北京郵電大學
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

大数据时代的管理

Management in Big Data Era

马宝君 博士 讲师

经济管理学院
电子商务中心
2014年12月1日

12月15日上课时间调整到12月19日

2014 年 12 月				December		
星期一	星期二	星期三	星期四	星期五	星期六	星期日
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15 18:30-20:20 此次课程不上	16	17	18	19 改到周五 18:30-20:20	20	21
22	23	24	25	26	27	28
29	30	31				

课前分享1：莫斯科地铁客流量一览图

课前分享2：大数据告诉你美国最赚钱的十大行业

- Powerlytics是一家为数百万企业提供经济预测的大数据分析公司。其分析显示，在美国利润最丰厚的行业是电气设备制造业。
- 通过从美国人口普查和劳工部（Census and the Department of Labor）等获得的20多个公开数据组，Powerlytics建立了一个囊括全美所有2,700万家企业的资料库。
- 排名是根据息税这就及摊销前的营业收入占销售额的百分比计算的。

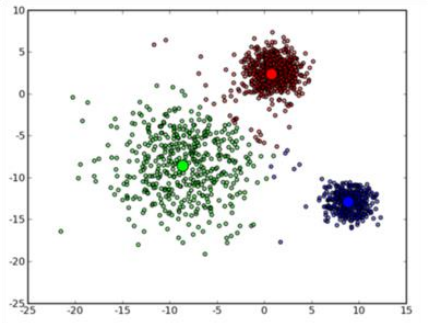
课程回顾：深度业务分析——原方法

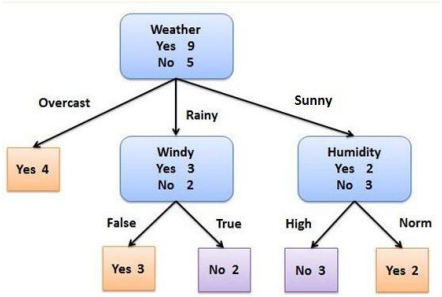
- 聚类（Clustering）
- 分类（Classification）
- 关联（Association）
- 模式（Pattern）
- ……

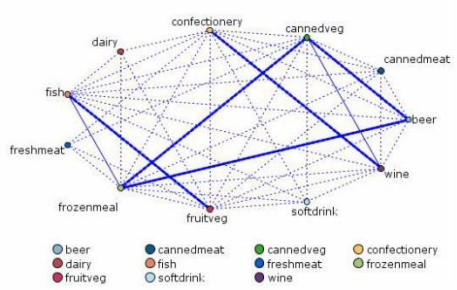
类别

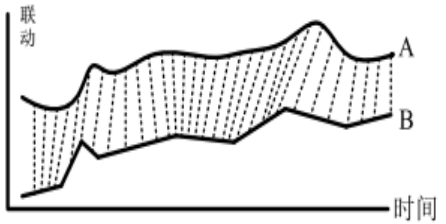
联系

轨迹









上次课程内容回顾

- 关联规则挖掘相关的基本概念
- 关联规则挖掘的有效方法
- 不同种类的关联规则
- 关联规则的应用

分类、预测分析方法

Classification & Prediction Analysis

分类、预测分析（Classification & Prediction）

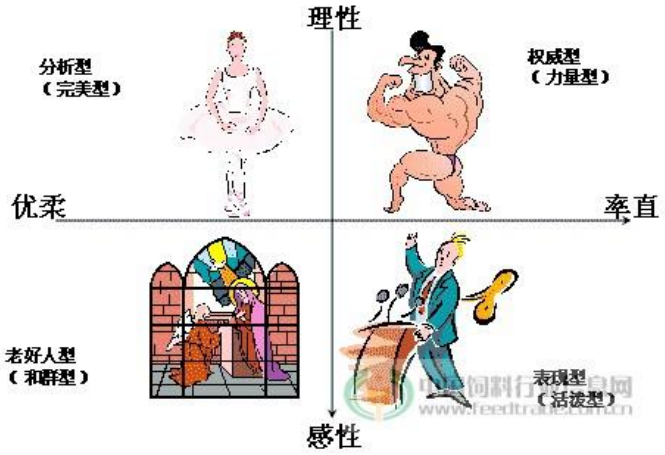
- 基本概念
- 分类分析的经典方法
- 预测分析的常用方法
- 分类、预测方法的评估
- 分类方法的应用案例
- 总结

