第三章 决策树

§3.1 背景知识

§3.2 算法细节

§3.3 应用举例

§3.1 背景知识

一、问题的引入

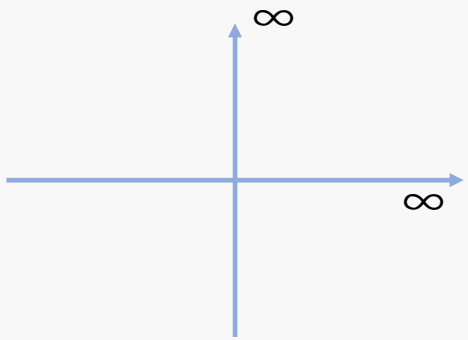
二、决策树组成

三、决策树推理

四、决策树学习

问题的引入

线性模型回顾



输入空间 $\boldsymbol{x} \in \mathbb{R}^m$

$$f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b$$

线性模型

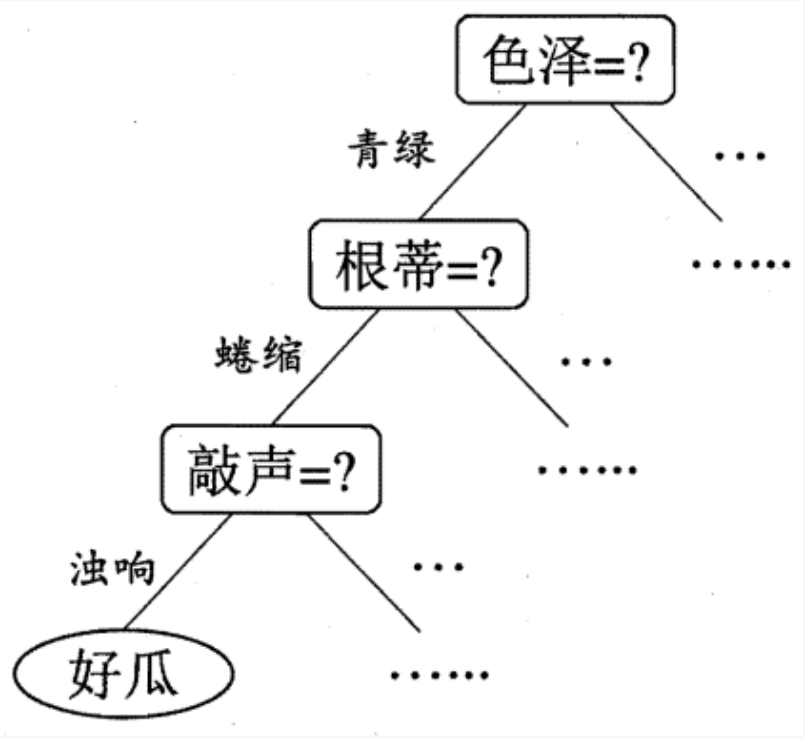
- 分类
- 回归

线性模型的不足

- 线性模型输入的特征向量一般由取值连续的实数组成，**如何处理非数值型数据的离散属性**？
- 线性模型具有一定的可解释性，但可解释性不强，**能否提出可解释性更强的机器学习模型**？

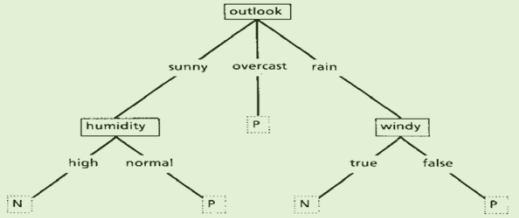
问题的引入

- 人类进行分类的机制
  - 机制：根据对象多种属性分情况进行讨论
  - 举例：如右图如何判定一个瓜是否为好瓜
  - 特点：
    - 属性取值可以离散，也可以是非数值
    - 每次根据单属性进行决策，可解释性强
    - 分类过程递归进行，形成树状结构
- 决策树是一种非参数监督机器学习方法，推理过程与人类进行分类的机制类似



决策树发展历程

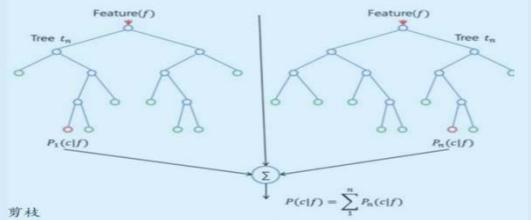
决策树的发展历程



E. B. Hunt提出Concept Learning System

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

CART算法・基尼指数
拓展到回归任务



随机森林
集成学习

1962 1979 1984 1992 2001

J. Ross Quinlan

$$\mathrm{Ent}(D) = -\sum_{k=1}^{|\mathcal{Y}|} p_k \log_2 p_k$$
$$\mathrm{Gain}(D,a) = \mathrm{Ent}(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Ent}(D^v)$$

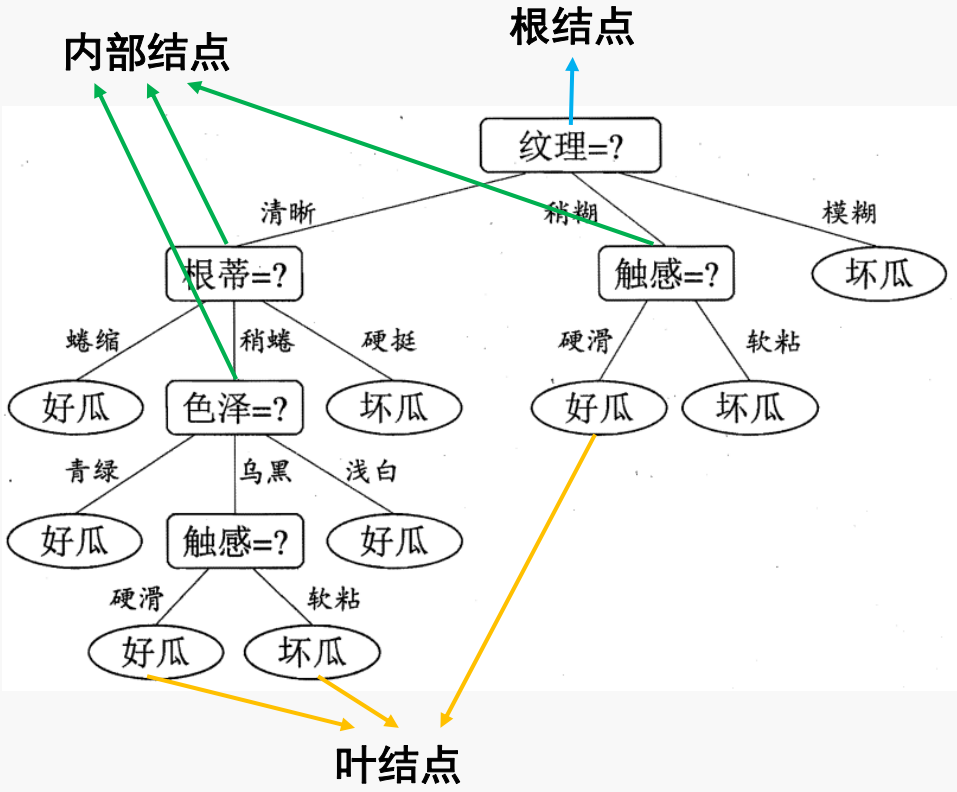
ID3算法・信息增益
首个决策树学习算法

$$\mathrm{Gain\_ratio}(D,a) = \frac{\mathrm{Gain}(D,a)}{\mathrm{IV}(a)}$$
$$\mathrm{IV}(a) = -\sum_{v=1}^{V} \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

C4.5算法・增益率
缺失值、连续值、剪枝等

决策树组成

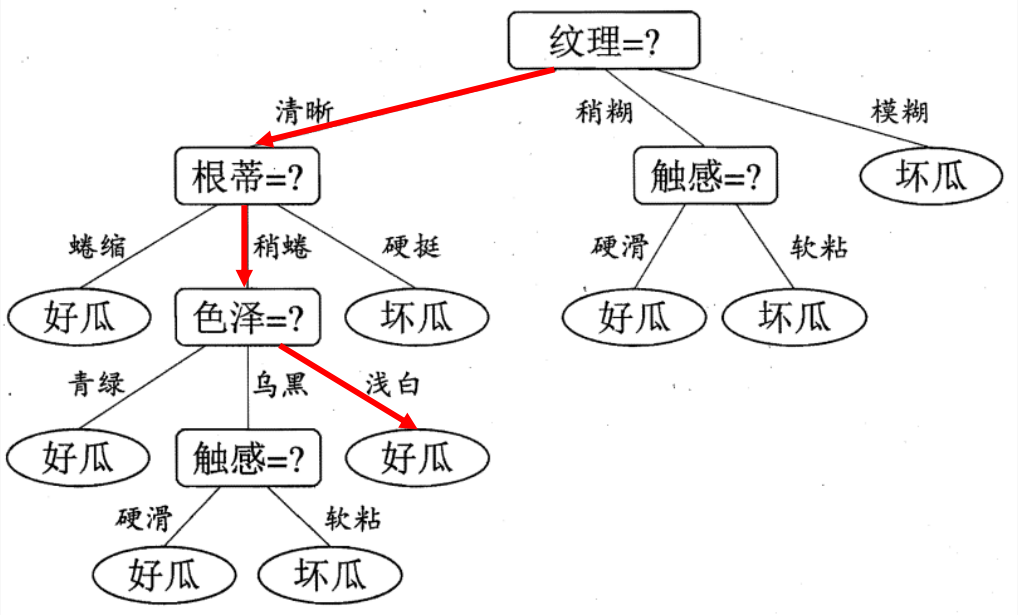
- 决策树的组成
  - 一棵决策树包含一个根结点，若干个内部结点和若干个叶结点
  - 叶结点对应于**决策结果**，其它每个结点对应于一个属性测试
  - 每个结点（除叶结点外）包含的样本集合根据属性测试的结果被划分到其子结点中，根结点包含样本全集
  - **根结点与内部结点无本质区别**



决策树推理

决策树的推理

- 对于一个样本，决策树的推理过程对应于寻找一条从根结点到某一叶结点的路径，也即一个判定测试序列
- 从数据中总结出决策规则，并用树状图的结构来呈现这些规则，预测结果由叶结点的标签决定



决策树学习

数据定义

- 样本集 $D = \{(x_1, y_1), \ldots, (x_m, y_m)\}$，属性集 $A = \{a_1, a_2, \ldots, a_d\}$

  x_i包含了第i个样本在A中各属性上的取值，y_i表示其标签

- 任意属性 $a \in A$ 可能有多个取值，表示为 $\{a^1, a^2, \ldots, a^V\}$

  若使用属性 a 对样本集进行划分，则可以将 D 分成 V 份，表示为 $\{D^1, D^2, \ldots, D^V\}$，其中 D^v 包含了 D 中所有在属性 a 上取值为 a^v 的样本

属性集 A

Day	Outlook	Temperature	Humidity	Wind	EnjoyTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes

