中山大學
SUN YAT-SEN UNIVERSITY

第6章 决策树

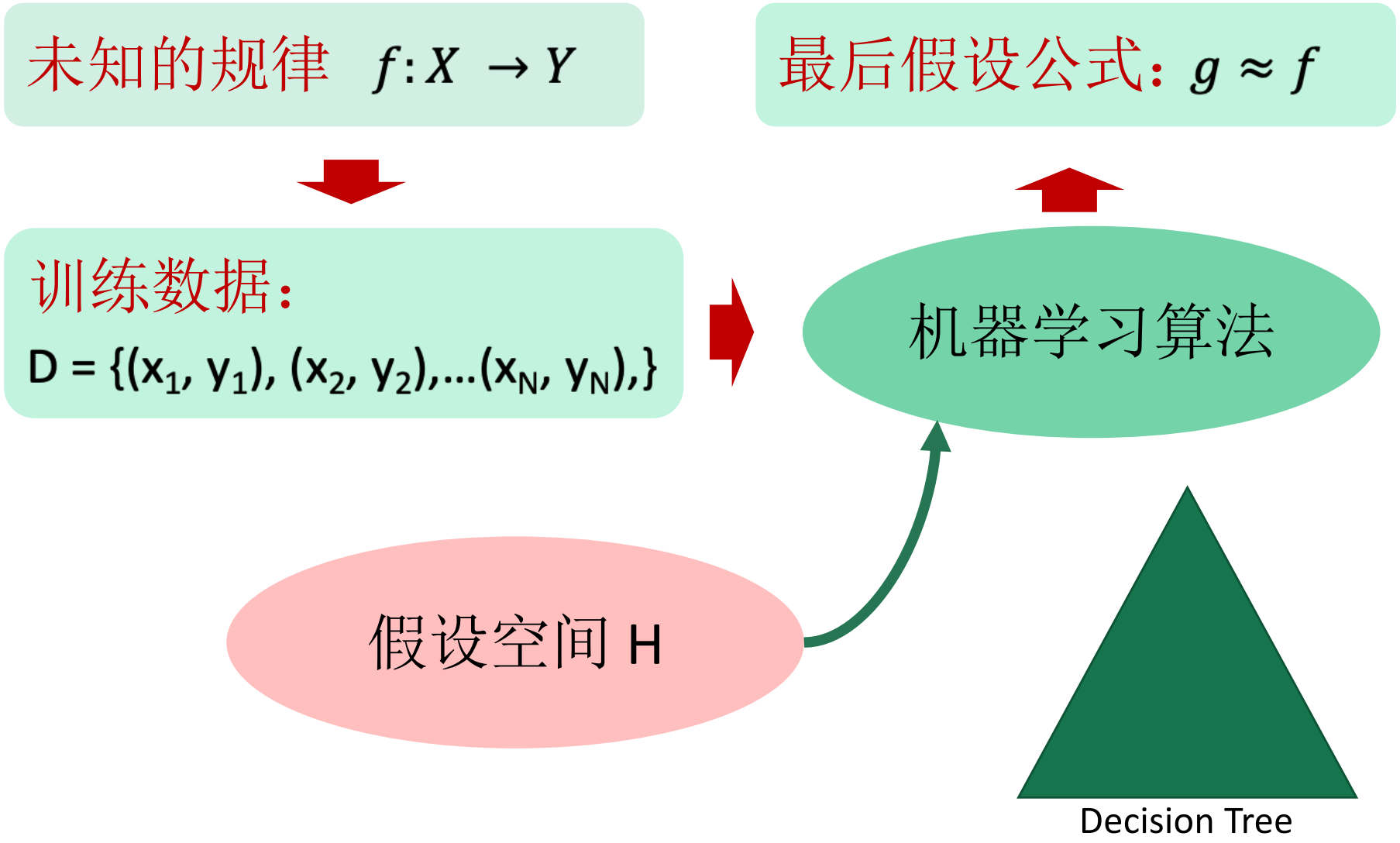
1. 基本流程
2. 划分选择

中山大學
SUN YAT-SEN UNIVERSITY

第6章 决策树

1. 基本流程
2. 划分选择

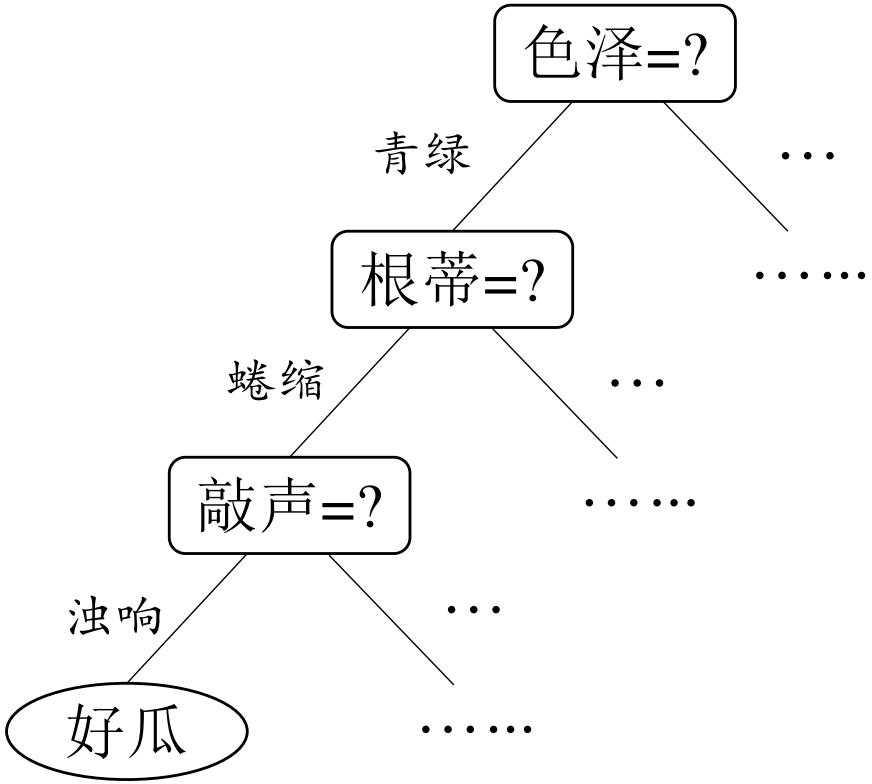
机器学习过程


未知的规律 $f: X \rightarrow Y$
最后假设公式：$g \approx f$
训练数据：
$D = \{(x_1, y_1), (x_2, y_2), \ldots (x_N, y_N),\}$
机器学习算法
假设空间 H
Decision Tree

