安徽理工大学
ANHUI UNIVERSITY OF SCIENCE & TECHNOLOGY

• 人工智能专业 学科基础教育必修模块

2025

Python与机器学习

Python and Machine Learning

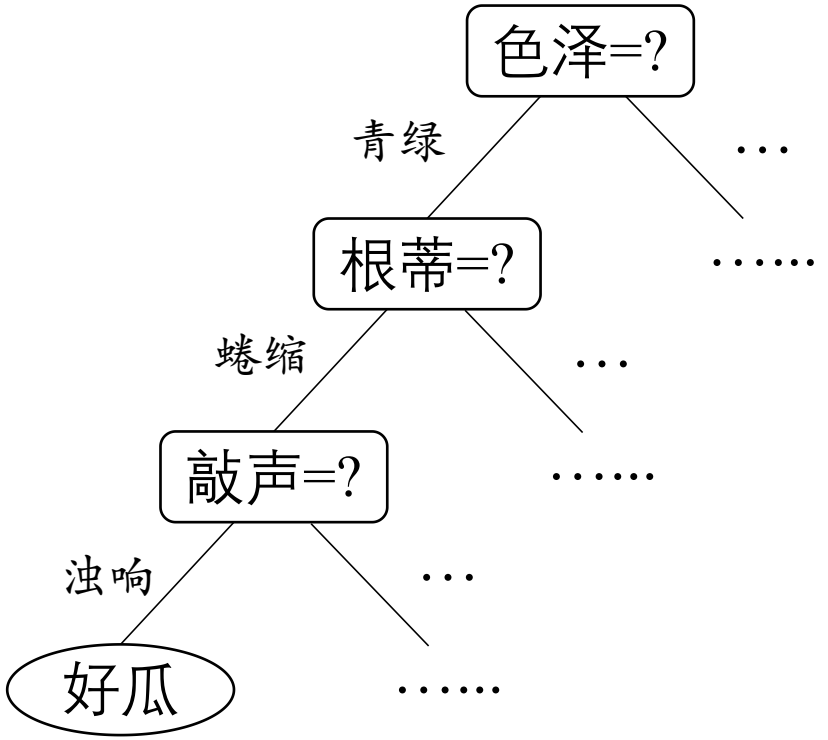
Chapter 9：
Decision Tree

• Lecturer ： Yuxian Liu（刘育仙）
• E-mail：yxl@aust.edu.cn

Chapter 9: Contents

9.1 Basic Process

- **The decision tree uses a tree structure to make predictions.**



- **Internal node: A "test" on a certain attribute**
- **Branch: A possible outcome of that test**
- **Leaf node: "Predicted result"**

- **Learning Process: Determine the "partitioning attribute" (i.e., the attribute corresponding to internal nodes) by analyzing training samples.**
- **Prediction Process: Starting from the root node, traverse the test example down the "decision sequence" formed by the partitioning attribute until reaching a leaf node.**

9.1 Basic Process

Principles of Decision Tree Model Construction:

1. Node Definition: Each internal node represents a test condition for a specific attribute.

2. Termination Condition: Leaf nodes correspond to target classification outcomes (desired decision outputs).

3. Recursive Mechanism: Test results trigger two branches—terminating the current path or generating a sub-decision problem, with subsequent tests constrained by parent node conditions.

4. Path Mapping: The complete path from root to leaf nodes represents a sequential attribute testing sequence.

9.1 Basic Process

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

递归返回，情形(1)

递归返回，情形(2)

利用当前结点的后验分布

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

递归返回，情形(3)

Decision Tree Termination Conditions:

① **Pure Node: Recursion stops when all samples within a node share the same category.**

② **Attribute Exhaustion: There are no available splitting attributes, or none of the attributes can provide an effective division.**

③ **Empty Node: No samples remain within the node to be processed.**

Chapter 9: Contents

9.2 Partitioning Selection

- **The key to decision tree learning lies in how to select the optimal splitting attribute. :**

As the partitioning process continues, we aim for the samples contained within each decision tree node to belong to the same category as much as possible, meaning the node's “purity” increases progressively.

Classic attribute partitioning methods:

Information gain

Gain rate

Gini index

9.2 Partitioning Selection - Information Gain

Information content(信息量)

The Concept of Information: Information is the reduction of uncertainty.

- **For example, "Tomorrow's temperature will drop by 8 degrees."**
- ✓ Eliminate uncertainty about tomorrow's weather changes.
- **The elimination of uncertainty is assessed based on people's prior knowledge.**

预言以往发生小概率的事件的消息所带来的信息量就要大

The probability of the event having occurred is called the prior probability, denoted by p.

信息量公式：$I(x) = \log\left(\frac{1}{p}\right) = -\log p$

9.2 Partitioning Selection - Information Gain

Information content(信息量)

Suppose the Chinese national football team and the Brazilian national football team have played eight matches, with China winning one.

Let U denote the event that China wins a future match between China and Brazil. The prior probability of U is 1/8, and its information content is:

$$I(U) = -\log_2 \frac{1}{8} = 3$$

If $\overline{U}$ denotes Brazil winning, then the prior probability of $\overline{U}$ is 7/8, and its information content is:

$$I(\overline{U}) = -\log_2 \frac{7}{8} = 0.19$$

9.2 Partitioning Selection - Information Gain

Information Entropy(信息熵)

The information content describes the uncertainty eliminated by a single event emitted from the information source, but it does not characterize the average uncertainty eliminated by the source.

If we take the average of the information content from all events emitted by the source, we can characterize the average uncertainty eliminated by the source, defined as information entropy:

$$H(X) = E[I(x_i)] = -\sum_{i=1}^{n} p_i \log_2 p_i$$

样本集合的信息熵越大，说明各样本相对均衡，

区别就越小，越不利于分类。

9.2 Partitioning Selection - Information Gain

- **Information entropy is the most commonly used metric for measuring the purity of a sample set. Given that the proportion of samples of category *k* in the current sample set *D* is denoted as $p_k(K = 1, 2, ..., |\mathcal{Y}|)$, the information entropy of *D* is defined as:**

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

The lower the $\mathrm{Ent}(D)$ value, the higher the purity of *D*.

The information entropy is computed as follows: if $p = 0$, then $p \log_2 p = 0$.

The minimum value of $\mathrm{Ent}(D)$ is 0, and the maximum value is $\log_2|\mathcal{Y}|$.

9.2 Partitioning Selection - Information Gain

Discrete Attribute Partitioning Mechanism and Information Gain Computation:

Partitioning Rule: Let discrete attribute *a* have *v* mutually exclusive values $\{a^1, a^2, ..., a^V\}$. Partitioning based on this attribute generates *v* subnodes.

Subset Definition: The *v*th subnode D^v contains all samples *a* in the original dataset *D* where a^v.

Information Gain:

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

The node weight is directly correlated with the number of samples it contains, and branches with larger samples have a higher impact on the overall model performance.

1. Metric: Information gain reflects an attribute's ability to enhance dataset purity; the larger the value, the more significant the splitting effect.

2. Algorithm Implementation: The ID3 decision tree [Quinlan, 1986] gives priority to splitting nodes based on attributes with the highest information gain by calculating the information gain value for each attribute.

9.2 Partitioning Selection - Information Gain

Example

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

This dataset contains 17 training samples with y=2, where positive examples account for p_1=8/17 and negative examples account for p_2=9/17. The information entropy of the root node is computed as:

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

9.2 Partitioning Selection - Information Gain

As an example, the attribute "color" corresponds to three data subsets: D^1 (color = bluish-green青绿), D^2 (color = jet black乌黑), and D^3 (color = pale white浅白).

The subset D^1 contains 6 samples numbered {1, 4, 6, 10, 13, 17}, where positive examples account for p_1=3/6 and negative examples account for p_2=3/6. Similarly for D2 and D3. The information entropy of the 3 nodes is:

$$\mathrm{Ent}(D^1) = -(\tfrac{3}{6}\log_2\tfrac{3}{6} + \tfrac{3}{6}\log_2\tfrac{3}{6}) = 1.000$$

$$\mathrm{Ent}(D^2) = -(\tfrac{4}{6}\log_2\tfrac{4}{6} + \tfrac{2}{6}\log_2\tfrac{2}{6}) = 0.918$$

$$\mathrm{Ent}(D^3) = -(\tfrac{1}{5}\log_2\tfrac{1}{5} + \tfrac{4}{5}\log_2\tfrac{4}{5}) = 0.722$$

The information gain for the attribute "color" is

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

9.2 Partitioning Selection - Information Gain

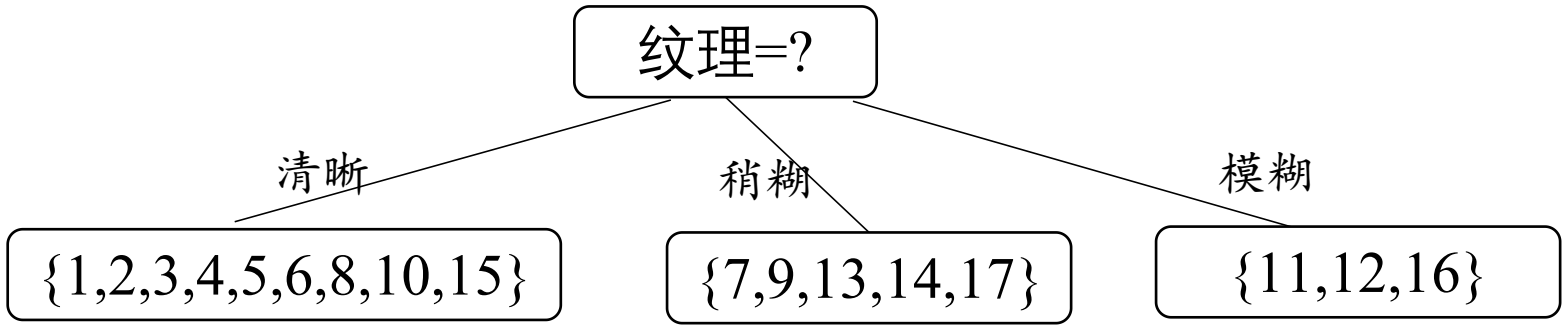
Similarly, the information gain for other attributes is