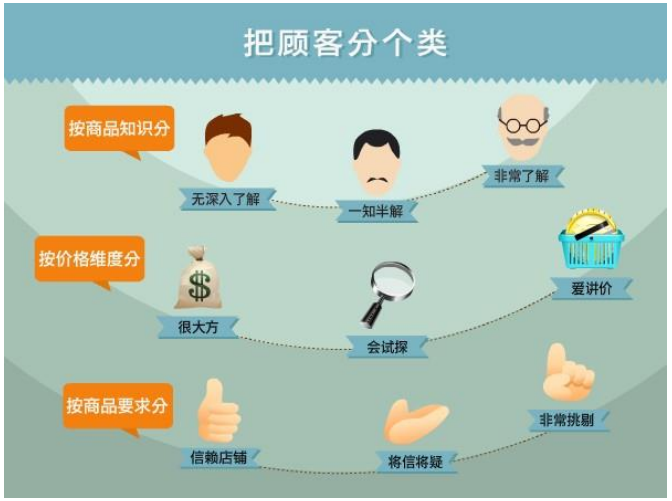
分类、预测分析（Classification & Prediction）

- 基本概念
- 分类分析的经典方法
- 预测分析的常用方法
- 分类、预测方法的评估
- 分类方法的应用案例
- 总结

基本概念

- 分类(Classification)是商务智能中重要且应用非常广泛的决策方法。简单说来，分类所要解决的问题是**将一个事件或者对象划分到给定的类别上。**
- 例如，我们可以基于收入水平、工作情况等对给定客户进行信用风险分析，确定客户的风险等级。在实际生活中，和风险定级一样，生物辨别、图书归类、航空常旅客类别、疾病诊断、行业划分、产品类型、人口统计等分类决策的例子比比皆是。





分类的例子

	食肉	产奶	有鳍	有毒	类别
1	1	0	1	0	鱼类
2	1	1	0	0	哺乳动物
3	0	0	1	0	鱼类
4	0	1	0	1	哺乳动物
5	1	0	1	1	鱼类
6	1	0	0	1	爬行动物
7	1	0	1	0	鱼类
8	1	0	0	1	爬行动物
9	1	1	0	0	哺乳动物
10	0	0	0	1	爬行动物
11	0	0	0	1	爬行动物

假设我们有历史数据，反映了动物类别的一些信息（如上表所示）。
于是我们可以根据这些数据来描述动物属性与类别之间的对应关系。进而，这些对应关系就可以成为对其它未知类别的动物进行分类的“标准”。具体说来，根据表中数据记录的属性取值（食肉，有鳍，产奶，有毒）和类别取值（鱼类，哺乳动物，爬行动物）的对应情况，可能获得的分类结果为：

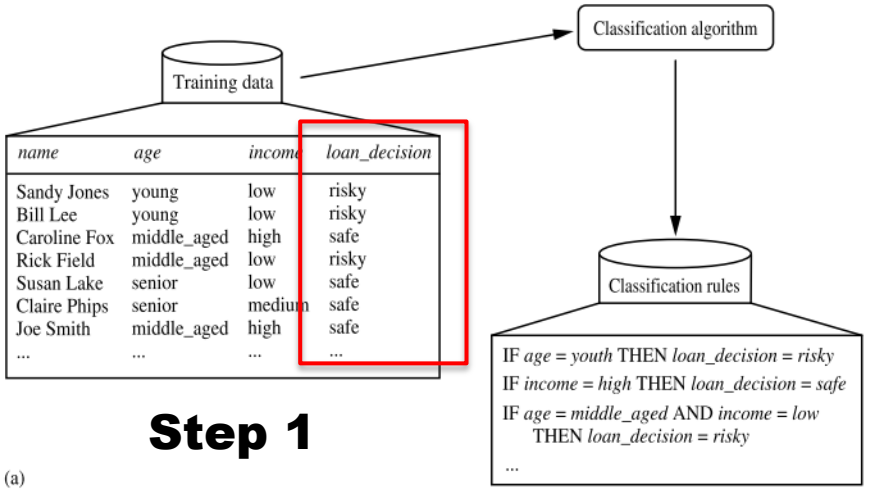
- 如果“有鳍”则为“鱼类”；
- 如果“无鳍”且“产奶”则为“哺乳动物”；
- 如果“无鳍”且“不产奶”则为“爬行动物”。

