信陽師範学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

第12章 MATLAB多元统计分析

讲授人：牛言涛

日期：2020年5月3日

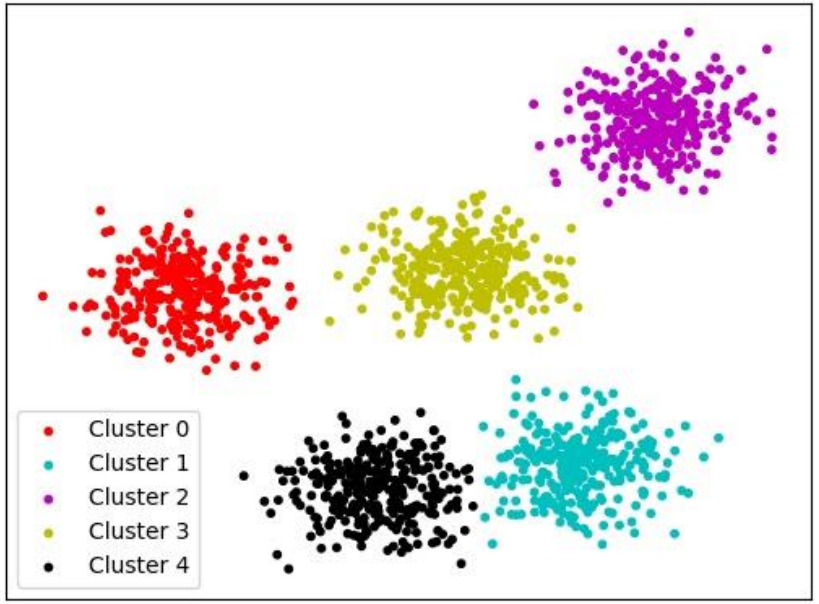
目录

CONTENTS

聚类分析简介

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- “人以类聚, 物以群分”。对事物进行分类，是人们认识事物的出发点，也是人们认识世界的一种重要方法。因此，分类学已成为人们认识世界的一门基础学科。
- 聚类（Clustering）是一种寻找数据之间内在结构的技术。聚类把全体数据实例组织成一些相似组，而这些相似组被称作簇。处于相同簇中的数据实例彼此相同，处于不同簇中的实例彼此不同。
- 聚类技术通常又被称为无监督学习，与监督学习不同的是，在簇中那些表示数据类别的分类或者分组信息是没有的。
- 数据之间的相似性是通过定义一个距离或者相似性系数来判别的。



聚类分析简介

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- 聚类分析可以应用在数据预处理过程中，对于复杂结构的多维数据可以通过聚类分析的方法对数据进行聚集，使复杂结构数据标准化。
- 聚类分析还可以用来发现数据项之间的依赖关系，从而去除或合并有密切依赖关系的数据项。聚类分析也可以为某些数据挖掘方法（如关联规则、粗糙集方法），提供预处理功能。
- 在商业上，聚类分析是细分市场的有效工具，被用来发现不同的客户群，并且它通过对不同的客户群的特征的刻画，被用于研究消费者行为，寻找新的潜在市场。
- 在生物上，聚类分析被用来对动植物和基因进行分类，以获取对种群固有结构的认识。
- 在保险行业上，聚类分析可以通过平均消费来鉴定汽车保险单持有者的分组，同时可以根据住宅类型、价值、地理位置来鉴定城市的房产分组。
- 在互联网应用上，聚类分析被用来在网上进行文档归类。
- 在电子商务上，聚类分析通过分组聚类出具有相似浏览行为的客户，并分析客户的共同特征，从而帮助电子商务企业了解自己的客户，向客户提供更合适的服务。

聚类分析简介

- 聚类分析又称群分析，它是研究（样品Q型或指标R型）分类问题的一种多元统计方法，所谓类，通俗地说，就是指相似元素的集合。
- 值得注意的是：判别分析和聚类分析是两种不同目的的分类方法，它们所起的作用是不同的。判别分析方法假定组（或类）已事先分好，判别新样品应归属哪一组，对组的事先划分有时也可以通过聚类分析得到。聚类分析方法是按样品（或变量）的数据特征，把相似的样品（或变量）倾向于分在同一类中，把不相似的样品（或变量）倾向于分在不同类中。
- 目前存在大量的聚类算法，算法的选择取决于数据的类型、聚类的目的和具体应用。聚类算法主要分为 5 大类：基于划分的聚类方法（k-平均（k-means）算法、 k-中心（k-medoids））、基于层次的聚类方法（自底向上法和自顶向下法，即凝聚式层次聚类算法和分裂式层次聚类算法）、基于密度的聚类方法、基于网格的聚类方法和基于模型的聚类方法。

