机器学习
Machine Learning

北京航空航天大学计算机学院
School of Computer Science and Engineering, Beihang University
刘庆杰　陈佳鑫

2025年春季学期
Spring 2024

部分内容来源于C. Bishop和A. NG等人的课程以及互联网资源

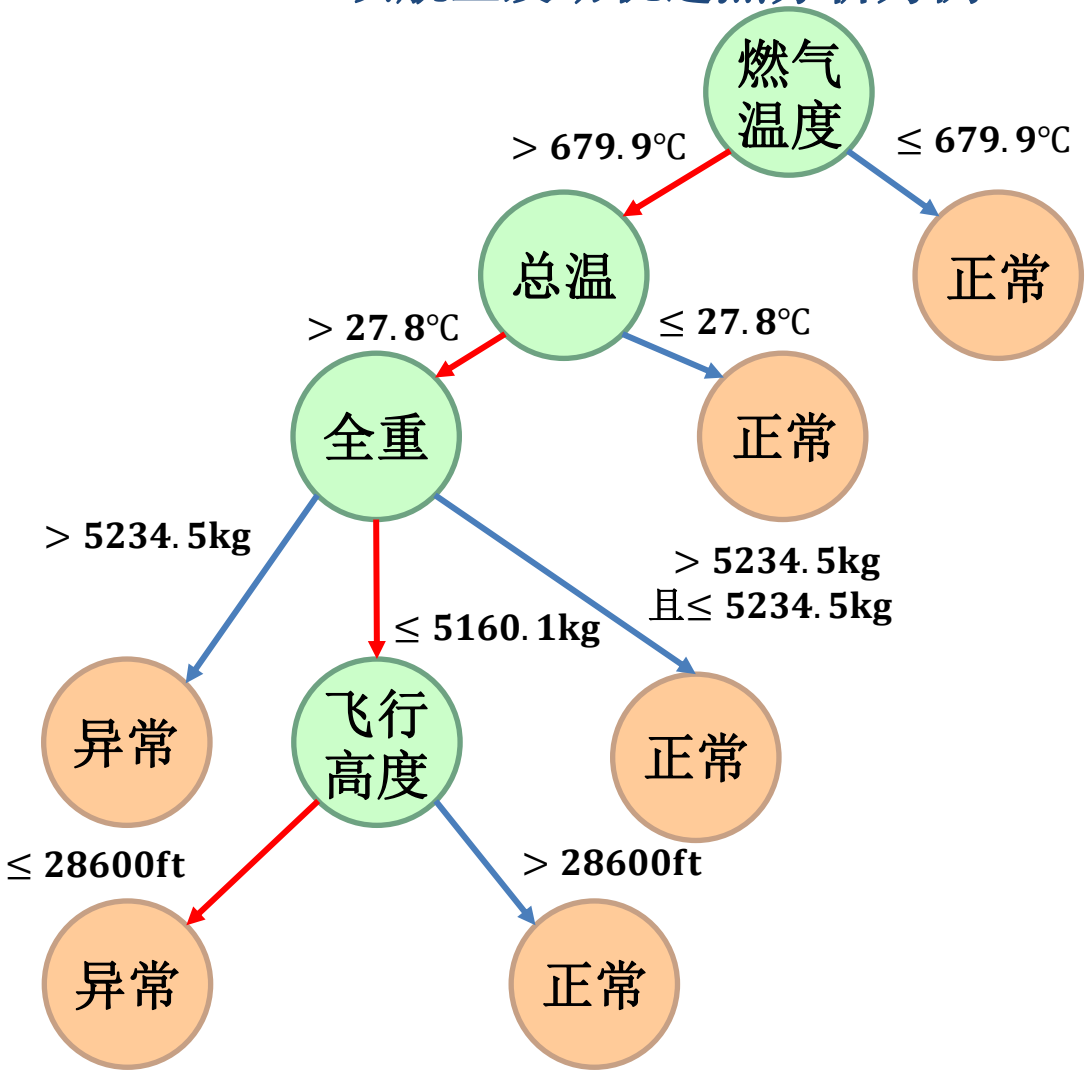
12.1 什么是决策树?

- 决策树的概念
- 决策树的构建
- 决策树的应用

决策树的概念

- **决策树 (Decision Tree)**
  - 是一种**树型结构**，由结点和有向边组成
  - 结点
    - **根结点**对应全部训练样本
    - **内部结点**表示一个属性或特征，对应满足从根结点到该结点所有条件的训练样本
    - **叶结点**代表一种类别
  - 有向边
    - **有向边**代表一个测试输出

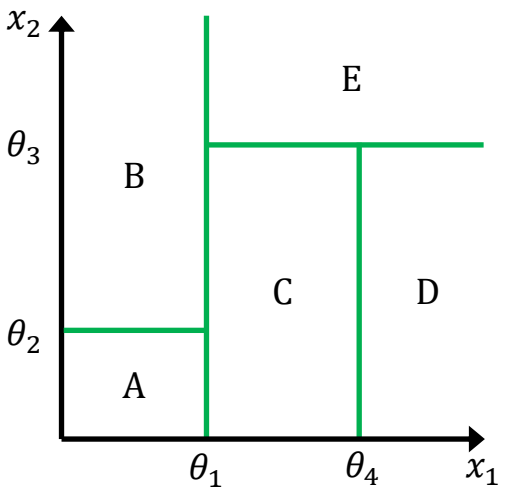
以航空发动机过热分析为例

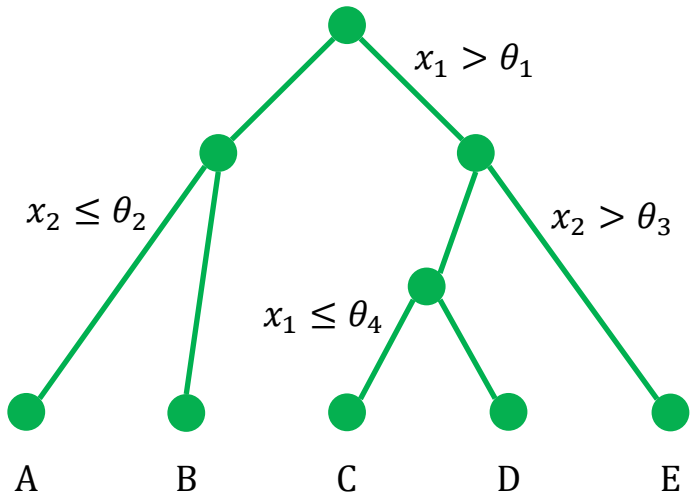


关雪飞,杨军,马小兵. 决策树在航空发动机状态数据分析中的应用[C]. //中国现场统计研究会第十三届学术年会论文集. 2007:156-159.

决策树的概念

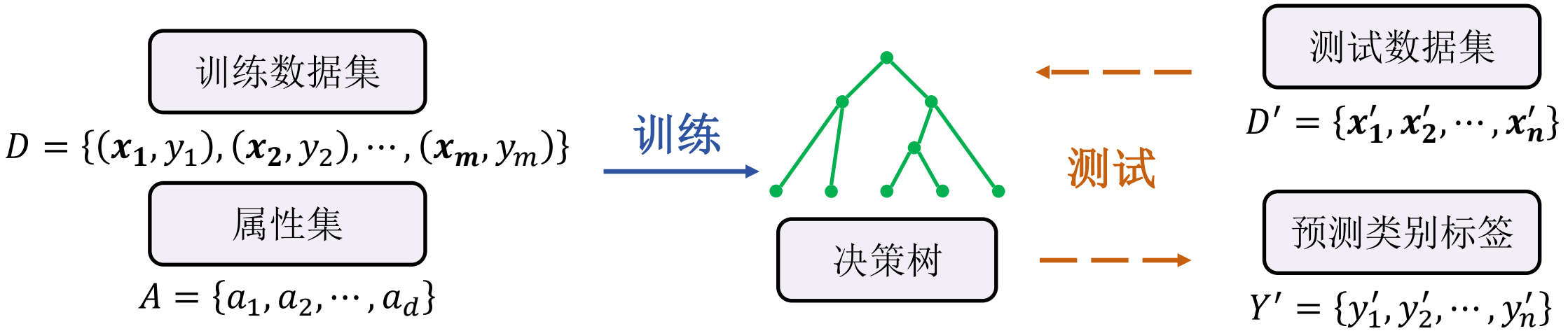
- **基本思想**
  - 采用**自顶向下**的递归方法，以信息熵为度量构建一棵熵值下降最快的树，到叶结点处的熵值为零，此时每个叶结点中的实例都属于同一类
    - 决策树具有直观的可视化形式，类似于人类的决策过程，易于理解与解释
    - 决策树可以看成一个**if-then规则集合**，根结点到叶结点的每一条路径构建一条规则
    - 决策树将特征空间划分为**互不相交**的单元或区域，并在每个单元定义一个类的概率分布





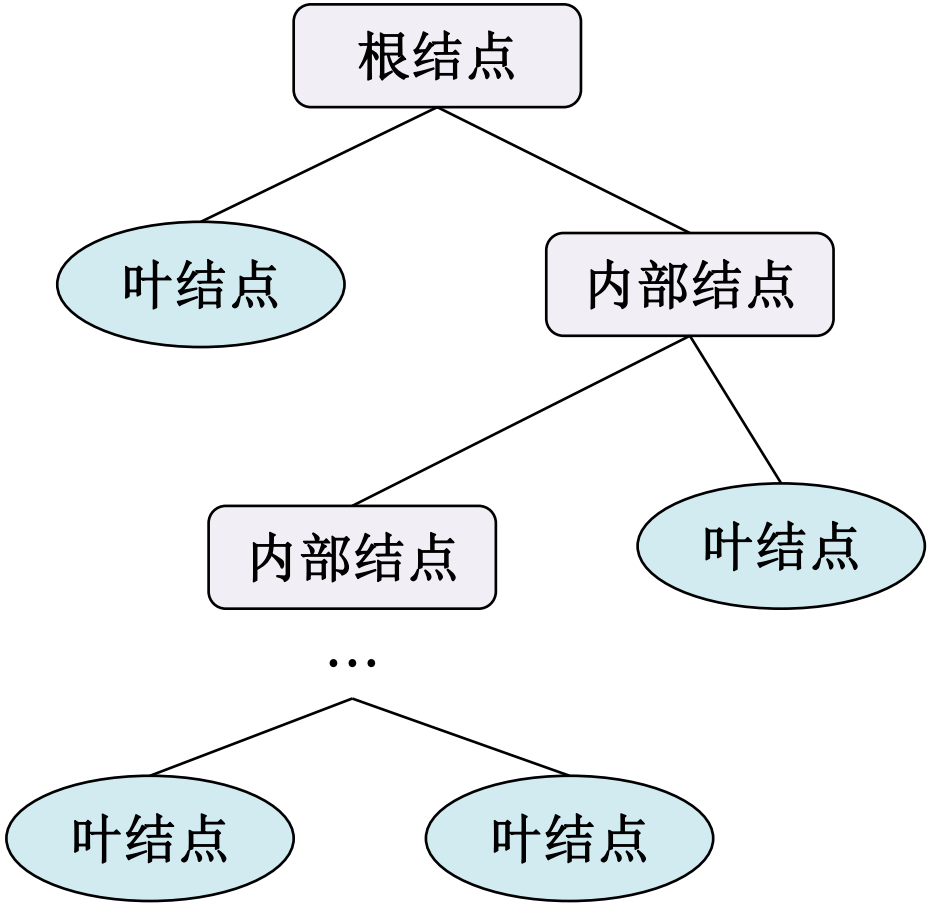
决策树的构建

- **算法基本流程**
  - **步骤1：训练，从数据中获取知识进行学习**
    - **利用训练集建立 (并精化)一棵决策树，构建决策树模型**
  - **步骤2：测试，利用构建的模型对输入数据进行分类**
    - **对测试集样本，从根结点依次测试记录的属性值，直至到达某个叶结点，找到该样本所在的类别**



