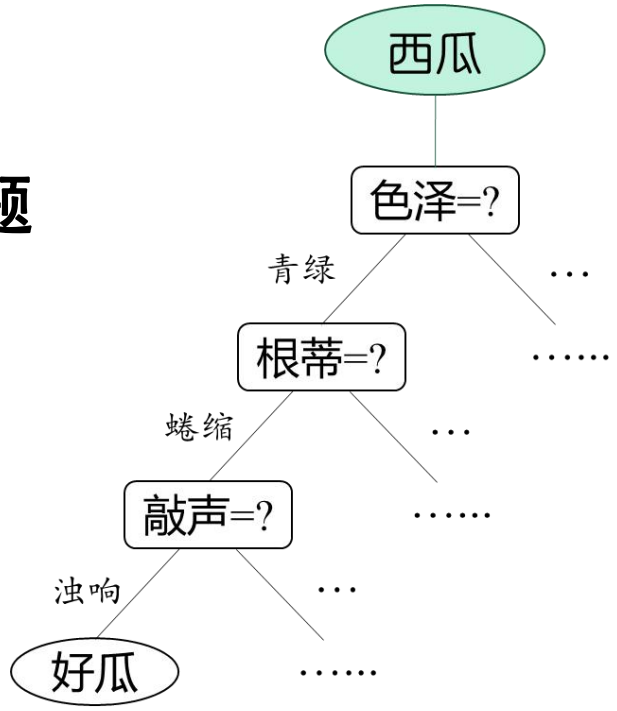


- 决策过程中，提出的每个判定问题都是对某个属性的“测试”
  - ✓ 每个“测试”考虑的范围：
    限定在上次决策结果的范围之内
  - ✓ 每个“测试”结果：
    导出 最终结论 或 进一步的判定问题

基本流程

决策

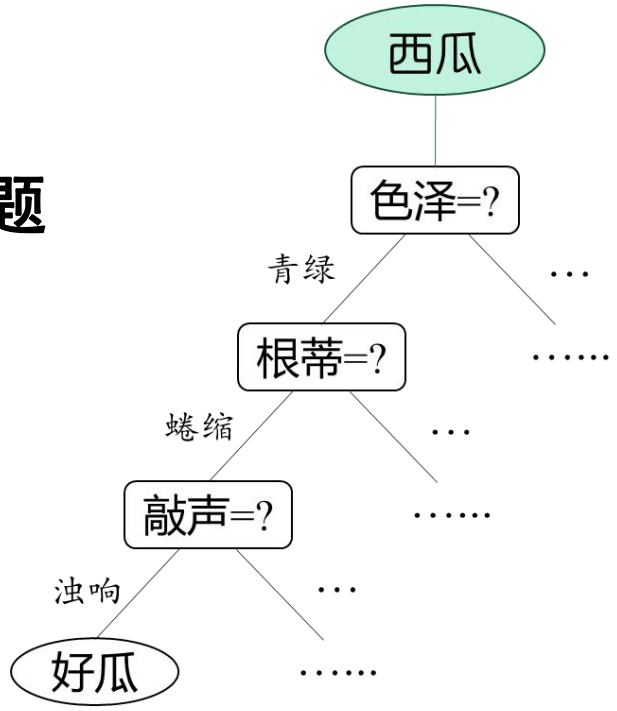
- 决策过程中，提出的每个判定问题都是对某个属性的“测试”
- 从根结点到每个叶结点的路径，对应了 一个判定测试序列
- 决策过程的最终结论对应了 我们所希望的判定结果



基本流程

决策

- 决策过程中，提出的每个判定问题都是对某个属性的“测试”
- 从根结点到每个叶结点的路径，对应了 一个判定测试序列
- 决策过程的最终结论对应了 我们所希望的判定结果



决策树学习的目的 是为了产生一棵泛化能力强，即处理未见示例能力强的决策树

基本流程

决策树学习：递归算法

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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 生成每一个分枝; 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:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

三种情况导致

- 递归返回
- 标记 叶结点

基本流程

决策树学习：递归算法

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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 生成每一个分枝; 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:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

三种情况导致

- 递归返回
- 标记 叶结点

1. 当前结点包含的样本，实际上全部属于同一类别，无需划分

标记 叶结点

基本流程

决策树学习：递归算法

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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** $A = \emptyset$ OR D 中样本在 A 上取值相同 **then**
6: 将 node 标记叶结点，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 生成每一个分枝; 令 D_v 表示 D 中在 a_* 上取值为 a_*^v 的样
11: **if** D_v 为空 **then**
12: 将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; ret
13: **else**
14: 以 TreeGenerate(D_v, $A - \{a_*\}$) 为分枝结点
15: **end if**
16: **end for**

输出: 以 node 为根结点的一棵决策树

三种情况导致

- 递归返回
- 标记 叶结点

2. 结点包含的样本，不同类别，

- 所有样本在所有属性值相同
- 当前属性集为空，无法划分

➢ 当前结点 标记为叶结点

✓利用 当前结点的后验分布

即类别标记为：该结点 所含样本 最多的类别

后验分布：已观测的分布/知识

基本流程

决策树学习：递归算法

三种情况导致递归返回

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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 生成每一个分枝; 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:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

3. 当前结点包含样本集为空，不能划分，把当前结点标记为叶结点

- ✓ 将父结点的样本分布 作为当前结点的 先验分布
- ✓ 类别 标记为：其父结点所含样本 最多的类别

先验分布：当前样本集分布未观测到，利用父结点分布推测当前结点分布

基本流程

决策树学习：递归算法

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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 生成每一个分枝; 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:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

三种情况导致递归返回

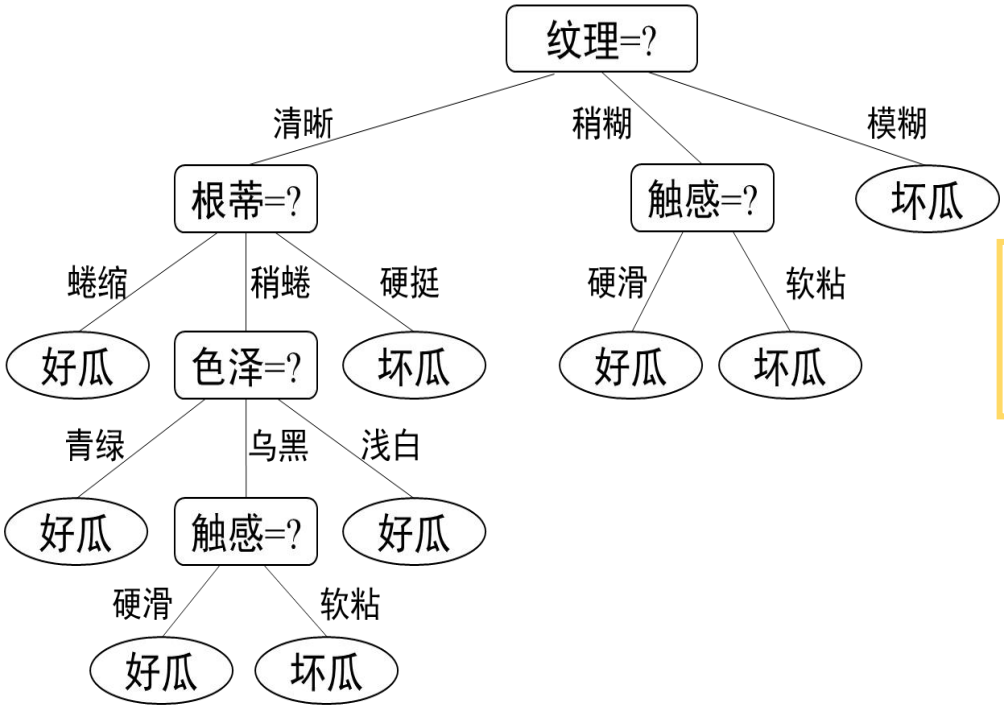
1. 同一类别，无需划分

2. 不同类别，但是
   - 属性集：取值相同或空，不能划分
   - 标记为叶结点，类别为该结点所含样最多的类别

3. 样本集为空，不能划分
   - 标记为叶结点，类别为父结点所含样本最多的类别

基本流程

- **完全决策树**：6个属性，每个属性有3个属性值，叶子结点数为$6!*3^6$
- 决策树学习算法**采用** **选择部分属性** 和 **剪枝** 的**技术**，得到最终的决策树



减少**属性值**个数/树宽度

减少**属性**个数/树高度

第四章　决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树

第四章 决策树

1. 基本流程
2. 核心技术
   - 划分选择
     - 信息增益、增益率、基尼指数
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树

划分选择 — 样本纯度

- 决策树学习的目标：提升 结点纯度
  - 一般而言，随着划分过程不断进行，我们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
- ◆ 信息熵度量样本集合纯度最常用的一种指标
  - 划分选择：信息增益、增益率
- ◆ 基尼值也度量样本集合纯度
  - 划分选择：基尼指数

划分选择 — 信息熵

信息熵

- 度量样本集合纯度最常用的一种指标
- 当前样本集合D 中，第 k 类样本所占的比例为 p_k $(k=1,2,\ldots,|Y|)$，则 D 的信息熵定义为

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

  - 约定：若 $p=0$，则 $p\log_2 p=0$

划分选择 — 信息熵

信息熵

- 度量样本集合纯度最常用的一种指标
- 当前样本集合D 中，第 k 类样本所占的比例为 p_k $(k=1,2,\ldots,|Y|)$，则 D 的信息熵定义为

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

  - 约定：若 $p=0$，则 $p\log_2 p=0$

- 信息熵的最小值为 0，最大值为$\log_2|Y|$
- 信息熵值越小，则当前样本集合 D的纯度越高。

  即，当前样本集合 D，属于同一类别的可能性越大

划分选择 — 信息熵

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

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

- 数据集包含17个训练样本

$|Y|=2$

正例 $p_1 = 8/17$，反例 $p_2 = 9/17$，

划分选择 — 信息熵

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

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

- 数据集包含17个训练样本

$|Y|=2$

正例 $p_1=8/17$，反例 $p_2=9/17$，

◆根结点的信息熵

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

- 信息熵值越小，纯度越高，属于同一类别可能性越大
- 信息熵大，纯度低

划分选择

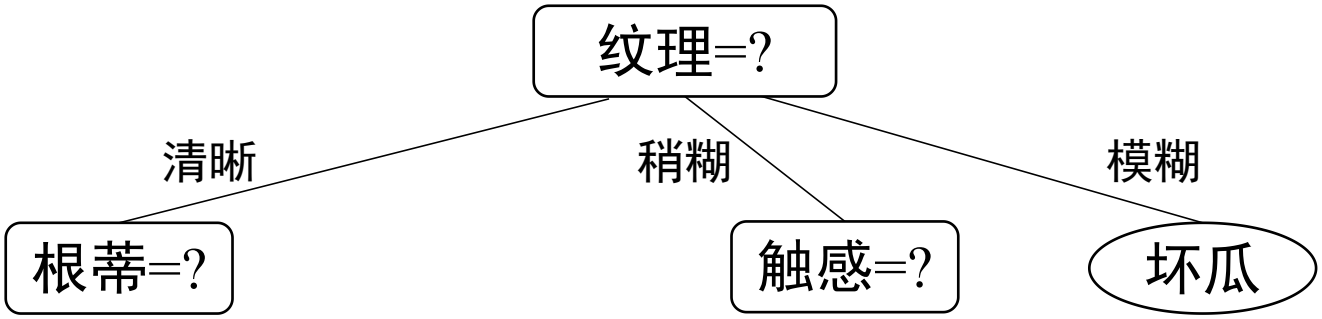
- 信息熵：度量纯度最常用的一种指标

- 属性划分 的方法
  - 信息增益
  - 增益率

划分选择 — 信息增益

- 离散属性 a 有 V 个可能的取值 $\{a^1, a^2, \ldots, a^V\}$
  - 离散属性 a=纹理 有 V=3 个可能的取值

    $\{a^1$ = 清晰, a^2 = 稍糊, a^3 = 模糊 $\}$



划分选择 — 信息增益

- 离散属性 a 有 V 个可能的取值 $\{a^1, a^2, \ldots, a^V\}$
  - 用属性a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。

信息增益

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

划分选择 — 信息增益

- 离散属性 a 有 V 个可能的取值 $\{a^1, a^2, \ldots, a^V\}$
  - 用属性a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。

信息增益

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

信息增益 为分支结点权重

- 说明：样本数越多的分支结点，对决策树的影响越大

划分选择 — 信息增益

- 离散属性 a 有 V 个可能的取值 $\{a^1, a^2, \ldots, a^V\}$，用属性$a$对样本集$D$进行划分，获得**信息增益**

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

- **信息增益越大**，则使用属性a来进行划分，所获得的**纯度提升越大**。
- 当前样本集合D中，**同一类**的样本比例，**增加**的**最快**，更有利于样本集合D中，所有样本趋于同一类

