决策树与分类

Machine Learning Engineer

机器学习工程师

寒小阳

稀牛学院 網易 NetEase

目录

CONTENTS

01 决策树模型概述

02 算法流程与最优属性选择方法

03 剪枝与控制过拟合

04 数据案例讲解

01 决策树模型概述

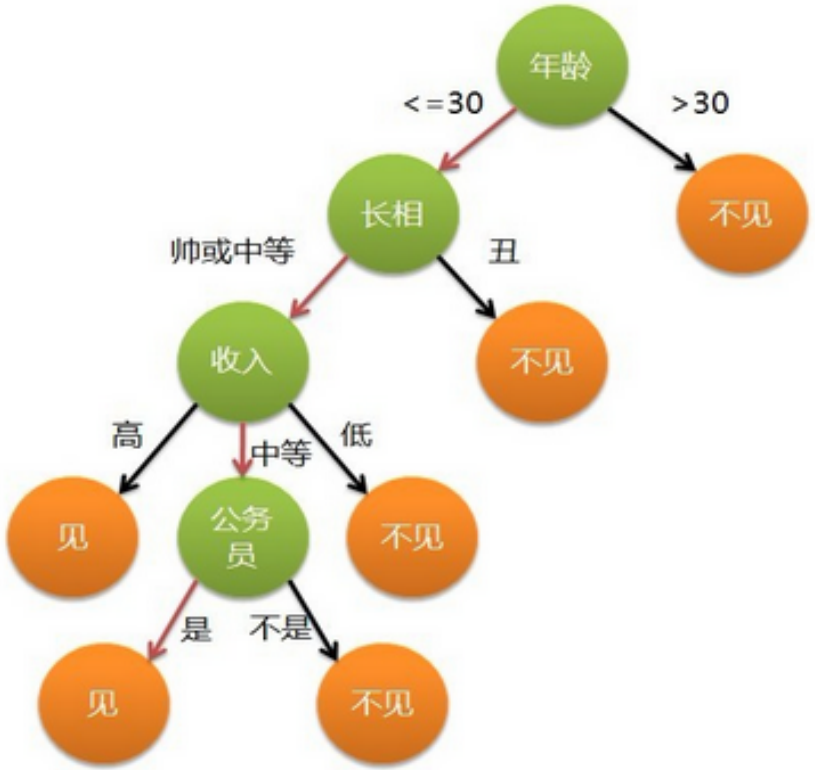
1.1 决策树模型

1.2 决策树简史

1.1 决策树模型

决策树模型(Decision Tree model) 是一个模拟人类决策过程思想的模型，以找对象为例，一个女孩的母亲要给这个女孩介绍男朋友，于是有了下面的对话：

女儿：多大年纪了？ (年龄)
母亲：26
女儿：长的帅不帅？ (长相)
母亲：挺帅的
女儿：收入高不？ (收入情况)
母亲：不算很高，中等情况
女儿：是公务员不？ (是否公务员)
母亲：是，在税务局上班呢。
女儿：那好，我去见见



简单、逻辑清晰、可解释性好

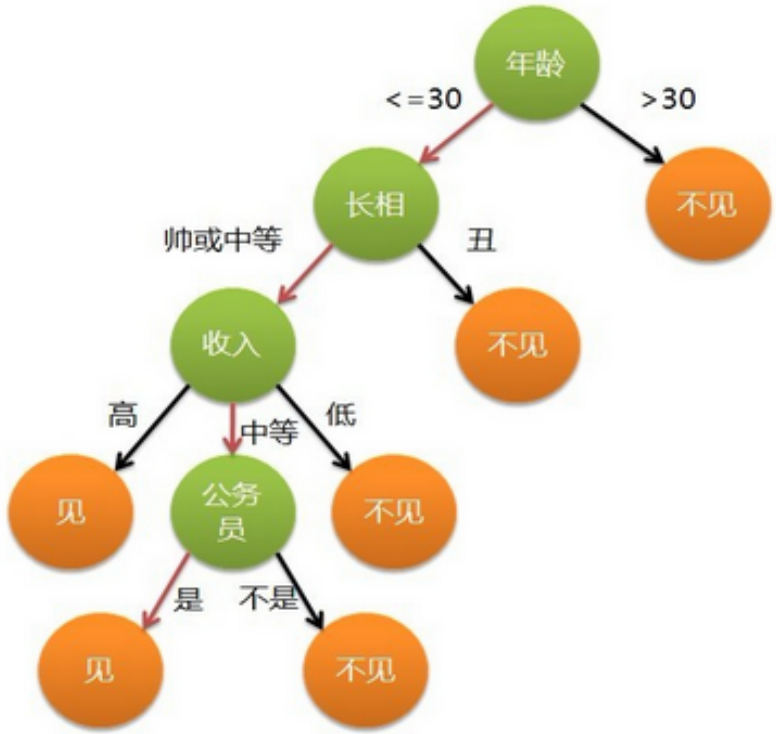
1.1 决策树模型

决策树基于“树”结构进行决策

- 每个“内部结点”对应于某个属性上的“测试”
- 每个分支对应于该测试的一种可能结果（即该属性的某个取值）
- 每个“叶结点”对应于一个“预测结果”

学习过程：通过对训练样本的分析来确定“划分属性”（即内部结点所对应的属性）

预测过程：将测试示例从根结点开始，沿着划分属性所构成的“判定测试序列”下行，直到叶结点



1.2 决策树简史

第一个决策树算法：CLS (Concept Learning System)

[E. B. Hunt, J. Marin, and P. T. Stone’s book “Experiments in Induction” published by Academic Press in 1966]

使决策树受到关注、成为机器学习主流技术的算法：ID3

[J. R. Quinlan’s paper in a book “Expert Systems in the Micro Electronic Age” edited by D. Michie, published by Edinburgh University Press in 1979]

最常用的决策树算法：C4.5

[J. R. Quinlan’s book “C4.5: Programs for Machine Learning” published by Morgan Kaufmann in 1993]

1.2 决策树简史

可以用于回归任务的决策树算法：CART (Classification and Regression Tree)

[L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone’s book “Classification and Regression Trees” published by Wadsworth in 1984]

基于决策树的最强大算法：RF (Random Forest)

[L. Breiman’s MLJ’01 paper “Random Forest”]

要点总结

- 决策树模型

  基于树的结构进行决策

  - 属性、测试、预测结果

  训练过程

  - 分析训练样本，确定划分属性

  预测过程

  - 沿着树结构根据属性进行下行判定

- 决策树简史
  - CLS
  - J. R. Quinlan 1979 ID3
  - J. R. Quinlan 1993 C4.5
  - L. Breiman 1984 CART
  - L. Breiman 2001 RandomForest

02 算法流程与最佳属性选择

2.1 决策树基本流程

2.2 最佳属性选择方法

2.3 熵与信息论视角

2.1 决策树基本流程

总体流程：

“分而治之”(divide-and-conquer)

- 自根至叶的递归过程
- 在每个中间结点寻找一个“划分”(split or test)属性

三种停止条件：

- 当前结点包含的样本全属于同一类别，无需划分；
- 当前属性集为空, 或是所有样本在所有属性上取值相同，无法划分；
- 当前结点包含的样本集合为空，不能划分。

2.1 决策树基本流程

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**

前面的(1)情形
递归返回

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**

前面的(2)情形
递归返回

8: 从 A 中选择最优划分属性 a_*;

利用当前结点的后验分布

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**

前面的(3)情形
递归返回

14: 以 TreeGenerate(D_v, $A \setminus \{a_*\}$)为分支结点
15: **end if**
16: **end for**

将父结点的样本分布作为
当前结点的先验分布

决策树算法的核心

输出： 以 node 为根结点的一棵决策树

信息熵 (entropy) 是度量样本集合“纯度”最常用的一种指标，假定当前样本集合 D 中第 k 类样本所占的比例为p_k，则 D 的信息熵定义为：

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

计算信息熵时约定：若 $p = 0$，则 $p \log_2 p = 0$.

$\mathrm{Ent}(D)$ 的最小值为 0，最大值为 $\log_2 |\mathcal{Y}|$.