D^1 D^2 D^3

样本集 D

决策树学习

算法要点

- 该算法递归地生成决策树，需注意**递归结束的条件**
- 如何**确定叶结点的标签**？
- 如何**选取结点的最优划分属性**？
- **非参数的有监督学习**

输入： 训练集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$;
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

决策树学习

- 递归结束条件
  - 如果继续划分，子树的预测结果也一定是类别C，对预测准确率没有影响
  - 此时D中样本除标签外完全相同，无法将它们分开
  - 样本集为空，无法继续划分

输入： 训练集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$;
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** $A = \varnothing$ **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

决策树学习

- 叶结点标签
  - 结点对应的样本集D非空，因此可以直接统计D上的类别分布（后验分布）确定叶结点的标签
  - 结点对应的样本集为空集，因此需要参考父节点的类别分布（先验分布）确定叶结点的标签

输入： 训练集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$;
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** $A = \varnothing$ **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

§3.2 算法细节

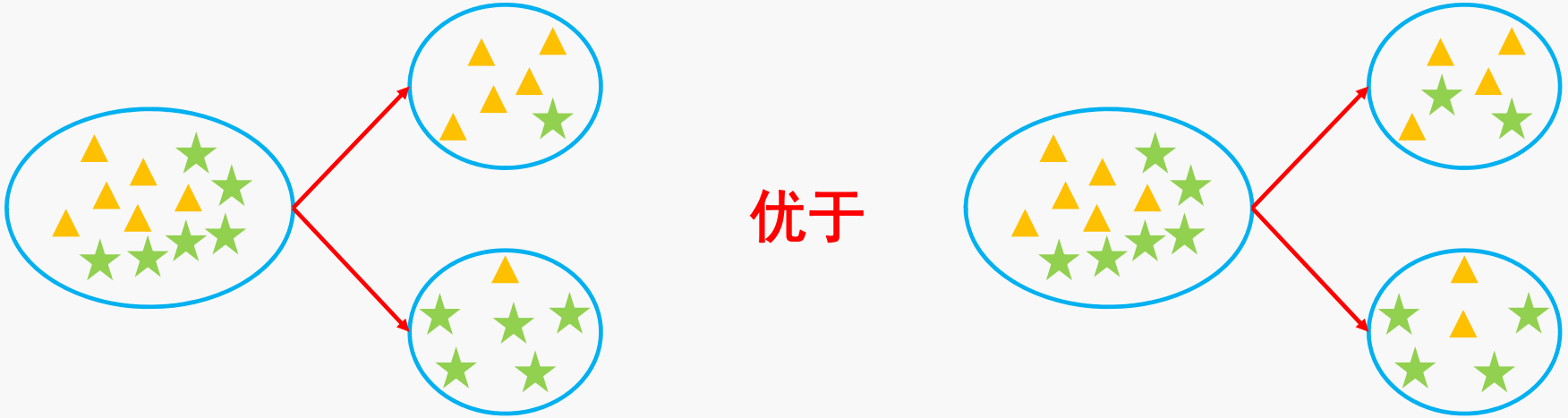
一、划分选择

二、剪枝处理

三、连续值处理

划分选择

- 对于样本集 D 和属性集 A，如何选取最优的划分属性?
- 直观理解



  - 左侧划分得到的子结点中，相同类别的样本占比更高，其“纯度”更高，分类错误率也更低
  - 如果利用一个属性进行分类的结果与随机分类的结果差别不大，则这个属性是没有分类能力的

划分选择：信息增益

- 理论基础
  - 熵：在信息论中，熵是表示随机变量不确定性的度量。设 X 是一个取有限个值的离散随机变量，其概率分布为

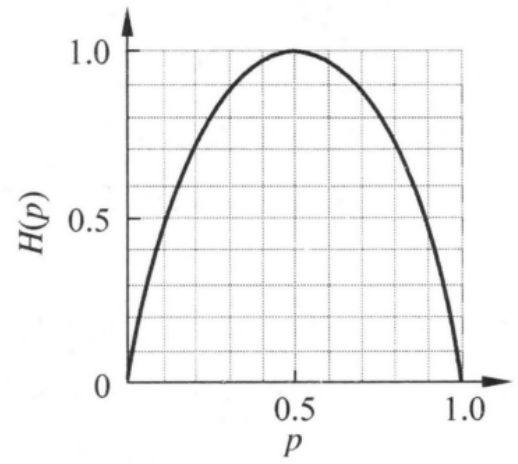
$$P(X = x_i) = p_i, \qquad i = 1, 2, \dots, n$$

则随机变量 X 的熵定义为

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

熵越大，随机变量的不确定越大

  - 伯努利分布的熵：
    - $H(p) = -p \log p - (1-p) \log(1-p)$
    - 当 $p = 0$ 或 1 时，随机变量完全确定
    - 当 $p = 1/2$ 时，随机变量的不确定性达到最大



划分选择：信息增益

理论基础

- **条件熵**：设离散随机变量 (X, Y)，其联合概率分布为

$$P(X = x_i, Y = y_i) = p_{ij}, \qquad i = 1, 2, .., n; \;\; j = 1, 2, .., m$$

条件熵 $H(Y|X)$ 表示在**已知随机变量 X 的条件下随机变量 Y 的不确定性**：

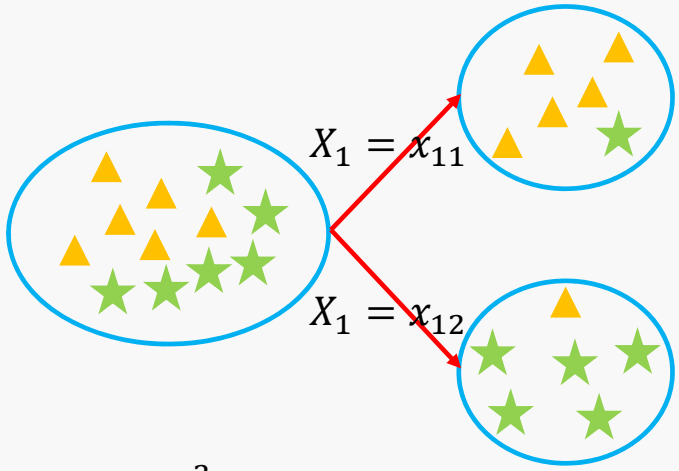
$$H(Y|X) = \sum_{i=1}^{n} P(X = x_i) H(Y|X = x_i)$$

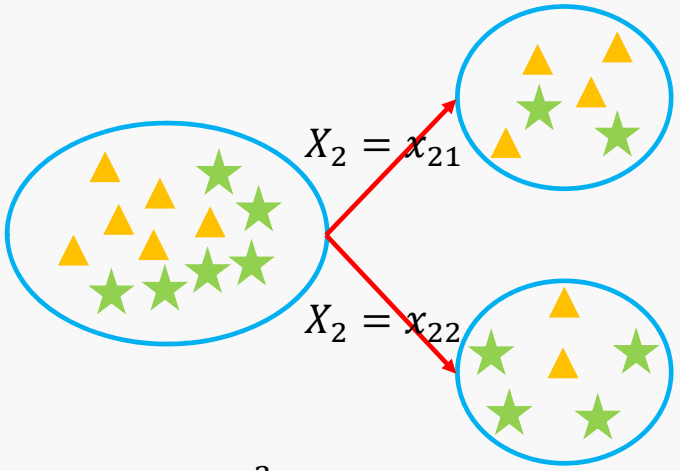
其中，$H(Y|X = x_i)$ 表示分布 $P(Y|X = x_i)$ 的熵。

- **经验熵**和**经验条件熵**：当熵和条件熵中的概率分布由**数据估计**（特别是极大似然估计）得到时，所对应的熵和条件熵分别称为经验熵和经验条件熵

划分选择：信息增益

- 举例：Y取值▲或★，在X确定时的经验条件熵





$$H(Y|X_1) = \sum_{i=1}^{2} P(X_1 = x_{1i})\, H(Y|X_1 = x_{1i})$$

$$= \frac{6}{12} * \left(-\frac{5}{6}\log\frac{5}{6} - \frac{1}{6}\log\frac{1}{6}\right) + \frac{6}{12} * \left(-\frac{1}{6}\log\frac{1}{6} - \frac{5}{6}\log\frac{5}{6}\right)$$

$$= 0.65$$

$$H(Y|X_2) = \sum_{i=1}^{2} P(X_2 = x_{2i})\, H(Y|X_2 = x_{2i})$$

$$= \frac{6}{12} * \left(-\frac{4}{6}\log\frac{4}{6} - \frac{2}{6}\log\frac{2}{6}\right) + \frac{6}{12} * \left(-\frac{2}{6}\log\frac{2}{6} - \frac{4}{6}\log\frac{4}{6}\right)$$

$$= 0.92$$

$H(Y|X_1) < H(Y|X_2)$ 说明在X_1确定时，Y的不确定度更小

划分选择：信息增益

- 信息增益
  - 代表在一个条件下信息不确定性减少的程度
  - 定义：属性 a 对样本集 D 的信息增益 $\mathrm{Gain}(D,a)$ 定义为样本集 D 的经验熵 $H(D)$ 与属性 a 给定条件下样本集 D 的经验条件熵 $H(D|a)$ 之差，即：

$$\begin{aligned}\mathrm{Gain}(D,a) &= H(D) - H(D|a)\\ &= H(D) - \sum_{v=1}^{V}\frac{|D^v|}{|D|}H(D^v)\end{aligned}$$

  - 上式中，$\frac{|D^v|}{|D|}$ 表示属性 a 的边缘分布；$H(D^v)$ 表示属性 a 取值 a^v 时，样本集 D 的条件概率分布的熵

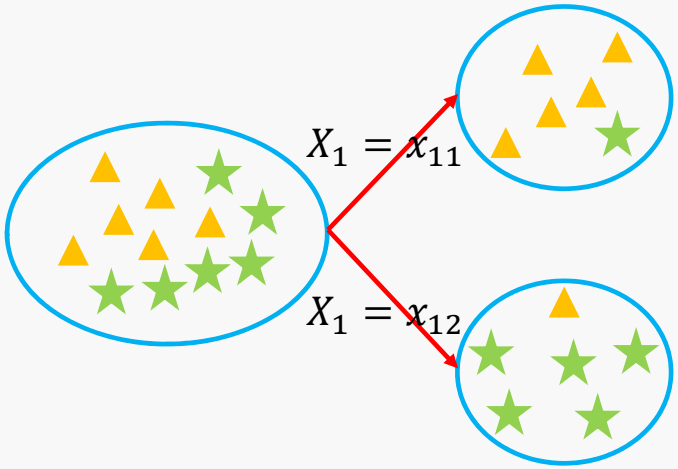
补充：

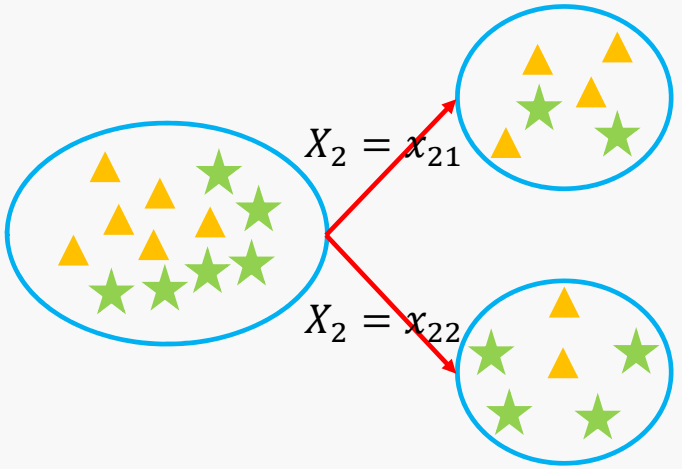
$$H(D) = -\sum_{k=1}^{|y|}\frac{|D_k|}{|D|}\log\frac{|D_k|}{|D|}$$

