

# 机器学习导论

## – Introduction to Machine Learning

### 第四章：决策树

(Chapter 4: Decision Tree)

华中科技大学电信学院

王邦 博士, 教授博导

wangbang@hust.edu.cn







- ① 决策树 (Decision Tree)
- ② 划分选择 (Partitioning)
  - 信息增益 (Information Gain)
  - 增益率 (Gain Ratio)
  - 基尼指数 (Gini Index)
- ③ 决策树剪枝 (Pruning the Tree)
  - 预剪枝 (Pre-Pruning)
  - 后剪枝 (Post-Pruning)
- ④ 连续值与缺失值 (Continuous Values & Missing Values.)
  - 连续值处理 (Continuous values)
  - 缺失值处理 (Missing values)
- ⑤ 多变量决策树 (Multivariate Decision Tree)
- ⑥ 小结 Summary

- 1 决策树 (Decision Tree)
- 2 划分选择 (Partitioning)
- 3 决策树剪枝 (Pruning the Tree)
- 4 连续值与缺失值 (Continuous Values & Missing Values.)
- 5 多变量决策树 (Multivariate Decision Tree)
- 6 小结 Summary

**APP 推荐** 如图所示：已知一些用户样本，每个样本的属性包括性别和年龄，以及其最常使用的 APP 类型。问：对一个新的年龄为 30 的女性用户，在这三种 APP 中，最先应推荐哪款。

我们可将该问题表示为一个机器学习的分类问题 (**classification**)。通过学习，建立一个分类模型，该模型可以很好地拟合训练数据中的属性与类别之间的联系，并能对新数据样本提供分类依据。

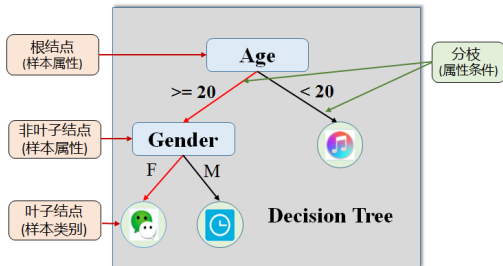
**训练集 (training set)**: 包含  $m$  个样本的训练集  $D$  (本例  $m = 6$ )， $(\mathbf{x}_j, y_j) \in D$  表示  $D$  中第  $j$  个样本及其标签。样本  $\mathbf{x}$  可以由一个  $d$  维特征向量  $(x_1, \dots, x_d)$  来描述 (本例中每个样本有两个属性“性别”和“年龄”(或称之为特征)， $d = 2$ ) 和一个 1 维的类别标签  $y_j$ 。如本例  $\mathbf{x}_1 = (F, 15)$ ，对应的样本标签为音乐。

Gender	Age	App
F	15	
F	25	
M	32	
F	40	
M	12	
M	14	
<b>F</b>	<b>30</b>	<b>?</b>

**决策树**：是一种树形分类模型，利用若干个属性变量进行逐步决策，最后输出分类结果。训练时，首先对数据进行处理，然后利用归纳算法生成可读的规则，并构建决策树。分类时，使用决策树对新数据的各个属性逐一进行决策判断，直至得到分类结果。

# 决策树的基本概念

决策树的组成包括：**决策结点**（样本属性，非叶子结点）、**叶子结点**（类别标签）以及**属性分支**（属性测试）；决策树中最上面的结点称为**根结点**：是整个决策树的开始。每个分支是一个新的决策结点，或者是树的叶子。每个决策结点代表一个样本属性，每个叶结点代表一种可能的分类结果。



**决策（分类）过程：**从根结点出发，沿着决策树从上到下的遍历；每个非叶子结点相当于一个 **IF 条件语句**，因此在每个非叶子结点都有一个属性测试。测试结果对应不同的分枝，最后抵达一个叶子结点，从而得到分类结果。

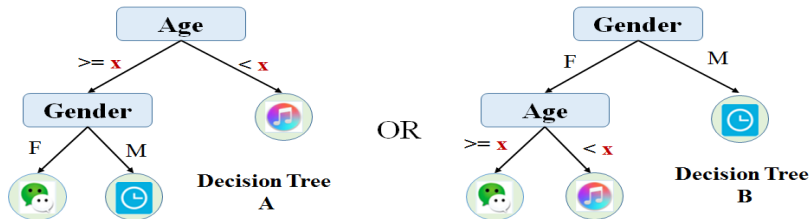
如果利用上图所示的决策树对新样本（年龄为 30 的女性用户）进行分类：首先从根结点“年龄”出发，按照判决条件  $30 \geq 20$  沿着左分枝向下抵达属性结点“性别”，接着按照判决条件“女性”，得到推荐结果应为推荐“微信”APP。

# 决策树的构建 (1)



在决策树构建过程中，按照何种顺序进行决策树构造？按照何种依据选择属性结点？对于连续属性（本例中“年龄”），如何进行分段处理？对于缺失属性，又应该如何处理？如何判断所构建的决策树性能？对这些问题的不同回答，可能导致构建出不同的决策树。

在引例问题中，按照属性选择顺序不同，可以构造如下两颗不同的决策树。



**决策树性能：** 与训练数据矛盾较小，同时具有很好的泛化能力。

# 决策树的构建 (2)

## 理想的决策树形态 (NP 难问题)

- 叶子结点数最少;
- 叶子结点深度最小;
- 叶子结点数最少且叶子结点深度最小。

## 基本流程 (自上而下的贪心算法, 局部最优)

- 初始: 从根结点开始, 包括所有的训练数据。
- 迭代: 每一步中**选择最优属性**对训练数据进行**递归划分**
- 停止: 一个结点上的数据都是属于同一个类别; 或者没有属性可以再用于对数据进行分割。

如何从  $A$  中选择最优划分属性  $a^*$ ?

