

# 决策树

翟婷婷

扬州大学  
信息工程（人工智能）学院  
zh tt@yzu.edu.cn

2023春

# 课程目标

- 理解决策树的基本原理和学习算法。
- 掌握决策树选择划分属性的3个方法：信息增益、增益率和基尼指数。
- 理解决策树剪枝的目的，以及剪枝的策略。
- 对包含离散属性和连续属性的数据集，能够编程实现决策树的学习和分类算法。

# 引言

- 人们在做决策或行动时，总是根据一些前提条件，做出相应的结论，这种决策方式可以用if-then规则来表示。  
例如：要判断一个西瓜是不是好瓜，人们通常会进行一系列的判断或“子决策”：

先看“它的色泽如何？”

如果是“青绿色”，

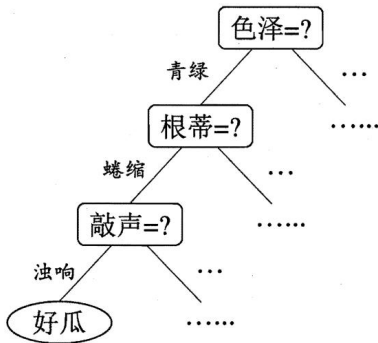
则再看“它的根蒂是什么形态？”

如果是“蜷缩的”，

则再看“它敲起来是什么声音？”

如果是“浊响”，

则得出最终决策：这是个好瓜。



西瓜问题的一棵决策树

# 引言

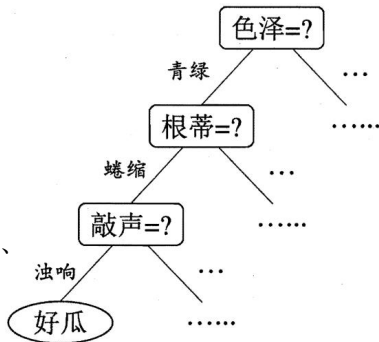
- 决策树可以看作是从训练数据集中学习到的一组if-then的判定规则，将**判定规则组织成树形结构**，基于该树型结构对未知类别的样本进行分类决策。所以，决策树亦称为“判定树”。
- 运用构建好的决策树进行分类，就是执行一系列的if-then的判断，最后得到结论的过程。
- 决策树可以很好地反应人类的决策机制，因此决策树具有良好的可解释性。
- 当提到"决策树"时，有时是指决策树的学习方法，有时是指学习得到的树。

# 决策树的表示

- 一棵决策树包含一个根结点、若干个内部结点和叶结点。叶结点对应决策结果，非叶子节点对应一个属性的“测试”，例如，“色泽=?”。

每个属性测试的结果或是导出最终结论，或是在本次测试结果的限定范围之内，导出进一步的判定问题。

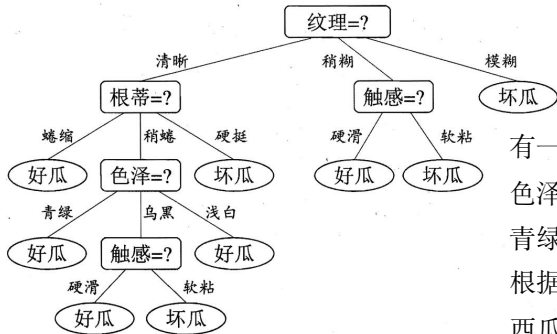
例如,在"根蒂=蜷缩"之后再判断“敲声=?”，则仅考虑青绿色、根蒂蜷缩的瓜的“敲声=?”



西瓜问题的一棵决策树

# 决策树的分类

- 利用决策树对一个样本进行分类，就是对样本所包含的特征按照决策树的属性测试顺序依次进行测试，将样本遍历到叶结点，从而得出分类结论。



有一个西瓜：

色泽 根蒂 敲声 纹理 脐部 触感  
青绿 稍蜷 浊响 清晰 凹陷 硬滑  
根据左侧的决策树，判断这个西瓜是好瓜 or 坏瓜？

从**根结点**到每个**叶子结点**的路径对应一条**分类判定规则**！

# 决策树的构建/学习

- 决策树学习是利用训练数据集构建一个决策树模型，使它不仅能对训练数据较好地分类，也能对未知数据进行很好地预测。
- 能对训练数据进行正确分类的决策树可能有多个，也可能一个也没有。我们要找的是一个与训练数据集的矛盾较小，且具有较好的泛化能力的决策树。
- 决策树构建的思路：
  - ① 生成一个根结点，该结点包含整个样本集，在根结点上选择一个属性进行测试，根据测试的结果生成若干分支结点，分支结点表示按“测试属性”的不同取值对父结点的样本集进行的某种划分，因此每个分支结点也包含一个样本集，其中的样本在父结点的属性测试上结果相同。