D_k 表示 D 中标签为 k 的样本组成的子集，式中的 D 也可以是 D^v。

划分选择：信息增益

- 举例：标签Y取值▲或★，划分属性选取X时的信息增益





$$\text{Gain}(D, X_1) = H(D) - H(D|X_1)$$

$$= \left(-\frac{6}{12}\log\frac{6}{12} - \frac{6}{12}\log\frac{6}{12}\right) - 0.65$$

$$= 1 - 0.65 = 0.35$$

$$\text{Gain}(D, X_2) = H(D) - H(D|X_2)$$

$$= \left(-\frac{6}{12}\log\frac{6}{12} - \frac{6}{12}\log\frac{6}{12}\right) - 0.92$$

$$= 1 - 0.92 = 0.08$$

$\text{Gain}(D, X_1) > \text{Gain}(D, X_2)$ 说明以X_1为划分属性，数据集D的不确定性减少更多

划分选择：信息增益

- 对信息增益的理解
  - **物理的角度**：熵可以简单地对应于样本集的“纯度”，熵越大，纯度越低。因此，信息增益可表示划分前后样本集纯度的增加量
  - **信息的角度**：样本集 D 的熵和条件熵实际上就是相应条件下**类别标签的熵**。因此，信息增益表示属性 a 确定前后，类别标签不确定性的减小量。类别标签的不确定性越低，越有利于对其进行准确预测
- 信息增益准则
  - 对样本集 D 计算其每个属性 $a \in A$ 的信息增益，并比较它们的大小，选择信息增益最大的属性进行划分
  - 著名的ID3算法就采用信息增益为准则来选择划分属性

划分选择：信息增益

- 本节使用的数据集如右表
  - 假设将所有数据都用于训练
  - 学习出一棵决策树，以判断一个没有剖开的西瓜是否为好瓜
- 信息增益准则的应用举例

  （确定根结点的最优划分属性）
  - 第一步，计算 $H(D)$
    - 根结点中正负样本占比分别为 $\frac{8}{17}$ 和 $\frac{9}{17}$
    - $H(D) = -\frac{8}{17}\log_2\frac{8}{17} - \frac{9}{17}\log_2\frac{9}{17} = 0.998$

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

划分选择：信息增益

- 信息增益准则的应用举例

（确定根结点的最优划分属性）

- 第二步，计算属性集 $A=\{$色泽，根蒂，纹理，触感$\}$ 中各属性的信息增益

（以色泽为例）

- 色泽属性共有 3 种取值，即{青绿，乌黑，浅白}，可将样本集划分为 3 个子集：

青绿：$D^1=\{1,\ 4,\ 6,\ 10,\ 13,\ 17\}$，$H(D^1)=-\frac{3}{6}log_2\frac{3}{6}-\frac{3}{6}log_2\frac{3}{6}=1.000$

乌黑：$D^2=\{2,\ 3,\ 7,\ 8,\ 9,\ 15\}$，$H(D^2)=-\frac{4}{6}log_2\frac{4}{6}-\frac{2}{6}log_2\frac{2}{6}=0.918$

浅白：$D^3=\{5,\ 11,\ 12,\ 14,\ 16\}$，$H(D^3)=-\frac{1}{5}log_2\frac{1}{5}-\frac{4}{5}log_2\frac{4}{5}=0.722$

- 计算色泽的信息增益：

$$\text{Gain}(D,\text{色泽})=H(D)-H(D|\text{色泽})=H(D)-\left(\frac{6}{17}H(D^1)+\frac{6}{17}H(D^2)+\frac{5}{17}H(D^3)\right)$$

$$=0.109$$

划分选择：信息增益

- 信息增益准则的应用举例

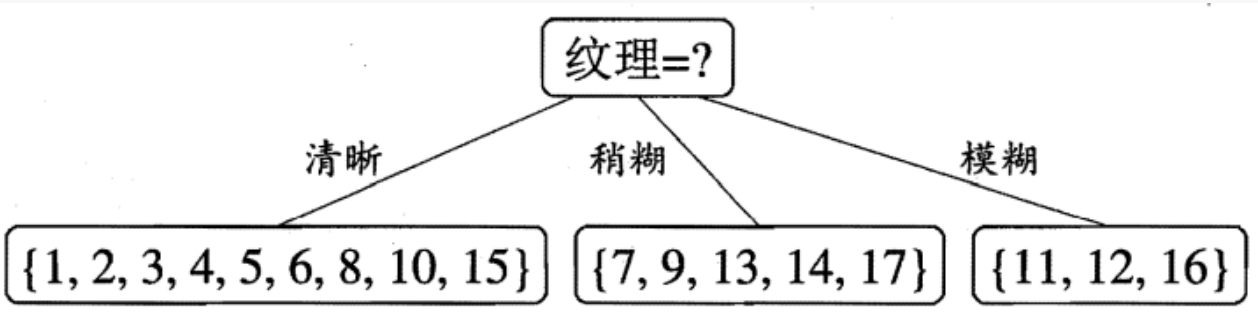
（确定根结点的最优划分属性）

- 第二步，计算属性集 $A=\{$色泽，根蒂，纹理，触感$\}$ 中各属性的信息增益

（同色泽理，可计算出所有属性的信息增益）

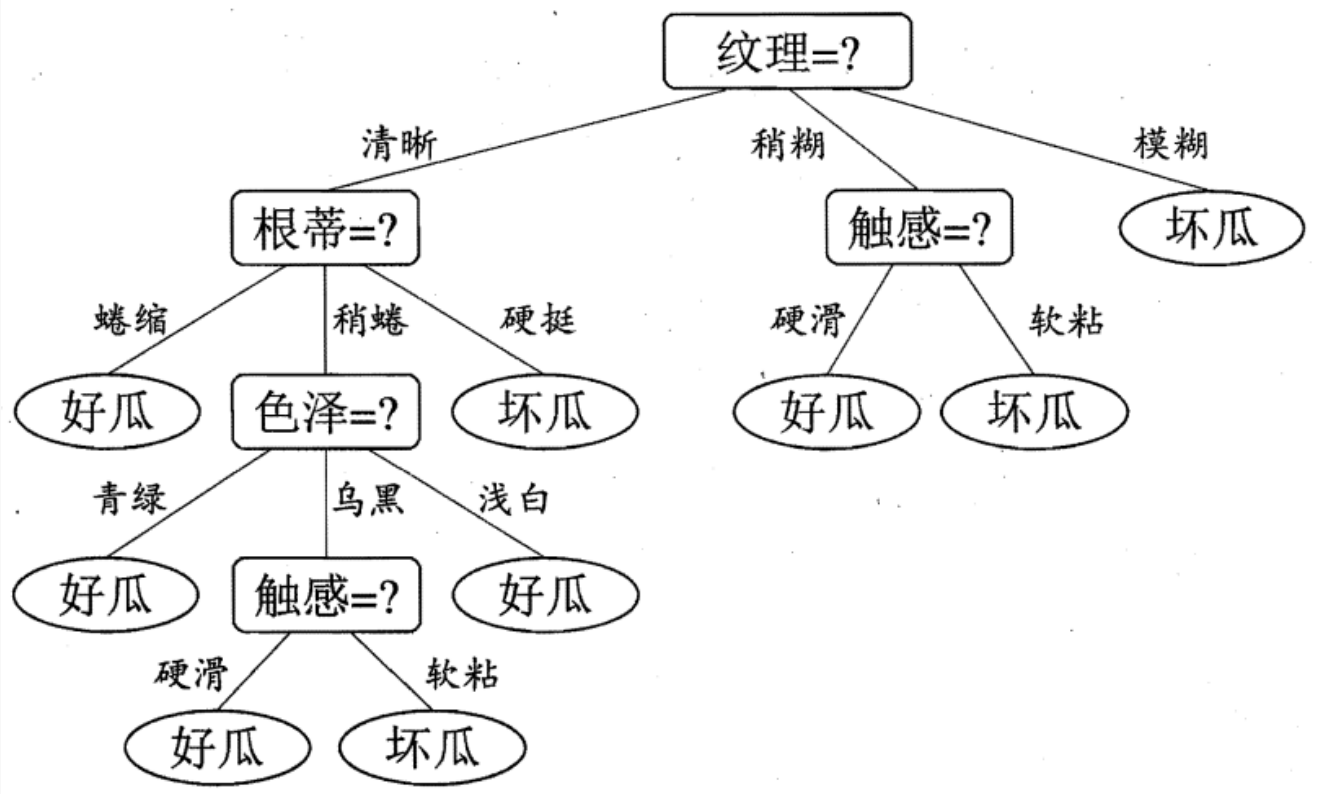
- $\mathbf{Gain}(D,$色泽$)=0.109$　　$\mathbf{Gain}(D,$根蒂$)=0.143$
- $\mathbf{Gain}(D,$纹理$)=0.381$　　$\mathbf{Gain}(D,$触感$)=0.006$

- 第三步，比较各属性的信息增益，取最大者作为最优划分属性
  - 显然，$\mathbf{Gain}(D,$纹理$)$最大，故取纹理作为根结点的划分属性



划分选择：信息增益

- 信息增益准则的应用举例
  - 对根结点的子结点**递归地执行上述过程**，最终可得到如下决策树



划分选择：增益率

- 信息增益的弊端
  - 若将右表中的“编号”也作为可选属性
    - 由于其每个子结点仅包含一个样本，则$H(D^v)=0,\ v=1,2,\ldots,17$，进一步有$\mathrm{Gain}(D,\text{编号})=H(D)=0.998$
    - 此时决策树仅进行一次划分后即停止递归。训练集准确率达到100%，但决策树完全没有泛化能力
  - 信息增益准则对可取值数目较多的属性有所偏好

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

划分选择：增益率

- 为减少信息增益准则对可取值数目较多的属性有所偏好所带来的不利影响，可使用**增益率**来选择最优划分属性
- 增益率
  - 属性 a 对样本集 D 的增益率 $\mathrm{GainRatio}(D,a)$ 定义为**其信息增益$Gain(D,a)$与属性 a 的固有值 $\mathrm{IV}(a)$ 之比**，即