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

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

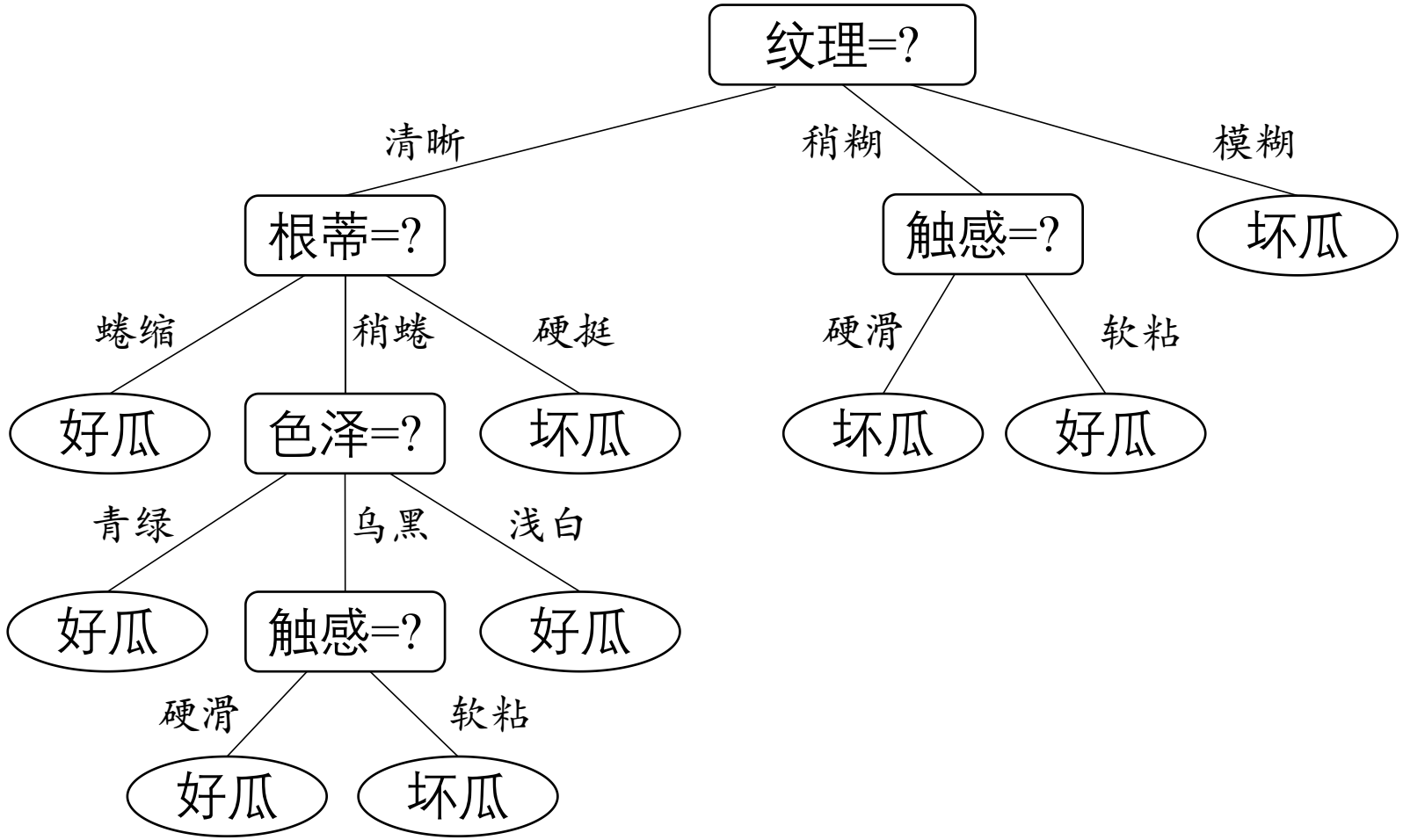
$$\text{Gain}(D, \text{触感}) = 0.006$$

Clearly, the attribute "texture" yields the greatest information gain and is selected as the partitioning attribute.



9.2 Partitioning Selection - Information Gain

The decision tree learning algorithm will perform further splits at each branch node, resulting in the final decision tree shown in the figure:



9.2 Partitioning Selection - Information Gain

Existing problems

If “ID” is also considered as a candidate splitting attribute, the information gain is generally much higher than that of other attributes. Clearly, such a decision tree lacks generalization capability and cannot effectively predict new samples.

Information gain exhibits a preference for attributes with a larger number of available values.

9.2 Partitioning Selection - Gain ratio(增益率)

Gain Rate Definition:

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

Among them

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

The "intrinsic value" of attribute *a* [Quinlan, 1993]. The greater the number of possible values for attribute *a* (i.e., the larger *V* is), the larger the value of IV(*a*) typically becomes.

Existing problems

The gain ratio criterion shows a preference for attributes with fewer possible values.

C4.5 [Quinlan, 1993] employs a heuristic: the attributes with information gains above the average are first identified from the candidate splitting attributes, and then the one with the highest gain rate is selected from them.

Practice

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	85	85	False	No
2	Sunny	80	90	True	No
3	Overcast	83	86	False	Yes
4	Rain	70	96	False	Yes
5	Rain	68	80	False	Yes
6	Rain	65	70	True	No
7	Overcast	64	65	True	Yes
8	Sunny	72	95	False	No
9	Sunny	69	70	False	Yes
10	Rain	75	80	False	Yes
11	Sunny	75	70	True	Yes
12	Overcast	72	90	True	Yes
13	Overcast	81	75	False	Yes
14	Rain	71	91	True	No

1、预处理数据：将属性temperature和humidity离散化，在此处的方法是将各个属性值从小到大排序，选取其每个相邻值的中点作为分裂点，计算其信息增益，选其最大值：

(1)、找到属性humidity的最大值为96和最小值65；

则区间[65,96]中取分段点，将要将属性值进行排序遍历到每一个可能的分裂点，然后找出信息增益最大的作为该属性的分段点。

Practice

68	[65,68]	1	1	0
	(68,96]	13	8	5
70	[65,70]	4	3	1
	(70,96]	10	6	4
78	[65,78]	5	4	1
	(78,96]	9	5	4
80	[65,80]	7	6	1
	(80,96]	7	3	4
86	[65,86]	9	7	2
	(86,96]	5	2	3
90	[65,90]	11	8	3
	(90,96]	3	1	2
91	[65,91]	12	8	4
	(65,96]	2	1	1
96	[65,96]	14	9	5
	(96,96]	0	0	0

计算出的相邻值的中值分别为：68，70，73，78，80，83，86，88，90，91，93，96。去掉相同的分布后，得左图。

分别计算把[min,A_i]和[A_i,max]作为该连续型属性变量的两类取值，分别命名为no1和no2。

Practice

68	[65,68]	1	1	0
	(68,96]	13	8	5

Step1、计算每个取值的信息熵：

68：

$$\text{entropy}\ (\text{humidity}=\text{no1}) = -(1/1)\log_2(1/1) - (0/1)\log_2(0/1) = 0$$

$$\text{entropy}\ (\text{humidity}=\text{no2}) = -(5/13)\log_2(5/13) - (8/13)\log_2(8/13) = 0.961$$