- **ID3决策树**学习算法，以**信息增益**为准则来选择**划分属性**

划分选择 — 信息增益

- 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）

- 用“色泽”划分后，所获得的 3 个分支结点的信息熵为：

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

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

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

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

- 属性“色泽”的信息增益为

$$\mathrm{Gain}(D,\text{色泽})=\mathrm{Ent}(D)-\sum_{v=1}^{3}\frac{|D^v|}{|D|}\cdot\mathrm{Ent}(D^v)$$

$$=0.998-\left(\frac{6}{17}\times1.000+\frac{6}{17}\times0.918+\frac{5}{17}\times0.722\right)=0.109$$

划分选择 — 信息增益

- 类似的，可计算所有属性的**信息增益**为

$$\mathrm{Gain}(D,\text{色泽})=\mathrm{Ent}(D)-\sum_{v=1}^{3}\frac{|D^v|}{|D|}\cdot\mathrm{Ent}(D^v)=0.109$$

$$\mathrm{Gain}(D,\text{根蒂})=0.143\quad \mathrm{Gain}(D,\text{敲声})=0.141\quad \mathrm{Gain}(D,\text{纹理})=0.381$$

$$\mathrm{Gain}(D,\text{脐部})=0.289\quad \mathrm{Gain}(D,\text{触感})=0.006$$

划分选择 — 信息增益

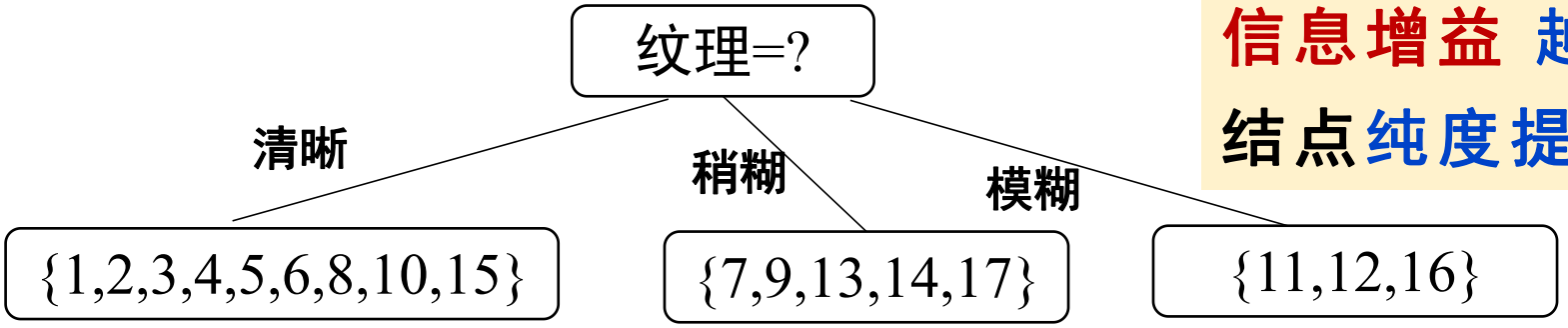
- 类似的，可计算所有属性的信息增益为

$$\text{Gain}(D,\text{色泽}) = \text{Ent}(D) - \sum_{v=1}^{3} \frac{|D^v|}{|D|} \cdot \text{Ent}(D^v) = 0.109$$

$$\text{Gain}(D,\text{根蒂}) = 0.143 \quad \text{Gain}(D,\text{敲声}) = 0.141 \quad \text{Gain}(D,\text{纹理}) = 0.381$$

$$\text{Gain}(D,\text{脐部}) = 0.289 \quad \text{Gain}(D,\text{触感}) = 0.006$$

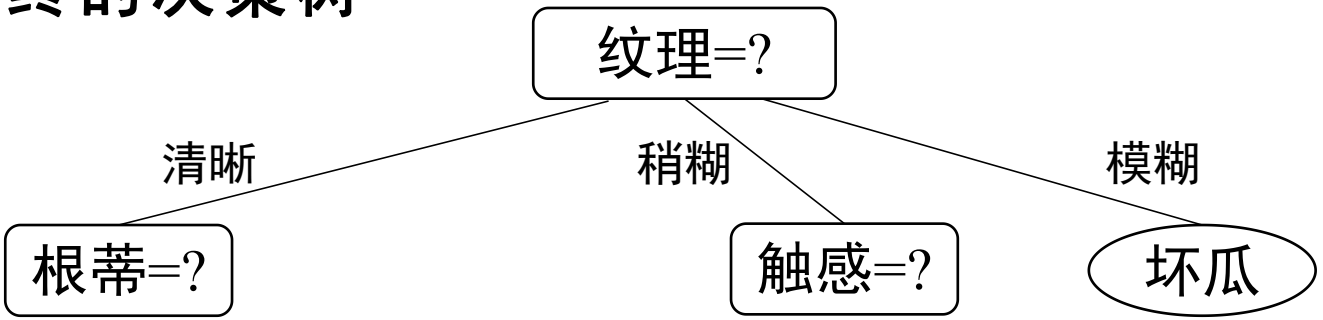
- 属性“纹理”的信息增益最大，被选为 划分属性
  - 基于“纹理”，对根结点进行划分
  - 各分支结点，包含了对应的样本子集。



信息增益 越大
结点纯度提升越大

划分选择 — 信息增益

- 对每个分支结点，计算其信息增益，做进一步划分，得到最终的决策树

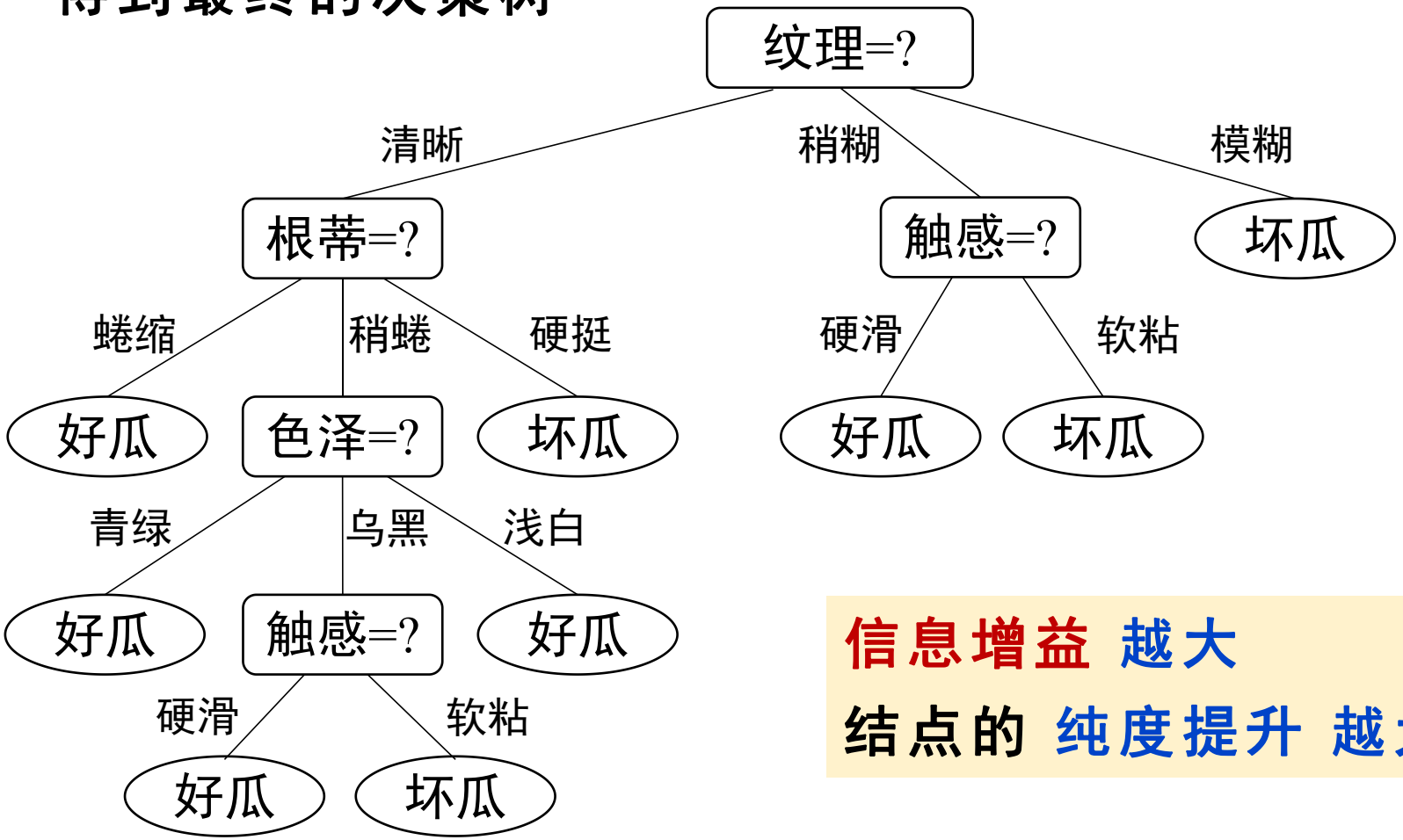


信息增益 越大

结点的 纯度提升 越大

划分选择 — 信息增益

- 对每个分支结点，计算其信息增益，做进一步划分，得到最终的决策树



信息增益 越大

结点的 纯度提升 越大

划分选择 — 信息增益

存在的问题

- 若把“编号”作为划分属性，将产生17 个分支，每个分支结点仅包含一个样本，则信息增益为0.998，远大于其他属性的信息增益，这些分支结点的纯度已达最大。
- 这样的决策树，不具有泛化能力，无法对新样本进行有效预测。

划分选择 — 信息增益

存在的问题

- 若把“编号”作为划分属性，将产生17 个分支，每个分支结点仅包含一个样本，则信息增益为0.998，远大于其他属性的信息增益，这些分支结点的纯度已达最大。
- 这样的决策树，不具有泛化能力，无法对新样本进行有效预测。

信息增益：对 属性值 数目 较多 的属性，有所偏好

- 编号——有17个属性值，纹理——有3个属性值
- 信息增益 偏好 编号。

划分选择 — 信息增益

存在的问题

- 若把“编号”作为划分属性，将产生17 个分支，每个分支结点仅包含一个样本，则信息增益为0.998，远大于其他属性的信息增益，这些分支结点的纯度已达最大。
- 这样的决策树，不具有泛化能力，无法对新样本进行有效预测。

信息增益：对 属性值 数目 较多 的属性，有所偏好

- 为减少这种偏好可能带来的不利影响，著名的C4.5决策树算法，不直接使用信息增益，使用“增益率”来选择 最优 划分属性。

划分选择

- 信息熵：度量纯度最常用的一种指标

- 属性划分 的方法
  - 信息增益
  - 增益率

划分选择 —增益率

增益率

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

- IV(a) 称为属性 a 的“固有值”
  - 属性 a 的属性值数目 越多，则IV(a)的值通常就越大，则 增益率 越小

划分选择 —增益率

增益率

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

- IV(a) 称为属性 a 的“固有值”
  - 属性 a 的属性值数目 越多，则IV(a)的值通常就越大，则 增益率 越小

存在的问题

增益率：对 属性值 数目 较少 的属性，有所偏好

- 编号有17个属性值，纹理有3个属性值
- 增益率偏好 纹理

划分选择 —增益率

□存在的问题

信息增益：对 属性值 数目 较多 的属性，有所偏好

增益率：对 属性值 数目 较少 的属性，有所偏好

◆C4.5决策树，不直接选择增益率最大的候选划分属性，使用 一个启发式 选择最优划分属性：

1. 先，从候选划分属性中找出信息增益高于 平均水平的属性
2. 再，从中选取增益率最高的作为最优划分属性

划分选择

- 决策树学习的目标：提升 结点纯度
  - 信息熵 度量样本集合 纯度
    - 划分选择：信息增益、增益率
  - 基尼值 度量样本集合 纯度
    - 划分选择：基尼指数

划分选择 —基尼指数

数据集D 的纯度可用“基尼值”来度量

$$\mathrm{Gini}(D)=\sum_{k=1}^{|\mathrm{Y}|}\sum_{k\neq k'} p_k \cdot p_{k'} = 1-\sum_{k=1}^{|\mathrm{Y}|} p_k^{\,2}$$