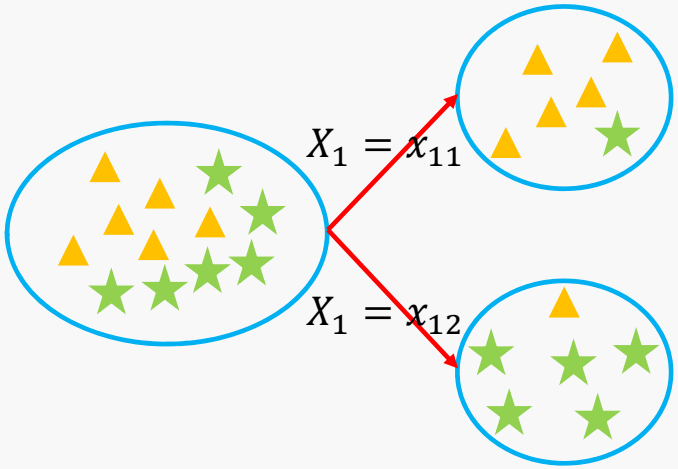
$$\mathrm{GainRatio}(D,a)=\mathrm{Gain}(D,a)/\mathrm{IV}(a)$$

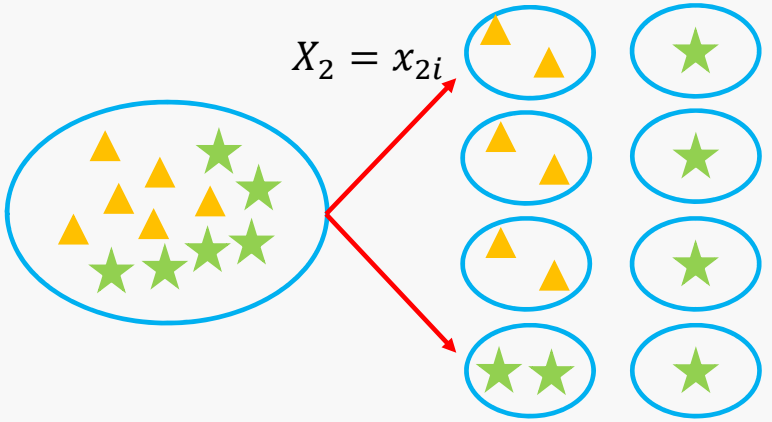
其中，$\mathrm{IV}(a)=-\sum_{v=1}^{V}\frac{|D^v|}{|D|}\log\frac{|D^v|}{|D|}$ 称为属性 a 的固有值，也即样本集 D 关于属性 a 的熵

  - 信息增益准则率**对可取值数目较少的属性有所偏好**
  - 著名的**C4.5算法即采用了增益率准则**：先从候选划分属性中找出信息增益高于平均水平的属性，再从从选择增益率最高的

划分选择：增益率

- 举例：根据增益率准则选取划分属性





$$\text{GainRatio}(D, X_1) = \text{Gain}(D, X_1)/\text{IV}(X_1)$$

$$= \frac{0.35}{-\frac{6}{12}\log\frac{6}{12} - \frac{6}{12}\log\frac{6}{12}} = 0.35$$

$$\text{Gain}(D, X_2) = H(D) - H(D|X_2) = 1 - 0 = 1$$

$$\text{GainRatio}(D, X_2) = \text{Gain}(D, X_2)/\text{IV}(X_2)$$

$$= \frac{1}{-4 * \frac{2}{12}\log\frac{2}{12} - 4 * \frac{1}{12}\log\frac{1}{12}} = 0.34$$

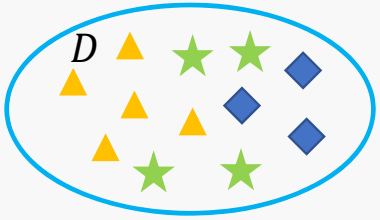
$\text{GainRatio}(D, X_1) > \text{GainRatio}(D, X_2)$ 表明归一化后，X_1为更优的划分属性

划分选择：基尼指数

- 著名的CART决策树使用了“基尼指数”来选择划分属性
- 基尼指数
  - **基尼值**：样本集 D 的“纯度”可用基尼值来度量：

$$\mathrm{Gini}(D) = 1 - \sum_{k=1}^{|y|} p_k^2$$

$\mathrm{Gini}(D)$ 反映了从样本集中随机抽取两个样本，其标签不一致的概率。 $\mathrm{Gini}(D)$ 越小，则样本集的纯度越高。



$$\mathrm{Gini}(D) = 1 - \sum_{k=1}^{3} p_k^2$$

$$= 1 - \left(\frac{5}{12}\right)^2 - \left(\frac{4}{12}\right)^2 - \left(\frac{3}{12}\right)^2$$

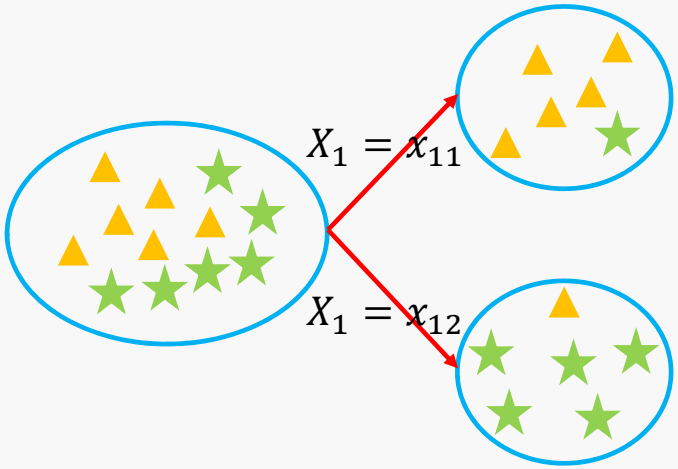
  - **基尼指数**：属性 a 各子结点的基尼值加权平均：

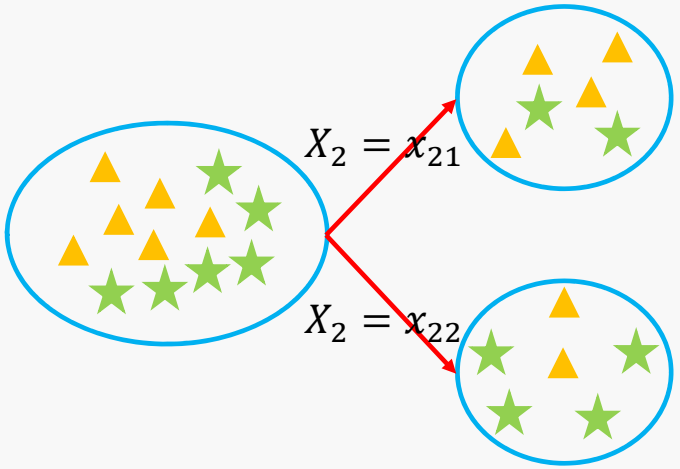
$$\mathrm{GiniIndex}(D, a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Gini}(D^v)$$

可选择基尼指数最小的属性作为最优划分属性

划分选择：基尼指数

- 举例：根据基尼指数准则选取划分属性





$$\text{GiniIndex}(D, X_1) = \sum_{v=1}^{2} \frac{|D^v|}{|D|} \text{Gini}(D^v)$$

$$= \frac{6}{12} * \left(1 - \left(\frac{5}{6}\right)^2 - \left(\frac{1}{6}\right)^2\right) * 2 = 0.28$$

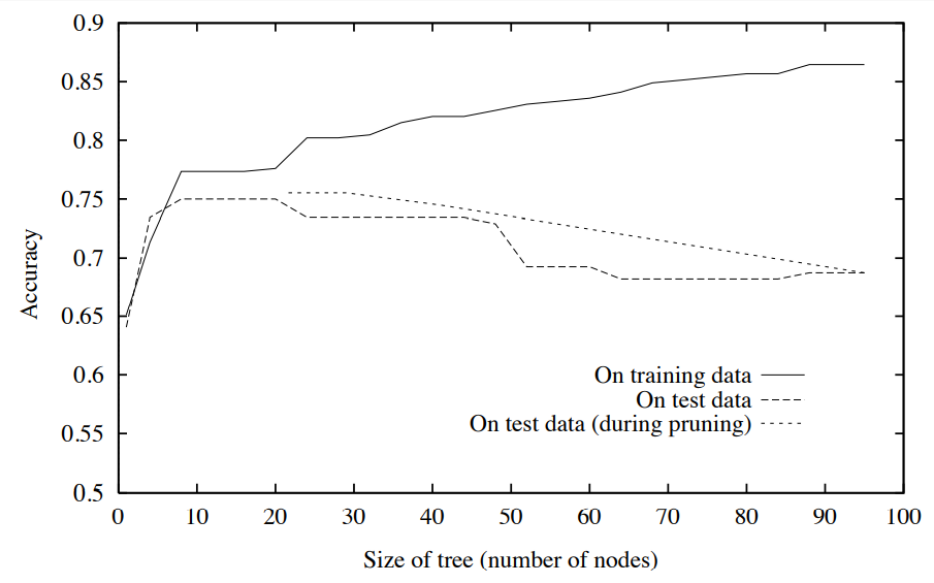
$$\text{GiniIndex}(D, X_2) = \sum_{v=1}^{2} \frac{|D^v|}{|D|} \text{Gini}(D^v)$$

$$= \frac{6}{12} * \left(1 - \left(\frac{4}{6}\right)^2 - \left(\frac{2}{6}\right)^2\right) * 2 = 0.44$$

$\text{GiniIndex}(D, X_1) < \text{GiniIndex}(D, X_2)$ 说明在基尼指数意义下，X_1为更优的划分属性

剪枝处理

- 剪枝是决策树学习算法对付“过拟合”的主要手段
  - “过拟合”发生的原因：为了尽可能正确分类训练样本，结点划分过程将不断重复，决策树分支过多，这时就可能把训练集自身的一些特点当作数据所具有的一般性质而导致过拟合
  - 解决的策略
    - 决策树模型的复杂度与叶结点的数目呈正相关
    - 通过主动去掉一些分支来减少叶结点数目，从而降低过拟合的风险



剪枝处理

- 剪枝策略的分类
  - 预剪枝：在决策树生成过程中，判断当前结点在划分后能否提升决策树的泛化能力。若不能，则停止划分并将当前节点标记为叶结点
  - 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提升泛化性能，则将该子树替换为叶结点
- 决策树泛化能力的评价
  - 采用留出法：即预留一部分数据用作“验证集”以进行性能评估