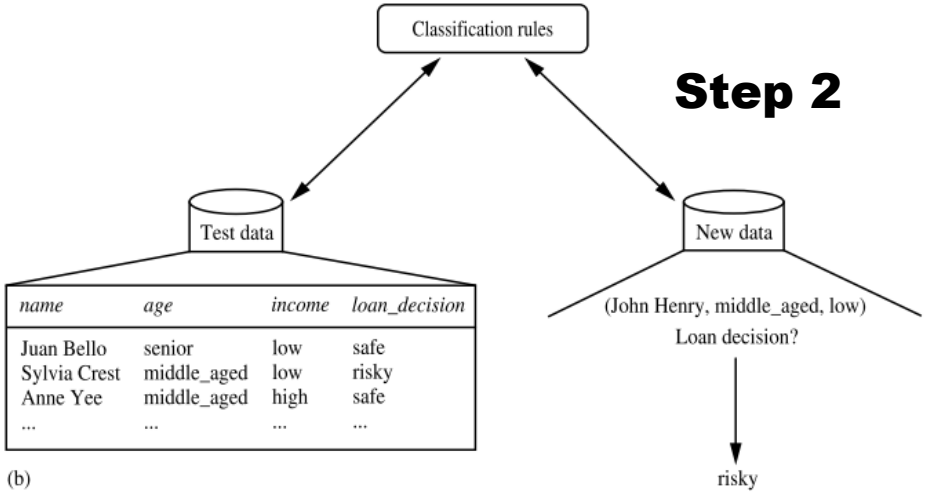
分类分析的类别

- **懒惰型分类方法（Lazy Classification）**
  - 不构造分类器，而是仅仅将训练集保存起来或只对训练集做简单分析，当需要对新记录进行分类时，在保存的记录中寻找与之最相似的样本，根据这个样本的类别来分类
  - 懒惰型学习方法进行分类时需要进行大量运算，对存储效率及并行运算等有很高的要求
- **急切型分类方法（ Eager Classification）**
  - 通过给定的训练集构造一个分类器，利用该分类器对新记录分类

基本概念

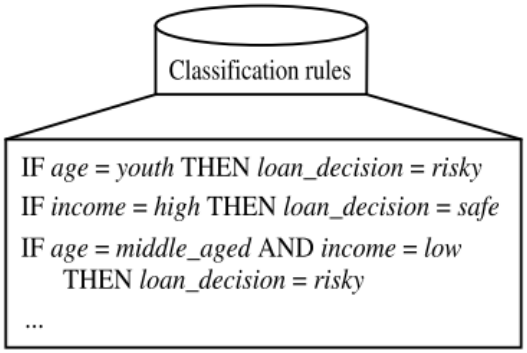
- （急切型）分类分析一般包含两步过程
  - Step 1: 通过分析已知的数据总结出分类模型或分类器
    - 分类器 Classifier
    - 类标签 Class Label
    - 训练集、训练样本 Training data set
  - Step 2: 使用获得的分类器对数据的类别进行预测
    - 测试集 Test data set

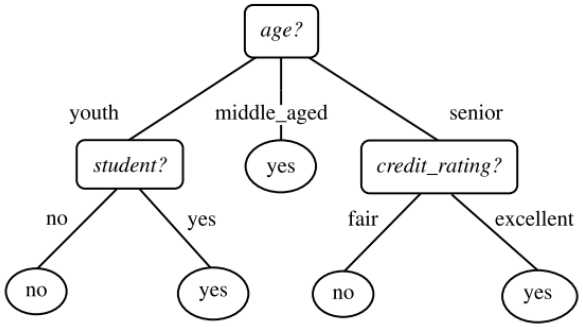


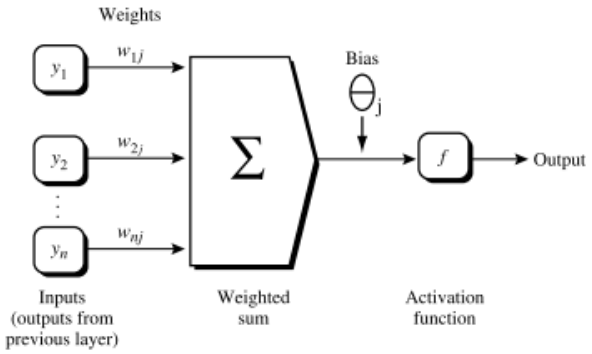


分类 v.s. 聚类

- **分类分析**
  - 有监督学习（有指导学习）：Supervised Learning
  - 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
  - 通常，学习模型通过**分类规则、决策树或数学公式**的形式提供
- **聚类分析**
  - 无监督学习（无指导学习）：Unsupervised Learning
  - 训练样本的类标识是未知的，要学习的类集合或数量也可能事先未知







分类 v.s. 预测

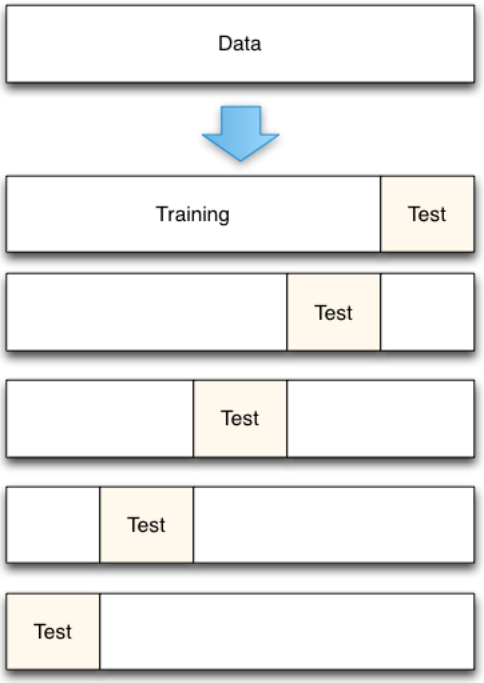
- **广义概念**
  - 预测是构造和使用模型评估无标号样本，或评估给定样本可能具有的属性值或值区间
  - 在宏观概念上，分类和回归是两类主要的预测问题
  - 分类是预测离散值或类别值；回归用于预测连续值
- **狭义概念**
  - 狭义上，预测类标号的为分类，预测连续值的为预测

