第四章 决策树

By Xian2207, 13689903575, wszhangxian@126.com

4.1 信息熵和信息增益

4.1.1 名词定义

[1] 信息熵 information entropy: 度量样本纯度的一种指标。公式如下

$$\mathrm{Ent}(D) = -\sum_{k=1}^{|\mathcal{Y}|} p_k \log_2 p_k \ . \tag{4.1}$$

Ent - 信息熵的缩写；

D - 样本空间；

y - 为分类数目，例如二分类，y = 2;

pk - 某样本所占样本总数的比例，例如二分类，正样本 10 个，负样本 5 个，则 pk 为 10/15 或 5/15；

[2] 信息增益 information gain: 分支节点的权值，其值越高，代表该属性来划分类的纯度越高。用信息增益划分是 ID3 决策树的算法核心。

$$\mathrm{Gain}(D,a) = \mathrm{Ent}(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Ent}(D^v) \ . \tag{4.2}$$

V - 某一属性子样本的总数；

D^V - 子样本占样本空间的个数；

D - 样本空间；

|| - 求绝对值。

4.1.2 举例 I 计算某属性的信息熵和增益

[1] 总样本空间

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
4	1	1	2	1	1	1	1
5	3	1	1	1	1	1	1
6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
8	2	2	1	1	2	1	1
9	2	2	2	2	2	1	0
10	1	3	3	1	3	2	0
11	3	3	3	3	3	1	0
12	3	1	1	3	3	2	0
13	1	2	1	2	1	1	0
14	3	2	2	2	1	1	0
15	2	2	1	1	2	2	0
16	3	1	1	3	3	1	0
17	1	1	2	2	2	1	0

[2] 子样本

D1

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
4	1	1	2	1	1	1	1
6	1	2	1	1	2	2	1

10	1	3	3	1	3	2	0
13	1	2	1	2	1	1	0
17	1	1	2	2	2	1	0

D2

2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
7	2	2	1	2	2	2	1
8	2	2	1	1	2	1	1
9	2	2	2	2	2	1	0
15	2	2	1	1	2	2	0

D3

5	3	1	1	1	1	1	1
11	3	3	3	3	3	1	0
12	3	1	1	3	3	2	0
14	3	2	2	2	1	1	0
16	3	1	1	3	3	1	0

[3]计算流程

A:求 Ent(D)

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

B:求 Ent(D^v)

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

C:求 Gain(D, color)

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

4.1.3 举例 II 计算根蒂属性的信息熵和增益

[1] 总样本空间

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
4	1	1	2	1	1	1	1
5	3	1	1	1	1	1	1
6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
8	2	2	1	1	2	1	1
9	2	2	2	2	2	1	0
10	1	3	3	1	3	2	0
11	3	3	3	3	3	1	0
12	3	1	1	3	3	2	0
13	1	2	1	2	1	1	0

14	3	2	2	2	1	1	0
15	2	2	1	1	2	2	0
16	3	1	1	3	3	1	0
17	1	1	2	2	2	1	0

[2] 子样本

D1

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
4	1	1	2	1	1	1	1
5	3	1	1	1	1	1	1
12	3	1	1	3	3	2	0
16	3	1	1	3	3	1	0
17	1	1	2	2	2	1	0

D2

6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
8	2	2	1	1	2	1	1
9	2	2	2	2	2	1	0
13	1	2	1	2	1	1	0
14	3	2	2	2	1	1	0
15	2	2	1	1	2	2	0

D3

10	1	3	3	1	3	2	0

11 3 3 3 3 3 1 0

[3]计算流程

A:求 Ent(D)

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

B:求 Ent(D^v)

Ent(D1) = -[(5/8)*log2(5/8) + (3/8)*log2(3/8)] = 0.95443;

Ent(D2) = -[(3/7)*log2(3/7) + (4/7)*log2(4/7)] = 0.98523;

Ent(D3) = -[(0/2)*log2(0/2) + (2/2)*log2(2/2)] = 0;

C:求 Gain(D, root)

Gain(D, root) = Ent(D) - sum[(D^v/D)*Ent(D^v)]

= 0.998 - [(8/17)*0.95443 + (7/17)*0.98523 + (0/2)*0]

= 0.14317

总结：

（1）任何属性的信息熵是一样的，但该属性的信息增益却不一样；

（2）计算信息增益，需对该属性的值先排序如 1,2,3，然后再计算；

（3）把第一列序号考虑进去，由序列号计算的信息增益远大于 color, root, knock,...etc.;

（4）信息增益划分偏好属性值多的属性，这样不利于学习器泛化能力的展示。

4.2 信息增益率

4.2.1 信息增益率定义

$$\text{Gain\_ratio}(D,a) = \frac{\text{Gain}(D,a)}{\text{IV}(a)}, \tag{4.3}$$

其中

$$\text{IV}(a) = -\sum_{v=1}^{V} \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|} \tag{4.4}$$

4.4 式称为属性的固定值（intrinsic value）。当属性的取值越多，D^v 的空间越大，则 4.4 式的值越大，即 4.3 式的分母越大，从而平衡信息增益准则偏好属性值多的影响。该方法是 C4.5 决策树算法的核心。

4.2.2 举例计算 color 属性的增益率

IV(alpha) = -[(D1/D)*log2(D1/D) +(D2/D)*log2(D2/D) + (D3/D)*log2(D3/D)]

= -[(6/17)*log2(6/17) +(6/17)*log2(6/17) + (5/17)*log2(5/17)]

= 1.5799

Gain_ratio(D, color) = 0.109/1.5799

= 0.068987

4.3 基尼指数

4.3.1 基尼指数定义

$$\begin{aligned}\mathrm{Gini}(D) &= \sum_{k=1}^{|\mathcal{Y}|}\sum_{k'\neq k} p_k p_{k'} \\ &= 1 - \sum_{k=1}^{|\mathcal{Y}|} p_k^2 \ . \end{aligned} \tag{4.5}$$

Gini(D) 反映了从数据集 D 中随机抽取两个样本，其类别标记不一致的概率。因此，Gini(D) 越小，则数据集 D 的纯度越高。

$$\mathrm{Gini\_index}(D,a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Gini}(D^v) \ . \tag{4.6}$$

同上，基尼指数越小，纯度越高，可作为属性划分依据。

4.3.2 举例计算 color 属性的基尼指数

[1]计算流程

A: D1 = 6; D2 = 6; D3 = 5; D = 17;

B: Gini(D1) = 1 - [(3/6)*(3/6)+(3/6)*(3/6)]

= 0.5;

Gini(D2) = 1 - [(4/6)*(4/6) + (2/6)*(2/6)]

= 0.4444;

Gini(D3) = 1 - [(1/5)*(1/5) + (4/5)*(4/5)]

= 0.32;

C: Gini_index = (6/17)*0.5 + (6/17)*0.4444 + (5/17)*0.32

= 0.42744

4.4 预剪枝

以下为预剪枝计算案例。

4.4.1 训练集

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
10	1	3	3	1	3	2	0
14	3	2	2	2	1	1	0
15	2	2	1	1	2	2	0
16	3	1	1	3	3	1	0
17	1	1	2	2	2	1	0

4.4.2 计算增益

[1] 以 color 为基准的样本空间变为

D1

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
6	1	2	1	1	2	2	1
10	1	3	3	1	3	2	0
17	1	1	2	2	2	1	0

D2

#	color	root	knock	pattern	umbilicus	touch	label

2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
7	2	2	1	2	2	2	1
15	2	2	1	1	2	2	0

