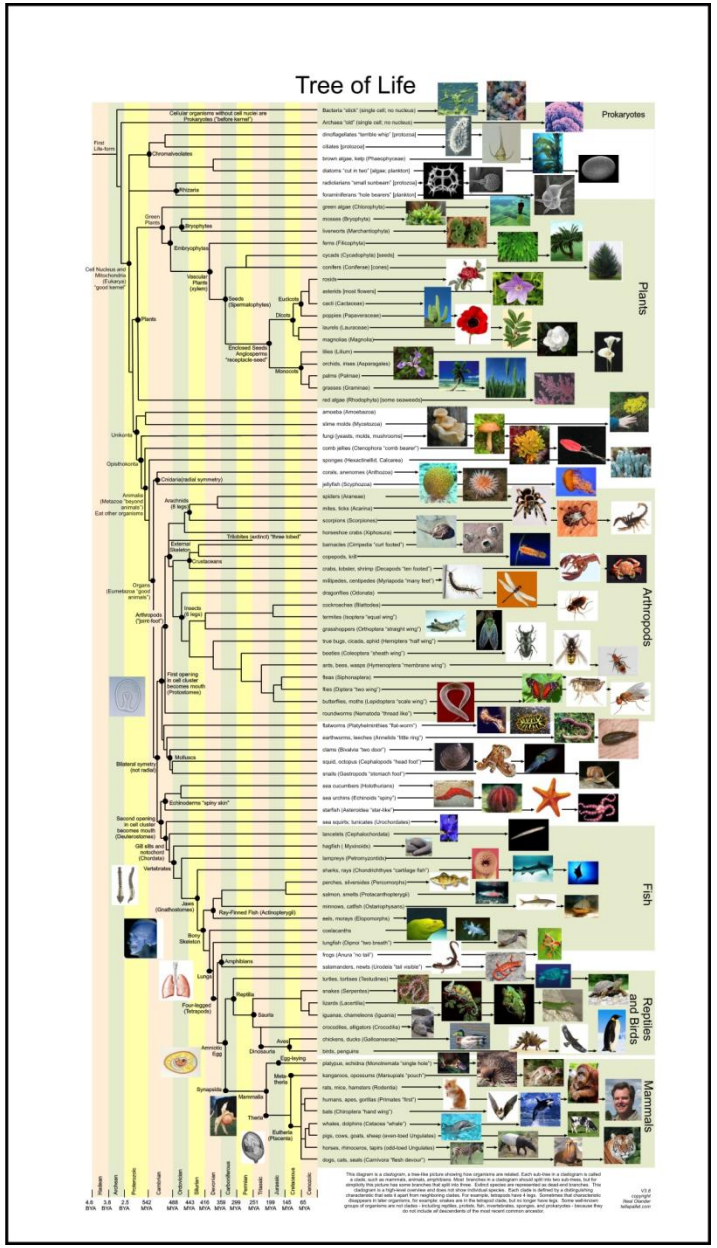
DATAGURU 炼数成金

数据分析、展现

与R语言 第13周



2013.4.28

法律声明

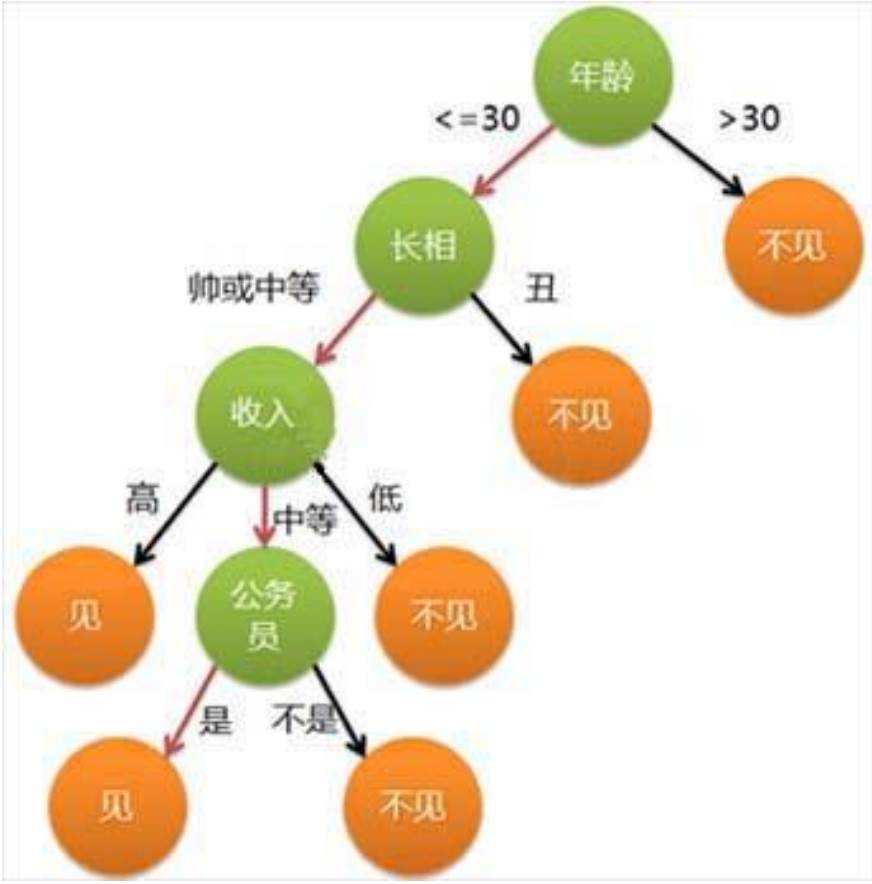
【声明】本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

http://edu.dataguru.cn

决策树 decision tree

- 什么是决策树
- 输入：学习集
- 输出：分类规则（决策树）



例子

- 用SNS社区中不真实账号检测的例子说明如何使用ID3算法构造决策树。为了简单起见，我们假设训练集合包含10个元素。其中s、m和l分别表示小、中和大。

日志密度	好友密度	是否使用真实头像	账号是否真实
s	s	no	no
s	l	yes	yes
l	m	yes	yes
m	m	yes	yes
l	m	yes	yes
m	l	no	yes
m	s	no	no
l	m	no	yes
m	s	no	yes
s	s	yes	no

