DM5 分类（基本概念）

5.1 基本概念

5.2 决策树归纳

5.3 贝叶斯分类

5.4 基于规则的分类

5.5 模型评估与选择

5.6 提高分类准确率的技术

分类 VS. 预测

❖ 分类与预测

- 两种数据分析形式，用于提取描述重要数据类或预测未来数据趋势的模型

❖ 典型应用

- 信誉证实（分类为低，中，高风险）
- 医疗诊断（肿瘤是良性还是恶性）
- 性能预测
- 市场定位

分类 VS. 预测

❖ 分类

- ➢ 根据训练数据集和类标号构建模型
- ➢ 用于预测新数据的类标号（离散值）

❖ 分类示例

- ➢ 银行贷款员需要分析数据，来弄清哪些贷款申请者是安全的，哪些是有风险的。就需要构造一个分类器来预测类属编号，比如预测顾客属类。
- ➢ 垃圾邮件分类。

❖ 分类方法

- ➢ 决策树、贝叶斯方法、K-近邻方法、支持向量机、神经网络、关联规则方法等

分类 VS. 预测

- **预测**
  - 构建连续函数值模型
  - 用于预测未知值或缺省值（连续值）
- **预测示例**
  - 银行贷款员需要预测贷给某个顾客多少钱是安全的。就需要构造一个预测器，预测一个连续值函数或有序值，比如贷款金额。
  - 红酒品质鉴别。
- **预测方法**
  - 线性回归、 多元回归、 非线性回归

分类 VS. 预测

❖ 相同点

- ➢ 两者都需要构建模型，再用模型来估计未知值
  - 模型看作一个映射或函数 y=f(X)， X是输入，y是输出；
  - 模型准确率，要使用单独的测试集进行测试；

❖ 不同点

- ➢ 分类法用来估计“类标号属性” （分类属性值、离散值）
- ➢ 预测法用来估计 “预测属性” （量化属性值、连续值）

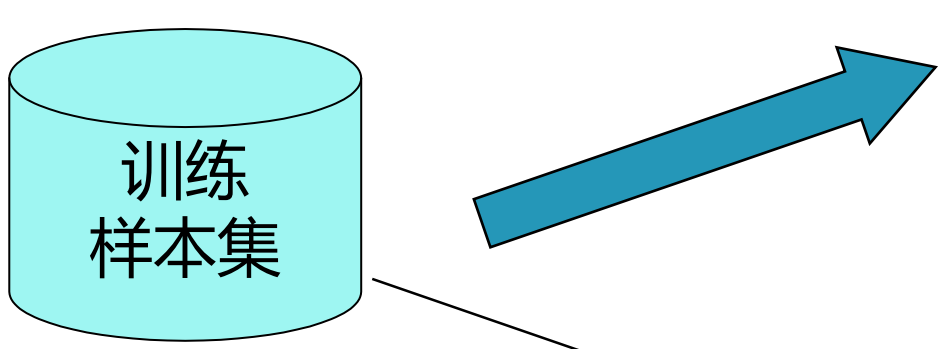
分类：一个两步过程

❖ **第一步：建立模型（分类器），描述预先定义的数据类或概念集**

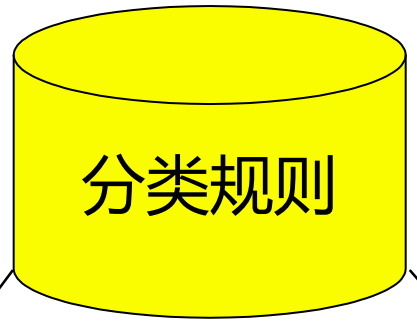
- **基本概念**
  - **训练集：为建立模型而被分析的样本（用属性向量表示）及其对应的类标号组成，这里假定每个样本属于一个预定义的类**
  - **训练样本：训练集中的单个样本**
- **分类算法通过分析或从训练集“学习”来构造分类器**
- **学习模型可以用分类规则、决策树或数学公式的形式提供**

分类：一个两步过程

❖ **第一步：建立模型**



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no



IF rank = ‘professor’
OR years > 6
THEN tenured = ‘yes’

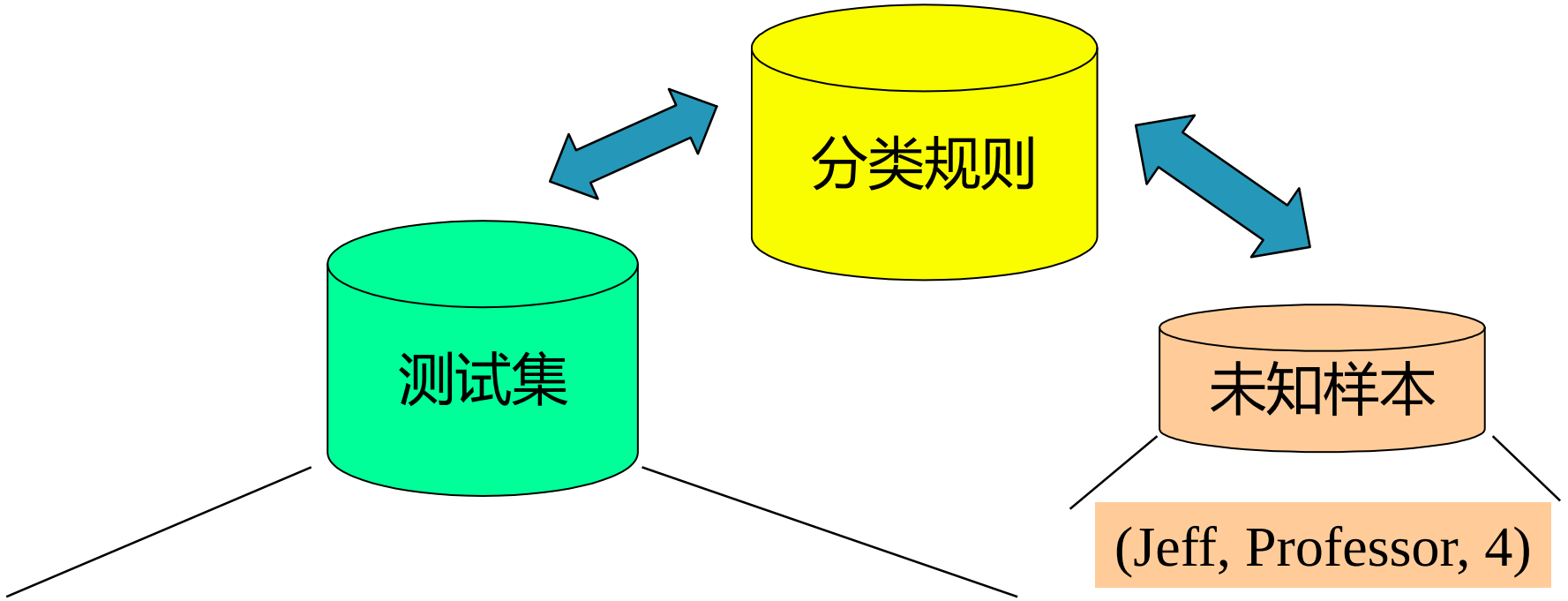
分类：一个两步过程

❖ **第二步：使用模型（分类器），对将来的或未知的对象进行分类**

- **基本概念**
  - 测试集：要独立于训练集，避免“过分拟合”的情况
  - 测试样本：对每个测试样本，将已知的类标号和该样本的学习模型预测的类比较
  - 准确率：被模型正确分类的测试样本的百分比
- **分类器采用准确率来评估**
- **如果准确率可以接受，那么使用该模型对将来的或未知的样本进行分类**