D3

#	color	root	knock	pattern	umbilicus	touch	label
14	3	2	2	2	1	1	0
16	3	1	1	3	3	1	0

[2]计算信息熵

Ent(D) = -[(5/10)*log2(5/10) +(5/10)*log2(5/10)] = 1;

Ent(D1) = -[(2/4)*log2(2/4) + (2/4)*log2(2/4)] = 1;

Ent(D2) = -[(3/4)*log2(3/4) + (1/4)*log2(1/4)] = 0.81128;

Ent(D3) = -[(0/2)*log2(0/2) + (2/2)*log2(2/2)] = 0;

[3]计算信息增益

Gain(D, color) = 1 - (1*(4/10) + 0.81128*(4/10) + 0*(2/10)) = 0.27549;

同理，以 root 为基准

D1

#	color	root	knock	pattern	umbilicus	touch	label
1	1	1	1	1	1	1	1
2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
16	3	1	1	3	3	1	0
17	1	1	2	2	2	1	0

D2

6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
14	3	2	2	2	1	1	0
15	2	2	1	1	2	2	0

D3

10	1	3	3	1	3	2	0

计算信息熵和增益

Ent(D) = 1, 同 color;

Ent(D1) = -[(3/5)*log2(3/5)+(2/5)*log2(2/5)] = 0.97095;

Ent(D2) = -((2/4)*log2(2/4) + (2/4)*log2(2/4))= 1;

Ent(D3) = -((0/1)*log2(0/1)+(1/1)*log2(1/1)) = 0;

Gain(D, root) = 1 - (0.97095*(5/10) + 1*(4/10) + 0*(1/10)) = 0.11452;

同理，以 knock 为基准

D1

1	1	1	1	1	1	1	1
3	2	1	1	1	1	1	1
6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
15	2	2	1	1	2	2	0
16	3	1	1	3	3	1	0

D2

2	2	1	2	1	1	1	1
14	3	2	2	2	1	1	0
17	1	1	2	2	2	1	0

D3

10 1 3 3 1 3 2 0

计算信息熵和增益

Ent(D) = 1;

Ent(D1) = -((4/6)*log2(4/6) + (2/6)*log2(2/6)) = 0.9183;

Ent(D2) = 0.9183;

Ent(D3) = 0;

Gain(D, knock) = 1 - (0.9183*(6/10) + 0.9183*(3/10) + 0*(1/10)) = 0.17353;

同理，以 pattern 为基准

D1

1 1 1 1 1 1 1 1

2 2 1 2 1 1 1 1

3 2 1 1 1 1 1 1

6 1 2 1 1 2 2 1

10 1 3 3 1 3 2 0

15 2 2 1 1 2 2 0

D2

7 2 2 1 2 2 2 1

14 3 2 2 2 1 1 0

17 1 1 2 2 2 1 0

D3

16 3 1 1 3 3 1 0

计算增益

Gain(D, pattern) = 0.17353;

同理，以 umbilicus 为基准

D1

1	1	1	1	1	1	1	1
2	2	1	2	1	1	1	1
3	2	1	1	1	1	1	1
14	3	2	2	2	1	1	0

D2

6	1	2	1	1	2	2	1
7	2	2	1	2	2	2	1
15	2	2	1	1	2	2	0
17	1	1	2	2	2	1	0

D3

10	1	3	3	1	3	2	0
16	3	1	1	3	3	1	0

Ent(D) = 1;

Ent(D1) = -((3/4)*log2(3/4) + (1/4)*log2(1/4)) = 0.81128;

Ent(D2) = 1;

Ent(D3) = 0;

Gain(D, umbilicus) = 1 - (0.81128*(4/10) + 1*(4/10) + 0*(2/10)) = 0.27549;

同理以 touch 为基准

Gain(D, touch) = 0.11110(假设)

4.4.3 确定划分属性

比较各增益，color 和 umbilicus 一样，任选其一，我们选择 umbilicus，子集为

D1(1, 2, 3, 14);

D2(6,7,15,17);

D3(10,16);

D1 以 color 为划分基准，子集为

D11(1) -> 好瓜

D12(2,3) -> 好瓜;

D13(14) -> 坏瓜;

D2 以 root 为基准，子集为

D21(17) -> 坏瓜;

D22(6, 7, 15);

D22 以 color 为例

D221(6) ->好瓜;

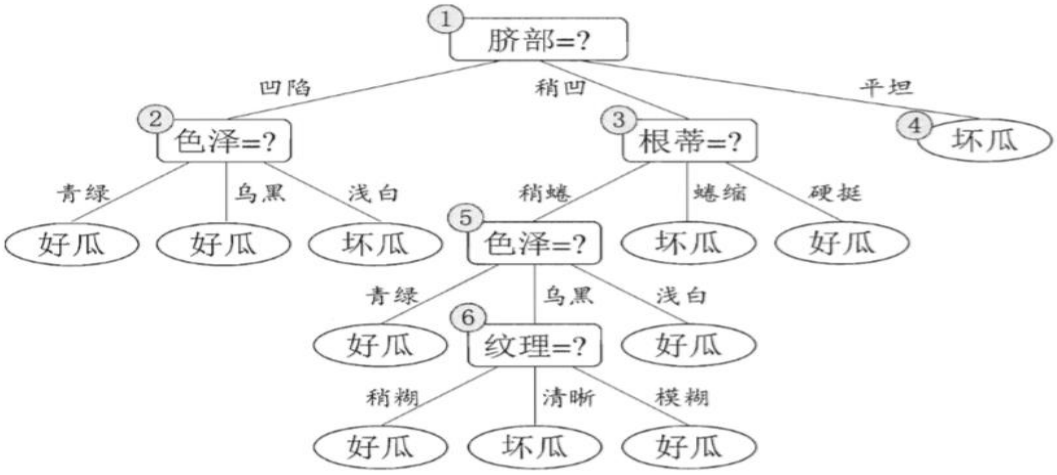
D222(7,15);

D222 以 pattern 为例

D2221(15) -> 坏瓜；

D2222(7) -> 好瓜；

D3(10,16) -> 坏瓜；最终决策树如下



4.4.4 测试集

有了训练好的决策树，现找以下测试集进行验证

D

#	color	root	knock	pattern	umbilicus	touch	label
4	1	1	2	1	1	1	1
5	3	1	1	1	1	1	1
8	2	2	1	1	2	1	1
9	2	2	2	2	2	1	0
11	3	3	3	3	3	1	0
12	3	1	1	3	3	2	0
13	1	2	1	2	1	1	0