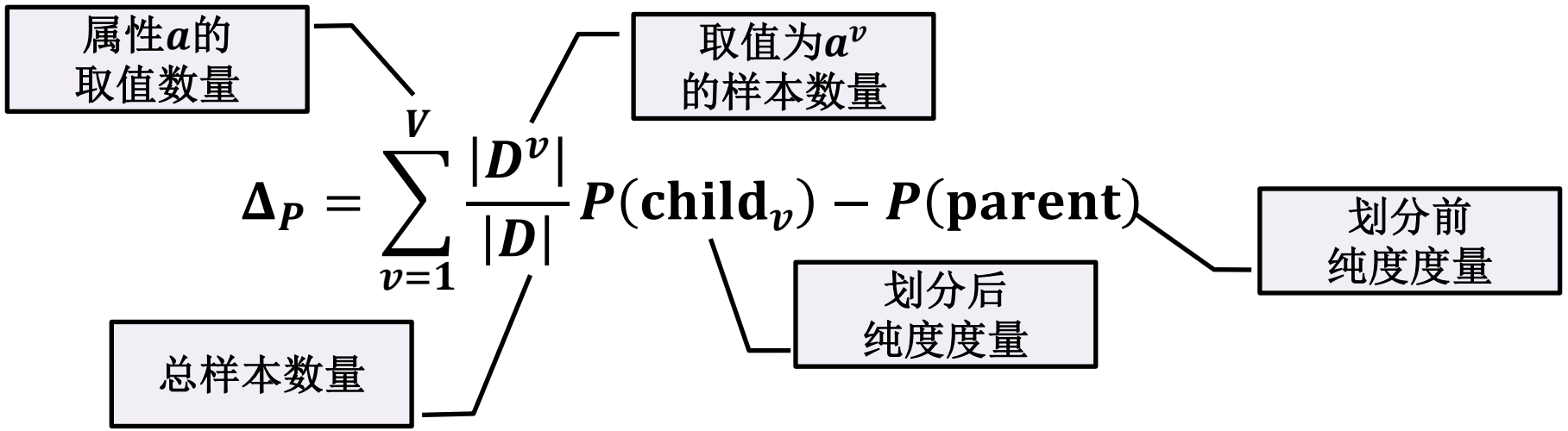
决策树的构建

- **决策树构建基本流程**
  - 步骤1：选取一个最佳划分属性作为决策树的根结点，并就该属性所有的取值创建树的分支
  - 步骤2：使用决策树对训练数据集进行分类
    - 如果一个结点的所有实例都属于同一类，则以该类为标记标识此叶结点
    - 如果所有的叶结点都有类标记，则算法终止
  - 步骤3：否则，选取一个从该结点到根结点路径中未出现过的最佳属性作为标记标识该结点
  - 步骤4：就该属性所有的取值继续创建树的分支，重复步骤2~4



决策树的构建

- 构建决策树的关键：选择当前状态下的**最佳划分属性**，作为分类依据
- 决策树学习目标：每个分支结点的样本尽可能属于同一类别，即结点的“纯度”(Purity)越来越高
  - 比较划分前和划分后纯度的上升程度，上升的越多，划分的效果越好
  - 纯度的上升程度记为Δ_P，则用于确定划分效果的度量标准表示为

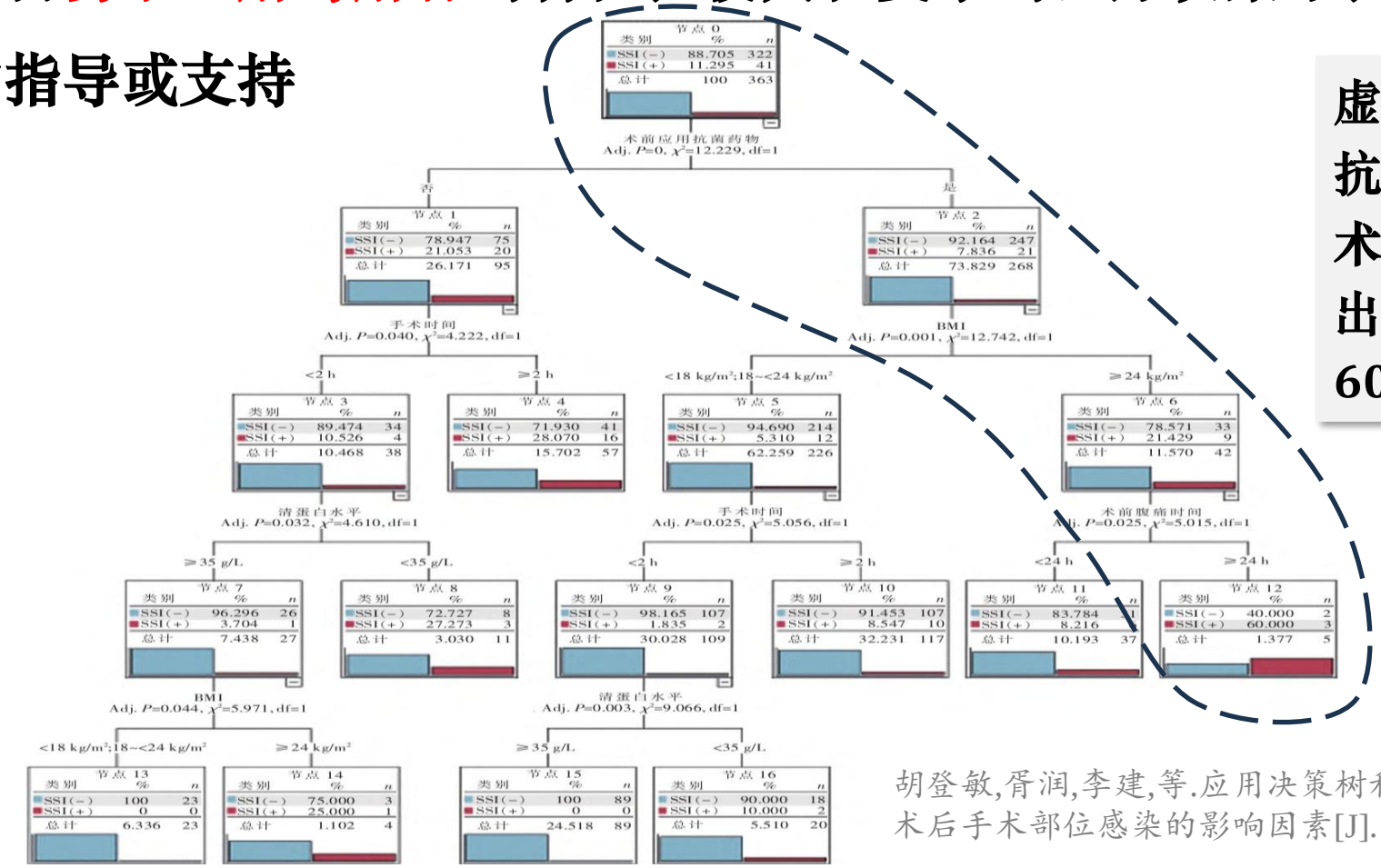


决策树的构建

- 决策树结点纯度度量方式包含：
  - 信息熵 (Information Entropy)
  - 基尼指数 (Gini Index)
- 根据不同的纯度度量方式，决策树学习主要包含以下三种算法：
  - ID3算法【1979年J. R. Quinlan提出】：信息增益 (Information Gain)
  - C4.5算法【1993年J. R. Quinlan提出】：信息增益率 (Information Gain Ratio)
  - CART (Classification And Regression Tree)算法【1984年L. Breiman提出】：基尼指数 (Gini Index)

决策树的应用

- **决策树的应用——医疗行业**
  - ➢ 决策树**易于理解与解释**的特性，使其在复杂的医疗决策中，能够为医患提供有力的指导或支持



虚线框分支表示：术前应用抗菌药物、 BMI ≥ 24kg/m² 、术前腹痛时间≥ 24h的患者，出现手术部位感染的概率为60%

胡登敏,胥润,李建,等.应用决策树和logistic回归模型分析胃肠道穿孔修补术后手术部位感染的影响因素[J].中国感染控制杂志,2024,23(07):826-832.

12.2 ID3算法

- 信息增益
- ID3算法

此节课程参考：周志华.(2016).机器学习.北京: 清华大学出版社.

ID3算法

- **ID3 (Iterative Dichotomiser 3)迭代二分器算法**

【1979年J. R. Quinlan提出】

  - 是一种最经典的决策树学习算法
  - 基本思想：以信息熵为结点纯度度量，每次优先选取信息增益最大的属性，即能使熵值最小的属性，构建一棵熵值下降最快的决策树，到叶结点的熵值为0，此时对应样本集中的所有样本属于同一类别

信息增益计算

- **信息熵 (Information Entropy)**
  - 信息熵表示随机变量不确定性的大小，是度量样本集合纯度最常用的一种指标。

    信息熵越大，随机变量的不确定性越大，样本集合的纯度越低
  - **离散随机变量的信息熵**：令一个取有限个值的离散随机变量X的概率分布为

    $P(X = x_i) = p_i$，则随机变量X的信息熵定义为

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

    信息按二进制位编码，因此以2为底
  - **连续随机变量的信息熵**：若X为连续随机变量，则概率分布替换为概率密度函数，且求和操作替换为积分操作即可

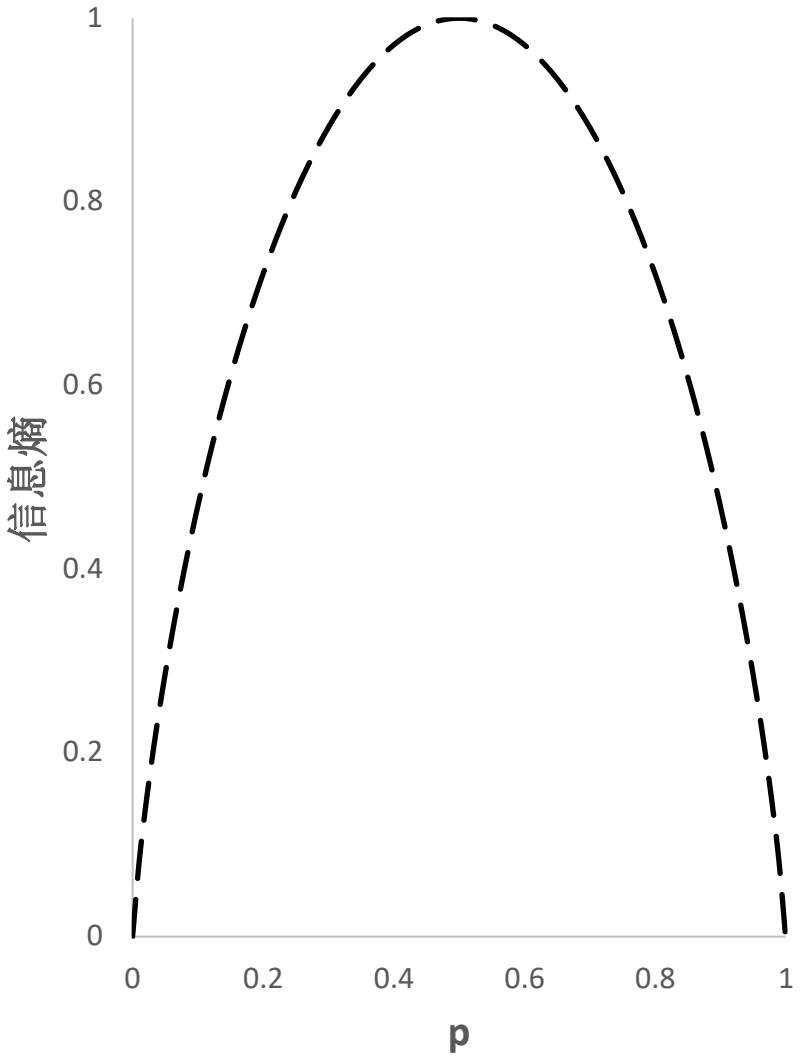
信息增益计算

- **信息熵 (Information Entropy)**
  - 信息熵定义了**概率密度函数**到**信息熵值**的映射关系，即$P(X = x_i) \rightarrow H(X)$
  - 示例：当随机变量X仅有两个取值，如0和1时，则其概率分布为$P(X = 0) = p$，$P(X = 1) = 1 - p$，则随机变量X的熵为

$$H(X) = -p\log_2 p - (1-p)\log_2(1-p)$$

当X退化为定值，即$p = 0$或1时，熵为0；

当X为均匀分布，即$p = 0.5$时，熵为1；



信息增益计算

- **经验熵 (Empirical Entropy)**
  - 经验熵表示样本集合的纯度的高低，经验熵越小，样本集合的纯度越高；假设当前样本集合D中第k类样本所占比例为p_k，则D的经验熵定义为

$$H(D) = -\sum_{k=1}^{K} p_k \log_2 p_k = -\sum_{k=1}^{K} \frac{D_k}{D} \log_2 \frac{D_k}{D}$$