**优化目标:** 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”(purity) 越来越高,

**最大化纯度提升**  $a^* = \arg \max_{a \in A} \text{purity}(\text{划分后}) - \text{purity}(\text{划分前})$ .

# 决策树的构建 (3)

基本流程：自上而下 (from top to bottom)，分而治之 (divide-and-conquer)。

## 决策树构建：基本流程

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

属性集  $A = a^1, a^2, \dots, a^d$ .

输出：以 node 为根结点的一棵决策树

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  $C$  (情形 (1)) **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 (情形 (2)) **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 (情形 (3)) **then**
- 12:         将分支结点标记为叶结点，其类别标记为  $D$  中样本数最多的类; **return**
- 13:     **else**
- 14:         以  $TreeGenerate(D_v, A \setminus \{a^*\})$  为分支结点 (递归)
- 15:     **end if**
- 16: **end for**

注意决策树的构建是一个**递归过程**，有三种情形导致递归返回

**情形 (1)**：当前节点包含的样本全属于同一类别，无需划分；

将当前节点作为叶子节点标记为该类别；

**情形 (2)**：当前属性集为空，或者所有样本在所有属性上取值相同，无法划分；

将当前节点作为叶子节点标记为所含样本最多的类别；

**情形 (3)**：当前节点包含的样本集合为空，不能划分；

将当前节点作为叶子节点标记为父节点的类别

**关键：**每一步中最优属性的选择。



- 1 决策树 (Decision Tree)
- 2 划分选择 (Partitioning)
- 3 决策树剪枝 (Pruning the Tree)
- 4 连续值与缺失值 (Continuous Values & Missing Values.)
- 5 多变量决策树 (Multivariate Decision Tree)
- 6 小结 Summary

## 信息熵 (Information Entropy)

- 熵是描述事物无序性的参数，熵越大则无序性越强。在信息领域定义为“熵越大，不确定性越大”。
- 设  $X$  是一个离散随机变量，概率分布为： $P(X = x_k) = p_k, k = 1, 2, \dots, n$ ，则该随机变量的熵定义为：

$$Ent(X) = - \sum_{k=1}^n p_k \log_2 p_k$$

- 注意：当  $p_k = 0$  时，令  $p_k \log_2 p_k = 0$ 。

我们可以利用熵来**度量样本集合纯度**（衡量集合中样本的多样性，纯度越高，多样性越低）。样本集合  $D$  中第  $k$  类样本所占比例为  $p_k (k = 1, 2, \dots, |\gamma|)$ ，**定义  $D$  的信息熵：**

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

$Ent(D)$  越小，则集合  $D$  的纯度越高。

当集合中所有类别样本数目相同 ( $p_k = \frac{1}{|\gamma|}$ ) 时， $Ent(D)$  最大，为  $\log_2 |\gamma|$ 。

当集合所有样本为同一类别时 ( $p_i = 1, p_k = 0, k \neq i$ )， $Ent(D)$  最小，为 0。

**信息增益 (Information Gain):** 通过信息熵, 我们可以定义信息增益  $Gain(D, a)$  来度量选用属性  $a$  划分集合  $D$  后所获得的纯度提升。

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

**纯度提升含义:** 选择含  $V$  个取值的属性  $a \in \{a^1, a^2, \dots, a^V\}$  划分集合  $D$  后, 会得到  $V$  个子集  $\mathbf{D}_V = \{D_v | v = 1, 2, \dots, V\}$ , 集合  $D$  的纯度 (信息熵  $Ent(D)$ ) 与所有划分子集  $D_v \in \mathbf{D}_V$  的纯度 (信息熵  $Ent(D_v)$  的加权平均) 相减。

**ID3 决策树:** 对训练数据集 (或子集)  $D$ , 计算其每个属性的信息增益, 并比较它们的大小, **选择信息增益最大的属性**, 即根据信息增益选择属性, 并依此构建决策树。

$$a^* = \arg \max_{a \in A} Gain(D, a)$$

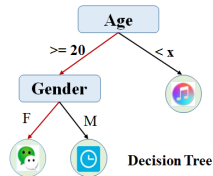
# 信息增益 (3)

引例 APP 推荐:  $D$  的熵为:  $Ent(D) = -\sum_{k=1}^3 p_k \log_2 p_k = -(\frac{3}{6} \log_2 \frac{3}{6} + \frac{1}{6} \log_2 \frac{1}{6} + \frac{2}{6} \log_2 \frac{2}{6}) = 1.46$ 。



Gender	Age	App
F	15	微信
F	25	微信
M	32	微信
F	40	微信
M	12	微信
M	14	微信

Gender	Age	App
F	15	微信
F	25	微信
M	32	微信
F	40	微信
M	12	微信
M	14	微信



属性“性别”的信息增益:

$$Ent(D_1) = -(\frac{1}{3} \log_2 \frac{1}{3} + \frac{2}{3} \log_2 \frac{2}{3}) = 0.92$$

$$Ent(D_2) = -(\frac{1}{3} \log_2 \frac{1}{3} + \frac{2}{3} \log_2 \frac{2}{3}) = 0.92$$

$$Gain(D, \text{gender}) = Ent(D) - \sum_{v=1}^2 \frac{|D_v|}{|D|} Ent(D_v) = 1.46 - (\frac{3}{6} \times 0.92 + \frac{3}{6} \times 0.92) = 0.54$$

$$Gain(D, \text{age}) = Ent(D) - \sum_{v=1}^2 \frac{|D_v|}{|D|} Ent(D_v) = 1.46 - (\frac{3}{6} \times 0 + \frac{3}{6} \times 0.92) = 1 > 0.54$$

