

决策树

一、基本流程

二、划分选择

2.1 信息增益 (决策树 ID3 训练算法)

- 2.1.1 信息与熵的概念及度量

- 概念

- 熵

- 事物的不确定性

- 信息

- 消除不确定性的事物

- 调整概率

- 排除干扰

- 确定情况

- 噪音

- 不能消除的干扰

- 数据

- 噪音+噪音

- 熵如何度量

- 参照一个不确定事物作为单位

- 例如猜一次硬币的不确定性，记作1bit

- 抛硬币次数与结果不确定性呈指数关系

- 硬币n个，结果就为 2^n

- 等概率均匀分布

-

$$n = \log_2 m$$

- 假设我有m=10种等概率的不确定情况，则 $10=2^n$, $n=\log_{10}$

- 概率不等的一般分布

第 k 类样本所占比例 $p_k (k = 1, 2, 3, \dots, |Y|)$

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

- 2.1.2 信息熵

度量样本集合纯度最常用的指标

第 k 类样本所占比例 $p_k (k = 1, 2, 3, \dots, |Y|)$

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

- 2.1.3 信息增益

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

信息增益越大，则意味着使用属性a来进行划分所获得的“纯度提升”越大。

2.2 增益率（决策树C4.5训练算法）

- 2.2.1 信息增益
 - 准则对可取值数目较多的属性有所偏好，为减少偏好带来的影响，则使用“增益率”来选择最优划分属性

$$Gain_{ratio}(D, a) = \frac{Gain(D, a)}{IV(a)}$$

其中

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

称为属性a的“固有值”。

- 2.2.2 使用方法
 - 增益率准则对可取值数目较少的属性有所偏好，因此使用一个启发式：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。

2.3 基尼指数（CART决策树）

- 2.3.1 数据集D的纯度可用基尼值度量

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

属性a的基尼指数定义：

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

- 2.3.2 使用方法：

选择基尼指数最小的属性作为最优划分属性

$$a_* = argmin Gini_index(D, a)$$

三、剪枝处理

- 3.1 剪枝是决策树学习算法对付“过拟合”的主要手段
- 3.2 预剪枝
 - 在决策树生成过程中，对每个结点在划分前先进性估计，若当前结点的划分不能带来决策树泛化性提升，则停止划分并标记为叶结点
 - 基于信息增益生成的决策树



图 4.5 基于表 4.2 生成的未剪枝决策树

详细说明：



预剪枝：

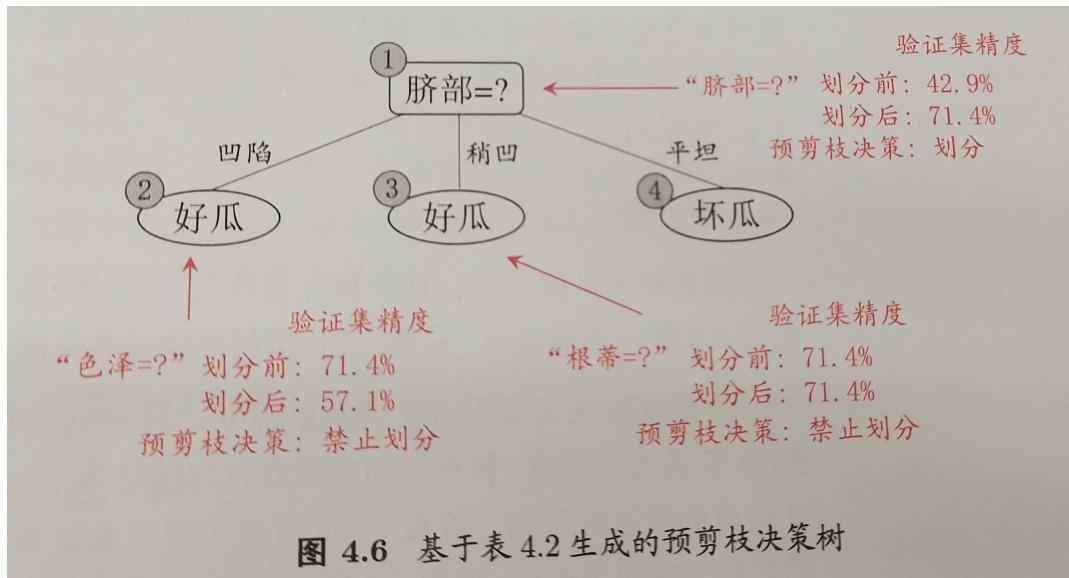


图 4.6 基于表 4.2 生成的预剪枝决策树

- 优点：预剪枝使得决策树很多分支没有展开，可以降低过拟合风险，显著减少了决策树的训练时间开销和测试时间开销
- 缺点：有些分支当前可能无法提升泛化性能，但可能后续会使性能显著提高带来了欠拟合的风险
- 3.3 后剪枝