[1] 以 umbilicus 为划分属性

D1

4	1	1	2	1	1	1	1
5	3	1	1	1	1	1	1
13	1	2	1	2	1	1	0

D2

8 2 2 1 1 2 1 1

9 2 2 2 2 2 1 0

D3

11 3 3 3 3 3 1 0（坏瓜）

12 3 1 1 3 3 2 0（坏瓜）

[2] 再以 color 为划分属性

D11

4 1 1 2 1 1 1 1 （好瓜）

13 1 2 1 2 1 1 0 （好瓜）

D12

5 3 1 1 1 1 1 1（坏瓜）

[3] 再以 root 为划分属性

D22(8,9) 好瓜

[4] 精确度

验证集精确度：

4 的 label 为 1，实际为 1；

5 的 label 为 0，实际为 1，错误；

8 的 label 为 1，实际为 1；

9 的 label 为 1，实际为 0，错误；

11 的 label 为 0，实际为 0；

12 的 label 为 0，实际为 0；

13 的 label 为 1，实际为 0，错误；

故根据训练集产生的决策树，测试集正确率 = 4/7 = 0.57413；

4.4.5 剪枝处理

假设剪枝如下，直接将凹陷和稍凹设置为好瓜

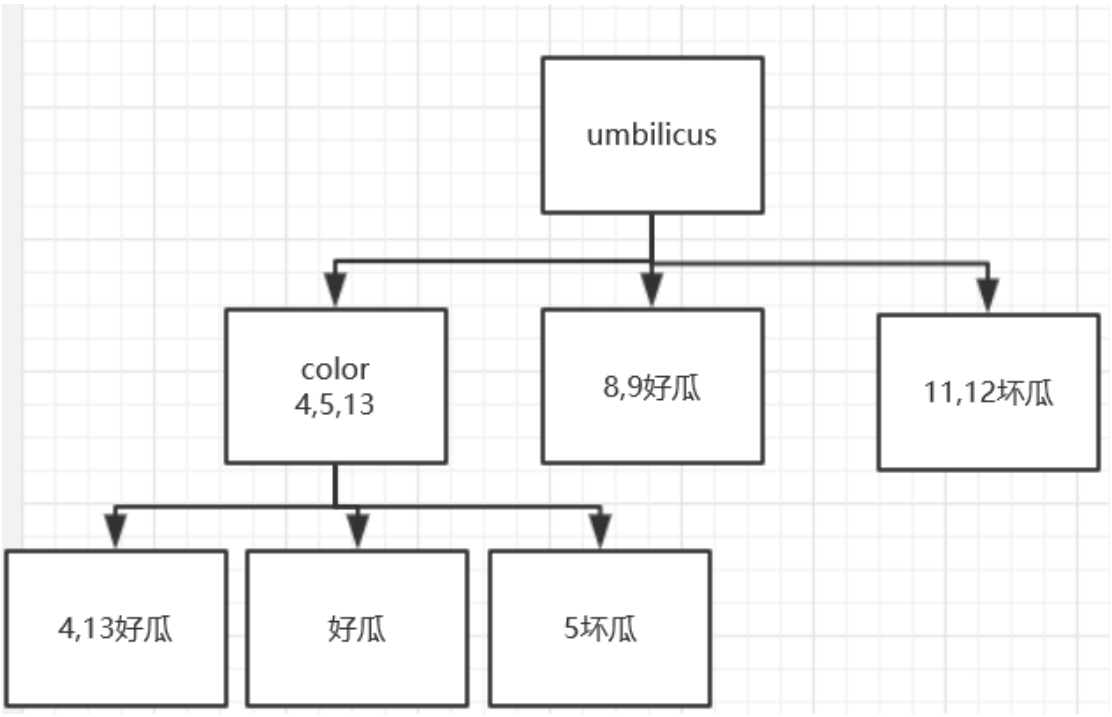
根据 D1, D2, D3 可知，

D1(4,513)好瓜；

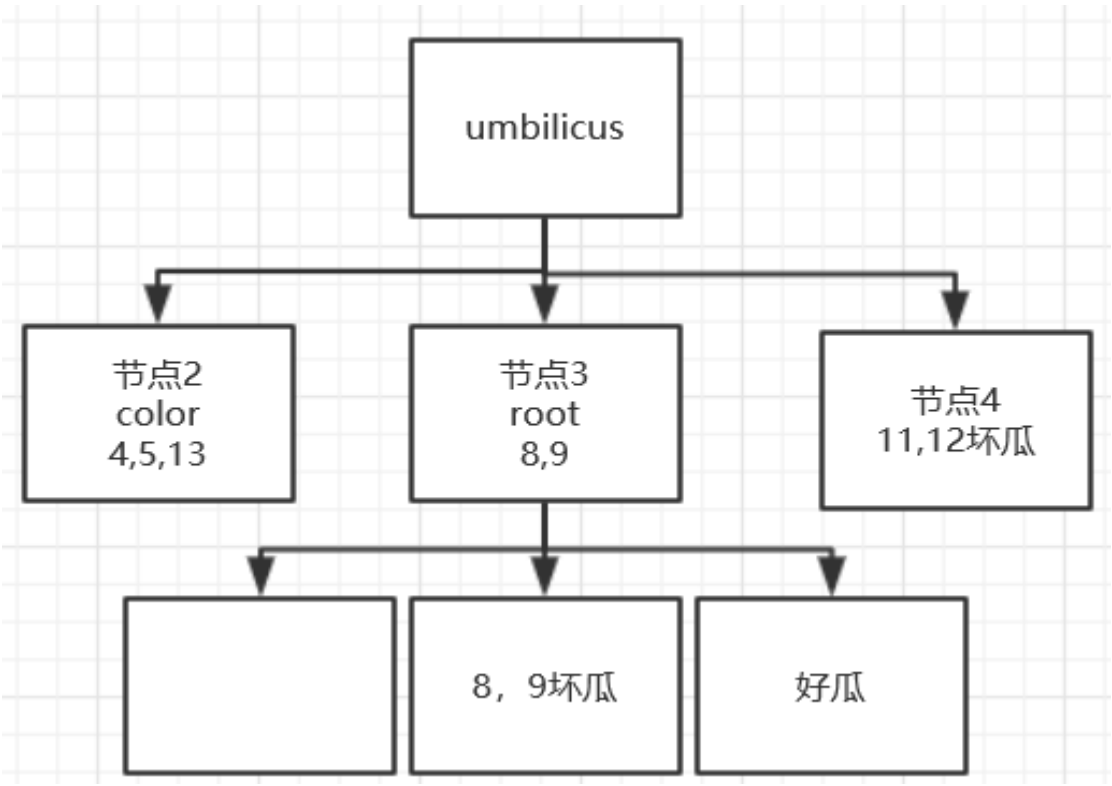
D2(8,9)好瓜；

D3(11,12)坏瓜

测试集正确率 = 5/7 = 0.71429 提升了。确定 umbilicus 可为第一次划分属性。接着对上图的节点 2 进行色泽划分，决策树为



正确率 = 4/7,相比 0.71429 低，所以色泽（上上图中的节点 2 好瓜）不可划分；对于节点 3，如果按 root 划分，决策树（决策树图因为书上有误，可能是数据集标签问题）为



正确率 = 4/7，相比 0.71429 低，所以不划分。同样节点 4,已经属于同一类，不可裁剪或划分。最终确定的决策树如下，只有一层，也叫决策树桩。

总结

预剪裁前提是需一棵训练好的决策树，从上往下，对非叶节点逐个考察，判断剪枝前后测试集正确率大小。优点是可降低过拟合风险，减少分支展开，降低训练开销，降低验证开销。但风险是会出现欠拟合。

4.5 后剪枝

总结：后剪枝前提是先生成一颗完全决策树，从小到上，对所有非叶节点逐个考察，判断剪枝后测试集的正确率大小，逐个剪枝。优点是比预剪枝泛化能力强，但验证开销大。

4.6 连续值处理

上述属性值均为离散值，若为连续值，则采用二分法。

4.6.1 二分法定义

$$T_a = \left\{ \frac{a^i + a^{i+1}}{2} \mid 1 \leqslant i \leqslant n-1 \right\} , \tag{4.7}$$

$$\begin{aligned} \mathrm{Gain}(D,a) &= \max_{t \in T_a} \mathrm{Gain}(D,a,t) \\ &= \max_{t \in T_a} \mathrm{Ent}(D) - \sum_{\lambda \in \{-,+\}} \frac{|D_t^\lambda|}{|D|} \mathrm{Ent}(D_t^\lambda) , \end{aligned} \tag{4.8}$$