信息增益

- 设L、F、H和R表示日志密度、好友密度、是否使用真实头像和账号是否真实，下面计算各属性的信息增益。

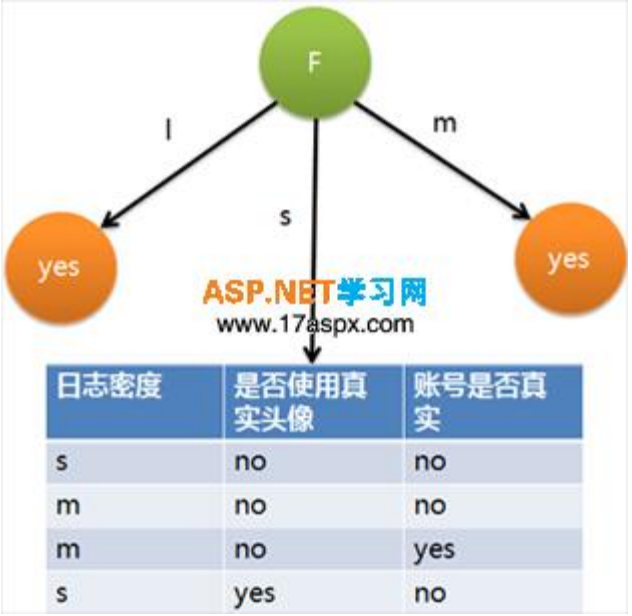
$$info(D) = -0.7log_20.7 - 0.3log_20.3 = 0.7 * 0.51 + 0.3 * 1.74 = 0.879$$

$$info_L(D) = 0.3 * (-\frac{0}{3}log_2\frac{0}{3} - \frac{3}{3}log_2\frac{3}{3}) + 0.4 * (-\frac{1}{4}log_2\frac{1}{4} - \frac{3}{4}log_2\frac{3}{4}) + 0.3 * (-\frac{1}{3}log_2\frac{1}{3} - \frac{2}{3}log_2\frac{2}{3}) = 0 + 0.326 + 0.277 = 0.603$$

$$gain(L) = 0.879 - 0.603 = 0.276$$

根据信息增益选择分裂属性

- 因此日志密度的信息增益是0.276。用同样方法得到H和F的信息增益分别为0.033和0.553。因为F具有最大的信息增益，所以第一次分裂选择F为分裂属性，分裂后的结果如下图表示：



日志密度	是否使用真实头像	账号是否真实
s	no	no
m	no	no
m	no	yes
s	yes	no

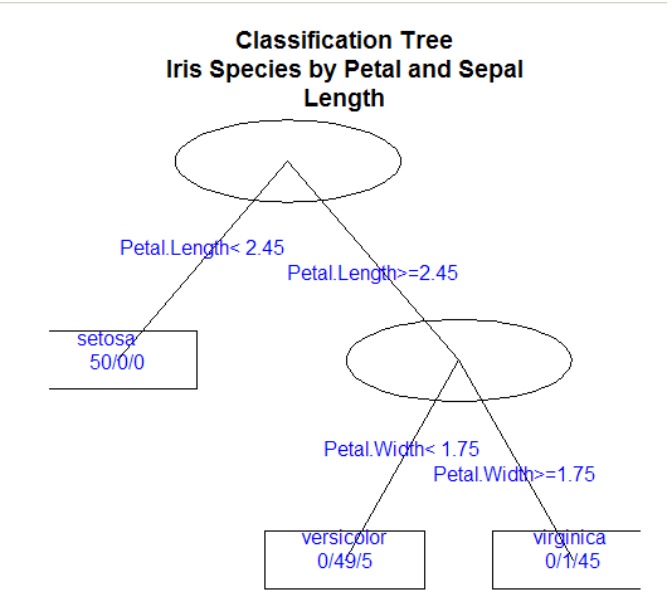
递归+分而治之

- 在上图的基础上，再递归使用这个方法计算子节点的分裂属性，最终就可以得到整个决策树。
- 这个方法称为ID3算法，还有其它的算法也可以产生决策树
- 对于特征属性为**连续值**，可以如此使用ID3算法：先将D中元素按照特征属性排序，则每两个相邻元素的中间点可以看做潜在分裂点，从第一个潜在分裂点开始，分裂D并计算两个集合的期望信息，具有最小期望信息的点称为这个属性的最佳分裂点，其信息期望作为此属性的信息期望。