属性“年龄”的信息增益: 以 20 划分年龄区间,

$$Ent(D_1) = -1 \log_2 1 = 0$$

$$Ent(D_2) = -(\frac{1}{3} \log_2 \frac{1}{3} + \frac{2}{3} \log_2 \frac{2}{3}) = 0.92$$

因此首先选择“年龄”属性进行数据划分, 最后得到的决策树如上图所示。对新样本“年龄为 30 的女性用户”, 依据决策树进行判断, 优先推荐“微信”APP。

# 信息增益 (4)

《机器学习》例子：以信息增益为划分指标，对如下西瓜数据集 2.0 构建决策树。数据集包含两个类别： $|\gamma| = 2$ ,  $p_1(yes) = \frac{8}{17}$ ,  $p_2(no) = \frac{9}{17}$ 。

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

根节点信息熵为：

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

如果使用“色泽”属性划分  $D$ ，得到 3 个子集：

$$D_1(\text{色泽} = \text{青绿}) = \{1, 4, 6, 10, 13, 17\}$$

$$D_2(\text{色泽} = \text{乌黑}) = \{2, 3, 7, 8, 9, 15\}$$

$$D_3(\text{色泽} = \text{浅白}) = \{5, 11, 12, 14, 16\}$$

其信息熵分别为：

$$Ent(D_1) = -(\frac{3}{6} \log_2 \frac{3}{6} + \frac{3}{6} \log_2 \frac{3}{6}) = 1.000$$

$$Ent(D_2) = -(\frac{4}{6} \log_2 \frac{4}{6} + \frac{2}{6} \log_2 \frac{2}{6}) = 0.918$$

$$Ent(D_3) = -(\frac{1}{5} \log_2 \frac{1}{5} + \frac{4}{5} \log_2 \frac{4}{5}) = 0.722$$

这样可以得到色泽对根节点的信息增益：

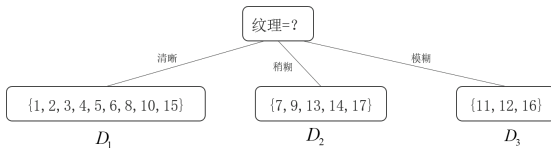
$$Gain(D, \text{色泽}) = Ent(D) - \sum_{v=1}^3 \frac{|D^v|}{|D|} Ent(D^v) = 0.998 - (\frac{6}{17} \times 1.000 + \frac{6}{17} \times 0.918 + \frac{5}{17} \times 0.722) = 0.109$$

# 信息增益 (5)



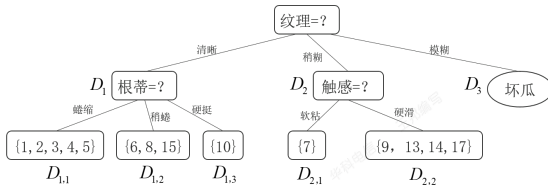
同理，可计算出其他属性对根节点的信息增益： $Gain(D, \text{根蒂}) = 0.143$ ;  $Gain(D, \text{纹理}) = 0.381$ ;  
 $Gain(D, \text{触感}) = 0.006$ ;  $Gain(D, \text{敲声}) = 0.141$ ;  $Gain(D, \text{脐部}) = 0.289$ .

纹理的信息增益最大，选择纹理属性划分根节点  $D$  为  $D_1, D_2, D_3$

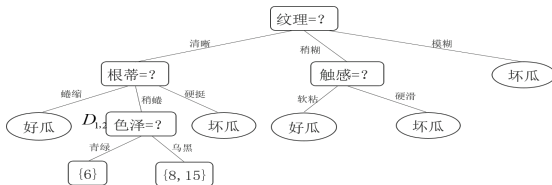


$D_3$  满足情形 (1) (所样本同类别)，成为叶子节点，继续以相同步骤划分  $D_1, D_2$ :

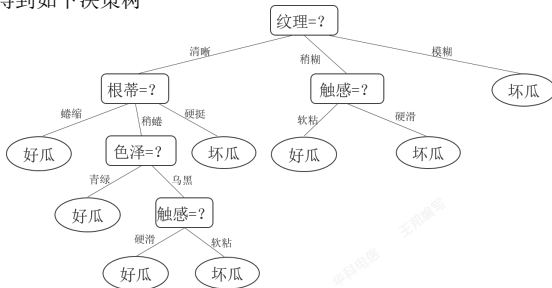
$Gain(D_1, \text{根蒂}) = 0.458$ ;  $Gain(D_1, \text{色泽}) = 0.043$ ;  $Gain(D_1, \text{触感}) = 0.458$ ;  
 $Gain(D_1, \text{敲声}) = 0.331$ ;  $Gain(D_1, \text{脐部}) = 0.458$ .