Practice

类似地，计算出其他情况下的信息熵。

70： entropy（humidity=no1）=-(1/4)log2(1/4)-(3/4)log2(3/4)=0.811

entropy（humidity=no2）=-(6/10)log2(6/10)-(4/10)log2(4/10)=0.971

78： entropy（humidity=no1）=-(4/5)log2(4/5)-(1/5)log2(1/5)=0.722

entropy（humidity=no2）=-(5/9)log2(5/9)-(4/9)log2(4/9)=0.991

80： entropy（humidity=no1）=-(6/7)log2(6/7)-(1/7)log2(1/7)=0.592

entropy（humidity=no2）=-(3/7)log2(3/7)-(4/7)log2(4/7)=0.987

Practice

86： entropy（humidity=no1）=-(2/5)log2(2/5)-(3/5)log2(3/5)=0.971

entropy（humidity=no2）=-(2/9)log2(2/9)-(7/9)log2(7/9)=0.764

90: entropy（humidity=no1）=-(8/11)log2(8/11)-(3/11)log2(3/11)=0.845

entropy（humidity=no2）=-(2/3)log2(2/3)-(1/3)log2(1/3)=0.918

91: entropy（humidity=no1）=-(8/12)log2(8/12)-(4/12)log2(4/12)=0.918

entropy（humidity=no2）=-(1/2)log2(1/2)-(1/2)log2(1/2)=1

96: entropy（humidity=no1）=-(5/14)log2(5/14)-(9/14)log2(9/14)=0.941

entropy（humidity=no2）不存在

Practice

68	[65,68]	1	1	0
	(68,96]	13	8	5

Step2、计算各个值的划分信息熵：

68：entropy（humidity）=13/14*0.961=0.893

类似地，计算出其他情况的划分信息熵。

70：entropy（humidity）=4/14*0.811+10/14*0.971=0.925

78：entropy（humidity）=5/14*0.722+9/14*0.991=0.895

80：entropy（humidity）=7/14*0.592+7/14*0.987=0.790

86：entropy（humidity）=5/14*0.971+9/14*0.764=0.838

90：entropy（humidity）=11/14*0.845+3/14*0.918=0.861

91：entropy（humidity）=12/14*0.918+2/14*1=0.930

Practice

Step3、计算各个值的信息增益：

68：Gain（humidity）=0.941−0.893=0.048

70：Gain（humidity）=0.941−0.925=0.016

78：Gain（humidity）=0.941−0.895=0.046

80：Gain（humidity）=0.941−0.790=0.151

86：Gain（humidity）=0.941−0.838=0.103

90：Gain（humidity）=0.941−0.861=0.080

91：Gain（humidity）=0.941−0.930=0.011

可知当分裂点为80时，信息增益率最大。

则求出对应的分裂信息熵与信息增益率：

splitE（humidity）$=-7/14\log_2(7/14)-(7/14)\log_2(7/14)=1$

Gain−Ratio（humidity）=0.151/1=0.151

Practice

利用同样的方法，计算出temperature取值为70时，信息增益率会最大。且其信息增益率为：

Gain-Ratio（temperature）=(0.941-0.895)/0.941=0.049

最终得到信息增益率：

Gain-Ratio（outlook）=0.248/1.578=0.157

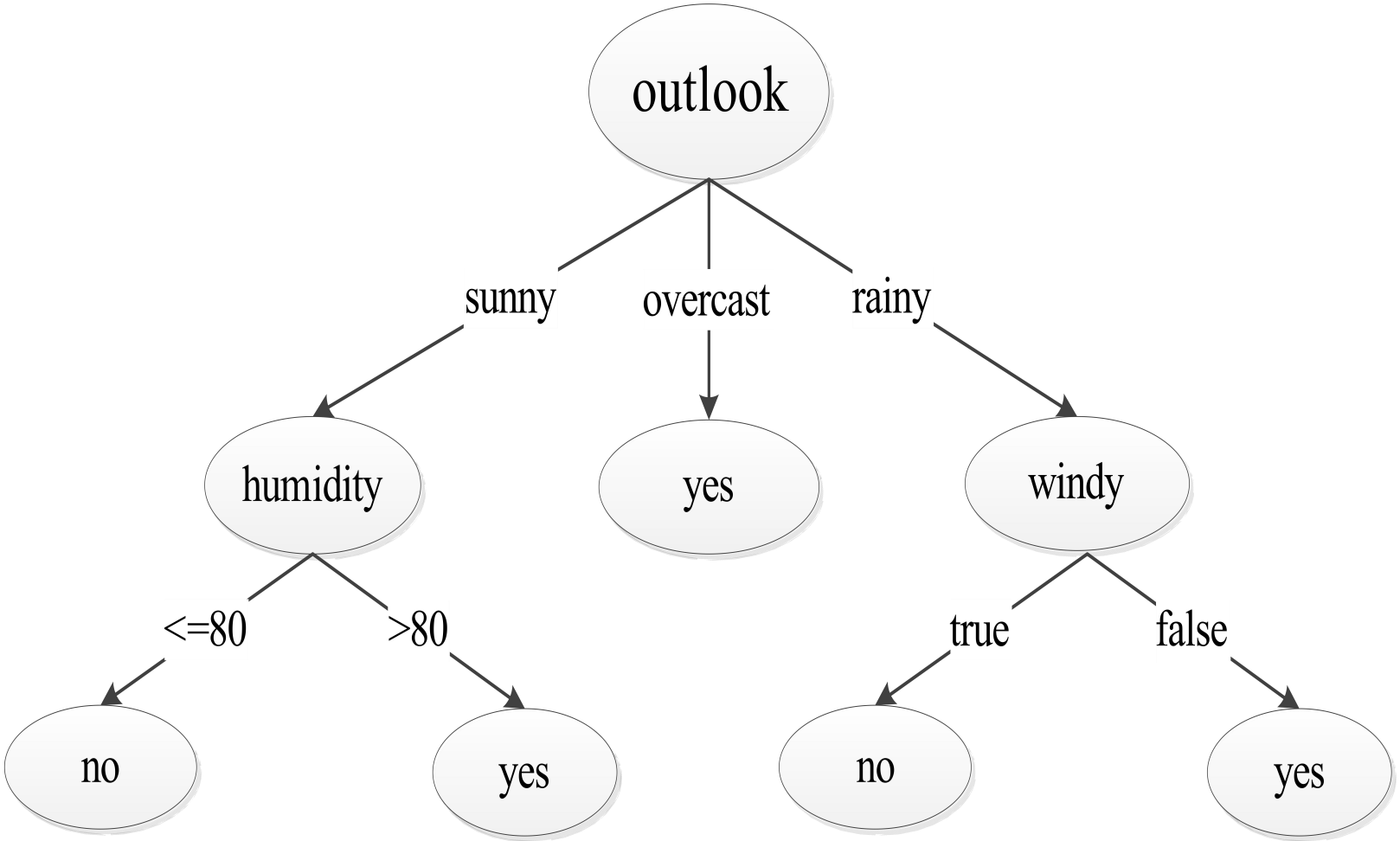
Gain-Ratio（windy）=0.049/0.985=0.050

Gain-Ratio（temperature）=(0.941-0.895)/0.941=0.049

Gain-Ratio（humidity）=(0.941-0.790)/1=0.151

可知应选择outlook作为根节点。然后仿照ID3的算法，继续进行分支，得到的树形结果如下：

Practice


outlook
sunny
overcast
rainy
humidity
yes
windy
<=80
>80
true
false
no
yes
no
yes

9.2 Partitioning Selection - Gini coefficient

The purity of dataset D can be measured using the Gini coefficient.

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

The smaller the Gini coefficient, the higher the purity of dataset D.

The Gini coefficient for attribute a is defined as:

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

It reflects the probability of randomly selecting two samples from D that have inconsistent category labels.

The attribute that minimizes the Gini coefficient after partitioning should be selected as the optimal partitioning attribute, i.e.,

$$a_* = \underset{a \in A}{\text{argmin}}\ \text{Gini\_index}(D, a)$$

CART [Breiman et al., 1984] employs the Gini index to select splitting attributes.

9.2 Partitioning Selection

- Research indicates that while various selection criteria significantly impact decision tree size, their influence on generalization performance is limited.
- For instance, the outcomes derived from information gain and Gini index differ in only about 2% of cases.
- Pruning methods and levels exert a more pronounced effect on decision tree generalization performance.
- When data contains noise, pruning may even enhance generalization performance by up to 25%.

Pruning(剪枝) is the primary means by which decision trees combat overfitting!

Chapter 9: Contents

9.3 Pruning

Why Prune:

1. Pruning is the primary method decision tree learning algorithms employ to combat overfitting.

2. Pruning can mitigate overfitting to some extent by preventing excessive decision branches from treating certain characteristics of the training set as universal properties shared by all data.

Basic Pruning Strategies： Pre-pruning, Post-pruning

Decision Tree Generalization Performance Evaluation Method (Leave-One-Out Method):

1. Core Principle: To achieve unbiased performance estimation by partitioning an independent validation set;

2. Implementation Approach: To partition the dataset into training and testing subsets, which are used as the validation set for quantitative model performance evaluation.

9.3 Pruning

- **Datatset**

Training set

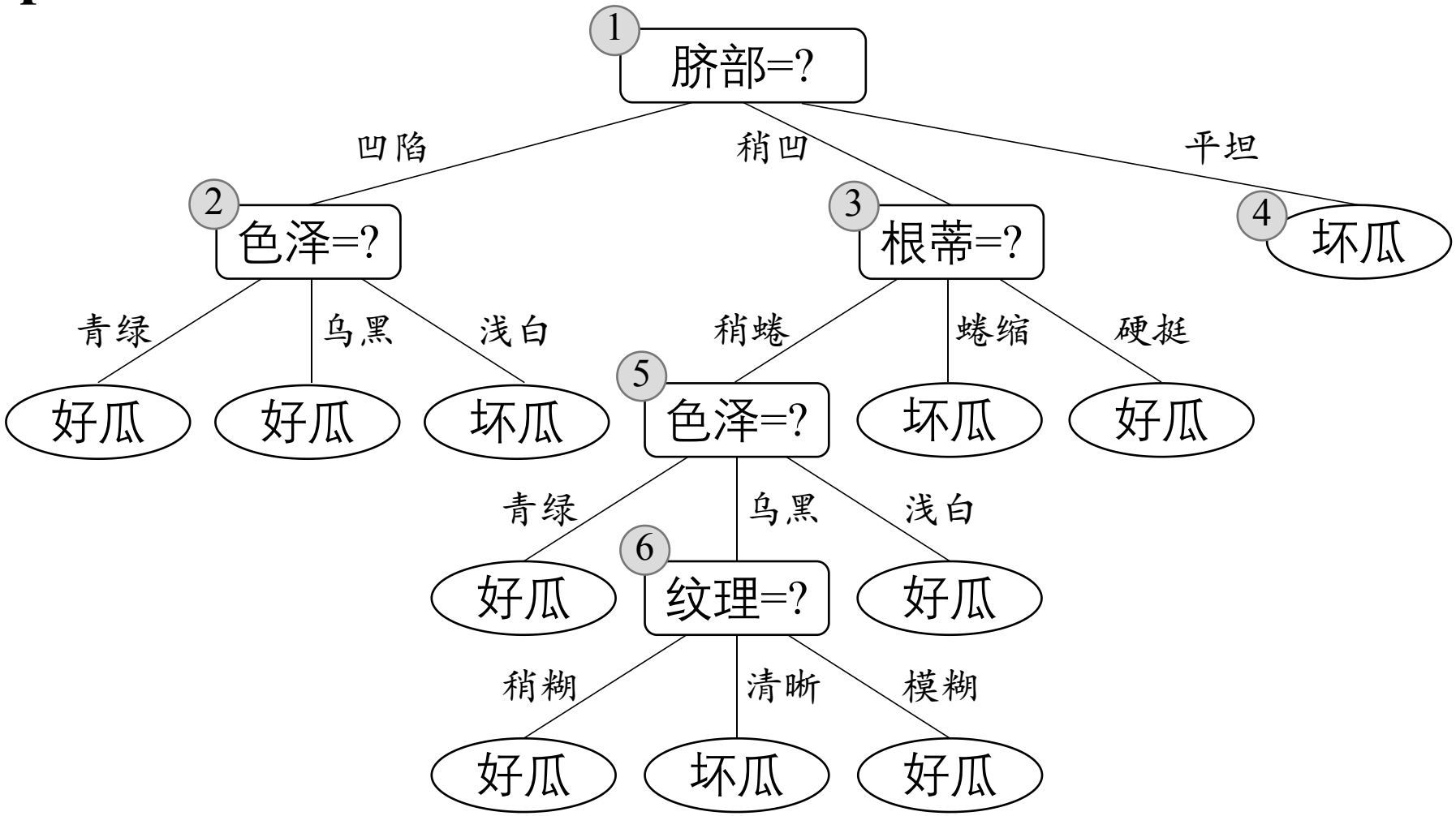
编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