剪枝处理

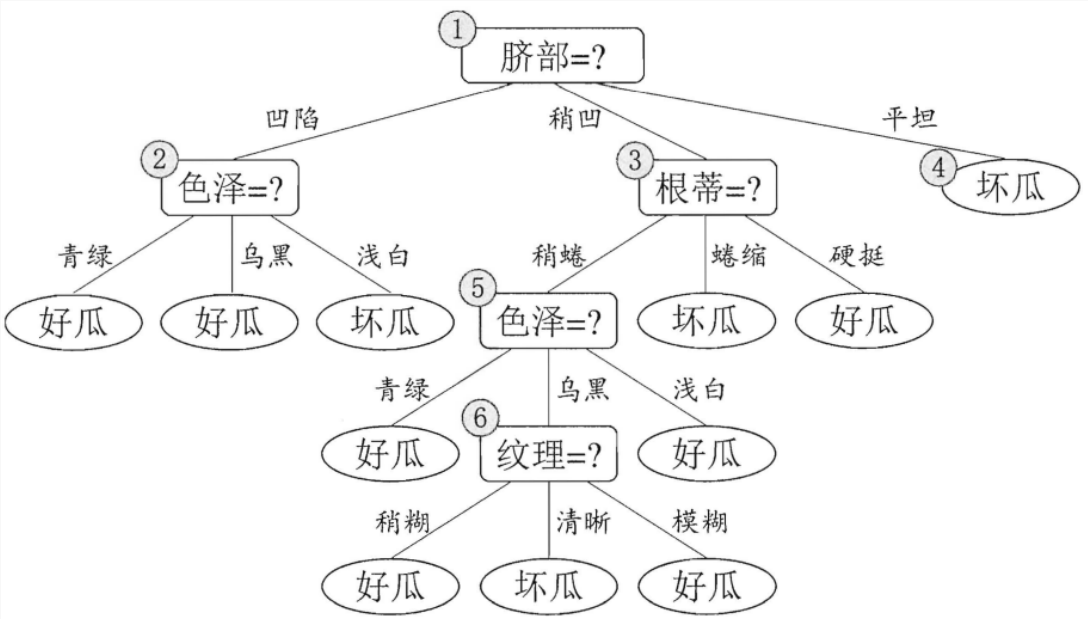
- 本节使用的数据集和使用信息增益准则生成的决策树

训练集

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

验证集

编号	色泽	根蒂	纹理	脐部	好瓜
4	青绿	蜷缩	清晰	凹陷	是
5	浅白	蜷缩	清晰	凹陷	是
8	乌黑	稍蜷	清晰	稍凹	是
9	乌黑	稍蜷	稍糊	稍凹	否
11	浅白	硬挺	模糊	平坦	否
12	浅白	蜷缩	模糊	平坦	否
13	青绿	稍蜷	稍糊	凹陷	否



不使用剪枝生成的决策树

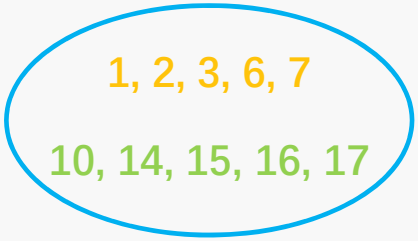
剪枝处理：预剪枝

- 在决策树**生成过程**中，通过测试集判断当前结点的划分是否有利于提升泛化性能
- 根据上面提供的数据集，使用**信息增益准则**和**预剪枝策略**
  - 第一步，计算划分根结点之前的测试集准确率。

    根结点包含所有训练集样本，正负样本比例为1：1

    根结点标签由占比最多的类别决定，不妨假设为正例

    划分前的测试集准确率为 $3/7 = 0.429$



划分前
训练集
标签：正例

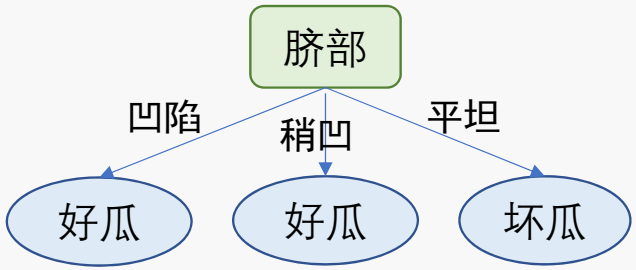
划分前
测试集
准确率：3/7

剪枝处理：预剪枝

- 根据上面提供的数据集，使用**信息增益准则**和**预剪枝策略**
  - 第二步（略），根据信息增益准则确定根结点的划分属性为“脐部”
  - 第三步，计算根结点划分之后的测试集准确率

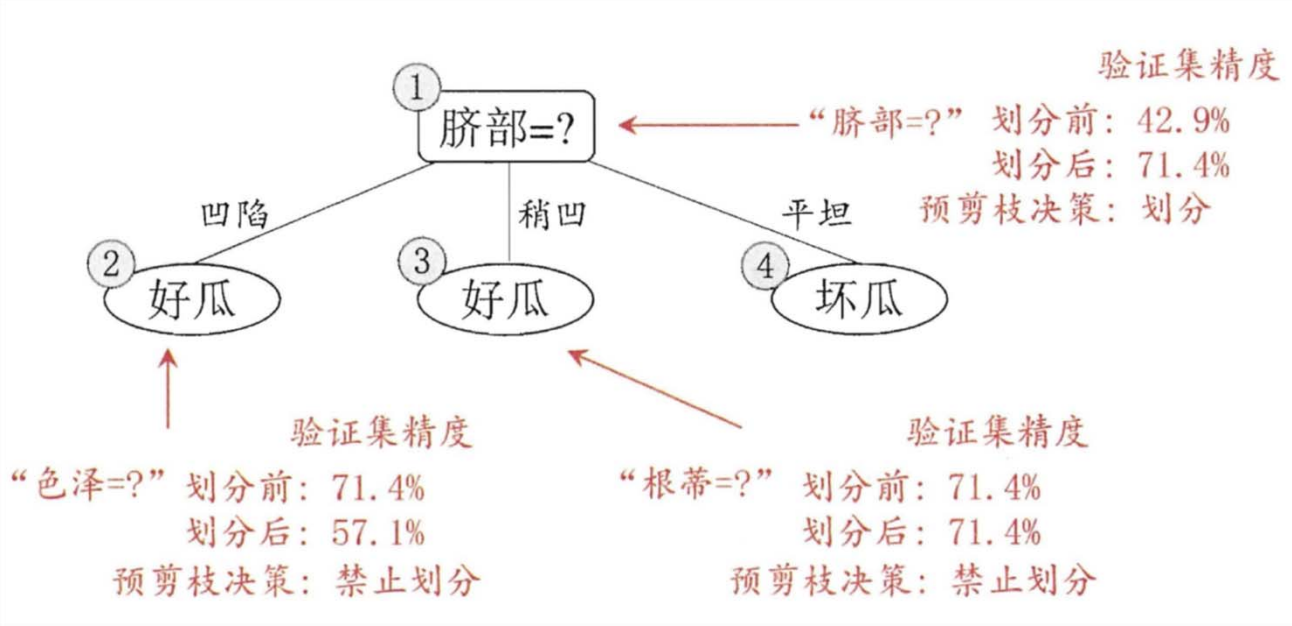
    根结点划分后得到的决策树如下图

    划分后的测试集准确率为 $5/7 = 0.714$
  - 第四步，$0.714 > 0.429$，应该划分根结点



剪枝处理：预剪枝

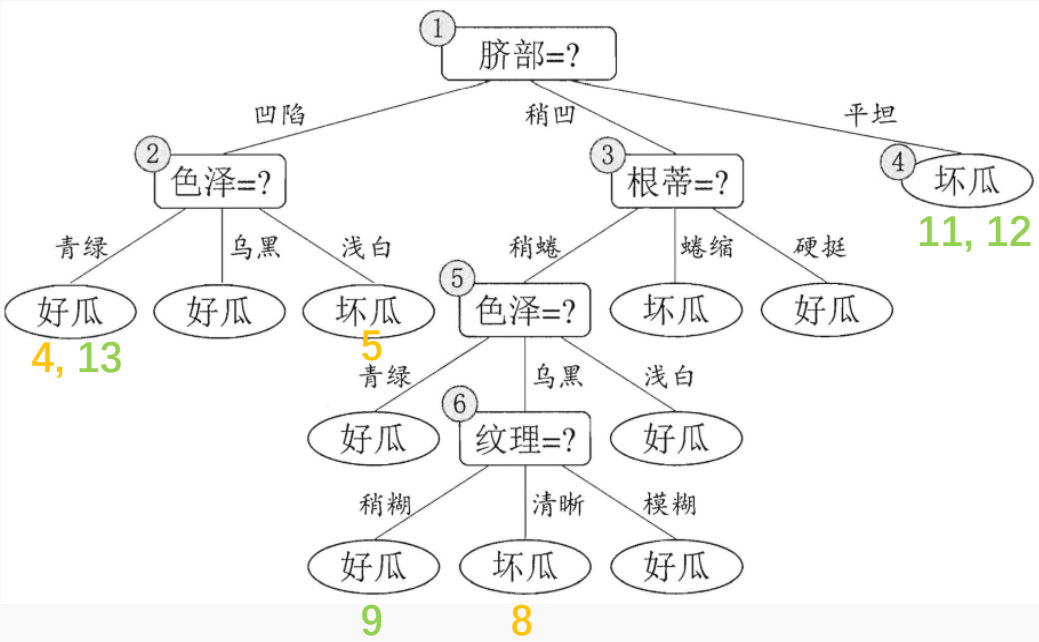
- 使用预剪枝策略的全过程如下



使用预剪枝生成的决策树

剪枝策略：后剪枝

- 原理：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，对比划分前后决策树的泛化性能
- 对第31页的决策树使用后剪枝策略
  - 第一步，确定后剪枝策略关注的当前结点。根据自底向上的顺序，首先考察6号结点
  - 第二步，计算剪枝前决策树在测试集上的准确率。只有编号为4，11，12的样本被正确预测，准确率为0.429