$\mathrm{Ent}(D)$的值越小，则D的纯度越高

信息增益直接以信息熵为基础，计算当前划分对信息熵所造成的变化。

信息增益 (information gain)：ID3中使用

离散属性 a 的取值 $\{a^1, a^2, a^3, \dots a^V\}$:

D^v: D 中在 a 上取值 = a^v 的样本集合

以属性 a 对 数据集 D 进行划分所获得的信息增益为：

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

划分前的信息熵　　划分后的信息熵

第v个分支的权重，样本越多越重要

2.2 最佳属性选择方法

信息增益示例：

该数据集包含17个 训练样例，结果有2个类别$|y| = 2$，其中正例占$P_1 = \frac{8}{17}$ 反例占$P_2 = \frac{9}{17}$

根结点的信息熵为

$$Ent(D) = -\sum_{k=1}^{2} p_k log_2^{p_k}$$

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

周志华老师《机器学习》西瓜数据集

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
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

信息增益示例：

- 以属性“色泽”为例，其对应的3个数据子集分别为 D^1(色泽=青绿)，D^2 (色泽=乌黑)，D^3 (色泽=浅白)
- 子集 D^1 包含编号为$\{1,4,6,10,13,17\}$的6个样例，其中正例占 $p_1=\frac{3}{6}$，反例占 $p_2=\frac{3}{6}$，D^2, D^3 同理，3 个结点的信息熵为：

$$\begin{aligned}\mathrm{Ent}(D^1) &= -(\tfrac{3}{6}\log_2\tfrac{3}{6}+\tfrac{3}{6}\log_2\tfrac{3}{6}) = 1.000\\ \mathrm{Ent}(D^2) &= -(\tfrac{4}{6}\log_2\tfrac{4}{6}+\tfrac{2}{6}\log_2\tfrac{2}{6}) = 0.918\\ \mathrm{Ent}(D^3) &= -(\tfrac{1}{5}\log_2\tfrac{1}{5}+\tfrac{4}{5}\log_2\tfrac{4}{5}) = 0.722\end{aligned}$$