概念——决策树

- 决策树（decision tree）

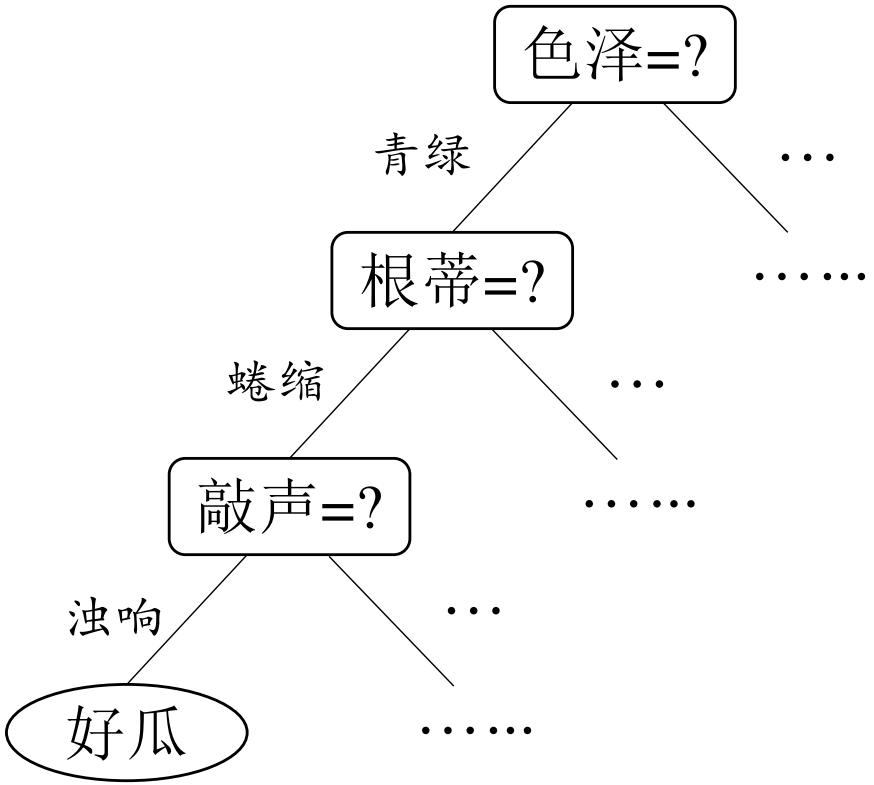
  决策树是一种基本的分类与回归方法，决策树模型呈树形结构，主要由节点（根节点、内部节点和叶节点）和边组成。



概念——决策树

决策树（decision tree）

决策树是一种基本的分类与回归方法，决策树模型呈树形结构，主要由节点（根节点、内部节点和叶节点）和边组成。

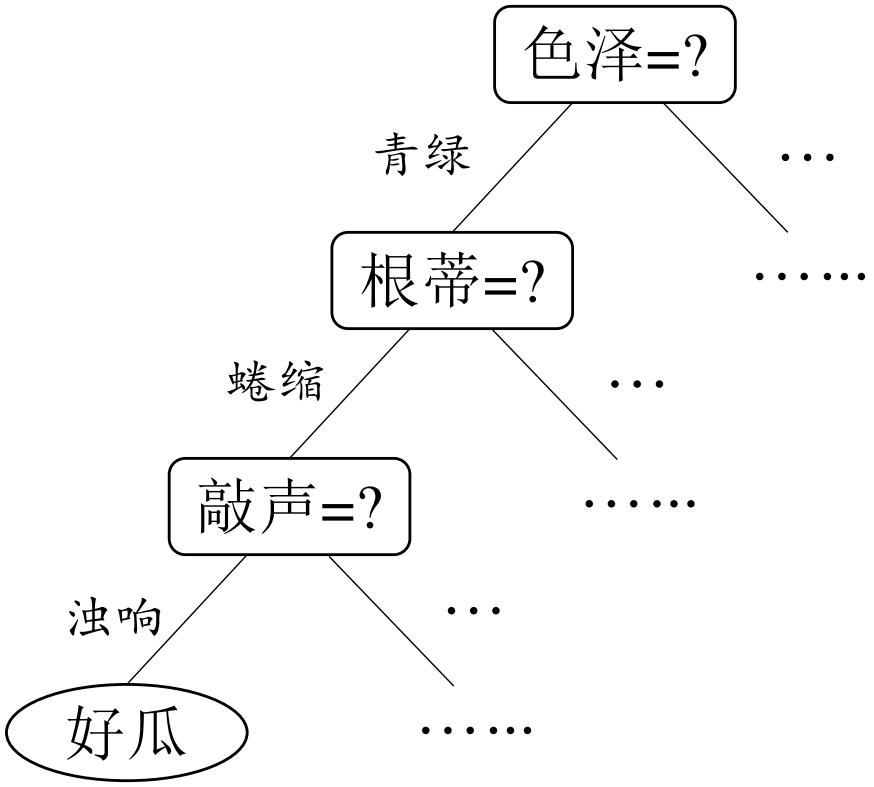


- 根节点（一个）
  包含样本全集
- 分支节点（若干）
  对应于一个属性的测试
- 叶节点（若干）
  对应于决策结果

概念——决策树（分类问题）

- 决策树（decision tree）

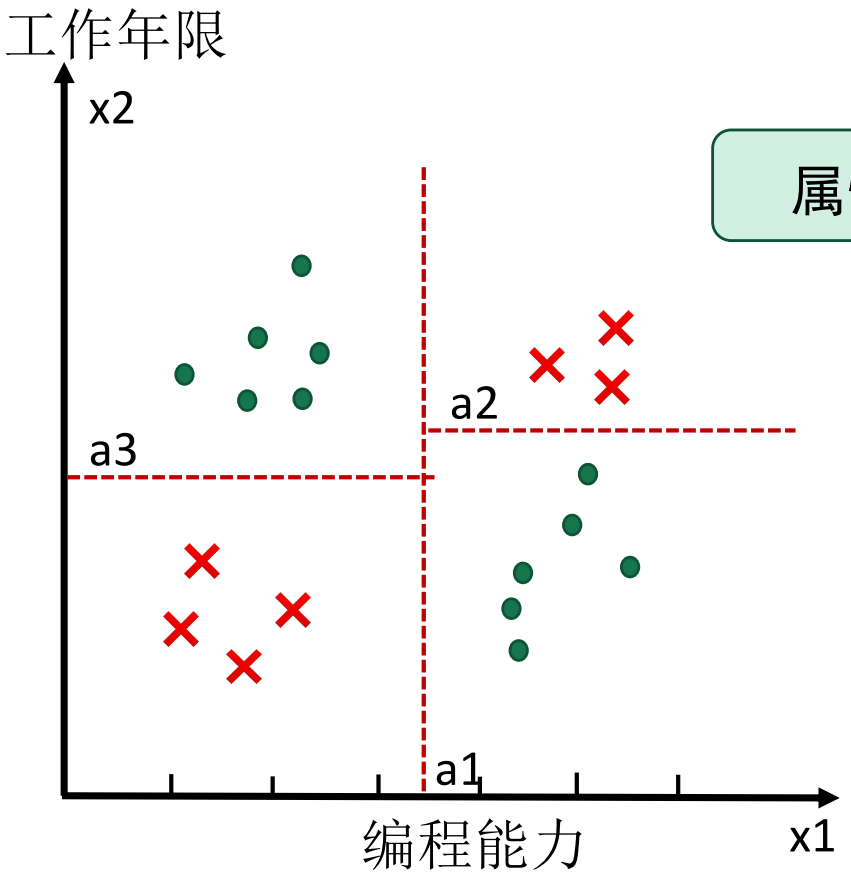
决策树是一种基本的分类与回归方法，决策树模型呈树形结构，主要由节点（根节点、内部节点和叶节点）和边组成。