- 从D中随机抽取两个样本，其类别标记不一致的概率
- 基尼值 Gini(D) 越小，当前样本集D，纯度越高

属性a 的基尼指数

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

◆ 选择使划分后基尼指数最小的属性作为最优划分属性

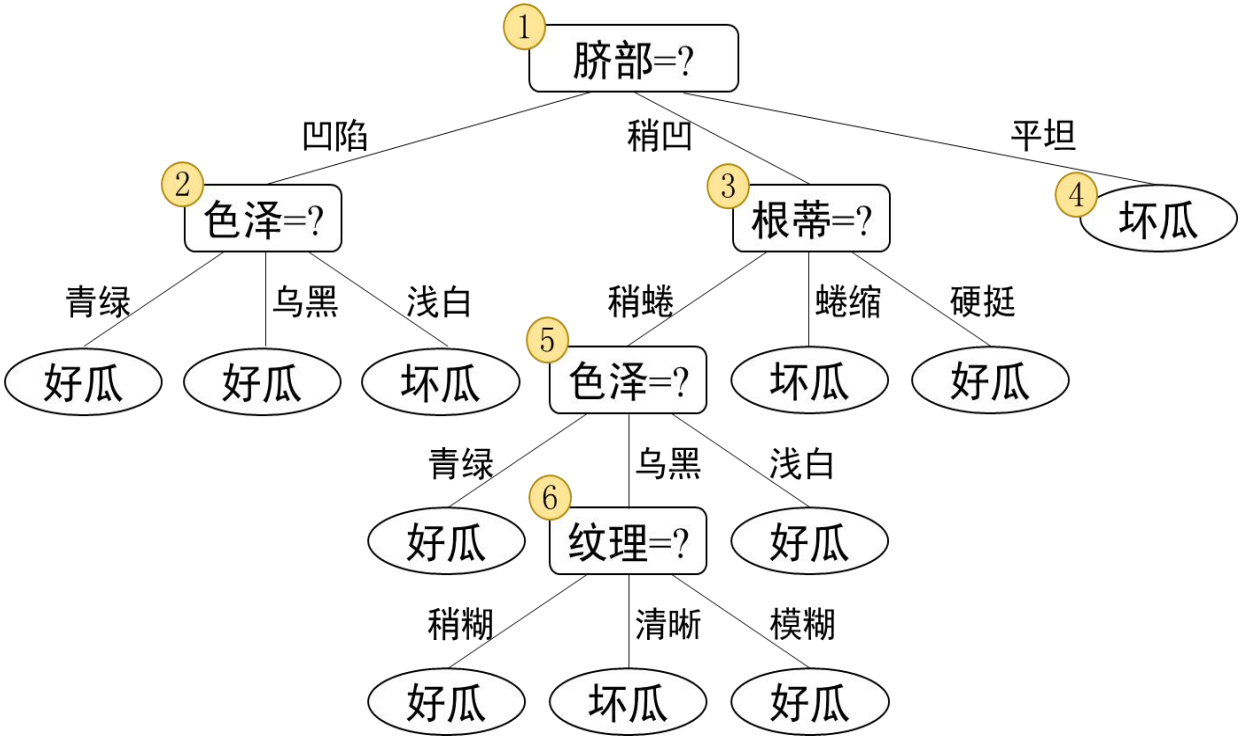
◆ CART，采用“基尼指数”来选择划分属性

第四章 决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
     - 预剪枝、后剪枝
3. 连续与缺失值
4. 多变量决策树

剪枝处理

未剪枝决策树



- **完全决策树**：6个属性，每个属性有3个属性值，叶子结点数为$6!*3^6$

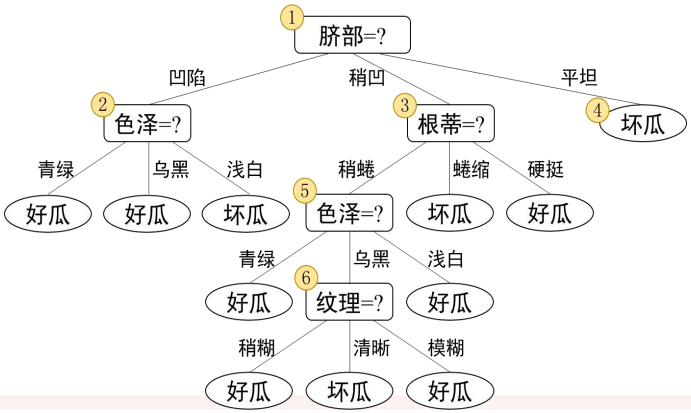
剪枝处理

为什么剪枝

- 决策树学习中，为了尽可能正确分类训练样本，结点划分过程将不断重复，有时会造成决策树分支过多
- 决策分支过多，把训练集自身的一些特点，当做所有数据都具有的一般性质，导致过拟合
- “剪枝”，即主动去掉一些分支，降低 过拟合的风险

剪枝的基本策略

- 预剪枝
- 后剪枝



剪枝处理

- 剪枝过程中，如何判断 决策树泛化性能 是否提升？该不该 剪枝？
- **留出法**：预留一部分数据用作“验证集”，进行性能评估，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

剪枝处理

- 生成决策树时，对每个结点在划分前，先估计泛化性能是否提高，若划分后：
  - 泛化性能 不能提升
  - 泛化性能 提升

剪枝处理

- 生成决策树时，对每个结点在划分前，先估计泛化性能是否提高，若划分后：
  - 泛化性能 不能提升，则 停止划分
    - ✓ 当前结点，标记为叶结点
    - ✓ 类别标记为：训练样例数最多的类别
  - 泛化性能 提升，能提高验证集精度，则 划分
    - ✓ 对划分后的属性，执行同样判断

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。

1

7个瓜, 标记: 好瓜

验证集精度

划分前 42.9%

结点1

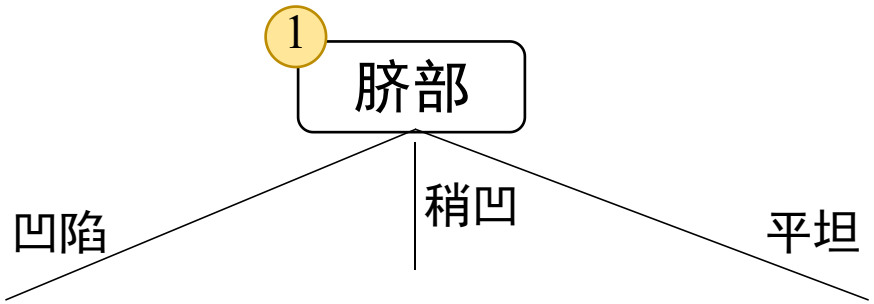
- 不划分 / 划分前，该结点为叶结点，类别标记为好瓜。
- 使用验证集进行验证，3个好瓜(标记正确)，验证集精度为 42.9%（3/7）

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



验证集精度

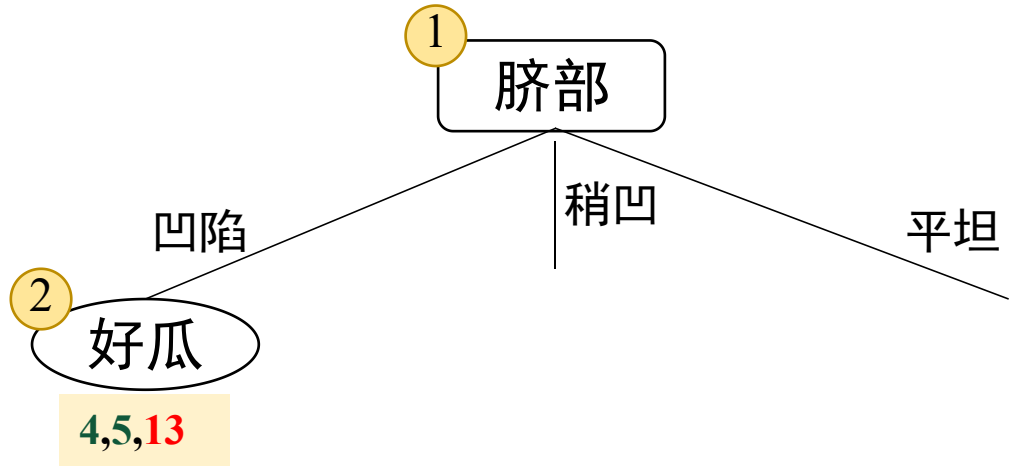
划分后 ？

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



4,5,13

验证集精度

划分后 ？

结点2

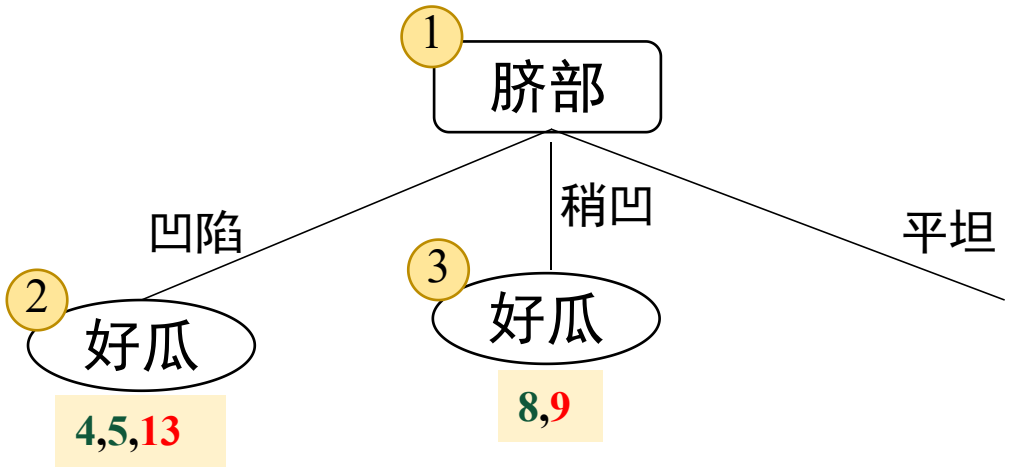
- 包含样本 4,5,13
- 其中，2个好瓜(4,5)，结点2标记为“好瓜” 。