- **条件熵 (Conditional Entropy)**
  - 条件熵表示在已知随机变量X的条件下，随机变量Y的不确定性；对于随机变量(X, Y)，联合概率分布为$P(X = x_i, Y = y_i) = p_{ij}$，则条件熵定义为

$$H(Y|X) = -\sum_{i=1}^{n} p_i H(Y|X = x_i) = H(X, Y) - H(X)$$

信息增益计算

- **经验条件熵 (Empirical Conditional Entropy)**
  - 经验条件熵表示属性a的信息对样本集合D的信息的不确定性减少的程度；假设当前样本集合D中共有K类，每一类有D_k个样本，属性a有V个可能的取值$\{a^1, a^2, \cdots, a^v\}$，属性为$a^v$的样本数为$D^v$，且每一类中包含$D_k^v$个样本，则$D$的经验条件熵定义为

$$H(D|a) = -\sum_{v,k} p(D_k, a^v) \log_2 (D_k|a^v)$$

$$= -\sum_{v=1}^{V} \frac{|D^v|}{|D|} \sum_{k=1}^{K} \frac{|D_k^v|}{|D^v|} \log_2 \frac{|D_k^v|}{|D^v|} = \sum_{v=1}^{V} \frac{|D^v|}{|D|} H(D^v)$$

信息增益计算

- **信息增益 (Information Gain)**
  - 信息增益表示使用属性a进行划分所获得的“纯度”上升程度，信息增益越大，则代表使用属性a进行划分所获得的“纯度”上升越快；属性a对训练数据集D的信息增益记为$G(D,a)$，定义为集合D的经验熵$H(D)$与在给定属性a的条件下D的经验条件熵$H(D|a)$之差，即

$$G(D,a) = H(D) - H(D|a)$$

$$= H(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} H(D^v)$$

  - ID3算法即是以此信息增益为准则，对每次递归的结点属性进行划分的

ID3算法流程

决策树构建算法

输入：训练数据集D，属性集A，信息增益阈值ϵ

过程：

步骤1：若D中所有样本属于同一类k，则T为单结点树，并将类k作为该结点类标记，返回T；

步骤2：若$A=\emptyset$，则T为单结点树，并将D中样本数最大类k作为该结点类标记，返回T；

步骤3：否则，计算A中各属性对D的信息增益，选择信息增益最大的属性a_*；

步骤4：如果a_*的信息增益小于阈值ϵ，则置T为单结点树，并将D中样本数最大类k作为该结点类标记，返回T；

步骤5：否则，对a_*的每一个可能值a_*^v，分割D为若干非空子集D^v，将D^v中样本数最大的类作为类标记，构建子结点，由结点及其子结点构成树T，返回T；

步骤6：对第v个子结点，以D^v为训练集，$A-\{a_*\}$为属性集，递归的调用步骤1~5，得到子树T_i，返回T_i。

输出：决策树T

ID3算法-示例

- 决策树构建算法——以判断西瓜好坏为例

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

ID3算法-示例

- **计算信息熵—以属性“色泽”为例**
  - **计算根结点的信息熵**

$$H(D) = -\left(\frac{8}{17}\log_2\frac{8}{17} + \frac{9}{17}\log_2\frac{9}{17}\right) = 0.998$$

  - **计算分支结点的信息熵**

$$H\left(D^{\text{青绿}}\right) = -\left(\frac{3}{6}\log_2\frac{3}{6} + \frac{3}{6}\log_2\frac{3}{6}\right) = 1.000$$

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

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

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

ID3算法-示例

- 计算信息增益—以属性“色泽”为例
  - 计算属性“色泽”的信息增益

$$G(D,\text{色泽}) = H(D) - \sum_{v=1}^{3} \frac{|D^v|}{|D|} H(D^v)$$

$$= 0.998 - \left(\frac{6}{17} \times 1.000 + \frac{6}{17} \times 0.918 + \frac{5}{17} \times 0.722 \right)$$

$$= 0.109$$

  - 计算其他属性的信息增益

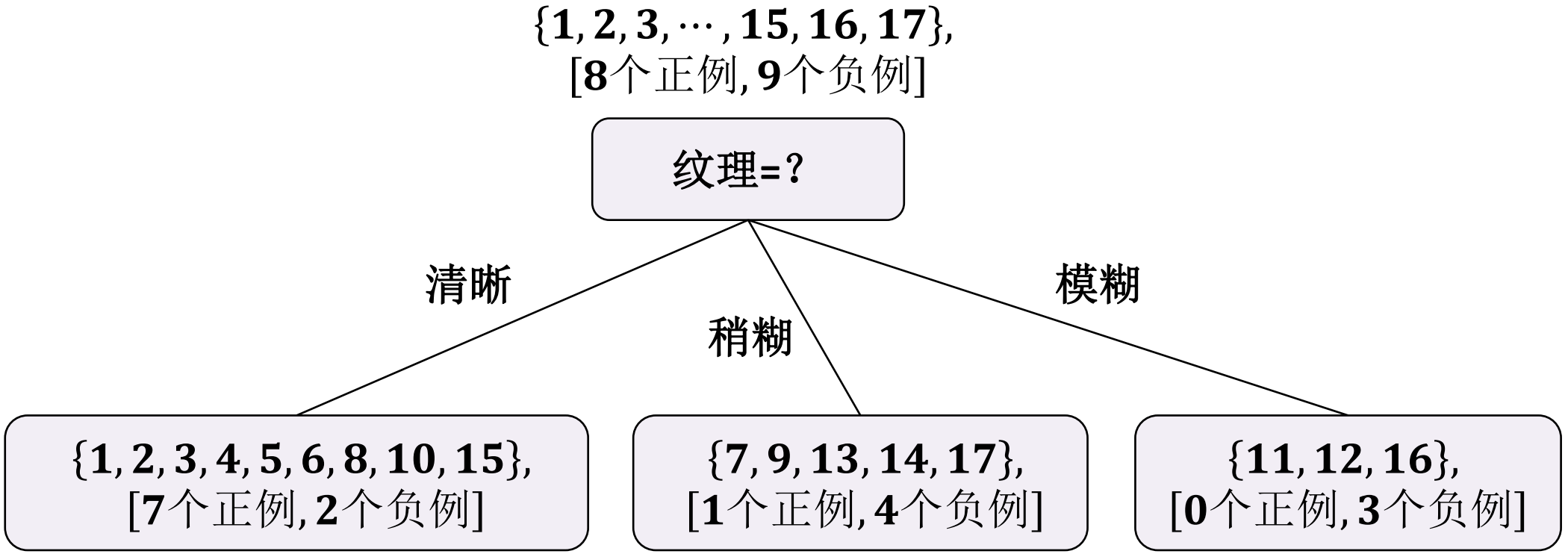
$G(D,\text{根蒂}) = 0.143$　　$G(D,\text{敲声}) = 0.141$

最佳划分属性　$G(D,\text{纹理}) = 0.381$　　$G(D,\text{脐部}) = 0.289$

$G(D,\text{触感}) = 0.006$

ID3算法-示例

- 基于属性“纹理”对根结点进行划分



ID3算法-示例

- 继续进行内部结点划分—以属性“纹理”的分支为例
  - 对于“纹理=清晰”分支，可用属性集合为{色泽,根蒂,敲声,脐部,触感}，计算各属性的信息增益：

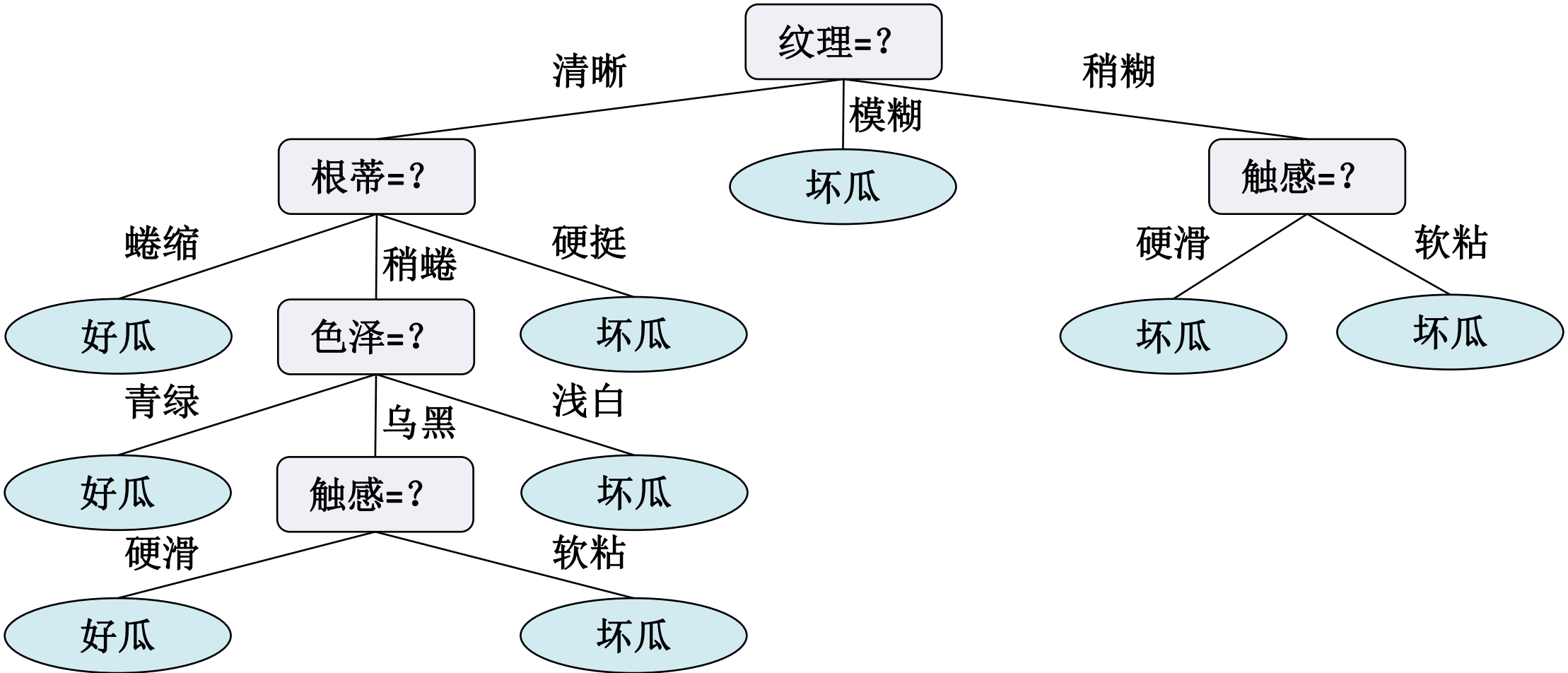
$$G\left(D^{\text{清晰}},\text{色泽}\right)=0.043 \quad G\left(D^{\text{清晰}},\text{根蒂}\right)=0.458 \quad G\left(D^{\text{清晰}},\text{敲声}\right)=0.331$$

$$G\left(D^{\text{清晰}},\text{脐部}\right)=0.458 \quad G\left(D^{\text{清晰}},\text{触感}\right)=0.458$$

  - 对于“纹理=稍糊”分支，同样计算各属性的信息增益
  - 对于“纹理=模糊”分支，包含的样本集合中有编号为{11,12,16}的3个样本，且属于同一类，因此直接将该结点归为叶结点

ID3算法-示例

- 对所有内部结点重复上述步骤，直至决策树中所有叶结点均有类标记，决策树停止生长，得到最终的决策树



ID3算法

- 算法优点
  - 能够从一类无序、无规则概念中推理出分类规则
  - 能够将决策树中到达每个叶结点的路径转换为if-then形式的分类规则，比较符合人类的理解方式
- 算法局限性
  - 信息增益偏好取值多的属性 (极限趋近于均匀分布)
  - 会受噪声或小样本影响，易出现过拟合问题
  - 无法处理连续值的属性
  - 无法处理属性值不完整的训练数据
  - 无法处理不同代价的属性

12.3 C4.5与CART算法

- C4.5算法
- CART算法

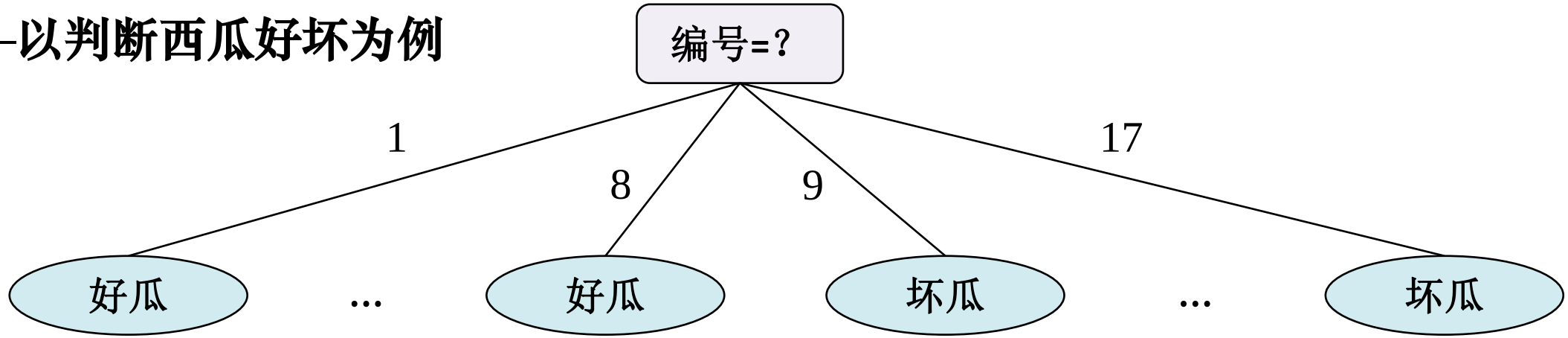
此节课程参考：周志华.(2016).机器学习.北京: 清华大学出版社.