# 决策树的构建/学习

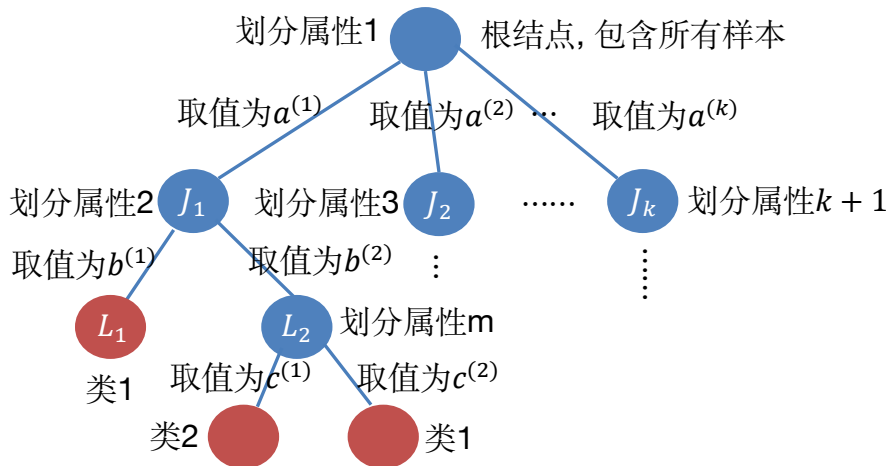
② 依次对每个分支结点处的样本集继续进行划分，直至

- ✱ 结点中仅包含同一类的样本，或者
- ✱ 已没有合适的属性可用于测试，或者
- ✱ 结点中包含的样本集为空时，为止，

此时，无分支的结点标记为叶节点，其所属的类别标记为：该结点包含的样本中样本数量最多的那一类，如果该结点包含的样本集为空集，则标记为其父结点包含的样本中样本数最多的那一类。



# 决策树的构建/学习



结点 $J_k$ 包含在划分属性1上取值为 $a^{(k)}$ 的所有样本。

结点 $L_1$ 包含在划分属性1上取值为 $a^{(1)}$ , 在划分属性2上取值为 $b^{(1)}$ 的所有样本。

# 决策树的构建/学习算法

输入: 训练集  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ ;

属性集  $A = \{a_1, a_2, \dots, a_d\}$ .

过程: 函数 TreeGenerate( $D, A$ )

1: 生成结点 node;

2: if  $D$  中样本全属于同一类别  $C$  then 无需划分, 递归返回

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 无法划分, 递归返回

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

正常划分, 递归构建决策树

输出: 以 node 为根结点的一棵决策树

# 决策树的构建/学习

- 决策树学习的关键是算法第8行：选择最优的划分属性。
- 什么样的属性对样本集进行划分是最优的？

在进行样本集划分时，我们希望每次划分出的样本子集，即每个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”越高越好，这样就能高效地从根结点遍历到叶结点得到分类结果。

- 3种度量结点纯度的指标：
  - ✧ 信息增益
  - ✧ 增益率
  - ✧ 基尼指数

# 决策树的构建/学习—信息增益

## ➤ 信息熵(information entropy):

信息论中，熵用来衡量一个随机变量取值的不确定程度。

设 $X$ 是一个取有限个值的离散随机变量，其概率分布为：

$$P(X = x_i) = p_i, \quad i = 1, \dots, m$$

则随机变量 $X$ 的熵定义为：

$$\text{Ent}(X) = - \sum_{i=1}^m p_i \log p_i$$

式中，若 $p_i = 0$ ，则定义 $0 \log 0 = 0$ 。

式中 $\log$  函数可以2为底，此时熵的单位是比特(bit)，或以e为底，此时单位是纳特(nat)。

# 决策树的构建/学习—信息增益

## ➤ 信息熵的性质:

✿ 熵只依赖于随机变量的分布，与随机变量的取值无关。

✿ 对于一个有 $m$ 个取值的随机变量 $X$ ，可以证明：

当 $p_1 = p_2 \cdots = p_m = \frac{1}{m}$ ， $X$ 的熵取得最大值为 $\log m$ ；

也即，当 $X$ 取每一个值的概率相等时，熵取得最大值；

当存在 $p_i = 1, \forall j \neq i, p_j = 0$ ， $X$ 的熵取得最小值为0。

综上： $0 \leq \text{Ent}(X) \leq \log m$

✿ 熵越大，表明随机变量取值的不确定性越大。

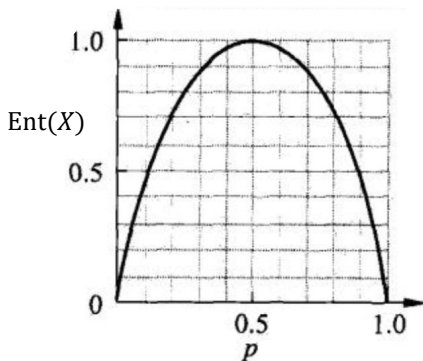
# 决策树的构建/学习—信息增益

➤ 例子：考虑一个伯努利分布的随机变量 $X$ ：

$$P(X = 1) = p, \quad P(X = 0) = 1 - p$$

则

$$\text{Ent}(X) = -p \log_2 p - (1 - p) \log_2 (1 - p)$$



$p=0.5$ 时，熵最大；  
 $p=0$ 或 $p=1$ ， $X$ 的取值完全没有不确定性，熵为0。

# 决策树的构建/学习—信息增益

- 条件熵 $\text{Ent}(X|Y)$ : 衡量在随机变量 $Y$ 的取值已知的条件下, 随机变量 $X$ 取值的平均不确定性, 定义为:

$$\text{Ent}(X|Y) = \sum_{i=1}^m P(Y = y_i) \text{Ent}(X|Y = y_i)$$

- $\text{Ent}(X) - \text{Ent}(X|Y)$ :