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



验证集精度

划分后 ？

结点3

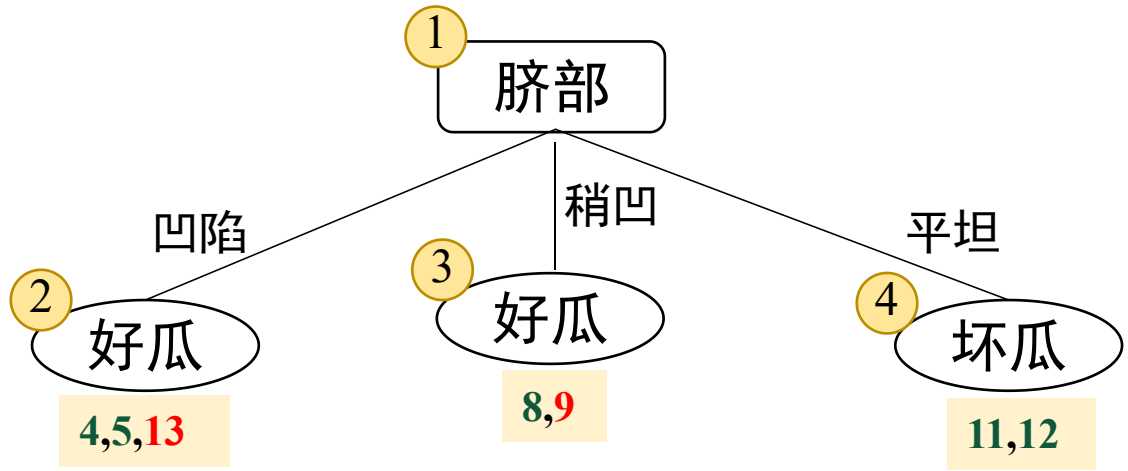
- 包含样本 8,9。标记为“好瓜” 。

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



验证集精度

划分后 ？

结点4

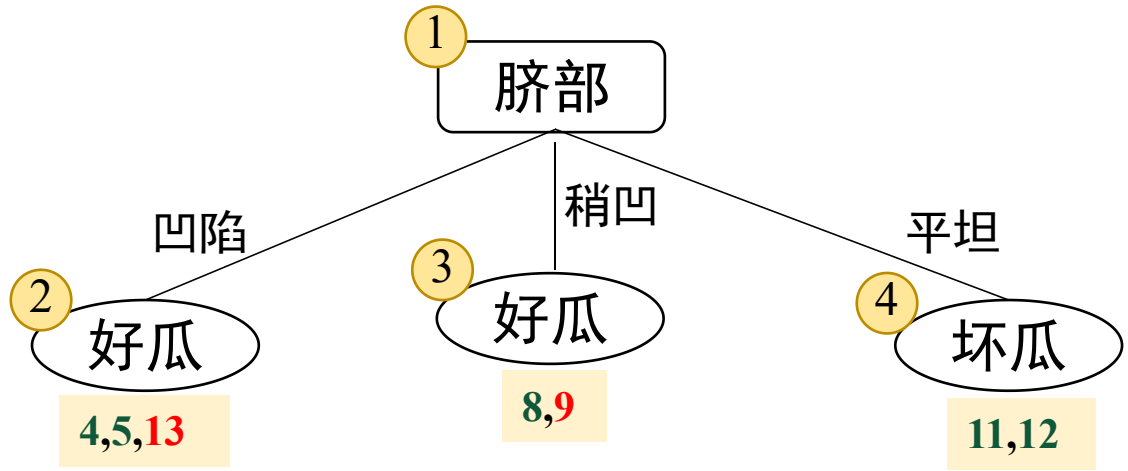
- 包含样本 11,12。标记为“坏瓜” 。

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



验证集精度

划分后 71.4%

- 若划分，4,5,8,11,12 标记正确
- 验证集精度为 5/7×100%=71.4%

剪枝处理 — 预剪枝

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

- 选取“脐部”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。

验证集精度

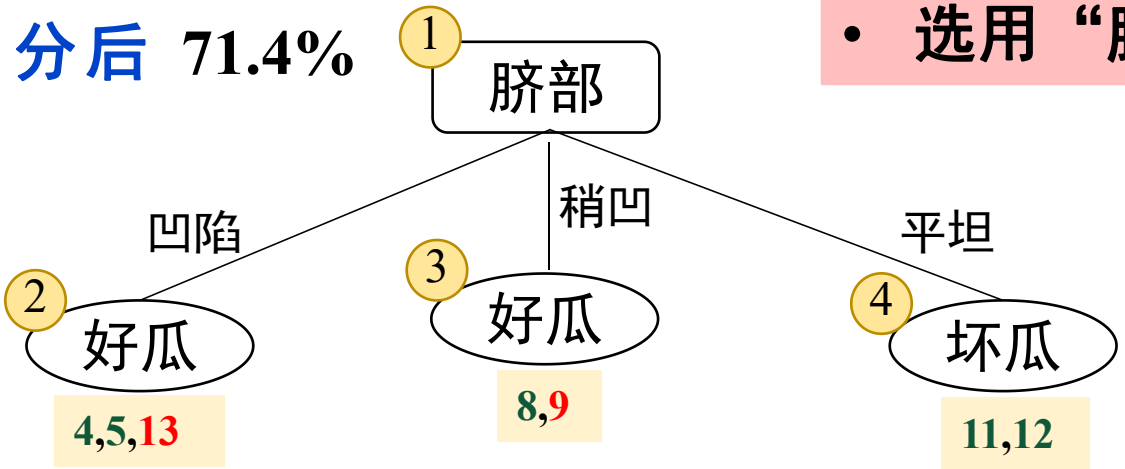
划分前 42.9%

1 7个瓜, 标记: 好瓜

脐部=?

- 预剪枝决策: 划分
- 选用“脐部”作为划分属性

划分后 71.4%

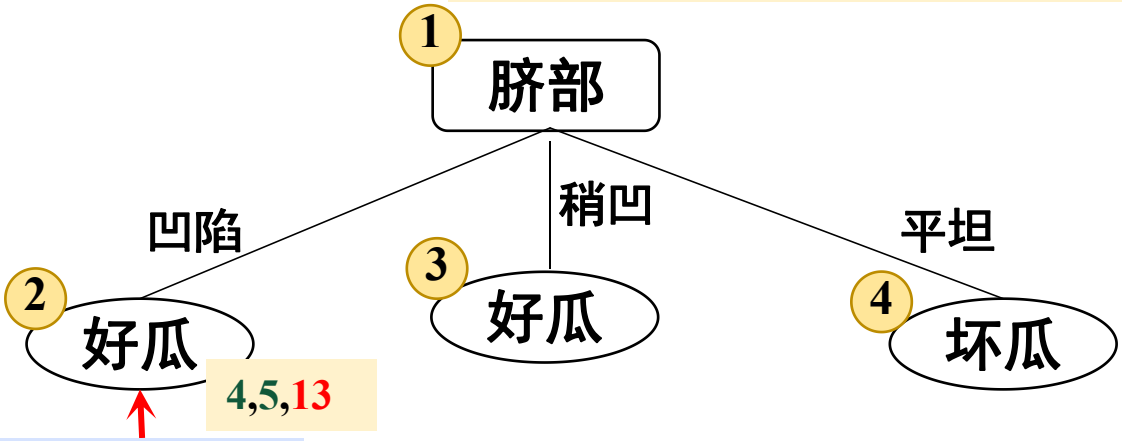


剪枝处理 — 预剪枝

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

- 选取“色泽”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



色泽=？ 验证集精度

划分前：71.4%

划分后：57.1%

预剪枝决策

验证集精度下降

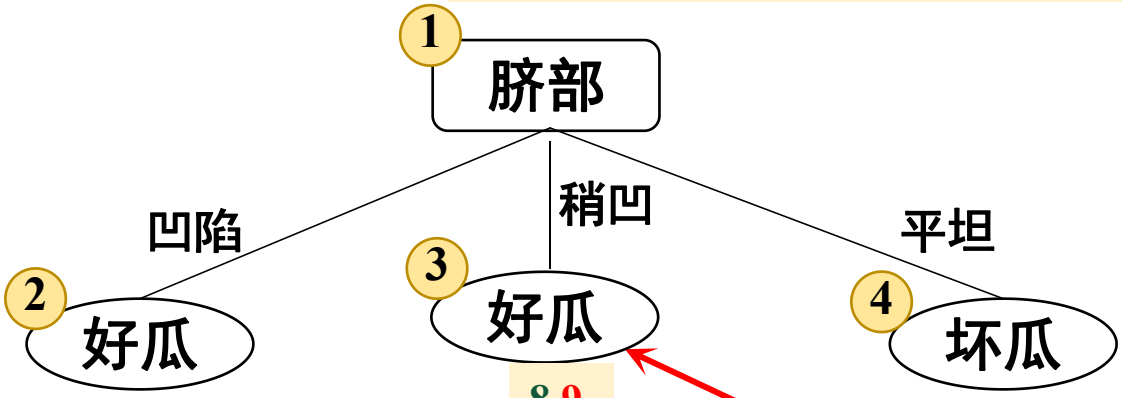
禁止划分

剪枝处理 — 预剪枝

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

- 选取“根蒂”划分训练集
- 计算 划分前（该结点为叶结点）
- 计算划分后 的验证集精度
- 判断是否需要划分。



根蒂=？

验证集精度

划分前：71.4%

划分后：71.4%

预剪枝决策

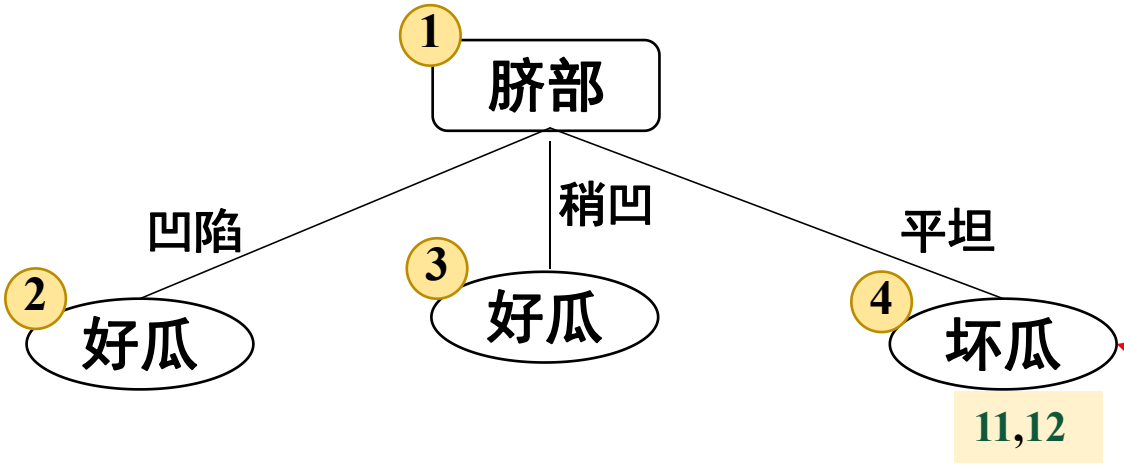
划分不能提升验证集精度

禁止划分

剪枝处理 — 预剪枝

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

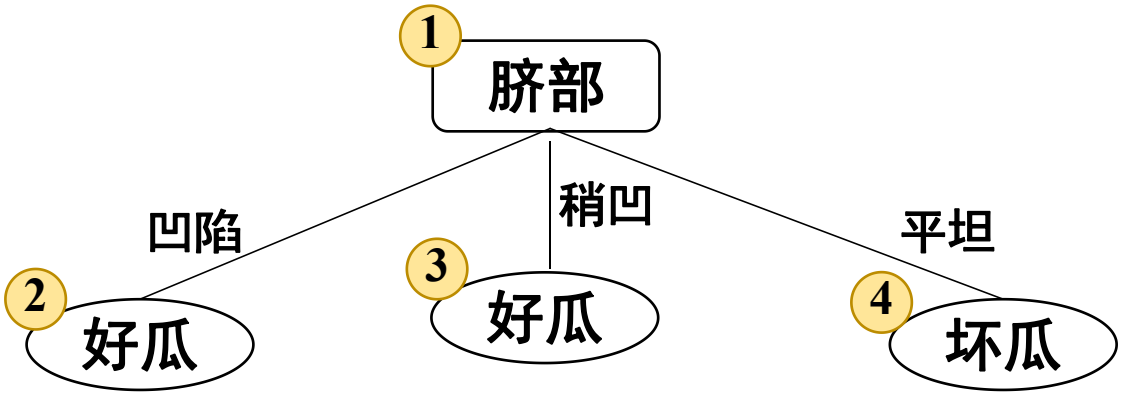


所含训练样例已属于同一类，不再进行划分

剪枝处理 — 预剪枝

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



- 计算 划分前/后的验证集精度，判断是否需要划分。
- 最终，得到仅有一层划分的决策树，称为“决策树桩”

剪枝处理 — 预剪枝

- 预剪枝决策树很多分支，没有“展开”

优点

- 降低了过拟合风险
- 显著减少了 训练和测试时间开销

剪枝处理 — 预剪枝

- 预剪枝决策树很多分支，没有“展开”

优点

- 降低了过拟合风险
- 显著减少了 训练和测试时间开销

缺点

- 带来了 欠拟合风险
  - 有些分支的当前划分，虽不能提升泛化性能
  - 但在其基础上继续划分，却可能导致整体性能显著提高
  - 预剪枝基于“贪心”本质，禁止这些分支展开，带来了欠拟合风险

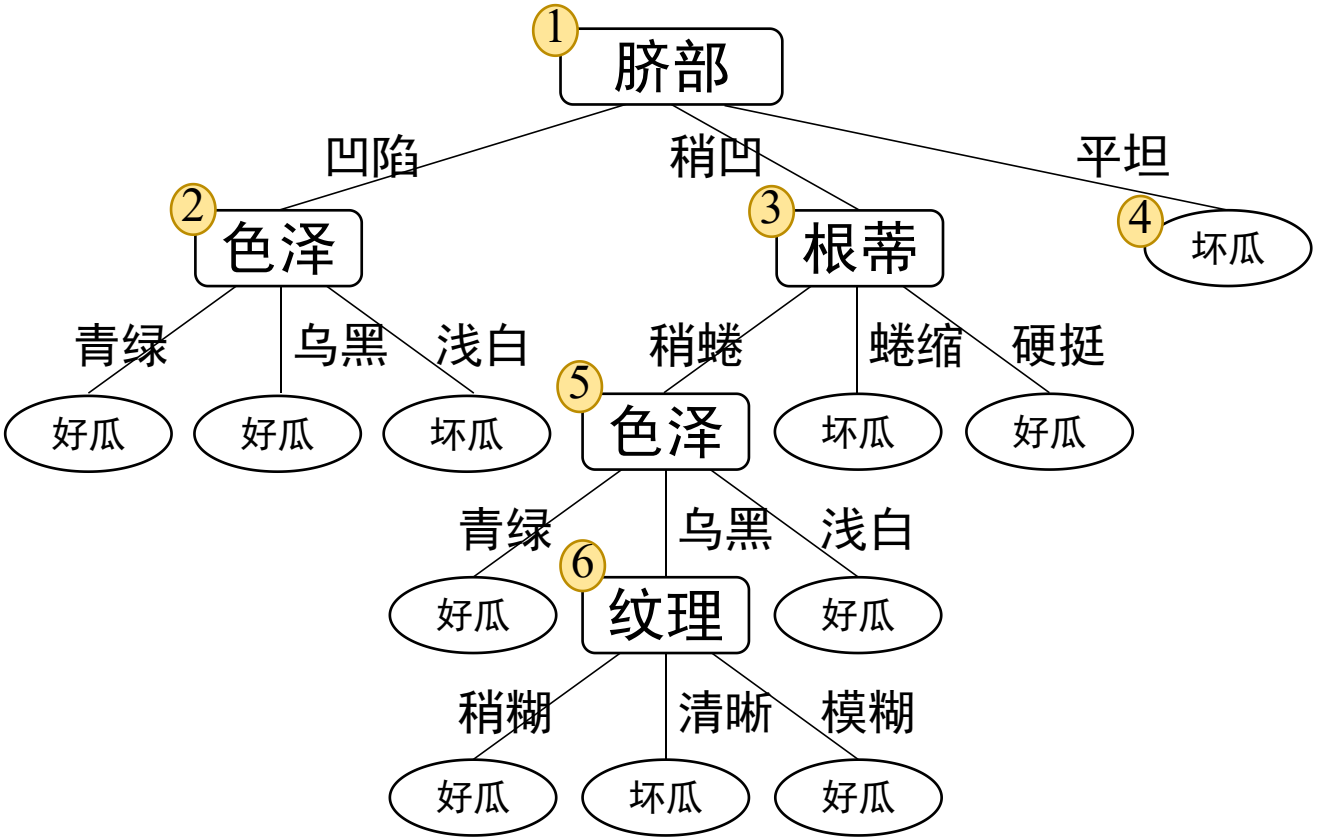
第四章　决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
     - 预剪枝、后剪枝
3. 连续与缺失值
4. 多变量决策树