分类方法的结果评估

- **分类的准确性（Accuracy）**
  - 最基础、最重要，模型正确预测新的或先前未见过的数据的类标签的能力
- **分类的速度（Speed）**
  - 计算成本
- **分类器的鲁棒性（Robustness）**
  - 给定噪音数据或有遗漏值的数据，模型正确预测的能力
- **分类器的可扩展性（Scalability）**
  - 给定大规模数据，有效构建分类模型的能力
- **分类器的可解释性（Interpretability）**
  - 分类器表示的知识被用户理解的程度

分类实践中需要关注的问题

- 训练集 和 测试集的选择
  - Training set (2/3), Test set (1/3)
  - 随机选择
  - Problem 1: What if all examples with a certain class were missed out of the training set?
- 交叉验证（Cross-validation）
  - Every example is used in training and testing in turn
  - Folds: partition of the data
  - 常用：ten-fold cross-validation
  - Leave-one-out（留一法）
  - Problem 2: What if the number of all data examples was too small?



分类实践中需要关注的问题（续）

- 自引导、重采样（Bootstrap）
  - Sampling with replacement（有放回的随机采样）
  - Dataset with *n* instances is sampled *n* times → training set
  - Instances which are not picked → test set
  - Every time, Probability of being picked: $\frac{1}{n}$
  - Probability of not being picked: $1-\frac{1}{n}$
  - Probability of an instance not picked by *n* times：

$$\left(1-\frac{1}{n}\right)^{n} \approx e^{-1} = 0.368$$

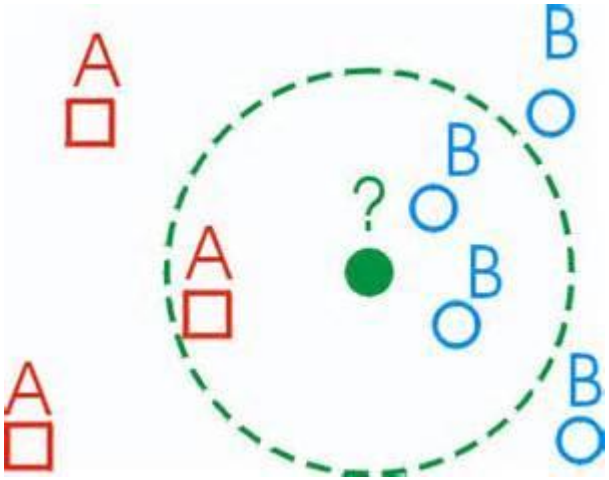
分类、预测分析（Classification & Prediction）

- 基本概念
- 分类分析的经典方法
- 预测分析的常用方法
- 分类、预测方法的评估
- 分类方法的应用案例
- 总结

分类分析典型方法

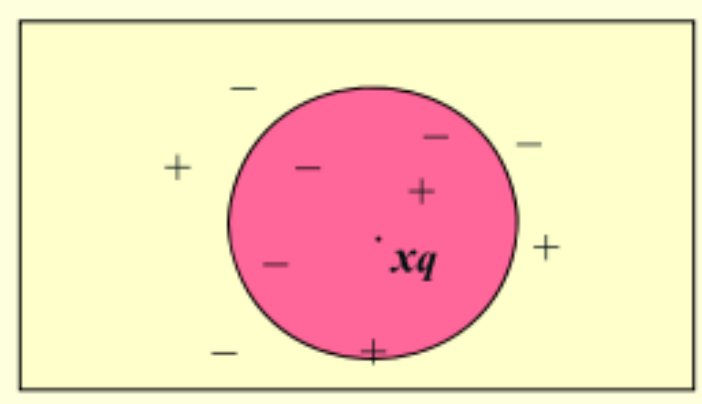
- K近邻方法（K nearest Neighbors）
- 决策树方法（Decision Tree）
- 朴素贝叶斯分类方法（Naïve Bayes）
- 关联分类方法（Associative Classification）
- 神经元网络方法（Neural Network）
- 支持向量机方法（Support Vector Machines）

K nearest Neighbors



K nearest Neighbors

- **懒惰型分类方法**
- **基本思想**
  - 如果一个样本在特征空间中的k个最相似(即特征空间中最邻近)的样本中的大多数属于某一个类别，则该样本也属于这个类别
- **最近邻一般基于欧拉距离来确定**



$$d(i,j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+...+|x_{i_p}-x_{j_p}|^2)}$$