4.6.2 举例

D：样本空间加入密度和糖度等连续属性；

#	color	root	knock	pattern	umbilicus	touch	density	suger	label
1	1	1	1	1	1	1	0.697	0.46	1
2	2	1	2	1	1	1	0.774	0.376	1
3	2	1	1	1	1	1	0.634	0.264	1
4	1	1	2	1	1	1	0.608	0.318	1
5	3	1	1	1	1	1	0.556	0.215	1
6	1	2	1	1	2	2	0.403	0.237	1
7	2	2	1	2	2	2	0.481	0.149	1

8	2	2	1	1	2	1	0.437	0.211	1
9	2	2	2	2	2	1	0.666	0.091	0
10	1	3	3	1	3	2	0.243	0.267	0
11	3	3	3	3	3	1	0.245	0.057	0
12	3	1	1	3	3	2	0.343	0.099	0
13	1	2	1	2	1	1	0.639	0.161	0
14	3	2	2	2	1	1	0.657	0.198	0
15	2	2	1	1	2	2	0.36	0.37	0
16	3	1	1	3	3	1	0.593	0.042	0
17	1	1	2	2	2	1	0.719	0.103	0

4.6.3 计算划分点

[1] 密度排序

$T_{密度}$= {0.243, 0.245, 0.343, 0.36, 0.403, 0.437, 0.481, 0.556, 0.593, 0.608, 0.634, 0.639, 0.657, 0.666, 0.697, 0.719, 0.774}.

D: 排序后空间如下

#	color	root	knock	pattern	umbilicus	touch	density	suger	label
10	1	3	3	1	3	2	0.243	0.267	0
11	3	3	3	3	3	1	0.245	0.057	0
12	3	1	1	3	3	2	0.343	0.099	0
15	2	2	1	1	2	2	0.36	0.37	0
6	1	2	1	1	2	2	0.403	0.237	1
8	2	2	1	1	2	1	0.437	0.211	1
7	2	2	1	2	2	2	0.481	0.149	1
5	3	1	1	1	1	1	0.556	0.215	1
16	3	1	1	3	3	1	0.593	0.042	0

4	1	1	2	1	1	1	0.608	0.318	1
3	2	1	1	1	1	1	0.634	0.264	1
13	1	2	1	2	1	1	0.639	0.161	0
14	3	2	2	2	1	1	0.657	0.198	0
9	2	2	2	2	2	1	0.666	0.091	0
1	1	1	1	1	1	1	0.697	0.46	1
17	1	1	2	2	2	1	0.719	0.103	0
2	2	1	2	1	1	1	0.774	0.376	1

[2] 计算 T_a

i = 1; T_a = (0.243+0.245)/2 = 0.244

I = 2; T_a = (0.245+0.343)/2 = 0.294

I = 3; T_a = (0.343+0.360)/2 = 0.351

...

Ta = {0.244, 0.294, 0.351, 0.381, 0.420, 0.459, 0.518, 0.574, 0.6, 0.621, 0.636, 0.648, 0.661, 0.681, 0.708, 0.746};

[3]计算增益

Ta = 0.244；

Dt- = {0.243};

Dt+={0.245, 0.343, 0.36, 0.403, 0.437, 0.481, 0.556, 0.593, 0.608, 0.634, 0.639, 0.657, 0.666, 0.697, 0.719, 0.774} ;

Ent(D) = -((8/17)*log2(8/17) + (9/17)*log2(9/17)) = 0.99750;

Ent(Dt-) = -((0/1)*log2(0/1) + (1/1)*log2(1/1)) = 0;

Ent(Dt+) = -((8/16)*log2(8/16) + (8/16)*log2(8/16)) = 1;

Gain(D, density, 0.244) = 0.99750 - (0*(1/17) + 1*(16/17)) = 0.056324

Ta = 0.294;

Dt- = {0.243, 0.245};

Dt+= {0.343, 0.36, 0.403, 0.437, 0.481, 0.556, 0.593, 0.608, 0.634, 0.639, 0.657, 0.666, 0.697, 0.719, 0.774} ;

Ent(Dt-) = -((0/2)*log2(0/2) + (2/2)*log2(2/2)) = 0;

Ent(Dt+) = -((8/15)*log2(8/15) + (7/15)*log2(7/15)) = 0.997;

Gain(D, density, 0.294) = 0.99750 - (0*(2/17) + 0.997*(15/17)) = 0.11779;

Ta = 0.351;

Dt- = {0.243, 0.245, 0.343};

Dt+= {0.36, 0.403, 0.437, 0.481, 0.556, 0.593, 0.608, 0.634, 0.639, 0.657, 0.666, 0.697, 0.719, 0.774};

Gain(D, density, 0.351) = 0.187;

.....

Ta = 0.381;

Gain(D, density, 0.381) = 0.263;

Ta = 0.42;

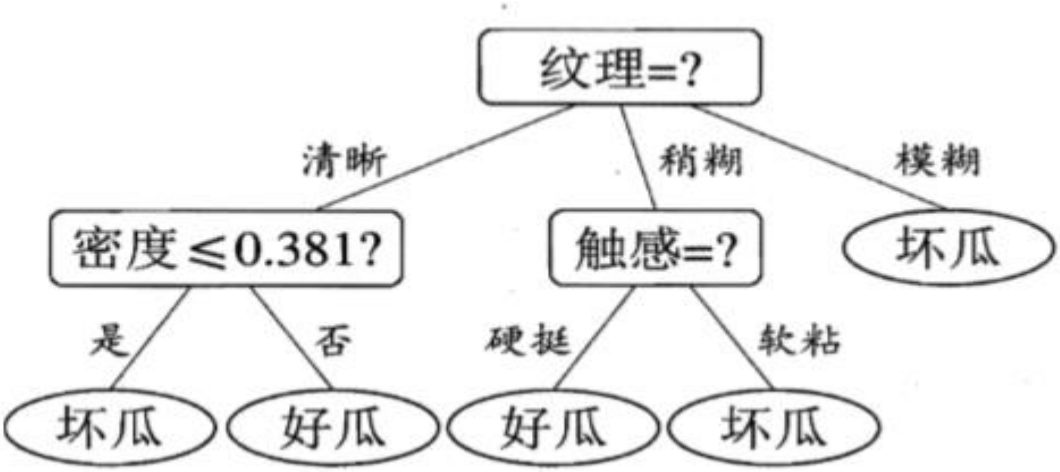
Gain(D, density, 0.42) = 0.094;

.....

