

# 决策树(Decision Tree)

# 一、分类(Classification)

- 1、分类的意义

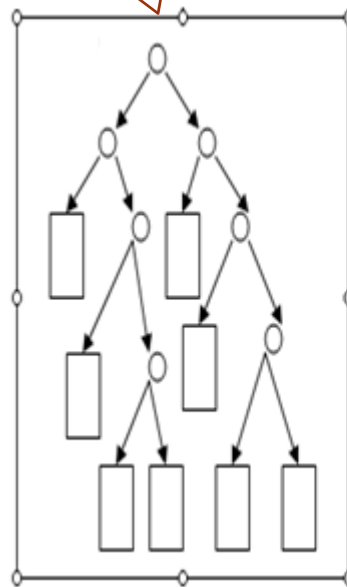
数据库

| 編號    | 性別     | 年齡  | 婚姻  | 家庭<br>人數 | 購買<br>RV房<br>車 |
|-------|--------|-----|-----|----------|----------------|
| A0001 | Male   | 45  | 未婚  | 1        | 是              |
| A0002 | Male   | 52  | 已婚  | 7        | 是              |
| A0003 | Female | 38  | 已婚  | 5        | 是              |
| A0004 | Male   | 25  | 已婚  | 5        | 否              |
| A0005 | Female | 48  | 已婚  | 4        | 是              |
| A0006 | Male   | 32  | 未婚  | 3        | 是              |
| A0007 | Female | 65  | 已婚  | 4        | 否              |
| A0008 | Male   | 33  | 已婚  | 3        | 是              |
| A0009 | Male   | 45  | 已婚  | 4        | 是              |
| A0010 | Female | 52  | 未婚  | 1        | 是              |
| A0011 | Male   | 38  | 未婚  | 1        | 否              |
| ...   | ...    | ... | ... | ...      | ...            |
| Z0099 | Male   | 22  | 未婚  | 4        | 是              |

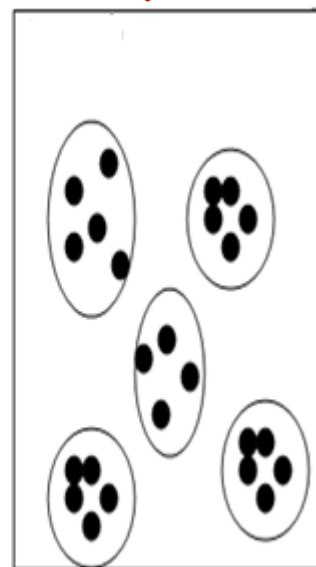
预测

了解类别属性  
与特征

分类模型—  
决策树



分类模型—  
聚类

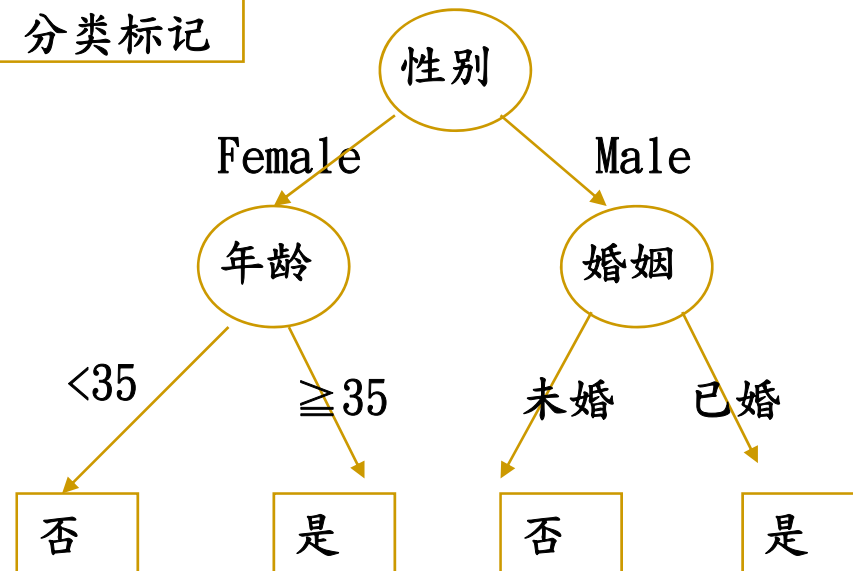


## 2、决策树分类技术

数据库

| 編號    | 性別     | 年齡  | 婚姻  | 家庭<br>人數 | 購買<br>RV房<br>車 |
|-------|--------|-----|-----|----------|----------------|
| A0001 | Male   | 45  | 未婚  | 1        | 是              |
| A0002 | Male   | 52  | 已婚  | 7        | 是              |
| A0003 | Female | 38  | 已婚  | 5        | 是              |
| A0004 | Male   | 25  | 已婚  | 5        | 否              |
| A0005 | Female | 48  | 已婚  | 4        | 是              |
| A0006 | Male   | 32  | 未婚  | 3        | 是              |
| A0007 | Female | 65  | 已婚  | 4        | 否              |
| A0008 | Male   | 33  | 已婚  | 3        | 是              |
| A0009 | Male   | 45  | 已婚  | 4        | 是              |
| A0010 | Female | 52  | 未婚  | 1        | 是              |
| A0011 | Male   | 38  | 未婚  | 1        | 否              |
| ...   | ...    | ... | ... | ...      | ...            |
| Z0099 | Male   | 22  | 未婚  | 4        | 是              |

分类标记



### 3、分类的程序

- 模型建立(Model Building)
- 模型评估(Model Evaluation)
- 使用模型(Use Model)

# 决策树分类的步骤

数据库

| 編號    | 性別     | 年齡  | 婚姻  | 家庭<br>人數 | 購買<br>RV房<br>車 |
|-------|--------|-----|-----|----------|----------------|
| A0001 | Male   | 45  | 未婚  | 1        | 是              |
| A0002 | Male   | 52  | 已婚  | 7        | 是              |
| A0003 | Female | 38  | 已婚  | 5        | 是              |
| A0004 | Male   | 25  | 已婚  | 5        | 否              |
| A0005 | Female | 48  | 已婚  | 4        | 是              |
| A0006 | Male   | 32  | 未婚  | 3        | 是              |
| A0007 | Female | 65  | 已婚  | 4        | 否              |
| A0008 | Male   | 33  | 已婚  | 3        | 是              |
| A0009 | Male   | 45  | 已婚  | 4        | 是              |
| A0010 | Female | 52  | 未婚  | 1        | 是              |
| A0011 | Male   | 38  | 未婚  | 1        | 否              |
| ...   | ...    | ... | ... | ...      | ...            |
| Z0099 | Male   | 22  | 未婚  | 4        | 是              |

训练样本(training samples)

测试样本(testing samples)