分类：一个两步过程

❖ **第二步：使用模型**



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

监督学习 VS. 无监督学习

❖ 监督学习（用于分类、预测）

- ➢ 模型的学习在被告知每个训练样本属于哪个类的“指导”下进行
- ➢ 新数据使用训练集中得到的规则进行分类

❖ 无监督学习（用于聚类）

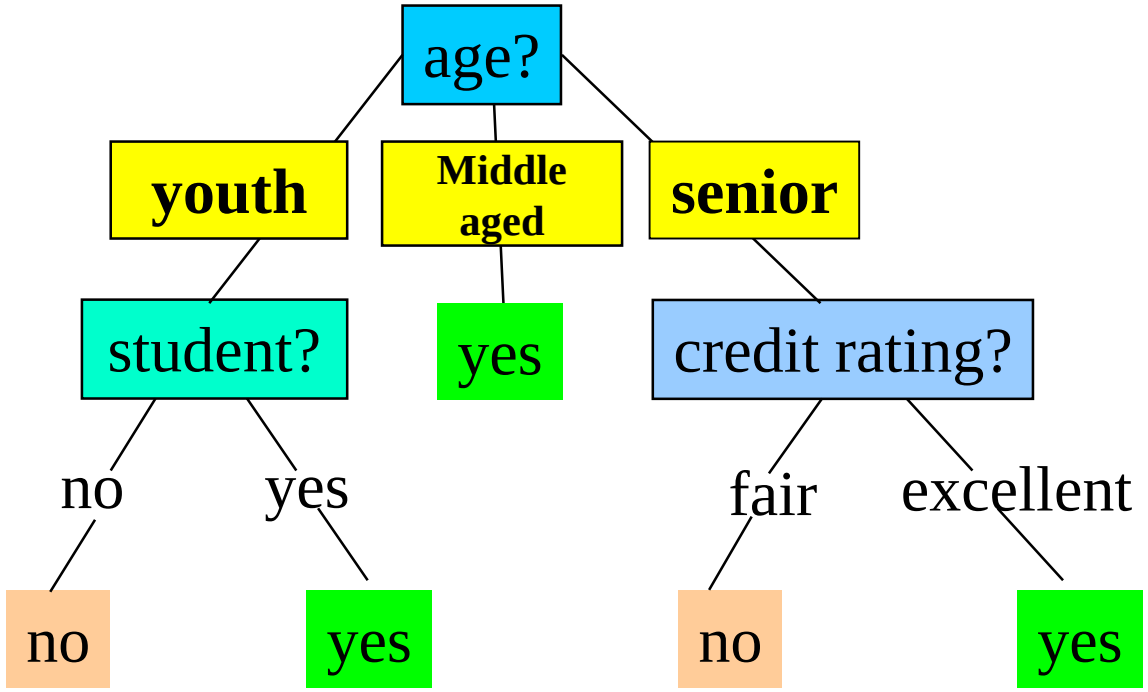
- ➢ 每个训练样本的类标号是未知的，要学习的类集合或数量也可能是事先未知的
- ➢ 通过一系列的度量、观察来建立数据中的类标号或进行聚类

❖ 什么是决策树?

- 类似于流程图的树结构
- 每个内部节点(非树叶结点)表示在一个属性上的测试
- 每个分枝代表一个测试输出
- 每个树叶节点存放一个类标号

决策树:

Buys_computer



决策树归纳

❖ 决策树的生成由两个阶段组成

- ➢ 树构建：自顶向下递归地分治方式
  - 使用属性选择度量选择将样本最好的划分为不同的类的属性
  - 递归地通过选择属性划分样本 （属性都是离散值，如果是连续的，将其离散化）
- ➢ 树剪枝
  - 识别和删除那些反映噪声或离群点的分枝

❖ 使用决策树分类

- ➢ 给定一个类标号未知的样本，在决策树上测试样本的属性值，跟踪一条由根到叶节点的路径，叶节点存放该样本的类预测
- ➢ 决策树容易转换为分类规则

决策树归纳

❖ 输入

- 数据分区D，它是训练样本和他们对应类标号的集合
- attribute_list，描述样本属性的列表
- attribute_selection_method，指定选择属性的启发性过程，用来选择可以按类“最好地”区分给定样本的属性

决策树归纳

❖ 算法步骤

- 树以代表训练样本的单个节点（N）开始；
- 如果样本都在同一个类，则该节点成为树叶，并用该类标记；
- 否则，算法调用attribute_selection_method，选择能够最好地将样本分类的属性；确定**分裂准则**，指出**分裂点**或**分裂子集**；
- 对测试属性每个已知的值，创建一个分支，并以此划分样本；
- 算法使用同样的过程，递归地形成每个划分上的决策树；一旦一个属性出现在一个节点上，就不在该节点的任何子节点上出现；
- 递归划分步骤停止的条件：
  - 划分D（在N节点提供）的所有样本属于同一类
  - 没有剩余属性可以用来进一步划分样本
  - 没有剩余的样本
  - 给定分支没有样本，则以D中多数类创建一个树叶

属性选择度量

❖ 属性选择度量

- 属性选择度量是一种选择**分裂准则**，将给定类标号的训练样本**最好的**进行划分的启发式方法，理想情况下，每个分区都是“纯”的，即落在一个给定分区间的所有样本都属于相同的类
- 为描述给定训练样本的每个属性提供秩评定，具有最好度量**得分**的属性被选为给定样本的分裂属性

❖ 常用的属性选择度量

- 信息增益（ID3）
- 增益率（C4.5）
- 基尼指标（CART）

信息增益

❖ 相关理论基础

- 若一个系统中存在多个事件$X_1,X_2,...X_n$，每个事件出现的概率是$p(X_i)$，且$p(X_1)+p(X_2)+...+p(X_n)=1$，$p(X_i)$小说明$X_i$发生可能性小，说明其不确定性大。
- 对于任意一个随机事件，它的熵定义如下：变量的不确定性越大，熵也就越大，把它搞清楚所需要的信息量也就越大。信息熵是信息论中用于度量信息量的一个概念，一个系统越有序（混乱），信息熵就越低（高）。
- 信息论中定义事件X_i的信息量为：$\log_2 \frac{1}{P(X_i)} = -\log_2 P(X_i)$
- 信息论中定义事件的平均信息量为单个事件的信息量的统计平均值，称为**期望信息（信息熵）**为：

$$-\sum_{i=1}^{m} P(X_i)\log_2 P(X_i)$$

信息增益

❖ 信息增益

- 假设P_i是D中任意样本属于类 i (i=1,2...m)的非零概率，并用$|D_i|/|D|$ 估计。对D中样本分类所需要的期望信息（信息熵）由下式给出：

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

- 用属性A将D划分为v个分区或子集后，为了得到准确的分类，我们还需要多少信息？这个量由下式度量：

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

- 信息增益为：

$$Gain(A) = Info(D) - Info_A(D)$$

- 选择具有**最高信息增益**的属性作为结点N的分裂属性！
- 注意：信息增益度量倾向于选择具有大量值的属性，例如，考虑充当唯一标识符的属性，这种划分信息增益最大，但是对分类没用。

信息增益 - 例5.1

例5.1:

age	income	student	credit_rating	buys_computer
youth	high	no	fair	no
youth	high	no	excellent	no
middle_aged	high	no	fair	yes
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
middle_aged	low	yes	excellent	yes
youth	medium	no	fair	no
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
middle_aged	medium	no	excellent	yes
middle_aged	high	yes	fair	yes
senior	medium	no	excellent	no

信息增益 - 例5.1

解：

Class P: buys_computer = “yes”

Class N: buys_computer = “no”