1. 聚类分析的原理

聚类问题的一般提法是：设有n个样品的p元观测数据组成一个数据矩阵

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

其中每一行表示一个样品，每一列表示一个指标，x_{ij}表示第i个样品关于第j项指标的观测值，要根据观测值矩阵X对样品或指标进行分类。

聚类分析的基本思想是：在样品之间定义距离，在指标之间定义相似系数。样品距离表明样品之间的相似度，指标之间的相似系数刻画指标之间的相似度。将样品(或变量)按相似度的大小逐一归类，关系密切的聚集到较小的一类，关系疏远的聚集到较大的一类，直到所有的样品(或变量)都聚集完毕。

（1）聚类常用距离

1. 欧氏距离 $d(x_i,x_j)=[\sum_{k=1}^{p}(x_{ik}-x_{jk})^2]^{1/2}$ ，标准化欧式距离 $d_{ij}=\sqrt{\sum_{k=1}^{n}\left(\frac{x_{ik}-x_{jk}}{s_k}\right)^2}$

如果将方差的倒数看成是一个权重，则是一种加权欧氏距离(WeightedEuclidean distance)。

2. 绝对距离（曼哈顿距离，城市街区距离）$d(x_i,x_j)=\sum_{k=1}^{p}|x_{ik}-x_{jk}|$

3. 明氏距离 $d(x_i,x_j)=[\sum_{k=1}^{p}|x_{ik}-x_{jk}|^m]^{1/m}$ ，其中m(m>0)为常数，一组距离的定义。

4. 切氏距离 $d(x_i,x_j)=\max_{1\le k\le p}|x_{ik}-x_{jk}|$

5. 方差加权距离 $d(x_i,x_j)=[\sum_{k=1}^{p}(x_{ik}-x_{jk})^2/s_k^2]^{1/2}$ ，其中 $s_k^2=\frac{1}{n-1}\sum_{j=1}^{n}(x_{jk}-\bar{X}_k)^2,\bar{X}_k=\frac{1}{n}\sum_{j=1}^{n}x_{jk}.$

6. 马氏距离 $d(x_i,x_j)=(x_i-x_j)^T\Sigma^{-1}(x_i-x_j)$ ，其中 Σ 为样品的协方差矩阵。

（1）聚类常用距离

几何中夹角余弦可用来衡量两个向量方向的差异，机器学习中借用这一概念来衡量样本向量之间的差异。夹角余弦取值范围为[-1,1]。夹角余弦越大表示两个向量的夹角越小，夹角余弦越小表示两向量的夹角越大。当两个向量的方向重合时夹角余弦取最大值1，当两个向量的方向完全相反夹角余弦取最小值-1。

7. 余弦距离 $\cos(\theta) = \frac{\sum_{k=1}^{n} x_{ik} x_{jk}}{\sqrt{\sum_{k=1}^{n} x_{ik}^2} \sqrt{\sum_{k=1}^{n} x_{jk}^2}}$

8. 相关系数(Correlation coefficient) 与相关距离(Correlation distance)

$$\rho_{XY} = \frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}} = \frac{E((X-EX)(Y-EY))}{\sqrt{D(X)}\sqrt{D(Y)}}, \quad D_{XY} = 1-\rho_{XY}$$

（1）聚类常用距离

- 想象你在曼哈顿要从一个十字路口开车到另外一个十字路口，驾驶距离是两点间的直线距离吗？显然不是，除非你能穿越大楼。实际驾驶距离就是这个“曼哈顿距离”。而这也是曼哈顿距离名称的来源， 曼哈顿距离也称为城市街区距离(CityBlock distance)。
- 明氏距离不是一种距离，而是一组距离的定义。当m=1时，就是曼哈顿距离；当m=2时，就是欧氏距离；当m→∞时，就是切比雪夫距离；根据变参数的不同，明氏距离可以表示一类的距离。
- 国际象棋中国王走一步能够移动到相邻的8个方格中的任意一个。那么国王从格子$a(x_i,y_i)$走到格子$b(x_j,y_j)$最少需要多少步？自己走走试试。你会发现最少步数总是$\max(|x_j-x_i|,|y_j-y_i|)$步。有一种类似的一种距离度量方法叫切比雪夫距离。
- 明氏距离，包括曼哈顿距离、欧氏距离和切比雪夫距离都存在明显的缺点。明氏距离的缺点主要有两个：将各个分量的量纲(scale)，也就是“单位”当作相同的看待了；没有考虑各个分量的分布（期望，方差等)可能是不同的。