建立模型

评估模型

例：

资料

| 編號    | 性別     | 年齡  | 婚姻 | 家庭所得 | 購買RV房車 |
|-------|--------|-----|----|------|--------|
| A0001 | Male   | <35 | 未婚 | 高所得  | 否      |
| A0002 | Male   | <35 | 未婚 | 小康   | 否      |
| A0003 | Female | ≥35 | 已婚 | 高所得  | 是      |
| A0004 | Male   | ≥35 | 未婚 | 低所得  | 是      |
| A0005 | Female | ≥35 | 已婚 | 高所得  | 否      |
| A0006 | Male   | ≥35 | 已婚 | 低所得  | 否      |
| A0007 | Female | ≥35 | 未婚 | 小康   | 否      |
| A0008 | Male   | ≥35 | 已婚 | 高所得  | 是      |
| A0009 | Male   | <35 | 已婚 | 低所得  | 是      |

测试样本

|   |
|---|
| 是 |
| 是 |
| 否 |

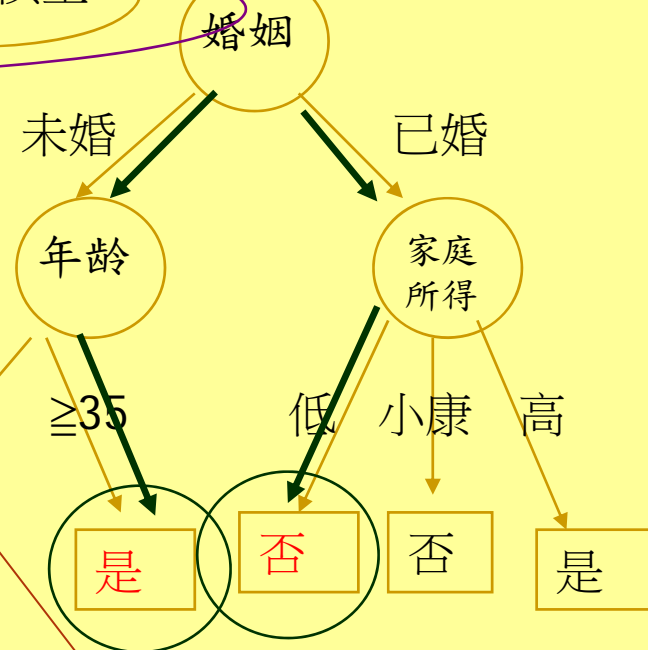
| 預測結果 |
|------|
| 錯誤   |
| 正確   |
| 錯誤   |

错误率为66.67%

修改模型

2.模型评估

1.建立模型



3.使用模型

| 編號    | 性別   | 年齡 | 婚姻 | 家庭所得 | 購買RV房車 |
|-------|------|----|----|------|--------|
| W0144 | Male | 55 | 已婚 | 高所得  | ?      |

## 4、分类算法的评估

- 预测的准确度：指模型正确地预测新的或先前未见过的数据的类标号的能力。
  - 训练测试法(training-and-testing)
  - 交叉验证法(cross-validation)
- 例如，十折交叉验证。即是将数据集分成十份，轮流将其中9份做训练1份做测试，10次的结果的均值作为对算法精度的估计，一般还需要进行多次10倍交叉验证求均值，例如10次10倍交叉验证，更精确一点。

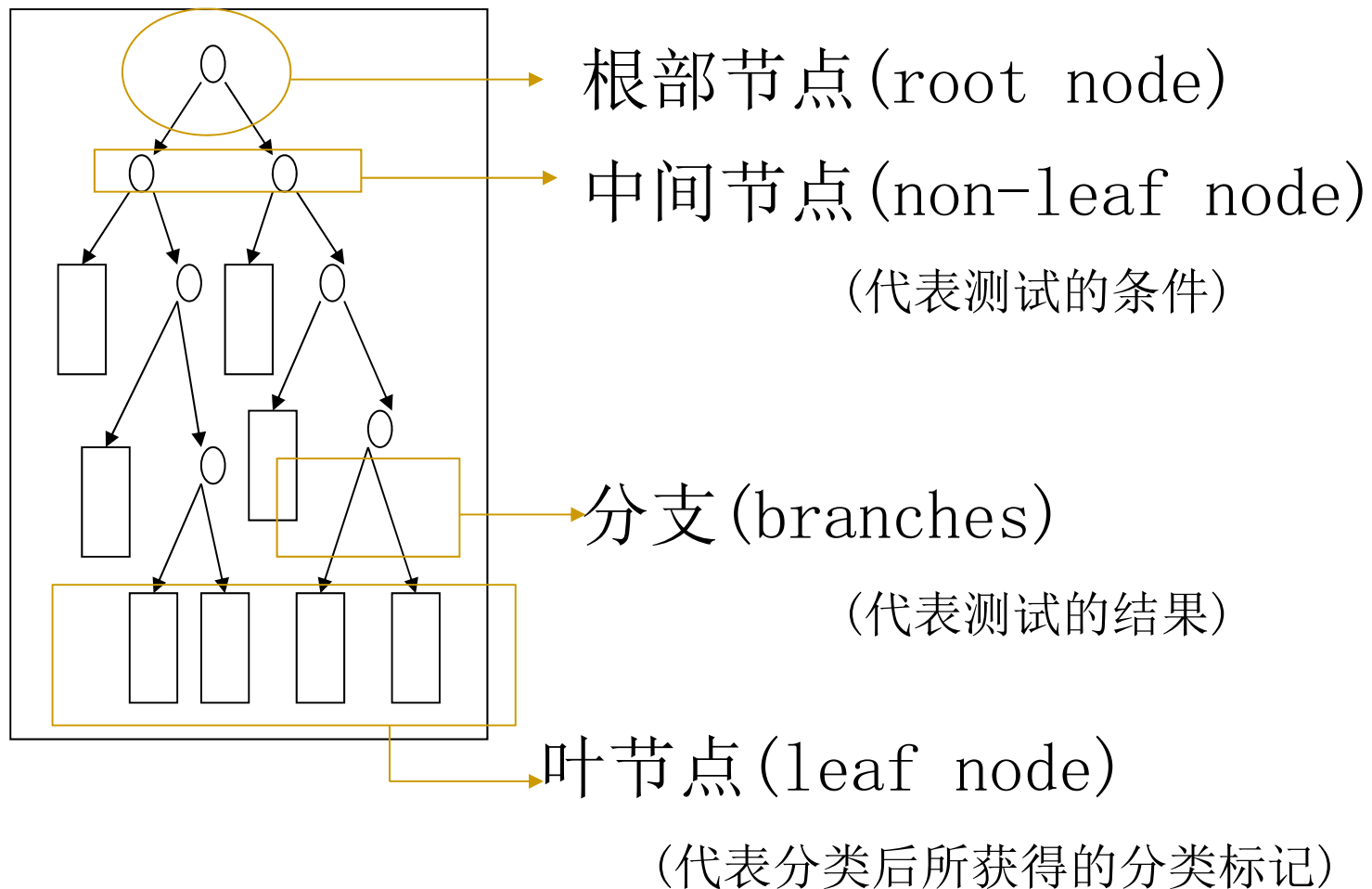
- 速度：指产生和使用模型的计算花费。
  - 建模的速度、预测的速度
- 强壮性：指给定噪声数据或具有缺失值的数据，模型正确预测的能力。
- 可诠释性：指模型的解释能力。