剪枝处理 — 后剪枝

1. 首先，根据训练集，生成一棵完整的决策树，该决策树的验证集精度为42.9%

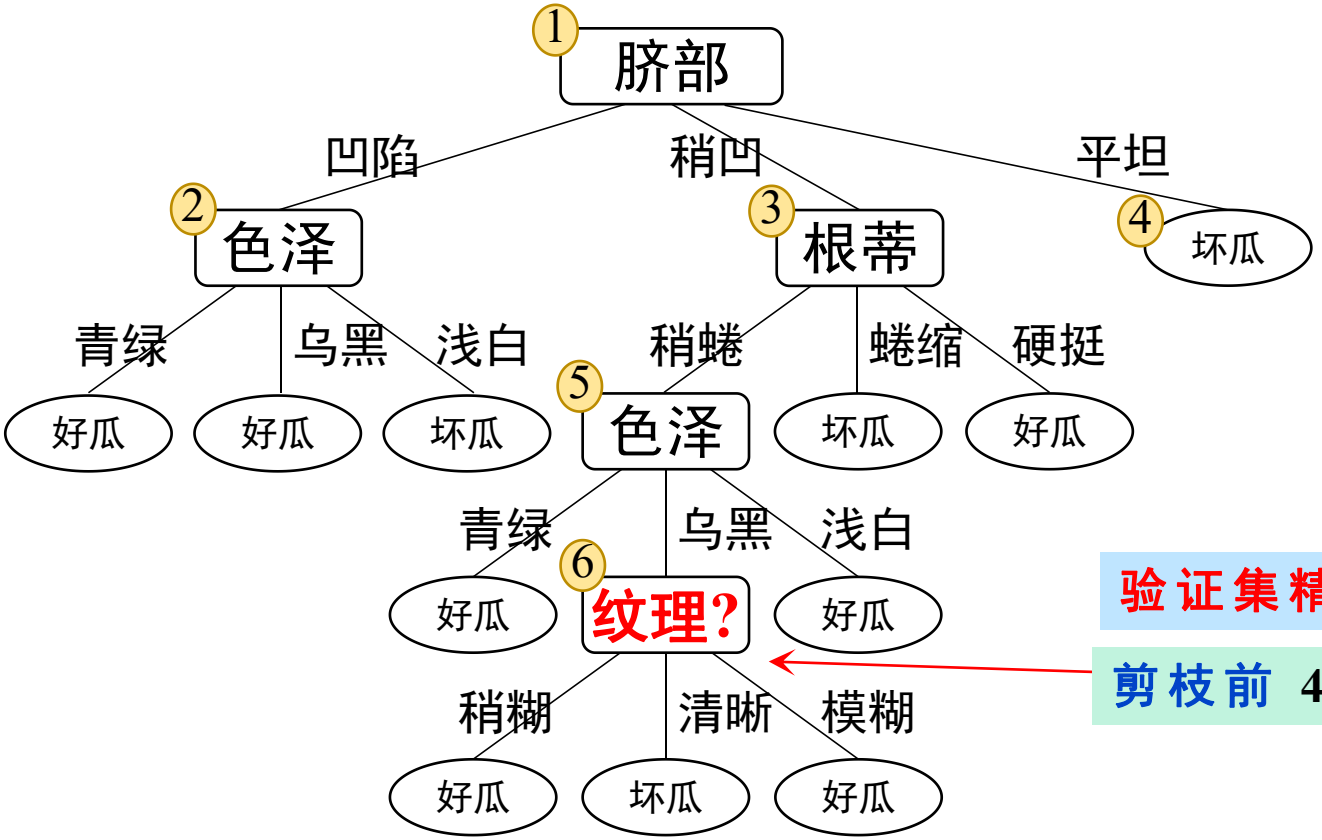
2. 然后，自底向上地对非叶结点进行考察。



剪枝处理 — 后剪枝

◆ 结点6

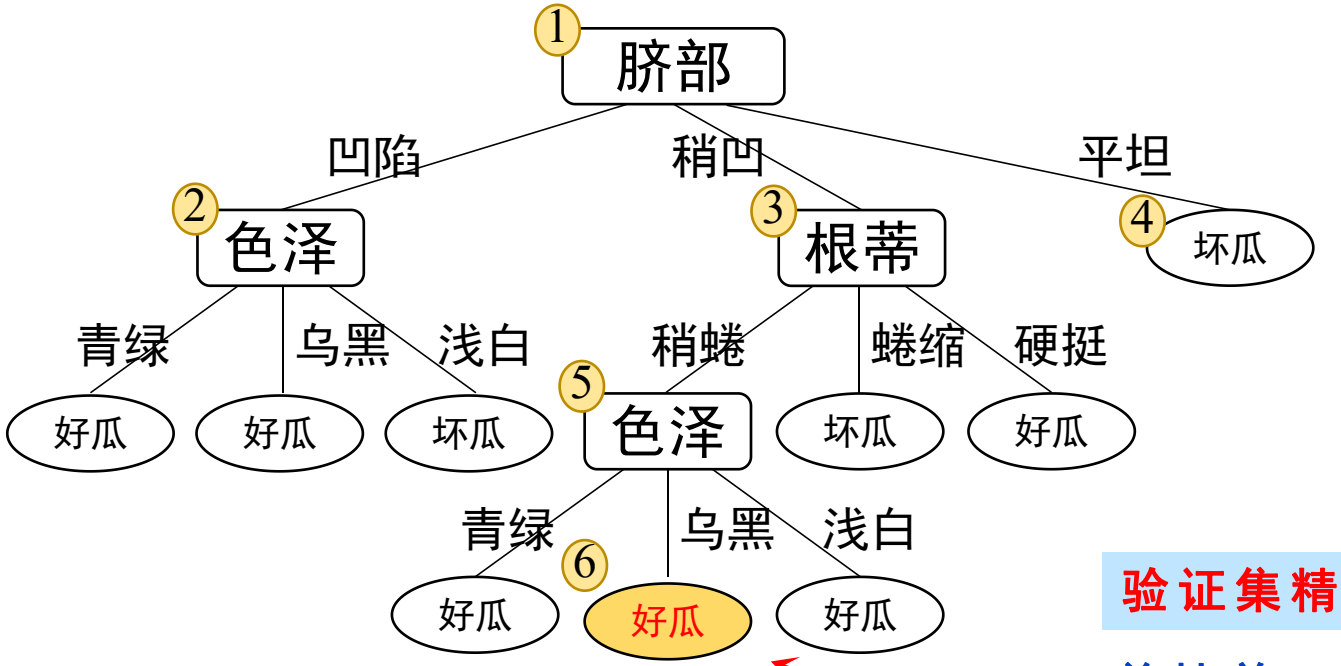
✓ 初始的决策树，验证集精度为 42.9%



剪枝处理 — 后剪枝

◆ 结点6

✓ 若将其**替换**为**叶结点**，根据落在其上的训练样本，将其标记为**好瓜**。验证集精度提高至 57.1%



落在其上的训练样本中
好瓜数目＞坏瓜数目
将其标记为“好瓜”