Validation set

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

9.3 Pruning

- **Unpruned decision tree**



9.3 Pruning Treatment - Pre-pruning

- Pruning Mechanism Design: Evaluate a node's potential generalization gain before splitting. If it fails to improve overall performance then terminate the split, and mark the node as a leaf node with the label of the most dominant category in the training set.
- Attribute Selection Criteria: Based on the information gain criterion, the attribute “umbilicus” was selected to partition the training set.
- Dynamic Validation Process:

1. Benchmark Comparison: Calculate the validation set accuracy when keeping the current node as a leaf node.
2. Gain Evaluation: Compare validation accuracy before and after splitting. Execute splitting only if accuracy significantly improves.
3. Recursive Iteration: Repeat the above evaluation process for newly generated child nodes until termination conditions are met.

9.3 Pruning Treatment - Pre-pruning

Validation set

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，即好瓜。验证集中，$\{4, 5, 8\}$ 被分类正确，得到验证集精度为 $\frac{3}{7} \times 100\% = 42.9\%$

验证集精度

① 脐部=?

“脐部=?” 划分前: 42.9%

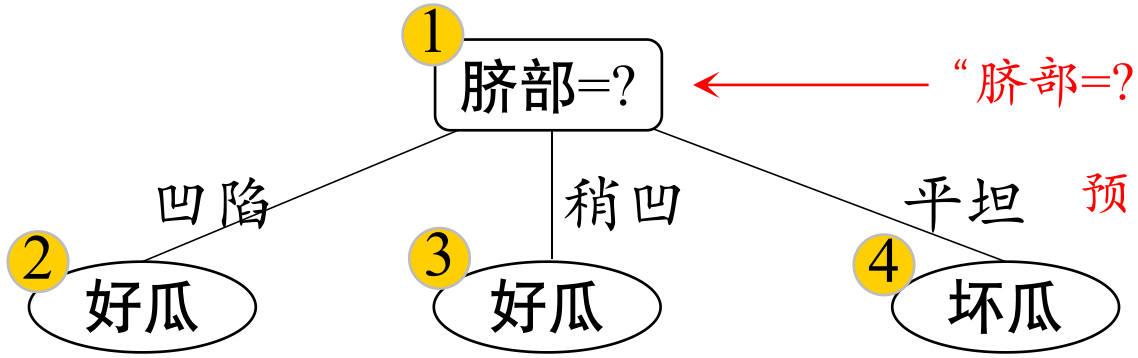
9.3 Pruning Treatment - Pre-pruning

Validation set

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 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

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



验证集精度

“脐部=?”划分前: 42.9%

划分后: 71.4%

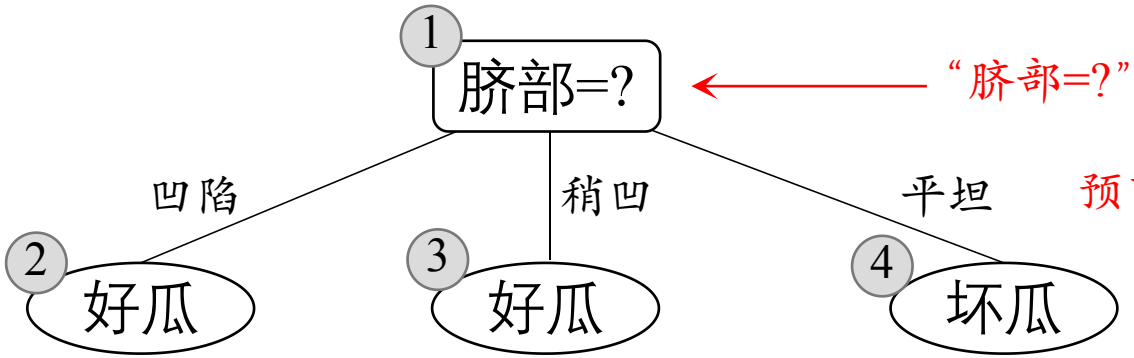
预剪枝决策: 划分

9.3 Pruning Treatment - Pre-pruning

Validation set

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

对结点②，③，④ 分别进行剪枝判断，结点② ③ 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**”



验证集精度

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

“色泽=?”

验证集精度

划分前: 71.4%
划分后: 57.1%
预剪枝决策: 禁止划分

“根蒂=?”

验证集精度

划分前: 71.4%
划分后: 71.4%
预剪枝决策: 禁止划分

9.3 Pruning Treatment - Pre-pruning

Advantages and Disadvantages of Pre-pruning

- **Advantages**

1. **Reduce the risk of overfitting**
2. **Significantly reduce training time and testing time overhead**

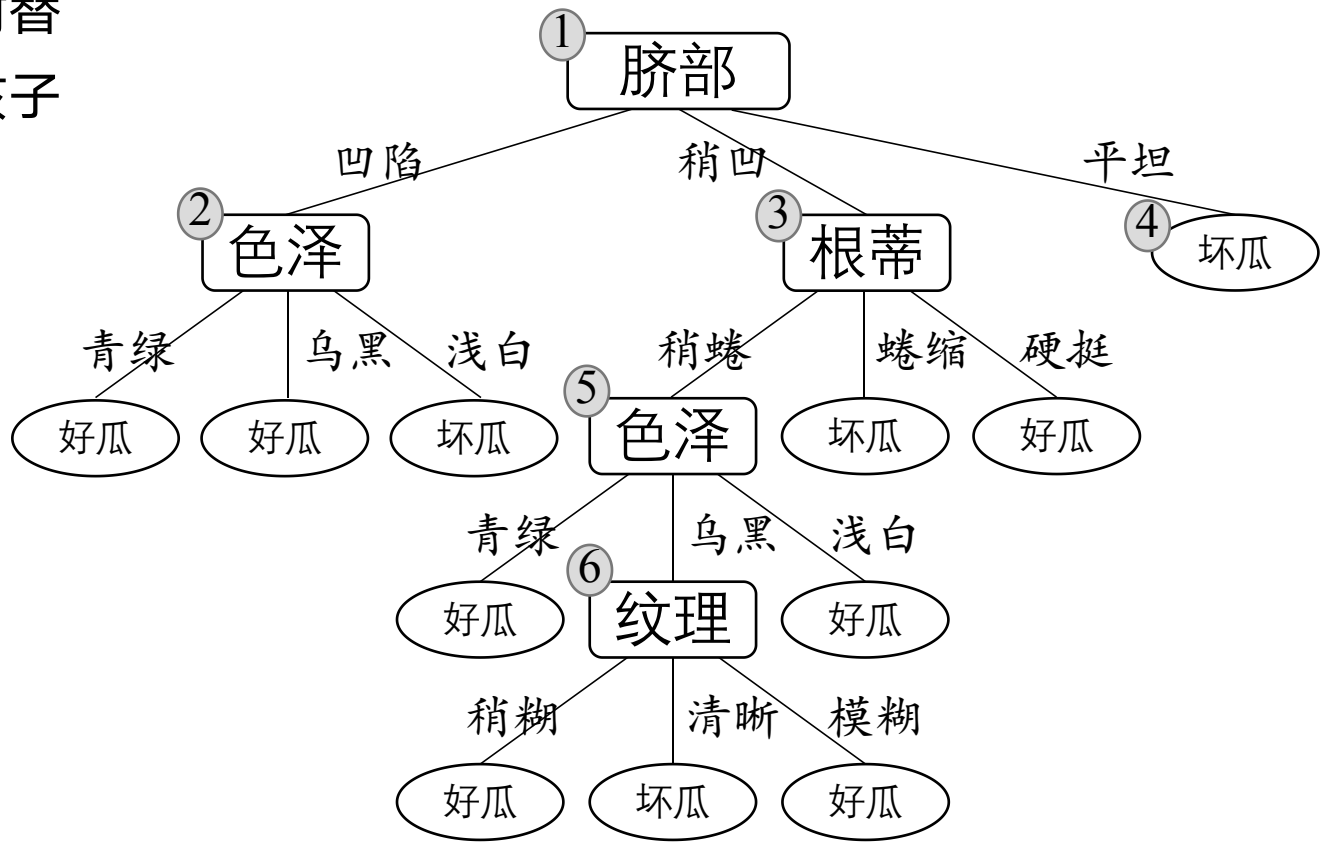
- **Disadvantages**

Risk Mechanisms of Underfitting Induced by Pre-Pruning:

1. **Potential Gain Suppression Phenomenon: Certain nodes may have significant optimization potential in the derived subtree, although splitting them currently yields no performance gains.**
2. **Limitations of Greedy Strategies: Pre-pruning algorithms employ a locally optimal decision-making approach, which prematurely terminates branch expansion with developmental potential, thereby restricting the model's expressive capability.**