ID3算法局限性

- **ID3算法局限性 (1)**

$$G(D,a) = H(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} H(D^v)$$

  - 信息增益准则对可取值数目V较多的属性有所偏好，**极限趋近于均匀分布**
  - 取值更多的属性容易使得数据的纯度更高，其信息增益更大。决策树会首先挑选该属性作为树的顶/结点；结果训练出来的形状是一棵庞大且深度很浅的树——以判断西瓜好坏为例



C4.5算法

- **C4.5算法** 【1993年J. R. Quinlan提出】
  - 基本思想：采用**信息增益率** (Information Gain Ratio)替代ID3算法中的信息增益，即以信息熵为结点纯度度量，每次优先选取信息增益率最大的属性

$$G_{ratio}(D,a)=\frac{G(D,a)}{\mathrm{IV}(a)}，其中，\mathrm{IV}(a)=-\sum_{v=1}^{V}\frac{|D^v|}{|D|}\log_2\frac{|D^v|}{|D|}$$

  - $\mathrm{IV}(a)$称为属性a的固有值 (Intrinsic Value)，属性a的可能取值V越大，则通常固有值也越大。因此，采用信息增益率，可缓解信息增益对取值较多属性的偏好

C4.5算法

决策树构建算法

输入：训练数据集D，属性集A，阈值ϵ

过程：

步骤1：若D中所有样本属于同一类k，则T为单结点树，并将类k作为该结点类标记，返回T；

步骤2：若$A=\emptyset$，则T为单结点树，并将D中样本数最大类k作为该结点类标记，返回T；

步骤3：否则，计算A中各属性对D的信息增益率，选择信息增益率最大的属性a_*；

步骤4：如果a_*的信息增益率小于阈值ϵ，则置T为单结点树，并将D中样本数最大类k作为该结点类标记，返回T；

步骤5：否则，对a_*的每一个可能值a_*^v，分割D为若干非空子集D^v，将D^v中样本数最大的类作为类标记，构建子结点，由结点及其子结点构成树T，返回T；

步骤6：对第v个子结点，以D^v为训练集，$A-\{a_*\}$为属性集，递归的调用步骤1~5，得到子树T_i，返回T_i。

输出：决策树T

CART (Classification And Regression Tree)算法

- **CART分类与回归树算法** 【1984年L. Breiman提出】
  - **基本思想**：CART算法是一种采用**基尼指数**选择划分属性的**二叉决策树，**相较于ID3与C4.5算法，CART更加高效灵活，可解释性更强
  - **基尼指数 (Gini Index)**直观反映了从数据集中随机抽取两个样本，其类别不一致的概率；基尼指数越小，数据集的纯度越高

$$Gini(D) = \sum_{k=1}^{K}\sum_{k'\neq k} p_k p_{k'} = 1 - \sum_{k=1}^{K} {p_k}^2 = 1 - \sum_{k=1}^{K}\left(\frac{|D_k|}{|D|}\right)^2$$

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

对于CART算法，属性a特征值a_v的基尼指数：$Gini(D,a_v) = \frac{|D_l|}{|D|}Gini(D_l) + \frac{|D_r|}{|D|}Gini(D_r)$

最优属性特征选择：$a_*^v = \arg\min_{a\in A} Gini(D,a_v)$

D_l, D_r是以a_v为分割点将D分割成的两部分

CART算法

决策树构建算法

输入：训练数据集D，属性集A

过程：

步骤1：若D中所有样本属于同一类k，则T为单结点树，并将类k作为该结点类标记，返回T；

步骤2：若$A=\emptyset$，则T为单结点树，并将D中样本数最大类k作为该结点类标记，返回T；

步骤3：否则，计算A中各属性的特征值对D的Gini系数，选择使得Gini系数最小的属性a_*及其对应的特征值a_*^v，分别记作最优属性和最优分割点

步骤4：以最优属性a_*结点类标记，按照最优分割点a_*^v把数据集D分为两部分D_l, D_r，构建二叉树T，返回T

步骤6：分别以D_l, D_r为训练集，递归的调用步骤1~5，得到子树T_i，返回T_i。

输出：决策树T

12.4 剪枝算法

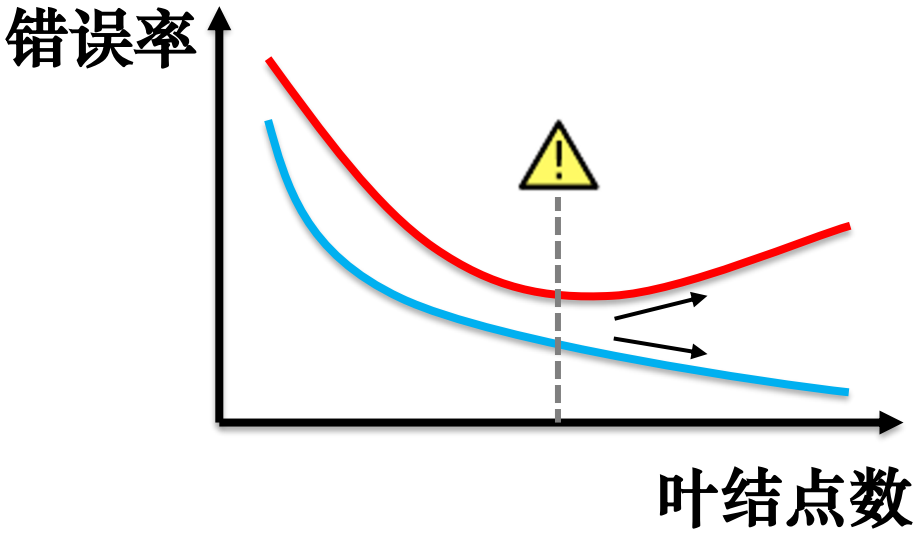
- 过拟合问题
- 预剪枝算法
- 后剪枝算法

此节课程参考：周志华.(2016).机器学习.北京: 清华大学出版社.

ID3算法局限性

- ID3算法局限性 (2)
  - 决策树对训练数据有很好的分类能力，但对未知的测试数据未必有好的分类能力，泛化性能弱，即可能发生过拟合现象

- 过拟合问题可能的原因
  - 训练数据有噪声，决策树同时拟合了数据和噪音，影响分类效果
  - 叶结点数量太多，每个结点的样本太少，易出现耦合的规律性，导致一些与真实数据分布无关的属性恰巧被正确分类



剪枝算法

- 针对过拟合问题
  - 剪枝是主要手段。剪枝的目的是通过剪去部分叶结点，提高决策树的泛化性能，即决策树在测试数据上的分类准确率
- 剪枝的基本算法
  - 预剪枝算法（Pre-pruning）：在决策树构建过程中，对每个结点在划分前进行估计，若划分不能带来决策树泛化性能提升，则停止划分并将该结点设为叶结点
  - 后剪枝算法（Post-pruning）：在决策树建立后，自底向上对非叶结点进行考察，若将该结点对应子树替换为叶结点能带来泛化性能提升，则将该子树替换为叶结点

预剪枝算法-示例

- 给定西瓜的不同属性特征，判断西瓜是否为好瓜

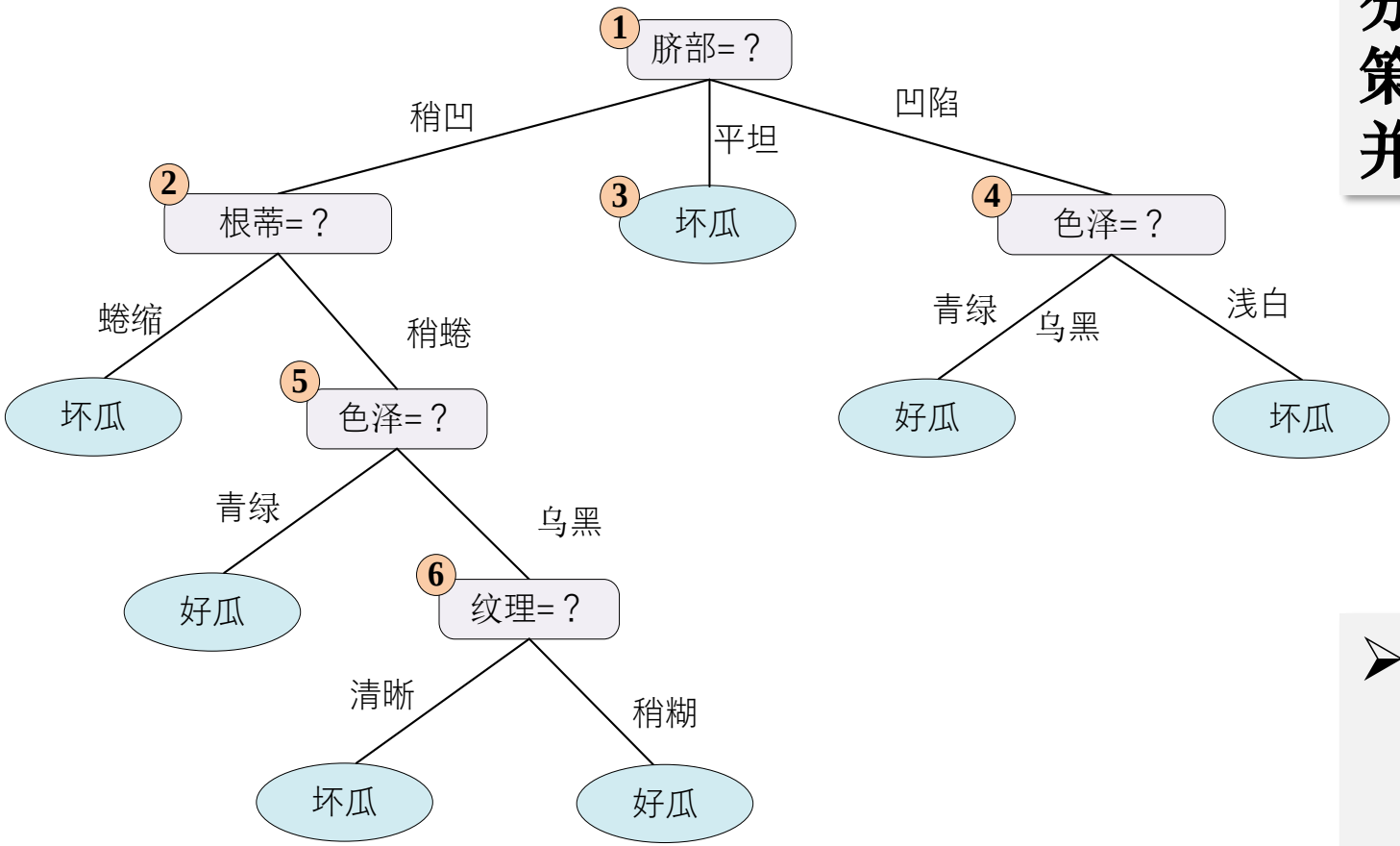
	编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
训练样本	1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
	2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
	3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
	6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
	7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
	10	青绿	硬挺	清脆	清晰	平坦	软粘	否
	14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
	15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
	16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
	17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否
测试样本	4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
	5	浅白	蜷缩	浊响	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