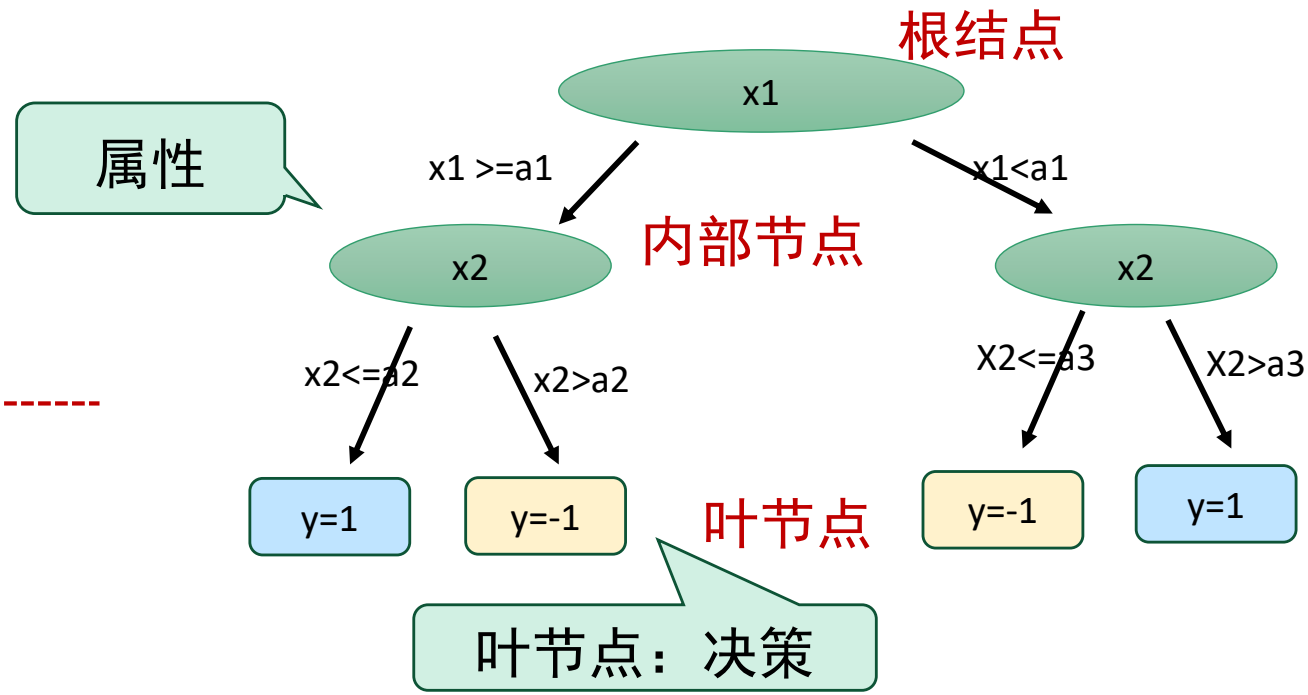


用决策树分类

- 从根节点开始，对示例的某一特征进行测试，根据测试结果，将示例分配到其子节点；这时，每一个子节点对应着该特征的一个取值
- 如此递归地对实例进行测试并分配，直至达到叶节点
- 最后将实例分配到叶节点的类中

决策树举例（连续值）





决策树即为轴平行分割属性空间

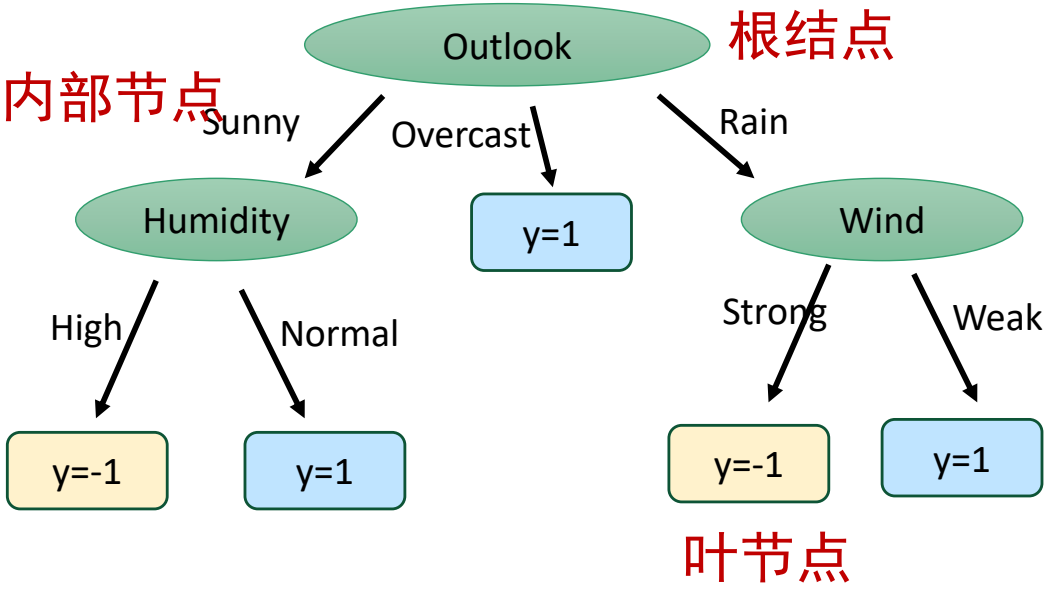
决策树举例（离散值）

Features				Classification
Outlook	Tempe rature	Humidity 湿度	Wind	Class Play Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast 多云	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

Feature： Outlook, Temperature, Humidity, Wind

训练数据：

$$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), .. (\boldsymbol{x}_i, y_i), \dots, (\boldsymbol{x}_m, y_m)\}$$



概念——决策树（分类问题）

- 决策树（decision tree）

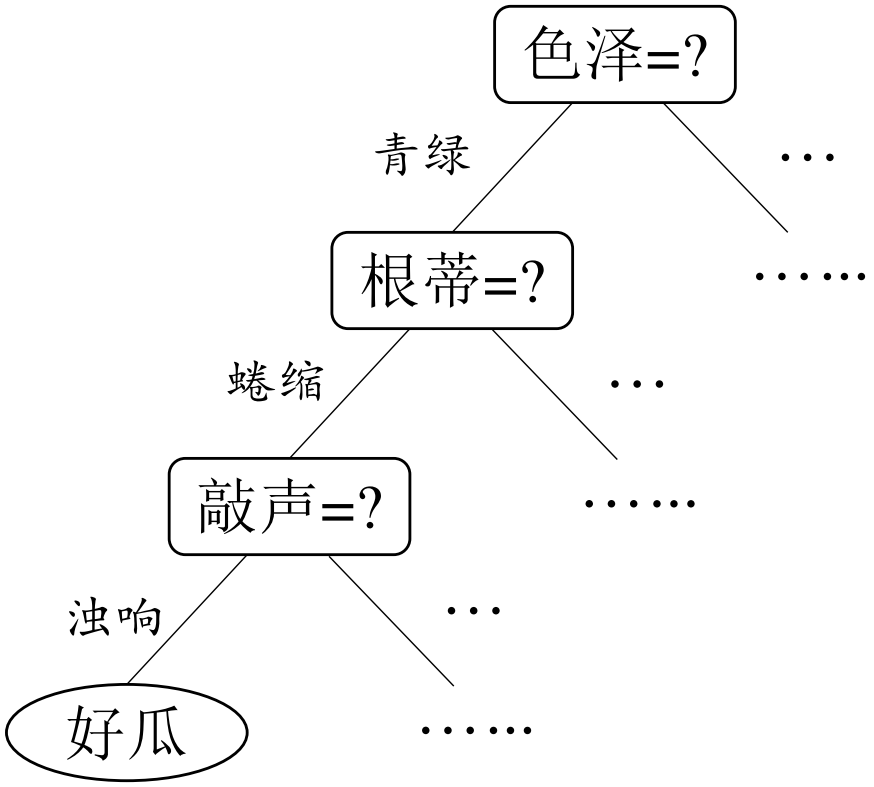
  - 决策过程中提出的每个判定问题都是对某个属性的“测试”
  - 每个测试的结果或是导出最终结论，或者导出进一步的判定问题，其考虑范围是在上次决策结果的限定范围之内
  - 决策过程的最终结论对应了我们所希望的判定结果
  - 从根结点到每个叶结点的路径对应了一个判定测试序列