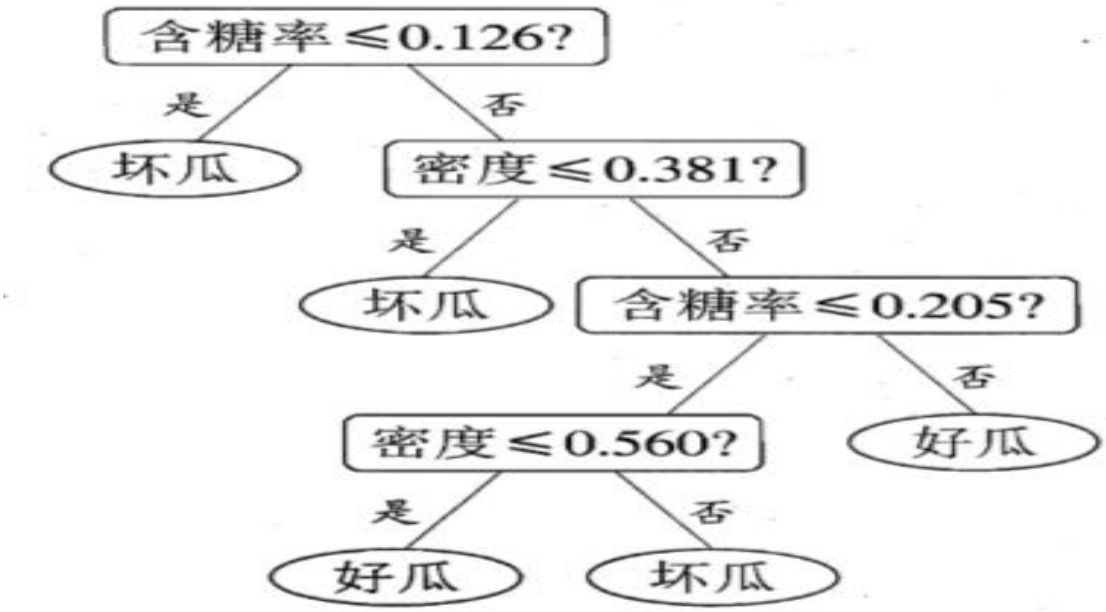
比较发现，Ta = 0.381 时，增益为 0.263（最大），故选择 0.381 为划分点，即高于 0.381 为好瓜，低于 0.381 为坏瓜；糖度依此方法，可发现，Ta = 0.126 时，增益为 0.349（最大），故选择 0.126 为糖度划分点，即糖度低于 0.126 为坏瓜，高于此为好瓜。结合其他信息增益，

$$\begin{aligned}
&\text{Gain}(D, \text{色泽}) = 0.109; \quad \text{Gain}(D, \text{根蒂}) = 0.143; \\
&\text{Gain}(D, \text{敲声}) = 0.141; \quad \text{Gain}(D, \text{纹理}) = 0.381; \\
&\text{Gain}(D, \text{脐部}) = 0.289; \quad \text{Gain}(D, \text{触感}) = 0.006; \\
&\text{Gain}(D, \text{密度}) = 0.262; \quad \text{Gain}(D, \text{含糖率}) = 0.349.
\end{aligned}$$

决策树生成如下



注意：离散属性如 color, root, knock, touch 划分时不可连续划分，如 pattern-color-knock-touch, 不能出现 pattern-pattern-color-knock-knock 这样划分；但连续属性可以，如下决策树



4.7 缺失值处理

样本空间如下：

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

定义

$$\rho = \frac{\sum_{\boldsymbol{x} \in \tilde{D}} w_{\boldsymbol{x}}}{\sum_{\boldsymbol{x} \in D} w_{\boldsymbol{x}}}, \tag{4.9}$$

$$\tilde{p}_k = \frac{\sum_{\boldsymbol{x} \in \tilde{D}_k} w_{\boldsymbol{x}}}{\sum_{\boldsymbol{x} \in \tilde{D}} w_{\boldsymbol{x}}} \quad (1 \leqslant k \leqslant |\mathcal{Y}|), \tag{4.10}$$

$$\tilde{r}_v = \frac{\sum_{\boldsymbol{x} \in \tilde{D}^v} w_{\boldsymbol{x}}}{\sum_{\boldsymbol{x} \in \tilde{D}} w_{\boldsymbol{x}}} \quad (1 \leqslant v \leqslant V). \tag{4.11}$$

4.9 - 无缺失值样本所占的比例；

4.10 - 无缺失值样本中第 K 类所占的比例；

4.11 - 无缺失值样本中某一属性值样本所占的比例

增益计算公式

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

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

1 样本空间

#	color	root	knock	pattern	umbilicus	touch	label
1	X	1	1	1	1	1	1
2	2	1	2	1	1	X	1
3	2	1	X	1	1	1	1
4	1	1	2	1	1	1	1
5	X	1	1	1	1	1	1
6	1	2	1	1	X	2	1
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1
9	2	X	2	2	2	1	0
10	1	3	3	X	3	2	0
11	3	3	3	3	3	X	0
12	3	1	X	3	3	2	0
13	X	2	1	2	1	1	0
14	3	2	2	2	1	1	0
15	2	2	1	1	X	2	0
16	3	1	1	3	3	1	0
17	1	X	2	2	2	1	0

2 计算增益

[1] 以 color 为基准计算其增益

#	color	root	knock	pattern	umbilicus	touch	label

2	2	1	2	1	1	X	1
3	2	1	X	1	1	1	1
4	1	1	2	1	1	1	1
6	1	2	1	1	X	2	1
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1
9	2	X	2	2	2	1	0
10	1	3	3	X	3	2	0
11	3	3	3	3	3	X	0
12	3	1	X	3	3	2	0
14	3	2	2	2	1	1	0
15	2	2	1	1	X	2	0
16	3	1	1	3	3	1	0
17	1	X	2	2	2	1	0

无缺失样本 D’ = {2,3,4,6,7,8,9,10,11,12,14,15,16,17}共 14 个，正样本 6 个，负样本 8 个，假定权值都为 1；则

rho = 14/17;

Ent(D’) = -((6/14)*log2(6/14) + (8/14)*log2(8/14)) = 0.98523;

[2] 样本进行划分

D1:

4	1	1	2	1	1	1	1
6	1	2	1	1	X	2	1
10	1	3	3	X	3	2	0
17	1	X	2	2	2	1	0

D2

2	2	1	2	1	1	X	1

3	2	1	X	1	1	1	1
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1
9	2	X	2	2	2	1	0
15	2	2	1	1	X	2	0

D3

11	3	3	3	3	3	X	0
12	3	1	X	3	3	2	0
14	3	2	2	2	1	1	0
16	3	1	1	3	3	1	0

计算熵

Ent(D1) = - ((2/4)*log2(2/4) + (2/4)*log2(2/4)) = 1;

Ent(D2) = -((4/6)*log2(4/6) + (2/6)*log2(2/6)) = 0.918;

Ent(D3) = 0;

计算增益

Gain(D', color) = 0.98523 - (1*(4/14) + 0.918*(6/14) + 0*(4/14)) = 0.306;

Gain(D, color) = rho*Gain(D', color) = (14/17)*0.306 = 0.252;

同理，

Gain(D, root) = 0.171;

Gain(D, knock)= 0.145;

Gain(D, pattern) = 0.424;

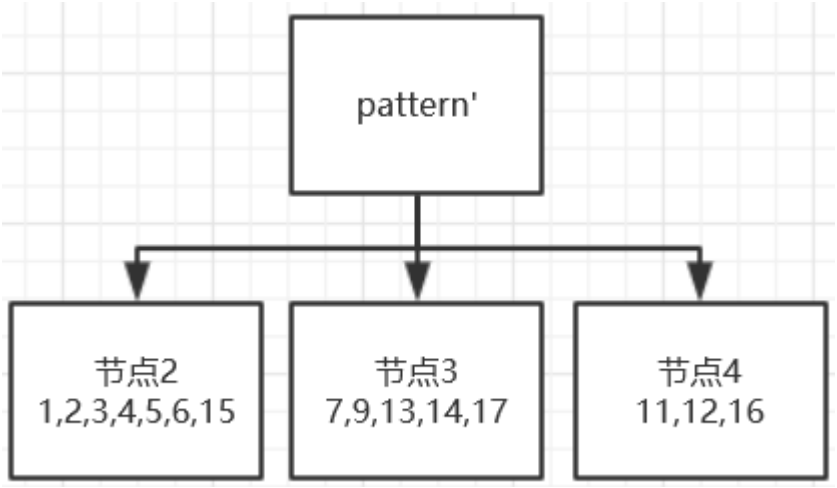
Gain(D, umbilicus) = 0.289;