用于**构建**决策树

用于**评估**决策树的**泛化性能**

预剪枝算法-示例

- **ID3算法构建的未剪枝决策树**



➢ **预剪枝**

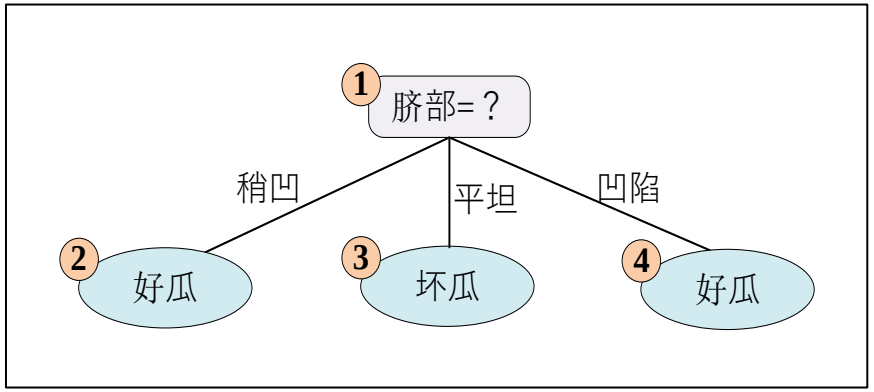
决策树构建过程中，对各结点在划分前进行估计，若划分不能带来决策树泛化性能提升，则停止划分，并将该结点设为叶结点

➢ **原始决策树泛化性能**

{4, 11, 12}被正确划分，

准确率： 3/7= 42.9%

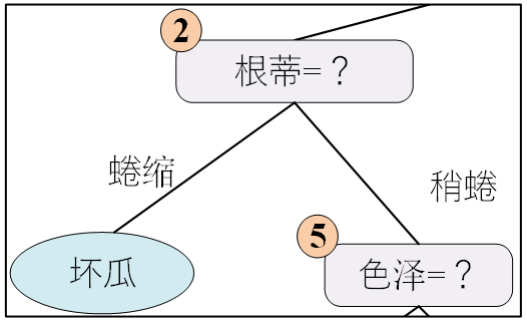
预剪枝算法-示例



- **第一步：评估结点1**
  - **属性选择：**基于信息增益准则，选择属性“脐部”
  - **不划分**
    - 标记为训练样例数最多的类别，如“好瓜”
    - 泛化性能：{4, 5, 8}被正确分类，准确率 **3/7= 42.9%**
  - **划分**
    - 结点2：稍凹{6, 7, 15, 17}　　“好瓜”
    - 结点3：平坦{10, 16}　　“坏瓜”
    - 结点4：凹陷{1, 2, 3, 14}　　“好瓜”
    - 泛化性能：{4, 5, 8, 11, 12}被正确分类，准确率 **5/7= 71.4%**

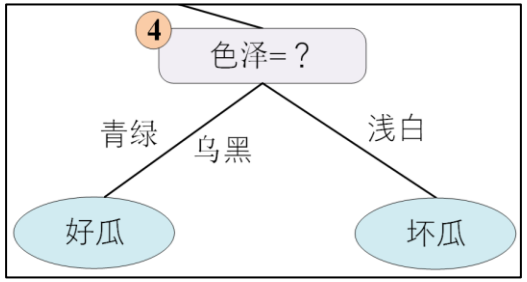
评估结果/预剪枝决策：划分

预剪枝算法-示例



- **第二步：评估结点2：训练样本{6, 7, 15, 17}**
  - **属性选择：**基于信息增益准则，选择属性“根蒂”
    - **不划分：**{4, 5, 8, 11, 12}被正确分类分类，准确率 **5/7= 71.4%**
    - **划分：**{4, 5, 8, 11, 12}被正确分类分类，准确率 **5/7= 71.4%**

评估结果/预剪枝决策： 不划分



- **第三步：评估结点4：训练样本{1, 2, 3, 14}**
  - **属性选择：**基于信息增益准则，选择属性“色泽”
    - **不划分：**{4, 5, 8, 11, 12}被正确分类分类，准确率 **5/7= 71.4%**
    - **划分：**{4, 8, 11, 12}被正确分类分类，准确率 **4/7= 57.1%**

评估结果/预剪枝决策： 不划分

预剪枝算法-示例

- 预剪枝流程总结

结点1：“脐部=？”

划分前：测试集精度 42.9%

划分后：测试集精度 71.4%

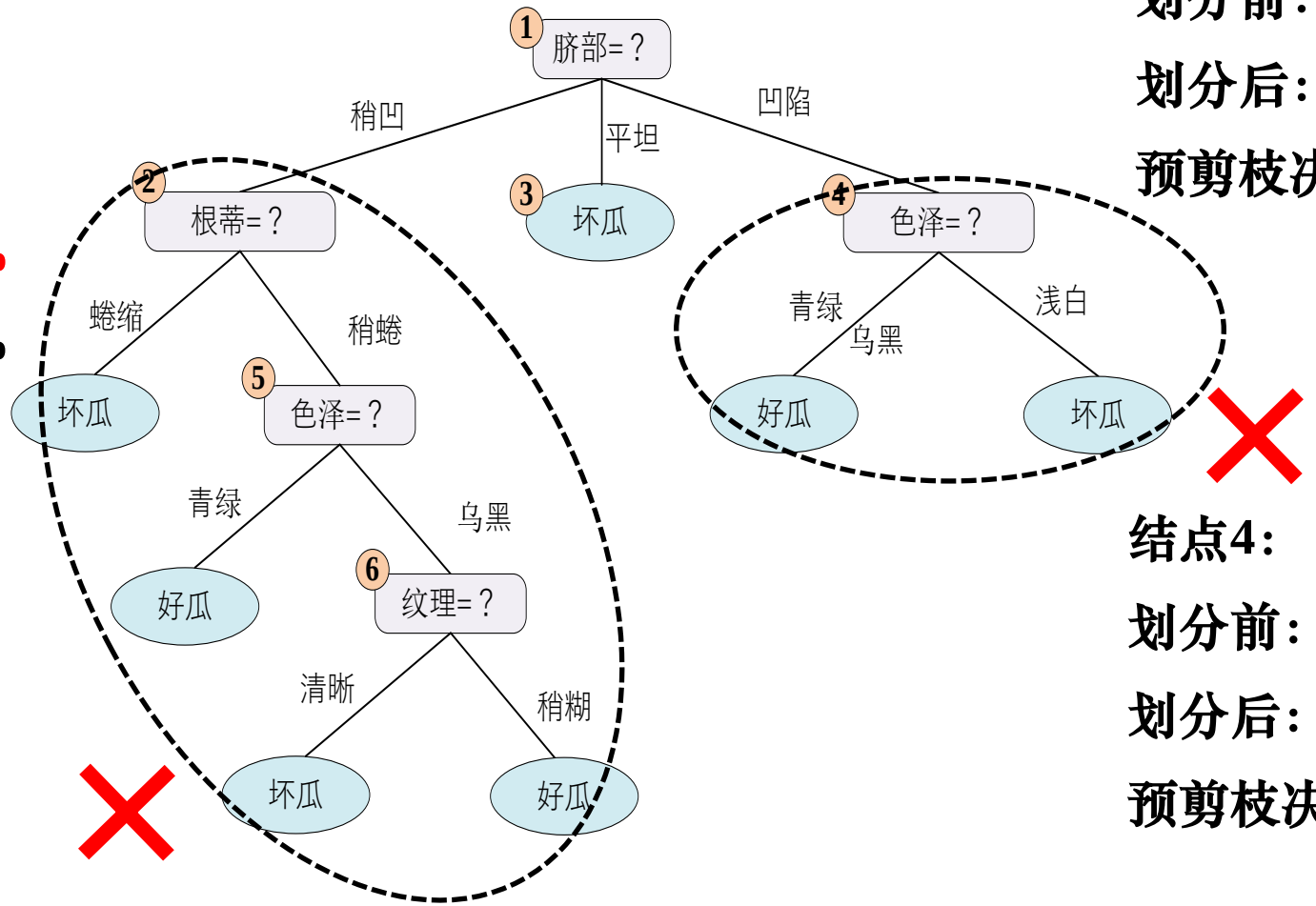
预剪枝决策：划分

结点2：“根蒂=？”

划分前：测试集精度 71.4%

划分后：测试集精度 71.4%

预剪枝决策：不划分



结点4：“色泽=？”

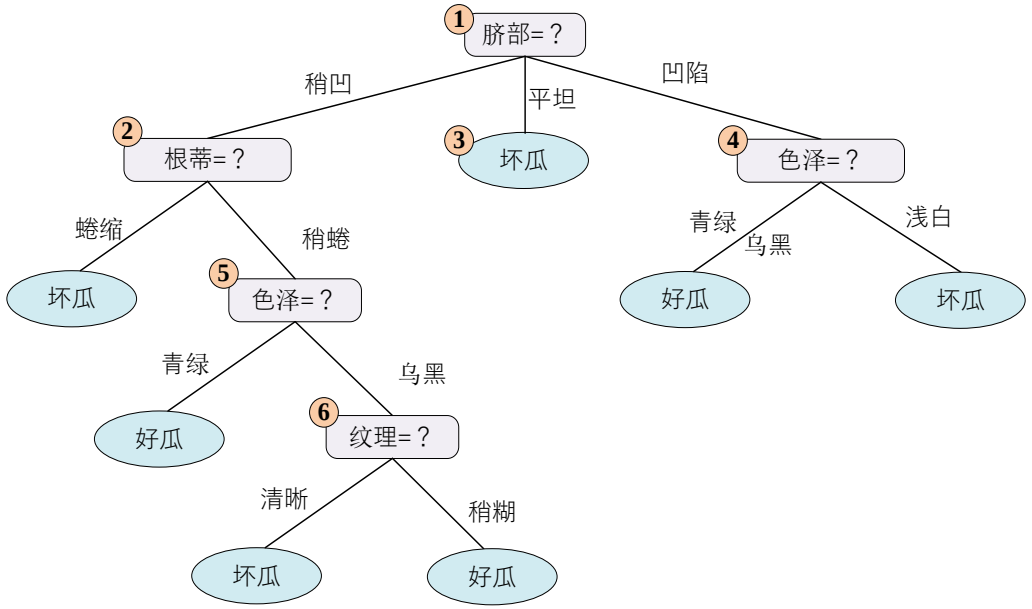
划分前：测试集精度 71.4%

划分后：测试集精度 57.1%

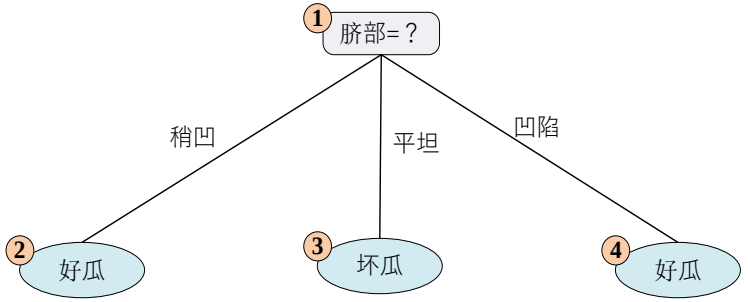
预剪枝决策：不划分

预剪枝算法-示例

- **ID3算法构建的未剪枝决策树**



- **预剪枝后的决策树**



➢ **原始决策树泛化性能**

{4, 11, 12}被正确划分，

准确率： 3/7= **42.9%**

➢ **预剪枝决策树泛化性能**

{4, 5, 8, 11, 12}被正确划分，

准确率： 5/7= **71.4%**

预剪枝算法

- **预剪枝算法特点**
  - 优势：“剪掉”很多没必要展开的分支，降低了过拟合风险，并且显著减少了决策树的训练时间开销和测试时间开销
  - 劣势：有些分支的当前划分有可能不能提高甚至降低泛化性能，但后续划分有可能提高泛化性能；预剪枝禁止这些后续分支的展开，可能会导致欠拟合