## 二、决策树(Decision Tree)

- 决策树归纳的基本算法是贪心算法，它以自顶向下递归各个击破的方式构造决策树。
  - 贪心算法：在每一步选择中都采取在当前状态下最好/优的选择。
- 在其生成过程中，分割方法即属性选择度量是关键。通过属性选择度量，选择出最好的将样本分类的属性。
- 根据分割方法的不同，决策树可以分为两类：基于信息论的方法（较有代表性的是ID3、C4.5算法等）和最小GINI指标方法（常用的有CART、SLIQ及SPRINT算法等）。

## (一) 决策树的结构



## (二) 决策树的形成

算法: `Generate_decision_tree` 由给定的训练数据产生一棵判定树。

输入: 训练样本 *samples*, 由离散值属性表示; 候选属性的集合 *attribute\_list*。

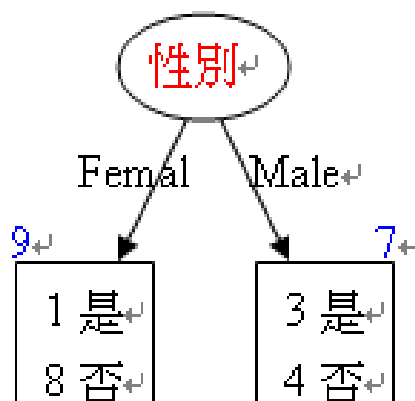
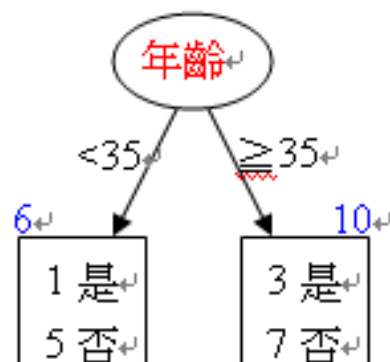
输出: 一棵判定树。

方法:

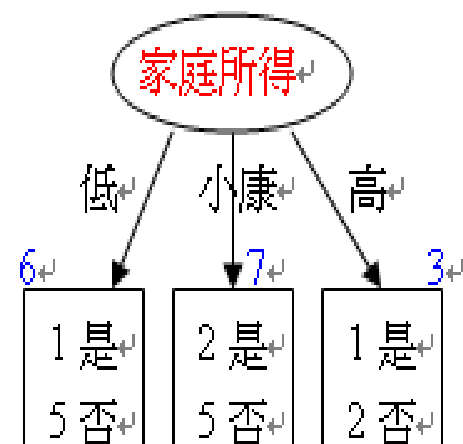
- (1) 创建节点 *N*;
- (2) if *samples* 都在同一个类 *C* then
- (3) 返回 *N* 作为叶节点, 以类 *C* 标记;
- (4) if *attribute\_list* 为空 then
- (5) 返回 *N* 作为叶节点, 标记为 *samples* 中最普通的类; //多数表决
- (6) 选择 *attribute\_list* 中具有最高信息增益的属性 *test\_attribute*;
- (7) 标记节点 *N* 为 *test\_attribute*;
- (8) for each *test\_attribute* 中的已知值  $a_i$  //划分 *samples*
- (9) 由节点 *N* 长出一个条件为  $test\_attribute = a_i$  的分枝;
- (10) 设  $s_i$  是 *samples* 中  $test\_attribute = a_i$  的样本的集合; //一个划分
- (11) if  $s_i$  为空 then
- (12) 加上一个树叶, 标记为 *samples* 中最普通的类;
- (13) else 加上一个由 `Generate_decision_tree( $s_i$ , attribute_list-test_attribute)` 返回的节点;

例：

| 年齡  | 性別     | 家庭所得 | 購買RV房車 |
|-----|--------|------|--------|
| <35 | Male   | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 低所得  | 否      |
| <35 | Male   | 高所得  | 否      |
| ≥35 | Female | 低所得  | 否      |
| <35 | Female | 低所得  | 否      |
| <35 | Female | 高所得  | 是      |
| ≥35 | Male   | 小康   | 是      |
| <35 | Male   | 高所得  | 否      |
| ≥35 | Female | 小康   | 否      |
| <35 | Male   | 低所得  | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Male   | 低所得  | 是      |
| ≥35 | Male   | 小康   | 是      |
| ≥35 | Female | 低所得  | 否      |



- 根部节点
- 中间节点
- 停止分支



### (三) ID3算法(C4.5,C5.0)

- Quinlan(1979)提出，以Shannon(1949)的信息论为依据。
- ID3算法的属性选择度量就是使用信息增益，选择最高信息增益的属性作为当前节点的测试属性。
- 信息论：若一事件有 $k$ 种结果，对应的概率为 $P_i$ 。则此事件发生后所得到的信息量 $I$ (视为Entropy)为：

$$I=-(p_1*\log_2(p_1)+ p_2*\log_2(p_2)+...+ p_k*\log_2(p_k))$$

Example 1:

- 设  $k=4 \rightarrow p_1=0.25, p_2=0.25, p_3=0.25, p_4=0.25$   
 $I=-(.25 * \log_2(.25) * 4)=2$

Example 2:

- 设  $k=4 \rightarrow p_1=0, p_2=0.5, p_3=0, p_4=0.5$   
 $I=-(.5 * \log_2(.5) * 2)=1$

Example 3:

- 设  $k=4 \rightarrow p_1=1, p_2=0, p_3=0, p_4=0$   
 $I=-(1 * \log_2(1))=0$

# 信息增益

设  $U$  为  $u$  个元组的集合, 类别属性中的分类有  $m$  个, 设  $u_i$  是分别属于这  $m$  个类的样本数,  $\frac{u_i}{u}$  是  $U$  中样本属于该分类的概率的估计值, 那么对于这个给定的样本分类的信息熵是

$$I(u_1, u_2, \dots, u_m) = - \sum_{i=1}^m \frac{u_i}{u} \log_2 \frac{u_i}{u}$$

具有值域  $\{a_1, a_2, \dots, a_v\}$  的属性  $A$  可以用来将  $U$  划分为子集  $\{U_1, U_2, \dots, U_v\}$ , 其中,  $U_j$  包含  $U$  中  $A$  值为  $a_j$  的那些样本, 设  $U_j$  包含第  $i$  类给定样本分类的  $u_{ij}$  个样本。则根据  $A$  划分的期望信息称作  $A$  的熵为

$$E(A) = \sum_{j=1}^v \frac{u_{1j} + \dots + u_{mj}}{u} I(u_{1j}, \dots, u_{mj})$$