9.3 Pruning Treatment - Post-pruning

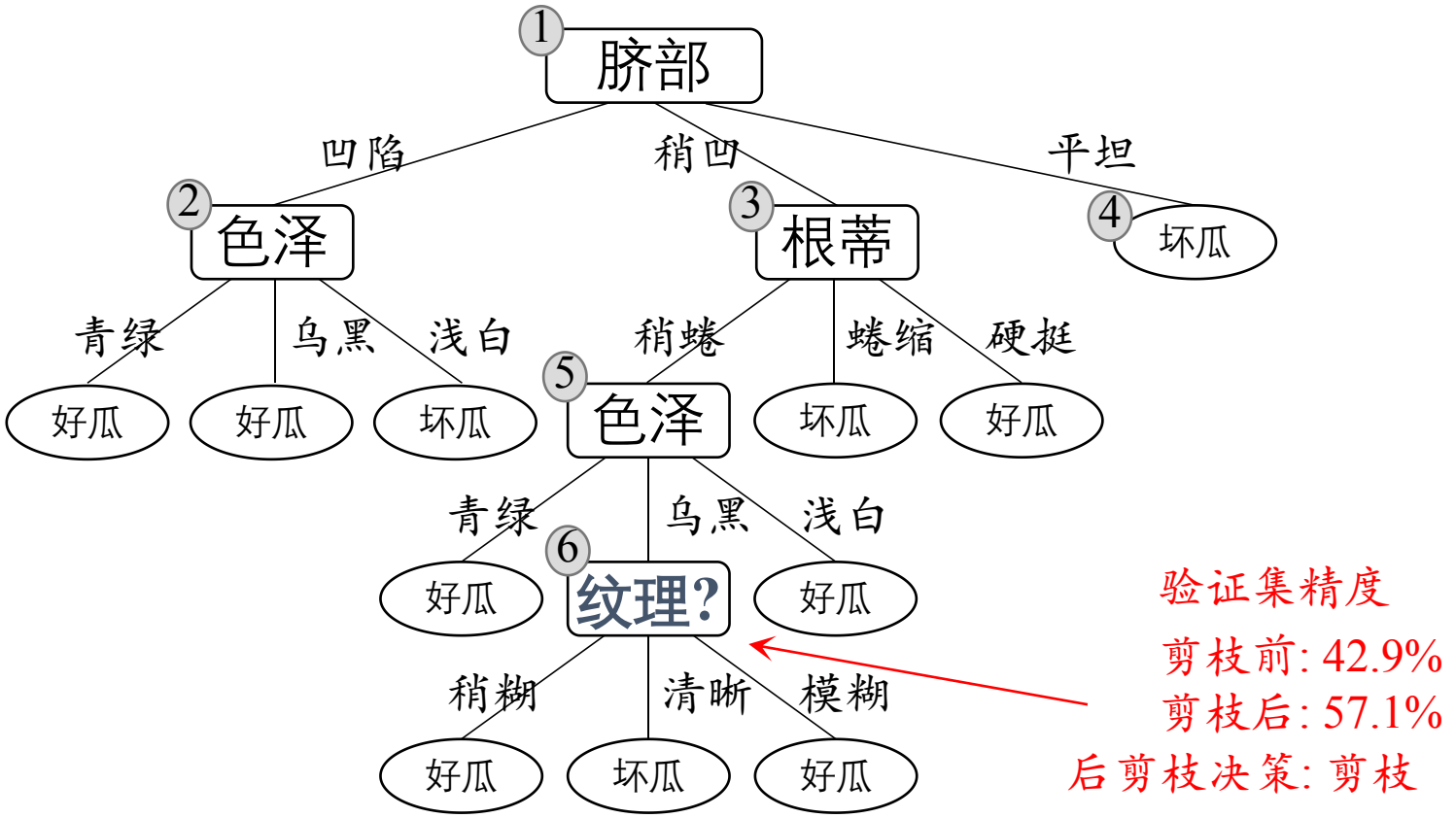
- 先从训练集生成一棵完整的决策树，然后自底向上地对非叶结点进行考察，若将该结点对应的子树替换为叶结点能带来决策树泛化性能提升，则将该子树替换为叶结点



First, generate a complete decision tree with a validation set accuracy of 42.9%.

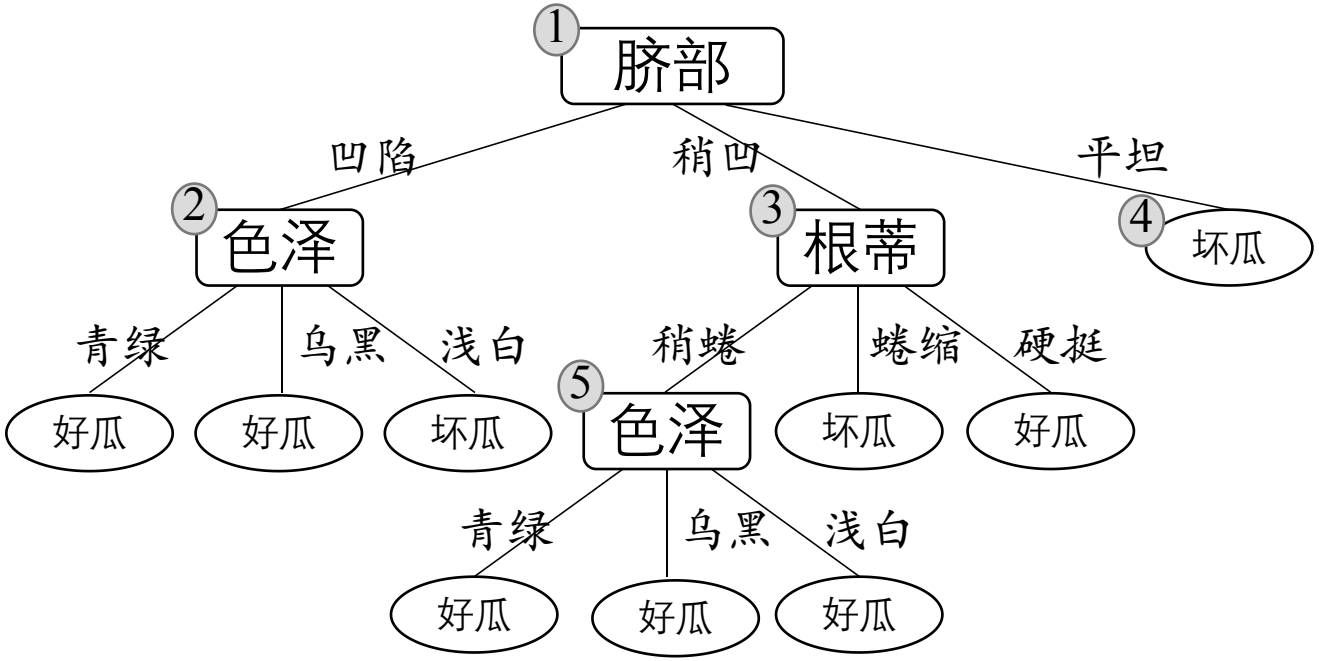
9.3 Pruning Treatment - Post-pruning

- 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本 $\{7, 15\}$ 将其标记为“好瓜”，得到验证集精度提高至 57.1%，则决定剪枝



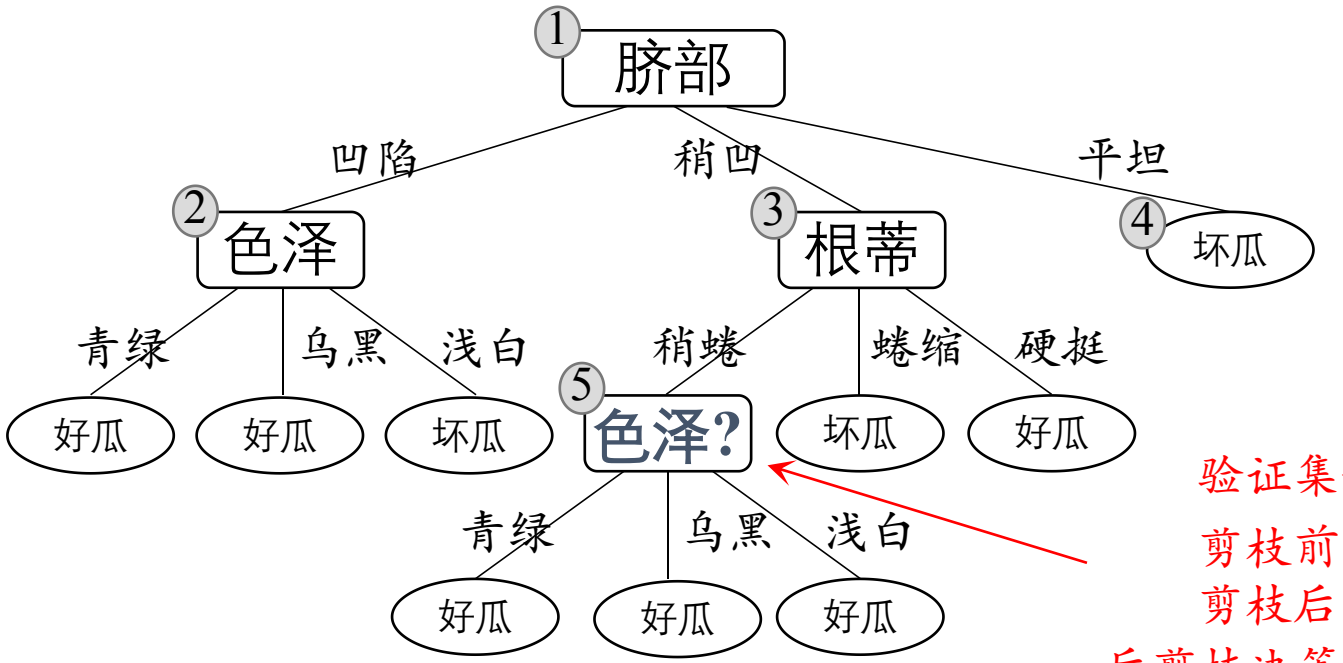
9.3 Pruning Treatment - Post-pruning

- 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本 $\{7, 15\}$ 将其标记为“好瓜”，得到验证集精度提高至 57.1%，则决定剪枝、