- 属性“色泽”的信息增益为

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

信息增益示例：

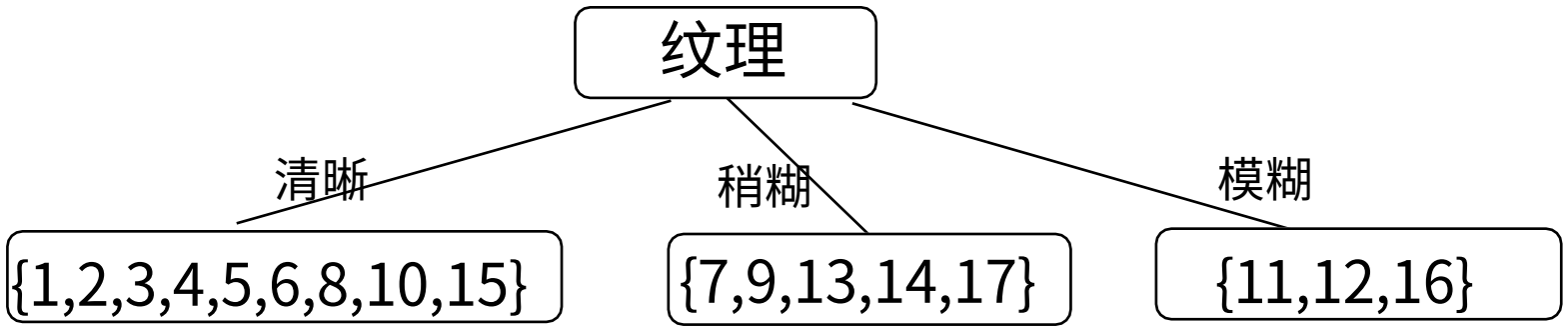
- 同样的方法，计算其他属性的信息增益为

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

$$\text{Gain}(D, \text{纹理}) = 0.381 \qquad \text{Gain}(D, \text{脐部}) = 0.289$$

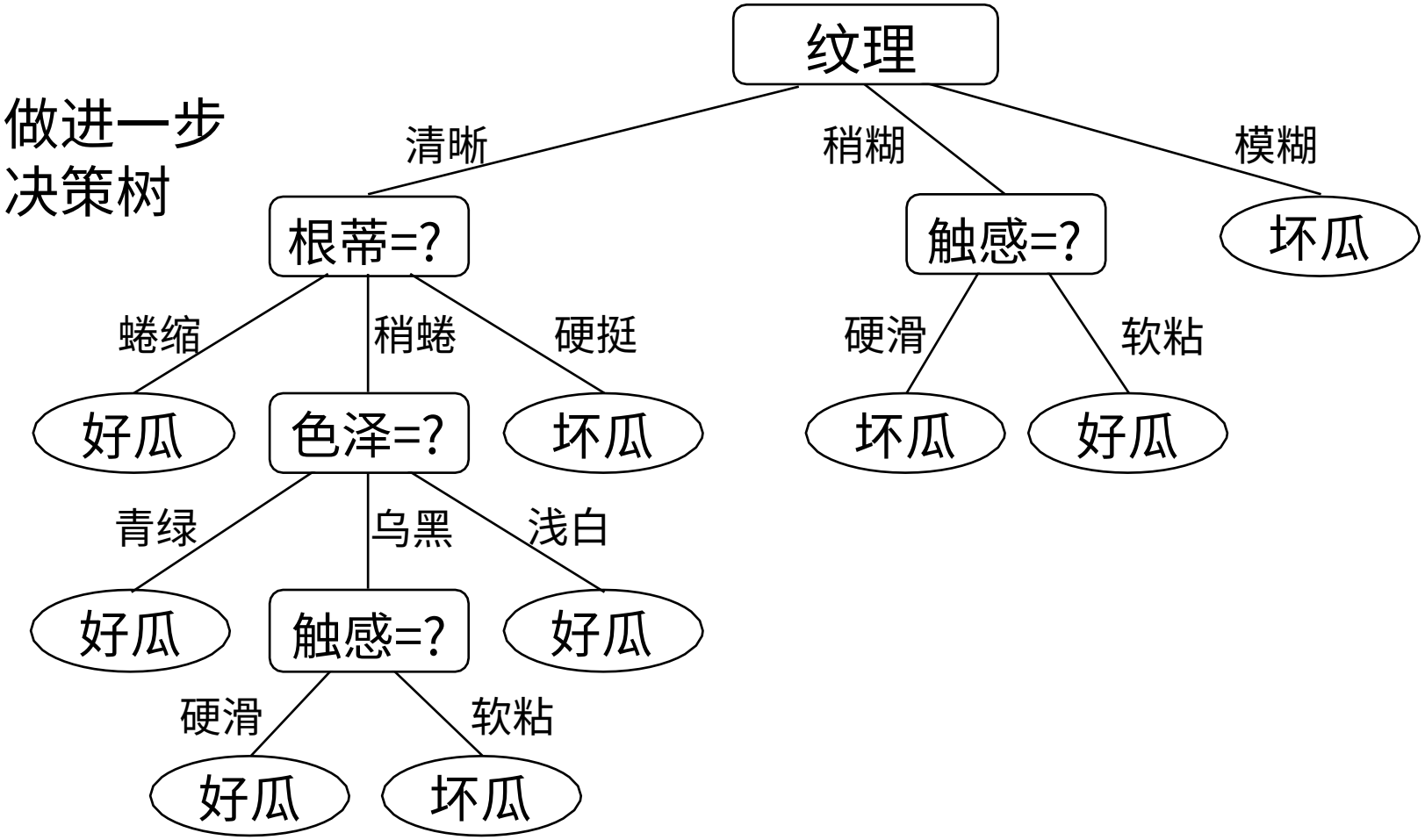
$$\text{Gain}(D, \text{触感}) = 0.006$$

- 显然，属性“纹理”的信息增益最大，其被选为划分属性



信息增益示例：

- 对每个分支结点做进一步划分，最终得到决策树



2.2 最佳属性选择方法

信息增益率 (gain ratio)：C4.5 中使用

信息增益的问题：对可取值数目较多的属性有所偏好

例如：考虑将“编号”作为一个属性

信息增益率：$$\text{Gain\_ratio}(D,a)=\frac{\text{Gain}(D,a)}{\text{IV}(a)}$$

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

属性a 的可能取值数目越多(即V 越大)，则IV(a) 的值通常就越大

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，再从中选取增益率最高的

基尼指数 (gini index)：CART 中使用

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

反映了从 D 中随机抽取两个样例，其类别标记不一致的概率

Gini(D) 越小，数据集D 的纯度越高

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

在候选属性集合中，选取那个使划分后基尼指数最小的属性

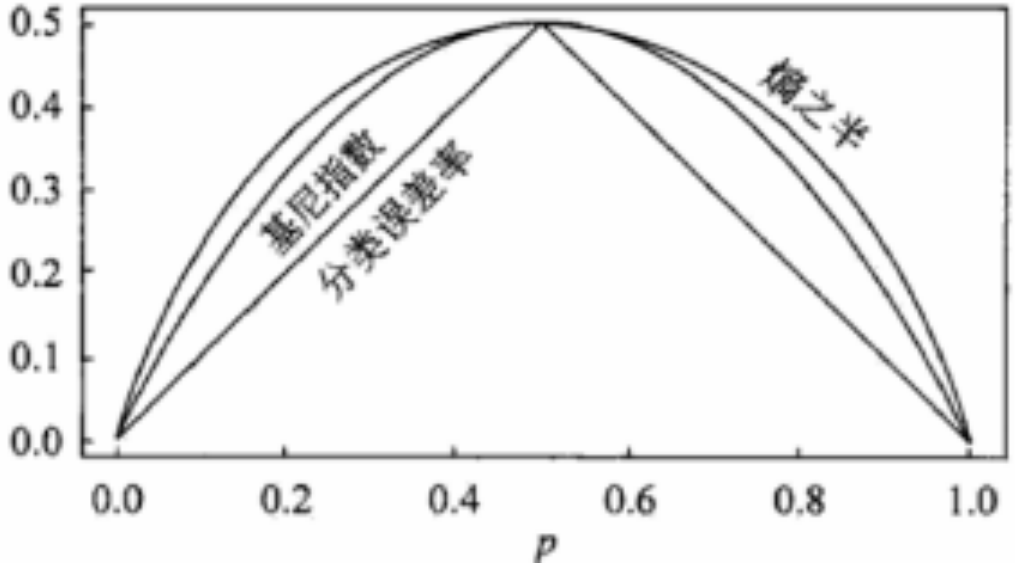
2.2 最佳属性选择方法

基尼指数、熵、分类误差率三者之间的关系：

- 将f(x)=-lnx在x=1处一阶泰勒展开，忽略高阶无穷小，得到f(x)≈1-x

$$H(X)=-\sum_{k=1}^{K} p_k \ln p_k$$

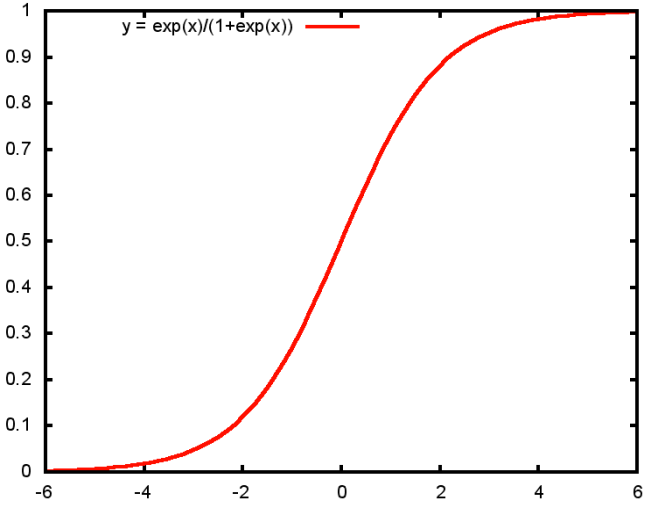
$$\approx \sum_{k=1}^{K} p_k(1-p_k)$$



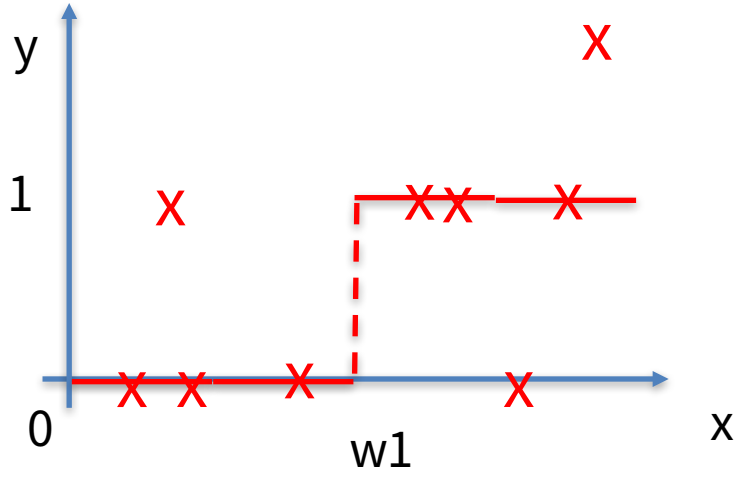
二分类视角看CART

- 每一个产生分支的过程就是一个二分类过程
- 这个过程叫作“决策树桩”： decision stump
- 一棵CART是由许多决策树桩拼接起来的
- decision stump是只有一层的决策树

逻辑回归

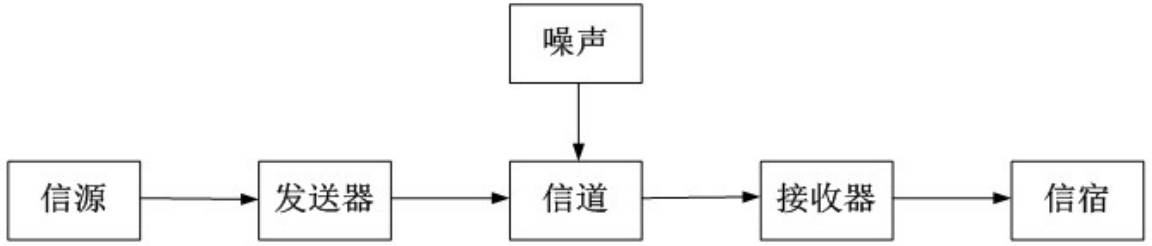


决策树桩



信息论的视角理解

- 对于多分叉树的情况，用信息论的视角来观察：机器学习其实是破解密码的过程



Y	X
信源	接受信号
标签	特征
英文字符	ASCII码
西瓜是否好瓜的结果	西瓜色泽等属性

信息论的视角理解

- 对于多分叉树的情况，用信息论的视角来观察：机器学习其实是破解密码的过程

信息论的视角	机器学习的视角
接受信号	特征
信源	标签
平均互信息	特征有效性分析
最大熵模型	极大似然法
交叉熵	逻辑回归损失函数

三种不同的决策树

- ID3：
  取值多的属性，更容易使数据更纯，其信息增益更大。
  训练得到的是一棵庞大且深度浅的树：不合理。

- C4.5
  采用信息增益率替代信息增益

- CART
  以基尼系数替代熵
  最小化不纯度，而不是最大化信息增益

要点总结

- 决策树模型
  - 分而治之
    - 递归划分至终止条件

① 结点样本属于同一属性

② 属性集空，所有样本属性相同

③ 结点包含样本集为空

- 最优属性选择
  - 信息增益 ID3决策树使用

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

  - 信息增益率 C4.5决策树使用

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

  - 基尼指数 CART决策树使用

$$\mathrm{Gini}(D) = 1 - \sum_{k=1}^{|\mathcal{Y}|} p_k^2 \quad \mathrm{Gini\_index}(D,a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|}\mathrm{Gini}(D^v)$$