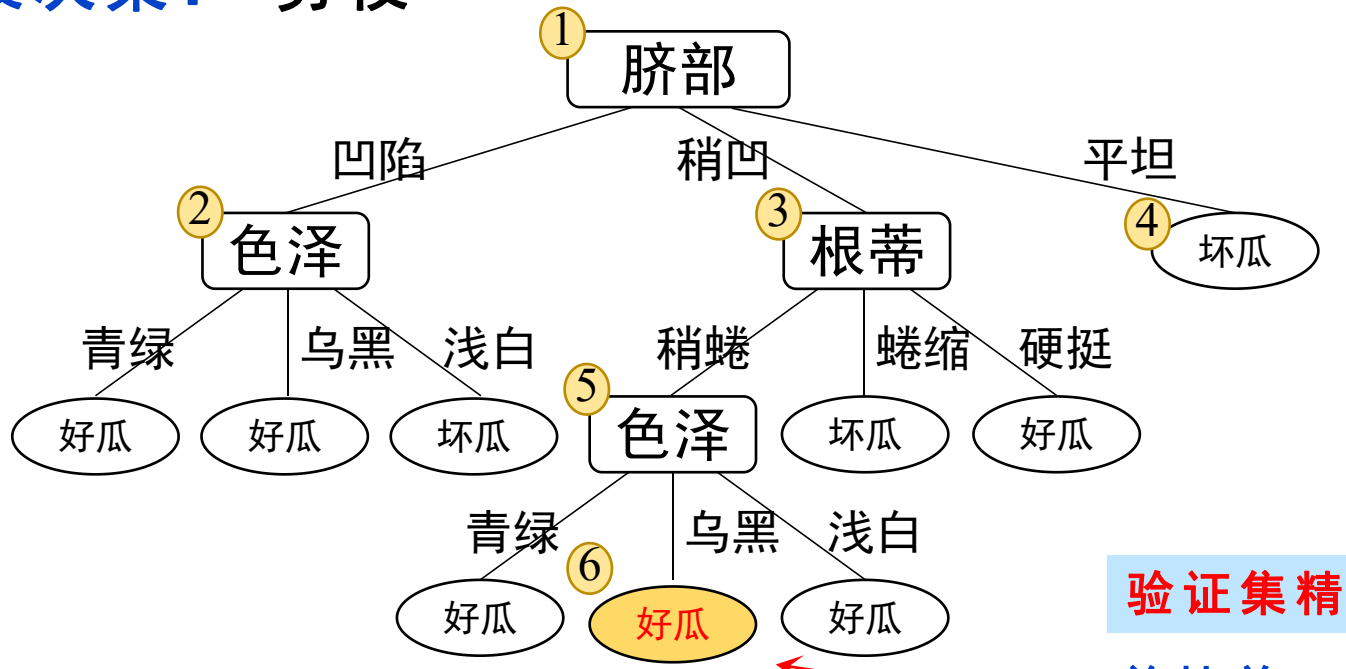
验证集精度

剪枝前 42.9 %

剪枝后 57.1 %

剪枝处理 — 后剪枝

- 结点6
  - ✓ 剪枝前，42.9%；剪枝后，提高至 57.1%
  - ✓ 后剪枝决策： 剪枝



验证集精度

剪枝前 42.9 %

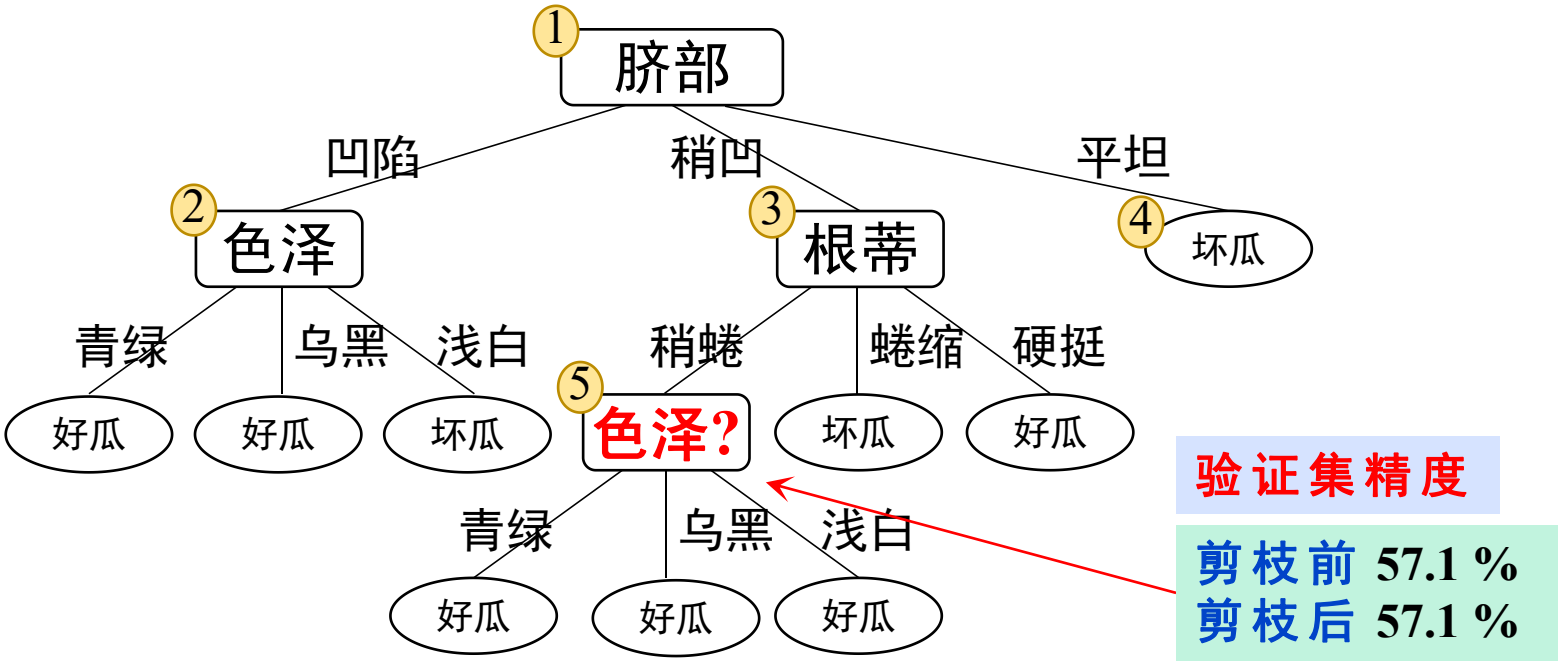
剪枝后 57.1 %

后剪枝决策：剪枝

剪枝处理 — 后剪枝

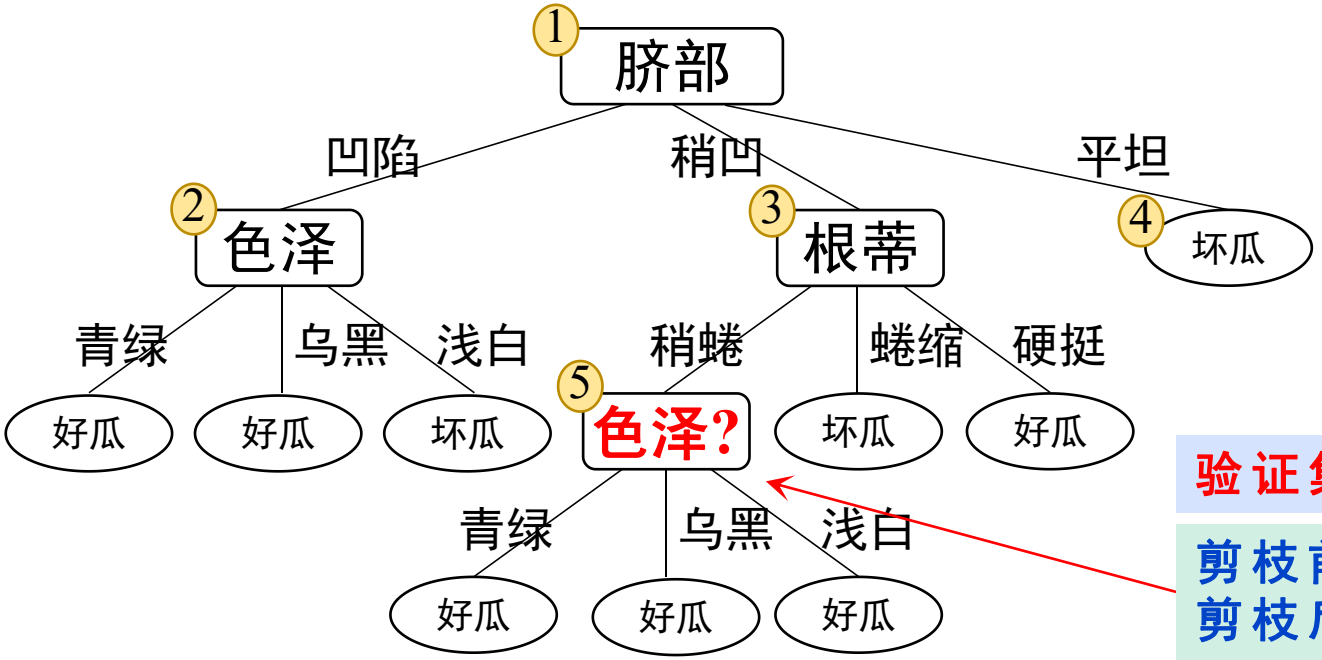
结点 5

- ✓ 初始的决策树，验证集精度为 57.1%
- ✓ 若将其替换为叶结点，根据落在其上的训练样本，将其标记为“好瓜”，得到验证集精度 仍为 57.1%



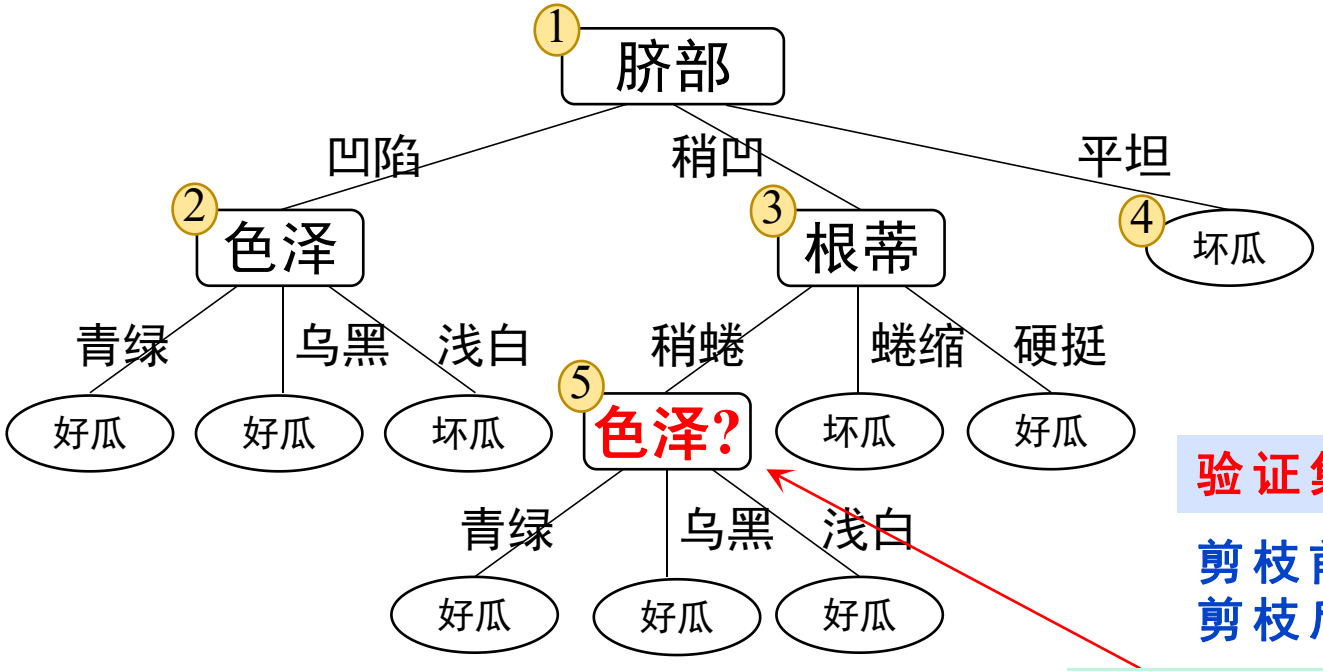
剪枝处理 — 后剪枝

- 结点 5
  - ✓ 剪枝前/后，验证集精度均为 57.1%



剪枝处理 — 后剪枝

- 结点 5
  - ✓ 剪枝前/后，验证集精度均为 57.1%
  - ✓ 后剪枝决策： 剪枝
    - 奥卡姆剃刀准则：多个模型满足条件，选择简单的那个



验证集精度

剪枝前 57.1 %
剪枝后 57.1 %

后剪枝决策：剪枝

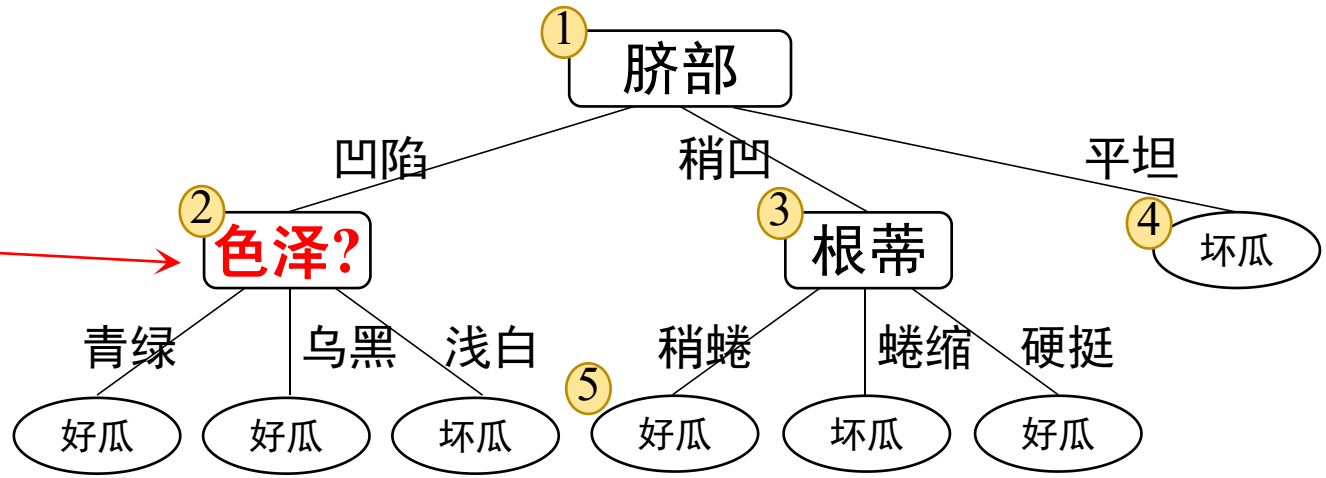
剪枝处理 — 后剪枝

◆ 结点 2

✓ 剪枝前，验证集精度为 57.1%。

验证集精度

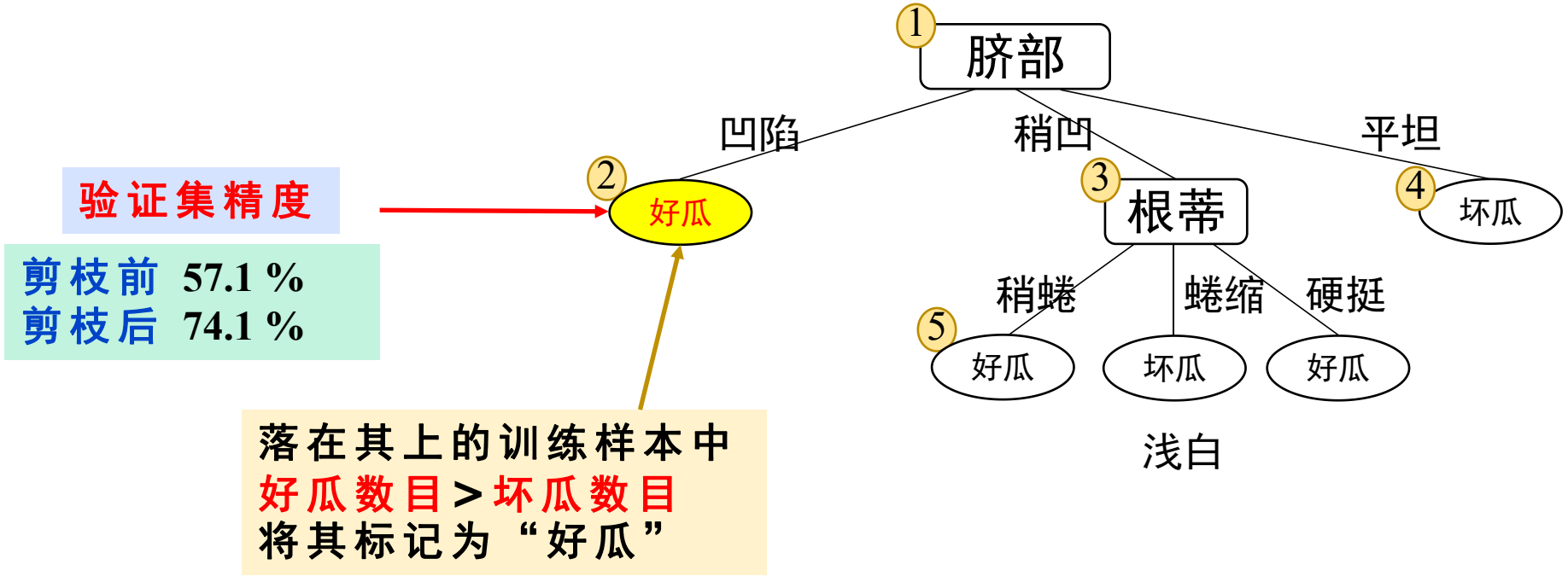
剪枝前 57.1 %



剪枝处理 — 后剪枝

◆ 结点 2

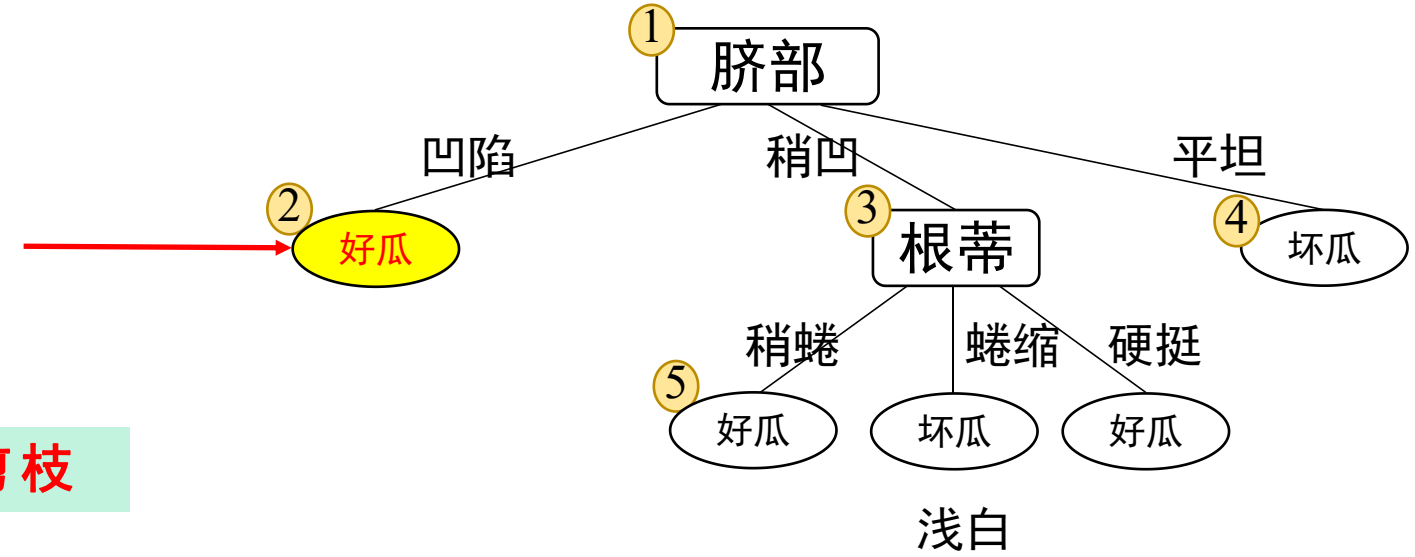
- ✓ **剪枝前**，验证集精度为 57.1%。
- ✓ **剪枝后**，根据落在其上的训练样本，将其标记为“好瓜”。验证集精度提高至 74.1%



剪枝处理 — 后剪枝

◆ 结点 2

- ✓ 剪枝前，验证集精度 57.1%。剪枝后，提高至 74.1%
- ✓ 后剪枝决策： 剪枝



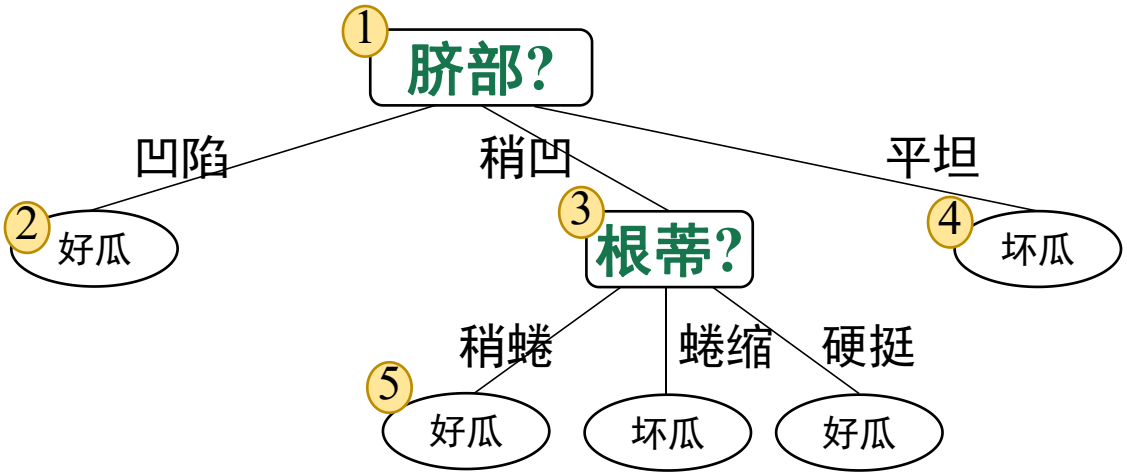
验证集精度

剪枝前 57.1 %
剪枝后 74.1 %

后剪枝决策：剪枝

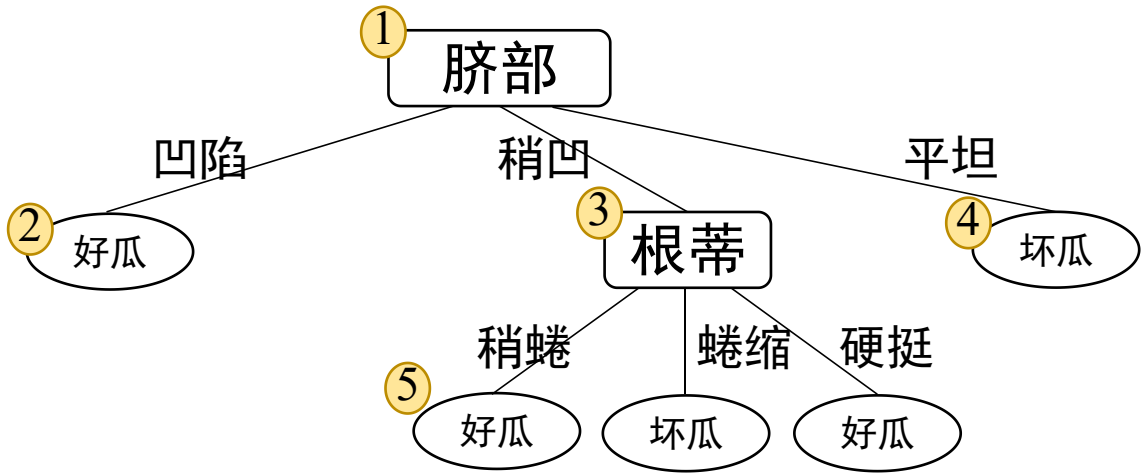
剪枝处理 — 后剪枝

- 结点 3 和 结点 1
  - ✓ 剪枝前/后，验证集精度降低，则分支得到保留



剪枝处理 — 后剪枝

- 最终，基于后剪枝策略得到的决策树：



剪枝处理 — 后剪枝

后剪枝

◆ 优点

- 后剪枝比预剪枝，保留了更多的分支，欠拟合风险小
- 后剪枝决策树 泛化性能 往往 优于 预剪枝决策树

剪枝处理 — 后剪枝

后剪枝

◆ 优点

- 后剪枝比预剪枝，保留了更多的分支，欠拟合风险小
- 后剪枝决策树 泛化性能 往往 优于 预剪枝决策树

◆ 缺点

- 训练时间开销大

  后剪枝过程，是在生成完全决策树之后进行的，需要自底向上，对所有非叶结点逐一考察

第四章　决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树

连续与缺失值— 连续值

连续值 处理

- 现实学习任务中，常会遇到连续属性，其可取值数目不再有限。
- 使用离散化技术将连续属性转化为离散属性

连续与缺失值— 连续值

连续值 处理

- 现实学习任务中，常会遇到连续属性，其可取值数目不再有限。
- 使用离散化技术（二分法）对连续属性进行处理，转化为离散属性
  - 属性“密度”，其候选划分点集合包含17个候选值

$$T_{\text{密度}} = \{0.244, 0.294, 0.351, 0.381, 0.420, 0.459, 0.518, 0.574, 0.600, 0.621, 0.636, 0.648, 0.661, 0.681, 0.708, 0.746\}$$

- 使密度信息增益最大的划分点为 0.381

连续与缺失值— 连续值

连续值 处理

- 现实学习任务中，常会遇到连续属性，其可取值数目不再有限。
- 使用离散化技术（二分法）对连续属性进行处理，转化为离散属性
  - 与离散属性不同，若当前结点划分属性为连续属性，该属性，还可作为其后代结点的划分属性

连续与缺失值— 缺失值

缺失值 处理

- 现实任务中，常会遇到不完整样本，即样本的某些属性值缺失