表示在得知 $Y$ 后, 随机变量 $X$ 的不确定性减少的程度, 也是确定性(信息量)增加的程度, 称为**信息增益**。

- 在信息论中,  $\text{Ent}(X) - \text{Ent}(X|Y)$ 称为 $X$ 与 $Y$ 的互信息 (mutual information), 且

$$\text{Ent}(X) - \text{Ent}(X|Y) = \text{Ent}(Y) - \text{Ent}(Y|X)$$

# 决策树的构建/学习—信息增益

- 在决策树学习问题中，将样本的类别class看作是一个随机变量，对于一个c类的分类问题，类别变量的分布为：

$$P(\text{class} = i) = \frac{\text{训练集}\mathcal{D}\text{中第}i\text{类样本的数目}}{\text{训练集}\mathcal{D}\text{中所有样本的总数}}, i = 1, 2, \dots, c$$

记 $p_i = P(\text{class} = i)$ ，则类别变量的熵为

$$\text{Ent}(\text{class}) = - \sum_{i=1}^c p_i \log p_i$$

其中 $p_i$ 是训练数据集 $\mathcal{D}$ 第 $i$ 类样本所占的比例。将类别变量的熵称为训练集 $\mathcal{D}$ 的熵，记为 $\text{Ent}(\mathcal{D})$ 。

- $\text{Ent}(\mathcal{D})$ 可作为样本集 $\mathcal{D}$ 的不纯度的度量， $\text{Ent}(\mathcal{D})$ 越小，表明训练集 $\mathcal{D}$ 的纯度越高， $\text{Ent}(\mathcal{D})$ 越大，纯度越低。



# 决策树的构建/学习—信息增益

- 假设选择一个特征 $a$ 对样本集 $\mathcal{D}$ 进行划分，特征 $a$ 有 $k$ 个不同的取值 $\{a^{(1)}, a^{(2)} \dots a^{(k)}\}$ ，根据 $a$ 的取值将 $\mathcal{D}$ 划分为 $k$ 个子集 $\mathcal{D}_1, \mathcal{D}_2 \dots \mathcal{D}_k$ ，其中 $\mathcal{D}_i$ 中的样本在特征 $a$ 上取值为 $a^{(i)}$ ，用 $|\mathcal{D}_i|$ 表示 $\mathcal{D}_i$ 中的样本数。

选择特征 $a$ 对样本集 $\mathcal{D}$ 进行划分所产生的**信息增益**为：

$$\begin{aligned}\text{gain}(\mathcal{D}, a) &= \text{Ent}(\mathcal{D}) - \text{Ent}(\mathcal{D}|a) \\ \text{Ent}(\mathcal{D}|a) &= \sum_{i=1}^k P(a = a^{(i)}) \text{Ent}(\mathcal{D}|a = a^{(i)}) \\ &= \sum_{i=1}^k \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \text{Ent}(\mathcal{D}_i)\end{aligned}$$

- 决策树学习中的信息增益等价于训练数据集中**类与特征之间的互信息**。

# 决策树的构建/学习—信息增益

- 使用信息增益来选择划分的属性：在决策树算法第8行，选择属性

$$a_* = \arg \max_{a \in A} gain(\mathcal{D}, a)$$

使用 $a_*$ 对样本集 $\mathcal{D}$ 进行划分所获的“纯度提升”最大。

著名的 ID3 决策树学习算法使用的就是信息增益的准则。

- 举例：使用下图所示的西瓜数据集来构建一棵决策树，用以预测一个没切开的西瓜是不是好瓜。

# 决策树的构建/学习—信息增益

西瓜数据集1.0

| 编号 | 色泽 | 根蒂 | 敲声 | 纹理 | 脐部 | 触感 | 好瓜 |
|----|----|----|----|----|----|----|----|
| 1  | 青绿 | 蜷缩 | 浊响 | 清晰 | 凹陷 | 硬滑 | 是  |
| 2  | 乌黑 | 蜷缩 | 沉闷 | 清晰 | 凹陷 | 硬滑 | 是  |
| 3  | 乌黑 | 蜷缩 | 浊响 | 清晰 | 凹陷 | 硬滑 | 是  |
| 4  | 青绿 | 蜷缩 | 沉闷 | 清晰 | 凹陷 | 硬滑 | 是  |
| 5  | 浅白 | 蜷缩 | 浊响 | 清晰 | 凹陷 | 硬滑 | 是  |
| 6  | 青绿 | 稍蜷 | 浊响 | 清晰 | 稍凹 | 软粘 | 是  |
| 7  | 乌黑 | 稍蜷 | 浊响 | 稍糊 | 稍凹 | 软粘 | 是  |
| 8  | 乌黑 | 稍蜷 | 浊响 | 清晰 | 稍凹 | 硬滑 | 是  |
| 9  | 乌黑 | 稍蜷 | 沉闷 | 稍糊 | 稍凹 | 硬滑 | 否  |
| 10 | 青绿 | 硬挺 | 清脆 | 清晰 | 平坦 | 软粘 | 否  |
| 11 | 浅白 | 硬挺 | 清脆 | 模糊 | 平坦 | 硬滑 | 否  |
| 12 | 浅白 | 蜷缩 | 浊响 | 模糊 | 平坦 | 软粘 | 否  |
| 13 | 青绿 | 稍蜷 | 浊响 | 稍糊 | 凹陷 | 硬滑 | 否  |
| 14 | 浅白 | 稍蜷 | 沉闷 | 稍糊 | 凹陷 | 硬滑 | 否  |
| 15 | 乌黑 | 稍蜷 | 浊响 | 清晰 | 稍凹 | 软粘 | 否  |
| 16 | 浅白 | 蜷缩 | 浊响 | 模糊 | 平坦 | 硬滑 | 否  |
| 17 | 青绿 | 蜷缩 | 沉闷 | 稍糊 | 稍凹 | 硬滑 | 否  |