由决策树的根节点到叶节点的每一条路径构成了一条规则。每一个示例都被一条规则且只被这条规则所覆盖。（互斥并且完备）

概念——决策树（分类问题）

决策树（decision tree）

决策树是一种基本的分类与回归方法，决策树模型呈树形结构，主要由节点（根节点、内部节点和叶节点）和边组成。



用决策树分类的关键问题

- 从根节点开始，对示例的某一特征进行测试(??)，根据测试结果，将示例分配到其子节点；这时，每一个子节点对应着该特征的一个取值
- 如此递归地对实例进行测试并分配，直至达到叶节点(??)
- 最后将实例分配到叶节点的类中

决策树学习（分类问题）

基本思想

数据：$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_m, y_m)\}$

其中$\boldsymbol{x}_i = \{x_{i1}, x_{i2}, \cdots, x_{id}\} \in \mathbb{R}^d, y_i \in \{1,2,\cdots,K\}$为类标记

模型： 分类决策树模型（一种由节点和有向边组成的用于描述对实例进行分类的树形结构）

策略：找到一个与训练数据矛盾较小同时泛化能力较强的决策树

算法：启发式方法（ID3、C4.5、CART等）

- 通常是一个递归地选择最优特征，并根据该特征对训练数据进行分割，使得对各个子数据集有一个最好的分类的过程。

决策树学习（分类问题）

基本思想

算法：启发式方法（ID3、C4.5、CART等）

- 开始，构建根节点，将所有训练数据都放在根节点。
- 同时，选择一个最优属性，按照这一属性将训练数据集分割成子集，使得各个子集有一个当前条件下最好的分类。
  - 如果，这些子集已经能够被基本正确分类，那么构建叶节点，并将这些子集分到所对应的叶节点中去；
  - 如果，还有子集不能被正确分类，那么构建中间节点，对这些子集选择新的最优属性，继续对其进行分割。
- 然后，如此递归地进行下去，直至所有训练数据子集被基本正确分类，或者没有合适的属性为止。
- 最后，构建叶节点，每个子集都被分到叶节点得到各自的类。

基本流程

年龄	Sal	信用等级
35	8000	1
36	6000	0
40	7000	1
25	2000	0
24	5000	1
20	4000	0
23	3600	1

Algorithm 1 决策树学习基本算法

输入:

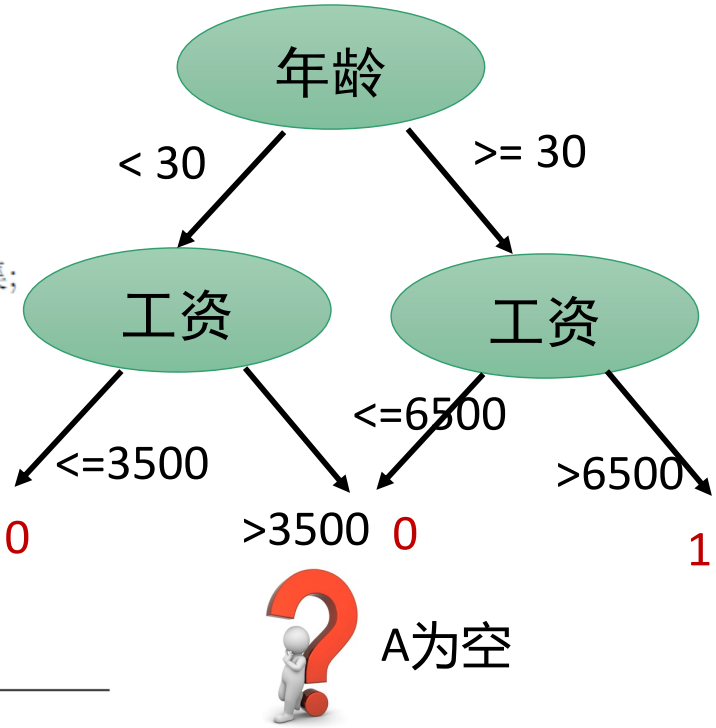
- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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** $A = \emptyset$ **OR** D 中样本在 A 上取值相同 **then**
6: 将 node 标记叶结点，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 生成每一个分枝; 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: 将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; **return**
13: **else**
14: 以 TreeGenerate(D_v, $A - \{a_*\}$) 为分枝结点
15: **end if**
16: **end for**

输出: 以 node 为根结点的一棵决策树

（1）当前结点包含的样本全部属于同一类别



基本流程

年龄	Sal	信用等级
35	8000	1
36	6000	0
40	7000	1
25	2000	0
24	5000	1
20	4000	0
23	3600	1

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

过程: 函数 TreeGenerate(D, A)