  - 在医疗领域，由于诊测成本、隐私保护等因素，患者的医疗数据在某些属性上的取值未知
  - 如果简单地放弃不完整样本，仅使用无缺失值的样本来进行学习，显然是对数据信息极大的浪费

连续与缺失值— 缺失值

缺失值 处理

◆ Q 1：如何在属性缺失的情况下 进行划分属性选择？

- 将信息增益的计算式推广为

$$\mathrm{Gain}(D,a)=\rho\times\mathrm{Gain}(\tilde{D},a)=\rho\times\left(\mathrm{Ent}(\tilde{D})-\sum_{v=1}^{V}\tilde{r}_v\cdot\mathrm{Ent}(\tilde{D}^v)\right)$$

连续与缺失值— 缺失值

缺失值 处理

◆ Q 1：如何在属性缺失的情况下 进行划分属性选择？

- 将信息增益的计算式推广为

$$\mathrm{Gain}(D,a)=\rho\times\mathrm{Gain}(\tilde{D},a)=\rho\times\left(\mathrm{Ent}(\tilde{D})-\sum_{v=1}^{V}\tilde{r}_v\cdot\mathrm{Ent}(\tilde{D}^v)\right)$$

◆ Q 2：若样本在该属性上的值缺失，如何对样本进行划分？

- 若样本 x 在划分属性 a 上的取值未知，则将 x 以不同的概率划入到不同的子结点中去，样本权值 在与属性值 a^v 对应的子结点中 调整为 $r_v w_x$

连续与缺失值— 缺失值

缺失值 处理

◆ 计算出所有属性在数据集上的信息增益

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

$$\text{Gain}(D,\text{根蒂})=0.171 \quad \text{Gain}(D,\text{敲声})=0.145 \quad \text{Gain}(D,\text{纹理})=0.424$$

$$\text{Gain}(D,\text{脐部})=0.289 \quad \text{Gain}(D,\text{触感})=0.006$$

- 在所有属性中，纹理的信息增益最大，被用于对根结点进行划分

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

连续与缺失值— 缺失值

缺失值 处理

◆计算出所有属性在数据集上的信息增益

- 纹理 的信息增益最大，被用于对根结点进行划分

根据划分结果，分别进入

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

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

连续与缺失值— 缺失值

缺失值 处理

◆计算出所有属性在数据集上的信息增益

- 纹理 的信息增益最大，被用于对根结点进行划分

根据划分结果，分别进入

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

样本8和10 在属性“纹理”上出现缺失值

样本8和10 同时进入3个分支，调整8和10在3分支权值分别为 7/15, 5/15, 3/15

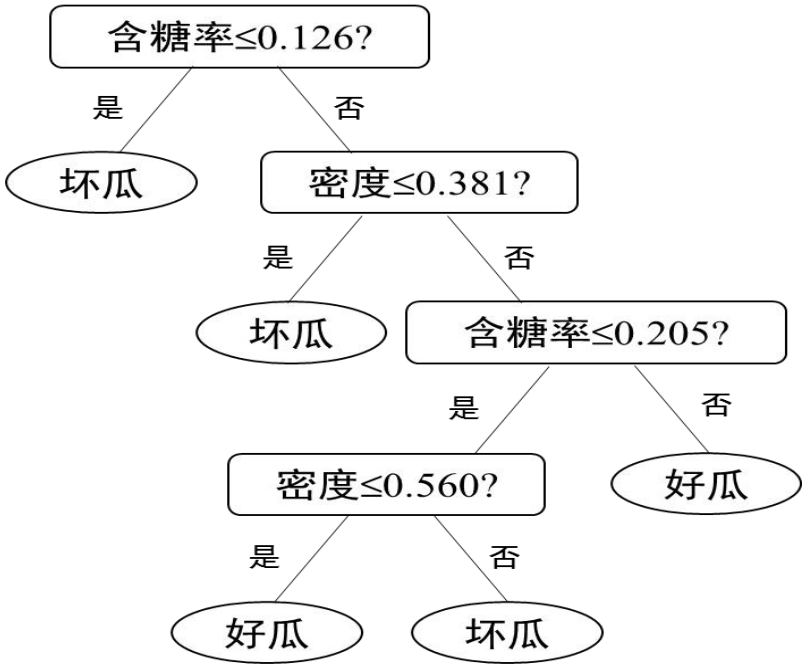
编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	–	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	–	是
3	乌黑	蜷缩	–	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	–	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	–	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	–	稍凹	硬滑	是
9	乌黑	–	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	–	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	–	否
12	浅白	蜷缩	–	模糊	平坦	软粘	否
13	–	稍蜷	浊响	稍糊	凹陷	硬滑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