03 剪枝与控制过拟合

3.1 决策树与剪枝操作

3.2 预剪枝过程与示例

3.3 后剪枝过程与示例

3.1 决策树与剪枝操作

为了尽可能正确分类训练样本，有可能造成分支过多，造成过拟合

剪枝：通过主动去掉一些分支来降低过拟合的风险

基本策略：

- 预剪枝(pre-pruning): 提前终止某些分支的生长
- 后剪枝(post-pruning): 生成一棵完全树，再“回头”剪枝

剪枝过程中需评估剪枝前后决策树的优劣

我们使用之前提到的“留出法”进行评估

再次以《机器学习》书中西瓜数据集为例，数据集如下：

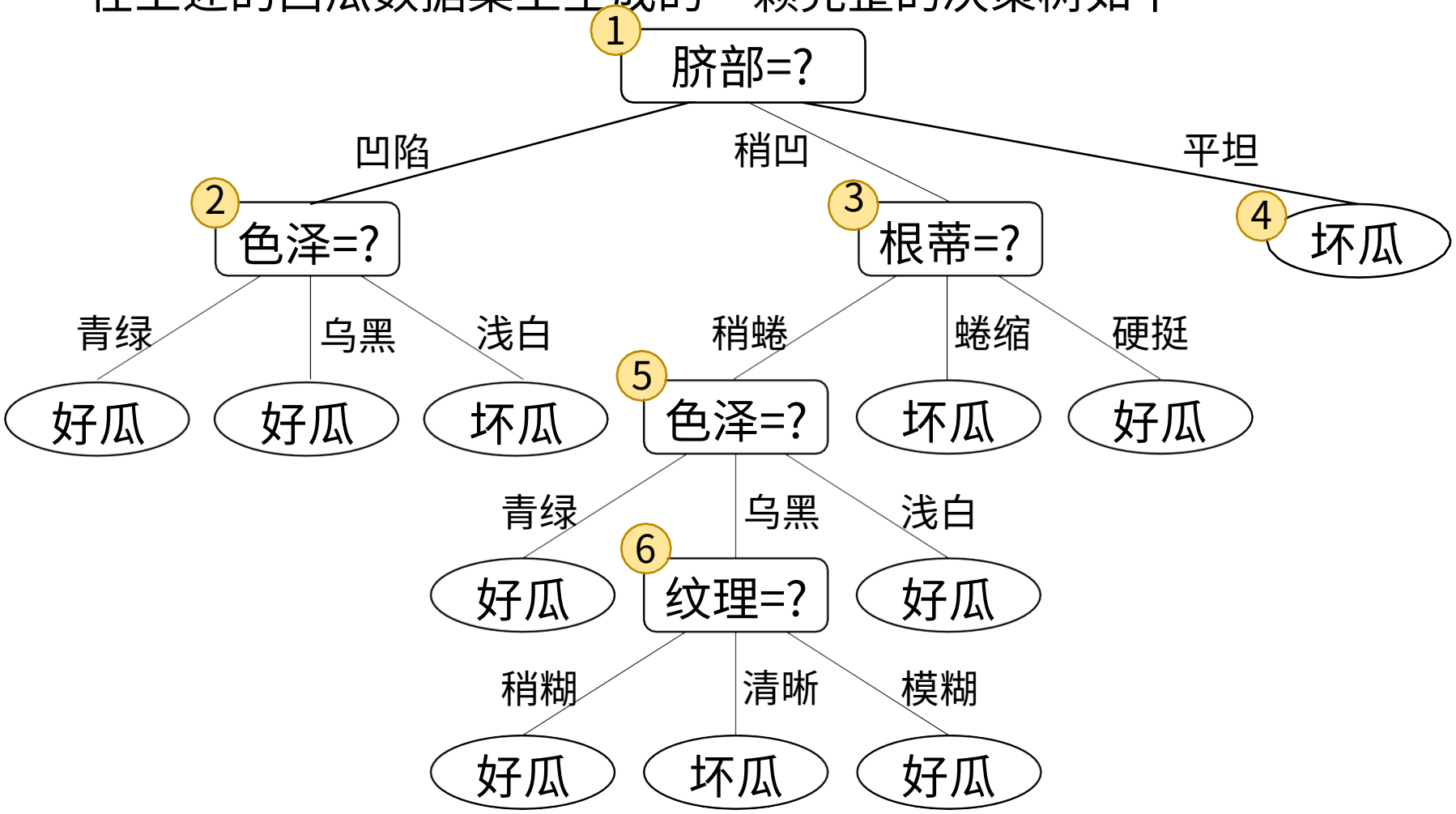
训练集

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
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

在上述的西瓜数据集上生成的一颗完整的决策树如下



预剪枝过程与示例

“预剪枝”过程如下

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

结点1：若不划分，则将其标记为 叶结点，类别标记为训练样例中最 多的类别，若选“好瓜”。验证集 中，{4,5,8}被分类正确，得到验 证集精度为 3/7x100%=42.9%

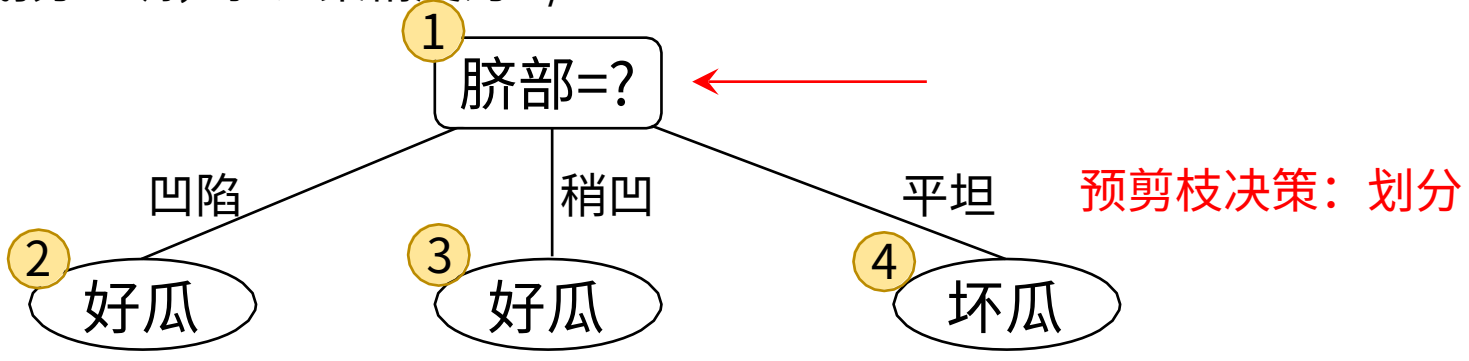
3.2 预剪枝过程与示例

“预剪枝”过程如下

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

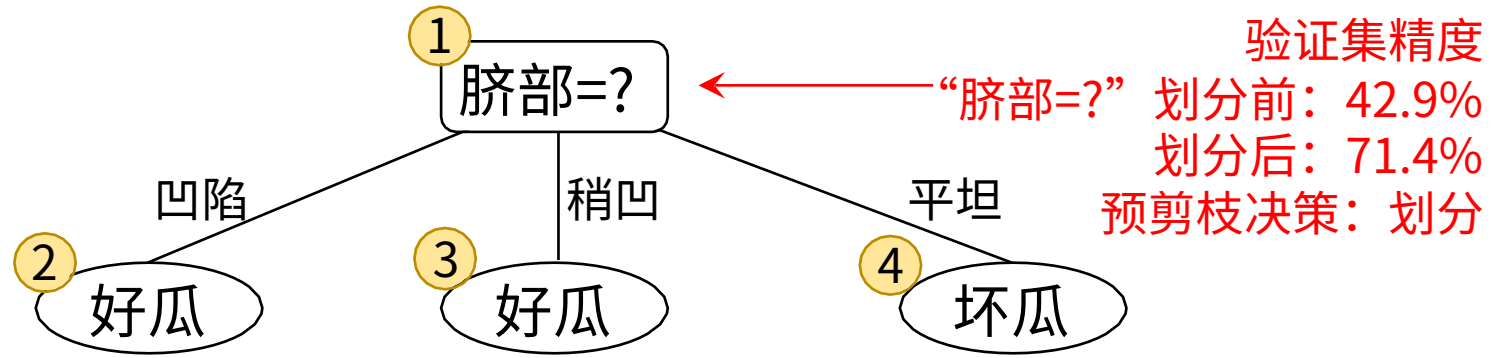
结点1：若划分，根据结点② ③ ④的训练样例，将这3 个结点分别标记为“好瓜”、“好瓜”、“坏瓜”。此时，验证集中编号为{4,5,8,11,12}的样例被划分正确，验证集精度为 5/7x100% = 71.4%



预剪枝决策：划分

“预剪枝”过程如下

对结点②③④分别进行剪枝判断，结点②③都禁止划分，结点④本身为叶子结点。最终得到仅有一层划分的决策树，称为“决策树桩”(decision stump)