后剪枝算法-示例

- 给定西瓜的不同属性特征，判断西瓜是否为好瓜

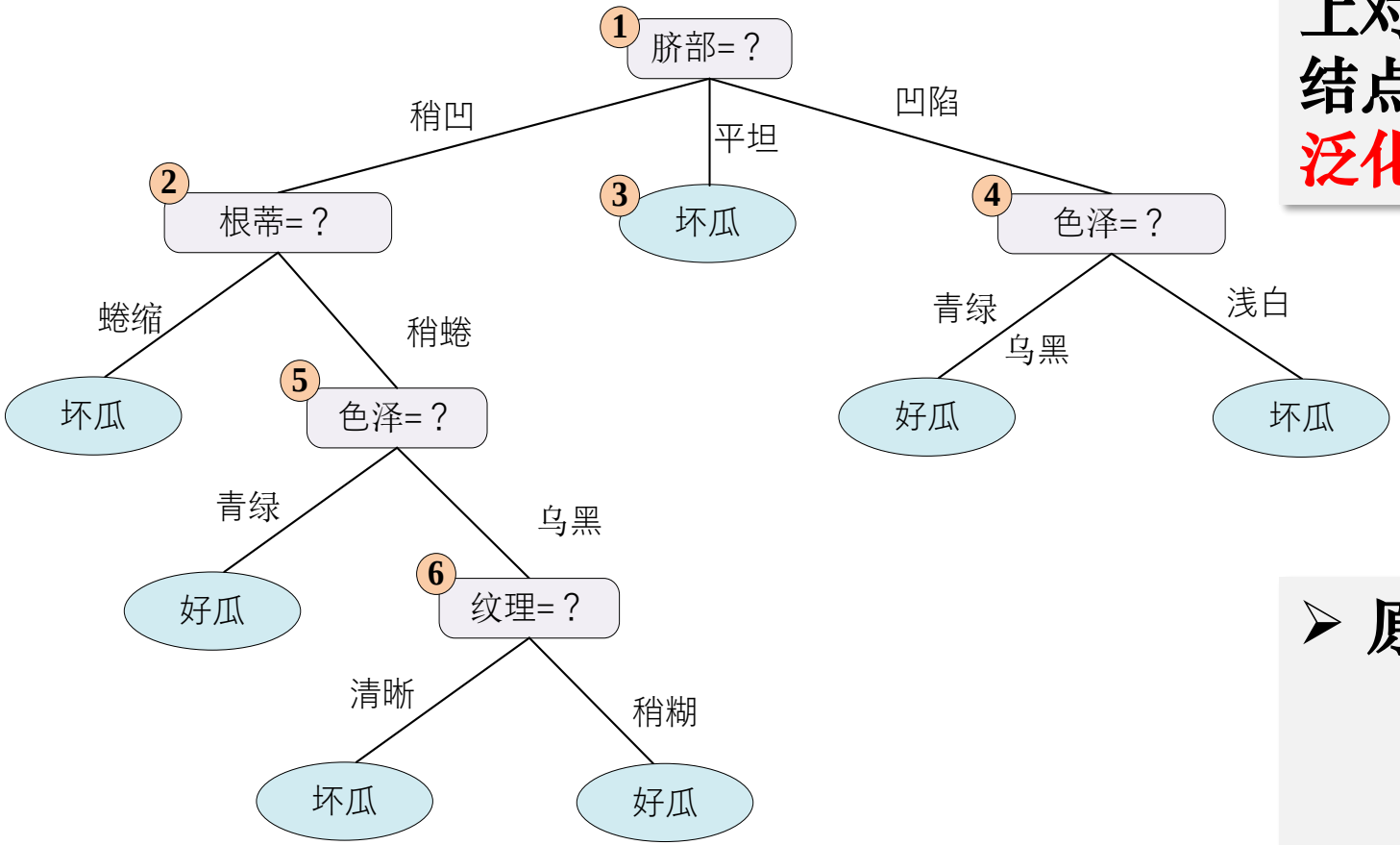
	编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
训练样本	1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
	2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
	3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
	6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
	7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
	10	青绿	硬挺	清脆	清晰	平坦	软粘	否
	14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
	15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
	16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
	17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否
测试样本	4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
	5	浅白	蜷缩	浊响	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

用于**构建**决策树

用于**评估**决策树的**泛化性能**

后剪枝算法-示例

- **ID3算法构建的未剪枝决策树**



➢ **后剪枝**

先利用训练集构建决策树，自底向上对非叶结点进行考察，若将该叶结点对应子树替换为叶结点能带来泛化性能提升，则进行替换.

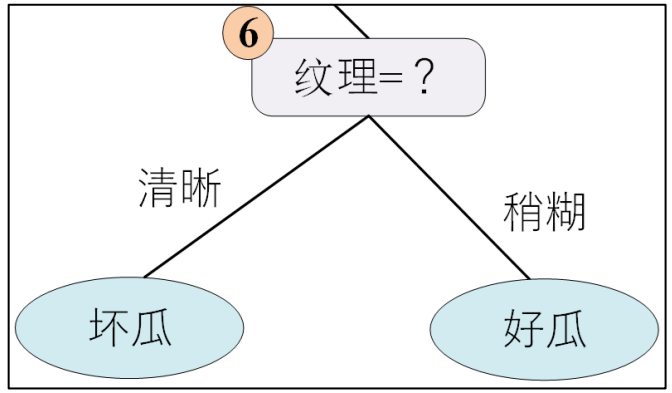
➢ **原始决策树泛化性能**

{4, 11, 12}被正确划分

准确率：3/7= 42.9%

后剪枝算法-示例

- **第一步：评估结点6**

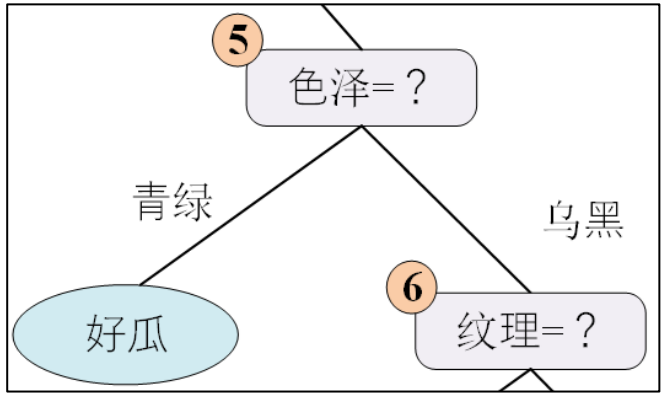


  - **剪枝前**
    - 属性为“纹理”；样本为{7, 15}
    - 泛化性能：{4, 11, 12}被正确分类 **3/7= 42.9%**
  - **剪枝后**
    - 把结点6替换为叶结点，“好瓜”（也可替换为“坏瓜”）
    - 泛化性能：{4, 8, 11, 12}被正确分类 **4/7= 57.1%**

评估结果/预剪枝决策： 剪枝

后剪枝算法-示例

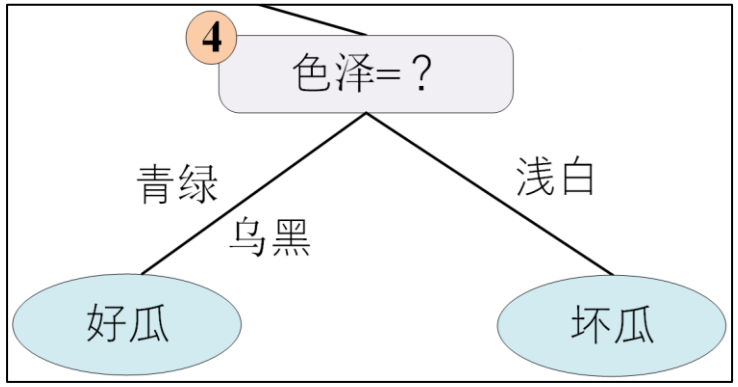
- **第二步：评估结点5**
  - **剪枝前**
    - 属性为“色泽”；样本为{6, 7, 15}
    - 泛化性能：{4, 8, 11, 12}被正确分类 **4/7 = 57.1%**
  - **剪枝后**
    - 把结点5替换为叶结点，“好瓜”
    - 泛化性能：{4, 8, 11, 12}被正确分类 **4/7= 57.1%**



评估结果/预剪枝决策：　不剪枝

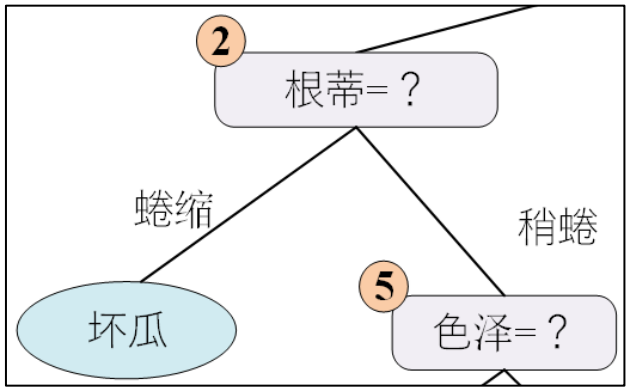
后剪枝算法-示例

- **第三步：评估结点4**
  - **剪枝前**
    - 属性为“色泽”，样本{1, 2, 3, 14}
    - 泛化性能：{4, 8, 11, 12}被正确分类 **4/7 = 57.1%**
  - **剪枝后**
    - 把结点4替换为叶结点，“好瓜”
    - 泛化性能：{4, 5, 8, 11, 12}被正确分类 **5/7= 71.4%**



评估结果/预剪枝决策： 剪枝

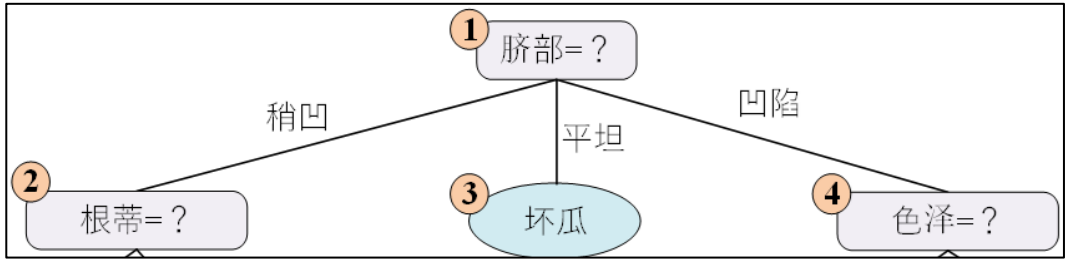
后剪枝算法-示例



- **第四步：评估结点2**
  - **剪枝前**
    - 属性为“根蒂”，样本{6, 7, 15, 17}
    - 泛化性能：{4, 5, 8, 11, 12}被正确分类 **5/7= 71.4%**
  - **剪枝后**
    - 把结点2替换为叶结点，“好瓜”（也可替换为“坏瓜”）
    - 泛化性能：{4, 5, 8, 11, 12}被正确分类 **5/7= 71.4%**