第四章　决策树

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树

多变量决策树

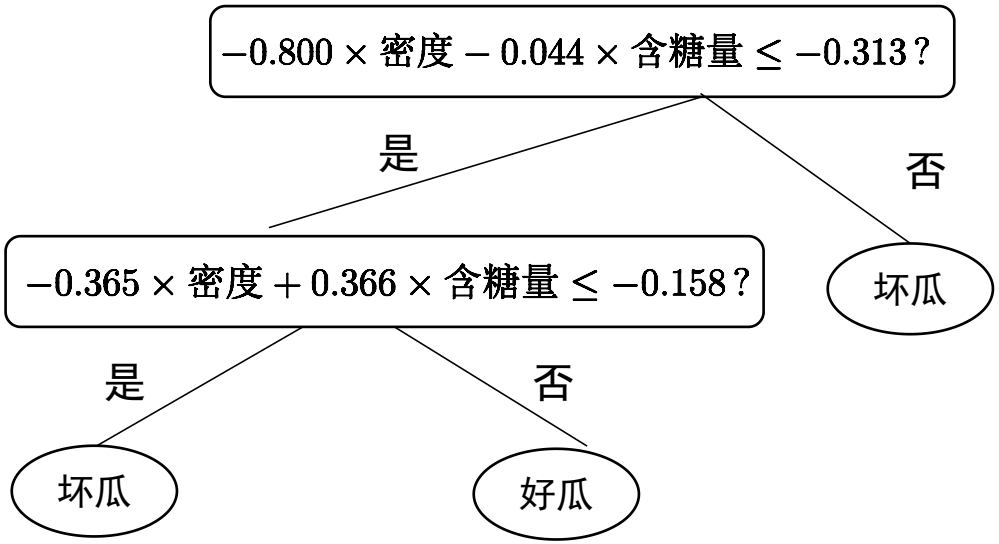
单变量决策树



非叶节点

对**某个**属性进行测试

多变量决策树



非叶节点

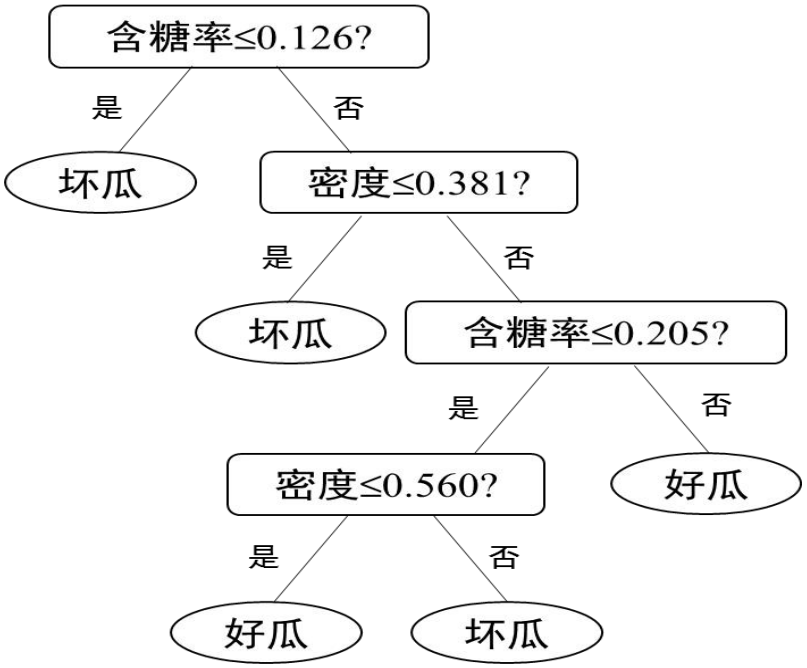
对**若干个**属性的线性组合进行测试

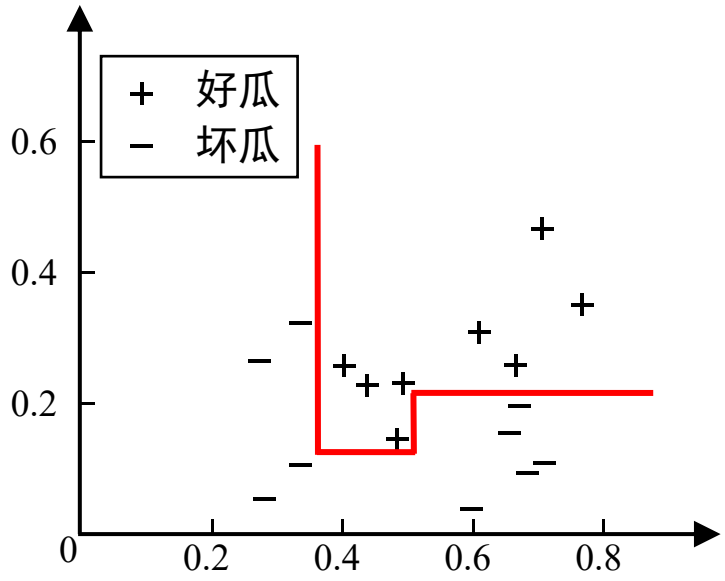
多变量决策树

单变量决策树

◆分类边界 轴平行

- 分类边界，由若干个与坐标轴平行的分段组成
- 每一段划分，都直接对应了某个属性取值

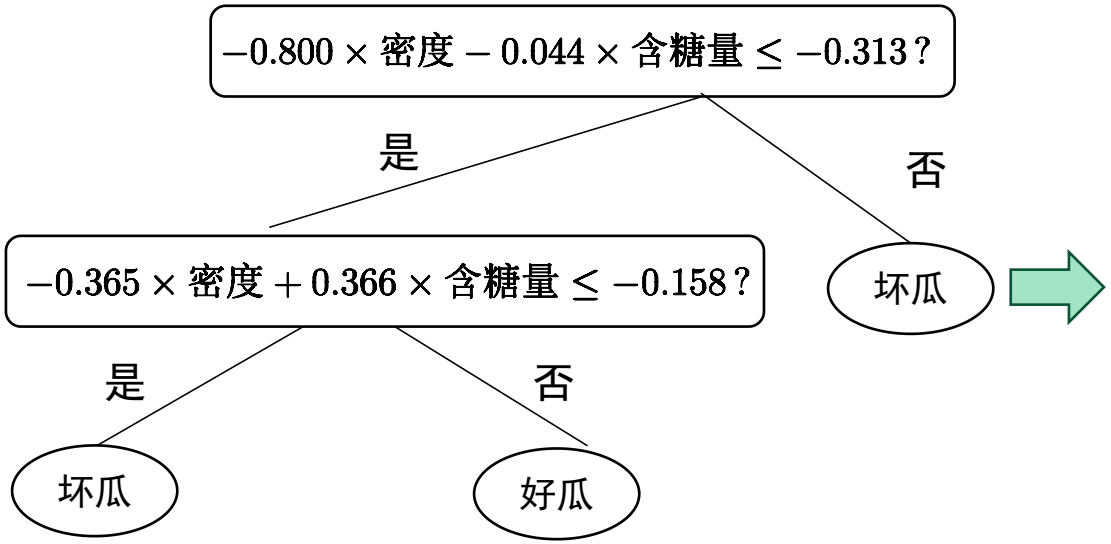


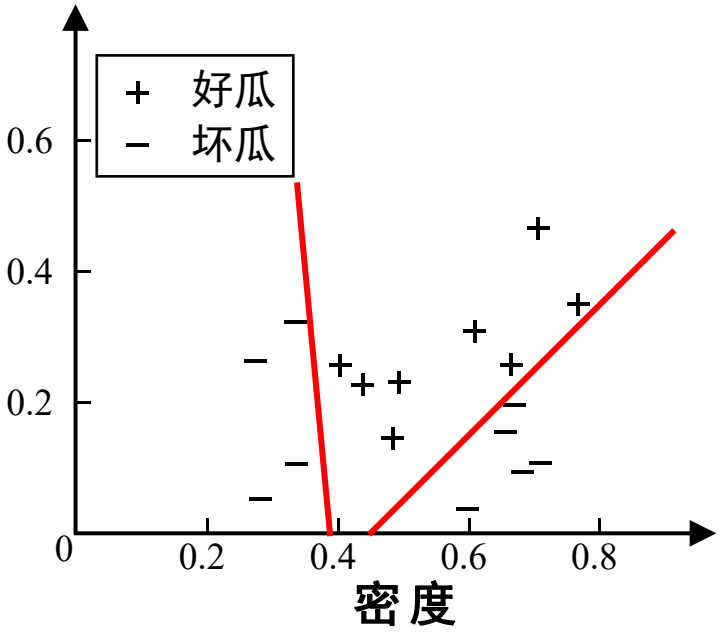


多变量决策树

多变量决策树

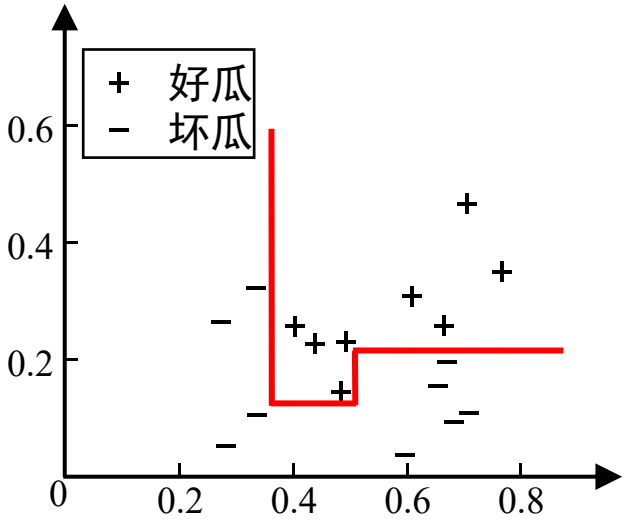
- 不是 为每个非叶结点，寻找一个最优划分属性
- 而是 试图建立，一个合适的线性分类器 $\sum_{i=1}^{d} w_i a_i = t$





多变量决策树

单变量决策树



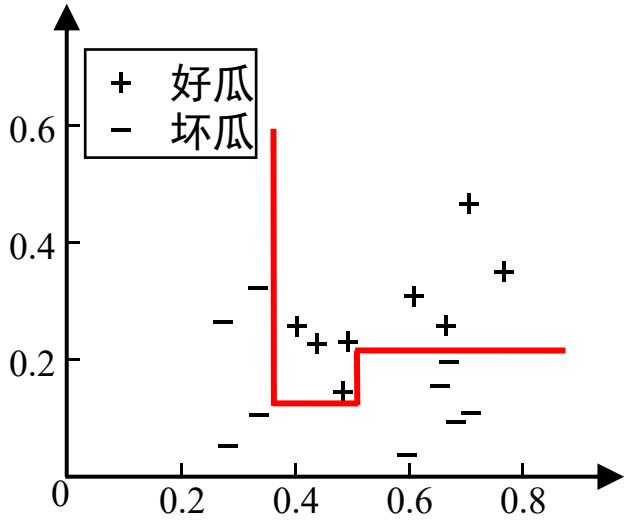
非叶节点

对某个属性进行测试

多变量决策树

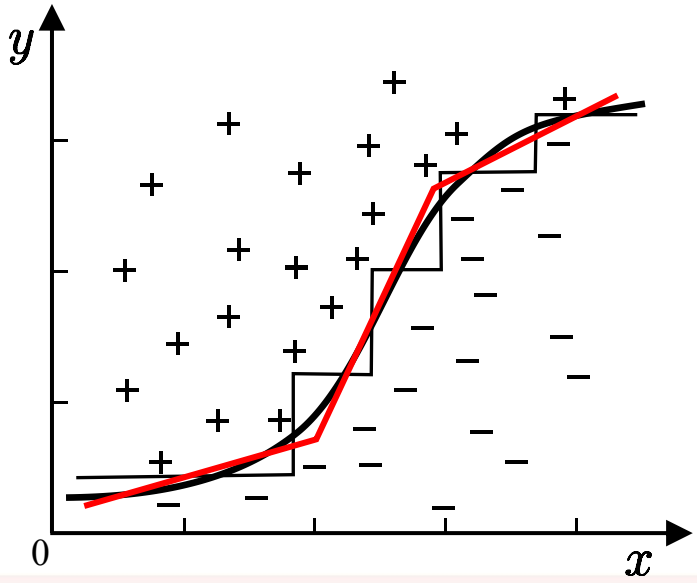
多变量决策树

单变量决策树



非叶节点
对**某个**属性进行测试

多变量决策树



非叶节点
对**若干个**属性的
线性组合进行测试

第四章　决策树总结

1. 基本流程
2. 核心技术
   - 划分选择
   - 剪枝处理
3. 连续与缺失值
4. 多变量决策树