根据  $A$  进行的划分获得的信息增益为

$$Gain(A) = I(u_1, u_2, \dots, u_m) - E(A)$$

# Example(Gain)

$n=16$

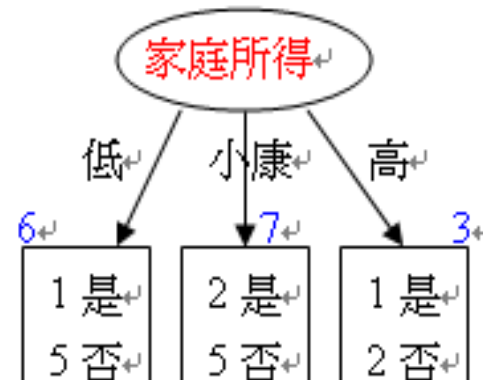
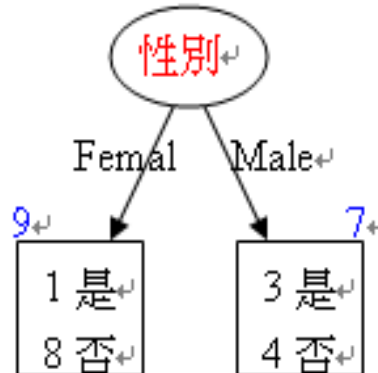
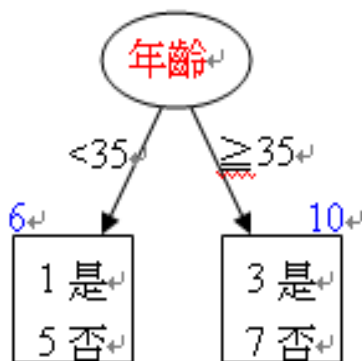
$n_1=4$

| 年齡  | 性別     | 家庭所得 | 購買RV房車 |
|-----|--------|------|--------|
| <35 | Male   | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 低所得  | 否      |
| <35 | Male   | 高所得  | 否      |
| ≥35 | Female | 低所得  | 否      |
| <35 | Female | 低所得  | 否      |
| <35 | Female | 高所得  | 是      |
| ≥35 | Male   | 小康   | 是      |
| <35 | Male   | 高所得  | 否      |
| ≥35 | Female | 小康   | 否      |
| <35 | Male   | 低所得  | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Male   | 低所得  | 是      |
| ≥35 | Male   | 小康   | 是      |
| ≥35 | Female | 低所得  | 否      |

$$I(16,4) = -((4/16) \cdot \log_2(4/16) + (12/16) \cdot \log_2(12/16)) = 0.8113$$

$$E(\text{年齡}) = (6/16) \cdot I(6,1) + (10/16) \cdot I(10,3) = 0.7946$$

$$\text{Gain}(\text{年齡}) = I(16,4) - E(\text{年齡}) = 0.0167$$



- $\text{Gain}(\text{年齡}) = 0.0167$
- $\text{Gain}(\text{性別}) = 0.0972$
- $\text{Gain}(\text{家庭所得}) = 0.0177$

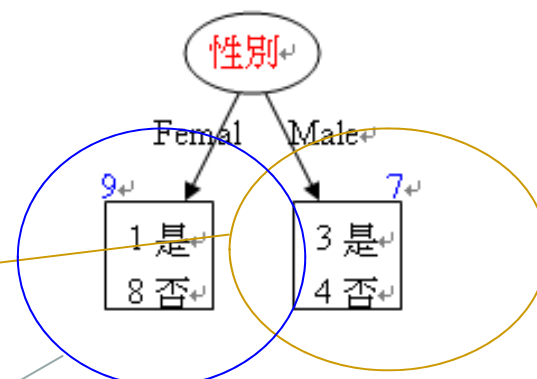
- Max: 作为第一个分类依据



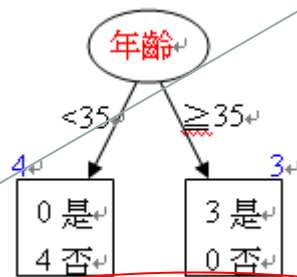
# Example(续)

| 年齡  | 性別   | 家庭所得 | 購買RV房車 |
|-----|------|------|--------|
| <35 | Male | 小康   | 否      |
| <35 | Male | 低所得  | 否      |
| <35 | Male | 高所得  | 否      |
| <35 | Male | 高所得  | 否      |
| ≥35 | Male | 小康   | 是      |
| ≥35 | Male | 小康   | 是      |
| ≥35 | Male | 低所得  | 是      |

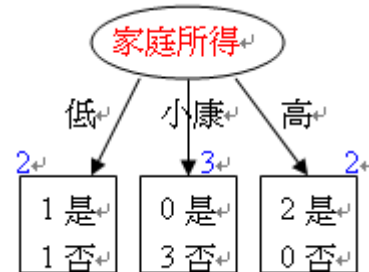
| 年齡  | 性別     | 家庭所得 | 購買RV房車 |
|-----|--------|------|--------|
| <35 | Female | 低所得  | 否      |
| <35 | Female | 高所得  | 是      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 低所得  | 否      |
| ≥35 | Female | 低所得  | 否      |
| ≥35 | Female | 低所得  | 否      |



$$I(7,3) = -((3/7) \cdot \log_2(3/7) + (4/7) \cdot \log_2(4/7)) = 0.9852$$

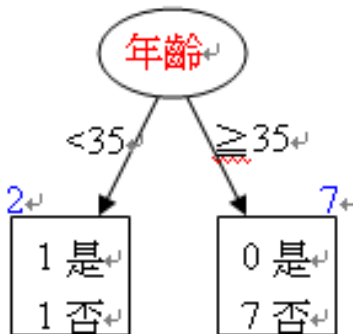


$$\text{Gain(年齡)} = 0.9852$$

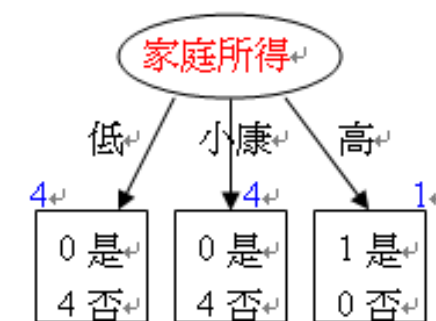


$$\text{Gain(家庭所得)} = 0.688$$

$$I(9,1) = -((1/9) \cdot \log_2(1/9) + (8/9) \cdot \log_2(8/9)) = 0.5032$$



$$\text{Gain(年齡)} = 0.2222$$



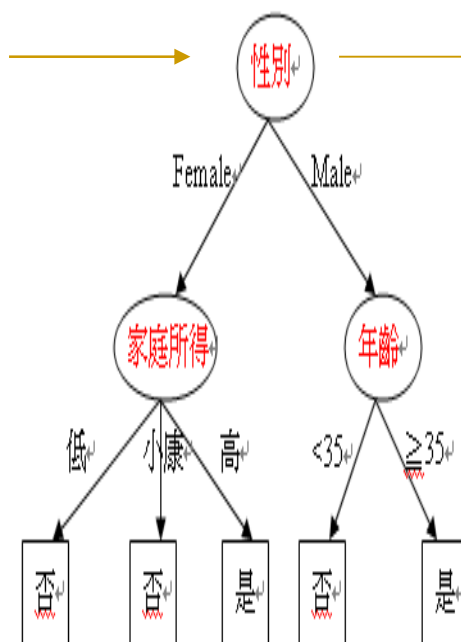
$$\text{Gain(家庭所得)} = 0.5032$$

# Example(end)ID3算法

## ■ 资料

| 年龄  | 性别     | 家庭所得 | 购买RV房车 |
|-----|--------|------|--------|
| <35 | Male   | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Female | 低所得  | 否      |
| <35 | Male   | 高所得  | 否      |
| ≥35 | Female | 低所得  | 否      |
| <35 | Female | 低所得  | 否      |
| <35 | Female | 高所得  | 是      |
| ≥35 | Male   | 小康   | 是      |
| <35 | Male   | 高所得  | 否      |
| ≥35 | Female | 小康   | 否      |
| <35 | Male   | 低所得  | 否      |
| ≥35 | Female | 小康   | 否      |
| ≥35 | Male   | 低所得  | 是      |
| ≥35 | Male   | 小康   | 是      |
| ≥35 | Female | 低所得  | 否      |