# 决策树的构建/学习—信息增益

- 在决策树学习开始时，根结点包含训练集 $\mathcal{D}$ 中所有样本，首先确定在根结点上选择什么属性进行划分？

数据集总共有17个训练样本，其中

正例样本(好瓜)所占比例为 $p_1 = \frac{8}{17}$

负例样本(坏瓜)所占比例为 $p_2 = \frac{9}{17}$

根结点的信息熵为：

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

当前属性集合为{色泽, 根蒂, 敲声, 纹理, 脐部, 触感}, 接下来，需要依次计算使用每个属性对 $\mathcal{D}$ 进行划分产生的信息增益。

# 决策树的构建/学习—信息增益

以属性“色泽”为例，其可能的取值为{青绿, 乌黑, 浅白}，按照该属性对 $\mathcal{D}$ 进行划分，得到3个子集 $\mathcal{D}_1$ 、 $\mathcal{D}_2$ 、 $\mathcal{D}_3$ ：

- ✱  $\mathcal{D}_1$ 包含色泽=青绿的所有样本，样本编号为{1, 4, 6, 10, 13, 17}，其中正例占  $p_1 = \frac{3}{6}$ ，负例占  $p_2 = \frac{3}{6}$ ：

$$\text{Ent}(\mathcal{D}_1) = -\left(\frac{3}{6} \log_2 \frac{3}{6} + \frac{3}{6} \log_2 \frac{3}{6}\right) = 1.000$$

- ✱  $\mathcal{D}_2$ 包含色泽=乌黑的所有样本，样本编号为{2, 3, 7, 8, 9, 15}，其中正、负例分别占  $p_1 = \frac{4}{6}$ ， $p_2 = \frac{2}{6}$ ：

$$\text{Ent}(\mathcal{D}_2) = -\left(\frac{4}{6} \log_2 \frac{4}{6} + \frac{2}{6} \log_2 \frac{2}{6}\right) = 0.918$$

- ✱  $\mathcal{D}_3$ 包含色泽=浅白的所有样本，样本编号为{5, 11, 12, 14, 16}，其中正、负例分别占  $p_1 = \frac{1}{5}$ ， $p_2 = \frac{4}{5}$ ：

$$\text{Ent}(\mathcal{D}_3) = -\left(\frac{1}{5} \log_2 \frac{1}{5} + \frac{4}{5} \log_2 \frac{4}{5}\right) = 0.722$$

# 决策树的构建/学习—信息增益

➤ 用“色泽”划分产生的信息增益为：

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

类似地，计算出其它属性能获得的信息增益：

$$\text{gain}(\mathcal{D}, \text{根蒂}) = 0.143; \quad \text{gain}(\mathcal{D}, \text{敲声}) = 0.141;$$

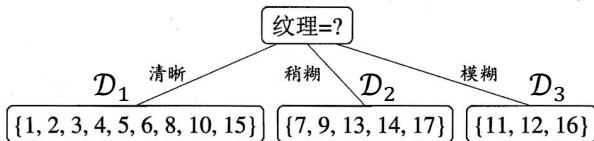
$$\text{gain}(\mathcal{D}, \text{纹理}) = 0.381; \quad \text{gain}(\mathcal{D}, \text{脐部}) = 0.289;$$

$$\text{gain}(\mathcal{D}, \text{触感}) = 0.006.$$

所以，在根结点处，选择“纹理”为最优的划分属性。

# 决策树的构建/学习—信息增益

- 基于"纹理"对根结点进行划分的结果如图所示，各分支结点包含的样本子集显示在结点中：



接下来，决策树学习算法将依次对每个分支结点做进一步的划分。

以图中第一个分支结点("纹理=清晰")为例，当前可用的属性集合为{色泽, 根蒂, 敲声, 脐部, 触感}，需要依次计算每个可用的属性对 $D_1$ 进行划分所产生的信息增益。

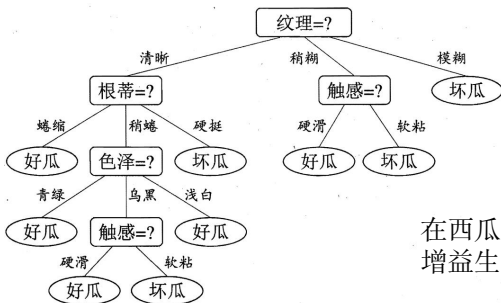
# 决策树的构建/学习—信息增益

$\text{gain}(\mathcal{D}_1, \text{色泽}) = 0.043$ ;  $\text{gain}(\mathcal{D}_1, \text{根蒂}) = 0.458$ ;