聚类距离选择的一种讨论

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- 距离是通过属性数据衡量对象之间相似程度的指标或者说分类统计量。通常在进行样本聚类之前会进行变量聚类，即使直接进行样本聚类，也需要对使用何种距离来进行聚类有所选择。
- 目前，聚类分析常用的距离主要有以下两种：一是加权的明氏距离，此种距离尺度通过权重来反映各指标不同的重要性对分类的影响，不足之处在于它忽视了指标变量间的交互关系；一是马氏距离，考虑到了指标变量的协方差矩阵结构（相关性）对分类的影响，但忽略了各指标变量相对重要程度的差异。
- 具体而言，在对样本进行聚类分析的过程中，样本总体的各个指标观测值的稳定性对使用不同的距离方法具有很大的影响，在使用欧式距离、明考斯基距离等时，它将样品的不同属性（即各指标或各变量）之间的差别等同看待。
- 明考斯基距离的思想就是尽量利用指标观测值之间的绝对差异，造成结果中方差大的指标权重更大一些，样本的指标观测值的方差越大，距离分析的结果意义越明确；而使用马氏距离则不同，经验表明，往往使用观测值的方差较小的指标来进行聚类效果更好。这和我们通常直观理解“选择方差大的指标分类清楚”的分类思想是有一定的抵触的。

聚类距离选择的一种讨论

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- 如果使用马氏距离，由于通过变量向量的协方差矩阵的“逆”对变量的绝对差加权，这就造成了两个结果：马氏距离的特性一方面使得它消除了量纲不同对聚类分析的影响，排除变量之间的相关性的干扰；另一方面也使得对样本测距而言，使方差小的指标变量得到了相对更大的权重，夸大了变化微小的变量的作用，尤其是R型聚类尤其明显。这个结果是不是可以接受？这个结果的产生符不符合我们的要求？

- 虽然在实践中，马氏距离得到了广泛的应用，许多学者也对它提出了各种各样的改进，比如朱惠倩在《聚类分析的一种改进方法》中提出的加权马氏距离等等。

- 目前对造成这样的变量聚类结果，他人的研究观点也是不甚明确。比如张文彤先生在《SPSS11统计分析教程》中讲到：“变量选择：在做聚类分析前，应从专业角度考虑尽量删去对分类不起作用的变量。原则上应当只引入在不同类间有显著差别的变量。变量的标准化：如果用于分析的变量其变异程度相差非常大，则变异大的变量会严重影响距离计算结果（相当于其权重大大增加），在这种情况下，需要对变量进行某种标准化，然后才能进行聚类分析。”如果需要用变量聚类的结果来进行样本聚类，而又对结果的产生原理不甚了解，不能不说这两段话是容易使人迷惑的。

聚类距离选择的一种讨论

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- 实际操作中，在Q型聚类之前先进行R型聚类，对使用各种距离进行比较的结果，是选用方差小的指标效果更好，或者说使用马氏距离，那么稳定的指标将在Q型聚类中计算距离的作用应该大一些；但是这与直观上的“选择方差大的指标分类清楚”的思想选择变量或指标有所不同。
- 笔者认为，两种距离从思想理论上并无优劣之分，它们是一体的两面。但是实践中如果数据属性难以把握，应当说采用马氏距离更为可靠。
- 另一方面，由于目前我们的许多研究学者，过于信赖对数量模型的小修小补，忽视了对变量的定性分析，正应了信息处理中的一句老话“输入进去的是垃圾，输出的也一样是垃圾”，为了更好地实现变量聚类的目的，只有在进行统计数据之前，就仔细地调查要进行聚类的对象，甄别真正有可能体现对象本质的属性来进行统计，才有可能把两种距离的思想优越性发挥出来，实现高质量的聚类分析。

王进. 聚类分析中的距离与变量选择[J]. 山西财经大学学报. 2007-04,(29):36-37.

（1）聚类常用距离

MATLAB计算距离的函数格式：Y =pdist(X,distance)

- 输入的X是一个矩阵，行为个体，列为指标，distance 是距离的类型。若缺省distance，则输出的Y是一个行向量，向量的长度为(N-1)*N/2,其中N是样本的容量，Y的元素分别为个体(1,2),(1,3),..., (1,N), (2,3),...(2,N),.....(N-1,N)之间的欧氏距离。
- 可选项distance有：'euclidean'欧氏距离；'cityblock'绝对距离；'minkowski'明氏距离(m=2)；'chebychev'切氏距离；'seuclidean'方差加权距离；'mahalanobis'马氏距离；'cosine'夹角余弦距离；'correlation'相关距离；'hamming'汉明距离；'jaccard'杰卡德距离。

案例分析

例1：2008年我国5省、区、市城镇居民人均年家庭收入如下表

省（市）	工薪收入(元/人)	经营净收入(元/人)	财产性收入(元/人)	转移性收入(元/人)
北 京	18738.96	778.36	452.75	7707.87
上 海	21791.11	1399.14	369.12	6199.77
安 徽	9302.38	959.43	293.92	3603.72
陕 西	8354.63	638.76	65.33	2610.61
新 疆	9422.22	938.15	141.75	1976.49