- **计算对D中样本分类所需要的期望信息：**

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

- **若样本根据age划分，则：**

$$Info_{age}(D) = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.694$$

- **这种划分的信息增益：**

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

- **同理：**

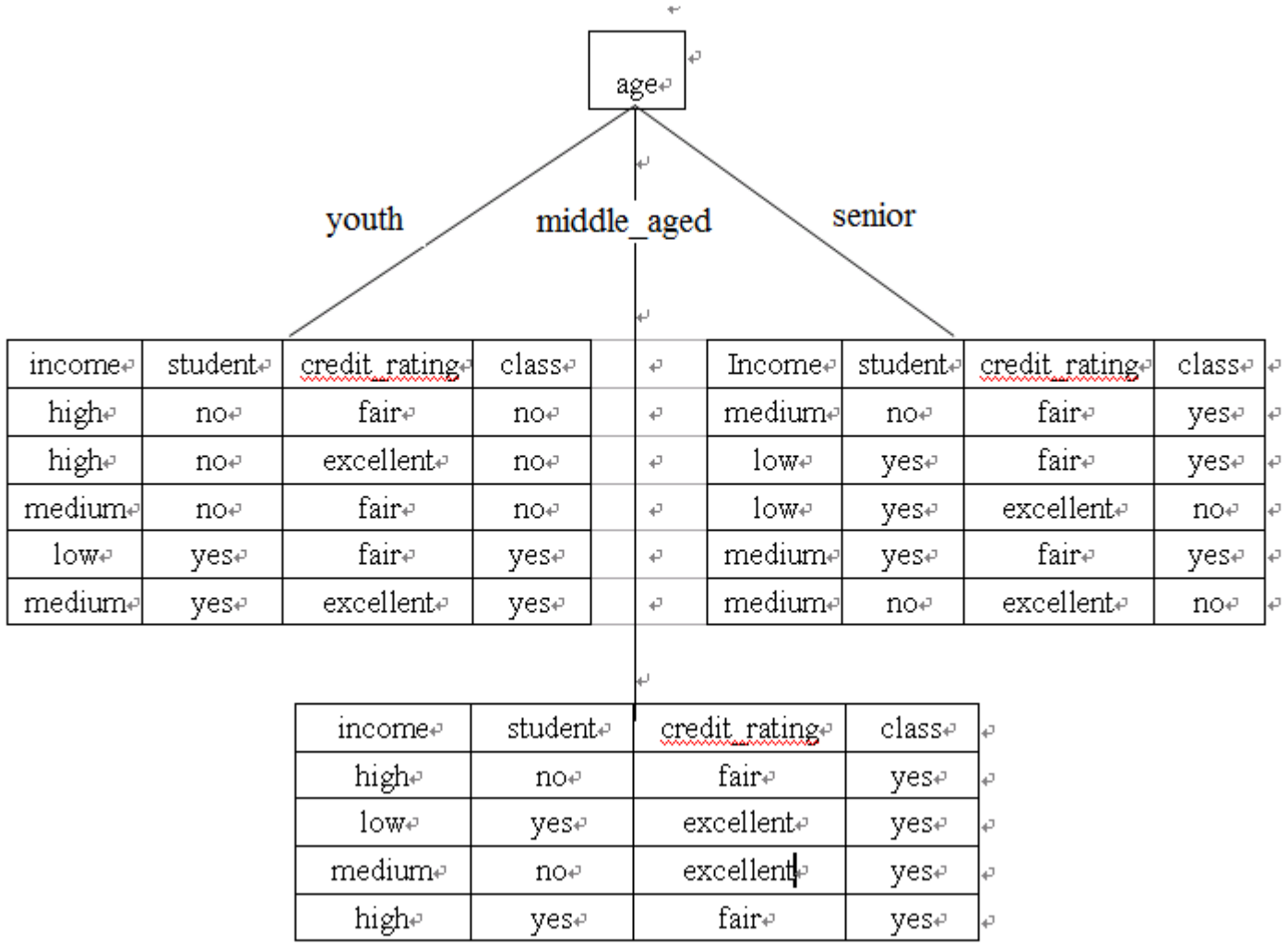
$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

信息增益 - 例5.1

- 由于age在属性中具有最高的信息增益，所以它被选作分裂属性：



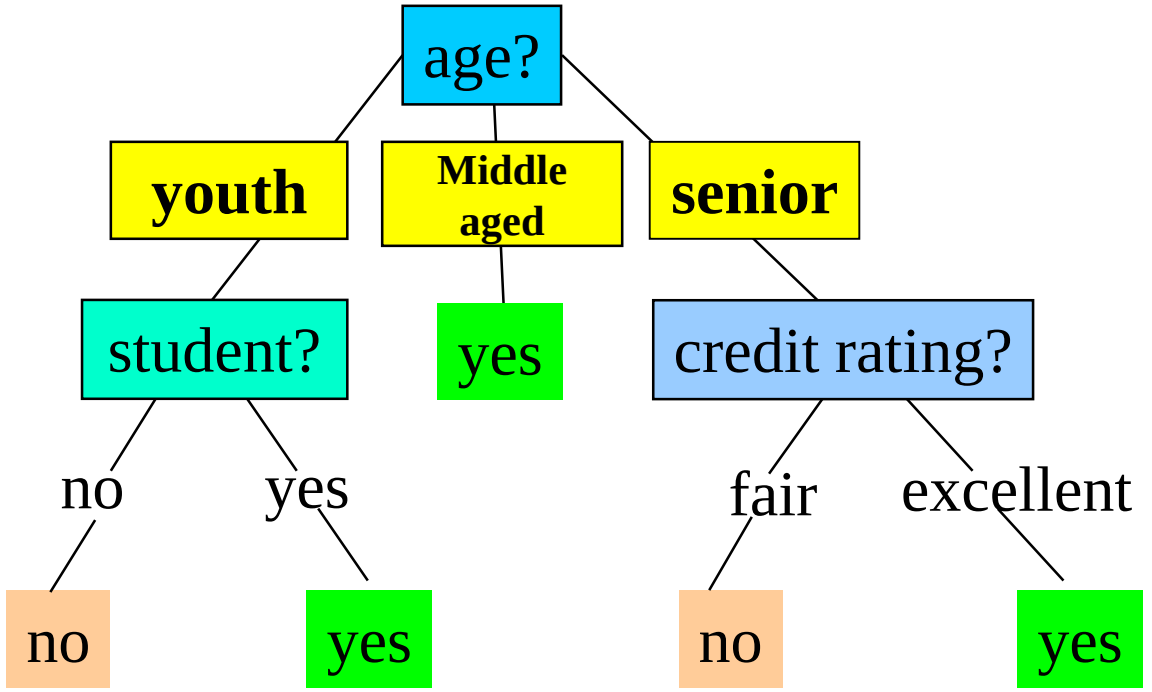
income	student	credit_rating	class
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

Income	student	credit_rating	class
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

income	student	credit_rating	class
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

信息增益 - 例5.1

- **最终的决策树：**



计算连续值属性的信息增益

- 假设A是连续值，而不是离散值
  - 必须确定A的“最佳”分裂点
    - 将A的值按递增序排序
    - 典型的，每对相邻值的中点被看作可能的分裂点
      - ✓A的值 ai 和 ai+1 之间的中点是 (ai+ai+1)/2
      - ✓A具有最小期望信息需求的点选做A的分裂点（split-point ）
  - 分裂
    - D1 是满足 A ≤ split-point 的样本集合, 而 D2 是满足A > split-point 的样本集合

增益率

❖ 增益率

- ID3 的后继 C4.5 使用一种称为增益率的信息增益扩充，试图克服信息增益度量倾向于选择具有大量值的属性这种偏倚，它用“分裂信息”值将信息增益规范化。
- 分裂信息定义如下：$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$
- 增益率为：$$GainRate(A) = \frac{Gain(A)}{SplitInfo_A(D)}$$
- 选择具有**最大增益率**的属性作为分裂属性！
- 注意：随着划分信息趋向于0，增益率变得不稳定，为了避免这种情况，增加一个约束：选取的测试的信息增益必须较大，至少与考察的所有测试的平均增益一样大。

增益率 - 例5.1

解：属性income的测试将表中的数据划分为3个分区，即low、medium和high，分别包含4、6和4个元组。

- **计算income的分裂信息：**

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

$$= -\frac{4}{14} \times \log_2 \frac{4}{14} - \frac{6}{14} \times \log_2 \frac{6}{14} - \frac{4}{14} \times \log_2 \frac{4}{14} = 1.557$$

- **计算income的信息增益：** $Gain(income) = 0.029$

- **计算income的增益率：** $GainRatio(income) = 0.029/1.557 = 0.019$

基尼指数

❖ 基尼指数

- 假设P_i是D中任意样本属于类 i (i=1,2...m)的非零概率，并用$|D_i|/|D|$ 估计。基尼指数度量数据分区或训练样本集D的**不纯度**，定义为：