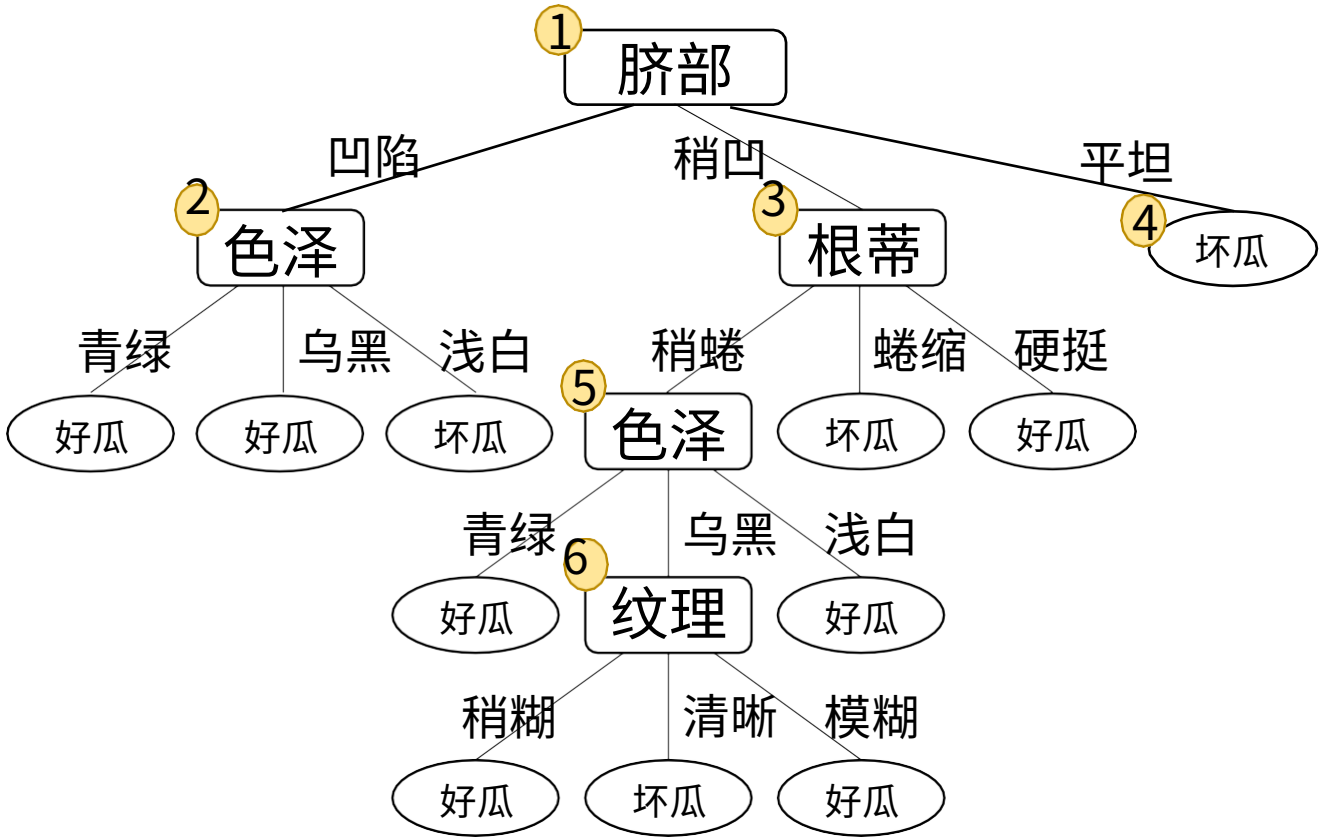
验证集精度
“脐部=?”划分前：42.9%
划分后：71.4%
预剪枝决策：划分