$D_1$  的分支  $D_{1,1}, D_{1,3}, D_2$  的分支  $D_{2,1}, D_{2,2}$  均满足情形 (1), 成为叶子节点, 继续以相同步骤划分  $D_{1,2}$



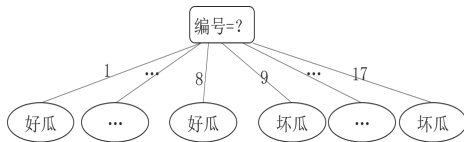
不断迭代, 最终可得到如下决策树



# 增益率 (Gain Ratio)

对信息增益的不足之处的思考：如果在上述例子中**将编号作为属性划分**？

每个样本都有一个编号，将产生 17 个分支，纯度将达到最大，其信息增益为  $Gain(D, \text{编号}) = Ent(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} Ent(D^v) = 0.998$ 。但是这样的模型却不具备泛化能力，无法预测新样本。



分析：**信息增益偏好可取值数目多的属性的特点**，不能公平对待每个属性。

改进：引入**增益率 (Gain Ratio)**。

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

$IV(a)$  为属性  $a$  的“固有值”，属性  $a$  的可能取值越多（即  $V$  越大），则  $IV(a)$  的值通常会越大。因此上式加入  $IV(a)$ ，用以惩罚取值数目多的属性。

**C4.5 决策树**结合了信息增益与增益率进行属性划分，先从候选划分属性中找出信息增益高于平均水平的属性，再从中选择增益率最高的属性。



# 基尼指数 (Gini Index)

以基尼值 (Gini Value) 衡量集合纯度

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

反映从  $D$  中随机抽取两个样本，其类别不一致的概率， $Gini(D)$  值越小，则集合  $D$  的纯度越高。

**基尼指数 (Gini Index):**

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

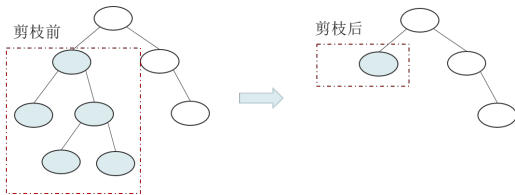
**CART 决策树**采用基尼指数进行属性划分。在候选属性集合中，每次选择使得**划分后基尼指数最小的属性**作为最优划分属性；即

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

- 1 决策树 (Decision Tree)
- 2 划分选择 (Partitioning)
- 3 决策树剪枝 (Pruning the Tree)**
- 4 连续值与缺失值 (Continuous Values & Missing Values.)
- 5 多变量决策树 (Multivariate Decision Tree)
- 6 小结 Summary

# 剪枝处理 (Pruning)

剪枝是指将一颗子树的子结点全部删掉，根结点作为叶子结点。决策树剪枝的基本策略有“预剪枝”(pre-pruning)和“后剪枝”(post-pruning)。



## 为什么要剪枝处理？

- 决策树充分考虑了所有的数据点，有可能出现过拟合的情况，决策树越复杂，过拟合的程度会越高。
- 剪枝修剪分裂前后分类误差相差不大的子树，能够降低决策树的复杂度，降低过拟合出现的概率。

# 预剪枝 (1)



**预剪枝**是指在决策树生成过程中, 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作为叶结点。

**如何判断决策树泛化性能是否提升?**

采用“**留出法**”, 预留一部分数据用作“**验证集**”进行评估。以西瓜数据集为例, 我们将西瓜数据集划分为训练集和验证集, 在接下来的例子中, 我们采用训练集训练模型, 用验证集评估模型的性能。如果通过某种方式 (如属性划分) 提升了模型在验证集上的性能 (如验证集准确率), 我们则认为该方法使模型的泛化性能得到了提升。

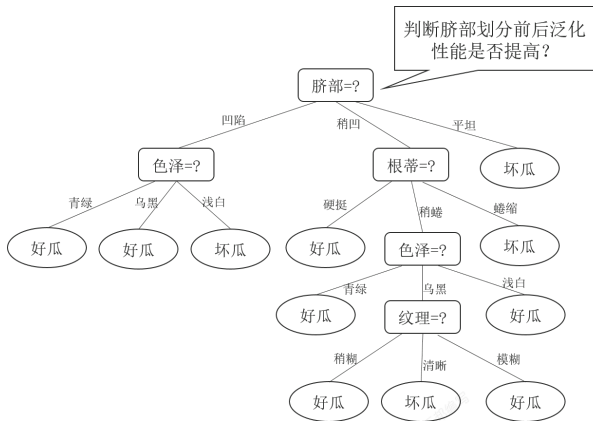
编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

训练集

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

验证集

我们对训练集生成的未剪枝决策树采用预剪枝方式进行剪枝：首先我们判断**脐部**划分前后泛化性能（验证集准确率）是否提高？



训练集生成的未剪枝决策树

# 预剪枝 (3)

## 不划分:

决策树只有一个节点,  $D$  作为叶结点, 以情形 (2) (属性为空不再划分) 标记为训练样例数目最多的类别 “好瓜”。

验证集 7 个样本中 4,5,8 正确, 验证集精度为:  
 $\frac{3}{7} \times 100\% = 42.9\%$

脐部 = ?

D

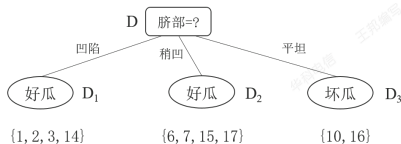
好瓜

## 划分:

$D$  被划分成  $D_1$ (凹陷),  $D_2$ (稍凹),  $D_3$ (平坦) 三个节点,  $D_1, D_2$  以情形 (2) 标记为训练样例数目最多的类别 “好瓜”,  $D_3$  满足情形 (1) 标记为坏瓜。