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

- 基尼指数考虑每个属性的二元划分。
- 如果A的二元划分将D划分成D1和D2，则给定该划分，D的基尼指数为：

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

- 不纯度降低为：$\Delta Gini(A) = Gini(D) - Gini_A(D)$

- 选择具有**最大化不纯度降低**的属性作为分裂属性！

基尼指数 - 例5.1

解：

Class P: buys_computer = “yes” （包含9个样本）

Class N: buys_computer = “no” （包含5个样本）

- **首先使用基尼指数式计算D的不纯度：**

$$Gini(D)=1-\left(\frac{9}{14}\right)^2-\left(\frac{5}{14}\right)^2=0.459$$

- **计算每个属性的基尼指数。**
  - **从属性income开始，并考虑每个可能的分裂子集。考虑子集{ low, medium }，导致10个满足条件的样本在分区D_1中，其它个样本将指派到分区D_2中。基于该划分计算出基尼指数值为：**

$$Gini_{income\in\{low,medium\}}(D)=\frac{10}{14}Gini(D_1)+\frac{4}{14}Gini(D_2)$$

$$=\frac{10}{14}\left(1-\left(\frac{7}{10}\right)^2-\left(\frac{3}{10}\right)^2\right)+\frac{4}{14}\left(1-\left(\frac{2}{4}\right)^2-\left(\frac{2}{4}\right)^2\right)=0.443=Gini_{income\in\{high\}}(D)$$

基尼指数 - 例5.1

解：

- 类似地，用其余子集划分的基尼指数值为：
- 0.458（子集{low, high}和 {medium} ）
- 0.450（子集{medium, high}和 {low} ）

➢ 因此，属性income的最好二元划分在{low, medium}和 {high}上，因为它最小化基尼指数，它的基尼指数为：0.443

➢ 评估属性age，得到 {youth, senior}和 {middle_aged}为最好划分，基尼指数为：0.357

➢ 属性student和credit_rating都是二元的，分别具有基尼指数0.367和0.429。

➢ 因此，属性 age和分裂子集 {youth, senior}产生最小的基尼指数，不纯度降低0.459-0.357=0.102。二元划分“age {youth, senior}”导致D中样本的不纯度降低最大，并返回作为分裂准则。结点N用该准则标记，从它生长出两个分析，并相应地划分样本。

属性选择度量比较

三种度量通常会得到好的结果，但这些度量并非无偏的

- **信息增益**
  - 偏向于多值属性
- **增益率**
  - 倾向于不平衡的划分，其中一个分区比其他分区小得多
- **基尼指数**
  - 偏向于多值属性
  - 当类的数量很大时会有困难
  - 倾向于导致相等大小的分区和纯度

过度拟合和树剪枝

- **产生的决策树会出现过分适应数据的问题**
  - 由于数据中的噪声和离群点，许多分枝反映的是训练数据的异常
  - 对未知样本判断不准确
- **防止过分拟合的两种方法**
  - 先剪枝
    - 通过提前停止树的构造，如果划分一个结点样本导致低于预定义阈值的划分，则给定子集的进一步划分将停止。
    - 选择一个合适的阈值往往很困难，高阈值可能导致分化过分简化的树，低阈值可能导致树的简化太少。
  - 后剪枝
    - 由“完全生长”的树剪去子集，删除结点的分枝用最频繁的类标记来替换。
    - 通常使用一个独立的测试集来评估每颗树的准确率，就能得到具有最小期望错误率的决策树，不同于C4.5使用一种称为悲观剪枝的方法。

可伸缩性和决策树归纳

❖ RainForest (雨林)

- ➢ 能适应可用的内存量，并用于任意决策树归纳算法
- ➢ 在每个结点，对每个属性维护一个AVC-集（其中AVC表示“属性-值，类标号”），描述该结点的训练样本
- ➢ 结点N上属性A的AVC-集给出N上样本的属性A的每个值的类标号计数
- ➢ 结点N上所有AVC-集的集合是N的AVC-组群

雨林 - 例子

❖ **训练集和它的AVC-集**

age	income	student	credit_rating	comp
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

AVC-set on *Age*

Age	Buy_Computer	
	yes	no
<=30	2	3
31..40	4	0
>40	3	2

AVC-set on *income*

income	Buy_Computer	
	yes	no
high	2	2
medium	4	2
low	3	1

AVC-set on *Student*

student	Buy_Computer	
	yes	no
yes	6	1
no	3	4

AVC-set on *credit_rating*

Credit rating	Buy_Computer	
	yes	no
fair	6	2
excellent	3	3

决策树 - 例5.2（作业）

例5.2：构建决策树

年龄	性别	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

决策树 - 例5.2（作业）

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

解：

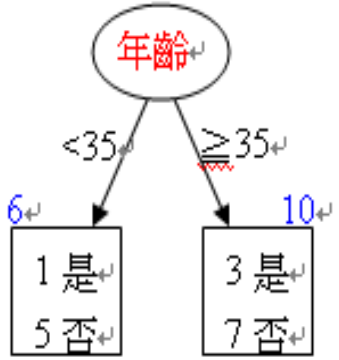
n=16 n_1=4

$Info(D)= -((4/16)*\log_2(4/16)+(12/16)*\log_2(12/16))$

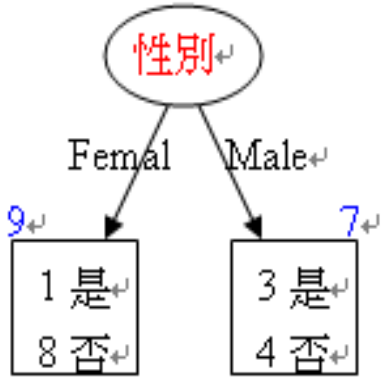
$=0.8113$

Info(年齡)=(6/16)*I(6,1)+(10/16)*I(10,3)=0.7946

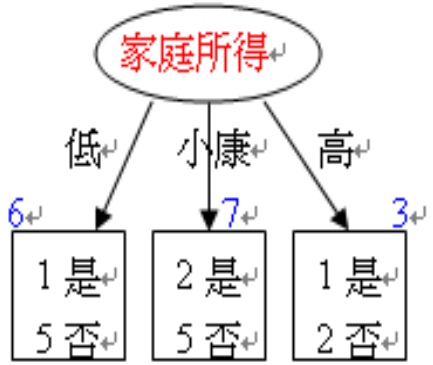
Gain(年齡)= *Info*(*D*)-*Info*(年齡)=0.0167



Gain(年齡)=0.0167



Gain(性別)=0.0972

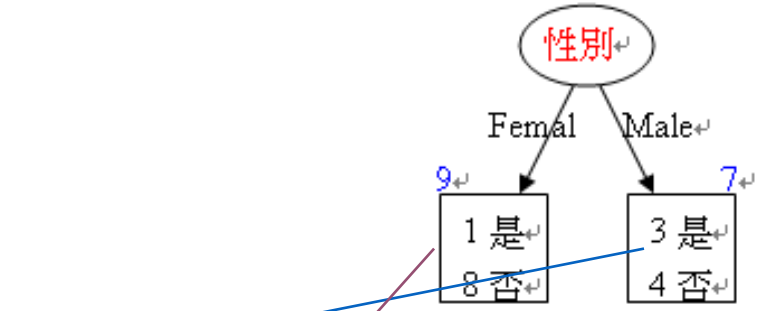


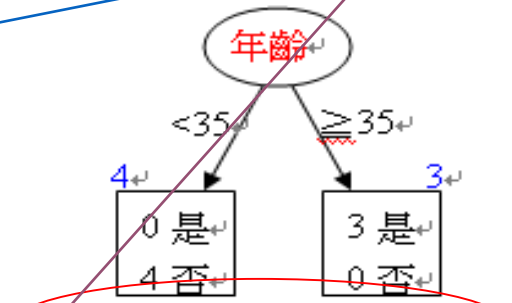
Gain(家庭所得)=0.0177

决策树 - 例5.2（作业）

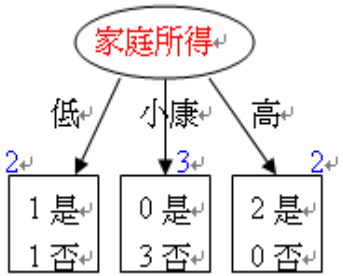
年齡	性別	家庭所得	購買RV房車
<35	Male	小康	否
<35	Male	低所得	否
<35	Male	高所得	否
<35	Male	高所得	否
≧35	Male	小康	是
≧35	Male	小康	是
≧35	Male	低所得	是