9.3 Pruning Treatment - Post-pruning

- 然后考虑结点 ⑤，若将其替换为叶结点，根据落在其上的训练样本 $\{6, 7, 15\}$ 将其标记为“好瓜”，得到验证集精度仍为 57.1% ，可以不进行剪枝



验证集精度

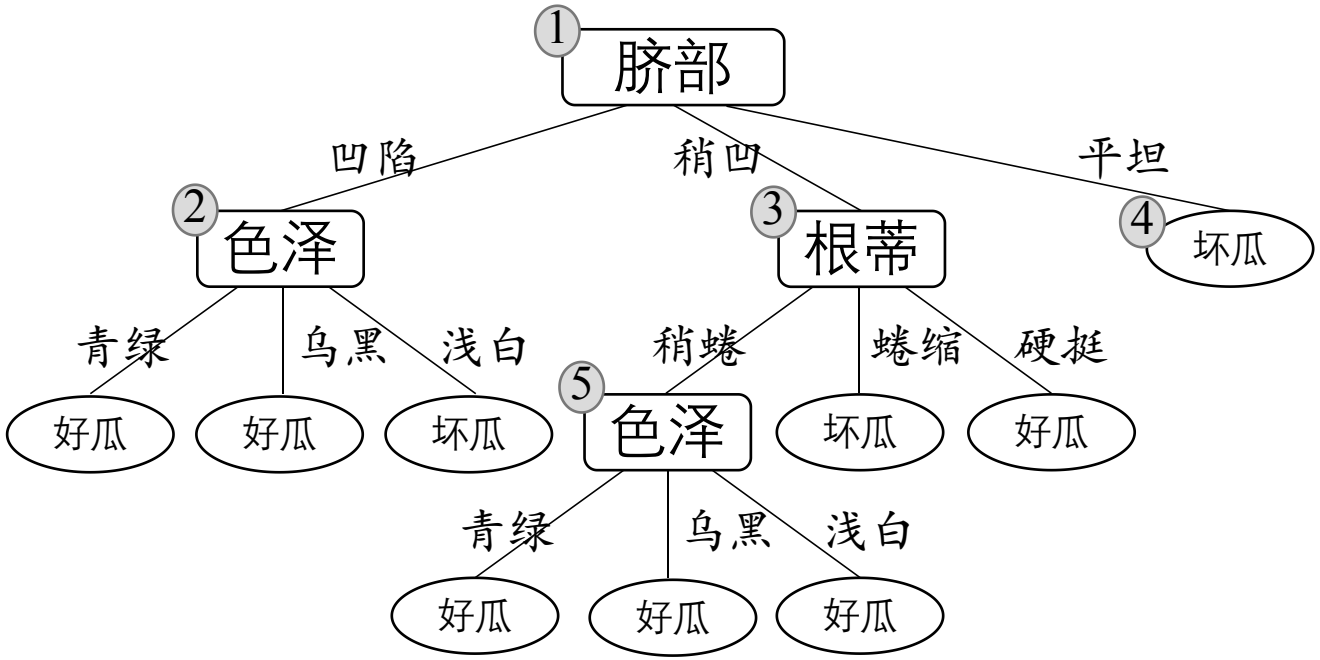
剪枝前: 57.1 %

剪枝后: 57.1%

后剪枝决策: 不剪枝

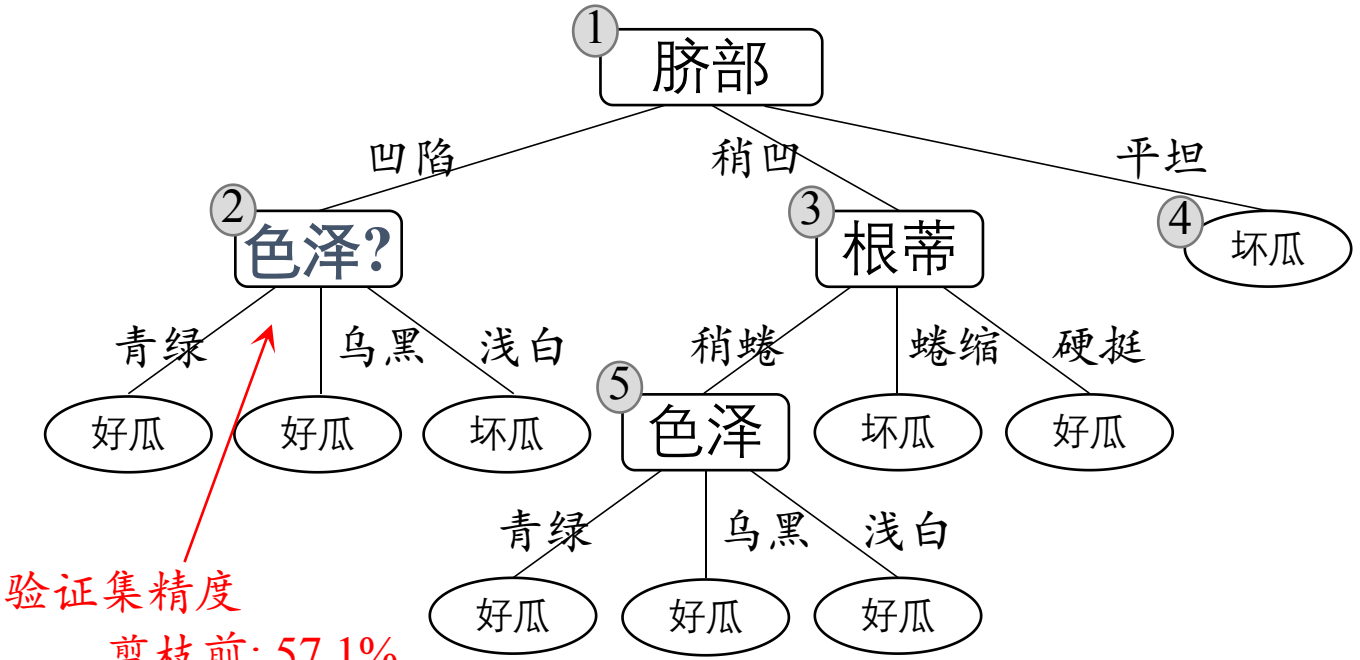
9.3 Pruning Treatment - Post-pruning

- 然后考虑结点 ⑤，若将其替换为叶结点，根据落在其上的训练样本 $\{6, 7, 15\}$ 将其标记为“好瓜”，得到验证集精度仍为 57.1% ，可以不进行剪枝



9.3 Pruning Treatment - Post-pruning

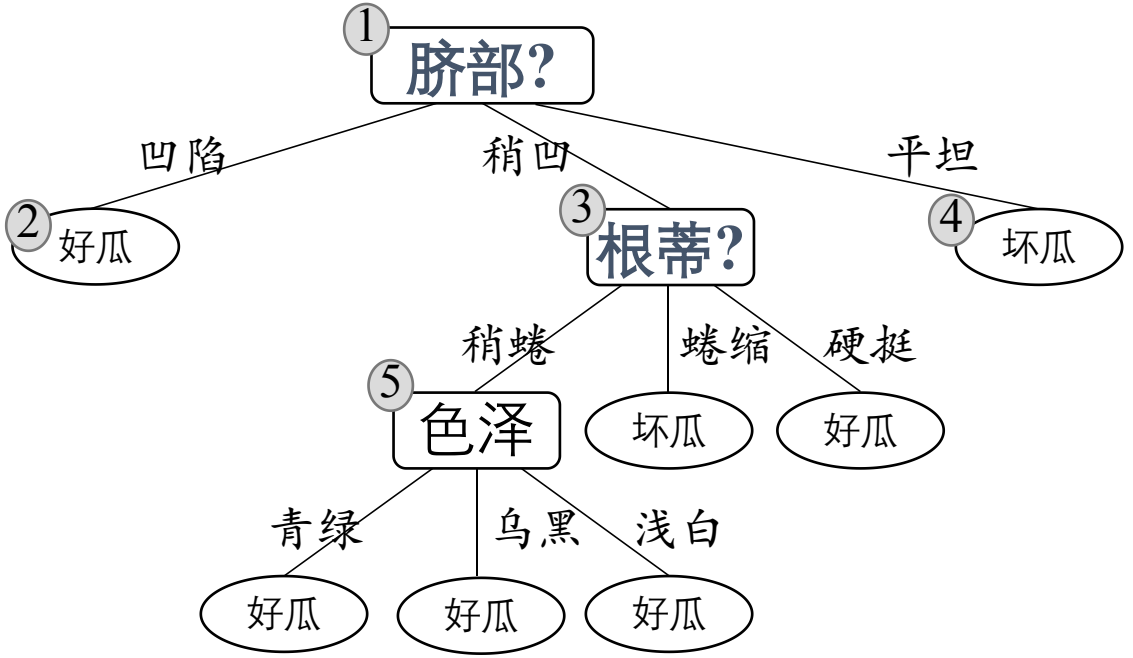
- 对结点 ②，若将其替换为叶结点，根据落在其上的训练样本$\{1, 2, 3, 14\}$，将其标记为“好瓜”，得到验证集精度提升至71.4%，则决定剪枝