## ■ Decision Tree



## 分类规则:

IF 性别=Female AND 家庭所得=低所得 THEN 购买RV房车=否

IF 性别=Female AND 家庭所得=小康 THEN 购买RV房车=否

IF 性别=Female AND 家庭所得=高所得 THEN 购买RV房车=是

IF 性别=Male AND 年龄<35 THEN 购买RV房车=否

IF 性别=Male AND 年龄≥35 THEN 购买RV房车=是

## (四) Decision Tree的建立过程

### 1、决策树的停止

- 决策树是通过递归分割 (recursive partitioning) 建立而成，递归分割是一种把数据分割成不同小的部分的迭代过程。
- 如果有以下情况发生，决策树将停止分割：
  - 该群数据的每一笔数据都已经归类到同一类别。
  - 该群数据已经没有办法再找到新的属性来进行节点分割。
  - 该群数据已经没有任何尚未处理的数据。

## 2、决策树的剪枝(pruning)

- 决策树学习可能遭遇模型过度拟合 (over fitting) 的问题，过度拟合是指模型过度训练，导致模型记住的不是训练集的一般性，反而是训练集的局部特性。
- 如何处理过度拟合呢？对决策树进行修剪。
- 树的修剪有几种解决的方法，主要为先剪枝和后剪枝方法。

# (1) 先剪枝方法

- 在先剪枝方法中，通过提前停止树的构造（例如，通过决定在给定的节点上不再分裂或划分训练样本的子集）而对树“剪枝”。一旦停止，节点成为树叶。
- 确定阈值法：在构造树时，可将信息增益用于评估岔的优良性。如果在一个节点划分样本将导致低于预定义阈值的分裂，则给定子集的进一步划分将停止。
- 测试组修剪法：在使用训练组样本产生新的分岔时，就立刻使用测试组样本去测试这个分岔规则是否能够再现，如果不能，就被视作过度拟合而被修剪掉，如果能够再现，则该分岔予以保留而继续向下分岔。

## (2) 后剪枝方法

- 后剪枝方法是由“完全生长”的树剪去分枝。通过删除节点的分枝，剪掉叶节点。
- 案例数修剪是在产生完全生长的树后，根据最小案例数阈值，将案例数小于阈值的树节点剪掉。
- 成本复杂性修剪法是当决策树成长完成后，演算法计算所有叶节点的总和错误率，然后计算去除某一叶节点后的总和错误率，当去除该叶节点的错误率降低或者不变时，则剪掉该节点。反之，保留。

# 应用案例：在农业中的应用

表 1 农业总产值信息表

| 省和城市 | 乡村劳动力<br>/万人 | 耕地面积<br>/km <sup>2</sup> | 农业生产总值<br>/亿元 | 省和城市 | 乡村劳动力<br>/万人 | 耕地面积<br>/km <sup>2</sup> | 农业生产总值<br>/亿元 |
|------|--------------|--------------------------|---------------|------|--------------|--------------------------|---------------|
| 北京   | 67.7         | 399.5                    | 176.58        | 河南   | 2940.3       | 6805.8                   | 1822.99       |
| 天津   | 79.4         | 426.1                    | 156.17        | 湖北   | 1232.9       | 3358.0                   | 1147.51       |
| 河北   | 1635.5       | 6517.3                   | 1505.94       | 湖南   | 2062.9       | 3249.7                   | 1232.75       |
| 山西   | 639.9        | 3645.1                   | 359.15        | 广东   | 1508.2       | 2317.3                   | 1614.64       |
| 内蒙古  | 512.4        | 5491.4                   | 534.39        | 广西   | 1604.1       | 2614.2                   | 865.91        |
| 辽宁   | 633.0        | 3389.7                   | 969.79        | 海南   | 170.2        | 429.2                    | 242.54        |
| 吉林   | 517.0        | 3953.2                   | 666.47        | 四川   | 2811.9       | 6189.6                   | 1394.14       |
| 黑龙江  | 760.3        | 8995.3                   | 736.34        | 贵州   | 1388.4       | 1840.0                   | 402.29        |
| 上海   | 76.3         | 290.0                    | 206.78        | 云南   | 1661.8       | 2870.6                   | 614.50        |
| 江苏   | 1531.5       | 4448.3                   | 1849.19       | 西藏   | 89.3         | 222.1                    | 42.34         |
| 浙江   | 1102.7       | 1617.8                   | 1003.71       | 陕西   | 1047.4       | 3393.4                   | 479.36        |
| 安徽   | 1992.9       | 4291.1                   | 1202.27       | 甘肃   | 683.8        | 3482.5                   | 335.79        |
| 福建   | 776.8        | 1204.0                   | 973.39        | 青海   | 138.2        | 589.9                    | 60.78         |
| 江西   | 1073.7       | 2308.4                   | 734.87        | 宁夏   | 146.6        | 807.2                    | 78.76         |
| 山东   | 2487.0       | 6696.0                   | 2174.54       | 新疆   | 310.7        | 3128.3                   | 498.41        |

# 第一步：属性离散化

| 省和城市 | 乡村劳动力<br>分类 | 耕地面积<br>分类 | 农业生产<br>总值分类 | 省和城市 | 乡村劳动力<br>分类 | 耕地面积<br>分类 | 农业生产<br>总值分类 |
|------|-------------|------------|--------------|------|-------------|------------|--------------|
| 北京   | 1           | 1          | 1            | 河南   | 3           | 3          | 3            |
| 天津   | 1           | 1          | 1            | 湖北   | 2           | 2          | 2            |
| 河北   | 2           | 3          | 2            | 湖南   | 2           | 2          | 2            |
| 山西   | 1           | 2          | 1            | 广东   | 2           | 2          | 2            |
| 内蒙古  | 1           | 3          | 1            | 广西   | 2           | 2          | 2            |
| 辽宁   | 1           | 2          | 2            | 海南   | 1           | 1          | 1            |
| 吉林   | 1           | 2          | 1            | 四川   | 3           | 3          | 2            |
| 黑龙江  | 1           | 3          | 1            | 贵州   | 2           | 2          | 1            |
| 上海   | 1           | 1          | 1            | 云南   | 2           | 2          | 1            |
| 江苏   | 2           | 2          | 3            | 西藏   | 1           | 1          | 1            |
| 浙江   | 2           | 2          | 2            | 陕西   | 2           | 2          | 1            |
| 安徽   | 2           | 2          | 2            | 甘肃   | 1           | 2          | 1            |
| 福建   | 1           | 2          | 2            | 青海   | 1           | 1          | 1            |
| 江西   | 2           | 2          | 1            | 宁夏   | 1           | 1          | 1            |
| 山东   | 3           | 3          | 3            | 新疆   | 1           | 2          | 1            |