验证集精度
“色泽=?”划分前：71.4%
划分后：57.1%
预剪枝决策：禁止划分

验证集精度
“根蒂=?”划分前：71.4%
划分后：71.4%
预剪枝决策：禁止划分

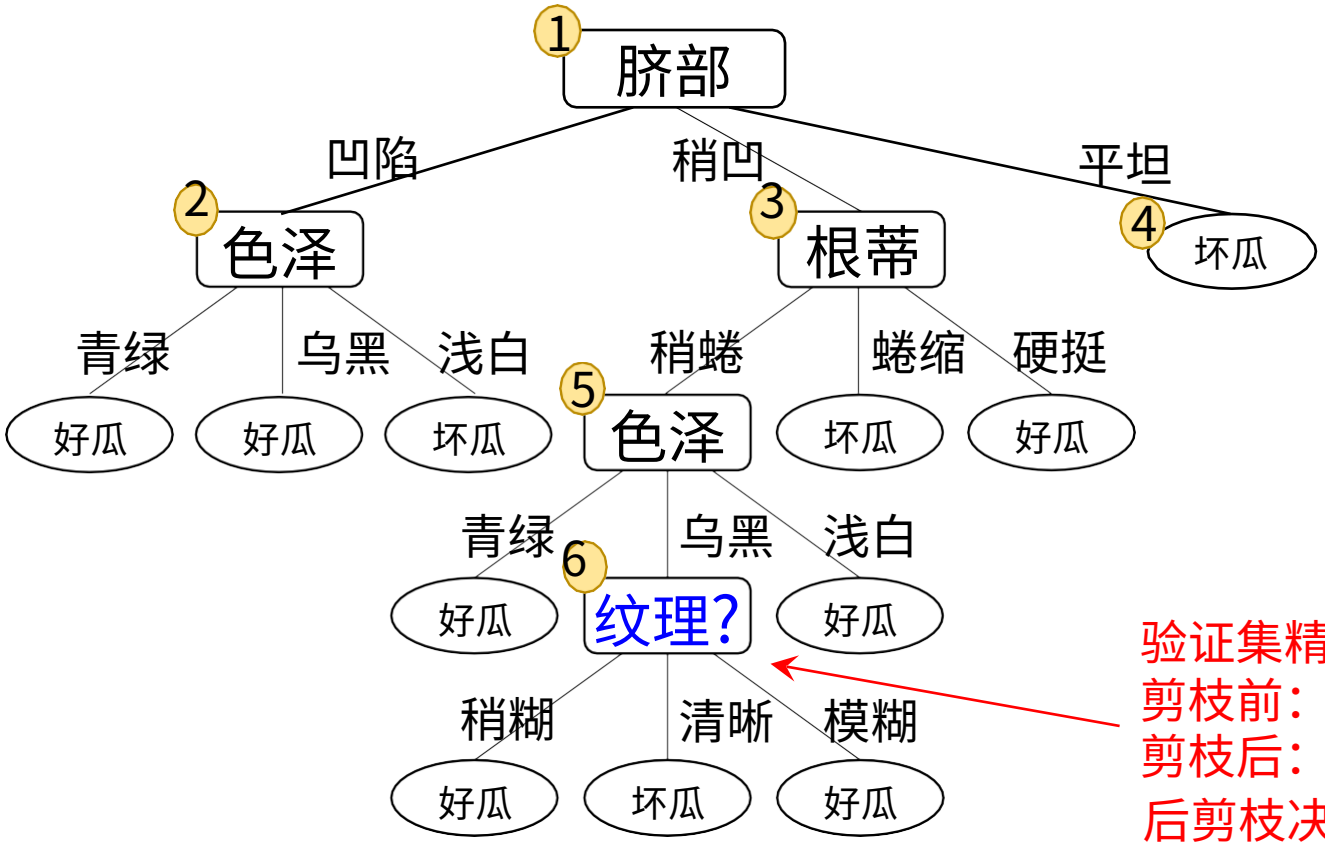
3.3 后剪枝过程与示例

我们在生成的完整决策树上进行“后剪枝”
完整决策树在验证集上准确率为42.9%



3.3 后剪枝过程与示例

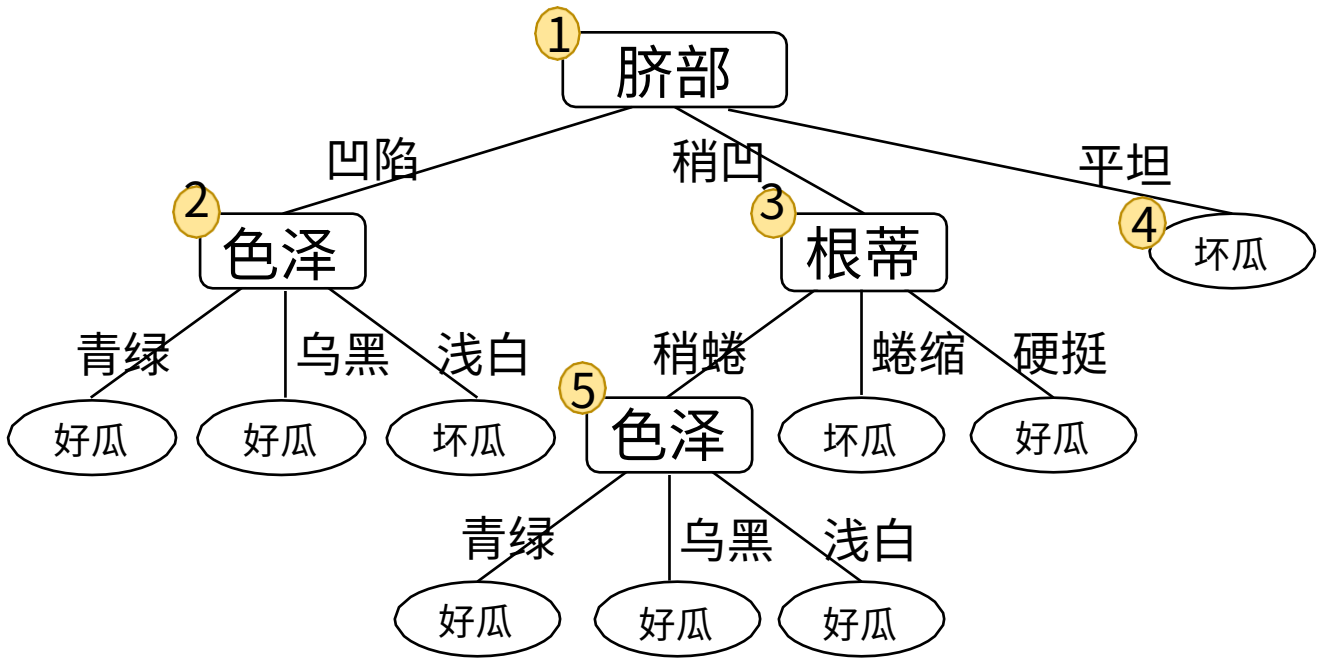
首先考虑结点⑥，若将其替换为叶结点，根据落在其上的训练样例{7,15}将其标记为“好瓜”，测得验证集精度提高至57.1%，于是决定剪枝