评估结果/预剪枝决策：　不剪枝

后剪枝算法-示例

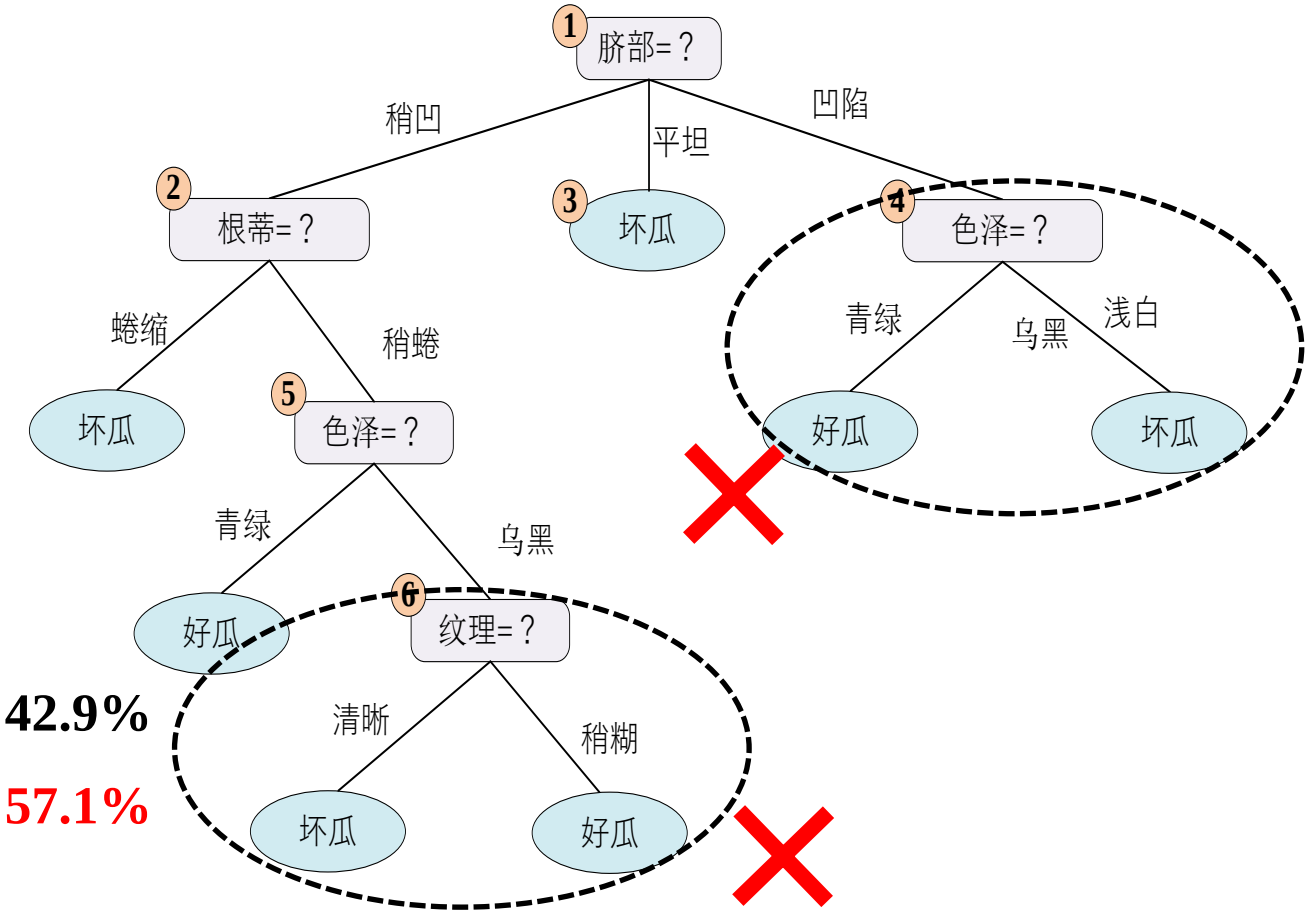


- **第五步：评估结点1**
  - **剪枝前**
    - 属性为“脐部”
    - 泛化性能：{4, 5, 8, 11, 12}被正确分类 **5/7= 71.4%**
  - **剪枝后**
    - 把结点1替换为叶结点，标记为训练样例数最多的类别，如“好瓜”
    - 泛化性能：{4, 5, 8}被正确分类，准确率 **3/7= 42.9%**

评估结果/预剪枝决策： 不剪枝

后剪枝算法-示例

- 后剪枝流程总结



结点4：“色泽=？”

剪枝前：测试集精度 57.1%

划分后：测试集精度 71.4%

后剪枝决策：剪枝

结点6：“纹理=？”

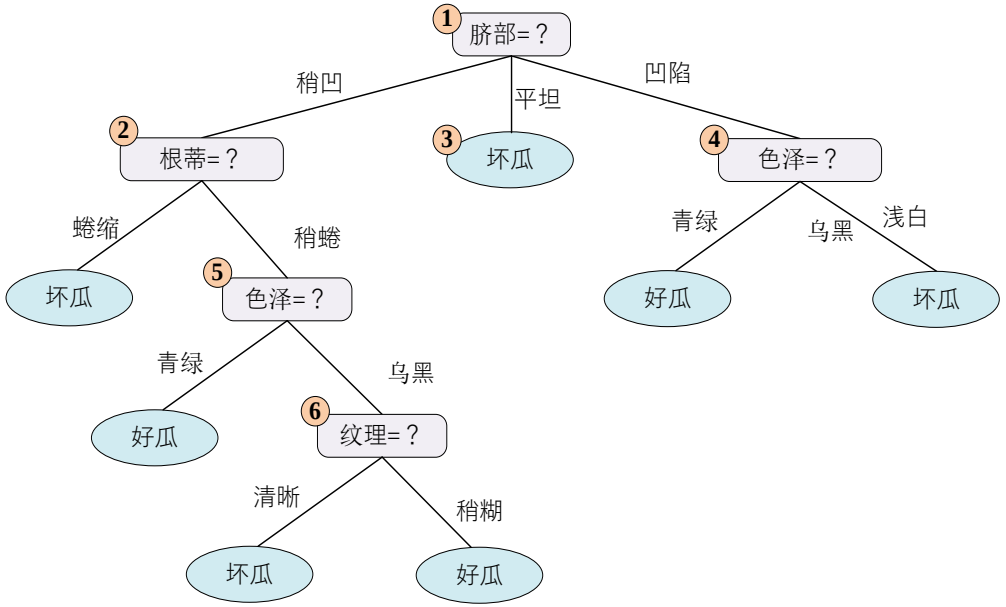
剪枝前：测试集精度 42.9%

划分后：测试集精度 57.1%

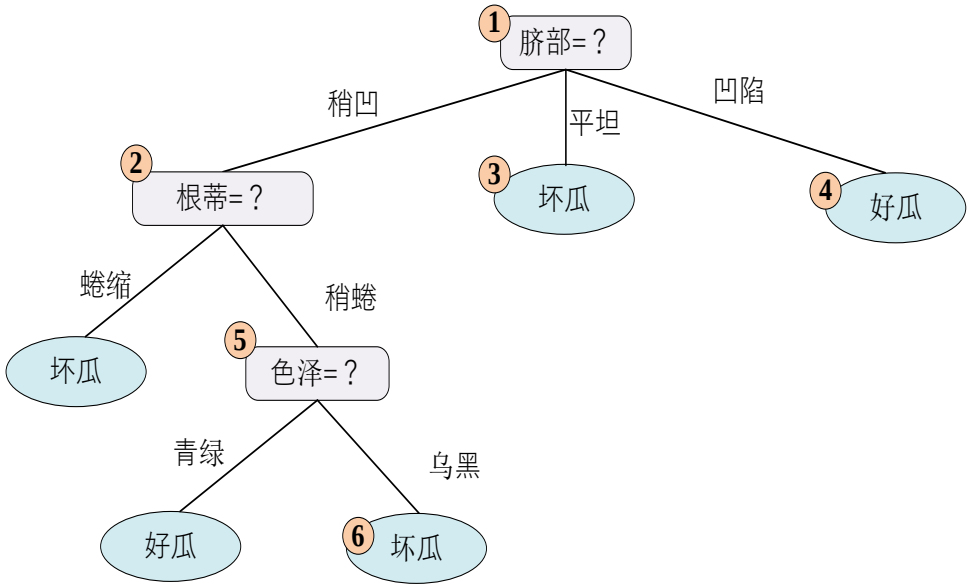
后剪枝决策：剪枝

后剪枝算法-示例

- ID3算法构建的未剪枝决策树



- 后剪枝后的决策树



- 原始决策树泛化性能

{4, 11, 12}被正确划分

准确率：3/7= 42.9%

- 后剪枝决策树泛化性能

{4, 5, 8, 11, 12}被正确划分

准确率：5/7= 71.4%

后剪枝算法

- **后剪枝算法特点**
  - 优势：测试了所有分支，比预剪枝决策树保留了更多分支，降低了欠拟合的风险，泛化性能一般优于预剪枝决策树
  - 劣势：后剪枝过程在完全构建决策树后再进行，且要自底向上对所有非叶结点逐一评估；因此，决策树的训练时间开销要高于未剪枝决策树和预剪枝决策树

12.5 对于不同类型属性的处理

- 连续值处理
- 缺失值处理
- 不同代价属性处理

此节课程参考：周志华.(2016).机器学习.北京: 清华大学出版社.

ID3算法局限性回顾

- **ID3算法局限性 (3~5)**
  - **连续属性**：现实任务中，属性不仅包含离散属性，还存在如身高、体重、密度等连续属性。连续属性的可取值数目无限，ID3算法无法直接处理无限取值属性
  - **缺失属性**：现实任务中，存在如因隐私保护问题导致的包含缺失属性的不完整样本，ID3算法无法直接处理存在属性值缺失的数据样本
  - **不同代价属性**：现实任务中，存在某些样本，其中不同的属性测量具有不同的代价，如不同的医疗检查需要不同的费用或时间，ID3算法无法直接处理具有不同代价的属性

连续值处理

● 连续值处理方法

➢基本思想：采用二分法 (Bi-Partition)对连续属性进行离散化处理

输入： 训练数据集D，属性集A

过程：

步骤1：给定样本集D和连续属性a，假定a在D上有n个不同取值，将这些值从小到大排序得到$\{a^1, a^2, \cdots, a^n\}$；

步骤2：计算候选划分点集合T_a，且基于划分点t，可将D分为子集D_t^+和D_t^- ；

步骤3：将使计算增益最大的划分点作为最佳划分属性值。

输出： 最佳划分属性值t_*

连续值处理

- 对于连续属性a，候选划分点集合表示为

$$T_a = \left\{ \frac{a^i + a^{i+1}}{2} \middle| 1 \le i \le n-1 \right\}$$

- 信息增益表示为

$$G(D,a) = \max_{t \in T_a} G(D,a,t)$$

$$= \max_{t \in T_a} \left(H(D) - \sum_{\lambda \in \{-,+\}} \frac{|D_t^\lambda|}{|D|} H\left(D_t^\lambda\right) \right)$$