R语言实现决策树：rpart扩展包

- 以鸢尾花数据集作为算例说明

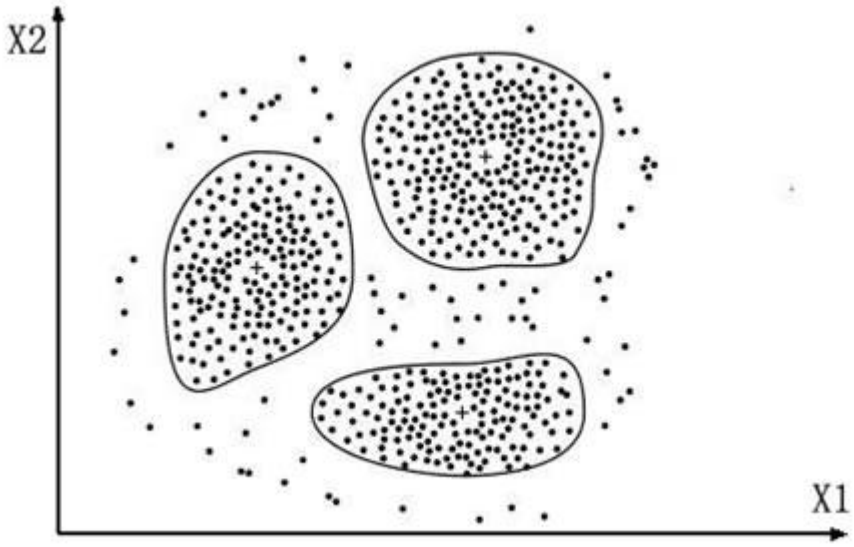
```
iris.rp = rpart(Species~., data=iris,
    method="class")
plot(iris.rp, uniform=T, branch=0,
    margin=0.1, main= “ Classification
    Tree\nIris Species by Petal and Sepal
    Length")
text(iris.rp, use.n=T, fancy=T, col="blue")
```



Rule 1: if Petal.Length>=2.45&Petal.Width<1.75, then it is versicolor(0/49/5)
Rule2: if Petal.Length>=2.45&Petal.Width>=1.75, then it is virginica (0/1/45)
Rule 3: if Petal.Length<2.45, then it is setosa (50/0/0)

聚类

聚类和分类判别有什么区别？



关键度量指标：距离

- 距离的定义
- 常用距离（薛毅书P469）

绝对值距离

欧氏距离

闵可夫斯基距离

切比雪夫距离

马氏距离

Lance和Williams距离

离散变量的距离计算

dist( )函数

x1=c(1,2,3,4,5)

x2=c(3,2,1,4,6)

x3=c(5,3,5,6,2)

x=data.frame(x1,x2,x3)

```
> dist(x,method="euclidean")
         1        2        3        4
2 2.449490
3 2.828427 2.449490
4 3.316625 4.123106 3.316625
5 5.830952 5.099020 6.164414 4.582576

> dist(x,method="minkowski")
         1        2        3        4
2 2.449490
3 2.828427 2.449490
4 3.316625 4.123106 3.316625
5 5.830952 5.099020 6.164414 4.582576
> dist(x,method="minkowski",p=5)
         1        2        3        4
2 2.024397
3 2.297397 2.024397
4 3.004922 3.143603 3.004922
5 4.323101 4.174686 5.085057 4.025455
```

dist( )函数

```
> y1=c("F","F","M","F","M")
> y2=c("A","B","B","C","A")
> y3=c(2,3,1,2,3)
> y=data.frame(y1,y2,y3)
> dist(y,method="binary")
  1 2 3 4
2 0
3 0 0
4 0 0 0
5 0 0 0 0
警告信息:
In dist(y, method = "binary") : 强制改变过程中产生了NA
> y1=c(1,0,1,1,0,0,1)
> y2=c(1,0,0,0,1,1,1)
> y3=c(1,1,1,0,0,1,1)
> y=data.frame(y1,y2,y3)
> dist(y,method="binary")
          1         2         3         4         5         6
2 0.6666667
3 0.3333333 0.5000000
4 0.6666667 1.0000000 0.5000000
5 0.6666667 1.0000000 1.0000000 1.0000000
6 0.3333333 0.5000000 0.6666667 1.0000000 0.5000000
7 0.0000000 0.6666667 0.3333333 0.6666667 0.6666667 0.3333333
```

数据中心化与标准化变换

- 目的：使到各个变量平等地发挥作用
- scale( )函数
- 极差化。 sweep( )函数（薛毅书P473）

```
> x
  x1 x2 x3
1  1  3  5
2  2  2  3
3  3  1  5
4  4  4  6
5  5  6  2
> scale(x,center=TRUE,scale=TRUE)
           x1         x2         x3
[1,] -1.2649111 -0.1039750  0.4868645
[2,] -0.6324555 -0.6238503 -0.7302967
[3,]  0.0000000 -1.1437255  0.4868645
[4,]  0.6324555  0.4159002  1.0954451
[5,]  1.2649111  1.4556507 -1.3388774
attr(,"scaled:center")
 x1  x2  x3
3.0 3.2 4.2
attr(,"scaled:scale")
      x1       x2       x3
1.581139 1.923538 1.643168
```