K=3

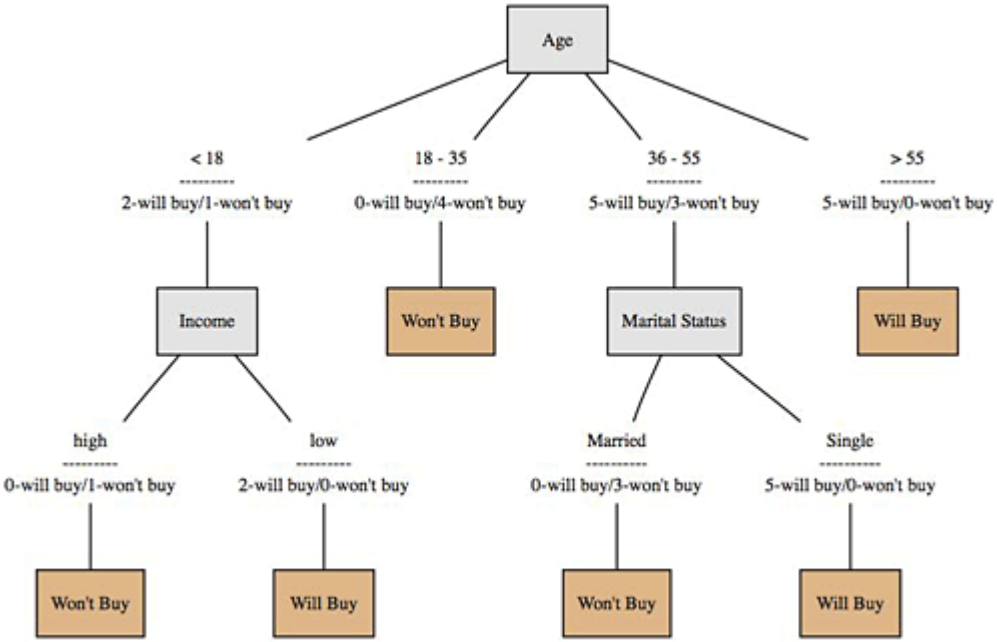
Sepal：
萼片

Petal：
花瓣

sep_length	sep_width	pet_length	pet_width	type
5.7	2.9	4.2	1.3	Iris-versicolor
6.2	2.9	4.3	1.3	Iris-versicolor
5.7	2.8	4.1	1.3	Iris-versicolor
6.3	3.3	6.0	2.5	Iris-virginica
5.8	2.7	5.1	1.9	Iris-virginica
7.1	3.0	5.9	2.1	Iris-virginica
5.1	3.8	1.6	0.2	Iris-setosa
4.6	3.2	1.4	0.2	Iris-setosa
5.3	3.7	1.5	0.2	Iris-setosa

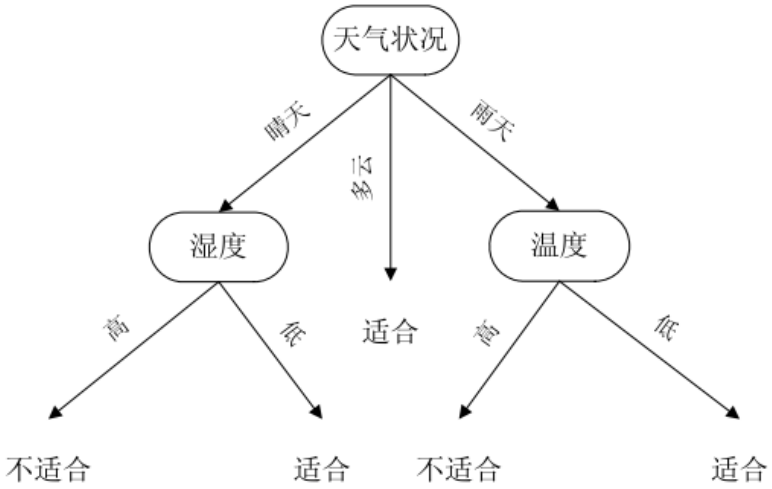
5.0	3.3	1.4	0.2	Iris-setosa
5.1	2.5	3.0	1.1	Iris-versicolor
6.3	2.9	5.6	1.8	Iris-virginica

Decision Tree



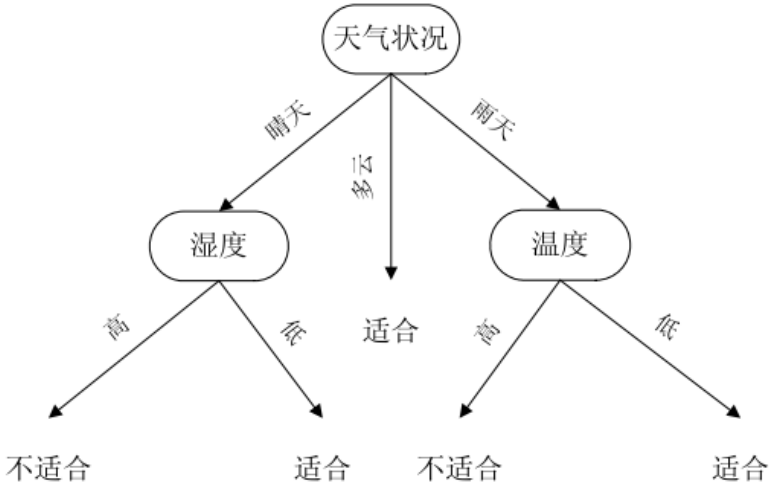
决策树（Decision Tree，DT）

- **决策树是一个类似流程图的树型结构，决策树分类方法以树的形式采用自上而下的方式给出分类规则**
- **决策树的结构**
  - 内部节点（Nodes）：分裂属性
  - 叶子节点（Leaves）：类别
- **优势**
  - 具有较好的分类准确率
  - 树状结构表达直观，易于理解
- **多种决策树分类方法**
  - **ID3, C4.5, CART, SLIQ 等**