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

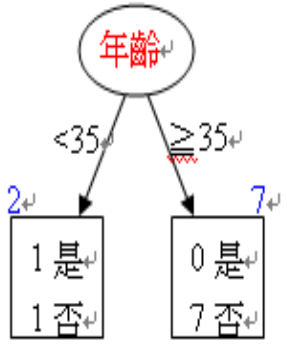




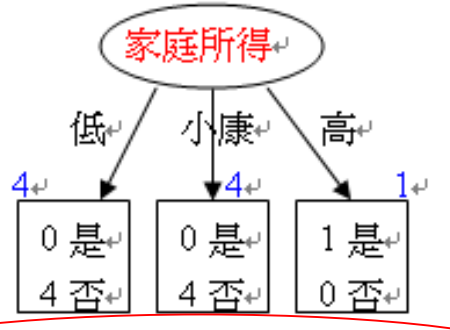
Gain(年齡)=0.9852



Gain(家庭所得)=0.688



Gain(年齡)=0.2222

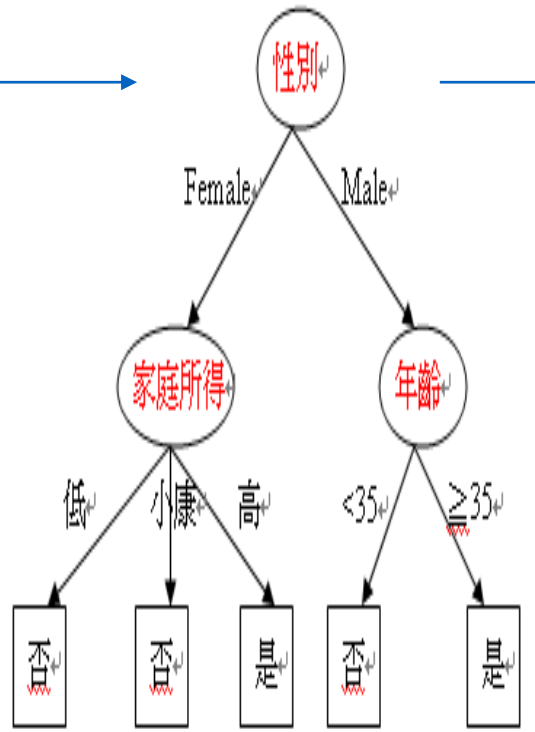


Gain(家庭所得)=0.5032

决策树 - 例5.2（作业）

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

Decision Tree



分类规则：

IF性別=Female AND家庭所得 = 低所得THEN购买RV房车=否

IF性別=Female AND家庭所得 = 小康THEN购买RV房车=否

IF性別=Female AND家庭所得 = 高所得THEN购买RV房车=是

IF性別=Male AND年龄<35 THEN购买RV房车=否

IF性別=Male AND年龄≧35 THEN购买RV房车=是

- 设X是样本，类标号未知
- 设H为某种假设，如样本X属于某个特定类C
- P(H|X)是后验概率，或在条件X下，H的后验概率
  - 例如，X是一位35岁的顾客，其收入为4万美元。令H为某种假设，如顾客将购买计算机
- P(H) (prior probability)是先验概率，或H的先验概率
  - 例如, X将购买电脑, 无论年龄和收入等等
- P(X)是 X的先验概率，可观察到样本数据
  - 例如，顾客集合中年龄为35岁且收入为四万美元的概率

❖ **贝叶斯定理：**

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

朴素贝叶斯分类

- 设D是训练样本和相应类标号的集合。通常，每个样本用一个n维属性向量X = (X_1, X_2, …, X_n) 表示，描述样本的n个属性值
- 设有m个类 C_1, C_2, …, C_m

❖ **给定样本X，分类法将预测X属于具有最高后验概率的类**

- 根据贝叶斯定理 $$P(C_i|\mathbf{X})=\frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$
- 由于 P(X) 对所有类为常数，所以只需将右式最大化 $$P(C_i|\mathbf{X})=P(\mathbf{X}|C_i)P(C_i)$$
- 类条件独立 $$P(\boldsymbol{X}|C_i)=\prod_{k=1}^{n}P(x_k|C_i)=P(x_1|C_i)P(x_2|C_i)\cdots P(x_n|C_i)$$

不足

- 需要计算一些概率值，开销大
- 零概率值问题
- 属性独立性假设问题

不足

❖ **零概率值问题：朴素贝叶斯分类需要每一个条件概率都必须非零，否则预测的概率将为零**

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i)$$

❖ **如何克服：拉普拉斯校准**

➢ **例如：假定样本集有1000 个样本，income=low (0), income= medium (990), and income = high (10), 用拉普拉斯校准为每一类增加 1 个样本，则：**

Prob(income = low) = 1/1003，

Prob(income = medium) = 991/1003，

Prob(income = high) = 11/1003

校准过的概率估计与相应未校准的估计很接近，避免了零概率值。

不足

- **属性独立性假设问题：** 使得朴素贝叶斯分类成为可能，但是实践中很少满足，因为属性（变量）通常是相关的
- **如何克服:**
  - 贝叶斯信念网络，联合属性的贝叶斯推理和因果关系
  - 决策树，在一个时刻只推理一个属性，首先考虑最重要的属性

朴素贝叶斯分类 - 例5.3

例5.3：两类：C1:buys_computer = ‘yes’，C2:buys_computer = ‘no’；希望预测以下样本的类标号：

X = (age <=30,Income = medium,Student = yes，Credit_rating = Fair)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
30…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

朴素贝叶斯分类 - 例5.3

❖ 解：

P(Ci): P(buys_computer = “yes”) = 9/14 = 0.643
P(buys_computer = “no”) = 5/14= 0.357

Compute P(X|Ci) for each class
P(age=“<30” | buys_computer=“yes”) = 2/9=0.222
P(age=“<30” | buys_computer=“no”) = 3/5 =0.6
P(income=“medium” | buys_computer=“yes”)= 4/9 =0.444
P(income=“medium” | buys_computer=“no”) = 2/5 = 0.4
P(student=“yes” | buys_computer=“yes)= 6/9 =0.667
P(student=“yes” | buys_computer=“no”)= 1/5=0.2
P(credit_rating=“fair” | buys_computer=“yes”)=6/9=0.667
P(credit_rating=“fair” | buys_computer=“no”)=2/5=0.4

X=(age<=30 , income =medium, student=yes, credit_rating=fair)
P(X|Ci) :
P(X|buys_computer=“yes”)= 0.222 x 0.444 x 0.667 x 0.0.667 =0.044
P(X|buys_computer=“no”)= 0.6 x 0.4 x 0.2 x 0.4 =0.019
P(X|Ci)*P(Ci) :
P(X|buys_computer=“yes”) * P(buys_computer=“yes”)=0.028
P(X|buys_computer=“no”) * P(buys_computer=“no”)=0.007

Therefore, X belongs to class “buys_computer=yes”

朴素贝叶斯分类 - 例5.4（作业）

❖ 例5.4：检测SNS社区中不真实账号

- 问题是这样的，对于SNS社区来说，不真实账号（使用虚假身份或用户的小号）是一个普遍存在的问题，作为SNS社区的运营商，希望可以检测出这些不真实账号，从而在一些运营分析报告中避免这些账号的干扰，亦可以加强对SNS社区的了解与监管。
- 如果通过纯人工检测，需要耗费大量的人力，效率也十分低下，如能引入自动检测机制，必将大大提升工作效率。这个问题说白了，就是要将社区中所有账号在真实账号和不真实账号两个类别上进行分类。