2013.4.28

对变量进行分类的指标：相似系数

- 距离：对样本进行分类
- 相似系数：对变量进行分类
- 常用相似系数：夹角余弦，相关系数（薛毅书P475）

（凝聚的）层次聚类法

- 思想

1 开始时，每个样本各自作为一类

2 规定某种度量作为样本之间的距离及类与类之间的距离，并计算之

3 将距离最短的两个类合并为一个新类

4 重复2-3，即不断合并最近的两个类，每次减少一个类，直至所有样本被合并为一类

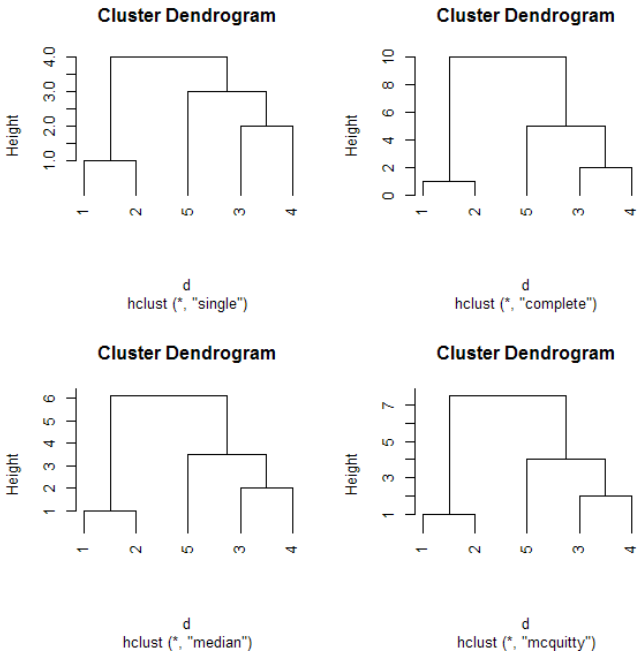
各种类与类之间距离计算的方法

- 薛毅书P476
- 最短距离法
- 最长距离法
- 中间距离法
- 类平均法
- 重心法
- 离差平方和法

hclust( )函数

- 简单的例子（薛毅书P480）

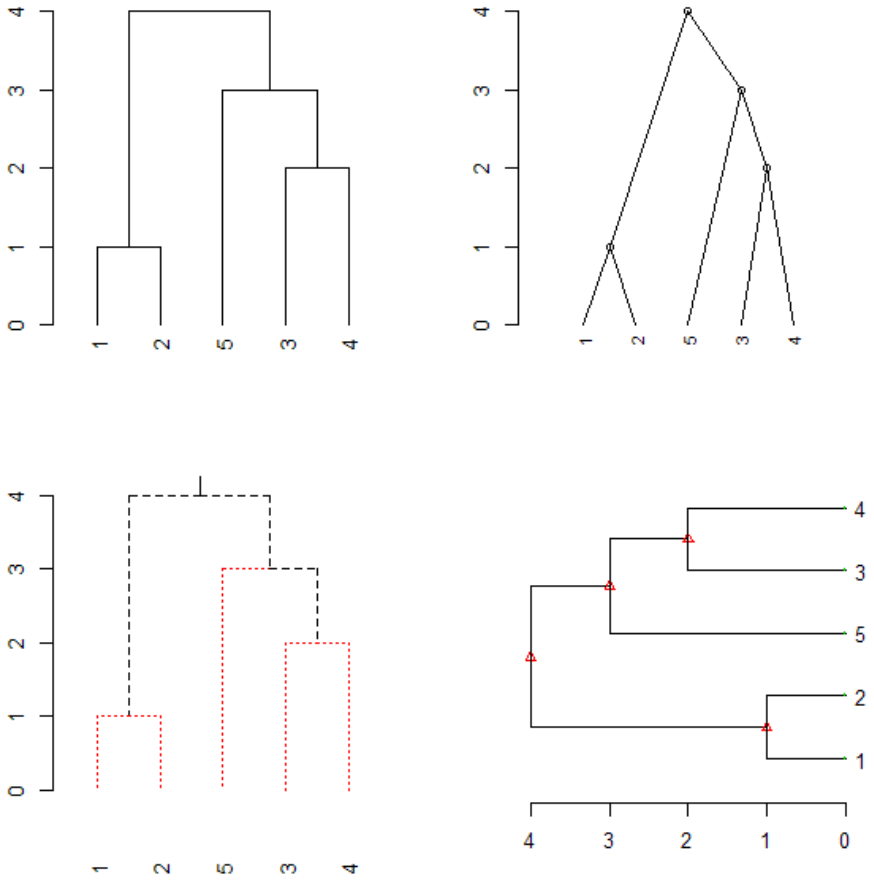
```
> x<-c(1,2,6,8,11); dim(x)<-c(5,1);
> x
     [,1]
[1,]    1
[2,]    2
[3,]    6
[4,]    8
[5,]   11
> d<-dist(x)
> d
   1  2  3  4
2  1
3  5  4
4  7  6  2
5 10  9  5  3
> hc1<-hclust(d, "single"); hc2<-hclust(d, "complete")
> hc3<-hclust(d, "median"); hc4<-hclust(d, "mcquitty")
> opar <- par(mfrow = c(2, 2))
> plot(hc1,hang=-1); plot(hc2,hang=-1)
> plot(hc3,hang=-1); plot(hc4,hang=-1)
> par(opar)
```