# 第二步：概化（泛化）

| 区域 | 乡村劳动力 | 耕地面积 | 农业总产值 | 区域 | 乡村劳动力 | 耕地面积 | 农业总产值 |
|----|-------|------|-------|----|-------|------|-------|
| 华北 | 少     | 小    | 低     | 中南 | 多     | 大    | 高     |
| 华北 | 少     | 小小   | 低     | 中南 | 中     | 大中   | 中     |
| 华北 | 中     | 大    | 中     | 中南 | 中     | 中    | 中     |
| 华北 | 少     | 中    | 低     | 中南 | 中     | 中    | 中     |
| 华北 | 少     | 大    | 低     | 中南 | 中     | 中    | 中     |
| 东北 | 少     | 中    | 中     | 中南 | 少     | 小    | 低     |
| 东北 | 少     | 中    | 低     | 西南 | 多     | 大    | 中     |
| 东北 | 少     | 大    | 低     | 西南 | 中     | 中    | 低     |
| 华东 | 少     | 小    | 低     | 西南 | 中     | 中    | 低     |
| 华东 | 中     | 中    | 高     | 西南 | 少     | 小    | 低     |
| 华东 | 中     | 中    | 中     | 西北 | 中     | 中    | 低     |
| 华东 | 中     | 中    | 中     | 西北 | 少     | 中    | 低     |
| 华东 | 少     | 中    | 中     | 西北 | 少     | 小    | 低     |
| 华东 | 中     | 中    | 低     | 西北 | 少     | 小小   | 低     |
| 华东 | 多     | 大    | 高     | 西北 | 少     | 中    | 低     |

### 第三步：计算各属性的期望信息

$$=(17/30)*\text{LOG}((17/30),2)+(10/30)*\text{LOG}((10/30),2)+(3/30)*\text{LOG}((3/30),2)$$

$$Info(D) = - \sum_{i=1}^n p_i \log_2(p_i) \quad (1)$$

根据式 (1), 得  $Info(D) = 1.3249$

2) 计算每个属性的期望信息需求

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times Info(D_j) \quad (2)$$

根据式 (2), 得  $Info_{\text{区域}}(D) = 0.9339$

同理可得  $Info_{\text{乡村劳动力}}(D) = 1.0301$

$$Info_{\text{耕地面积}}(D) = 1.0332$$

# 计算各属性的信息增益

计算该划分的各信息增益量

$$Gain(A) = Info(D) - Info_A(D) \quad (3)$$

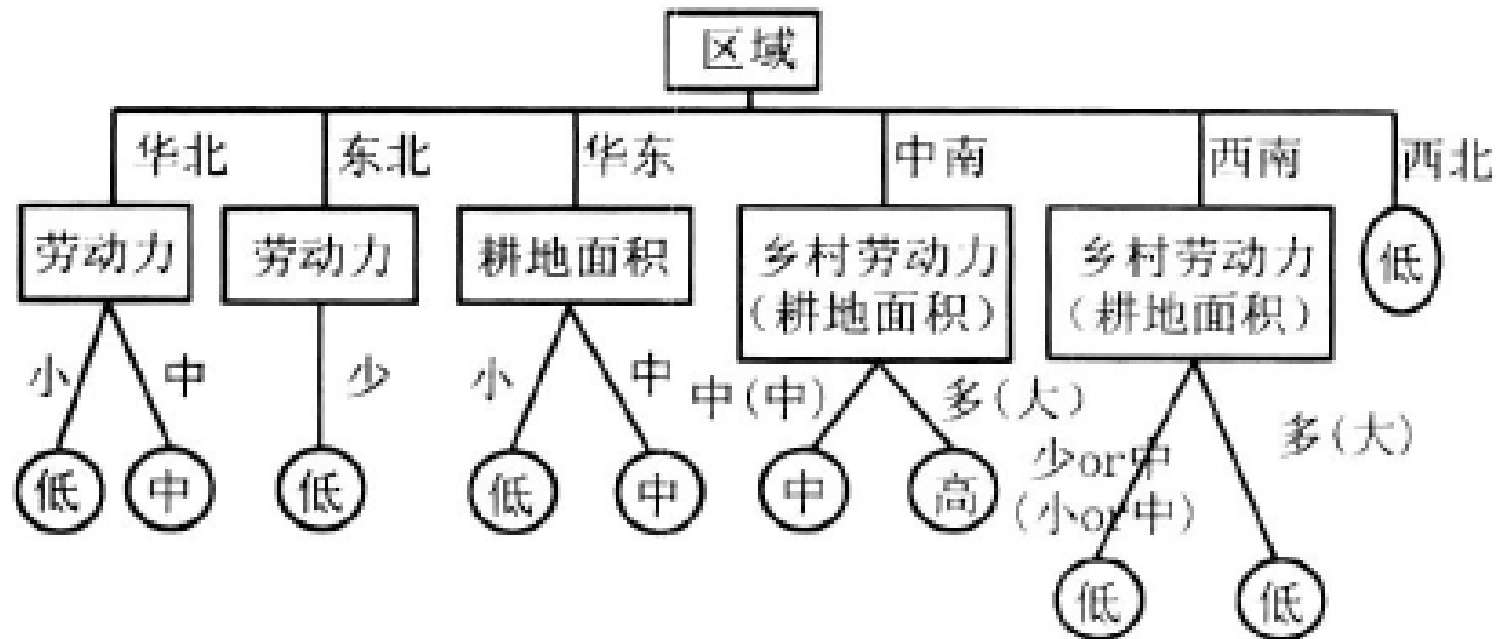
根据式(3), 得

$$Gain(\text{区域}) = Info(D) - Info_{\text{区域}}(D) = \\ 1.3249 - 0.9339 = 0.3910$$

$$Gain(\text{乡村劳动力}) = Info(D) - \\ Info_{\text{乡村劳动力}}(D) = 1.3249 - \\ 1.0301 = 0.2948$$

$$Gain(\text{耕地面积}) = Info(D) - Info_{\text{耕地面积}}(D) = \\ 1.3249 - 1.0332 = 0.2917$$

## 第四步：决策树



# 案例2：银行违约率

表 1 训练数据集

| 标识 | 利润情况 | 企业类别 | 企业规模 | 经营行业 | 类：违约记录 |
|----|------|------|------|------|--------|
| 1  | 亏    | 国有   | 大    | 制造业  | H      |
| 2  | 亏    | 国有   | 大    | 商业   | H      |
| 3  | 亏    | 外资   | 大    | 制造业  | L      |
| 4  | 亏    | 个体   | 中    | 制造业  | L      |
| 5  | 盈    | 个体   | 小    | 制造业  | L      |
| 6  | 盈    | 个体   | 小    | 商业   | H      |
| 7  | 盈    | 外资   | 小    | 商业   | L      |
| 8  | 亏    | 国有   | 中    | 制造业  | H      |
| 9  | 盈    | 国有   | 小    | 制造业  | L      |
| 10 | 盈    | 个体   | 中    | 制造业  | L      |
| 11 | 盈    | 国有   | 中    | 商业   | L      |
| 12 | 亏    | 外资   | 中    | 商业   | L      |
| 13 | 盈    | 外资   | 大    | 制造业  | L      |
| 14 | 亏    | 个体   | 中    | 商业   | H      |