为了研究上述5个省、区、市的城镇居民收入差异，需要利用统计资料对其进行分类，指标变量有4个，计算各省、区、市之间的前6种距离。

案例分析


```
>> X = [18738.96 778.36 452.75 7707.87;21791.11 1399.14 369.12 6199.77;9302.38 959.43 293.92 3603.72;8354.63
638.76 65.33 2610.61;9422.22 938.15 141.75 1976.49];
>> d1 = pdist(X)  %计算出各行之间的欧氏距离
>> D = squareform(d1) % 注意此时d1必须是一个行向量，结果为实对称距离矩阵
>> S = tril(squareform(d1))  %得到下三角距离矩阵
%欧氏距离与量纲有关，有时需要对数据进行预处理，如标准化(zscore(x))等.
>> XZ = zscore(X)   %标准化
>> d2 = pdist(XZ)  %计算标准化后各行之间的欧氏距离
>> S2 = tril(squareform(d2))  %得到下三角距离矩阵
>> d3 = pdist(X,'cityblock')  %计算绝对距离
>> d4 = pdist(X,'minkowski',3) %计算明氏距离,d4为1行10列的行向量
>> d5 = pdist(X,'chebychev')  %计算切氏距离.
>> d6 = pdist(X,'seuclidean')  %计算方差加权距离.
>> d7 = pdist(X,'mahalanobis')  %计算马氏距离
```

（2）聚类分析对变量分类

聚类分析方法不仅可以对样品进行分类，而且可以对变量进行分类，在对变量进行分类时，常常采用相似系数来度量变量之间的相似性。对p个指标变量进行聚类时，用相似系数来衡量变量之间的相似程度（关联度），若用$C_{\alpha\beta}$表示变量α,β之间的相似系数，则应满足：

(1) $|C_{\alpha\beta}| \le 1$且$C_{\alpha\alpha}=1$；

(2) $C_{\alpha\beta}=\pm 1$ 当且仅当 $\alpha=k\beta,\ k\neq 0$；

(3) $C_{\alpha\beta}=C_{\beta\alpha}$

相似系数中最常用的是相关系数与夹角余弦。

```
>> R=corrcoef(X) %指标之间的相关系数
   1.0000    0.6183    0.8138    0.8931
   0.6183    1.0000    0.4287    0.2927
   0.8138    0.4287    1.0000    0.9235
   0.8931    0.2927    0.9235    1.0000
>> Xn = normc(X)      % 将x的各列化为单位向量
>> J = Xn'*Xn  %计算夹角余弦
   1.0000    0.9536    0.9609    0.9797
   0.9536    1.0000    0.9026    0.8990
   0.9609    0.9026    1.0000    0.9833
   0.9797    0.8990    0.9833    1.0000
```

（3）类间距离与递推公式

设d_{ij}表示两个样品x_i, x_j之间的距离，G_p, G_q分别表示两个类别，各自含有n_p, n_q个样品。

(1) 最短距离 $D_{pq} = \min_{i \in G_p, j \in G_q} d_{ij}$，即用两类中样品之间的距离最短者作为两类间距离。

(2) 最长距离 $D_{pq} = \max_{i \in G_p, j \in G_q} d_{ij}$，即用两类中样品之间的距离最长者作为两类间距离。

(3) 类平均距离 $D_{pq} = \frac{1}{n_p n_q} \sum_{i \in G_p} \sum_{j \in G_q} d_{ij}$

即用两类中所有两两样品之间距离的平均作为两类间距离。

(4) 重心距离 $D_{pq} = d\left(\bar{x}_p, \bar{x}_q\right) = \sqrt{\left(\bar{x}_p - \bar{x}_q\right)^T \left(\bar{x}_p - \bar{x}_q\right)}$

其中$\bar{x}_p, \bar{x}_q$分别是G_p, G_q的重心，这是用两类重心之间的欧氏距离作为类间距离。

（3）类间距离与递推公式

（5）离差平方和距离（ward）$D_{pq}^2 = \frac{n_p n_q}{n_p + n_q}(\bar{x}_p - \bar{x}_q)^T(\bar{x}_p - \bar{x}_q)$

显然，离差平方和距离与重心距离的平方成正比.

设有两类G_p, G_q合并成新的一类G_r，包含了$n_r = n_p + n_q$个样品，如何计算G_r与其他类别$G_k(k \neq p, q)$之间的距离，这就需要建立类间距离的递推公式。

最短距离	$D_{rk} = \min(D_{pk}, D_{qk})$	类平均距离	$D_{rk} = \frac{n_p}{n_r} D_{pk} + \frac{n_q}{n_r} D_{qk}$
最长距离	$D_{rk} = \max(D_{pk}, D_{qk})$	重心距离	$D_{rk}^2 = \frac{n_p}{n_r} D_{pk}^2 + \frac{n_q}{n_r} D_{qk}^2 - \frac{n_p}{n_r}\frac{n_q}{n_r} D_{pq}^2$
离差平方和距离	$D_{rk}^2 = \frac{n_p + n_k}{n_r + n_k} D_{pk}^2 + \frac{n_q + n_k}{n_r + n_k} D_{qk}^2 - \frac{n_k}{n_r + n_k} D_{pq}^2$		