各种谱系图画法

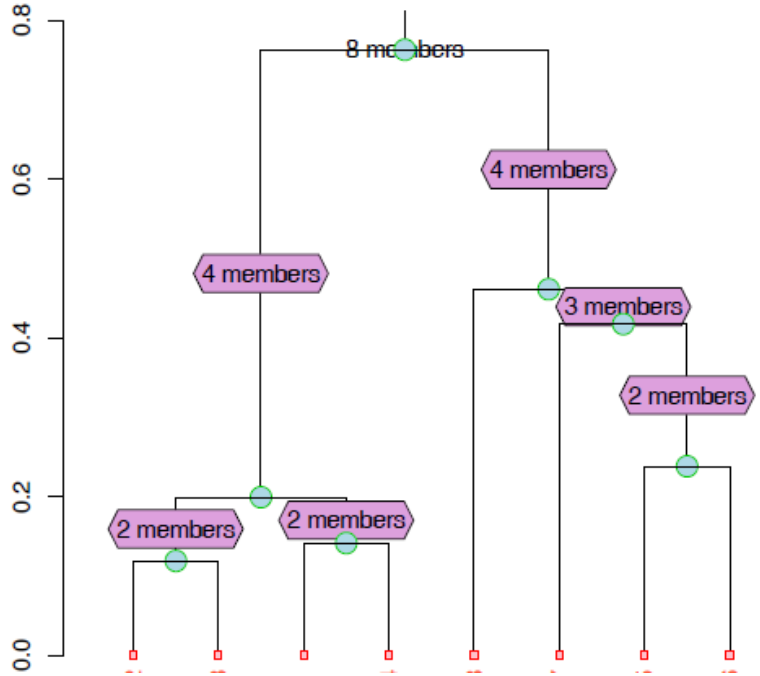
- as.dendrogram( )函数（薛毅书P482）

```
dend1<-as.dendrogram(hc1)
opar <- par(mfrow = c(2, 2),mar = c(4,3,1,2))
plot(dend1)
plot(dend1, nodePar=list(pch = c(1,NA),
      cex=0.8, lab.cex=0.8),
type = "t", center=TRUE)
plot(dend1, edgePar=list(col = 1:2, lty = 2:3),
dLeaf=1, edge.root = TRUE)
plot(dend1, nodePar=list(pch = 2:1,
      cex=.4*2:1, col=2:3),
horiz=TRUE)
par(opar)
```



对变量进行聚类分析

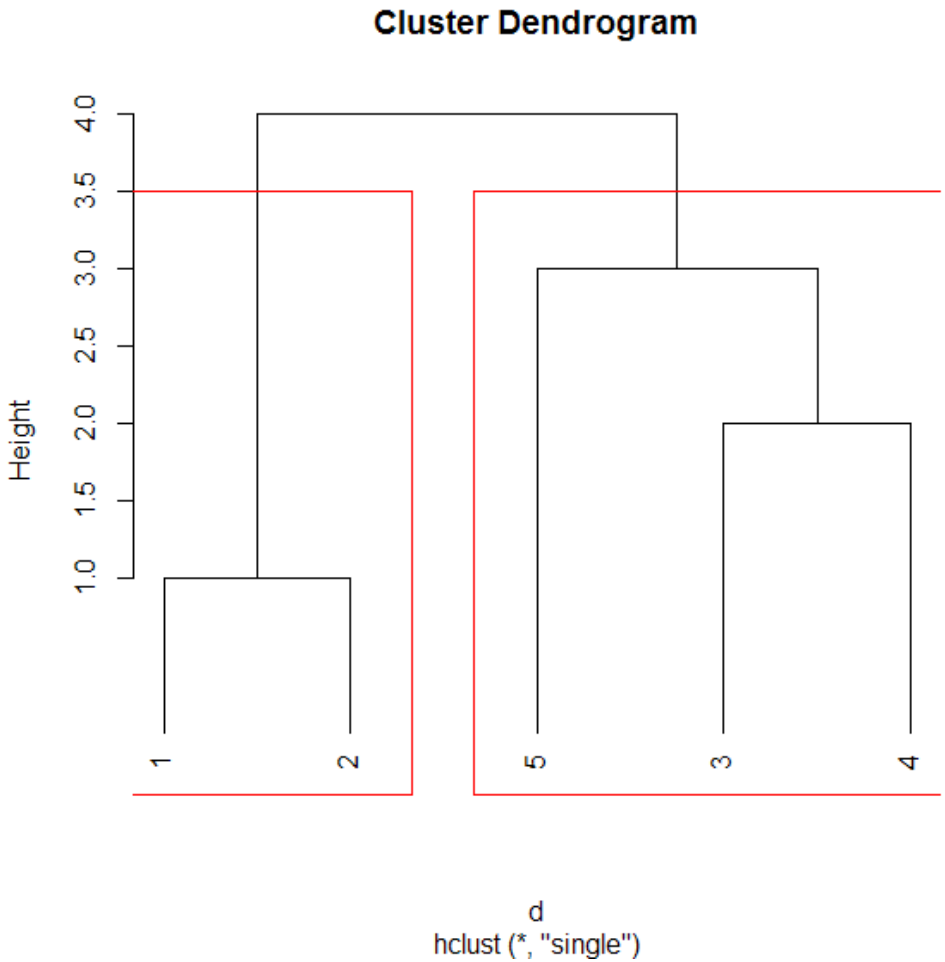
- 例子（薛毅书P483）



分多少个类？

- rect.hclust( )函数

```
> plot(hc1,hang=-1)
> rect.hclust(hc1,k=2)
```



2013.4.28

综合实例

- 薛毅书P487

补充：分类算法

- 最近邻算法Knn
- 算法主要思想：

1 选取**k个**和待分类点**距离**最近的样本点

2 看1中的样本点的分类情况，**投票**决定待分类点所属的类

动态聚类：K-means方法

- 算法：

1 选择K个点作为初始质心

2 将每个点指派到最近的质心，形成K个簇（聚类）

3 重新计算每个簇的质心

4 重复2-3直至质心不发生变化

kmeans( )函数

```
> X=iris[,1:4]
> km=kmeans(X,3)
>
>
> km
K-means clustering with 3 clusters of sizes 62, 50, 38

Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1     5.901613    2.748387     4.393548    1.433871
2     5.006000    3.428000     1.462000    0.246000
3     6.850000    3.073684     5.742105    2.071053

Clustering vector:
  [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [37] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [73] 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 3 3 3 3 1 3
[109] 3 3 3 3 3 1 1 3 3 3 3 1 3 1 3 1 3 3 1 1 3 3 3 3 3 1 3 3 3 3 1 3 3 3
[145] 3 3 1 3 3 1
```