$\text{gain}(\mathcal{D}_1, \text{敲声}) = 0.331$ ;  $\text{gain}(\mathcal{D}_1, \text{脐部}) = 0.458$ ;

$\text{gain}(\mathcal{D}_1, \text{触感}) = 0.458$

"根蒂"、"脐部"、"触感" 3个属性均取得了最大的信息增益，可任选其中之一作为划分属性。类似地，对每个分支结点进行上述操作，最终得到的决策树如图：



在西瓜数据集上基于信息增益生成的决策树



# 决策树的构建/学习—信息增益

- 在上述的决策树学习中，如果把“编号”也作为一个候选的划分属性，则“编号”把训练集 $\mathcal{D}$ 分为17个集合，每个集合中只有一个样本，每个集合 $\mathcal{D}_i$ 的信息熵为：

$$\text{Ent}(\mathcal{D}_i) = -1 \log_2 1 = 0, \quad i = 1, 2 \dots 17$$

所以，属性“编号”的信息增益为：

$$\text{gain}(\mathcal{D}, \text{编号}) = \text{Ent}(\mathcal{D}) - \sum_{i=1}^{17} \frac{1}{17} \text{Ent}(\mathcal{D}_i) = 0.998$$

远大于其它候选属性的信息增益，但是使用“编号”进行属性划分显然没有实际意义，产生的决策树无泛化能力。

- 这体现信息增益准则的缺点：偏爱取值数目较多的属性。

# 决策树的构建/学习—信息增益率

- 属性 $a$ 对样本集 $\mathcal{D}$ 进行划分产生的**信息增益率**为:

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

$$IV(a) = - \sum_{i=1}^k \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \log_2 \frac{|\mathcal{D}_i|}{|\mathcal{D}|}$$

属性 $a$ 的熵

称为属性 $a$ 的“固有值”。属性 $a$ 取值越多,  $IV(a)$ 的值越大。

例如在西瓜数据集中,  $IV(\text{触感})=0.874(k=2)$ ,

$IV(\text{色泽})=1.580(k=3)$ ,  $IV(\text{编号})=4.088(k=17)$ 。

- 信息增益率准则对取值数目较少的属性有所偏好。
- 著名的C4.5算法综合考虑了信息增益和增益率的准则:  
先从候选划分属性中找出**信息增益**高于平均水平的属性集, 再从中选择**增益率**最高的。

# 决策树的构建/学习—基尼指数

- 给定一个样本集 $\mathcal{D}$ ，记 $p_i$ 是 $\mathcal{D}$ 中第 $i$ 类样本所占的比例， $i = 1, 2 \dots c$ ；从 $\mathcal{D}$ 中随机抽取两个样本，则两个样本的类别不相同的概率为：

$$\sum_{i=1}^c p_i (1 - p_i) = 1 - \sum_{i=1}^c p_i^2$$

- 样本集 $\mathcal{D}$ 的**基尼值**定义为：

$$Gini(\mathcal{D}) = 1 - \sum_{i=1}^c p_i^2$$

$Gini(\mathcal{D})$ 反映了样本集 $\mathcal{D}$ 的不纯度。 $Gini(\mathcal{D})$ 越大，样本集 $\mathcal{D}$ 的不纯度越大，即纯度越低。

# 决策树的构建/学习—基尼指数

- 属性 $a$ 关于样本集 $\mathcal{D}$ 的**基尼指数**定义为:

$$Gini\_index(\mathcal{D}, a) = \sum_{i=1}^k \frac{|\mathcal{D}_i|}{|\mathcal{D}|} Gini(\mathcal{D}_i)$$

- 基于基尼指数选择划分属性:

在候选属性集合 $A$ 中, 选择使得划分后的**基尼指数最小**的那个属性作为最优划分属性

$$a_* = \arg \min_{a \in A} Gini\_index(\mathcal{D}, a)$$

- CART 决策树学习算法使用基尼指数来选择划分属性。

# 决策树的构建/学习—剪枝处理

- 在决策树学习中，为了尽可能正确分类训练样本，结点划分过程将不断重复进行，有时会造成决策树分支过多，以致于把训练集自身的一些特点当作所有数据都具有的一般性质而导致**过拟合**。

过拟合现象无法彻底避免，只能“缓解”。

- **剪枝**：为了降低过拟合的风险，可主动剪掉决策树的一些分支。剪枝策略包括：预剪枝和后剪枝。
- **预剪枝**是指在决策树构建过程中，对每个结点在划分前先进行估计，若当前结点的划分不能带来决策树泛化性能提升，则停止划分，并将当前结点标记为叶结点。