  - 其中，$G(D,a,t)$是样本集D基于划分点t二分后的信息增益，因此需选择使$G(D,a,t)$最大的划分点t作为最佳划分属性值

连续值处理-示例

- **连续值处理示例——以判断西瓜好坏为例**

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

连续值处理-示例

- 计算候选划分点集合

$$T_{\text{密度}} = \{0.244, 0.294, 0.351, \cdots, 0.708, 0.746\}$$

$$T_{\text{含糖率}} = \{0.049, 0.074, 0.095, \cdots, 0.373, 0.126\}$$

- 计算信息增益

$$G(D, \text{密度}) = 0.262 \qquad t_{\text{密度}} = 0.381$$

$$G(D, \text{含糖率}) = 0.349 \qquad t_{\text{含糖率}} = 0.126$$

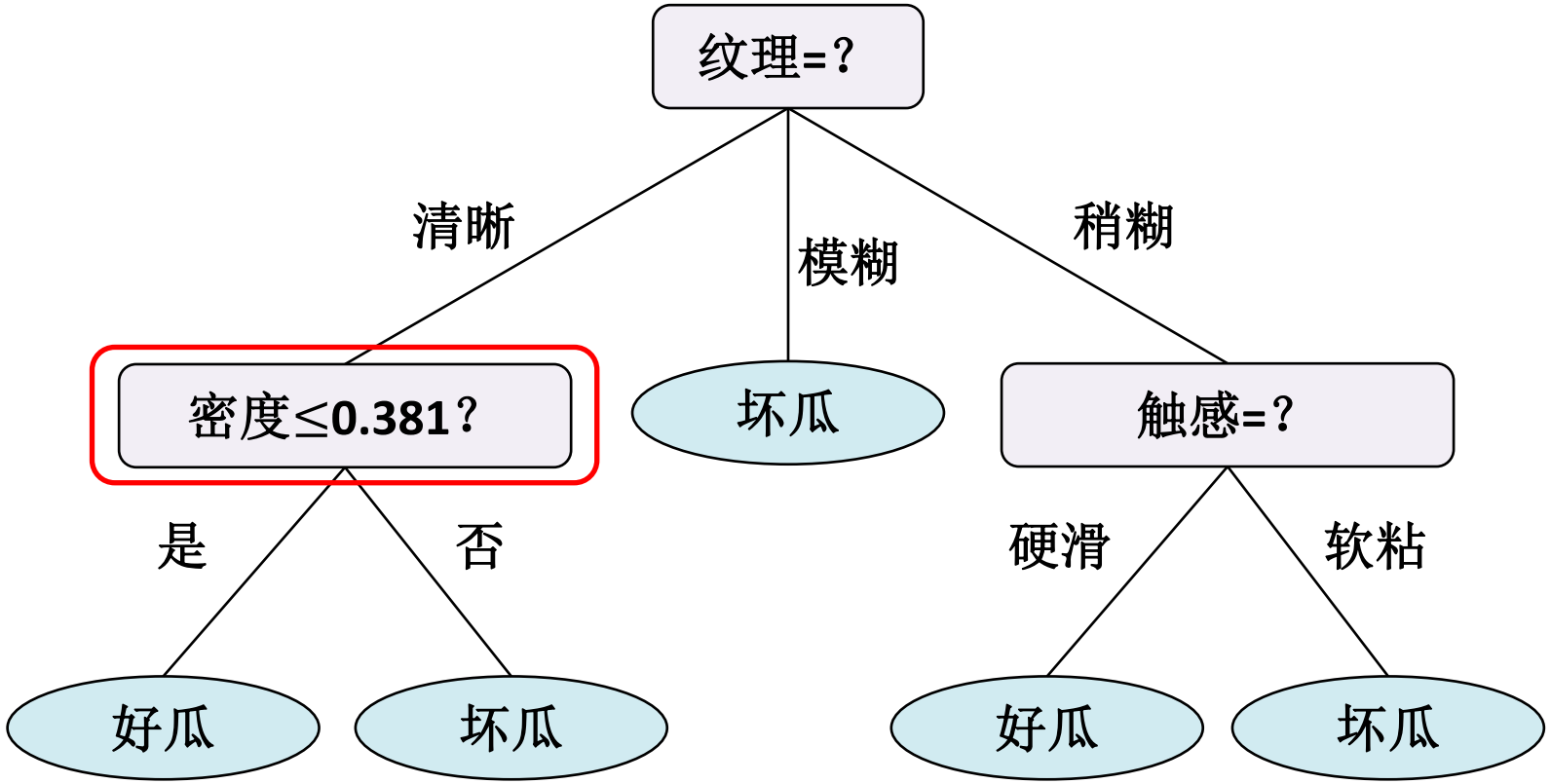
$$G(D, \text{色泽}) = 0.109 \qquad G(D, \text{根蒂}) = 0.143$$

$$G(D, \text{敲声}) = 0.141 \qquad \boxed{G(D, \text{纹理}) = 0.381}$$ 最佳划分属性

$$G(D, \text{脐部}) = 0.289 \qquad G(D, \text{触感}) = 0.006$$

连续值处理-示例

- 最终构建的决策树



缺失值处理

缺失值处理方法

1、在属性值缺失情况下，如何进行划分属性选择?
2、给定划分属性，若该样本在属性上的值缺失，如何对样本进行划分?

色泽	根蒂
—	蜷缩
乌黑	蜷缩
乌黑	蜷缩
青绿	蜷缩
—	蜷缩
青绿	稍蜷
乌黑	稍蜷
乌黑	稍蜷
乌黑	—
青绿	硬挺

编号	色泽	根蒂	敲声	纹理	脐部	触感
1	—	蜷缩	浊响	清晰	凹陷	硬滑
2	乌黑	蜷缩	沉闷	清晰	凹陷	—
3	乌黑	蜷缩	—	清晰	凹陷	硬滑

- 基本思想：在属性值缺失情况下，**仅使用无缺失值样本计算信息增益**，并选择最佳划分属性；在给定划分属性情况下，**将在该属性上的值缺失的样本以不同的概率划分到不同分支中**

缺失值处理

- 形式化定义
  - $\tilde{D}$为样本集D中在属性a上没有缺失值的样本子集；属性a有V个可能取值；$\tilde{D}^v$为$\tilde{D}$中在属性a上取值为a^v的样本子集；$\tilde{D}_k$为$\tilde{D}$中属于第k类的样本子集；ω_x为每个样本x的权重
- 缺失值处理方法
  - 在属性值缺失情况下，信息增益推广为

$$G(D,a) = \rho \times G(\tilde{D},a)$$

$$= \rho \times \left(H(\tilde{D}) - \sum_{v=1}^{V} \tilde{r}_v H(\tilde{D}^v) \right)$$

其中，$H(\tilde{D}) = -\sum_{k=1}^{K} \tilde{p}_k \log_2 \tilde{p}_k$

$\rho = \frac{\sum_{x\in\tilde{D}} \omega_x}{\sum_{x\in D} \omega_x}$为无缺失值样本所占比例；

$\tilde{p}_k = \frac{\sum_{x\in\tilde{D}_k} \omega_x}{\sum_{x\in\tilde{D}} \omega_x}$为无缺失样本中第$k$类所占比例；

$\tilde{r}_v = \frac{\sum_{x\in\tilde{D}^v} \omega_x}{\sum_{x\in\tilde{D}} \omega_x}$为无缺失样本中属性$a$上取值为$a^v$的样本所占比例；

缺失值处理

- **缺失值处理方法**
  - 在给定划分属性情况下，若样本x在划分属性a上的取值已知，则将x划入与取值对应的子结点，且样本权值在子结点中保持为ω_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，且样本权值在与属性值a^v对应的子结点中调整为$\tilde{r}_v \cdot \omega_x$，即令样本以不同的概率划分到不同子结点中

其中，$\rho = \frac{\sum_{x \in \tilde{D}} \omega_x}{\sum_{x \in D} \omega_x}$为无缺失值样本所占比例；

$\tilde{p}_k = \frac{\sum_{x \in \tilde{D}_k} \omega_x}{\sum_{x \in \tilde{D}} \omega_x}$为无缺失样本中第$k$类所占比例；

$\tilde{r}_v = \frac{\sum_{x \in \tilde{D}^v} \omega_x}{\sum_{x \in \tilde{D}} \omega_x}$为无缺失样本中属性$a$上取值为$a^v$的样本所占比例；

缺失值处理-示例

- 缺失值处理示例——以判断西瓜好坏为例

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

缺失值处理-示例

- **计算信息熵—以属性“色泽”为例**
  - **计算根结点的信息熵**

$$H(\tilde{D}) = -\left(\frac{6}{14}\log_2\frac{6}{14} + \frac{8}{14}\log_2\frac{8}{14}\right) = 0.985$$

  - **计算分支结点的信息熵**

$$H\left(\tilde{D}^{\text{青绿}}\right) = -\left(\frac{2}{4}\log_2\frac{2}{4} + \frac{2}{4}\log_2\frac{2}{4}\right) = 1.000$$

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

$$H\left(\tilde{D}^{\text{浅白}}\right) = -\left(\frac{0}{4}\log_2\frac{0}{4} + \frac{4}{4}\log_2\frac{4}{4}\right) = 0.000$$

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

缺失值处理-示例

- 计算信息增益—以属性“色泽”为例
  - 计算属性“色泽”的信息增益

$$G(\tilde{D}, \text{色泽}) = H(\tilde{D}) - \sum_{v=1}^{3} \tilde{r}_v H(\tilde{D}^v)$$

$$= 0.985 - \left(\frac{4}{14} \times 1.000 + \frac{6}{14} \times 0.918 + \frac{4}{14} \times 0.000 \right)$$

$$= 0.306$$

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

  - 计算其他属性的信息增益

$G(D, \text{根蒂}) = 0.171$　　$G(D, \text{敲声}) = 0.145$　　$G(D, \text{触感}) = 0.006$

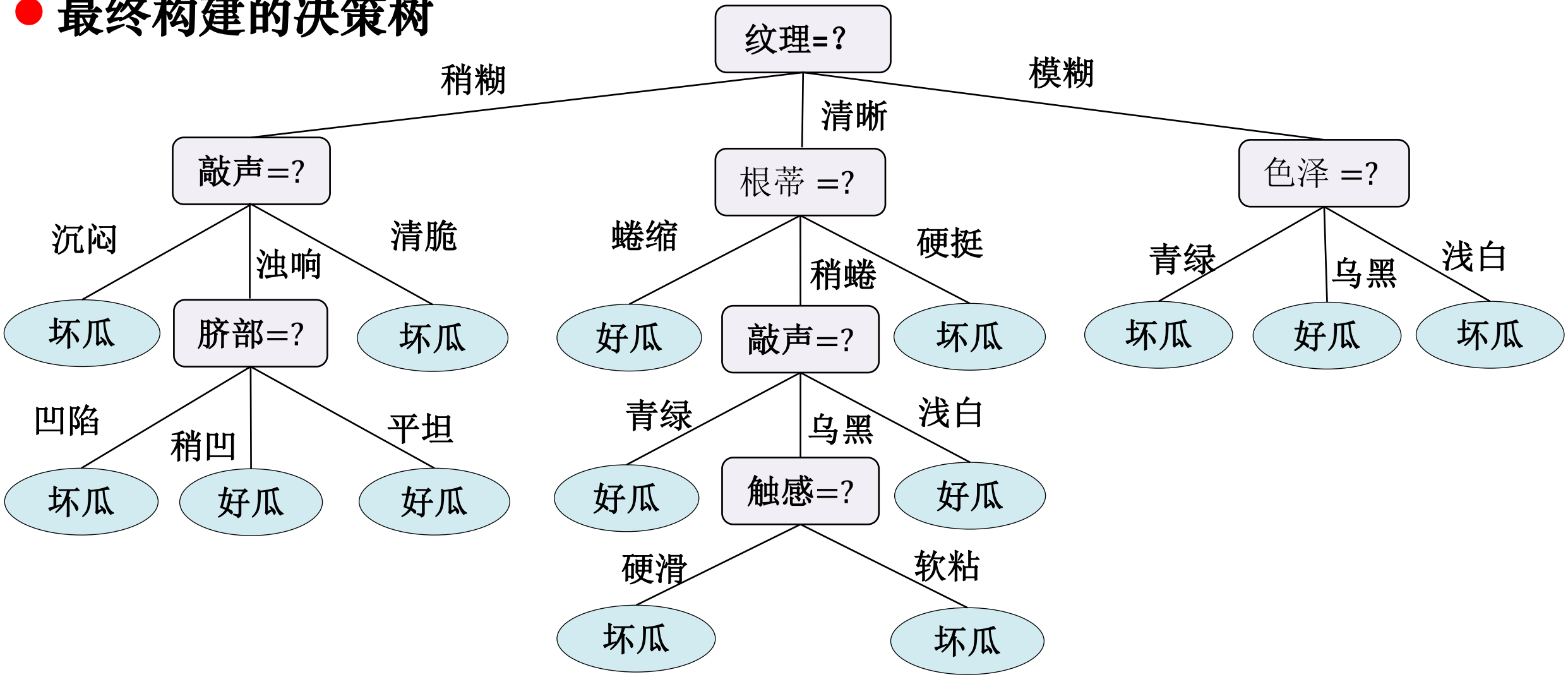
$G(D, \text{脐部}) = 0.289$　　$G(D, \text{纹理}) = 0.424$　　最佳划分属性

缺失值处理-示例

- 基于属性“纹理”对根结点进行划分
  - 对于属性不缺失的样本
    - “纹理=清晰”分支，包含编号为$\{1,2,3,4,5,6,15\}$的样本
    - “纹理=稍糊”分支，包含编号为$\{7,9,13,14,17\}$的样本
    - “纹理=模糊”分支，包含编号为$\{11,12,16\}$的样本
    - 且样本在各结点中的权重ω保持为1
  - 对于属性缺失的样本
    - 编号为$\{8,10\}$的样本同时进入三个分支，权重ω分别为$\frac{7}{15},\frac{5}{15},\frac{3}{15}$

缺失值处理-示例

- 最终构建的决策树



不同代价属性的处理

- 不同代价属性处理方法
  - 基本思想：在属性筛选度量中考虑属性的不同代价，优先选择低代价属性的决策树，在必要时才依赖高代价属性
  - 属性筛选度量标准1

$$G_{Cost}(D,a)=\frac{G(D,a)}{Cost(a)}$$

  - 属性筛选度量标准2

$$G_{Cost}(D,a)=\frac{2^{G(D,a)}-1}{(Cost(a)+1)^{\omega}}$$

其中，$Cost(a)$为属性a的代价；
$\omega \in [0,1]$ 为常数，决定代价对于信息增益的相对重要性；