K-means算法的优缺点

- 有效率，而且不容易受初始值选择的影响
- 不能处理非球形的簇
- 不能处理不同尺寸，不同密度的簇
- 离群值可能有较大干扰（因此要先剔除）

基于有代表性的点的技术：K中心聚类法

- 算法步骤

1 随机选择k个点作为“中心点”

2 计算剩余的点到这k个中心点的距离，每个点被分配到最近的中心点组成聚簇

3 随机选择一个非中心点O_r，用它代替某个现有的中心点O_j，计算这个代换的**总代价S**

4 如果S<0，则用O_r代替O_j，形成新的k个中心点集合

5 重复2，直至中心点集合不发生变化

K中心法的实现：PAM

- PAM使用离差平方和来计算成本S（类似于ward距离的计算）
- R语言的cluster包实现了PAM
- K中心法的优点：对于“噪音较大和存在离群值的情况，K中心法更加健壮，不像Kmeans那样容易受到极端数据影响
- K中心法的缺点：执行代价更高

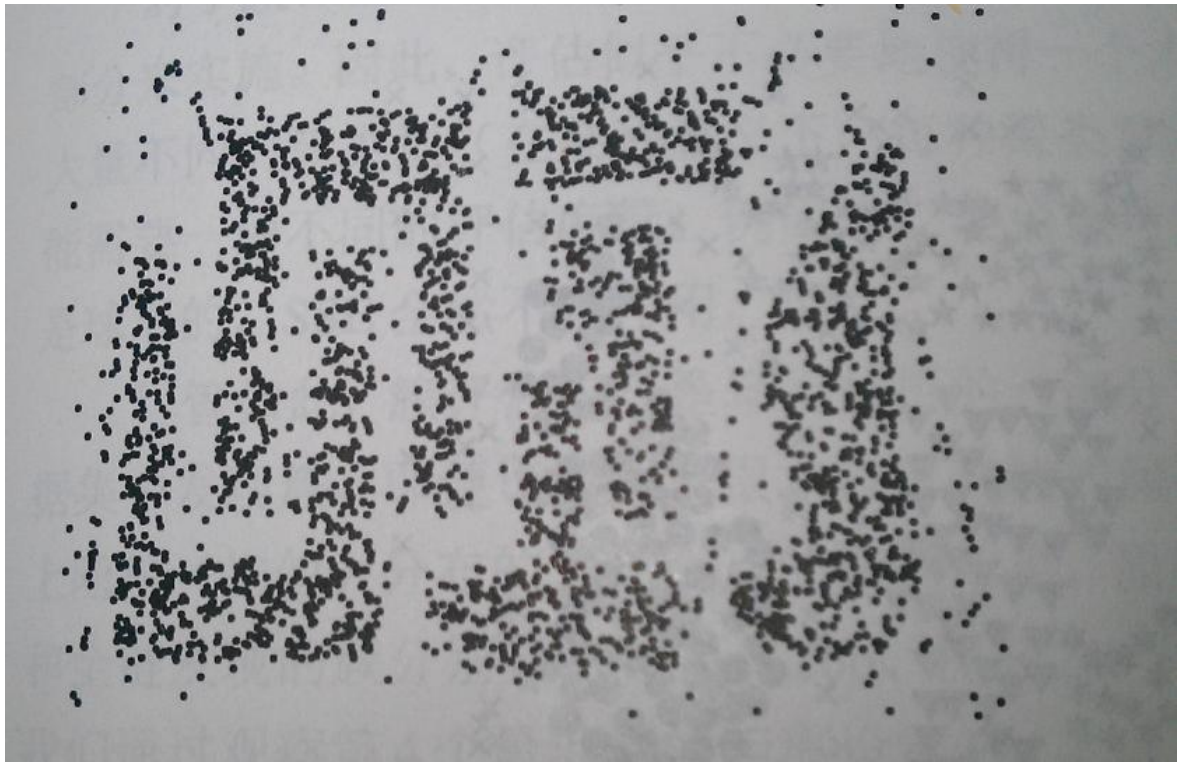
cluster包的pam( )函数

```
> x=iris[,1:4]
> kc=pam(x,3)
> kc
Medoids:
      ID Sepal.Length Sepal.Width Petal.Length Petal.Width
[1,]   8          5.0         3.4          1.5         0.2
[2,]  79          6.0         2.9          4.5         1.5
[3,] 113          6.8         3.0          5.5         2.1
Clustering vector:
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [37] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [73] 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 3 3 3 2 3
[109] 3 3 3 3 3 2 2 3 3 3 3 2 3 2 3 2 3 3 2 2 3 3 3 3 3 2 3 3 3 3 2 3 3 3 2 3
[145] 3 3 2 3 3 2
Objective function:
    build      swap
0.6709391 0.6542077

Available components:
 [1] "medoids"    "id.med"     "clustering" "objective"  "isolation"
 [6] "clusinfo"   "silinfo"    "diss"       "call"       "data"
```