Gain(D, touch)= 0.006;

比较发现，pattern 属性的增益值最大，故作为第一次划分属性

[3] pattern 划分

第一次划分，决策树如下



D: 总样本空间

#	color	root	knock	pattern	umbilicus	touch	label
1	X	1	1	1	1	1	1
2	2	1	2	1	1	X	1
3	2	1	X	1	1	1	1
4	1	1	2	1	1	1	1
5	X	1	1	1	1	1	1
6	1	2	1	1	X	2	1
7	2	2	1	2	2	2	1
9	2	X	2	2	2	1	0
11	3	3	3	3	3	X	0
12	3	1	X	3	3	2	0
13	X	2	1	2	1	1	0
14	3	2	2	2	1	1	0
15	2	2	1	1	X	2	0
16	3	1	1	3	3	1	0
17	1	X	2	2	2	1	0

D’1

#	color	root	knock	pattern	umbilicus	touch	label
1	X	1	1	1	1	1	1
2	2	1	2	1	1	X	1
3	2	1	X	1	1	1	1
4	1	1	2	1	1	1	1
5	X	1	1	1	1	1	1
6	1	2	1	1	X	2	1
15	2	2	1	1	X	2	0

D’2

7	2	2	1	2	2	2	1
9	2	X	2	2	2	1	0
13	X	2	1	2	1	1	0
14	3	2	2	2	1	1	0
17	1	X	2	2	2	1	0

D’3

11	3	3	3	3	3	X	0
12	3	1	X	3	3	2	0
16	3	1	1	3	3	1	0

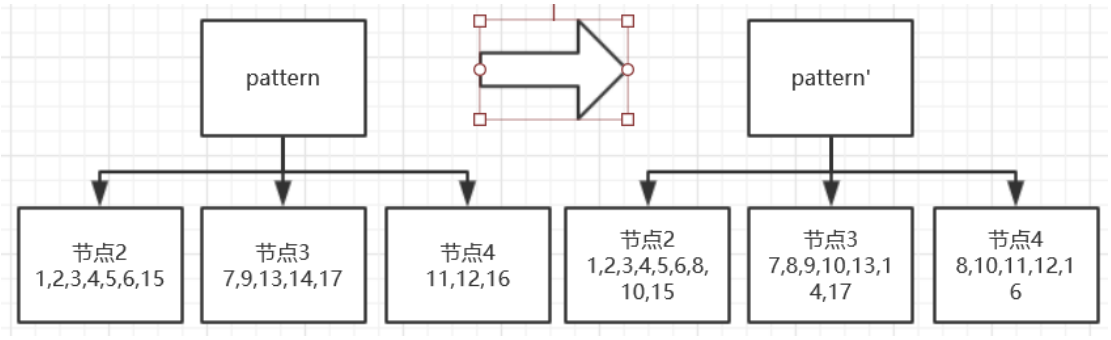
[3] 任选一子节点继续划分属性

如选择节点 3，将其作为新 D

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1
9	2	X	2	2	2	1	0
13	X	2	1	2	1	1	0

14	3	2	2	2	1	1	0
17	1	X	2	2	2	1	0

[4] 对样本空间的缺失值处理



如图，将第一次划分前缺失的 8 和 10 全部放入节点 3 中，注意权值发生变化:

节点 2: 8 和 10 的权值为 7/15;

节点 3: 8 和 10 的权值为= 5/15 = 1/3;

节点 4: 8 和 10 的权值为= 3/15 = 1/5;

对于节点 3，加入 8 和 10 后

D：

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1
9	2	X	2	2	2	1	0
10	1	3	3	X	3	2	0
13	X	2	1	2	1	1	0
14	3	2	2	2	1	1	0
17	1	X	2	2	2	1	0

对于节点 3，去掉缺失值 D’为

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1

8 2 2 1 X 2 1 1

9 2 X 2 2 2 1 0

10 1 3 3 X 3 2 0

14 3 2 2 2 1 1 0

17 1 X 2 2 2 1 0

rho3 = [原来 D 中无缺失样本总数 + 外来无缺失总数(注意权值 1/3 对应节点 3)]/[原来 D 中样本总数(包含缺失) + 外来样本总数(注意权值为 1/3))

= (4+2/3)/(5 + 2/3) = 14/17;

正样本比例 = (1 + 1/3) / (4+ 2/3) = 4/14;

负样本比例 = (4+1/3) / (4+2/3) = 10/14;

注意：若 10 也是 label 1，则

正样本比例 = (1+ 2/3)/(4+2/3) = 0.35714;

负样本比例 = 3/(4 + 2/3) = 0.64286;

Ent(D') = -((4/14)*log2(4/14) + (10/14)*log2(10/14)) = 0.86312;

以 color 为准，无缺失值样本空间为

D'1

10 1 3 3 X 3 2 0

17 1 X 2 2 2 1 0

D'2

7 2 2 1 2 2 2 1

8 2 2 1 X 2 1 1

9 2 X 2 2 2 1 0

D'3

14 3 2 2 2 1 1 0

R1 = (1+1/3)/(4+2/3) = 4/14;

R2 = (2+1/3)/(4+2/3) = 7/14;

R3 = 1/(4+2/3) = 3/14;

Ent(D'1) = -((0/(1+1/3))*log2(0/(1=1/3)) + ((1+1/3)/(1+1/3))*log2((1+1/3)/(1+1/3))) = 0;

Ent(D'2) = -(((1+1/3)/(2+1/3))*log2(((1+1/3)/(2+1/3))) +(1/(2+1/3))*log2(1/(2+1/3))) = 0.98523;

Ent(D'3) = 0;

Gain(D, color) = Ent(D') - (R1*Ent(D1) + R2*Ent(D2) + R3*Ent(D3))

= 0.86312 - (0*(4/14) + 0.98523*(7/14) + 0*(3/14)) = 0.3705;

同理

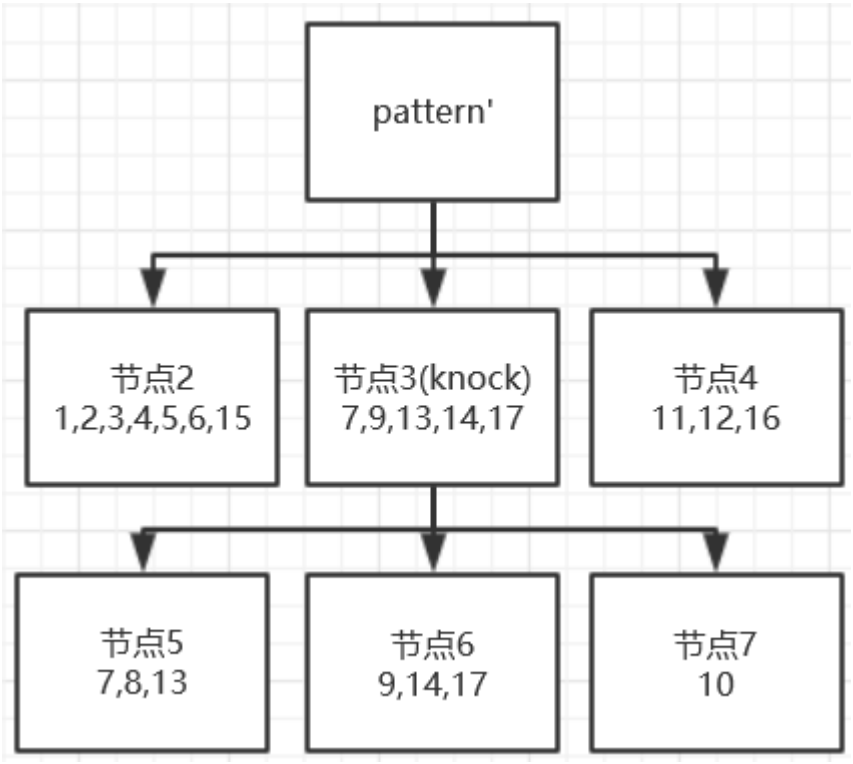
Gain(D, root) = 0.039;