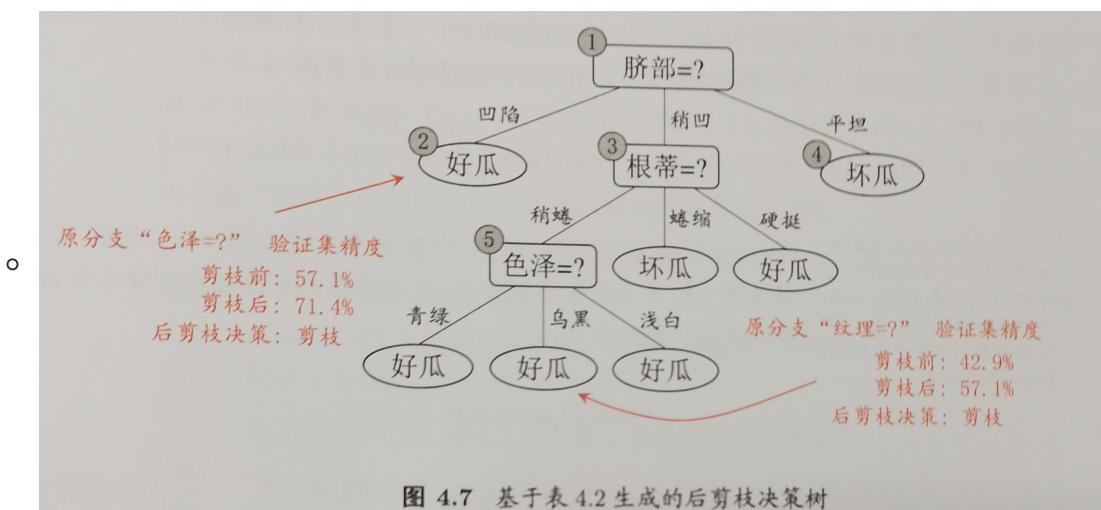


图 4.7 基于表 4.2 生成的后剪枝决策树

四、连续与缺失值

4.1 连续值处理

- C4.5算法
 - 4.1 二分法

对于连续属性 a ，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
选取最优的划分点进行样本集合的划分

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

$Gain(D, a, t)$ 是样本集D基于划分点t二分后的信息增益，然后就可以选择使 $Gain(D, a, t)$ 最大化的划分点

- 连续值基于信息增益生成的决策树：

于是，“纹理”被选作根结点划分属性，此后结点划分过程递归进行，最终生成如图 4.8 所示的决策树。

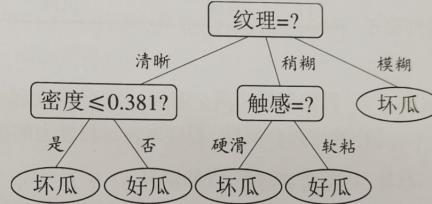


图 4.8 在西瓜数据集 3.0 上基于信息增益生成的决策树

例如在父结点上使用了“密度≤0.381”，不会禁止在子结点上使用“密度≤0.294”。

需注意的是，与离散属性不同，若当前结点划分属性为连续属性，该属性还可作为其后代结点的划分属性。

- 若当前结点划分属性为连续值，该属性还可以作为其后代结点的划分属性，注意看旁边的小字所说例子。

4.2 缺失值处理

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

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

D 训练集， \tilde{D} 表示在 D 中在属性 a 上没有缺失值的样本子集
 a, ρ 表示无缺失样本所占比例， \tilde{p}_k 表示无缺失样本中第 k 类所占的比例。

五、多变量决策树

- 单个的决策树，分类边界的每一段都是与坐标轴平行的，因为每一段划分直接对应了某个属性取值。

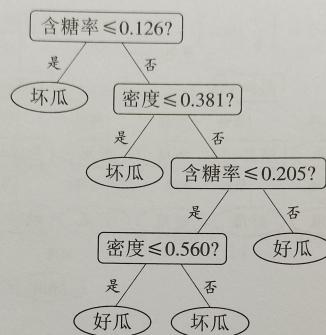
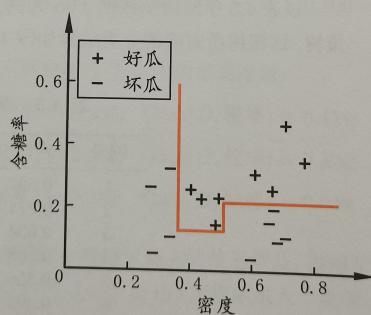
图 4.10 在西瓜数据集 3.0 α 上生成的决策树

图 4.11 图 4.10 决策树对应的分类边界

- 而在多变量决策树中，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。

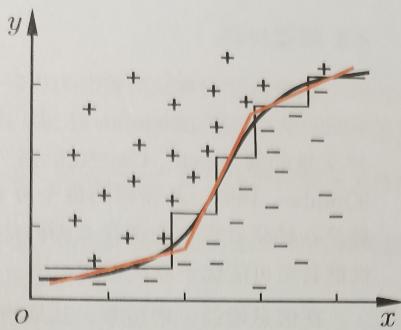


图 4.12 决策树对复杂分类边界的分段近似

分类器参见第 3 章。分类器，例如对西瓜数据 3.0 α ，我们可学得图 4.13 这样的多变量决策树，其分类边界如图 4.14 所示。

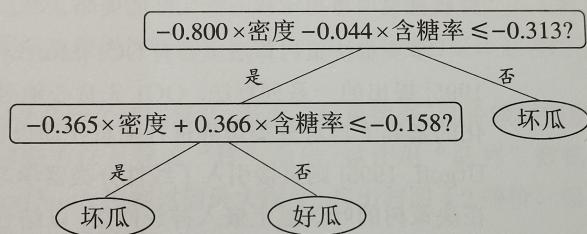


图 4.13 在西瓜数据集 3.0 α 上生成的多变量决策树

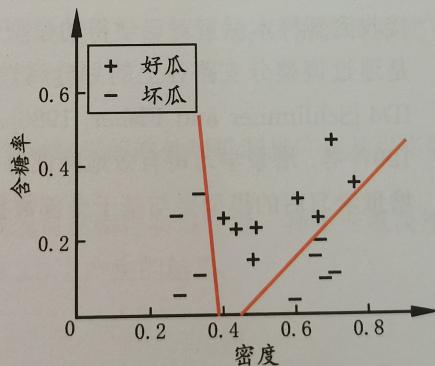


图 4.14 图 4.13 多变量决策树对应的分类边界