# 决策树的构建/学习—剪枝处理

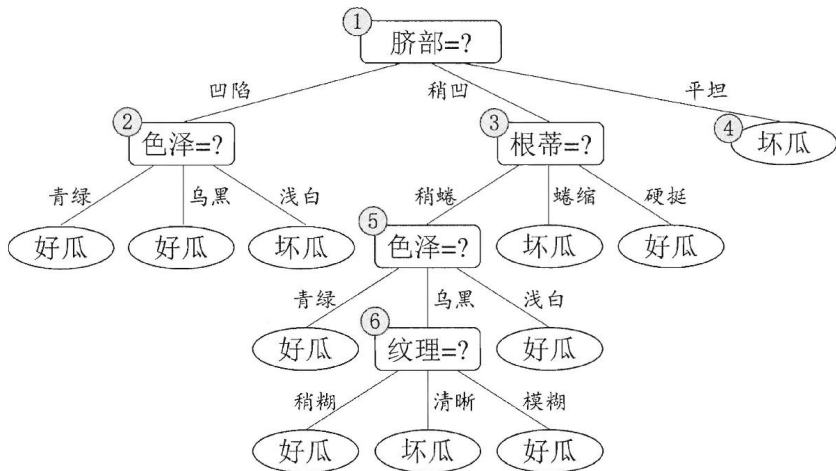
- **后剪枝**则是先从训练集生成一棵完整的决策树，然后自底向上地对**非叶结点**进行考察，若将该结点对应的子树替换为叶结点能带来决策树泛化性能提升，则将该子树替换为叶结点。
- 如何判断决策树泛化性能是否提升呢？  
经常将整个数据集划分为两个**互不相交**的集合：训练集和验证集，训练集用于学习决策树，验证集用于进行泛化性能的评估。

# 决策树的构建/学习—剪枝处理

西瓜数据集划分出的训练集(双线上部)与验证集(双线下部)

| 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  |
| 训练集 | 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  |
|     |    |    |    |    |    |    |    |    |
|     |    |    |    |    |    |    |    |    |

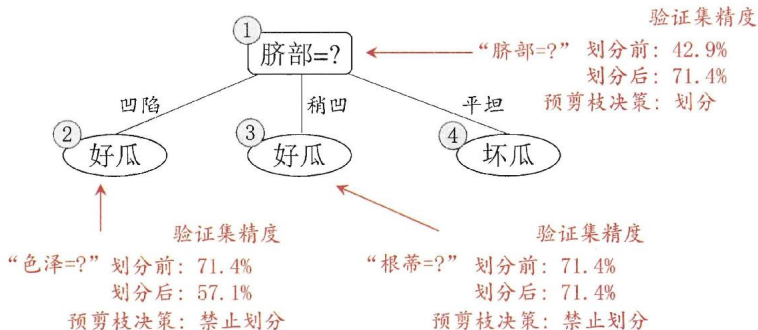
# 决策树的构建/学习—剪枝处理



基于上表所示的训练集生成的未剪枝决策树



# 决策树的构建/学习—预剪枝

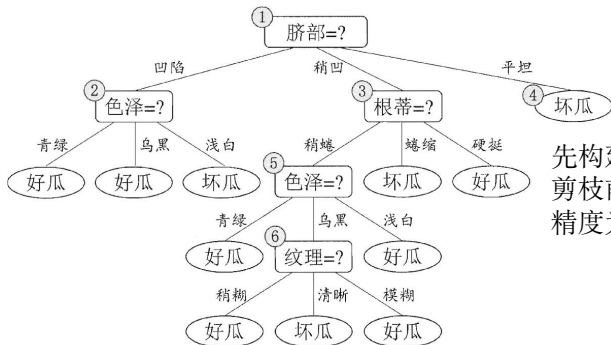


基于上表所示的训练集生成的**预剪枝决策树**

预剪枝使得决策树的很多分支都没有“展开”。

- ✿ 优点: 降低过拟合风险, 减小训练和测试时间开销。
- ✿ 缺点: 生成的树比较简单, 带来欠拟合风险。

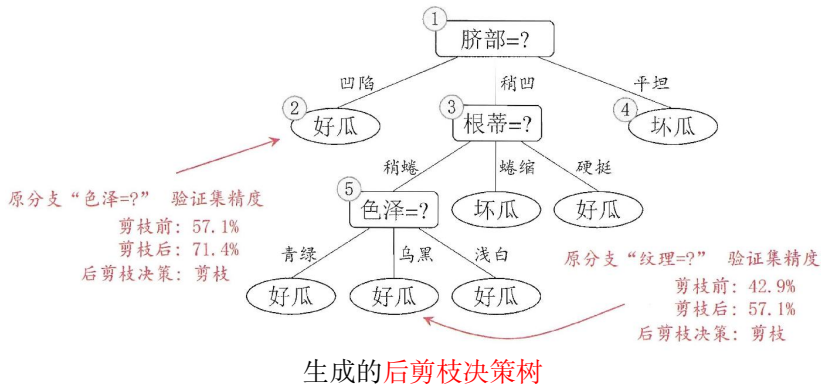
# 决策树的构建/学习—后剪枝



先构建一颗完整的决策树；  
剪枝前，决策树的验证集  
精度为42.9%。

- 自底向上考察每个非叶子结点，顺序为：⑥⑤②③①
- 首先考察结点⑥，若将其领衔的分支剪除，相当于把⑥替换为叶结点，并将其类别标记为⑥所包含的样本中样本数量最多的那个类，此时决策树的验证集精度提高至57.1%，于是后剪枝策略决定剪枝。依次处理其余非叶子结点。