决策树（Decision Tree，DT）分类

- **两个主要部分**
  - 决策树的构建
  - 决策树剪枝

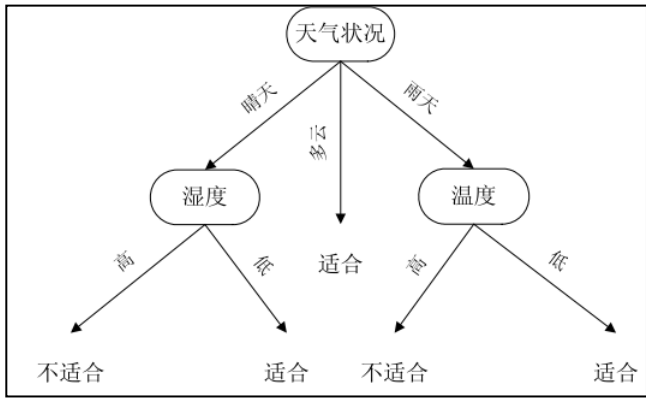


- **决策树构建**
  - 基本思想：从上至下递归地从所有可选的属性中选择最优的分裂属性，直至满足某个结束条件为止
  - 所谓的“最优”意为根据该属性上的不同值能够把训练集分为彼此之间“差异”最大的几部分
  - 决策树通常要求属性值为离散值，如果是连续的属性，则需要将连续值离散化。例如“温度”是连续值，但为了能够将训练集分成有限个部分，则将“温度”离散为“高”和“低”两种值

决策树构建的常用方法

方法：Generate_decision_tree(S, CA) 根据给定数据集产生一个决策树。
输入：训练集S，各属性均取离散数值；候选属性集CA。
输出：决策树。
处理流程：

1. 创建一个结点N；
2. if该结点中的所有样本均为同一类别C，then
3. 返回N作为一个叶结点并标志为类别C；
4. if候选属性集为空，then
5. 返回N作为一个叶结点并标志为默认类别C_default；
6. 从候选属性集中选择最优分裂属性A,将结点N标记为A,从CA中删除A；
7. 对于A中的每一个已知取值Vi
8. 为A建立测试为A=Vi的分枝；
9. 设S_i为A=Vi所对应的样本集；
10. if S_i为空，then
11. 创建叶结点并标志为默认类别C_default；
12. else 加上结点Generate_decision_tree(S_i, CA)



- 默认类别C_default: 多数投票（majority voting）方式来确定
- 每一个递归构建过程的终止条件：
  - → 当前结点的所有样本均为同一类别；
  - → 候选属性集为空，此时标记该结点为默认类别C_default；
  - → 某一分枝没有符合测试条件的样本，创建一个叶结点并将其标记为默认类别C_default

例子：构建决策树

Class-labeled training tuples from the *AllElectronics* customer database.

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

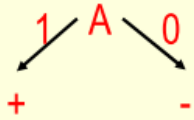
如何选择分裂属性：例子

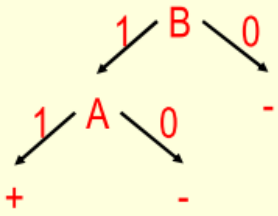
• Consider data with two Boolean attributes (A,B).

A	B	class	count
0	0	-	50
0	1	-	50
1	0	+	0
1	1	+	100

• What should be the first attribute we select?

• Splitting on A: we get purely labeled nodes.





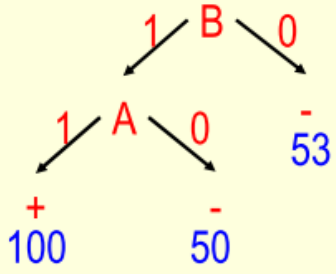
•Splitting on B: we don't get purely labeled nodes.

• What if we have: <(A=1,B=0), - >: 3 examples

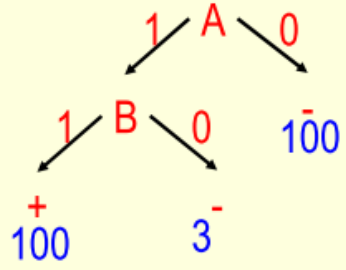
• Consider data with two Boolean attributes (A,B).

A	B	class	count
0	0	-	50
0	1	-	50
1	0	-	3
1	1	+	100

• Trees looks structurally similar; which attribute should we choose?



Which one is better?