验证集精度
剪枝前：42.9%
剪枝后：57.1%
后剪枝决策：剪枝

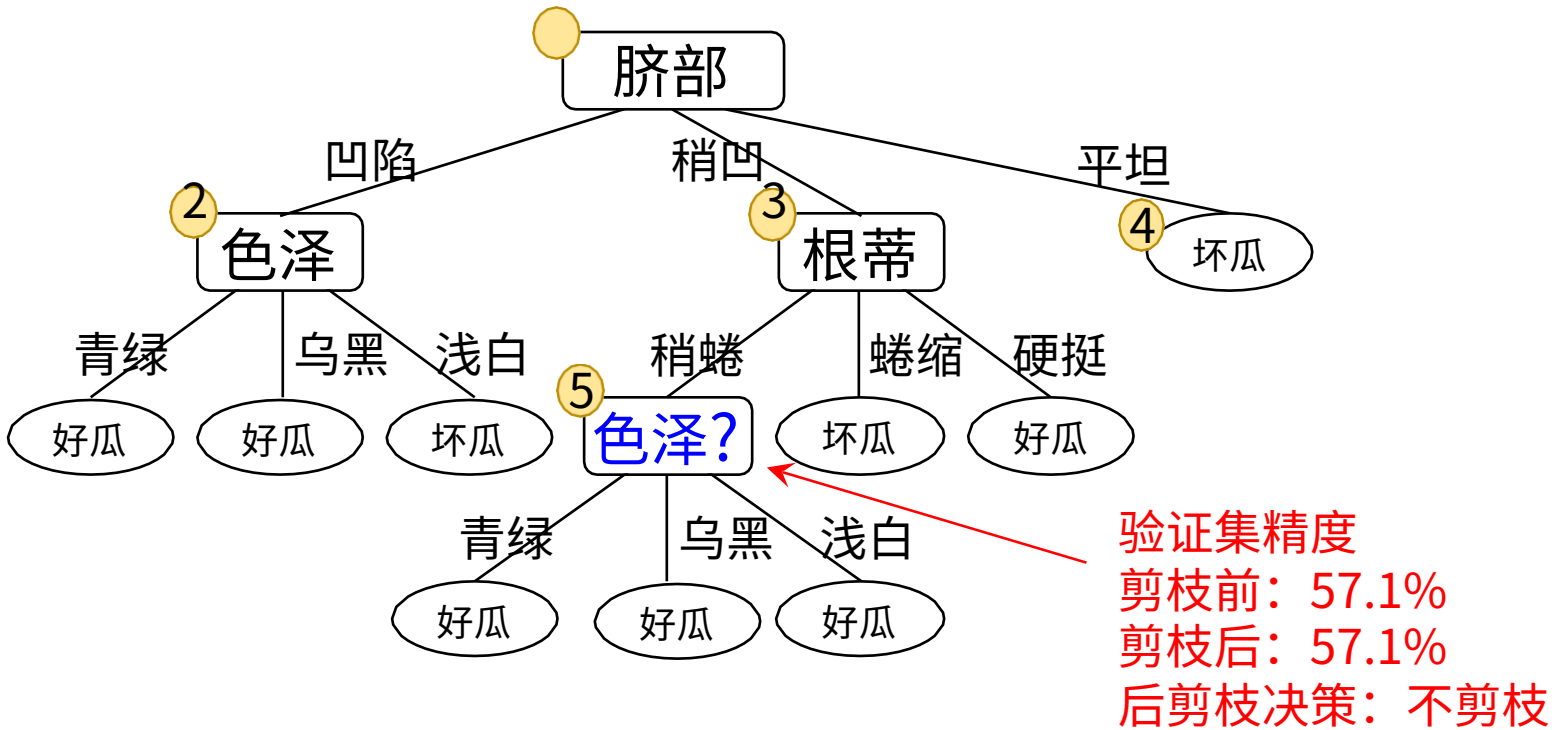
3.3 后剪枝过程与示例

首先考虑结点⑥，若将其替换为叶结点，根据落在其上的训练样例{7,15}将其标记为“好瓜”，测得验证集精度提高至57.1%，于是决定剪枝



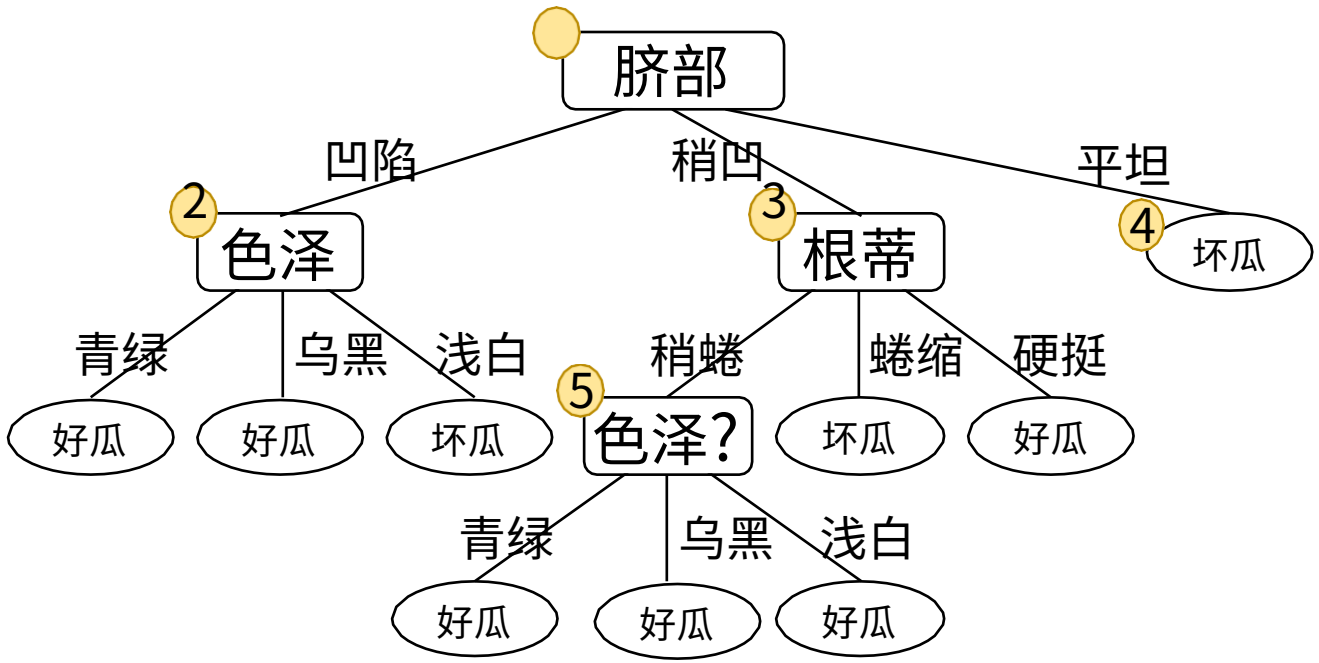
3.3 后剪枝过程与示例

首先考虑结点⑤，若将其替换为叶结点，根据落在其上的训练样例{6,7,15}将其标记为“好瓜”，测得验证集精度仍为57.1%，可以不剪枝



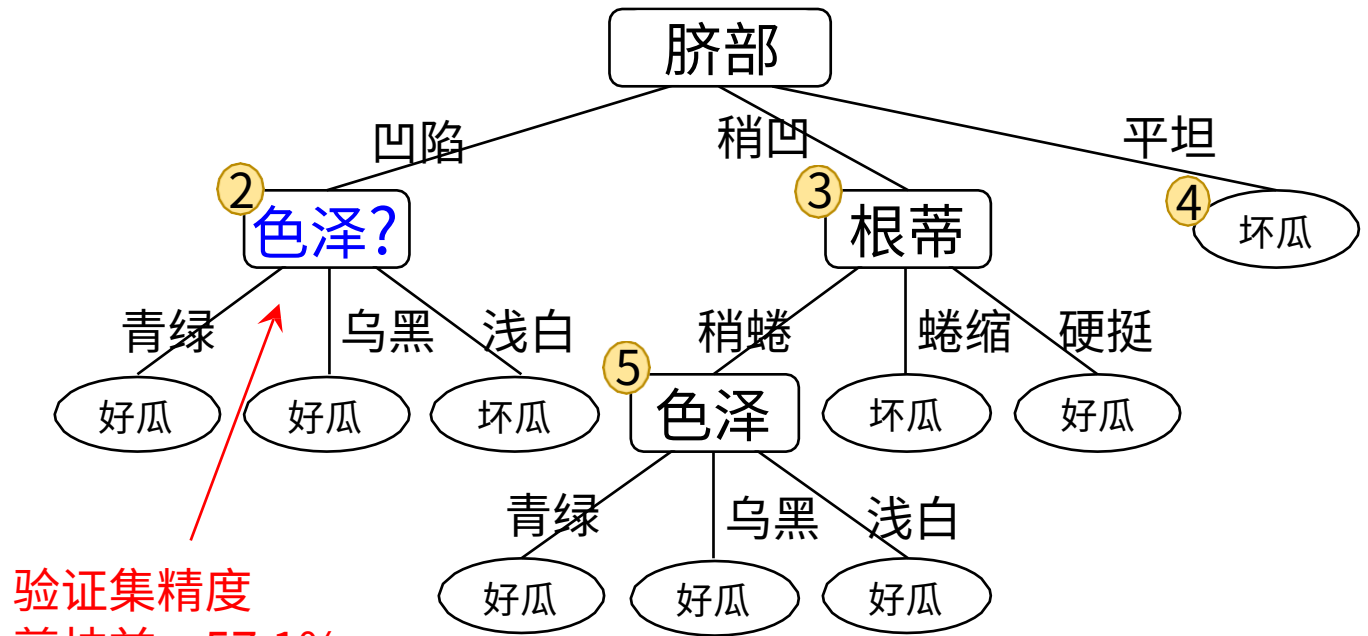
3.3 后剪枝过程与示例

首先考虑结点⑤，若将其替换为叶结点，根据落在其上的训练样例{6,7,15}将其标记为“好瓜”，测得验证集精度仍为57.1%，可以剪枝



3.3 后剪枝过程与示例

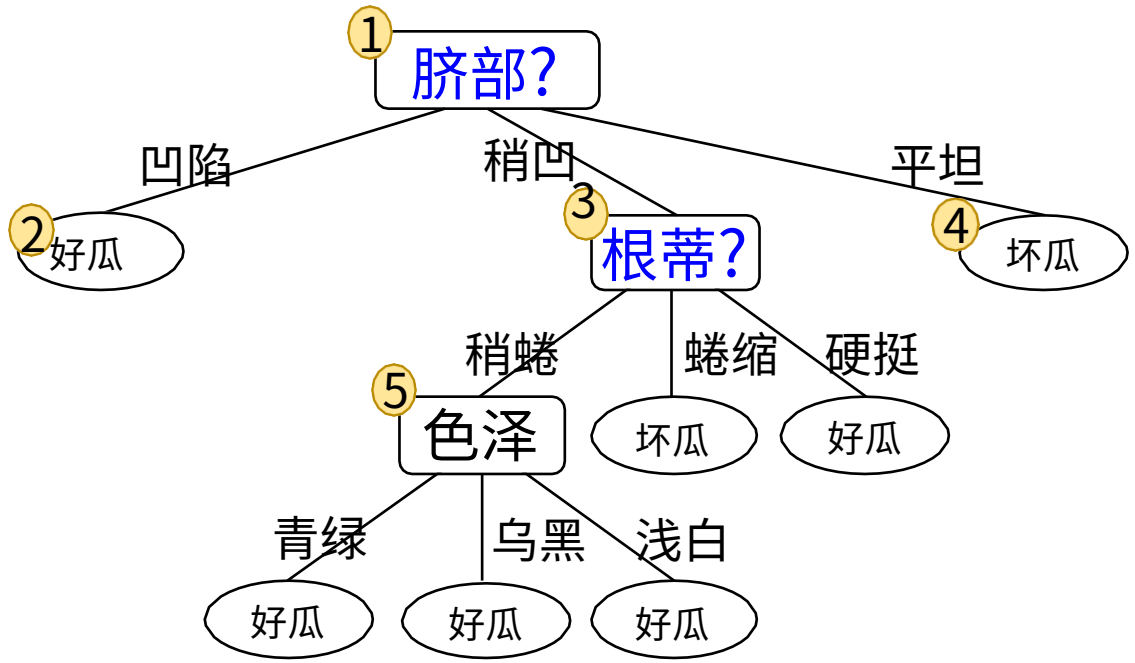
首先考虑结点②，若将其替换为叶结点，根据落在其上的训练样例{1,2,3,14}将其标记为“好瓜”，测得验证集精度提升至71.4%，决定剪枝