剪枝前
测试集
准确率：3/7

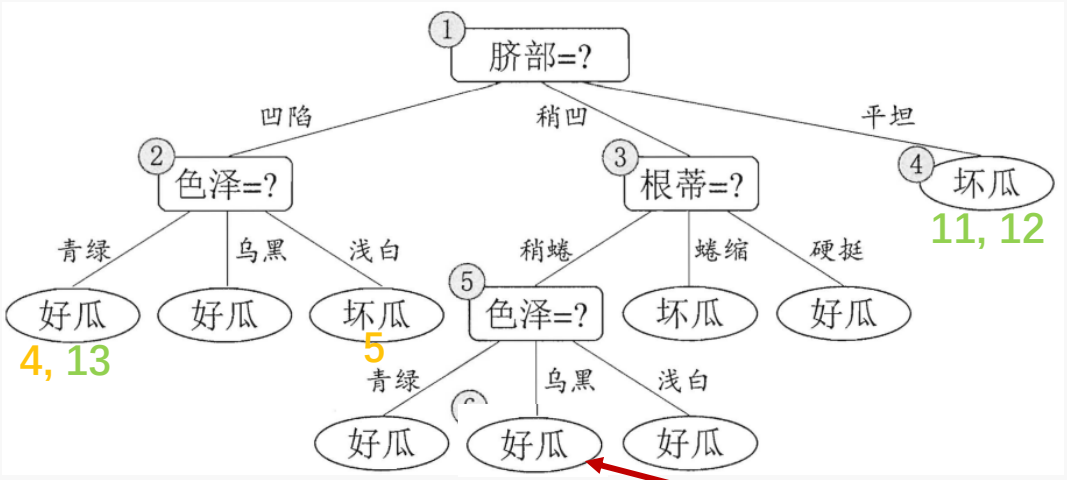
剪枝策略：后剪枝

- 对第31页的决策树使用后剪枝策略
  - 第三步，计算剪枝后决策树在测试集上的准确率

剪枝后，6号结点变为叶结点，包含7，15两个训练样本，不妨设其标签为正例

编号为4，8，11，12的样本被正确预测，准确率为0.571

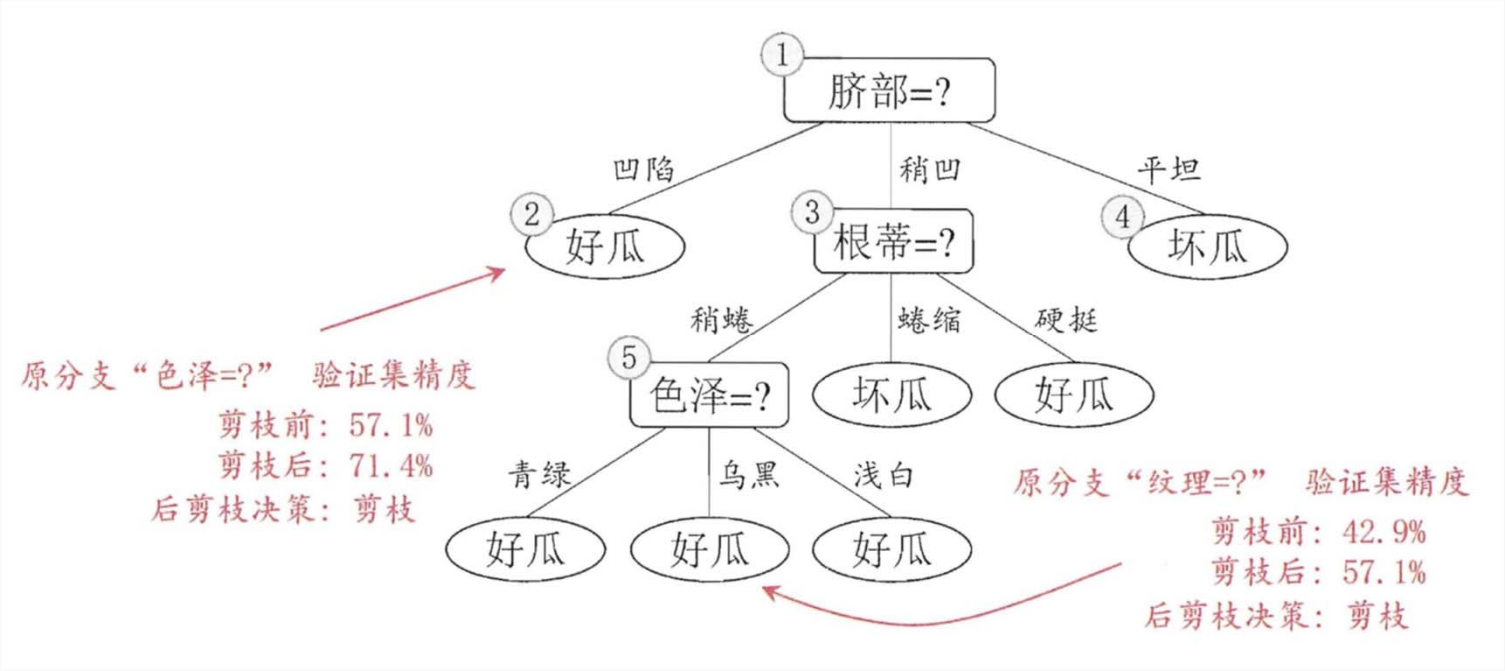
  - 第四步，因为0.571>0.429，应该对6号结点进行剪枝



剪枝后
测试集
准确率：4/7

剪枝处理：后剪枝

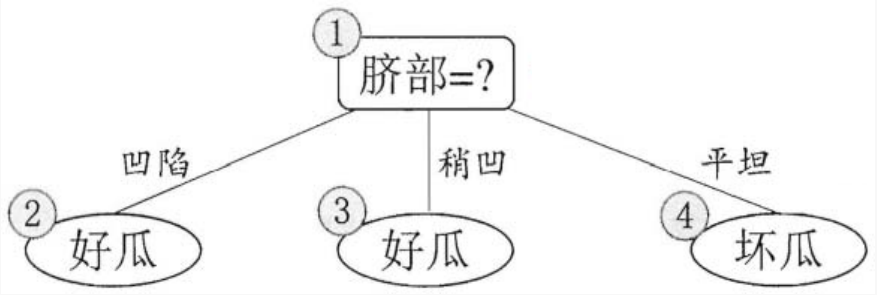
- 使用后剪枝策略的全过程如下



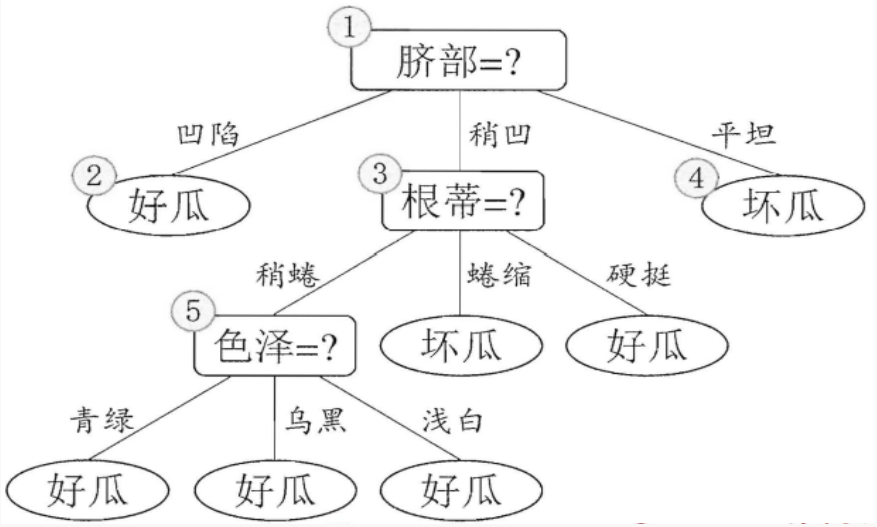
使用后剪枝生成的决策树

剪枝处理

- 预剪枝和后剪枝对比
  - 预剪枝得到的决策树往往比后剪枝简单
    - 预剪枝存在欠拟合风险
    - 后剪枝欠拟合风险更小，泛化性能往往更好
  - 后剪枝决策树的训练和推理时间开销远大于预剪枝决策树



预剪枝



后剪枝

连续值处理

- 前面讨论了各属性取离散值的情形，实际情况不一定如此
  - 如存在某一属性取值为连续实数，属性可取值数目为无限多，不能直接划分
- 连续值处理
  - C4.5决策树算法使用**二分法**处理连续值，假设连续属性 a 在样本集 D 上共有 n 个不同的取值，从小到大依次为 $\{a^1, a^2, \dots, a^n\}$
  - **候选划分点**：取相邻取值的中点为划分点，可得到 $n-1$ 个划分点

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

  - **计算各划分点信息增益**：每一个划分点 $t \in T_a$，都可将 D 划分为两部分，可据此计算划分点 t 的信息增益 $\mathrm{Gain}(D, a, t)$
  - **确定划分点**：选取信息增益最大的划分点作为最终划分点，其信息增益作为 a 的信息增益

连续值处理

- 二分法处理连续值举例

Temperature:	40	48	60	72	80	90
EnjoyTennis:	No	No	Yes	Yes	Yes	No

  - 第一步，确定候选的划分点

$$\{44, 54, 66, 76, 85\}$$

  - 第二步，计算各划分点的信息增益

$$\mathbf{Gain}(44) = 0.191 \quad \mathbf{Gain}(54) = 0.459$$

$$\mathbf{Gain}(66) = 0.082 \quad \mathbf{Gain}(76) = 0.000 \quad \mathbf{Gain}(85) = 0.191$$

  - 第三步，确定最终划分点和信息增益

因为 $\mathbf{Gain}(54)$ 最大，因此划分点确定为54，对应的信息增益为0. 459

缺失值处理

- 现实应用中经常遇到不完整的样本，即样本的某些属性值缺失。
  - 如右图所示，如果放弃不完整样本，则只剩下四个样本可以使用，**极大地浪费了数据集信息**！
  - **如何处理缺失值**？
    - 如何在属性值缺失的情况下选择划分属性？
    - 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？

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

- 如何处理缺失值？
  - 如何在属性值缺失的情况下选择划分属性？
    - 训练集中每个属性的**缺失程度**可能不一样，因此各属性信息增益不应该被平等对待，即它们的**可信程度**不一样
  - 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？
    - 不能简单地将该样本划入某一分支
    - 以**某一概率分布**划入不同分支
- 可以为每个样本 x 引入一个权重 ω_x
  - 用该权重修正信息增益准则
  - 用该权重表示样本分入不同分支的比例

缺失值处理

- 给定数据集 D 和属性 a，样本 x 的权重为 ω_x
  - $\tilde{D}$ ：D 中在属性 a 上没有缺失值的样本子集
  - $\tilde{D}^v$：$\tilde{D}$ 中在属性 a 上取值为 a^v 的样本子集
  - $\tilde{D}_k$：$\tilde{D}$ 中属于第 k 类的样本子集
  - ρ ：$\tilde{D}$ 占 D 的比例
  - $\tilde{p}_k$：$\tilde{D}$ 中第 k 类占的比例
  - $\tilde{r}_v$：$\tilde{D}^v$ 占 $\tilde{D}$ 的比例

用权重而非数量表示

编号	色泽	好瓜
1	–	是
2	乌黑	是
3	乌黑	是
4	青绿	是
5	–	是
6	青绿	是
7	乌黑	是
8	乌黑	是

编号	色泽	好瓜
9	乌黑	否
10	青绿	否
11	浅白	否
12	浅白	否
13	–	否
14	浅白	否
15	乌黑	否
16	浅白	否
17	青绿	否

$$\rho = \frac{\sum_{\boldsymbol{x}\in\tilde{D}} w_{\boldsymbol{x}}}{\sum_{\boldsymbol{x}\in D} w_{\boldsymbol{x}}},$$

$$\tilde{p}_k = \frac{\sum_{\boldsymbol{x}\in\tilde{D}_k} w_{\boldsymbol{x}}}{\sum_{\boldsymbol{x}\in\tilde{D}} w_{\boldsymbol{x}}} \quad (1 \leqslant k \leqslant |\mathcal{Y}|),$$

$$\tilde{r}_v = \frac{\sum_{\boldsymbol{x}\in\tilde{D}^v} w_{\boldsymbol{x}}}{\sum_{\boldsymbol{x}\in\tilde{D}} w_{\boldsymbol{x}}} \quad (1 \leqslant v \leqslant V).$$