如何选择分裂属性：原则

- **整体目标使得得到的决策树尽可能简洁（小）**
  - Occam's Razor（奥卡姆剃刀定律）：如无必要，勿增实体
- **选择的分裂属性最好使得落入给定划分的样本尽可能属于同一个类别（即每一个划分尽可能纯净）**
  - If we were to split D into smaller partitions according to the outcomes of the splitting criterion, ideally each partition would be pure (i.e., all of the tuples that fall into a given partition would belong to the same class)
  - 使得每一个划分尽可能接近于某个叶子节点的情况
- **最流行、最常用的启发式（heuristics）方法是基于信息增益（information gain）和基尼系数（Gini index）**

信息增益：Information Gain

- **信息量、信息不确定性**
  - 信息论（Claude Shannon, 1948）：信息熵（Information Entropy）
  - 信息熵是信息论中用于度量信息量的一个概念。一个系统越是有序，信息熵就越低；反之，一个系统越是混乱，信息熵就越高。所以，信息熵也可以说是系统有序化程度的一个度量

$$info(T) = -\sum_{k=1}^{g} \frac{freq(C_k, T)}{|T|} \times \log_2 \left(\frac{freq(C_k, T)}{|T|} \right)$$

- **信息增益：信息熵的减少、信息不确定性的降低**
  - 信息增益方法选择具有最高信息增益（信息熵减少的程度最大）的属性作为当前结点的分裂属性，以便使划分获得的训练样本子集进行分类所需要信息最小
  - 选择该分裂属性对样本集合划分，将会使得所产生的各样本子集中的“不同类别混合程度”降为最低
  - 因此采用这样一种方法将帮助有效减少对象分类所需要的次数，从而确保所产生的决策树比较简洁，尽管不一定是最简洁的（启发式）

信息增益：Information Gain

- 选择具有**最高信息增益**的属性进行分裂
- 集合D包含|D|个样本对象，对象可以划分为m个类别（$C_1,C_2,\ldots,C_m$）每一个类别中包含的样本数量为$|D_i|$，i={1,2,…,m}
- 用来给任意样本对象分类所需的平均（期望）信息量（信息熵）为：

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

- 某个属性A有n个不同的取值，可以将集合D分为n个子集合，在此划分的基础上给任意样本对象分类还需要的平均（期望）信息量（信息熵）为：

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

- 信息增益：

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

信息增益：Information Gain

- 信息熵的一些特点：
  - 如果所有的样本都数以同一个类别，信息熵为多少？

$$Info(D) = -1 \times \log_2(1) = 0$$

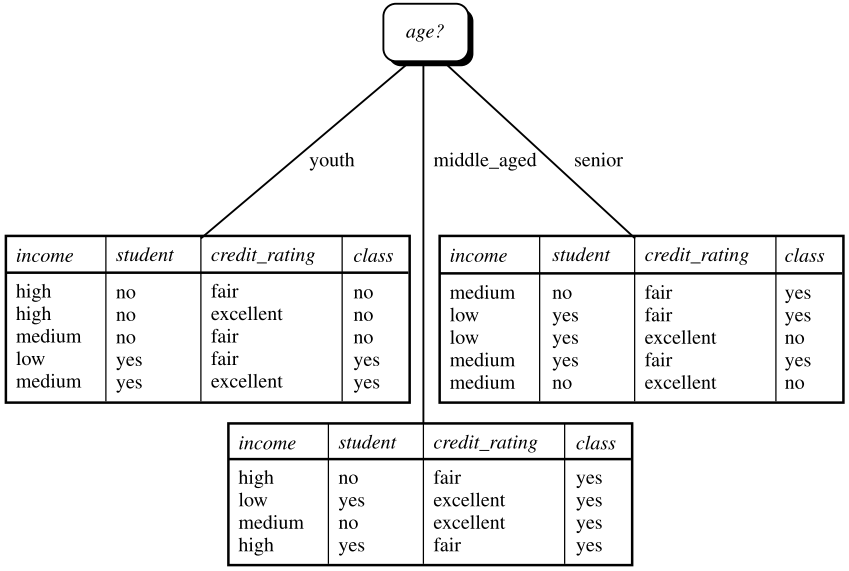
  - 如果所有样本在m个类别上均匀分布，信息熵为多少？

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

利用信息增益选择分裂属性

Class-labeled training tuples from the *AllElectronics* customer database.

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no



$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ bits.}$$

$$\begin{aligned} Info_{age}(D) &= \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) \\ &+ \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}\right) \\ &+ \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right) \\ &= 0.694 \text{ bits.} \end{aligned}$$

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.694 = 0.246 \text{ bits.}$$