基于密度的方法: DBSCAN

- DBSCAN = Density-Based Spatial Clustering of Applications with Noise
- 本算法将具有**足够高密度**的区域划分为簇，并可以发现**任何形状**的聚类



若干概念

r-邻域：给定点半径**r**内的区域

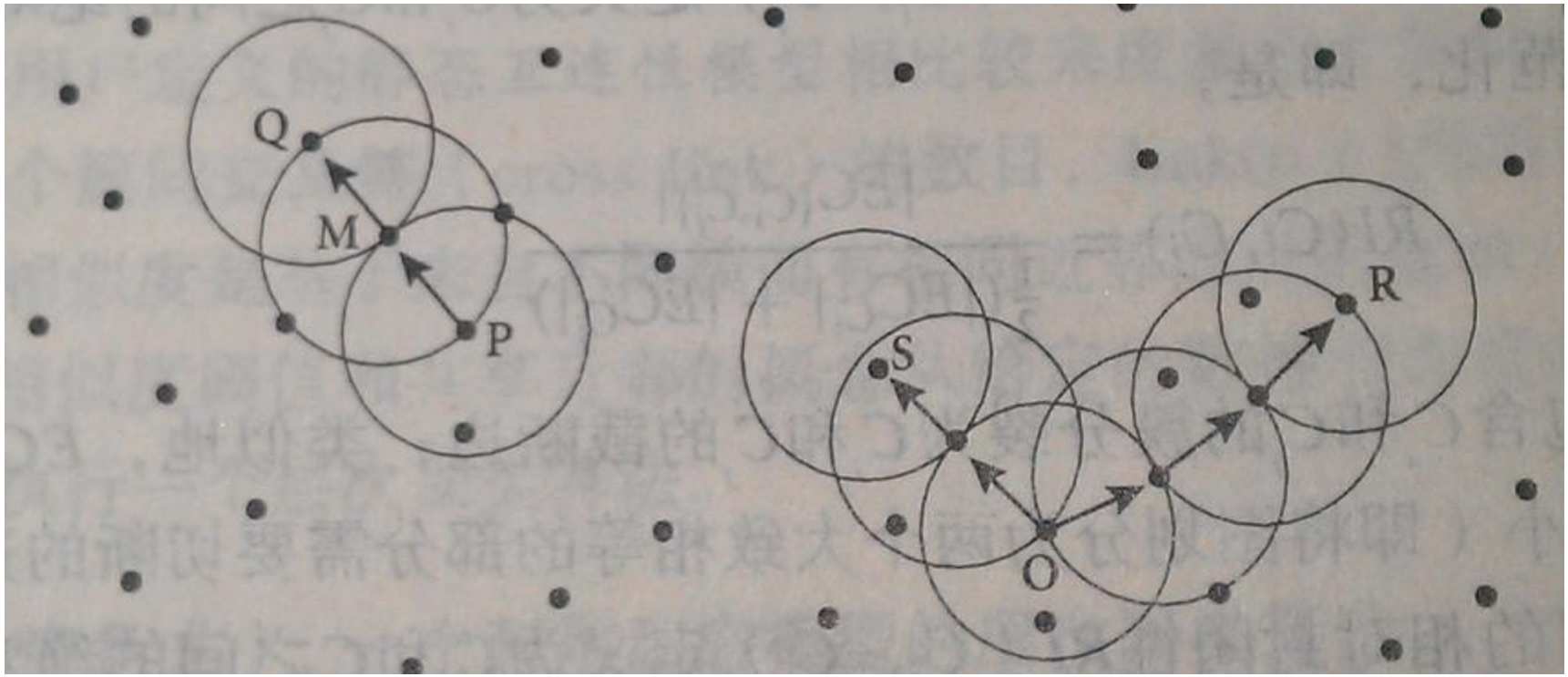
核心点：如果一个点的r-邻域至少包含最少数目**M**个点，则称该点为核心点

直接密度可达：如果点p在核心点q的r-邻域内，则称p是从q出发可以直接密度可达

如果存在点链$p_1,p_2, ..., p_n$，$p_1=q$，$p_n=p$，p_{i+1}是从p_i关于r和M直接密度可达，则称点p是从q关于r和M**密度可达**的

如果样本集D中存在点o，使得点p、q是从o关于 r和M密度可达的，那么点p、q是关于r和M**密度相连**的

DBSCAN



DBSCAN

- 算法基本 思想

1 指定合适的 r 和 M

2 计算所有的样本点，如果点p的r邻域里有超过M个点，则创建一个以p为核心点的新簇

3 反复寻找这些核心点直接密度可达（之后可能是密度可达）的点，将其加入到相应的簇，对于核心点发生“密度相连”状况的簇，给予合并

4 当没有新的点可以被添加到任何簇时，算法结束

DBSCAN算法描述

输入: 包含n个对象的数据库，半径e，最少数目MinPts;

输出:所有生成的簇，达到密度要求。

(1)Repeat

(2)从数据库中抽出一个未处理的点；

(3)IF抽出的点是核心点 THEN 找出所有从该点密度可达的对象，形成一个簇；

(4)ELSE 抽出的点是边缘点(非核心对象)，跳出本次循环，寻找下一个点；

(5)UNTIL 所有的点都被处理。

DBSCAN对用户定义的参数很敏感，细微的不同都可能导致差别很大的结果，而参数的选择无规律可循，只能靠经验确定。

孤立点检测

- 又称为异常检测，离群值检测等
- 什么是孤立点？**孤立点是一个观测值，它与其它观测值的差别如此之大，以至于怀疑它是由不同的机制产生的**
- 孤立点的一些场景