9.3 Pruning Treatment - Post-pruning

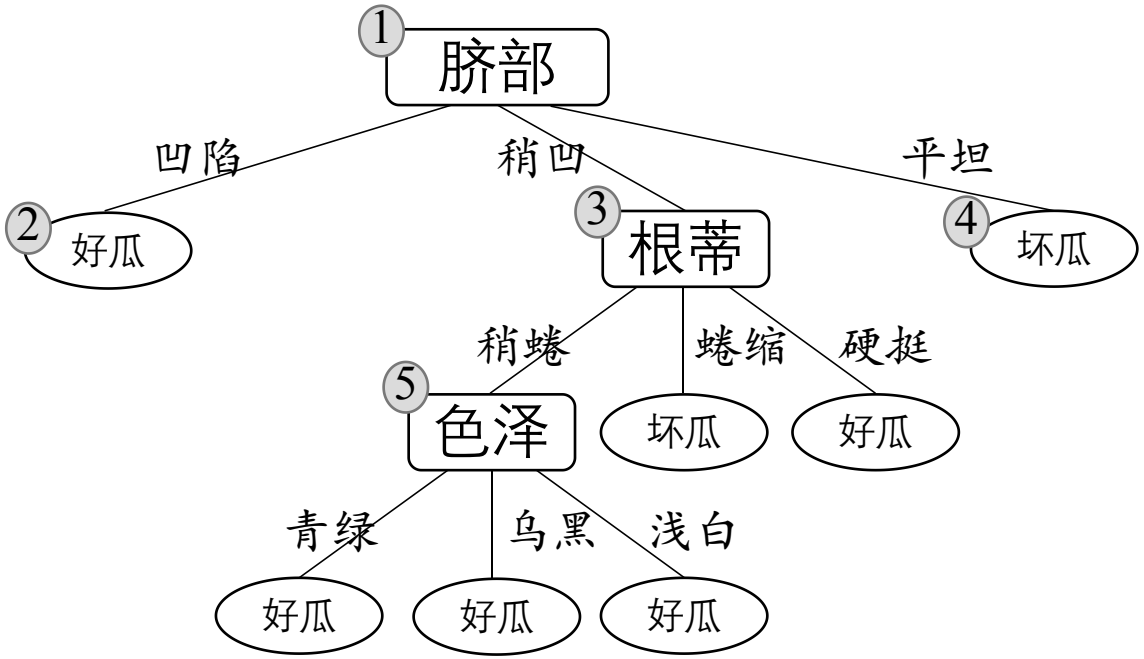
For nodes 3 and 1, which are replaced with leaf nodes in sequence, and none of the validation set accuracies improve, then the branch is retained.

- 对结点③和 ①，先后替换为叶结点，验证集精度均未提升，则分支得到保留



9.3 Pruning Treatment - Post-pruning

The final decision tree obtained using the post-pruning strategy is shown in the figure.



9.3 Pruning Treatment - Post-pruning

Advantages and Disadvantages of Post-pruning

- **Advantages**

  1. **Post-pruning has retained more branches than pre-pruning, there is less risk of underfitting, and the generalization performance is often superior to that of pre-pruning decision trees.**

- **Disadvantages**

  1. **High training time overhead: The post-pruning process is performed after generating the complete decision tree, which requires examining all non-leaf nodes one by one in a bottom-up manner.**

Chapter 9: Contents

Continuous and Missing Values – Continuous Value Handling

Discretization of Continuous Attributes (Binarization)

Step 1: Assume that the continuous attribute a takes *n* distinct values in the sample set *D*. Noted as a^1, a^2, ..., a^n in ascending order, *D* can be divided into subsets D_t^- and D_t^+ based on the division point *t*. D_t^- contains samples with attribute values no greater than *t*, while D_t^+ contains samples with attribute values greater than *t*. Consider the candidate division point set containing *n*-1 elements.

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

That is, take the midpoint $\frac{a^i + a^{i+1}}{2}$ of interval $[a^i, a^{i-1})$ as the candidate splitting point.

Continuous and Missing Values – Continuous Value Handling

Discretization of Continuous Attributes (Binarization)

Step 2: Employ the discrete attribute value method to evaluate these division points and select the optimal ones for dividing the sample set.

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

where $\mathrm{Gain}(D,a,t)$ **represents the information gain of sample set *D* after binary splitting at decision point *t*. Thus, the decision point that maximizes** $\mathrm{Gain}(D,a,t)$ **can be selected.**

Continuous and Missing Values – Continuous Value Handling

Examples of Continuous Value Handling

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

For the attribute "density," its candidate split point set contains 16 candidate values.

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

The information gain can be computed as 0.262, with the corresponding threshold set at 0.381.

The same is performed for the attribute "sugar content".

Unlike discrete attributes, if the current node's division attribute is a continuous attribute, this attribute can also serve as the division attribute for its descendant nodes.

Continuous and Missing Values – Continuous Value Handling

1. Incomplete samples, i.e., samples with missing attribute values
2. To learn only using samples without missing values?

A tremendous waste of data resources

The use of samples with missing values requires addressing the following issues:

Q1: How to perform attribute selection when attributes are missing?

Q2: Given a division attribute, how can the sample be classified if its value is missing?

Continuous and Missing Values – Continuous Value Handling

$\tilde{D}$ denotes the subset of samples in D with no missing values for attribute a. $\tilde{D}^v$ denotes the subset of samples in $\tilde{D}$ with attribute a taking the value a^v. $\tilde{D}_k$ denotes the subset of samples in $\tilde{D}$ belonging to class k.

Assign a weight w_x to each sample x, and define:

The proportion of samples with no missing values

$$\rho = \frac{\sum_{x \in \tilde{D}} w_x}{\sum_{x \in D} w_x}$$

Q1: How to perform attribute selection when attributes are missing?

The proportion of category k in the sample with no missing values

$$\tilde{p}_k = \frac{\sum_{x \in \tilde{D}_k} w_x}{\sum_{x \in \tilde{D}} w_x} \quad (1 \le k \le |\mathcal{Y}|)$$

The proportion of samples with missing values in the dataset that take the value a^v for attribute a.

$$\tilde{r}_v = \frac{\sum_{x \in \tilde{D}^v} w_x}{\sum_{x \in \tilde{D}} w_x} \quad (1 \le v \le V)$$

Continuous and Missing Values – Continuous Value Handling

Based on the above definition, we obtain

$$\begin{aligned}\mathrm{Gain}(D,a) &= \rho \times \mathrm{Gain}(\tilde{D},a) \\ &= \rho \times \left(\mathrm{Ent}(\tilde{D}) - \sum_{v=1}^{V} \tilde{r}_v \mathrm{Ent}(\tilde{D}^v)\right)\end{aligned} \qquad \textbf{where} \qquad \mathrm{Ent}(\tilde{D}) = -\sum_{k=1}^{|\mathcal{Y}|} \tilde{p}_k \log_2 \tilde{p}_k$$

For Q2

If the sample *x* has a known value for the division attribute *a*, then *x* is assigned to the subnode corresponding to its value, and the sample weight remains w_x within the subnode.

If the sample *x* has an unknown value for the dividing attribute *a*, then *x* is assigned to all subnodes simultaneously. The sample weight is adjusted to $\tilde{r}_v \cdot w_x$ in the subnode corresponding to the attribute value a^v. (Intuitively, this is equivalent to assigning the same sample to different subnodes with varying probabilities.)

Continuous and Missing Values – Continuous Value Handling

Example of Missing Value Handling

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

At the start of training, the root node contains all 17 examples from the sample set D, with each example having a weight of 1.

Taking the attribute "color" as an example, the subset $\tilde{D}$ of examples with no missing values for this attribute contains 14 examples. The information entropy of $\tilde{D}$ is

$$\mathrm{Ent}(\tilde{D}) = -\sum_{k=1}^{2} \tilde{p}_k \log_2 \tilde{p}_k$$

$$= -\left(\frac{6}{14}\log_2\frac{6}{14} + \frac{8}{14}\log_2\frac{8}{14}\right) = 0.985$$

Continuous and Missing Values – Continuous Value Handling

- Let $\tilde{D}^1$, $\tilde{D}^2$, and $\tilde{D}^3$ denote the sample subsets with values "cyan-green青绿," "jet-black乌黑," and "pale white浅白" respectively for the attribute "color," then

$$\mathrm{Ent}(\tilde{D}^1) = -\left(\tfrac{2}{4}\log_2\tfrac{2}{4} + \tfrac{2}{4}\log_2\tfrac{2}{4}\right) = 1.000 \quad \mathrm{Ent}(\tilde{D}^2) = -\left(\tfrac{4}{6}\log_2\tfrac{4}{6} + \tfrac{2}{6}\log_2\tfrac{2}{6}\right) = 0.918$$

$$\mathrm{Ent}(\tilde{D}^3) = -\left(\tfrac{0}{4}\log_2\tfrac{0}{4} + \tfrac{4}{4}\log_2\tfrac{4}{4}\right) = 0.000$$

Therefore, the information gain for the attribute "color" on the sample subset $\tilde{D}$ is

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

Thus, the information gain for the attribute "color" on sample set D is

$$\mathrm{Gain}(D, 色泽) = \rho \times \mathrm{Gain}(\tilde{D}, 色泽) = \tfrac{14}{17} \times 0.306 = 0.252$$

Continuous and Missing Values – Continuous Value Handling

Similarly, the information gain for all attributes across the dataset can be computed.