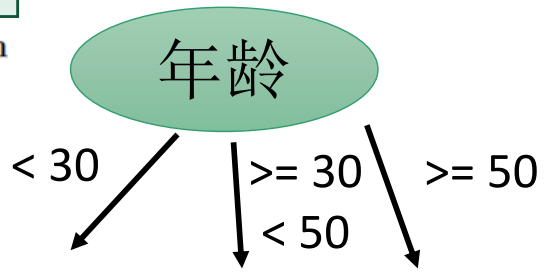
```
1: 生成结点 node;
2: if D 中样本全属于同一类别 C then
3:     将 node 标记为 C 类叶结点; return
4: end if
5: if A = ∅ OR D 中样本在 A 上取值相同 then
6:     将 node 标记叶结点，其类别标记为 D 中样本数最多的类; return
7: end if
8: 从 A 中选择最优划分属性 a_*;
9: for a_* 的每一个值 a_*^v do
10:    为 node 生成每一个分枝; 令 D_v 表示 D 中在 a_* 上取值为 a_*^v 的样本子集;
11:    if D_v 为空 then
12:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

（1）当前结点包含的样本全部属于同一类别

（2）当前属性集为空，或所有样本在所有属性上取值相同

对于每一个子集，年龄<30, 年龄>=30 两个子集



基本流程

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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 = ∅ OR D 中样本在 A 上取值相同 then
6:     将 node 标记叶结点，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 生成每一个分枝; 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:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

（1）当前结点包含的样本全部属于同一类别

（2）当前属性集为空，或所有样本在所有属性上取值相同

对于每一个子集，年龄<30，(3, 50)，>50 三个子集

（3）当前结点包含的样本集合为空

中山大學
SUN YAT-SEN UNIVERSITY

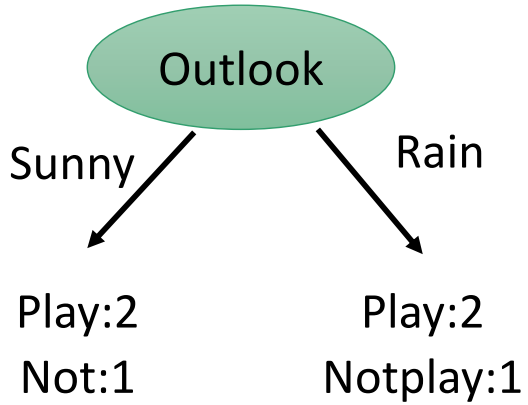
第6章 决策树

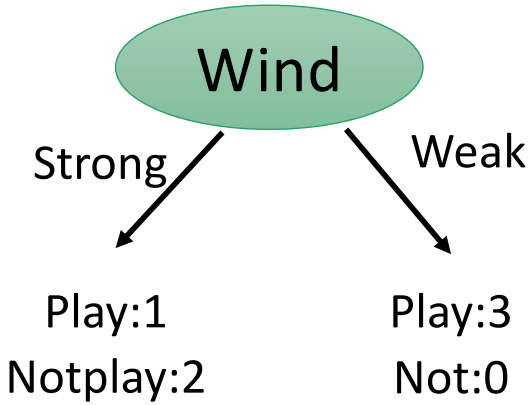
1. 基本流程
2. 划分选择

划分选择

- 决策树学习的关键在于如何选择最优划分属性。一般而言，随着划分过程不断进行，我们希望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

Outlook	Wind	Humidity	Play or Not
Sunny	Weak	High	1
Sunny	Strong	High	-1
Sunny	Weak	Normal	1
Rain	Strong	Normal	1
Rain	Weak	High	1
Rain	Strong	High	-1





经典的属性划分方法：

- 信息增益 (Information gain ID3)
- 增益率 (Gain ratio C4.5)
- 基尼指数 (Gini index CART)

划分选择--信息增益(ID3)

信息熵

- 信息的度量很难
  - 1948年香农在他著名的论文“通信的数学原理”中提出了信息熵，解决了信息的度量问题，量化出信息的作用
  - 熵的概念最早起源于物理学，用于度量一个热力学系统的无序程度。在信息论里面，熵是对不确定性的测量。
- 信息量就等于不确定性的多少
- 如何量化信息的度量呢?

划分选择——信息增益

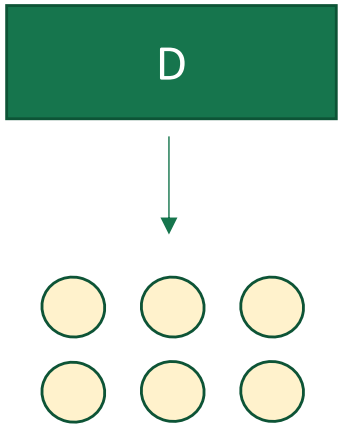
信息熵

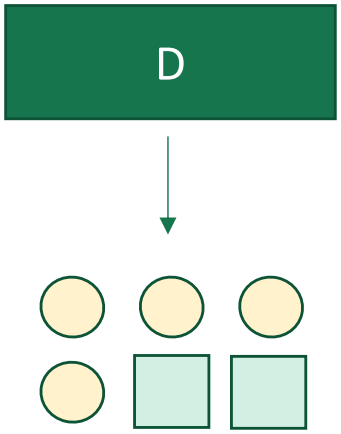
“信息熵”是度量样本集合纯度最常用的一种指标，假定当前样本集合D中第k类样本所占的比例为$p_k(k = 1,2\cdots,|\mathcal{Y}|)$，则$D$的信息熵定义为

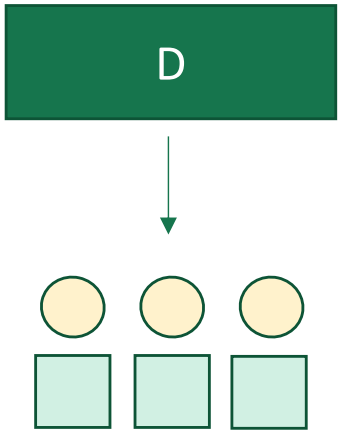
$$\mathrm{Ent}(D) = -\sum_{k=1}^{|\mathcal{Y}|} p_k \log_2 p_k$$

- $\mathrm{Ent}(D)$的值越小，则D的纯度越高
- 计算信息熵时约定：若$p_k = 0$，则$p_k \log_2 p_k = 0$
- $\mathrm{Ent}(D)$的最小值为0，最大值为$\log_2|\mathcal{Y}|$

信息增益-举例







$$Ent(D) = -(\frac{0}{6}\log_2\frac{0}{6} + \frac{6}{6}\log_2\frac{6}{6}) = 0$$

$$Ent(D) = -(\frac{3}{6}\log_2\frac{3}{6} + \frac{3}{6}\log_2\frac{3}{6}) = 1$$

$$Ent(D) = -(\frac{4}{6}\log_2\frac{4}{6} + \frac{2}{6}\log_2\frac{2}{6}) = 0.92$$

信息熵与数据纯度

- 假设有一个数据集D，分为正例和反例，计算下面三种情况的信息熵
  - （1）20%正例，80%反例子

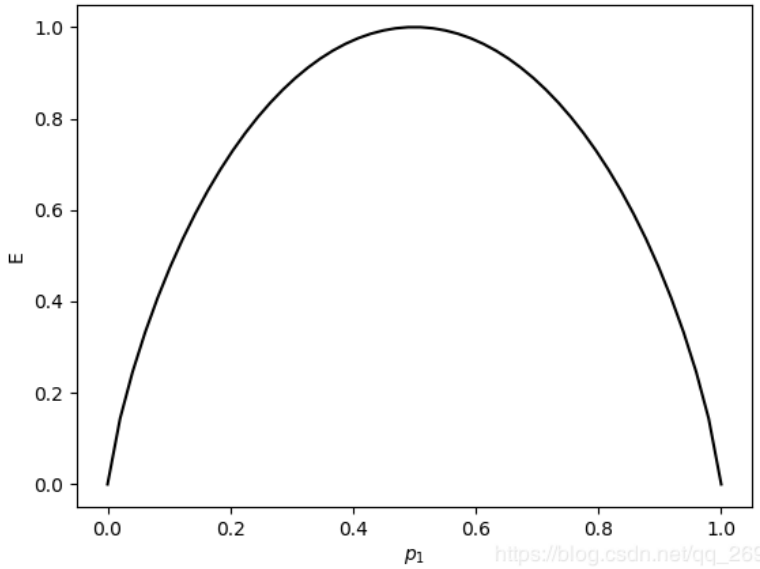
$$-\frac{2}{10}\log_2 0.2 \ -\frac{8}{10}\log_2 0.8 = 0.722$$

  - （1）50%正例，50%反例子

$$-\frac{5}{10}\log_2 0.5 \ -\frac{5}{10}\log_2 0.5 = 1$$

  - （1）80%正例，20%反例子

$$-\frac{8}{10}\log_2 0.8 \ -\frac{2}{10}\log_2 0.2 = 0.722$$



$$\text{Ent} = -p_1 \log_2 \ p_1 - (1-p_1)\log_2(1-p_1)$$

数据纯度越高时，信息熵越小