2. 谱系聚类

- 谱系聚类（又称层次聚类、系统聚类）法是目前应用较为广泛的一种聚类法。谱系聚类是根据生物分类学的思想对研究对象进行分类的方法。
- 在生物分类学中，分类的单位是：门、纲、目、科、属、种，其中种是分类的基本单位，分类单位越小，它所包含的生物就越少，生物之间的共同特征就越多。
- 利用这种思想，谱系聚类首先将各样品自成一类，然后把最相似(距离最近或相似系数最大)的样品聚为小类，再将已聚合的小类按各类之间的相似性(用类间距离度量)进行再聚合。随着相似性的减弱，最后将一切子类都聚为一大类，从而得到一个按相似性大小聚结起来的一个谱系图。

2. 谱系聚类

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

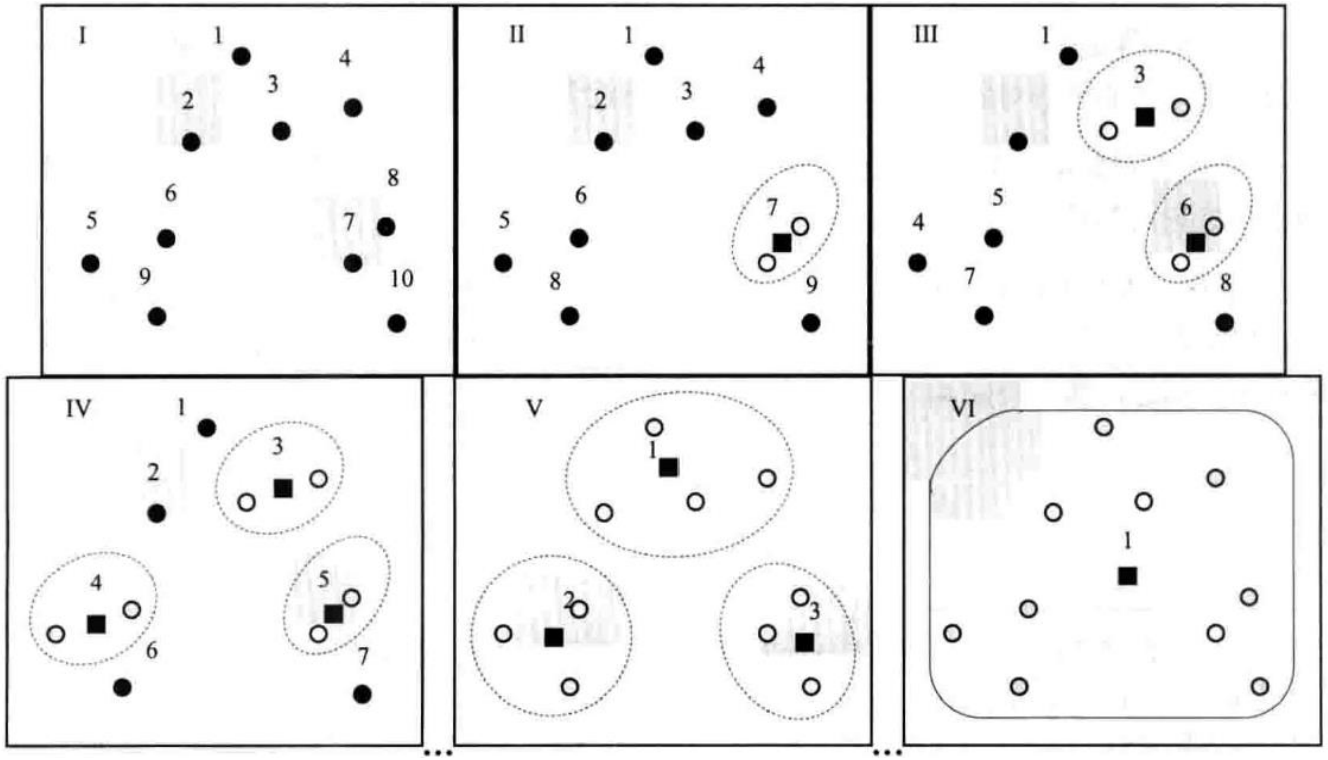
谱系聚类的步骤如下：

（1）n个样品开始作为n个类，计算两两之间的距离或相似系数，得到实对称矩阵；

（2）从D_0的非主对角线上找最小(距离)或最大元素(相似系数)，设该元素是D_{pq}，则将G_p, G_q合并成新的一类G_r，在D_0中去掉G_p, G_q所在的两行、两列，并加上新类G_r与其余各类之间的距离或相似系数，得到$n-1$阶矩阵D_1；

（3）从D_1出发重复步骤(2)的做法得到D_2，再由D_2出发重复上述步骤，直到两个样品聚为一个大类为止；

（4）在合并过程中要记下合并样品的编号及两类合并时的水平，并绘制聚类谱系图。



谱系聚类的MATLAB实现

信阳师范学院
数学与统计学院

层次凝聚聚类算法，格式：Z = linkage (Y,method)