$$\text{Gain}(D, \text{色泽}) = 0.252 \qquad \text{Gain}(D, \text{根蒂}) = 0.171$$

$$\text{Gain}(D, \text{敲声}) = 0.145 \qquad \text{Gain}(D, \text{纹理}) = 0.424$$

$$\text{Gain}(D, \text{脐部}) = 0.289 \qquad \text{Gain}(D, \text{触感}) = 0.006$$

进入“纹理=清晰”分支

进入“纹理=稍糊”分支

进入“纹理=模糊”分支

样本权重在各子结点仍为1

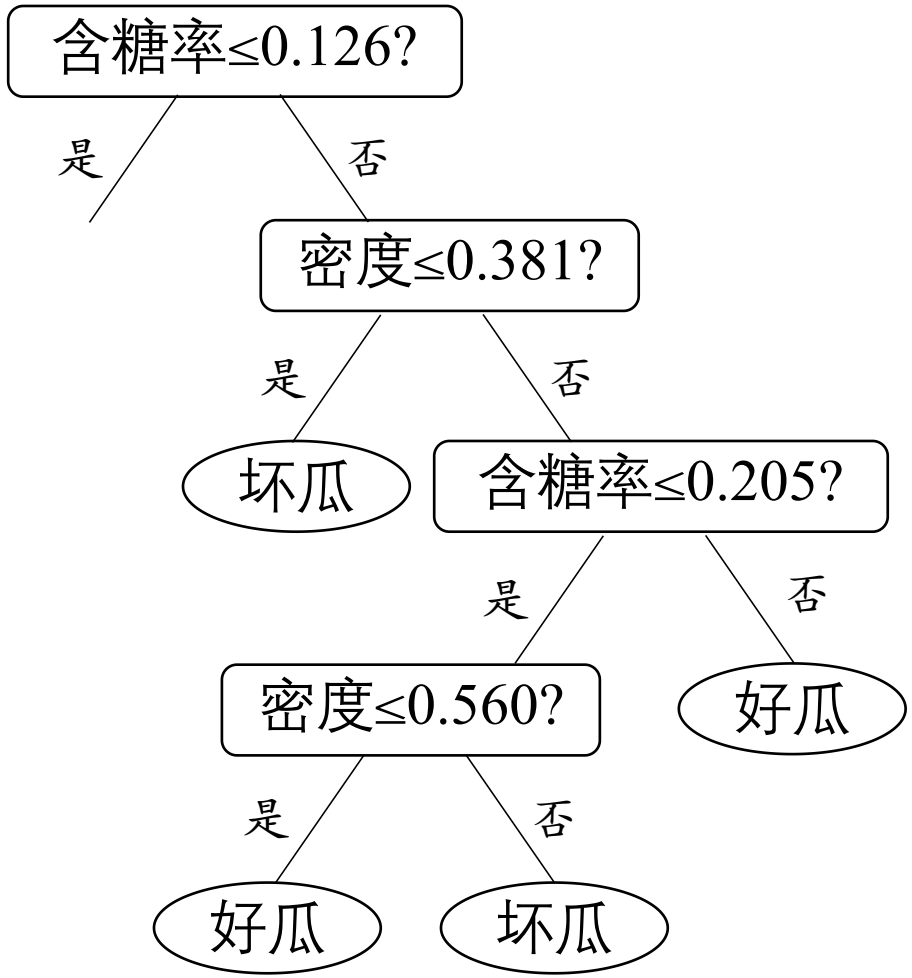
在属性“纹理”上出现缺失值，样本8和10同时进入3个分支，调整8和10在3分分支权值分别为7/15，5/15，3/15

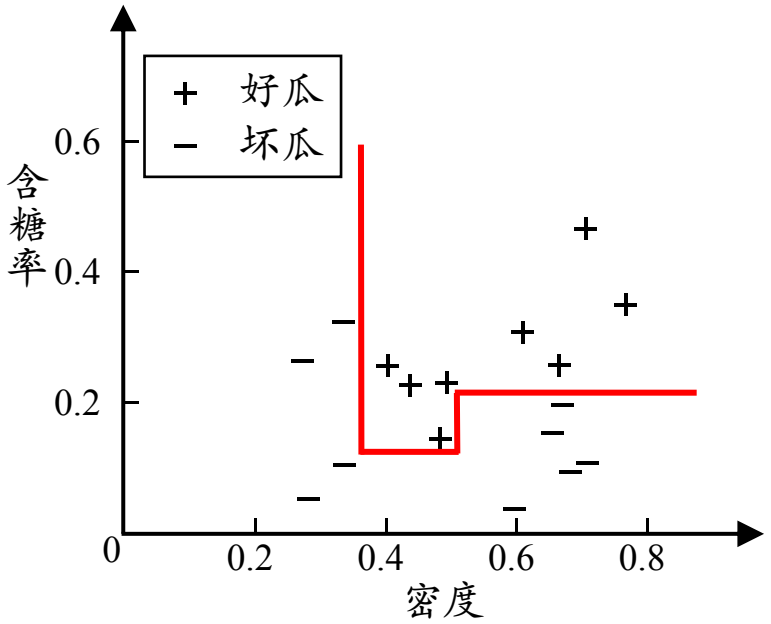
编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	–	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	–	是
3	乌黑	蜷缩	–	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	–	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	–	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	–	稍凹	硬滑	是
9	乌黑	–	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	–	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	–	否
12	浅白	蜷缩	–	模糊	平坦	软粘	否
13	–	稍蜷	浊响	稍糊	凹陷	硬滑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

Chapter 9: Contents

9.5 Multivariate Decision Trees

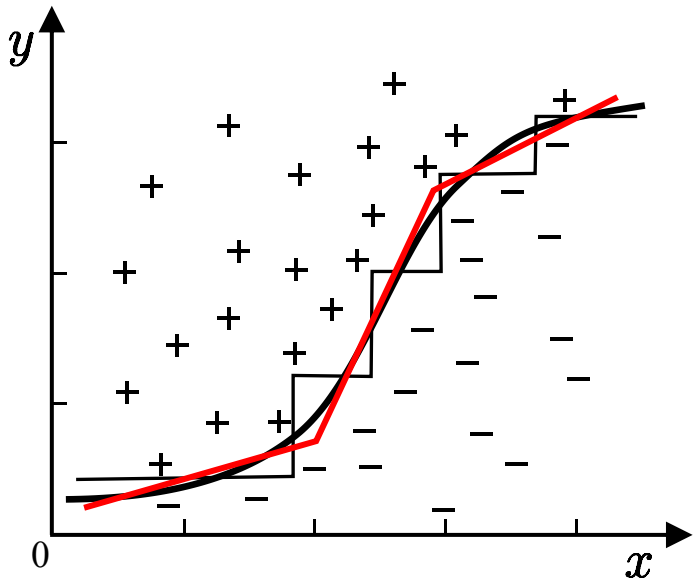
- **Univariate Decision Tree**





9.5 Multivariate Decision Trees

- **Univariate Decision Tree Classification Boundary: Axis Parallel**
- **Multivariate Decision Tree**

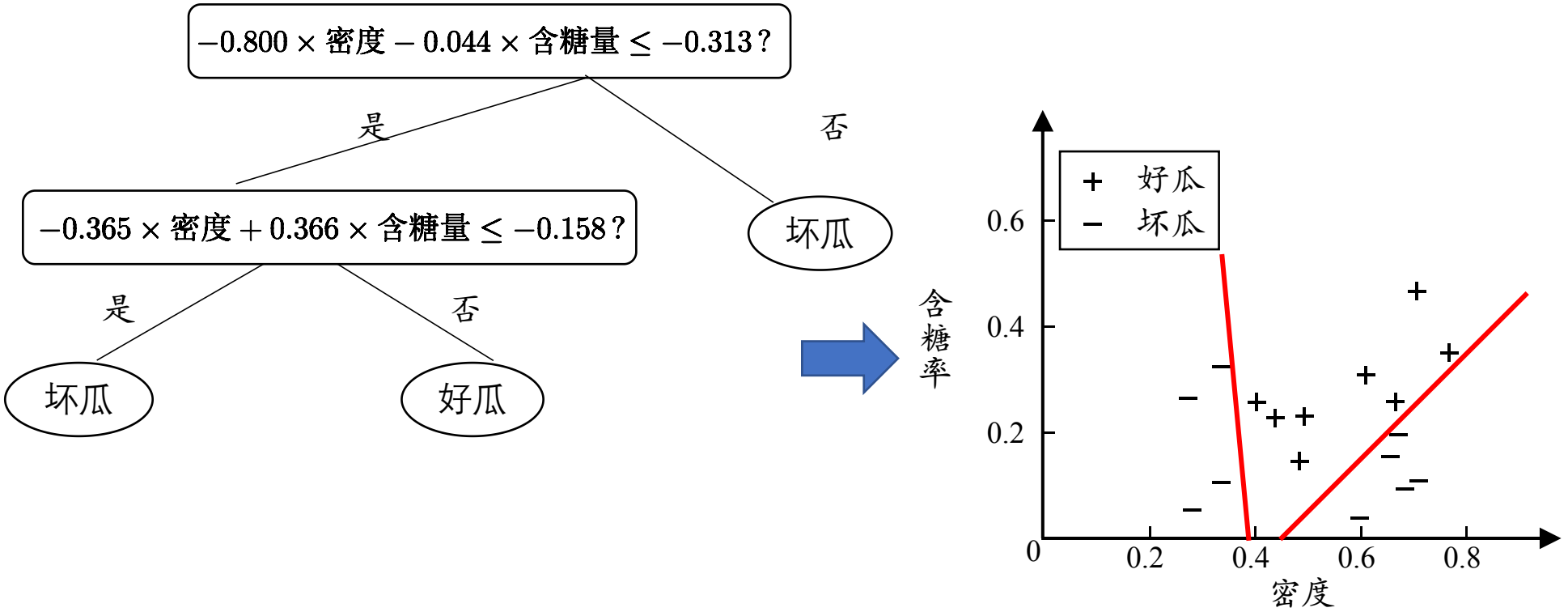


- **Non-leaf nodes are no longer restricted to a single attribute but represent linear combinations of attributes.**

Each non-leaf node is a linear classifier of the form $\sum_{i=1}^{d} w_i a_i = t$, where w_i is the weight for at$a_i$ibute w_iBoth t and can be learned from the sample set and attribute set contained in the node.

9.5 Multivariate Decision Trees

- **Multivariate Decision Tree**



9.5 Multivariate Decision Trees

- Attribute Partitioning Selection
- Pruning Treatment (Pre-pruning, Post-pruning)
- Handling Continuous and Missing Attribute Values
- Univariate Decision Trees to Multivariate Decision Trees