以“色泽”为例：

$\tilde{\mathbf{D}} = \{2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17\}$

$\tilde{\mathbf{D}}(v = \text{乌黑}) = \{2, 3, 7, 8, 9, 15\}$

$\tilde{\mathbf{D}}(k = \text{好瓜}) = \{2, 3, 4, 6, 7, 8\}$

$\tilde{\mathbf{D}}(k = \text{好瓜}|v = \text{乌黑}) = \{2, 3, 7, 8\}$

缺失值处理

推广后的信息增益

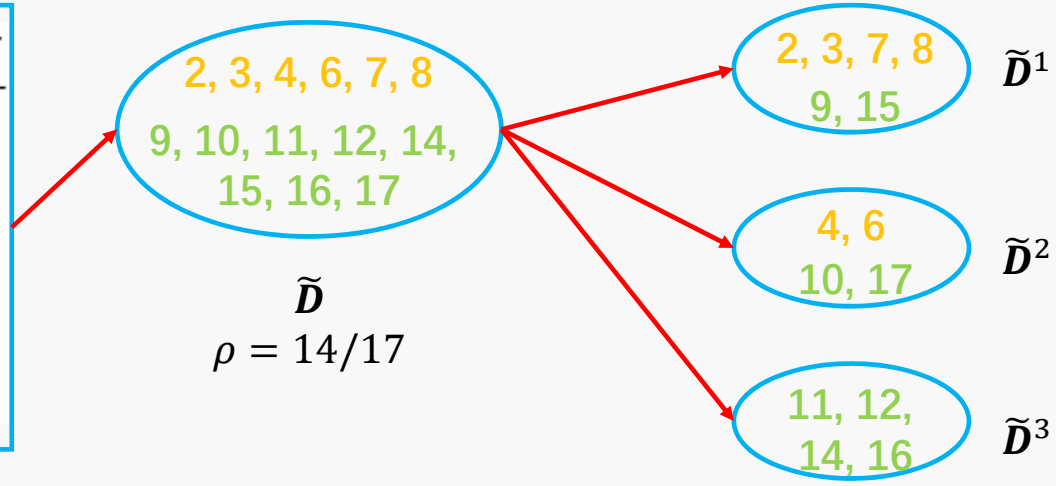
$$\mathrm{Gain}(D,a)=\rho\times\mathrm{Gain}(\tilde{D},a)=\rho\times\left(H(\tilde{D})-\sum_{v=1}^{V}\tilde{r}_v H(\tilde{D}^v)\right)$$

$$H(\tilde{D})=-\sum_{k=1}^{|y|}\tilde{p}_k\log_2\tilde{p}_k$$

- 使用 ρ 对 $\tilde{D}$ 上的信息增益加权，即考虑了属性 a 的缺失程度
- 在估计样本的类别分布 $\tilde{p}_k$ 时，需要将每个样本的贡献由“1”替换为权重

编号	色泽	好瓜	编号	色泽	好瓜
1	–	是	9	乌黑	否
2	乌黑	是	10	青绿	否
3	乌黑	是	11	浅白	否
4	青绿	是	12	浅白	否
5	–	是	13	–	否
6	青绿	是	14	浅白	否
7	乌黑	是	15	乌黑	否
8	乌黑	是	16	浅白	否
			17	青绿	否

D

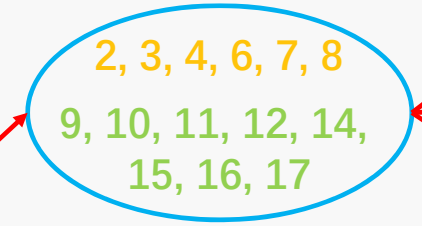


缺失值处理

- 举例：划分根结点时“色泽”属性的信息增益计算
  - 因为是根结点，每个样本的初始权重为1.0

D

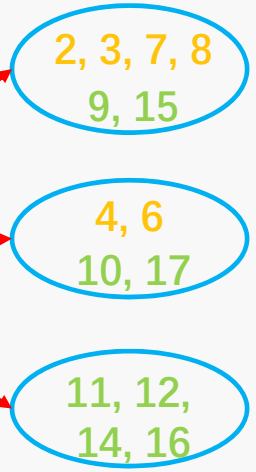
编号	色泽	好瓜	编号	色泽	好瓜
1	–	是	9	乌黑	否
2	乌黑	是	10	青绿	否
3	乌黑	是	11	浅白	否
4	青绿	是	12	浅白	否
5	–	是	13	–	否
6	青绿	是	14	浅白	否
7	乌黑	是	15	乌黑	否
8	乌黑	是	16	浅白	否
			17	青绿	否



$\tilde{D}$

$\rho = 14/17$

$\tilde{p}_1 = 6/14,\ \tilde{p}_2 = 8/14$



$\tilde{D}^1$

$\tilde{r}_1 = 6/14$

$\tilde{p}_1 = 4/6,\ \tilde{p}_2 = 2/6$

$\tilde{D}^2$

$\tilde{r}_2 = 4/14$

$\tilde{p}_1 = 2/4,\ \tilde{p}_2 = 2/4$

$\tilde{D}^3$

$\tilde{r}_3 = 6/14$

$\tilde{p}_1 = 0/4,\ \tilde{p}_2 = 4/4$

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

同理，可计算得：

$H(\tilde{D}^1) = 0.918,\quad H(\tilde{D}^2) = 1.000,\quad H(\tilde{D}^3) = 0.000$

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

$$= 0.985 - \frac{6}{14} * 0.918 - \frac{4}{14} * 1.000 - \frac{6}{14} * 0 = 0.306$$

推广后的信息增益为

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

缺失值处理

举例：确定根结点的划分属性后，样本的分配规则

- 通过比较各属性的（推广）信息增益，可确定根结点的划分属性为“纹理”
- 对于在纹理属性上不为空的样本，直接将其**在父节点中的权重全部分入对应分支**的子结点即可
  - $\{1,2,3,4,5,6,15\}$ 分入清晰分支
  - $\{7,9,13,14,17\}$ 分入稍糊分支
  - $\{11,12,16\}$ 分入模糊分支
- 对于在纹理属性上为空的样本，**按照** $\{\tilde{r}_v\}_{v=1,\ldots,V}$ **分配**，8号和10号样本分入三分支的权重分别为$\frac{7}{15}$，$\frac{5}{15}$和$\frac{3}{15}$

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

§3.4 应用举例

一、决策树应用场景

二、C4.5决策树及应用

三、思考题

决策树应用场景

- 何时考虑决策树方法？
  - 样本由“属性-值”对描述
  - 特征的不同元素之间有较大的类型差异
  - 对模型的可解释性有要求

- 应用举例
  - 故障诊断
  - 信用分析
  - 日程规划

C4.5决策树及应用

C4.5决策树简介

- 划分准则：改进的增益率准则

  因为增益率偏好取值数目更少的属性，因此C4.5决策树先从候选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

- C4.5决策树采用的其他技术
  - 剪枝处理
  - 连续值处理
  - 缺失值处理
  - 属性的代价

C4.5决策树及应用

- 缺失值处理
  - **如何为含有缺失值的属性计算信息增益**：在不含缺失值的样本子集上计算信息增益，并用样本子集占样本集的比例进行加权
  - **如何将含有缺失值的样本划分进子结点**：为每个样本赋予一个权重，按子结点的样本数分配权重
- 属性的代价
  - **获取不同属性的代价可能不一样**：在医疗领域年龄属性很容易得到但MRI检测结果则代价更大，代价更高的属性往往缺失值的比例更高
  - **如何尽量避免使用代价高的属性**：
    - 引入惩罚机制：$\mathbf{GainRatio}_{\mathbf{new}} = \frac{\mathbf{GainRatio}}{\mathbf{Cost}}$
    - Cost需要手工指定

C4.5决策树及应用

银行个人信用评级

- 数据集：某个人信贷数据集合，包含1000条记录，每一条记录由21个字段组成。前20个字段构成属性集合，最后一个字段是信用评级，分为“好客户”和“坏客户”
- 任务：训练机器学习模型，判断样本的信用评级

	样本集合字段归类
Character(特征)	信贷期限、信贷历史纪录、贷款目的、贷款金额、其他分期付款方式、在本银行现有的信贷纪录数
Capacity(能力)	现有支票账户、分期付款金额占可支配收入的比率、工作、法律规定需要扶养的人数
Capital(资本)	储蓄存款账户
Collateral(抵押担保)	其他债务人/保证人、资产
Condition(环境和条件)	年龄、个人身份和性别
Stability (稳定性)	现任工作时间、在目前住址居住时间、住房、电话注册、是否外国国籍

C4.5决策树及应用

- 银行个人信用评级
  - 动机
    - 个人信用评级中包含大量的非数值离散数据（如学历，职业等），不能直接使用神经网络技术和支持向量机等技术
    - 决策树是数据挖掘领域应用最广泛的方法之一，有着准确率高、简单、高效等优点，且对输入属性的类别没有强制要求
  - 方法
    - 以C4.5决策树为基础，做出以下改进：
    - 引入Boosting技术（后面集成学习会涉及）
    - 引入误判的成本矩阵，构造对代价敏感的决策树

C4.5决策树及应用

- 银行个人信用评级
  - 改进前后的决策树都能取得较低的错误率
  - 引入Boosting技术提高了C4.5决策树的性能
  - 引入成本矩阵显著降低了将“差客户”预测为“好客户”的概率

表 7 成本矩阵

实际类别 \ 预测类别	C21	1 (好客户)	2 (差客户)
1 (好客户)		0	1
2 (差客户)		2	0

C4.5决策树在测试集上的性能

		预测类别		合计	正确率 (%)	错误率 (%)
		1(好客户)	2(差客户)			
实际类别	1(好客户)	**120**	22	142	84.51	15.49
	2(差客户)	42	**17**	59	28.81	71.19
合计				201	68.16	31.84

改进的C4.5决策树在测试集上的性能

		预测类别		合计	正确率 (%)	错误率 (%)
		1(好客户)	2(差客户)			
实际类别	1(好客户)	101	41	142	71.13	28.87
	2(差客户)	19	40	59	67.8	32.2
合计				201	70.15	29.85%

思考题

- 下表由15个样本组成的贷款申请数据集，包括申请人的年龄、收入情况、是否有车、信用情况等四项属性，最后一列为是否同意贷款作为预测结果，请采用决策树进行预测

（1）假如我们按照小于30岁、30到60岁、60岁以上将申请人的年龄分为三组，请利用ID3算法建立决策树，写出决策树的建立过程并画出最终的决策树结构

（2）将（1）中的ID3算法换为C4.5算法重新建树，写出决策树的建立过程并画出最终的决策树结构

序号	年龄	是否有车	收入情况	信用情况	是否同意贷款
1	19	否	一般	一般	否
2	22	否	一般	好	否
3	75	否	一般	一般	否
4	21	否	一般	一般	否
5	36	否	一般	一般	否
6	40	否	一般	好	否
7	69	是	一般	好	是
8	45	是	良好	好	是
9	52	是	一般	非常好	是
10	66	是	一般	非常好	是
11	25	否	良好	好	是
12	42	是	一般	非常好	是
13	60	否	良好	好	是
14	61	否	良好	非常好	是
15	29	是	良好	一般	是