Features				Classification
Outlook	Temperature	Humidity	Wind	Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

当前样本集合D的信息熵

14个训练样本

分类数： 2 $|Y| = 2$

$$p_1 = \frac{9}{14}$$

$$p_2 = \frac{5}{14}$$

$$Ent(D) = -(p_1 \log_2 p_1 + p_2 \log_2 p_2)$$

$$= -(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14})$$

$$= 0.94$$

数据集的信息熵

信息熵实例

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

该数据集包含 17 个训练样本，$|\mathcal{Y}| = 2$，其中正例占 $p_1 = \frac{8}{17}$，反例占 $p_2 = \frac{9}{17}$，计算得到根结点的信息熵为？？？

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

划分选择——信息增益

◘ 数据集的信息熵

数据：$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_m, y_m)\}$

其中$\boldsymbol{x}_i = \{x_{i1}, x_{i2}, \cdots, x_{id}\} \in \mathbb{R}^d, y_i \in \{1,2, \cdots, K\}$为类标记

- 训练数据集D中有K个类$C_k, k = 1,2, \cdots, K$，$|D|$为总样本个数，$|C_k|$为属于第k类的样本个数，有$|D| = \sum_{k=1}^{K} |C_k|$。
- 当前样本集合D中第k类样本所占的比例为$p_k = |C_k|/|D|, (k = 1,2 \cdots, K)$，则$D$的信息熵定义为

$$\mathrm{Ent}(D) = -\sum_{k=1}^{K} \frac{|C_k|}{|D|} \log_2 \frac{|C_k|}{|D|}$$

$\mathrm{Ent}(D)$ 表示数据集的“纯度”以及对数据集进行分类的不确定性

划分选择——信息增益

- 信息增益（因某一属性使数据集分类的“纯度提升”的程度）

离散属性a有V个可能的取值$\{a^1, a^2, \cdots, a^V\}$，用$a$来进行划分，则会产生$V$个分支结点，其中第$v$个分支结点包含了$D$中所有在属性$a$上取值为$a^v$的样本，记为$D^v$。则可计算出用属性$a$对样本集$D$进行划分所获得的“信息增益”：

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

一般而言，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

Features				Classification
Outlook	Temperature	Humidity	Wind	Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

Outlook

v = 3

$Ent(D^1)$

$= -(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}) = 0.97$

Features				Classification
Outlook	Temperature	Humidity	Wind	Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

Outlook

v = 3

$$Ent(D^1) = -(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}) = 0.97$$

$$Ent(D^2) = -(\frac{4}{4}\log_2\frac{4}{4} + 0\log_2\frac{0}{4}) = 0$$

Features				Classification
Outlook	Temperature	Humidity	Wind	Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

Outlook

v = 3

$$Ent(D^1) = -(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}) = 0.97$$

$$Ent(D^2) = -(\frac{4}{4}\log_2\frac{4}{4} + 0\log_2\frac{0}{4}) = 0$$

$$Ent(D^3) = -(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}) = 0.97$$

$$Gain(D, o) = Ent(D) - \frac{5}{14}Ent(D^1)$$

$$-\frac{4}{14}Ent(D^2) \ -\frac{5}{14}Ent(D^3)$$

$$= 0.94 - \frac{5}{14} * 0.97 - 0 - \frac{5}{14} * 0.97$$

$$= 0.24$$

划分选择-信息增益

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

以属性“色泽”为例，其对应的 个数据子集分别为 3

(色泽=青绿) (色泽=乌黑) (色泽=浅白)

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

划分选择-信息增益

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

以属性“色泽”为例，其对应的 个数据子集分别为 3