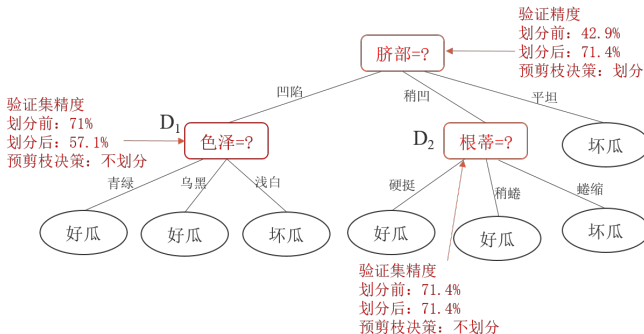
验证集中 4,5,8,11,12 正确, 验证集精度为:

$$\frac{5}{7} \times 100\% = 71.4\% > 42.9\%$$



预剪枝决策: 划分

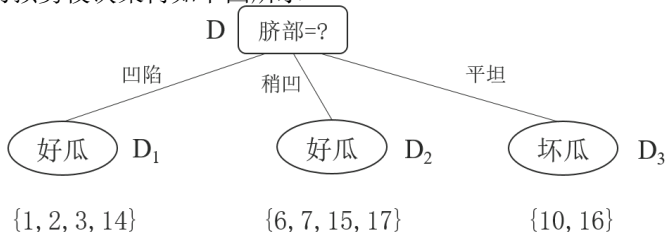
继续以同样方式划分节点  $D_1$ ,  $D_2$  如下图所示



决策树的预剪枝示意图

注意, “色泽”节点  $D_1$  被划分后, 验证集数据 5 被错误分类为坏瓜, 而根蒂节点  $D_2$  的划分对验证集分类结果没有影响, 故两个节点都不进行划分。

最终生成的预剪枝决策树如下图所示



- **预剪枝优点：**预剪枝使决策树的很多分支没有“展开”，这不仅降低了过拟合的风险，还显著减少了决策树的训练时间开销和测试时间开销。
- **预剪枝不足：**有些分支的当前划分虽不能提升泛化性能、甚至导致泛化性能暂时下降，但在其基础上进行的后续划分却有可能导致性能显著提高；预剪枝基于“贪心”本质禁止这些分支展开，给预剪枝决策树带来了欠拟合的风险。



# 后剪枝 (1)

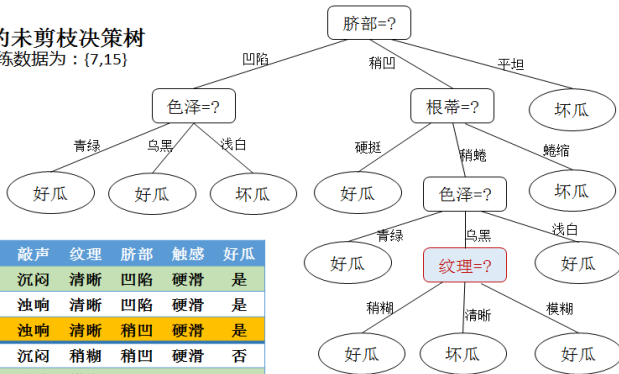


后剪枝是先从训练集生成一棵完整的决策树，然后自底向上地对非叶结点进行考察，若将该结点对应的子树替换为叶结点能带来决策树泛化性能提升，则将该子树替换为叶结点。

以训练集生成的如下决策树为例子，我们对“纹理”节点进行剪枝。

训练集生成的未剪枝决策树

纹理结点的训练数据为：{7,15}



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

剪枝前：验证集{4,11,12}正确分类

剪枝前精度为：3/7 = 42.9%

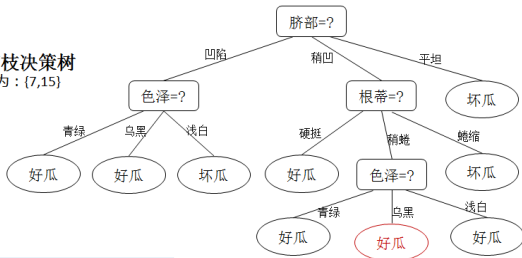
# 后剪枝 (2)



后剪枝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。

以训练集生成的如下决策树为例子，我们对“纹理”节点进行剪枝。

训练集生成的未剪枝决策树  
纹理结点的训练数据为：{7,15}



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

纹理结点的训练数据为：{7“好瓜”,15“坏瓜”}。

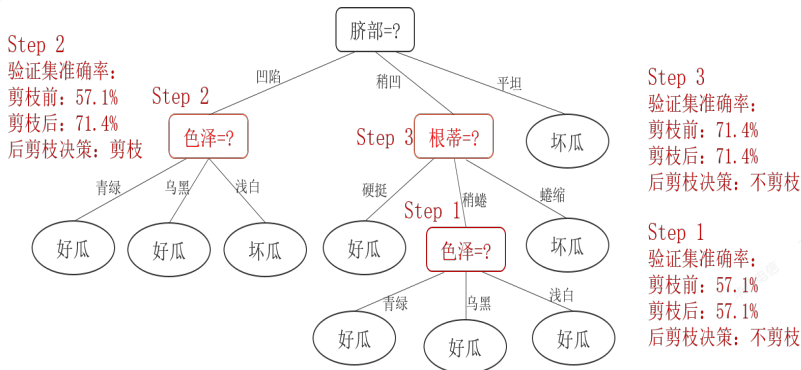
- 若剪枝为“好瓜”，则验证集中样本8分类正确；
- 若剪枝为“坏瓜”，则验证集中样本9分类正确。