思考题

（1） 对训练数据集D，首先计算划分前样本的信息熵：

$$I(D)=-\frac{9}{15}\log_2\frac{9}{15}-\frac{6}{15}\log_2\frac{6}{15}=0.9710$$

然后计算各特征对数据集D的信息增益，分别以A_1,A_2,A_3,A_4表示年龄、收入情况、有车和信贷情况4个特征，则：

$$\begin{aligned}\Delta I(D,A_1)&=I(D)-\left[\frac{5}{15}I(D_1)+\frac{6}{15}I(D_2)+\frac{4}{15}I(D_3)\right]\\&=0.9710\\&\quad-\left[\frac{5}{15}\left(-\frac{2}{5}\log_2\frac{2}{5}-\frac{3}{5}\log_2\frac{3}{5}\right)+\frac{5}{15}\left(-\frac{2}{5}\log_2\frac{2}{5}-\frac{3}{5}\log_2\frac{3}{5}\right)\right.\\&\quad\left.+\frac{5}{15}\left(-\frac{1}{5}\log_2\frac{1}{5}-\frac{4}{5}\log_2\frac{4}{5}\right)\right]=0.0830\end{aligned}$$

$$\Delta I(D,A_2)=0.9710-\left[0+\frac{10}{15}\left(-\frac{4}{10}\log_2\frac{4}{10}-\frac{6}{10}\log_2\frac{6}{10}\right)\right]=0.3237$$

$$\Delta I(D,A_3)=0.9710-\left[0+\frac{9}{15}\left(-\frac{3}{9}\log_2\frac{3}{9}-\frac{6}{9}\log_2\frac{6}{9}\right)\right]=0.4200$$

$$\begin{aligned}\Delta I(D,A_4)&=0.9710\\&\quad-\left[\frac{5}{15}\left(-\frac{1}{5}\log_2\frac{1}{5}-\frac{4}{5}\log_2\frac{4}{5}\right)+\frac{6}{15}\left(-\frac{2}{6}\log_2\frac{2}{6}-\frac{4}{6}\log_2\frac{4}{6}\right)+0\right]\\&=0.3630\end{aligned}$$

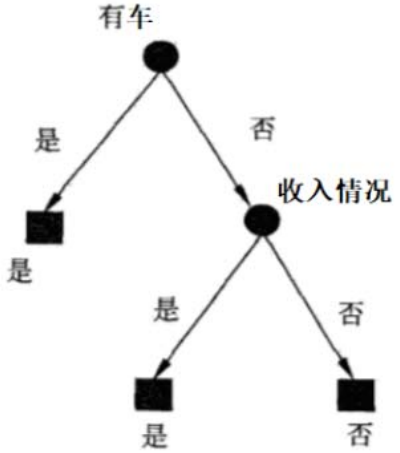
所以首先选择A_3作为分类特征，这将训练集D划分为两个子集D_1（A_3取是）和D_2（A_3取否），由于D_1只有同一类的样本点，所以它成为一个叶节点，类标记为“是”。对于D_2，继续挑选特征，计算信息增益：

$$I(D_2)=-\frac{3}{9}\log_2\frac{3}{9}-\frac{6}{9}\log_2\frac{6}{9}=0.9183$$

$$\begin{aligned}\Delta I(D_2,A_1)&=0.9183-\left[\frac{4}{9}\left(-\frac{1}{4}\log_2\frac{1}{4}-\frac{3}{4}\log_2\frac{3}{4}\right)+0+\frac{3}{9}\left(-\frac{1}{3}\log_2\frac{1}{3}-\frac{2}{3}\log_2\frac{2}{3}\right)\right]\\&=0.2516\end{aligned}$$

$$\Delta I(D_2,A_2)=0.9183-0=0.9183,\Delta I(D_2,A_3)=0.474$$

所以选择A_2作为分类特征，此时划分为的子数据集中的样本都属于同一类，因此决策树建立完成，仅用了两个特征。建立的决策树如下图所示：



思考题

因为采取后剪枝策略，故首先建立决策树。

对于根结点D，首先计算划分前的信息熵：

$$H(D) = -\frac{7}{11}\log\frac{7}{11} - \frac{4}{11}\log\frac{4}{11} = 0.946$$

然后计算各属性对根结点D的信息增益，分别以A_1，A_2，A_3，A_4表示年龄、收入情况、有车和信贷情况 4 个特征，则：

$$\begin{aligned}\text{Gain}(D, A_1) &= H(D) - H(D|A_1)\\ &= 0.946 - \left(\frac{4}{11} * \left(-\frac{3}{4}\log\frac{3}{4} - \frac{1}{4}\log\frac{1}{4}\right) + \frac{4}{11} * 0 + \frac{3}{11} * \left(-\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3}\right)\right)\\ &= 0.401\end{aligned}$$

$$\begin{aligned}\text{Gain}(D, A_2) &= H(D) - H(D|A_2)\\ &= 0.946 - \left(\frac{8}{11} * \left(-\frac{4}{8}\log\frac{4}{8} - \frac{4}{8}\log\frac{4}{8}\right) + \frac{3}{11} * 0\right)\\ &= 0.219\end{aligned}$$

$$\begin{aligned}\text{Gain}(D, A_3) &= H(D) - H(D|A_3)\\ &= 0.946 - \left(\frac{6}{11} * \left(-\frac{4}{6}\log\frac{4}{6} - \frac{2}{6}\log\frac{2}{6}\right) + \frac{5}{11} * 0\right)\\ &= 0.445\end{aligned}$$

$$\begin{aligned}\text{Gain}(D, A_4) &= H(D) - H(D|A_4)\\ &= 0.946 - \left(\frac{3}{11} * 0 + \frac{5}{11} * \left(-\frac{1}{5}\log\frac{1}{5} - \frac{4}{5}\log\frac{4}{5}\right) + \frac{3}{11} * 0\right)\\ &= 0.618\end{aligned}$$

思考题

因为$\mathrm{Gain}(D, A_4)$最大，所以首先选择A_4作为划分属性，这将根结点分为三个子结点，即D_1（一般）、D_2（好）和D_3（非常好）。因为D_1中的样本标签全为“否”，故将D_1的标签置为“否”后停止划分，同理将D_3的标签置为“是”后停止划分。下面考虑对D_2的进一步划分：

$$H(D_2) = -\frac{1}{5}\log\frac{1}{5} - \frac{4}{5}\log\frac{4}{5} = 0.722$$

$$\begin{aligned}\mathrm{Gain}(D_2, A_1) &= H(D_2) - H(D_2|A_1)\\ &= 0.722 - \left(\frac{2}{5} * \left(-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2}\right) + \frac{2}{5} * 0 + \frac{1}{5} * 0\right)\\ &= 0.322\end{aligned}$$

$$\begin{aligned}\mathrm{Gain}(D_2, A_2) &= H(D_2) - H(D_2|A_2)\\ &= 0.722 - \left(\frac{2}{5} * \left(-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2}\right) + \frac{3}{5} * 0\right)\\ &= 0.322\end{aligned}$$

$$\begin{aligned}\mathrm{Gain}(D_2, A_3) &= H(D_2) - H(D_2|A_3)\\ &= 0.722 - \left(\frac{3}{5} * \left(-\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3}\right) + \frac{2}{5} * 0\right)\\ &= 0.171\end{aligned}$$

因为$\mathrm{Gain}(D_2, A_1)$和$\mathrm{Gain}(D_2, A_2)$相等，所以划分属性可以在两者之间任取，不妨选择A_1作为划分属性。A_1将D_2分为三个子结点，分别为D_4（小于 30 岁）、D_5（30 到 60 岁）和D_6（大于 60 岁），其中D_5和D_6的标签可以直接取“是”。下面考虑对D_4的划分：

D_4包含 2 号和 11 号两个样本，它们在A_3上的取值相同，在A_2上的取值不同，故只能选取A_2作为划分属性。得到决策树如图 1 所示。

思考题

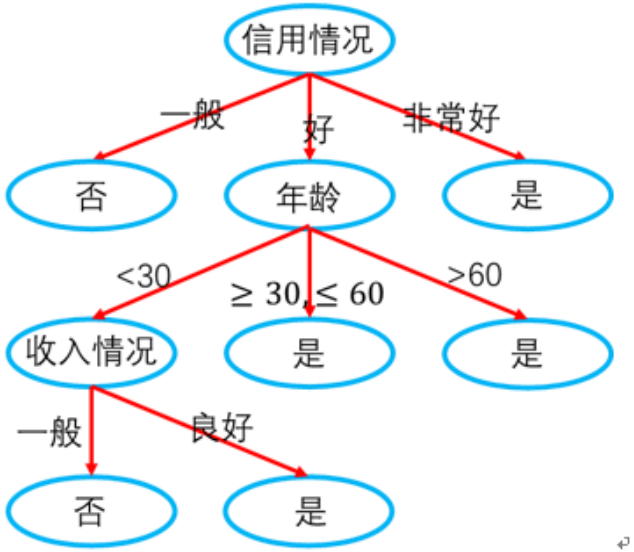


图 1

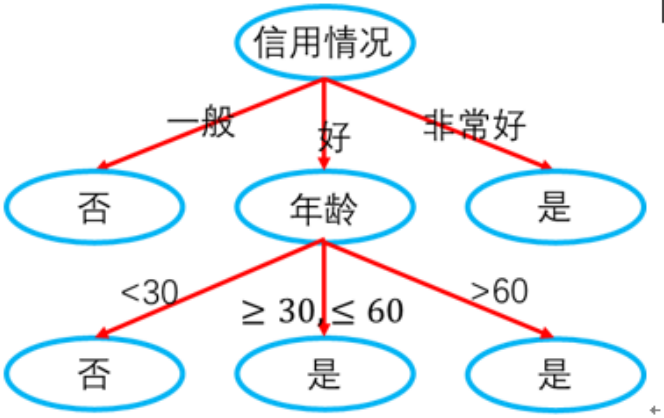


图 2

后剪枝过程：

首先考虑“收入情况”节点，剪枝前验证集分类正确的样本为{5，14}，故剪枝前的验证集准确率为 0.5；剪枝后，收入情况节点包含训练集样本{2，11}，不妨假设其标签为“否”，得到决策树如图 2 所示，经计算，验证集分类正确的样本仍为{5，14}，故剪枝后的验证集准确率为 0.5。由于剪枝前后验证集准确率保持不变，故不做剪枝，剪枝操作停止。

最终的决策树如图 1 所示。