(色泽=青绿) (色泽=乌黑) (色泽=浅白)

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

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

划分选择-信息增益

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

以属性“色泽”为例，其对应的 个数据子集分别为 3

(色泽=青绿) (色泽=乌黑) (色泽=浅白)

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

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

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

划分选择-信息增益

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

以属性“色泽”为例，其对应的 个数据子集分别为 3

(色泽=青绿) (色泽=乌黑) (色泽=浅白

$$\text{Gain}(D, \text{色泽}) = \text{Ent}(D) - \sum_{v=1}^{3} \frac{|D^v|}{|D|} \text{Ent}(D^v)$$

$$= 0.998 - \left(\frac{6}{17} \times 1.000 + \frac{6}{17} \times 0.918 + \frac{5}{17} \times 0.722\right)$$

$$= 0.109$$

$$\text{Gain}(D, \text{根蒂}) = 0.143$$

$$\text{Gain}(D, \text{敲声}) = 0.141$$

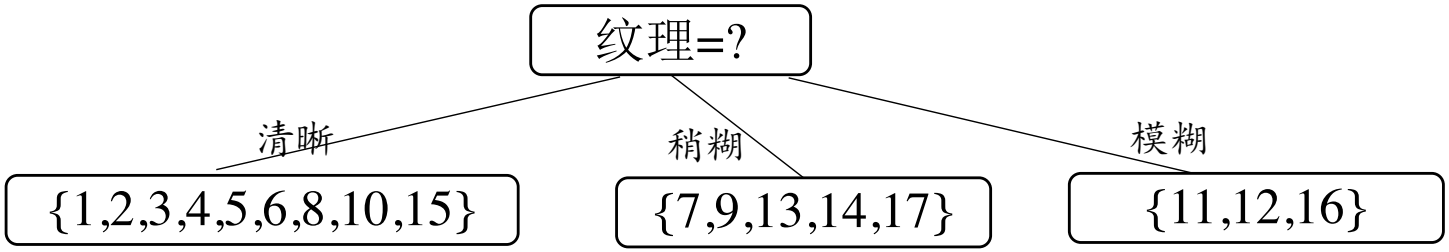
$$\text{Gain}(D, \text{纹理}) = 0.381$$

$$\text{Gain}(D, \text{脐部}) = 0.289$$

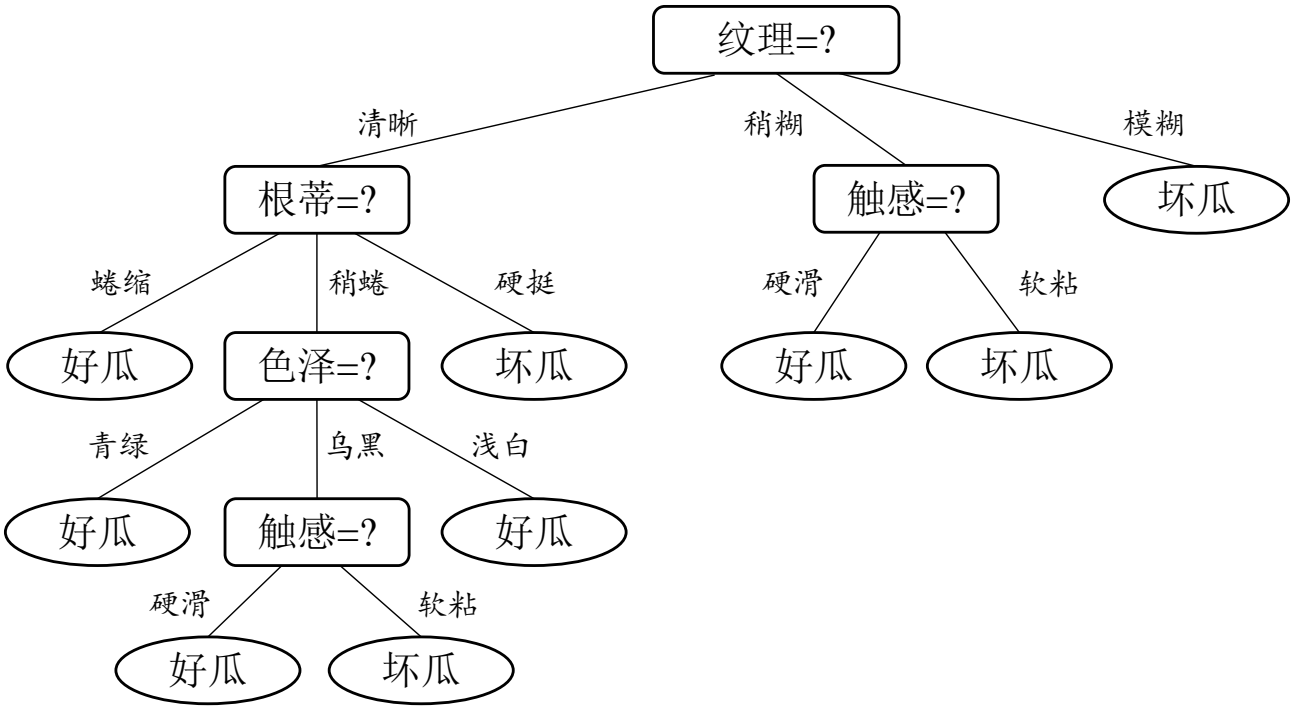
$$\text{Gain}(D, \text{触感}) = 0.006$$

划分选择-信息增益

- 显然，属性“纹理”的信息增益最大，其被选为划分属性



- 决策树学习算法将对每个分支结点做进一步划分，最终得到的决策树如图：



划分选择——信息增益

信息增益（ID3决策树）

离散属性a有V个可能的取值$\{a^1, a^2, \cdots, a^V\}$，用$a$来进行划分，则会产生$V$个分支结点，其中第$v$个分支结点包含了$D$中所有在属性$a$上取值为$a^v$的样本，记为$D^v$。则可计算出用属性$a$对样本集$D$进行划分所获得的“信息增益”：

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

$\frac{|D^v|}{|D|}$为分支结点权重，样本数越多的分支结点的影响越大

存在的问题：信息增益对可取值数目较多的属性有所偏好

划分选择-信息增益

- 若把“编号”也作为一个候选划分属性，则其信息增益一般远大于其他属性。显然，这样的决策树不具有泛化能力，无法对新样本进行有效预测

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

划分选择--增益率(C4.5)

划分选择——增益率

- 增益率（ C4.5决策树）

离散属性a有V个可能的取值$\{a^1, a^2, \cdots, a^V\}$，用$a$来进行划分，则会产生$V$个分支结点，其中第$v$个分支结点包含了$D$中所有在属性$a$上取值为$a^v$的样本，记为$D^v$。则可计算出用属性$a$对样本集$D$进行划分所获得的“增益率”：

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

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

属性a的固有值，属性a的可能取值越多则$\text{IV}(a)$通常会越大

Features				Classification
Outlook	Temperature	Humidity	Wind	Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

Outlook v = 3

$$Ent(D) = 0.94$$

$$Gain(D, o) = 0.24$$

$$IV(o) = -(\frac{5}{14}\log_2\frac{5}{14} + \frac{4}{14}\log_2\frac{4}{14} + \frac{5}{14}\log_2\frac{5}{14})$$

$$= 1.57$$

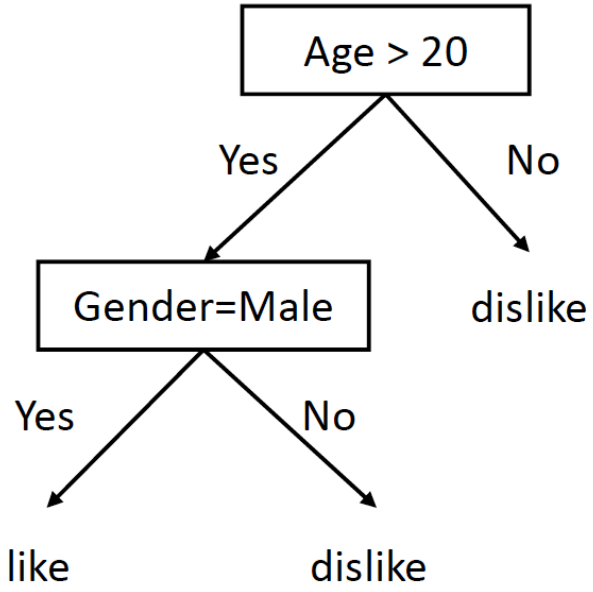
$$Gain\_ratio(D, o) = \frac{Gain(D, o)}{IV(o)}$$

$$= 0.15$$

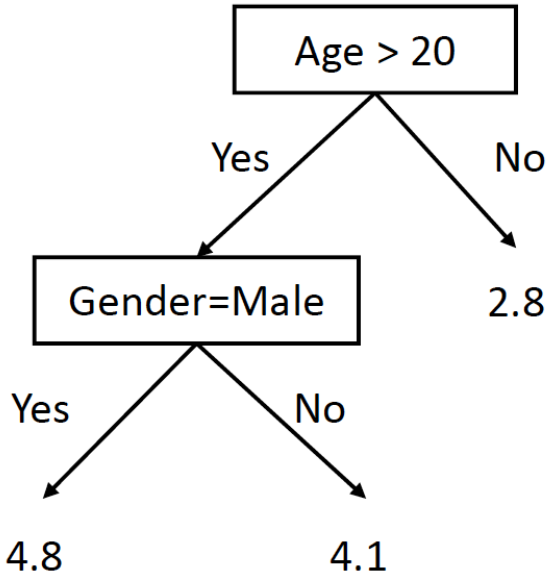
划分选择--基尼指数(CART)

CART (Classification and Regression Tree)

- CART既能是分类树，又可以做回归树
  - 分类树
    - 划分选择--基尼指数
  - 回归树
    - 最小方差
  - 二叉树



For example: predict whether the user like a move



For example: predict the user's rating to a movie

划分选择-基尼指数

Outlook	Temperature	Humidity	Wind	Yes or No