- 输入Y是一个距离矩阵，如Y是由pdist命令生成的欧氏距离向量。
- Method 是可选项：'single' ---- 最短距离（缺省状态）、'complete'----最长距离、'average' ----类平均距离、'weighted'----加权平均距离、'centroid'----重心距离、'ward' ----离差平方和距离；
- 输出Z是一个矩阵(N-1行，3列)，Z的第一列和第二列均为正整数，第3列表示聚类的水平，每一行表示在相同的聚类水平上将个体合并成新的一类，每生成一个新的类，其编号将在现有基础上增加1。

谱系聚类的MATLAB实现

`dendrogram(tree)` generates a dendrogram plot of the hierarchical binary cluster tree. A dendrogram consists of many U-shaped lines that connect data points in a hierarchical tree. The height of each U represents the distance between the two data points being connected. tree由linkage命令生成

- If there are 30 or fewer data points in the original data set, then each leaf in the dendrogram corresponds to one data point.
- If there are more than 30 data points, then `dendrogram` collapses lower branches so that there are 30 leaf nodes. As a result, some leaves in the plot correspond to more than one data point.

`dendrogram(tree,Name,Value)` uses additional options specified by one or more name-value pair arguments.

`dendrogram(tree,P)` generates a dendrogram plot with no more than `P` leaf nodes. If there are more than `P` data points in the original data set, then `dendrogram` collapses the lower branches of the tree. As a result, some leaves in the plot correspond to more than one data point. 若显示所有节点，P设置为0

`dendrogram(tree,P,Name,Value)` uses additional options specified by one or more name-value pair arguments.

`H = dendrogram( ___ )` generates a dendrogram plot and returns a vector of line handles. You can use any of the input arguments from the previous syntaxes.

`[H,T,outperm] = dendrogram( ___ )` also returns a vector containing the leaf node number for each object in the original data set, `T`, and a vector giving the order of the node labels of the leaves as shown in the dendrogram, `outperm`.

- It is useful to return `T` when the number of leaf nodes, `P`, is less than the total number of data points, so that some leaf nodes in the display correspond to multiple data points.
- The order of the node labels given in `outperm` is from left to right for a horizontal dendrogram, and from bottom to top for a vertical dendrogram.

谱系聚类的MATLAB实现

信阳师范学院
数学与统计学院

作谱系聚类图，格式H=dendrogram(z,N)

- 输入Z是一个(N-1)行3列的矩阵，由linkage命令生成，N是样本容量。输出产生一个树谱系聚类图，每两类通过线段连接，高度表示类间的距离。% 此命令作出m个样本的图形，缺省时默认为30，若显示所有节点，设置为0。

输出聚类结果，格式T=cluster(z,k)

- 输入Z是一个(N-1)行3列的矩阵，由linkage命令生成，N是样本容量。k是分类数目；
- 输出T是一个列向量(N行1列)，每一个元素均为正整数，且最大的数字不超过k，第i行的数字l表示第i个个体属于第l类。如果遇到大样本数据，为了便于得到每一类样本的编号，可以利用如下命令：find(T==l) % 找出属于第l类的样品编号。

谱系聚类的MATLAB实现

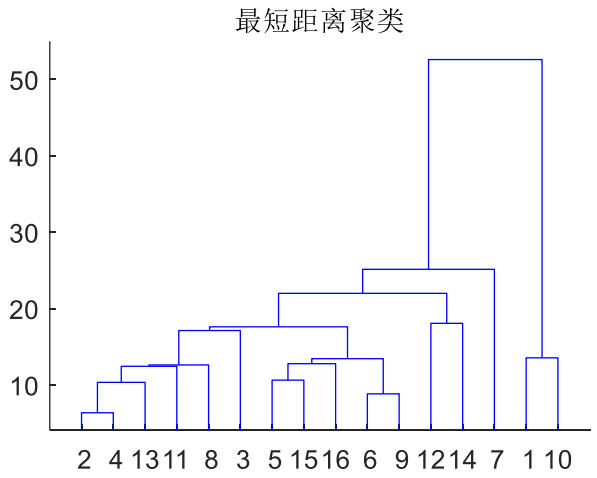
信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

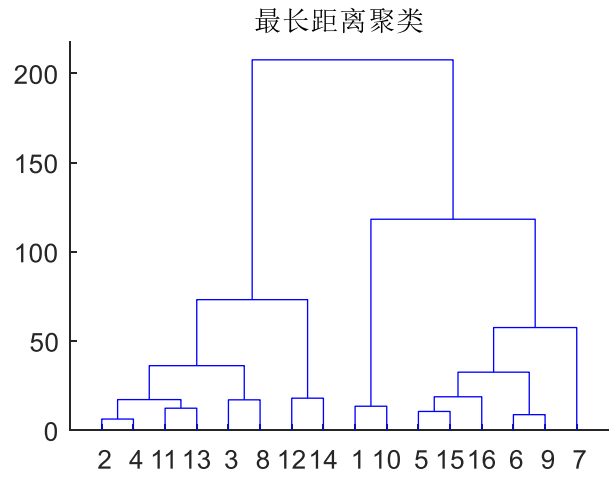
例2：帮助企业对市场上的产品进行分类，从而更准确地指定营销策略。如，某饮料企业收集了市场上16种饮料的热量、咖啡因、钠含量和价格4种变量数据。