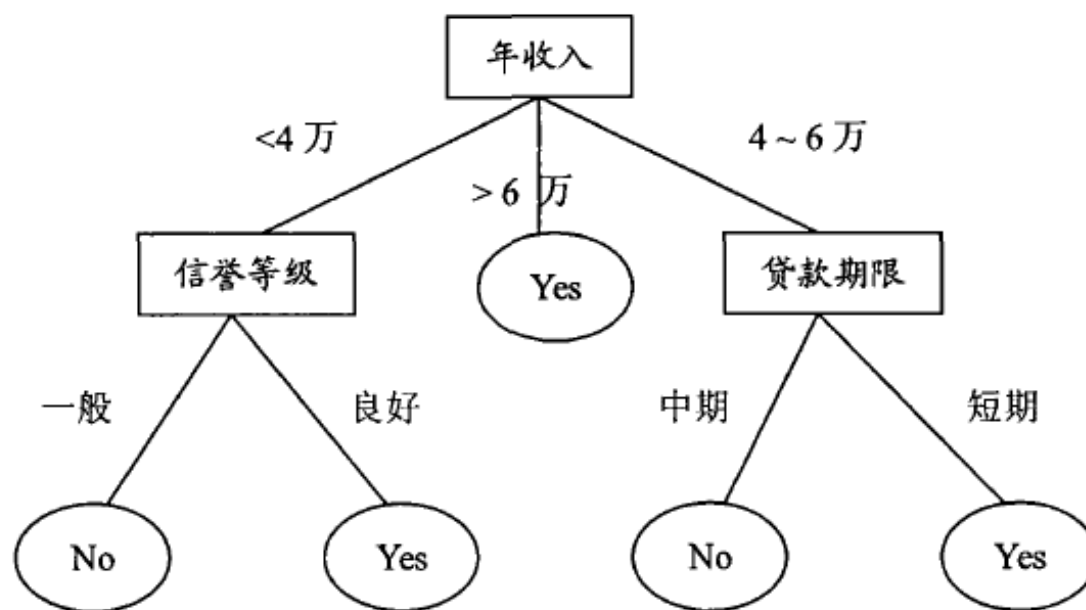
图 1 违约记录判定树

## 案例3 对电信客户的流失率分析

- 1、影响客户流失的最为关键的因素是客户在网时间，在网时间短的客户，其流失比例较大，在网时间越长的客户越稳定，越不容易流失；
- 2、如果在网时长小于 12 个月，通话时长变化率小于 -0.5 的用户，那么其流失概率是 100%；
- 3、如果如果在网时长大于等于 12 个月，通话时长变化率大于等于 -0.5 的用户，那么其流失概率是 2.8%；
- 4、如果在网时长大于等于 12 个月，通话时长变化率小于 -0.5，并且年龄小于等于 30 岁的用户，那么其流失概率为 0%；
- 5、如果在网时长大于等于 12 个月，通话时长变化率小于 -0.5，并且年龄大于 30 岁，主叫通话变化率在 -0.5 和 0.5 之间的用户，那么其流失概率为 10%；
- 6、如果在网时长大于等于 12 个月，通话时长变化率小于 -0.5，并且年龄大于 30 岁，主叫通话变化率小于 -0.5，长途占比等于 0 的用户，那么其流失概率为 20%；
- 7、如果在网时长大于等于 12 个月，通话时长变化率小于 -0.5，并且年龄大于 30 岁，主叫通话变化率小于 -0.5，长途占比大于 0 的用户，那么其流失概率为 0，即该类用户 100%不会流失。

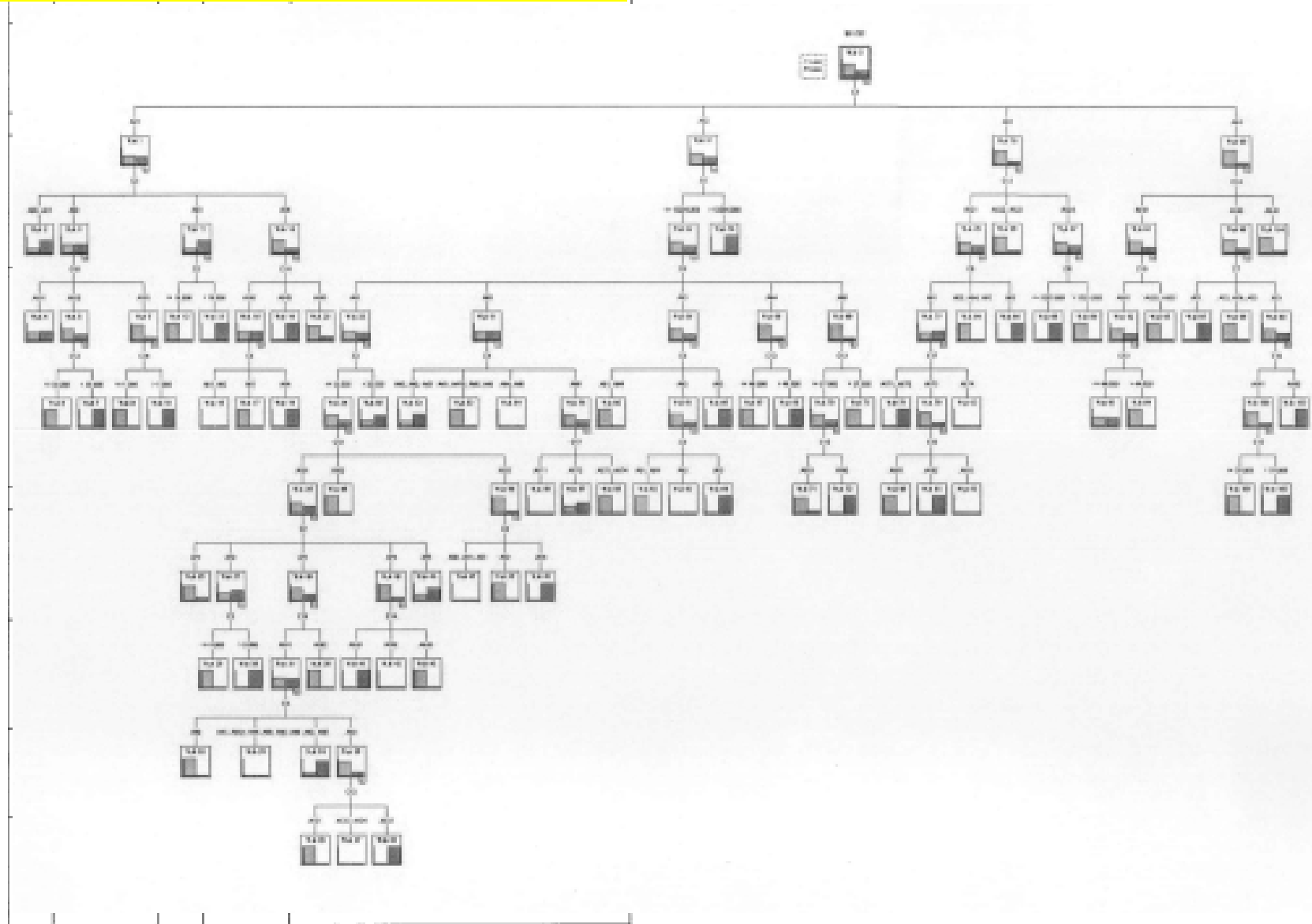
## 案例4：在银行中的应用

下图是一棵典型的决策树，它用于根据申请贷款的客户信息来决策是否为该用户发放贷款。若一个客户年收入5万，信誉等级一般，申请贷款期限为短期，那么根据该决策树的判断就应该为该客户发放贷款。