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Midd	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Midd	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Midd	Normal	Weak	Yes
Sunny	Midd	Normal	Strong	Yes
Overcast	Midd	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Midd	High	Strong	No

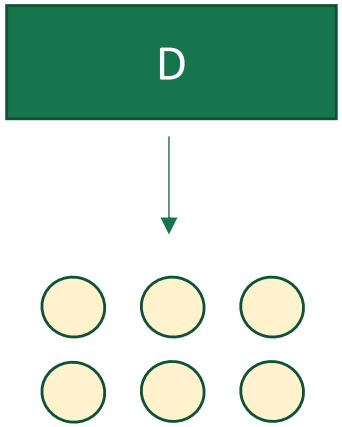
- 数据集 D 的纯度可用“基尼值”来度量

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

$\mathrm{Gini}(D)$ 越小，数据集 D 的纯度越高

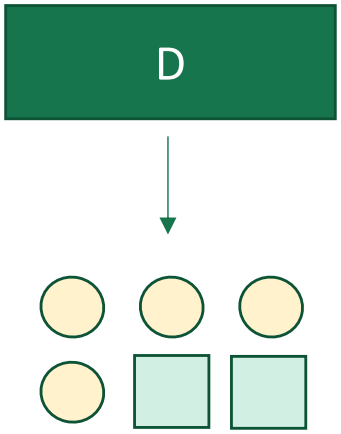
$$\mathrm{Gini}(D) = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2 = 0.33$$

划分选择-基尼指数



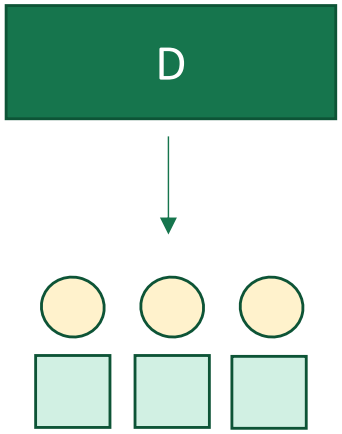
$$\text{Gini}(D) = 1 - \left(\frac{0}{6}\right)^2 - \left(\frac{6}{6}\right)^2$$

$$= 0$$



$$\text{Gini}(D) = 1 - \left(\frac{4}{6}\right)^2 - \left(\frac{2}{6}\right)^2$$

$$= 0.44$$



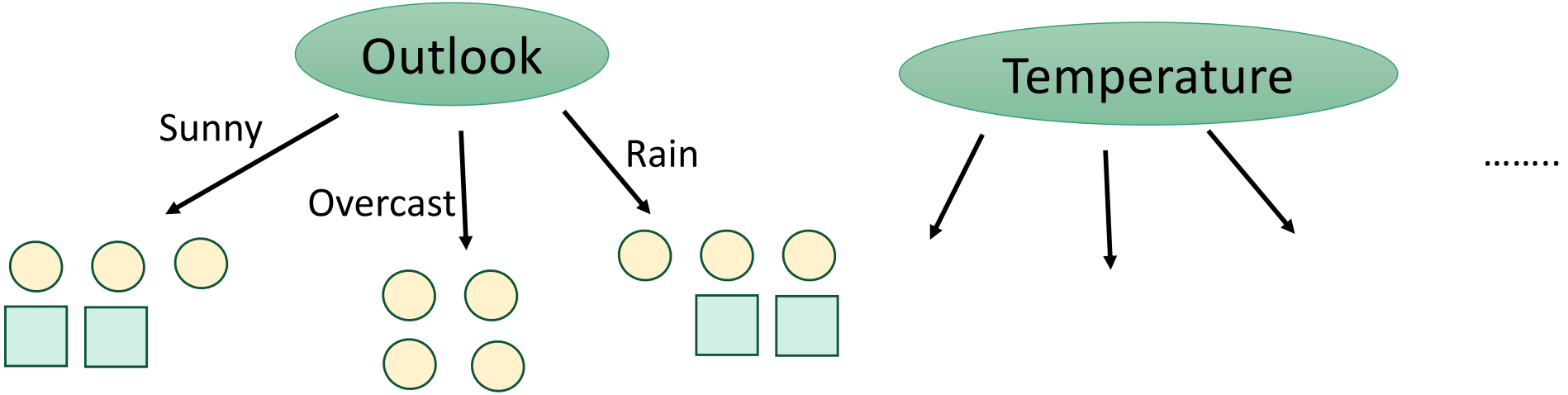
$$\text{Gini}(D) = 1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2$$

$$= 0.5$$

划分选择-基尼指数

- 属性 a 的基尼指数定义为：

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



划分选择-基尼指数

- 属性 a 的基尼指数定义为：

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

- 应选择那个使划分后基尼指数最小的属性作为最优划分属性，即

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

- CART [Breiman et al., 1984]采用“基尼指数”来选择划分属性

决策树学习（分类问题）

基本思想

数据：$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_m, y_m)\}$

其中$\boldsymbol{x}_i = \{x_{i1}, x_{i2}, \cdots, x_{id}\} \in \mathbb{R}^d, y_i \in \{1,2, \cdots, K\}$为类标记

模型：分类决策树模型（一种由节点和边组成的用于描述对实例进行分类的树形结构）

策略：找到一个与训练数据矛盾较小同时泛化能力较强的决策树（也可采用正则化的对数似然函数来作为损失函数）

算法：启发式方法（ID3、C4.5、CART等）

- ID3决策树学习算法[Quinlan, 1986]以信息增益为准则来选择划分属性
- C4.5决策树学习算法[Quinlan, 1993] 先从候选划分属性中找出信息增益高于平均水平的属性，再从中选取增益率最高的
- CART决策树学习算法[Breiman et al., 1984]采用基尼指数来选择划分属性

决策树学习（分类问题）

基本思想

算法：启发式方法（ID3、C4.5、CART等）

- 开始，构建根节点，将所有训练数据都放在根节点。
- 同时，选择一个最优属性，按照这一属性将训练数据集分割成子集，使得各个子集有一个当前条件下最好的分类。
  - 如果，这些子集已经能够被基本正确分类，那么构建叶节点，并将这些子集分到所对应的叶节点中去；
  - 如果，还有子集不能被正确分类，那么构建中间节点，对这些子集选择新的最优属性，继续对其进行分割。
- 然后，如此递归地进行下去，直至所有训练数据子集被基本正确分类，或者没有合适的属性为止。
- 最后，构建叶节点，每个子集都被分到叶节点得到各自的类。

决策树学习（分类问题）

基本流程

Algorithm 1 决策树学习基本算法

输入:

- 训练集 $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$;
- 属性集 $A = \{a_1, \ldots, a_d\}$.

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 = ∅ OR D 中样本在 A 上取值相同 then
6:     将 node 标记叶结点，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 生成每一个分枝; 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:        将分枝结点标记为叶结点, 其类别标记为 D 中样本最多的类; return
13:    else
14:        以 TreeGenerate(D_v, A − {a_*}) 为分枝结点
15:    end if
16: end for
```

输出: 以 node 为根结点的一棵决策树

如何选择最后划分属性

（1）信息增益（最大）

（2）增益率（最大）

（3）基尼指数（最小）

如何判断达到叶节点，如何确定叶节点的类

（1）当前结点包含的样本全部属于同一类别

（2）当前属性集为空，或所有样本在所有属性上取值相同

（3）当前结点包含的样本集合为空

谢谢！