朴素贝叶斯分类 - 例5.4（作业）

解：

❖ 1、确定特征属性及划分

- 这一步要找出可以帮助我们区分真实账号与不真实账号的特征属性，在实际应用中，特征属性的数量是很多的，划分也会比较细致，但这里为了简单起见，我们用少量的特征属性以及较粗的划分，并对数据做了修改。我们选择三个特征属性：
  - a1：日志数量/注册天数
  - a2：好友数量/注册天数
  - a3：是否使用真实头像
- 在SNS社区中这三项都是可以直接从数据库里得到或计算出来的。
- 下面给出划分：
  - a1：{a<=0.05, 0.05<a<0.2, a>=0.2}
  - a2：{a<=0.1, 0.1<a<0.8, a>=0.8}
  - a3：{a=0（不是）,a=1（是）}

朴素贝叶斯分类 - 例5.4（作业）

- **2、获取训练样本**
  - 这里使用运维人员曾经人工检测过的1万个账号作为训练样本。
- **3、计算训练样本中每个类别的频率**
  - 用训练样本中真实账号和不真实账号数量分别除以一万，得到：
  - P(C = 0) = 8900/10000 = 0.89
  - P(C = 1) = 1100/10000 = 0.11
- **4、计算每个类别条件下各个特征属性划分的频率**
  - P(a1<=0.05| C = 0) = 0.3　　P(a1<=0.05| C = 1) = 0.8
  - P(0.05<a1<0.2|C = 0) = 0.5　　P(0.05<a1<0.2| C = 1) = 0.1
  - P(a1>0.2| C = 0) = 0.2　　P(a1>0.2| C = 1) = 0.1
  - P(a2<=0.1| C = 0) = 0.1　　P(a2<=0.1| C = 1) = 0.7
  - P(0.1<a2<0.8 | C=0) = 0.7　　P(0.1<a2<0.8 | C=1) = 0.2
  - P(a2>0.8| C = 0) = 0.2　　P(a2>0.8| C = 0) = 0.1
  - P(a3 = 0|C = 0) = 0.2　　P(a3 = 1|C = 0) = 0.8
  - P(a3 = 0|C = 1) = 0.9　　P(a3 = 1|C = 1) = 0.1

朴素贝叶斯分类 - 例5.4（作业）

❖ 5、使用分类器进行鉴别

- 下面我们使用上面训练得到的分类器鉴别一个账号，属性如下:
  - a1:日志数量与注册天数的比率为0.1
  - a2:好友数与注册天数的比率为0.2
  - a3:不使用真实头像 (a = 0)
- P(C=0)P(x|C=0)

= P(C=0)P(0.05<a1<0.2|C=0)P(0.1<a2<0.8|C=0)P(a3=0|C=0)

= 0.89*0.5*0.7*0.2=0.0623

- P(C=1)P(x|C=1)

=P(C=1)P(0.05<a1<0.2|C=1)P(0.1<a2<0.8|C=1)P(a3=0|C=1)

= 0.11*0.1*0.2*0.9= 0.00198

- **可以看到：**虽然这个用户没有使用真实头像，但是通过分类器的鉴别，更倾向于将此账号归入真实账号类别。

❖ IF-THEN 规则

- 表达式：IF 条件 THEN 结论。
- 规则R是一个例子：

R: IF age = youth AND student = yes（规则前件）

THEN buys_computer = yes（规则的结论）

❖ 规则质量的度量

- 覆盖率：coverage(R) = n_{covers} /|D|　/* D: training data set */
- 准确率：accuracy(R) = $n_{correct}$ / n_{covers}
  - n_{covers} ： 规则R覆盖的样本数
  - $n_{correct}$ ： 规则R正确分类的样本数

使用IF-THEN规则分类

- 用基于规则的分类预测样本X的类标号，如果规则被满足，则称该规则被**触发**；如果被满足的规则唯一，则称该规则被**激活**，返回X的类预测。注意：触发并不总意味着激活，因为可能有多个规则被满足！
- 如果**多个规则被触发**，则需要一种解决冲突的策略来决定激活哪一个规则，并对X指派它的类预测
  - **规模序:** 把最高优先权赋予具有“最苛刻”要求的被触发的规则，其中苛刻性用规则前件的规模度量，激活具有最多属性测试的被触发的规则。
  - **规则序:**
    - **基于类的序:** 类按“重要性”递减排序，如按普遍性的降序排序
    - **基于规则的序:** 根据规则质量的度量（如准确率、覆盖率），或领域专家的建议，把规则组织成一个优先权列表
- 如果没有规则被触发
  - 建立一个缺省或默认规则，根据训练集指定一个默认类

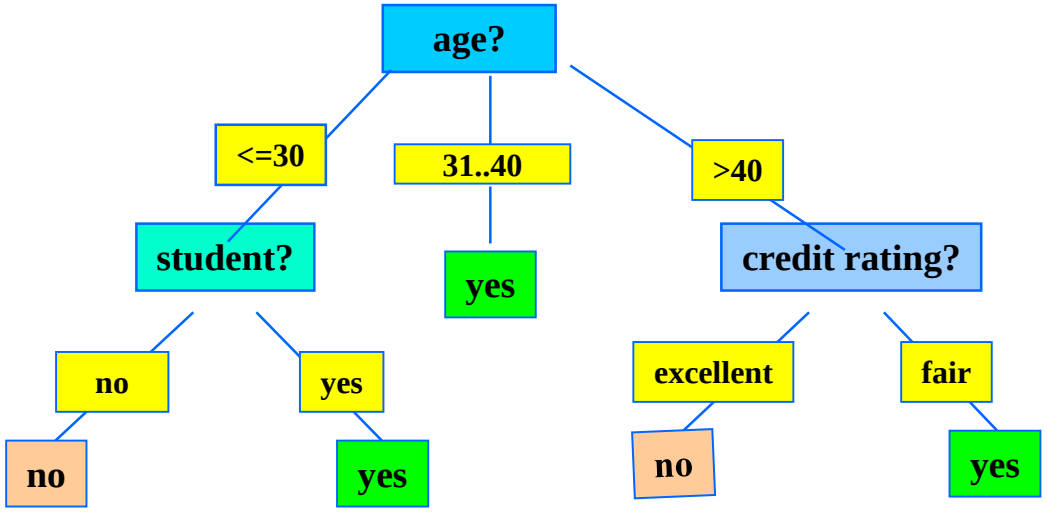
由决策树提取规则

- 规则可能比决策树更容易理解，特别是当决策树大时
- 对每条从根到树叶结点的路径创建一个规则
- 规则是互斥的和穷举的
  - 互斥意味不可能存在规则冲突，因为没有两个规则被相同的样本触发。（每个树叶有一个规则，并且任何样本都只能映射到一个树叶）
  - 穷举意味对于每种可能的属性 - 值组合都存在一个规则，使得该规则集不需要默认规则。因此，规则的序不重要，它们是无序的。
- 由于每个树叶一个规则，所以提取的规则集并不比对应的决策树简单多少！在某些情况下，提取的规则集可能比原来的树更难解释

由决策树提取规则 - 例子

❖ **例：由决策树提取分类规则**

- IF age = young AND student = no THEN buys_computer = no
- IF age = young AND student = yes THEN buys_computer = yes
- IF age = mid-age THEN buys_computer = yes
- IF age = old AND credit_rating = excellent THEN buys_computer = no
- IF age = old AND credit_rating = fair THEN buys_computer = yes



5.4 模型评估与选择

- **分类器的评估度量：**用来评估分类器预测样本类标号的性能或“准确率”。注意，尽管准确率一词是一个特定的度量，但是“准确率”一词也经常用于谈论分类器预测能力的通用术语。
- **分类器的准确率最好在检验集上估计**
- **评估一个分类器准确率的方法**
  - 保持方法, 随机二次抽样
  - 交叉验证
  - 自助法
- **模型选择（即选择一个分类器）**
  - 统计显著性检验
  - 基于成本效益和ROC 曲线

评估分类器性能的度量

- **正样本（P）：感兴趣的主要类的样本。**
- **负样本（N）：其他样本。**
- **真正例（True Positive，TP）：被分类器正确分类的正样本。**
- **真负例（True Negative，TN）：被分类器正确分类的负样本。**
- **假正例（False Positive，FP）：被错误地标记为正样本的负样本。**
- **假负例（False Negative，FN）：被错误地标记为负样本的正样本。**