$Gain(income) = 0.029$ bits

$Gain(student) = 0.151$ bits

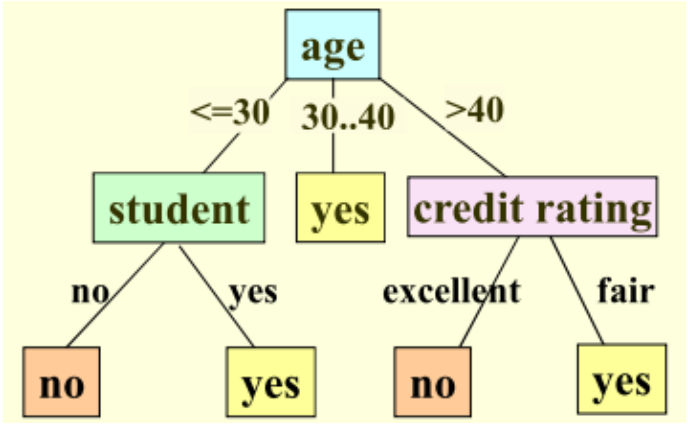
$Gain(credit\_rating) = 0.048$ bits

决策树的剪枝

- **当利用训练集生成决策树之后，树的很多分枝属于噪音或者会对分类准确率造成负面影响，这种情况我们称作模型“过适应于（Overfitting）”数据，因此需要对决策树进行剪枝来提高决策树的分类能力。**
- **先剪枝**
  - 通过提前停止生成分枝对决策树进行剪枝，可以利用信息增益等测度来对分枝生成情况（优劣）进行评估
- **后剪枝**
  - 首先完全地构建一个决策树，然后删除不必要的结点和对应的分枝
  - 基于代价复杂性方法（分类错误率）
- **两种剪枝策略的比较**
  - 从过程上看，后剪枝方法经过了“构建”到“剪枝”这样的过程，显然它要比事前剪枝需要更多的计算时间
  - 对应的，后剪枝可以获得更可靠的决策树
  - 先剪枝可以与后剪枝方法相结合，从而构成一个混合的剪枝方法

由决策树提取分类规则

- 可以提取决策树表示的知识，并以“IF-THEN”形式的分类规则表示
- 对从根节点到叶子节点的每条路径创建一个规则：易于理解



IF *age* = "<=30" AND *student* = "*no*"
THEN *buys_computer* = "*no*"
IF *age* = "<=30" AND *student* = "*yes*"
THEN *buys_computer* = "*yes*"
IF *age* = "31…40"
THEN *buys_computer* = "*yes*"
IF *age* = ">40" AND *credit_rating* = "*excellent*"
THEN *buys_computer* = "*yes*"
IF *age* = "<=30" AND *credit_rating* = "*fair*"
THEN *buys_computer* = "*no*"

其他一些扩展的决策树分类方法

- SLIQ (EDBT'96 — Mehta et al.)
  - builds an index for each attribute and only class list and the current attribute list reside in memory
- SPRINT (VLDB'96 — J. Shafer et al.)
  - constructs an attribute list data structure
- PUBLIC (VLDB'98 — Rastogi & Shim)
  - integrates tree splitting and tree pruning: stop growing the tree earlier
- RainForest (VLDB'98 — Gehrke, Ramakrishnan & Ganti)
  - separates the scalability aspects from the criteria that determine the quality of the tree
  - builds an AVC-list (attribute, value, class label)

本次课程小结

- **分类、预测分析的基本概念**
  - 懒惰型分类 v.s. 急切性分类
  - 分类分析 v.s. 聚类分析
  - 分类分析 v.s. 预测分析
  - 分类方法评估的角度
  - 训练集、测试集、交叉验证、Bootstrap

- **分类分析的典型方法**
  - K近邻方法（K nearest Neighbors）
  - 决策树方法（Decision Tree）

期末课程论文说明

- **主题要求**
  - 必须与“大数据管理”相关
  - 建议围绕所学专业背景下的“大数据管理问题”展开
- **内容要求**
  - 不少于4000字，版式：word中正文小四字体，1.5倍行距
  - 独立完成，不得大段拷贝或直接引用网上、书上及他人已发布内容，需要适当引用时请在引用位置注明参考文献来源（查重）
  - 论文内容框架（建议）：
    - 1. 学习本课程的心得体会、感受，对本课程教学的建议和意见（必有）
    - 2. 论文背景介绍
    - 3. 论文涉及的大数据问题及管理需求、策略和意义（可举实例说明）
    - 4. 本人对该大数据问题的看法、观点及讨论
    - 5. 总结
    - 6. 参考文献和资料

期末课程论文说明（续）

- **论文提交要求**
  - 需要以电子版提交，建议提交word版本
  - 作业提交邮箱：bigdata_homework@163.com
  - 作业提交截止时间：**第19周周日（2015.01.11）24时**
- **其他说明**
  - **电子版论文文件请务必按照“学号_班级_姓名.docx”命名，例如“2014211234_2014212103_张三.docx”，也请在邮件中留下姓名、学号及联系方式，以备论文有问题时能够联系到；**
  - 请在截止时间之前提交论文（不要在截止时间附近，以避免系统原因过期），过期将不再接收论文提交，成绩为0，请务必注意；
  - 每次提交论文后，作业邮箱都会有“已收到邮件”的自动回复，如未收到自动回复，表示发送不成功，请在截止时间内重新提交；
  - 论文评分的关注重点
    - 有效的课程建议和意见
    - 关注问题的新颖度
    - 个人分析和讨论的深度
    - 论文的整体工作量