# 个人信用评级决策树

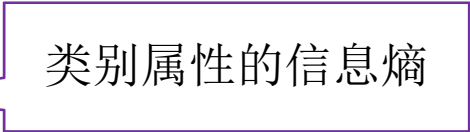


## (五) 其他算法

- C4.5与C5.0算法
- Gini Index算法
- CART算法
- PRISM算法
- CHAID算法

# 1、C4.5与C5.0算法

**ID3** 算法存在如下缺点：在信息增益的计算中，属性  $A$  将  $U$  划分为子集  $\{U_1, U_2, \dots, U_v\}$ ，当每一集合中所有记录得出的结果相同。那么根据  $A$  划分的期望信息  $E(A)$  就为 0，此时增益  $Gain(A)$  就为最大值。为了避免这种情况，算法 **C4.5** 使用了信息增益比例作为属性选择度量

$$Gainratio = \frac{Gain(A)}{I(u_1, u_2, \dots, u_m)}$$


类别属性的信息熵

- C5.0算法则是C4.5算法的修订版，适用在处理大数据集，采用Boosting（提升）方式提高模型准确率，又称为Boosting Trees，在软件上的计算速度比较快，占用的内存资源较少。

## 2、Gini Index算法

- ID3 and PRISM适用于类别属性的分类方法。
- Gini Index能数值型属性的变量来做分类。着重解决当训练集数据量巨大，无法全部放入内存时，如何高速准确地生成更快的，更小的决策树。

# Gini Index算法

- 集合T包含N个类别的记录，那么其Gini指标就是

$$gini(T) = 1 - \sum_{j=1}^N p_j^2 \quad p_j \text{ 为 } j \text{ 类别出现的频率}$$

- 如果集合T分成两部分N1 和 N2。则此分割的Gini就是

$$gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$

先对数值型字段排序，假设排序后的结果为  $v_1, v_2, \dots, v_m$ ，因为分裂只会发生在两个节点之间，所以有  $n-1$  种可能性。通常取中点  $(v_i + v_{i+1})/2$  作为分裂点，从小到大依次取不同的 split point，取 Information Gain 指标最大 (gini 最小) 的一个就是分裂点。

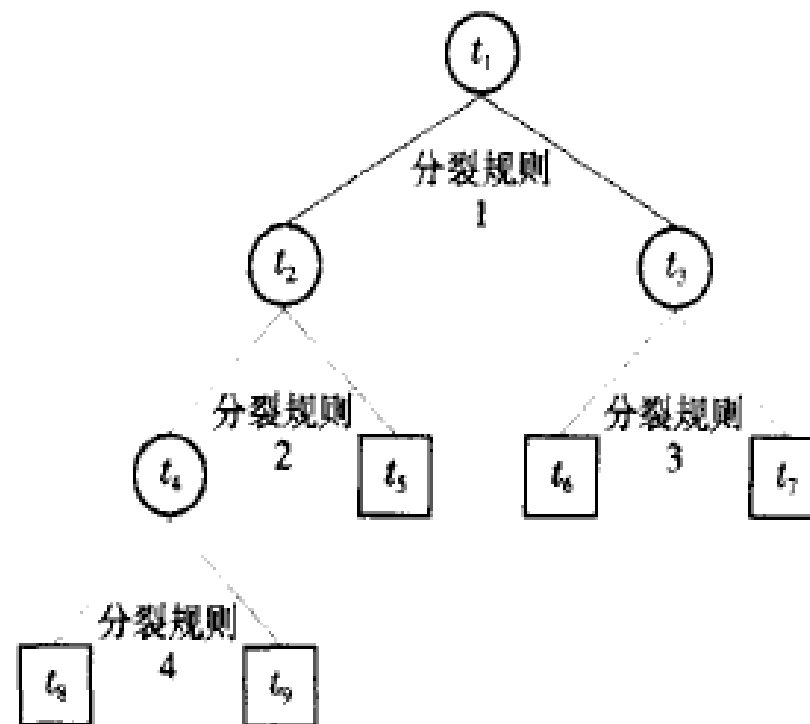
### 3、CART算法

- 由Friedman等人提出，1980年以来就开始发展，是基于树结构产生分类和回归模型的过程，是一种产生二元树的技术。
- CART与C4.5/C5.0算法的最大的区别是：其在每一个节点上都是采用二分法，也就是一次只能够有两个子节点，C4.5/5.0则在每一个节点上可以产生不同数量的分枝。

设训练样本集  $L = \{X_1, X_2, \dots, X_n, Y\}$ , 其中,  $X_i$  ( $i = 1, 2, \dots, n$ ) 称为属性向量;  $Y$  称为标签向量或类别向量. 当  $Y$  是有序的数量值时, 称为回归树; 当  $Y$  是离散值时, 称为分类树.

## 构建树的步骤:

(1) 在根节点  $t_1$  处, 搜索问题集 (特征空间), 找到使得下一代子节点中数据集的非纯度下降最大的最优分裂变量和相应的分裂阈值. 在这里, 非纯度指标用 Gini 指数来衡量



$$i(t) = \sum_{i \neq j} p(i|t)p(j|t) = 1 - \sum_j [p(j|t)]^2$$

其中,  $i(t)$  是节点  $t$  的 Gini 指数,  $p(i|t)$  表示在节点  $t$  中属于  $i$  类的样本所占的比例,  $p(j|t)$  是节点  $t$  中属于  $j$  类的样本所占的比例。假定节点  $t$  的下一代子节点分为  $t_L$  和  $t_R$ , 则非纯度指标的下降量表示为:

$$\Delta i(t) = i(t) - i(t_L)p(i|L) - i(t_R)[1 - p(i|L)]$$

其中:  $\Delta i(t)$  是非纯度的下降量;  $t_L$  和  $t_R$  分别是节点  $t$  的左右子结点,  $i(t_L)$  和  $i(t_R)$  分别是左右子节点的不纯度指数。  $p(i|L)$  为节点  $t$  分到左子节点的样本所占的比例。

(2) 用该分裂变量和分裂阈值把根节点  $t_1$  分裂成  $t_2$  和  $t_3$ 。

(3) 如果在某个节点  $t_i$  处, 不可能再有进一步非纯度的显著降低, 则节点  $t_i$  成为叶结点。否则像步骤(1)那样, 寻找它的最优分裂变量和分裂阈值继续进行分裂。

(4) 当叶节点中只有一个类, 那么这个类就作为叶节点所属的类, 若节点中有多个类中的样本存在, 根据叶节点中样本最多的那个类来确定节点所属的类别。