混淆矩阵

❖ **混淆矩阵**

实际的类\预测的类	yes	no
yes	**TP**	**FN**
no	**FP**	**TN**

❖ **混淆矩阵的例子**

实际的类\预测的类	buy_computer = yes	buy_computer = no	合计
buy_computer = yes	**6954**	**46**	7000
buy_computer = no	**412**	**2588**	3000
合计	7366	2634	10000

❖ **给定m个类，混淆矩阵前m行和m列中的表目$CM_{i,j}$指出类i的样本被分类器标记为类j的个数**

准确性、错误率、灵敏性和特效性

A\P	yes	no	合计
yes	**TP**	**FN**	**P**
no	**FP**	**TN**	**N**
合计	**P′**	**N′**	**P+N**

- **类分布相对平衡**
  - **准确率=灵敏性×P/(P+N)+特效性×N/(P+N)=(TP+TN)/(P+N)**
  - **错误率=(FP+FN)/(P+N)**
- **类不平衡问题：感兴趣的类（正类）是稀少的，即数据集的分布反映负类显著地占多数，而正类占少数，例如“欺诈检测”**
  - **灵敏性(召回率)： 正确识别的正样本的百分比，灵敏性 = TP/P**
  - **特效性：正确识别的负样本的百分比，特效性 = TN/N**

精度、召回率、 F 度量

- 精度（精确性的度量）：即标记为正类的样本实际为正类所占的百分比，

  precision=TP/(TP+FP) = TP/P’

- 召回率（完整性的度量）：即正类的样本标记为正类的百分比，recall=TP/(TP+FN) = TP/P

- F 度量 (F_1 或 F分数)：精度和召回率的调和均值，它赋予召回率和精度相等的权重

  F=2* precision*recall/ precision+recall

- F_β：精度和召回率的加权度量，它赋予召回率权重是赋予精度的β倍

$$F_\beta = \frac{(1+\beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$$

评估分类器性能的度量

- **除了基于准确率的度量，还可以根据其他方面来比较分类器**
  - 准确率：模型正确分类或预测的能力
  - 速度：产生和使用模型的计算花销
  - 健壮性：给定噪声数据或有空缺值的数据，模型正确分类或预测的能力
  - 可伸缩性：对大量数据，有效的构建分类器或预测器的能力
  - 可解释性：学习模型提供的理解和洞察的层次

例子

实际的类\ 预测的类	cancer = yes	cancer = no	合计
cancer = yes	**90（TP）**	**210（FN）**	300（**P**）
cancer = no	**140（FP）**	**9560（TN）**	9700（**N**）
合计	230	9770	10000

- **Accuracy=(90+9560)/10000=96.50%**
- **Sensitivity(Recall)= 90/300 = 30.00%**
- **Specificity=9560/9700=98.56%**
- **Precision = 90/(90+140) = 39.13%**

准确率	（TP+TN）/(P+N)
错误率	（FP+FN）/(P+N)
灵敏性（召回率）	TP/P
特效性	TN/N
精度	TP/(TP+FP)

评估一个分类器准确率的方法

❖ 保持方法

- ➢ 给定的数据随机的划分为两个独立的集合
  - 训练集，通常2/3的数据被分配到训练集
  - 检验集，通常1/3的数据被分配到检验集

❖ 随机二次抽样

- ➢ 保持方法的变形，将保持方法重复k次，总准确率估计取每次迭代准确率的平均值

❖ 交叉验证(k-折交叉验证)

- ➢ 初始数据随机地划分成k个互不相关的子集，每个子集的大小大致相等；训练和检验进行k次；在第i次迭代，分区D_i用作检验集，其他区用作训练集
- ➢ 留一：4每次只给检验集“留出”一个样本
- ➢ 分层交叉验证：折被分层，使的每个折中样本的类分布与在初始数据中的大致相同

评估一个分类器准确率的方法

❖ 自助法

- ➢ 处理较小的数据集合比较有效
- ➢ 从给定训练样本中有放回的均匀抽样
  - 在有放回的抽样中，允许机器多次选择同一个样本

❖ 有多种自助方法, 最常用的一种是 .632 自助法

- ➢ 假设给定的数据集包括d个样本。该数据集有放回地抽样d次，产生d个样本的自助样本集或训练集。结果是，在平均的情况下， 63.2% 的原数据元组将出现在自助样本中，而其余 36.8%的元组将形成检验

使用统计显著性检验选择模型

- 假设已经由数据产生了两个分类模型 M_1 和 M_2，如何确定哪一个更好?
- 进行10折交叉验证，得到每一个分类模型的平均错误率err(M_1) 和 err(M_2)
- M_1 和 M_2 的平均错误率虽不同，但差别可能不是统计显著的，如果二者之间的差别只是偶然的，该如何处理?
  - 统计显著性检验

使用统计显著性检验选择模型

- **进行统计显著性检验，我们需要做什么?**
  - 进行10轮10-折交叉验证
  - 假设它们服从具有k-1个自由度的t分布，其中k=10
  - 利用 t-检验
  - 原假设: M_1 & M_2 相同
  - 如果我们能拒绝原假设, 则
    - 可以断言模型之间的差是统计显著的
    - 在此情况下，我们可以选择具有较低错误率的模型

t-检验

❖ 如果使用单个检验集: 逐对比较

- ➢ 进行10轮10-折交叉验证
- ➢ 使用相同的交叉验证得到 $err(M_1)_i$ and $err(M_2)_i$
- ➢ 对10轮的M_1和M_2 的错误率分别取平均值
- ➢ t- 检验计算样本具有k-1自由度的t-统计量:

$$\text{var}(M_1 - M_2) = \frac{1}{k}\sum_{i=1}^{k}\left[err(M_1)_i - err(M_2)_i - (\overline{err}(M_1) - \overline{err}(M_2))\right]^2$$

❖ 如果有两个检验集

$$var(M_1 - M_2) = \sqrt{\frac{var(M_1)}{k_1} + \frac{var(M_2)}{k_2}},$$

使用统计显著性检验选择模型

- **M1 & M2 是否显著不同?**
  - 计算 t. 选择显著水平sig (e.g. sig = 5%)
  - 查找t分布表
  - 查找 置信界 z=sig/2 (这里, 0.025)
  - 如果 t > z or t < -z, 则t落在拒绝区域，两个模型间存在统计显著差别
  - 否则, 两者之间的差是随机的

基于成本效益和ROC曲线比较分类器

- 真正例、真负例、假正例和假负例可以用于评估与分类模型相关联的成本效益（或风险增益）。作为选择，通过计算每种决策的平均成本（或效益），可以考虑成本效益。
  - 假负例（将癌症患者分类为非癌症患者）相关联的代价比与假正列（将非癌症患者分类为癌症患者）相关联的代价大得多。
  - 类似地，与真正例决策相关联的效益也可能不同于真负例
- 在总体分析中考虑的其他代价包括收集数据和开发分类工具的开销

基于成本效益和ROC曲线比较分类器

接收者操作特征（Receiver Operating Characteristic，ROC）**曲线**是一种比较两个分类模型有用的可视化工具。ROC 曲线源于信号检测理论，是第二次世界大战期间为雷达图像分析开发的。ROC 曲线显示了给定模型的真正例率（*TPR*）和假正例率（*FPR*）之间的权衡[⊖]。给定一个检验集和模型，*TPR* 是该模型正确标记的正（或“*yes*”）元组的比例；而 *FPR* 是该模型错误标记为正的负（或“*no*”）元组的比例。假定 *TP*、*FP*、*P* 和 *N* 分别是真正例、假正例、正和负元组数，由 8.5.1 节，我们知道 $TPR = \frac{TP}{P}$，这是灵敏度。此外，$TFR = \frac{FP}{N}$，它是 1 − *specificity*。