Gain(D, knock) = 0.381; 没有缺失值，正常算增益，注意 8 和 10 的权值即可

Gain(D, umbilicus) = 0.216;

Gain(D, touch) = 0.291;

比较发现，knock 属性可做划分，决策树变为



发现 9,14,17 都是 label 0，一类的，所以直接标记为叶子，即坏瓜；10 也是坏瓜。但节点 5 中有 1 和 0，需继续划分，下面要对节点 5 重复上述计算，权值仍是 1/3，继承自节点 3。

D = {7,8,13}

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1
13	X	2	1	2	1	1	0

rho5 = (1 + 1/3) / (2+ 1/3) = 0.57143;

正样本比例 = ((1+1/3) / (2 + 1/3))= 0.57143;

负样本比例 = (1/(2+1/3)) = 0.42857;

Ent(D') = -(0.57143*log2(0.57143) + 0.42857*log2(0.42857)) = 0.98523;

以 color 为基准，D'

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1

8 2 2 1 X 2 1 1

D1 = {0};

D2 = {7,8};

D3 = {0};

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1

Ent(D'1) = 0;

Ent(D'2) = -((1+1/3)/(1+1/3)*log2((1+1/3)/(1+1/3)) + 0) = 0;

Ent(D'3) = 0;

Gain(D', color) = 0.98523 - (0*0 + 0*2/2 + 0*0)) = 0.98523;

Gain(D, color) = 0.98523*0.57143 = 0.56299;

同理，

root: 无缺失值，则直接计算增益

#	color	root	knock	pattern	umbilicus	touch	label
7	2	2	1	2	2	2	1
8	2	2	1	X	2	1	1
13	X	2	1	2	1	1	0

D1 = {0};

D2 = {7,8,13};

D3 = {0};

Ent(D1) = 0;

Ent(D2) = -(((1+1/3)/(2+1/3))*log2(((1+1/3)/(2+1/3)))+(1/(2+1/3))*log2((1/(2+1/3)))) =0.98523;

Ent(D3) = 0;

Gain(D, root) = 0.98523 - (0*0/3+0.98523*(3/3)+0*0/3) = 0;

同理，

Gain(D, knock) = 0;

同理，

Umbilicus

D1

13 X 2 1 2 1 1 0

D2

7 2 2 1 2 2 2 1

8 2 2 1 X 2 1 1

Ent(D) = -(((1+1/3)/(2+1/3))*log2(((1+1/3)/(2+1/3))) + ((1/(2+1/3))*log2(1/(2+1/3)))) = 0.98523;

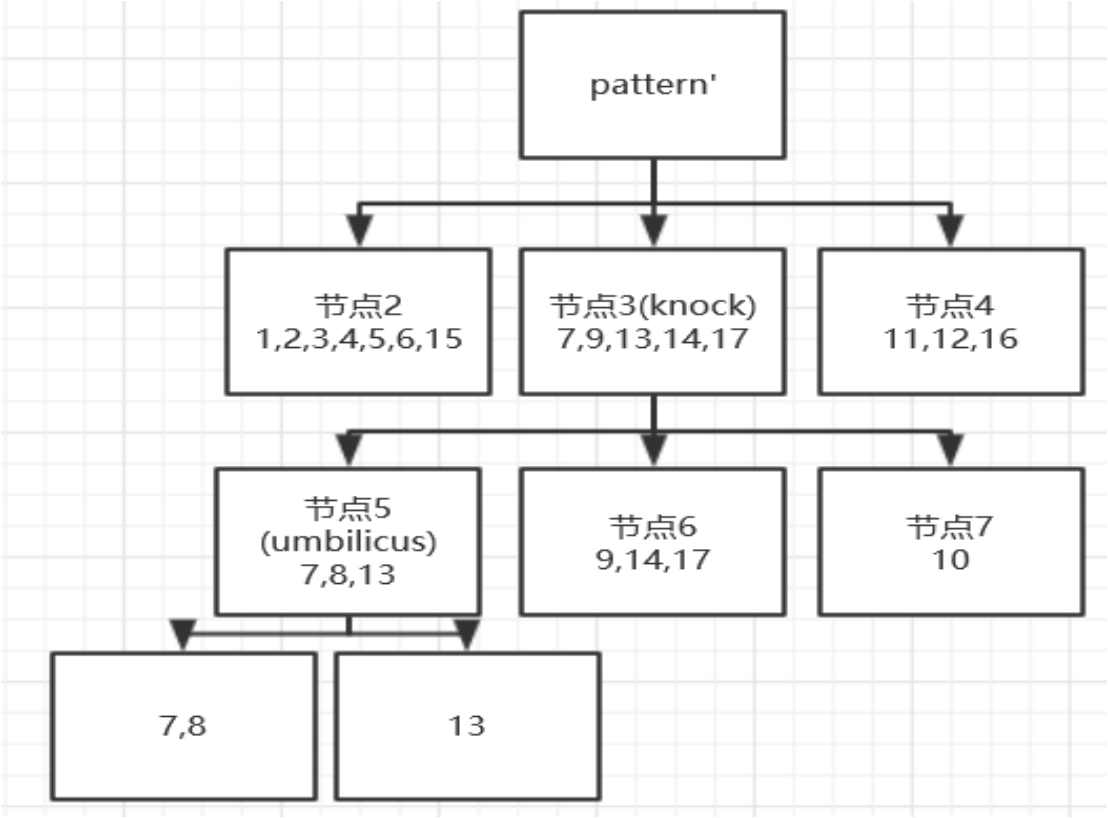
Ent(D1) = 0;

Ent(D2) =0;

Gain(D, umbilicus) = 0.98523;

...