# 决策树的构建/学习—后剪枝



后剪枝策略保留了更多的分支。

- ✿ 优点：欠拟合风险很小，泛化能力优于预剪枝决策树。
- ✿ 缺点：训练开销大，需要先得到完整的决策树，再剪枝。

# 连续属性的处理

- 前面我们学习了基于离散属性来构建决策树，现实的学习任务中常遇到连续属性。
- 与离散属性不同，连续属性的取值为实值，其可能的取值个数是无限的，因此，不能直接根据连续属性的取值来对结点进行划分。
- 解决办法：连续属性离散化技术

给定样本集 $\mathcal{D}$ 和连续属性 $a$ ，假设属性 $a$ 在 $\mathcal{D}$ 中的样本上有 $n$ 个不同的取值，对这些值按照从小到大的顺序进行排序，记为

$$\{a^{(1)}, a^{(2)}, \dots, a^{(i)} \mid a^{(i+1)}, \dots, a^{(n)}\}$$

寻找一个划分点 $p$ 将 $\mathcal{D}$ 划分为两个子集 $\mathcal{D}_L$ 和 $\mathcal{D}_R$ ，使得

# 连续属性的处理

$\mathcal{D}_L$  包含在属性  $a$  上取值  $\leq p$  的样本集合,

$\mathcal{D}_R$  包含在属性  $a$  上取值  $> p$  的样本集合。

计算使用属性  $a$  的划分点  $p$  对  $\mathcal{D}$  划分所产生的信息增益:

$$gain(\mathcal{D}, a, p) = Ent(\mathcal{D}) - \frac{|\mathcal{D}_L|}{|\mathcal{D}|} Ent(\mathcal{D}_L) - \frac{|\mathcal{D}_R|}{|\mathcal{D}|} Ent(\mathcal{D}_R)$$

如何寻找属性  $a$  的最优的划分点  $p$ ? 候选划分点集  $P_a$ :

$$P_a = \left\{ \frac{a^{(i)} + a^{(i+1)}}{2} \mid 1 \leq i \leq n - 1 \right\}$$

连续属性  $a$  的最优划分点  $p^*$  为

$$p^* = \arg \max_{p \in P_a} gain(\mathcal{D}, a, p)$$

连续属性  $a$  对  $\mathcal{D}$  划分的信息增益为  $gain(\mathcal{D}, a, p^*)$ 。

# 连续属性的处理—举例

包含连续属性的西瓜数据集2.0

| 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  |

# 连续属性的处理—举例

➤ 属性“密度”的候选划分点集为:

$P_{\text{密度}}$

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

计算得到最优的划分点为0.381，其对应的信息增益为0.262，因此

$$\text{gain}(\mathcal{D}, \text{“密度”}) = 0.262$$

类似地，可以得到，属性“含糖率”的最优划分点为0.126，其对应的信息增益为0.349，因此

$$\text{gain}(\mathcal{D}, \text{“含糖率”}) = 0.349$$

# 连续属性的处理—举例

- 每个属性对 $\mathcal{D}$ 划分所产生的信息增益为：

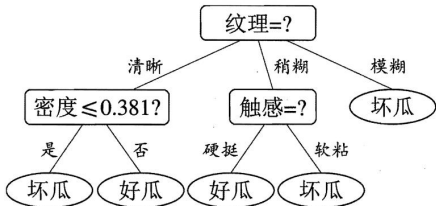
$\text{gain}(\mathcal{D}, \text{色泽}) = 0.109$ ;  $\text{gain}(\mathcal{D}, \text{根蒂}) = 0.143$ ;

$\text{gain}(\mathcal{D}, \text{敲声}) = 0.141$ ;  **$\text{gain}(\mathcal{D}, \text{纹理}) = 0.381$** ;

$\text{gain}(\mathcal{D}, \text{脐部}) = 0.289$ ;  $\text{gain}(\mathcal{D}, \text{触感}) = 0.006$ ;

$\text{gain}(\mathcal{D}, \text{密度}) = 0.262$ ;  $\text{gain}(\mathcal{D}, \text{含糖率}) = 0.349$ .