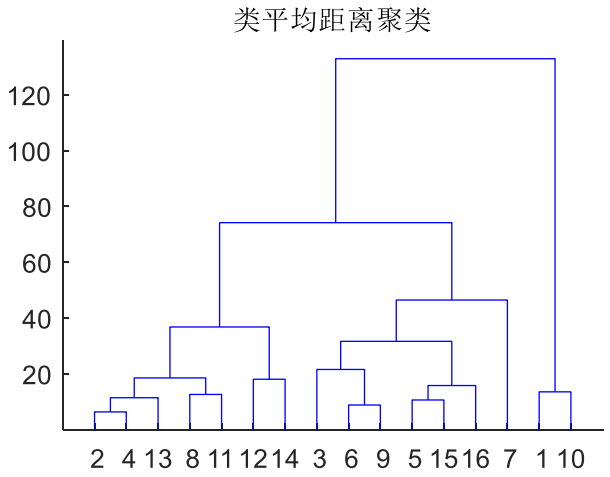
index	heat	caffeine	Na	price	index	heat	caffeine	Na	price
1	207.2	3.3	15.5	2.8	9	95.9	0	8.5	1.3
2	36.8	5.9	12.9	3.3	10	199	0	10.6	3.5
3	72.2	7.3	8.2	2.4	11	49.8	8	6.3	3.7
4	36.7	0.4	10.5	4	12	16.6	4.7	6.3	1.5
5	121.7	4.1	9.2	3.5	13	38.5	3.7	7.7	2
6	89.1	4	10.2	3.3	14	0	4.2	13.1	2.2
7	146.7	4.3	9.7	1.8	15	118.8	4.7	7.2	4.1
8	57.6	2.2	13.6	2.1	16	107	0	8.3	4.2

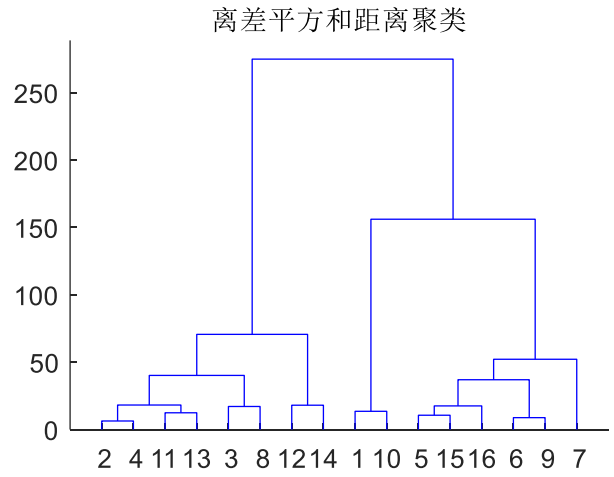
谱系聚类的MATLAB实现

```
drink = xlsread('drink.xlsx');
d = pdist(drink);  %欧式距离
z1= linkage(d);   %默认最短距离
subplot(2,2,1)
H= dendrogram(z1);  %谱系聚类图
subplot(2,2,2)
z2= linkage(d,'complete');  %使用最长距离
Hc= dendrogram(z2);  %谱系聚类图
subplot(2,2,3)
z3= linkage(d,'average');   %类平均距离
Ha= dendrogram(z3);  %谱系聚类图
subplot(2,2,4)
z4= linkage(d,'ward');   %离差平方和距离
Hw= dendrogram(z4);  %谱系聚类图
title('离差平方和距离聚类')
```