验证集精度
剪枝前：57.1%
剪枝后：71.4%
后剪枝决策：剪枝

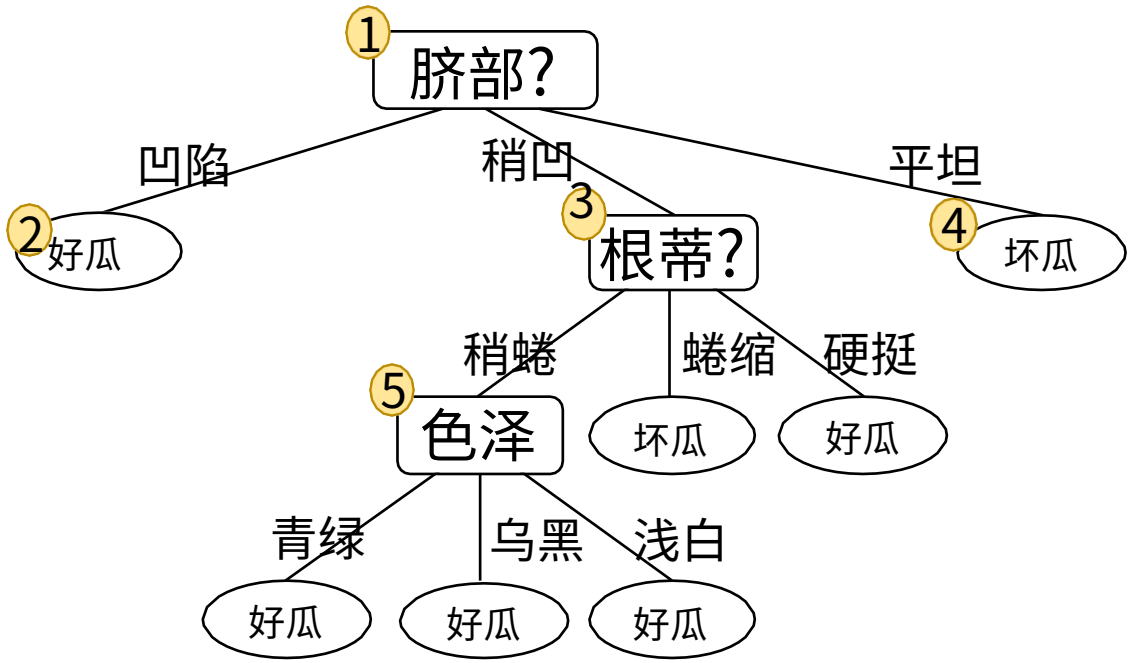
后剪枝过程与示例

对结点③和①，先后替换为叶结点，均未测得验证集精度提升，于是不剪枝



后剪枝过程与示例

得到最终后剪枝之后的决策树



3.3 预剪枝 vs 后剪枝

- 时间开销：
  - 预剪枝：训练时间开销降低，测试时间开销降低
  - 后剪枝：训练时间开销增加，测试时间开销降低

- 过/欠拟合风险：
  - 预剪枝：过拟合风险降低，欠拟合风险增加
  - 后剪枝：过拟合风险降低，欠拟合风险基本不变

- 泛化性能：后剪枝通常优于预剪枝

要点总结

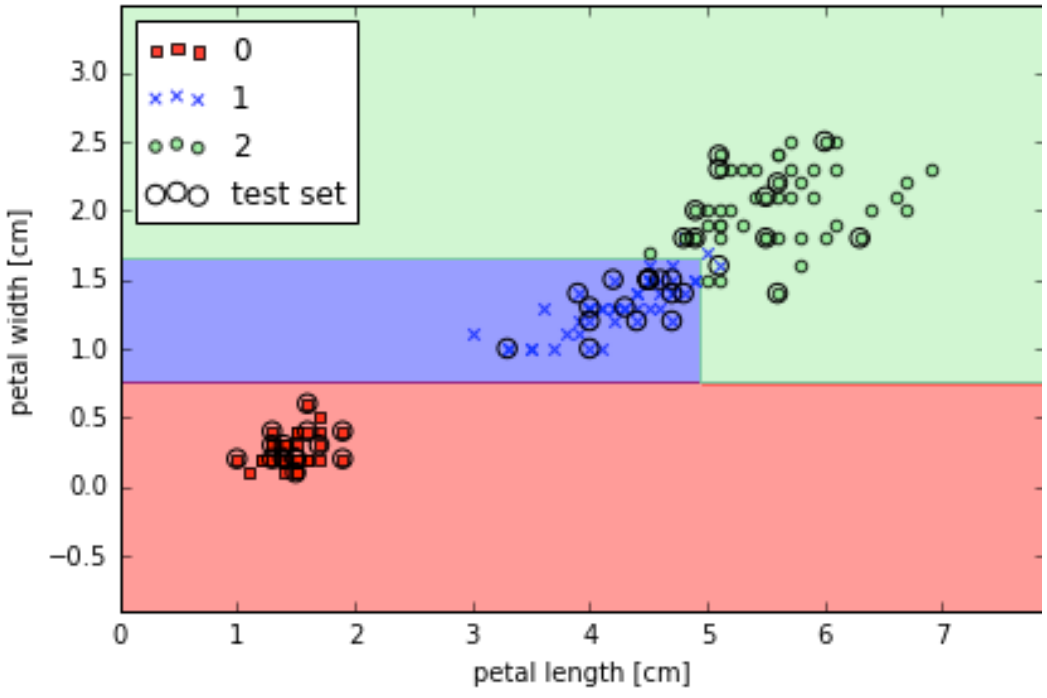
- 剪枝
  - 通过主动去掉一些分支来降低过拟合风险
- 预剪枝
  - 在决策树生成过程中,在划分节点时,若该节点的划分没有提高其在验证集上的准确率,则不进行划分
- 后剪枝
  - 后剪枝决策树先生成一棵完整的决策树，再从底往顶进行剪枝处理。

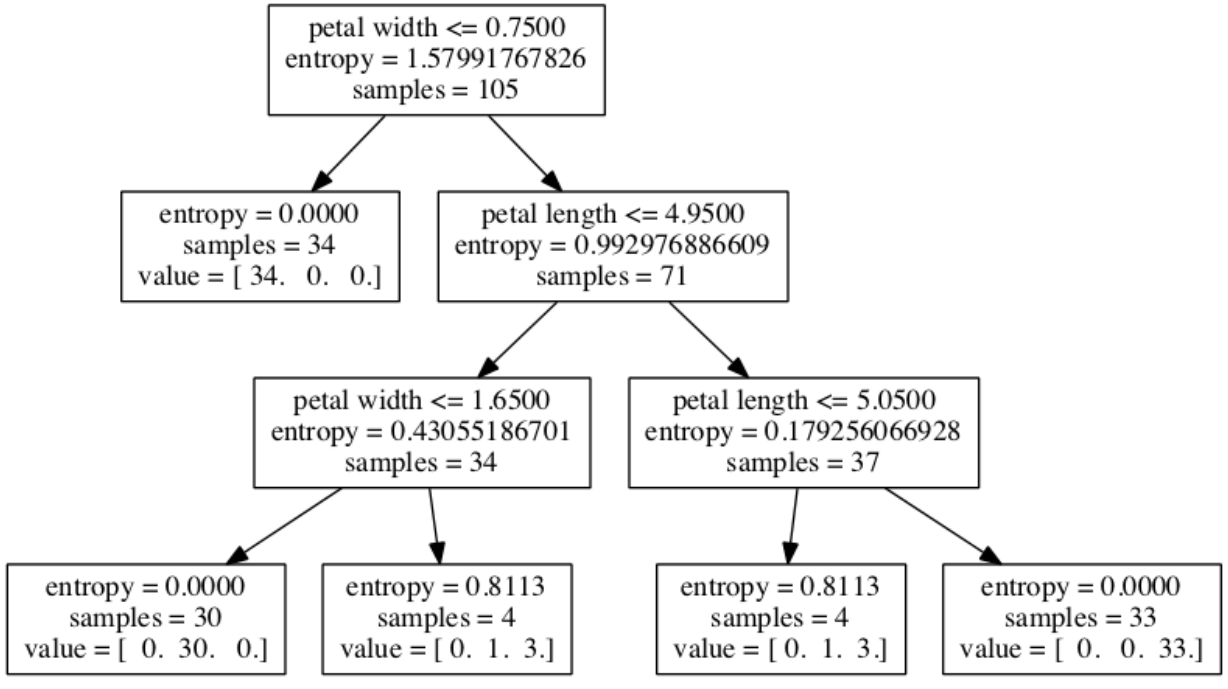
04 数据案例讲解

4.1 决策树完成分类与可视化

3.1 决策树分类与可视化

详见课程案例讲解





参考文献/Reference

- 周志华，机器学习，清华大学出版社，2016
- 李航，统计学习方法，清华大学出版社，2012
- Thomas M. Cover, Joy A. Thomas. Elements of Information Theory. 2006
- Christopher M. Bishop. Pattern Recognition and Machine Learning. Springer-Verlag. 2006
- Scikit-learn, http://scikit-learn.org/stable/index.html

THANK YOU !

Machine Learning Engineer
机器学习工程师微专业

稀牛学院 網易 NETEASE