剪枝后精度：4/7=57.1%

剪枝前精度：3/7 = 42.9%

后剪枝决策：剪枝

接下来，将如下决策树以同样方式依次对“色泽”，“根蒂”节点进行剪枝。

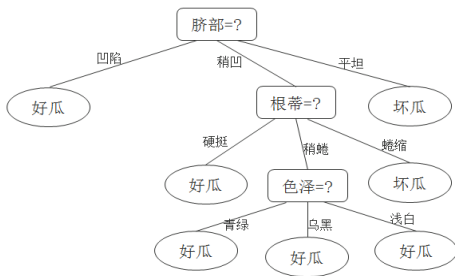


注意，Step 1，Step 3 不能提升泛化性能不进行剪枝。在 Step2 处，验证集数据 5 被正确分类，提升了验证集准确率，进行了剪枝。

## 后剪枝 (4)



最终，我们对训练集生成的决策树进行后剪枝如下图所示，剪枝后的验证集准确率为 71.4%



- **后剪枝优点：**一般情况下，后剪枝决策树的欠拟合风险很小，泛化性能往往优于预剪枝决策树。
- **后剪枝不足：**后剪枝过程是在生成完全决策树之后进行的，并且要自底向上对树中所有非叶节点逐一考察，因此训练时间开销比未剪枝和预剪枝决策树要大得多。

- 1 决策树 (Decision Tree)
- 2 划分选择 (Partitioning)
- 3 决策树剪枝 (Pruning the Tree)
- 4 连续值与缺失值 (Continuous Values & Missing Values.)
- 5 多变量决策树 (Multivariate Decision Tree)
- 6 小结 Summary

# 连续值处理 (1)



之前关于属性划分的讨论都是基于离散属性的假设，但下列西瓜 3.0 数据集中包含了密度，含糖率两个连续属性，那么如何在含有连续值属性的数据集上进行决策树的构建呢？

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	0.46	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	0.36	0.37	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

西瓜数据集 3.0，来源：周志华，《机器学习》，清华大学出版社

可以采用连续属性离散化的方法进行，比如采用二分法 (bi-partition) 把连续属性化作两个区间，C4.5 决策树也采用了这种方法。

连续属性的候选划分点集合  $T_a$ : 给定样本集  $D$  和连续属性  $a$ , 假定  $a$  在  $D$  上出现了  $n$  个不同的取值, 将这些值从小到大进行排序, 记为  $\{a_1, a_2, \dots, a_n\}$ . 基于划分点  $t$  可将  $D$  分为子集  $D_t^-$  和  $D_t^+$ , 其中  $D_t^-$  包含在属性  $a$  上取值不大于  $t$  的样本, 而  $D_t^+$  包含在属性  $a$  上取值大于  $t$  的样本. 对相邻的属性取值  $a_i$  与  $a_{i+1}$  来说,  $t$  在区间  $[a_i, a_{i+1})$  中取任意值所产生的划分结果相同. 因此, 对连续属性  $a$ , 可考察包含  $n-1$  个元素的候选划分点集合:

$$T_a = \left\{ \frac{a_i + a_{i+1}}{2} \mid 1 \leq i \leq n-1 \right\}$$

即把区间  $[a_i, a_{i+1})$  的中位点  $\frac{a_i + a_{i+1}}{2}$  作为候选划分点, 即可以像离散属性值一样来考察这些划分点, 从候选集合  $T_a$  中选择最优的划分点  $t_m$  进行划分.

连续属性的信息增益  $Gain(D, a)$ :

$$Gain(D, a) = \max_{t \in T_a} Gain(D, a, t) = \max_{t \in T_a} Ent(D) - \sum_{\lambda \in \{-, +\}} \frac{|D_t^\lambda|}{|D|} Ent(D_t^\lambda)$$

其中,  $Gain(D, a, t)$  是样本集  $D$  基于划分点  $t$  二分后的信息增益, 我们选择使  $Gain(D, a, t)$  最大化的划分点  $t_m \in T_a$ , 求得的信息增益  $Gain(D, a, t_m)$  即为连续属性  $a$  对集合  $D$  的信息增益.

# 连续值处理 (3)

计算西瓜数据集 3.0 中密度和含糖率对整体数据集的信息增益？

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	0.46	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	0.36	0.37	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

求取步骤如下

- ① 对连续属性排序
- ② 计算候选划分点集合  $T_a$
- ③ 计算对  $\forall \in T_a$  的  $Gain(D, a, t)$ , 取最大值, 即可求得  $Gain(D, a)$

- $T_{\text{密度}} = \{0.244, 0.294, 0.351, 0.381, 0.420, 0.459, 0.518, 0.574, 0.600, 0.621, 0.636, 0.648, 0.661, 0.681, 0.708, 0.746\}$   
当  $t = 0.381$  时,  $Gain(D, a, t)$  取最大值,  
 $Gain(D, \text{密度}) = 0.262$
- $T_{\text{含糖率}} = \{0.049, 0.074, 0.095, 0.101, 0.126, 0.155, 0.179, 0.204, 0.213, 0.226, 0.250, 0.265, 0.292, 0.344, 0.373, 0.418\}$   
当  $t = 0.126$  时,  $Gain(D, a, t)$  取最大值,  
 $Gain(D, \text{含糖率}) = 0.349$

**注意:** 与离散属性不同的是, 若当前节点划分属性为连续属性, 该属性还可作为后代节点的划分属性。如在父节点上使用了”密度  $\leq 0.381$ “, 不会禁止在其子节点上使用”密度  $\leq 0.294$ “



# 缺失值处理 (1)

现实任务中，往往会遇到如西瓜数据集 2.0 $\alpha$  所示的含有缺失值的数据，图中大部分样本与属性都含有缺失值，不能删除这些样本或属性，那么决策树应该怎么应对这样的数据集呢？

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

对于缺失值的处理，需要解决两个问题：

- ① 训练阶段：如何在属性值缺失的情况下进行划分属性选择？
- ② 验证（剪枝）阶段及测试阶段：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？

西瓜数据集 2.0 $\alpha$ ，来源：周志华，《机器学习》，清华大学出版社

# 缺失值处理 (2)

首先我们做如下定义来针对含缺失值的数据集:

给定训练集  $D$  和属性  $a$ ,  $\tilde{D}$  表示  $D$  中在属性  $a$  上没有缺失值的样本子集。属性  $a$  可取值  $\{a_1, a_2, \dots, a_v, \dots, a_V\}$ , 令  $\tilde{D}_v$  表示  $\tilde{D}$  在属性  $a$  上取值为  $a_v$  的样本子集, 令  $\tilde{D}^{(k)}$  表示  $\tilde{D}$  中属于第  $k$  类 ( $k = 1, 2, \dots, |\gamma|$ ) 的样本子集。我们为每一个样本  $x$  赋予一个权重  $\omega_x$  ( $\omega$  初始化为 1), 并定义

$$\rho = \frac{\sum_{x \in \tilde{D}} \omega_x}{\sum_{x \in D} \omega_x}, \quad \text{表示 } a \neq null \text{ 的样本所占的比例,}$$

$$\tilde{p}_k = \frac{\sum_{x \in \tilde{D}^{(k)}} \omega_x}{\sum_{x \in \tilde{D}} \omega_x} \quad (1 \leq k \leq |\gamma|), \quad \text{表示 } a \neq null \text{ 的样本中第 } k \text{ 类所占的比例,}$$

$$\tilde{r}_v = \frac{\sum_{x \in \tilde{D}_v} \omega_x}{\sum_{x \in \tilde{D}} \omega_x} \quad (1 \leq v \leq V), \quad \text{表示 } a \neq null \text{ 的样本中取值为 } a = a_v \text{ 的样本所占的比例}$$

对于问题 (1), 我们可基于上述定义将信息增益推广为

$$\begin{aligned} Gain(D, a) &= \rho \times Gain(\tilde{D}, a) \\ &= \rho \times (Ent(\tilde{D}) - \sum_{v=1}^V \tilde{r}_v Ent(\tilde{D}_v)) \end{aligned}$$

$$\text{其中, } Ent(\tilde{D}) = - \sum_{k=1}^{|\gamma|} \tilde{p}_k \log_2 \tilde{p}_k$$

对于问题 (2), 给定划分属性划分样本:

- ① 若样本  $x$  在划分属性  $a$  上的取值已知, 则将  $x$  划入与其取值对应的子节点, 且样本权值在子节点中保持  $\omega_x$  不变;
- ② 若样本  $x$  在划分属性  $a$  上的取值缺失  $a = null$ , 则将  $x$  同时划入所有子节点, 且样本权值在属性值  $a_v$  对应的子节点中调整为  $\tilde{r}_v \cdot \omega_x$ , 相当于将一个样本以不同的概率划入到不同的子节点中。

# 缺失值处理 (3)

对含有缺失值的西瓜数据集 2.0 $\alpha$  构建决策树。

在学习开始时，根结点包含样本集  $D$  中全部 17 个样例，各样例的权值均为 1。以属性“色泽”为例，该属性上无缺失值的样例子集  $\tilde{D}$  包含编号为 {2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17} 的 14 个样例。

$\tilde{D}$  的信息熵为： $Ent(\tilde{D}) = -\sum_{k=1}^2 \tilde{p}_k \log_2 \tilde{p}_k = -(\frac{6}{14} \log_2 \frac{6}{14} + \frac{8}{14} \log_2 \frac{8}{14}) = 0.985$

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

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

$$Ent(\tilde{D}^1) = -(\frac{2}{4} \log_2 \frac{2}{4} + \frac{2}{4} \log_2 \frac{2}{4}) = 1.000,$$

$$Ent(\tilde{D}^2) = -(\frac{4}{6} \log_2 \frac{4}{6} + \frac{2}{6} \log_2 \frac{2}{6}) = 0.918,$$

$$Ent(\tilde{D}^3) = -(\frac{0}{4} \log_2 \frac{0}{4} + \frac{4}{4} \log_2 \frac{4}{4}) = 0.000,$$

因此，样本子集  $\tilde{D}$  上属性“色泽”的信息增益为：

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

(来源：周志华，《机器学习》，清华大学出版社)

于是，样本集  $D$  上属性“色泽”的信息增益为：

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

# 缺失值处理 (4)

类似的，可计算出所有属性在  $D$  上的信息增益：

$$Gain(D, \text{色泽}) = 0.252, \quad Gain(D, \text{根蒂}) = 0.171, \quad Gain(D, \text{敲声}) = 0.145$$

$$Gain(D, \text{纹理}) = \mathbf{0.424}, \quad Gain(D, \text{脐部}) = 0.289, \quad Gain(D, \text{触感}) = 0.006$$

“纹理”在所有属性中取得了最大的信息增益，被用于对根结点进行划分。划分结果是使编号为  $\{1, 2, 3, 4, 5, 6, 15\}$  的样本进入“纹理 = 清晰”分支，编号为  $\{7, 9, 13, 14, 17\}$  的样本进入“纹理 = 模糊”分支，而编号为  $\{11, 12, 16\}$  的样本进入“纹理 = 模糊”分支，且样本在各子节点上的权值保持为 1。需要注意的是，编号为  $\{8\}$  的样本在属性“纹理”上出现了缺失值，因此它将同时进入三个分支中，但权值在三个子结点中分别调整为  $\frac{7}{15}$ ， $\frac{5}{15}$  和  $\frac{3}{15}$ 。编号为  $\{10\}$  的样本有类似划分结果。