- "纹理"被选作根结点的划分属性，此后结点划分过程递归进行，最终生成的决策树如图：



与离散属性不同，若当前结点的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。

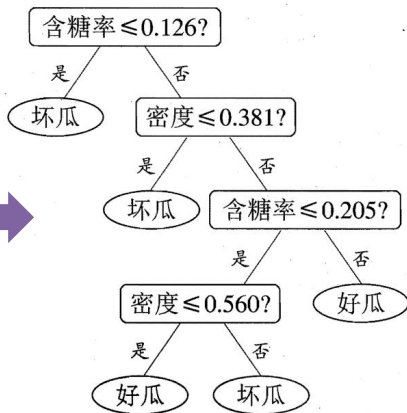


# 决策树的决策边界

西瓜数据集3.0

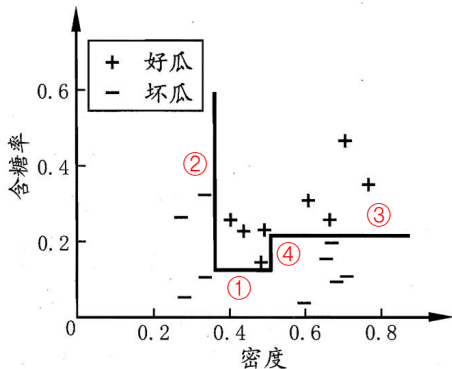
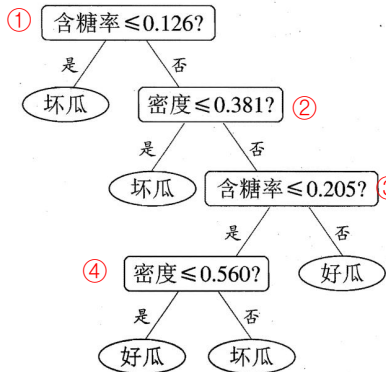
| 编号 | 密度    | 含糖率   | 好瓜 |
|----|-------|-------|----|
| 1  | 0.697 | 0.460 | 是  |
| 2  | 0.774 | 0.376 | 是  |
| 3  | 0.634 | 0.264 | 是  |
| 4  | 0.608 | 0.318 | 是  |
| 5  | 0.556 | 0.215 | 是  |
| 6  | 0.403 | 0.237 | 是  |
| 7  | 0.481 | 0.149 | 是  |
| 8  | 0.437 | 0.211 | 是  |
| 9  | 0.666 | 0.091 | 否  |
| 10 | 0.243 | 0.267 | 否  |
| 11 | 0.245 | 0.057 | 否  |
| 12 | 0.343 | 0.099 | 否  |
| 13 | 0.639 | 0.161 | 否  |
| 14 | 0.657 | 0.198 | 否  |
| 15 | 0.360 | 0.370 | 否  |
| 16 | 0.593 | 0.042 | 否  |
| 17 | 0.719 | 0.103 | 否  |

得到的决策树



# 决策树的决策边界

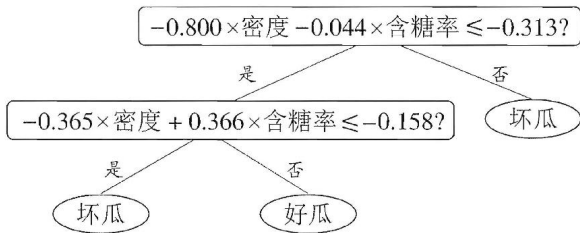
决策树对应的分类决策边界



决策树的分类决策边界是分段线性的，且每段边界都与某个坐标轴平行，即它的决策边界由若干个与坐标轴平行的分段组成。决策树本质上是非线性分类器。

# 决策树的扩展学习—课后

- 在样本的某些属性值缺失时，如何学习决策树？
- 多变量决策树：在对结点进行划分时，不是只对单个属性进行测试，而是对所有属性的线性组合进行测试。



西瓜数据集3.0上生成的多变量决策树

# 小结

- 决策树是基于一系列的属性测试对样本进行分类的树形结构，决策树可以转换成一个if-then规则的集合。
- 决策树的学习旨在构建一棵与训练样本集的矛盾较小且具有良好泛化能力的树。
- 决策树学习算法的关键是如何选择划分结点的属性，常用的有3种结点纯度的度量方法：

(1) 使用属性 $a$ 对样本集 $\mathcal{D}$ 进行划分产生的信息增益：

$$\text{gain}(\mathcal{D}, a) = \text{Ent}(\mathcal{D}) - \text{Ent}(\mathcal{D}|a)$$

$$\text{Ent}(\mathcal{D}|a) = \sum_{i=1}^k \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \text{Ent}(\mathcal{D}_i)$$

重点

其中 $\text{Ent}(\mathcal{D})$ 是 $\mathcal{D}$ 的熵， $\mathcal{D}_i$ 是 $\mathcal{D}$ 中样本在属性 $a$ 取值为 $a^{(i)}$ 的样本子集。

# 小结

(2) 属性 $a$ 对样本集 $\mathcal{D}$ 进行划分产生的信息增益率为:

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

$$IV(a) = - \sum_{i=1}^k \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \log_2 \frac{|\mathcal{D}_i|}{|\mathcal{D}|}$$

重点

(3) 属性 $a$ 对样本集 $\mathcal{D}$ 划分的基尼指数:


$$\text{Gini\_index}(\mathcal{D}, a) = \sum_{i=1}^k \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \text{Gini}(\mathcal{D}_i)$$

$$\text{Gini}(\mathcal{D}) = 1 - \sum_{i=1}^c p_i^2$$

重点

其中,  $p_i$ 是 $\mathcal{D}$ 中第 $i$ 类样本所占的比例。

# 小结

- 决策树的构建过程是递归地选择某个属性对样本集进行划分，直到划分出的样本子集都属于同一类，或不能继续划分为止。属性选择的依据是最大化信息增益/增益率，或者最小化基尼指数。
- 为了降低过拟合的风险，往往需要对从训练集学习到的完整的决策树进行剪枝处理。剪枝策略包括：预剪枝和后剪枝。
- 对离散属性和连续属性，都要会编程实现决策树的学习算法，并会进行分类。