1 网站日志中的孤立点，试图入侵者

2 一群学生中的孤立点，天才 or 白痴？

3 天气数据，灾害，极端天气

4 信用卡行为，试图欺诈者

5 低概率事件，接种疫苗后却发病的

6 实验误差或仪器和操作问题造成的错误数据

等等

统计方法

- 检测一元正态分布中的离群点，指出离均值标准差数

表 10-1　均值为 0，标准差为 1 的高斯分布的样本对(c, α)，$\alpha = prob(|x| \geqslant c)$

c	$N(0,1)$的α
1.00	0.3173
1.50	0.1336
2.00	0.0455
2.50	0.0124
3.00	0.0027
3.50	0.0005
4.00	0.0001

多元正态分布的离群值

- 判断点到分布中心的距离（马氏距离，why？）

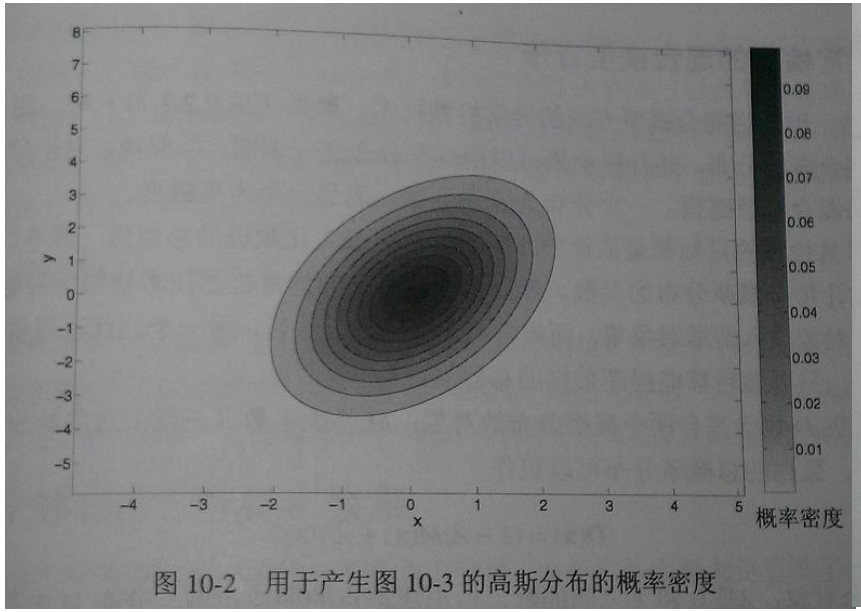
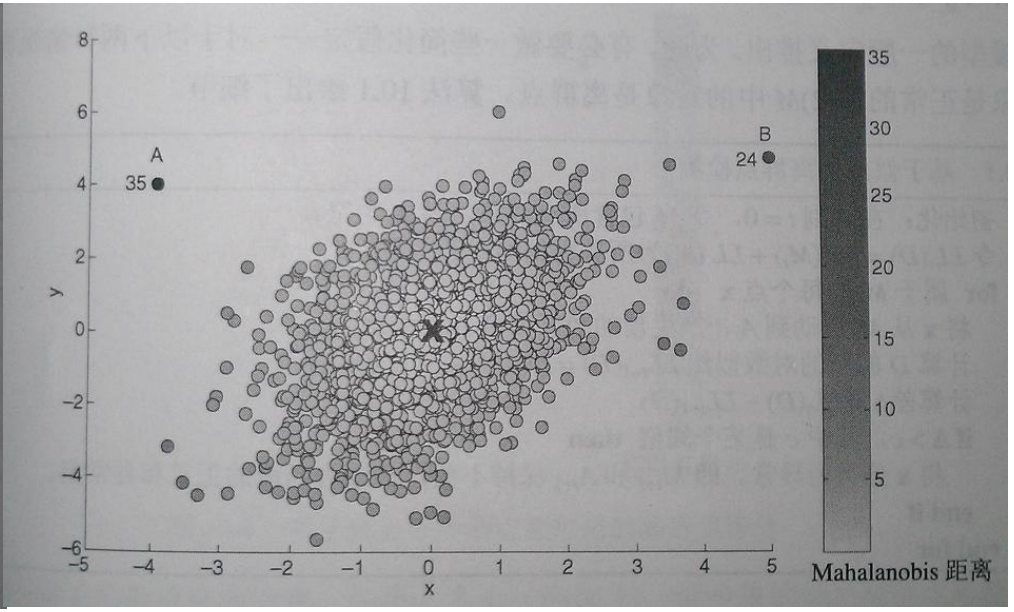


图 10-2　用于产生图 10-3 的高斯分布的概率密度



基于邻近度的孤立点检测

- 选取合适的正整数k
- 计算每个点和前k个最近邻的平均距离，得到孤立度指标
- 如果孤立度超过预定阈值，则找到孤立点

基于聚类的孤立点检测

- 首先聚类所有的点
- 对某个待测点评估它属于某一簇的程度。方法是设定一目标函数（例如kmeans法时的簇的误差平方和），如果删去此点能显著地改善此项目标函数，则可以将该点定位为孤立点

炼数成金逆向收费式网络课程

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 http://edu.dataguru.cn

Thanks

FAQ时间