基于成本效益和ROC曲线比较分类器

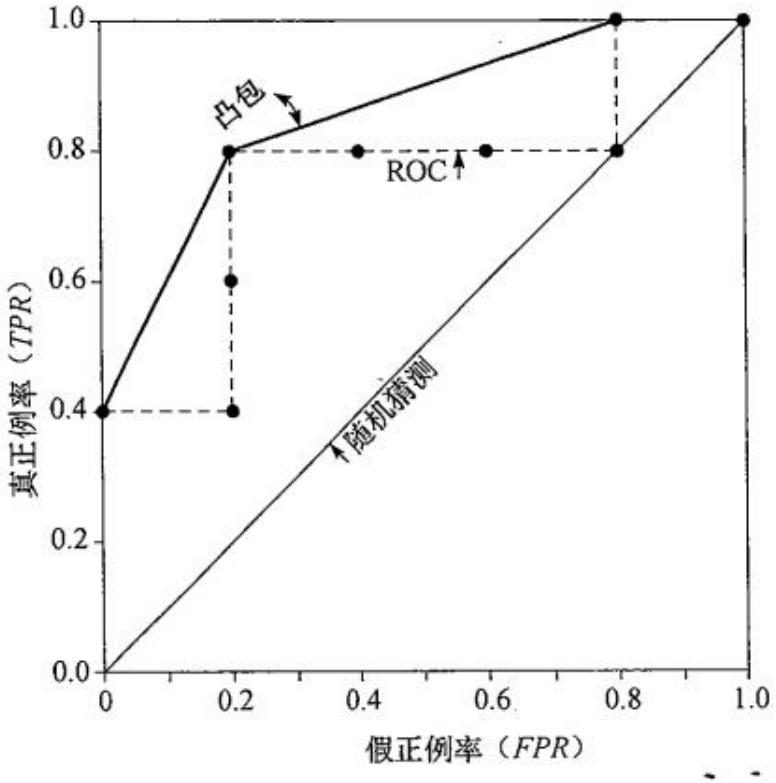


图 8.19　图 8.18 的数据的 ROC 曲线

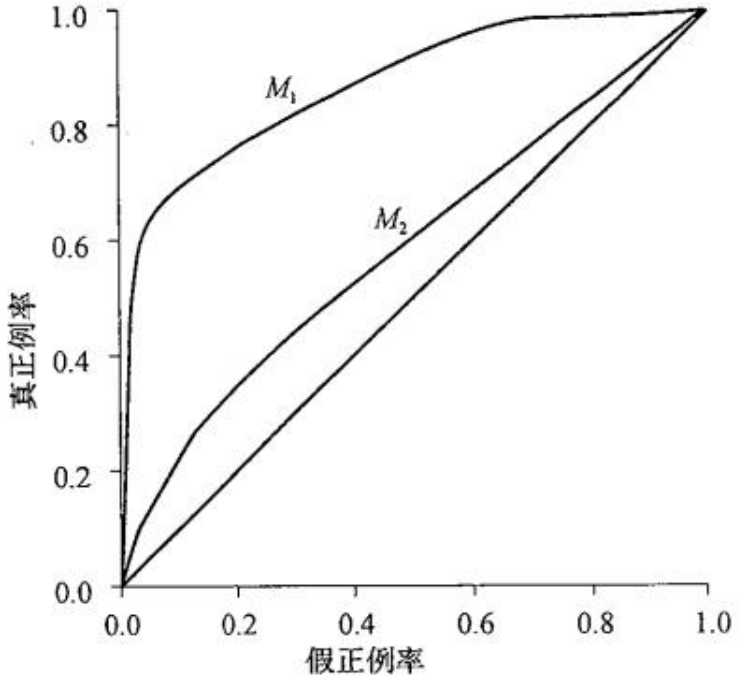


图 8.20　两个分类模型 M_1 和 M_2 的 ROC 曲线。对角线显示，对于每个真正例，都等可能地遇到一个假正例。ROC 曲线越接近该对角线，模型越不准确。因此，M_1 更准确

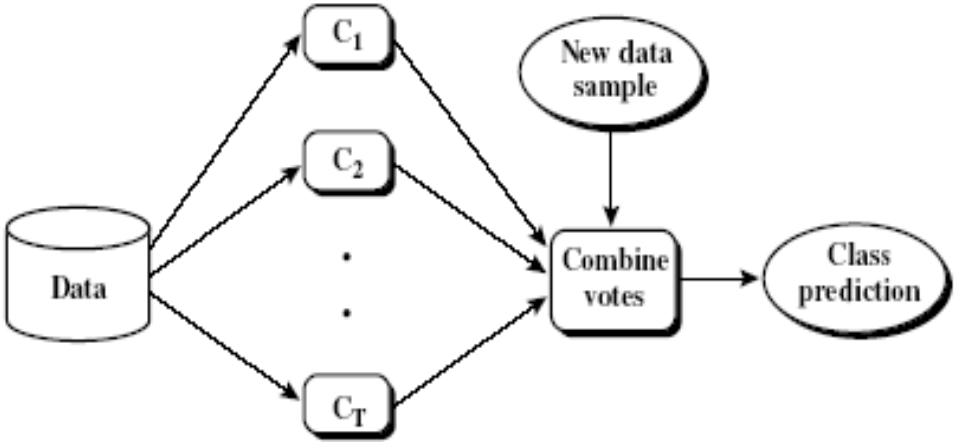
5.6 提高分类准确率的技术

❖ 组合分类方法

- 利用模型（**基分类器**）的组合来提高准确率
- 把k个学习得到的模型, M_1, M_2, …, M_k, 组合在一起，旨在创建一个改进的复合分类模型 M*

❖ 常见的组合分类方法

- 装袋
- 提升
- 随机森林



装袋

- **引例: 最终诊断根据多数表决做出，其中每个医生具有相同的投票权重**
- **步骤：**
  - **给定d个样本的集合D，对于迭代i，d个样本的训练集 D_i 采用有放回抽样，由原始样本集D抽取**
  - **由每个训练集D_i 学习，得到一个分类模型M_i**
  - **为了对未知样本 X 分类，每个分类器 M_i 返回它的类预测**
  - **装袋分类器 M* 统计得票，并将得票最高的类赋予X**
- **准确率：通常高于从原训练集 D导出的单个分类器的准确率**

提升

- 引例: 根据医生先前的诊断准确率，对每位医生的诊断赋予一个权重，然后这些加权诊断的组合作为最终的诊断。
- 怎么提升?
  - 权重赋予每个训练样本
  - 迭代地学习k个分类器
  - 学习得到分类器 M_i 后，更新权重， 使其后的分类器 M_i+1 “更关注” M_i 误分类的训练样本
  - 最终提升的分类器 M* 组合每个个体分类器的表决，其中每个分类器投票的权重是其准确率的函数

Adaboost

- **Adaboost是一种流行的提升算法**
- **步骤：**
  - 给定数据集D，它包含d个类标记的样本：$(X_1, y_1), \ldots, (X_d, y_d)$
  - 开始，对每个训练样本赋予相等的权重 (1/d)
  - 为组合分类器产生k个基分类器需要执行算法其余部分k轮
    - 在第i轮，从D中样本抽样，形成大小为d的训练集D_i，使用有放回抽样——同一个样本可能被选中多次。每个样本被选中的机会由它的权重决定。从训练集D_i导出分类器M_i。然后使用D_i作为检验集计算M_i的误差。训练样本的权重根据分类情况调整（正确分类，权重减少，不正确分类，权重增加）。

Adaboost

- **Adaboost是一种流行的提升算法**
- **涉及的数学问题：**
  - **错误率error(X_j) 是样本 X_j 的误分类误差**

$$error(M_i) = \sum_{j}^{d} w_j \times err(\mathbf{X_j})$$

  - **M_i的表决权重**

$$\log \frac{1-error(M_i)}{error(M_i)}$$

提高类不平衡数据的分类准确率

❖ 类不平衡问题

- ➢ 感兴趣的主类只有少量样本代表，而大多样本都代表负类
- ➢ 类不平衡问题与代价敏感学习密切相关，那里每个类的错误代价并不相等，其中较高代价的类比较低代价的类稀少

❖ 传统的分类算法

- ➢ 旨在最小化分类误差，假定类平衡分布和相等的错误代价，不适合类不平衡数据

❖ 提高类不平衡数据分类准确率的方法

- ➢ 过抽样
- ➢ 欠抽样
- ➢ 阈值移动
- ➢ 组合技术