上述结点划分过程递归执行，最终生成的决策树如图所示：



在西瓜数据集 2.0 $\alpha$  上基于信息增益生成的决策树

来源：周志华，《机器学习》，清华大学出版社

# 思考题



以信息增益为划分标准，以如下含有缺失值和连续值属性的西瓜数据集 2.0 $\beta$  构建决策树。

编号	色泽	根蒂	敲声	纹理	脐部	触感	密度	含糖率	好瓜
1	-	蜷缩	浊响	清晰	凹陷	硬滑	0.697	0.46	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	-	0.774	0.376	是
3	乌黑	蜷缩	-	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	-	蜷缩	浊响	清晰	凹陷	硬滑	0.556	0.215	是
6	青绿	稍蜷	浊响	清晰	-	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	-	稍凹	硬滑	0.437	0.211	是
9	乌黑	-	沉闷	稍糊	稍凹	硬滑	0.666	0.091	否
10	青绿	硬挺	清脆	-	平坦	软粘	0.243	0.267	否
11	浅白	硬挺	清脆	模糊	平坦	-	0.245	0.057	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	0.343	0.099	否
13	-	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	-	软粘	0.36	0.37	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	0.593	0.042	否
17	青绿	-	沉闷	稍糊	稍凹	硬滑	0.719	0.103	否

西瓜数据集 2.0 $\beta$ 。来源：周志华，《机器学习》，清华大学出版社

- 1 决策树 (Decision Tree)
- 2 划分选择 (Partitioning)
- 3 决策树剪枝 (Pruning the Tree)
- 4 连续值与缺失值 (Continuous Values & Missing Values.)
- 5 多变量决策树 (Multivariate Decision Tree)
- 6 小结 Summary

# 多变量决策树 (1)



若我们将每个属性看成是坐标空间中的一个坐标轴，则 $d$  个属性描述的样本就对应了 $d$  维空间中的一个数据点，对样本进行分类则意味着在这个坐标空间中寻找不同样本类别之间的**分类边界**。决策树所形成的分类边界有一个明显的特点：**轴平行**(axis-parallel)，即它的分类边界由若干个与坐标轴平行的分段组成。

以下图中的西瓜数据 3.0 $\alpha$  为例，将它作为训练集可学得如下决策树，及其对应的分类边界。显然，分类边界的每一段都是与坐标轴平行的这样的分类边界使得学习结果有较好的可解释性，因为每一段划分都直接对应了某个属性取值。但在学习任务的真实分类边界比较复杂时，必须使用很多段划分才能获得较好的近似。

表 4.5 西瓜数据集 3.0 $\alpha$

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

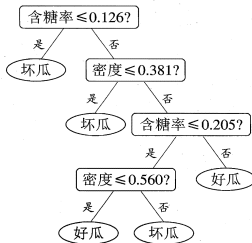


图 4.10 在西瓜数据集 3.0 $\alpha$  上生成的决策树

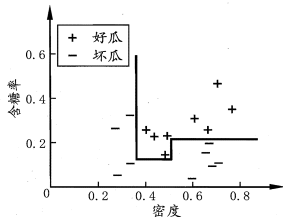


图 4.11 图 4.10 决策树对应的分类边界

(来源：周志华，《机器学习》，清华大学出版社)

# 多变量决策树 (2)



**多变量决策树**(multivariate decision tree) 使用斜的划分边界，从而简化决策树模型。

以实现斜划分的多变量决策树为例，在此类决策树中，非叶结点不再是仅对某个属性，而是对属性的线性组合进行测试；换言之，每个非叶结点是一个形如  $\sum_{i=1}^d w_i a_i = t$  的线性分类器，其中  $w_i$  是属性  $a_i$  的权重， $w_i$  和  $t$  可在该结点所含的样本集和属性集上学习得到。与前面介绍的**单变量决策树**(univariate decision tree) 不同，在在多变量决策树的学习过程中，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。

例如对西瓜数据 3.0。我们可学得如下图这样的多变量决策树，其分类边界如下图所示。

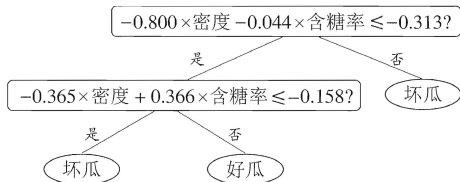


图 4.13 在西瓜数据集 3.0 $\alpha$  上生成的多变量决策树

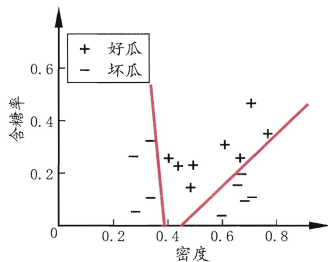


图 4.14 图 4.13 多变量决策树对应的分类边界

(来源：周志华，《机器学习》，清华大学出版社)



- 1 决策树 (Decision Tree)
- 2 划分选择 (Partitioning)
- 3 决策树剪枝 (Pruning the Tree)
- 4 连续值与缺失值 (Continuous Values & Missing Values.)
- 5 多变量决策树 (Multivariate Decision Tree)
- 6 小结 Summary

在本章中，我们主要讲解了决策树的原理及使用时的需注意的情况，主要包含以下内容：

- 决策树的概念与决策树构建的基本流程——递归过程；
- 如何选择当前节点的最优划分属性：可以使用信息增益、增益率以及 Gini 指数等等，掌握决策树构建过程；
- 如何通过剪枝解决决策树过拟合问题，提升泛化性能：可采用预剪枝、后剪枝，理解其过程与特点；
- 如何处理连续值与缺失值：针对连续值与缺失值推广信息增益等划分标准，理解其过程；
- 多变量决策树的概念：属性空间中可以采用“非轴平行”的方法对样本集进行划分。