比较可知，umbilicus 增益最大，{7,8,13}按此属性划分，决策树为



4.8 多变量处理决策树

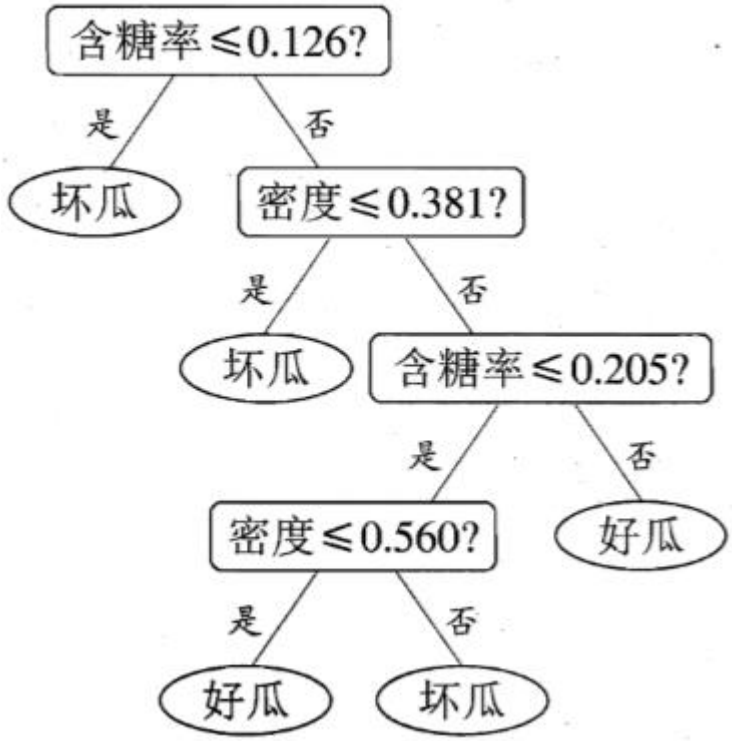
上述例子多是二分类，若是多分类，需用到多变量决策树。示例如下：

训练集样本 D

依据上述数据求增益，决策树如下

#	density	suger	label
1	0.697	0.46	1
2	0.774	0.376	1
3	0.634	0.264	1
4	0.608	0.318	1
5	0.556	0.215	1
6	0.403	0.237	1
7	0.481	0.149	1
8	0.437	0.211	1

9	0.666	0.091	0
10	0.243	0.267	0
11	0.245	0.057	0
12	0.343	0.099	0
13	0.639	0.161	0
14	0.657	0.198	0
15	0.36	0.37	0
16	0.593	0.042	0
17	0.719	0.103	0



则决策树对应的分类边界为

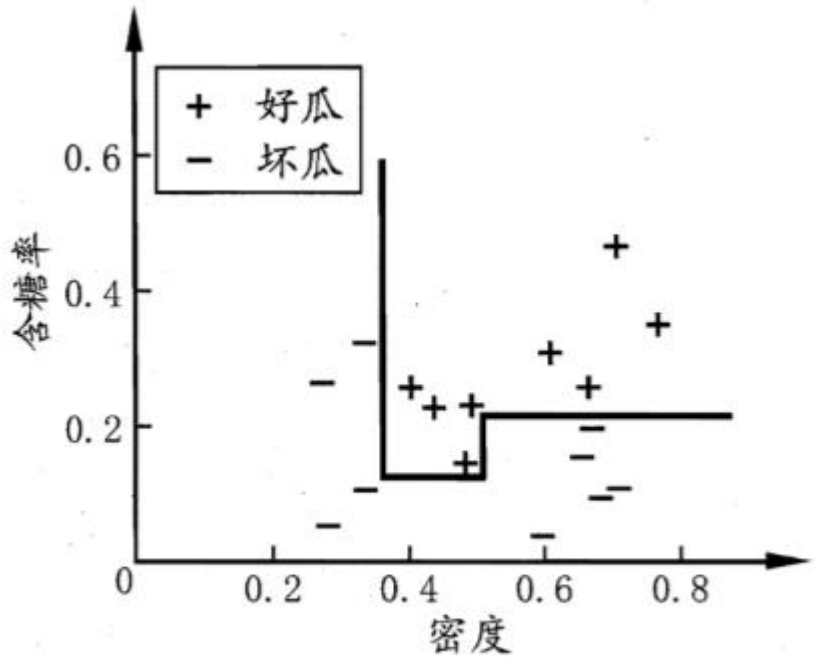
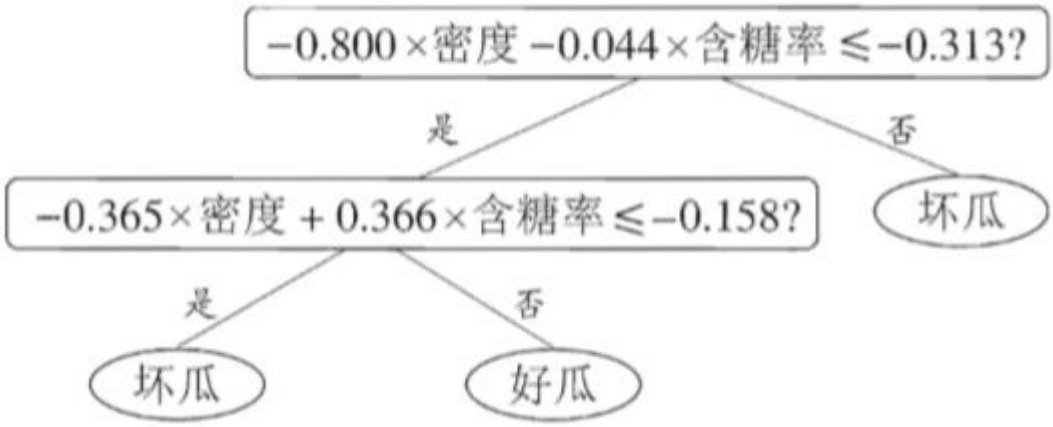


图 4.11　图 4.10 决策树对应的分类边界

分析：对大量连续性数据而言，上述分类边界训练开销过大，验证开销也很大。如果能用斜边界即斜划分，时间将大大缩短，方法是对属性组合求增益，然后分类：



以上即多变量决策树，可实现斜划分解决复杂的多分类问题。最终分类结果如下图多变量决策树（线性分类器）

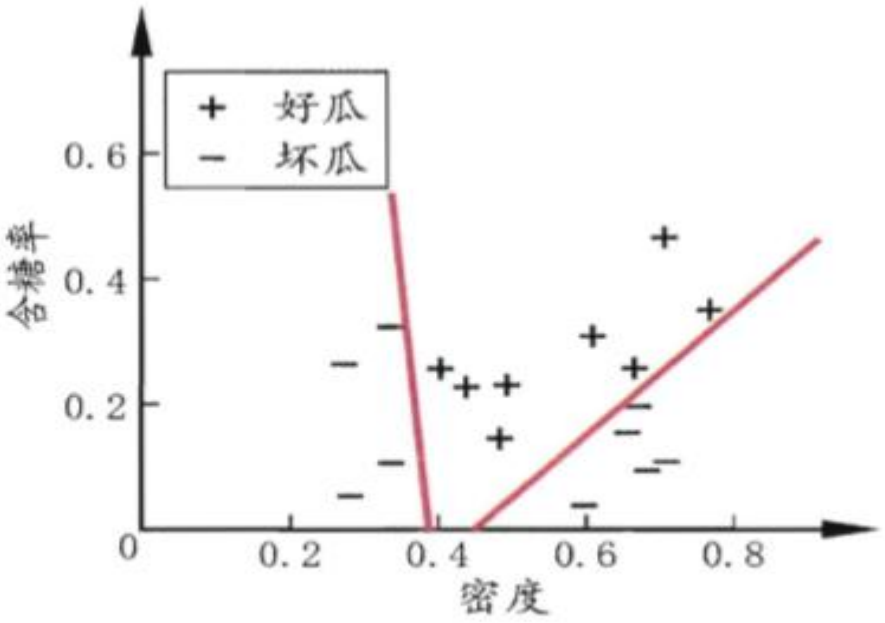


图 4.14 图 4.13 多变量决策树对应的分类边界

至此，决策树算法总结完毕。