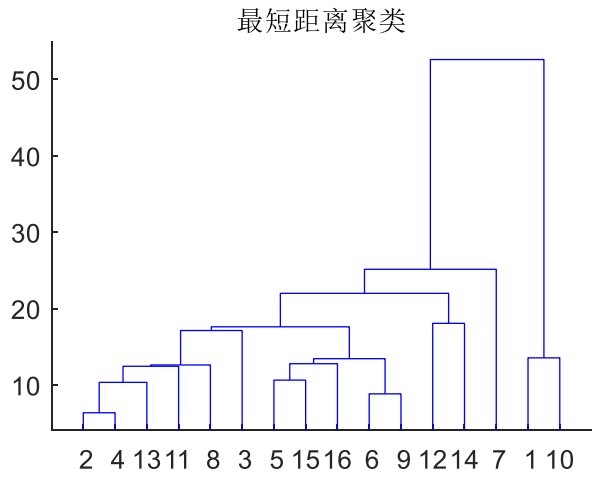


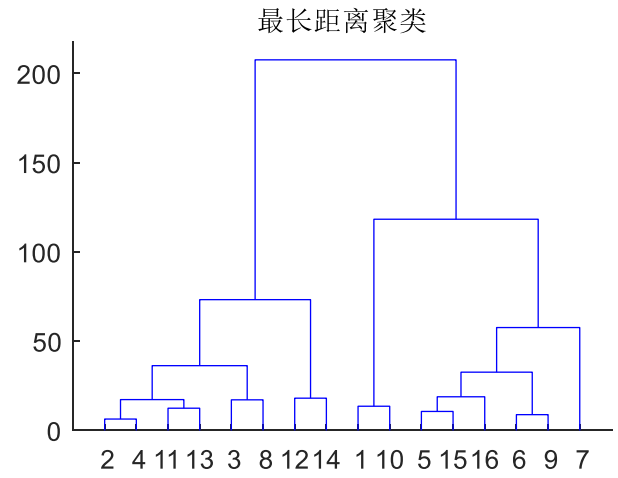


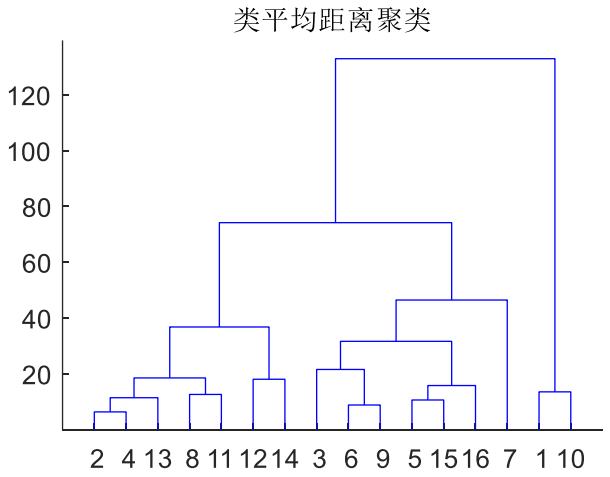


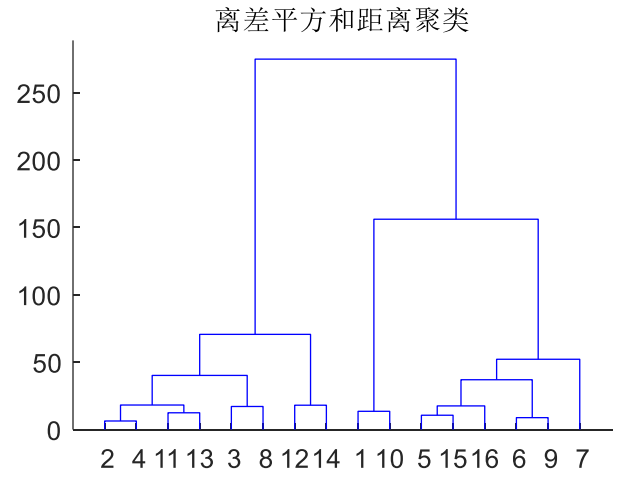
谱系聚类的MATLAB实现

- 四种方法由于类与类之间的距离计算方法不同，所得到的结果略有出入，其中最短距离法与其余三种差别最大。以离差平方和为例，编号为12、14、2、4、13、3、8、11的饮料归为第一大类，其余在第二类。
- 根据图形效果，把16种饮料细分为4类，通过函数cluster(z4,4)可以将数据分组，以序号1、2、3、4分别表示4个类别：12,14为一类；2,3,4,8,11,13为一类；1,10为一类；5,6,7,9,15,16为一类。









谱系聚类的MATLAB实现

以上是样品之间是欧氏距离，类间距离是最短距离、最长距离、类平均距离和离差平方和距离聚类的结果，那么哪一种最好呢？为此可以计算复合相关系数，若该系数越接近于1则该聚类越理想。在MATLAB中计算复合相关系数的命令：R=cophenet(z,d)，其中z 是用某种类间距离linkage后的结果，d是样品之间的某种距离。

```
>> R=[cophenet(z1,d),cophenet(z2,d),cophenet(z3,d),cophenet(z4,d)]
R =
    0.7679    0.6636    0.8190    0.6642
```

由于0.8190 最大，故认为若样品之间采用欧氏距离，则类间距离以类平均距离最好，如果要找到最理想的分类方法，可以对每一种样品之间的距离，都计算上述的复合相关系数，这样就可以找到最理想的样品距离与对应的类间距离。

谱系聚类的MATLAB实现

```
drink = xlsread('drink.xlsx');
drinkzs = zscore(drink);
distance = {'euclidean','mahalanobis','cityblock','seuclidean','chebychev','hamming'};
classdist = {'single','complete','average','ward','centroid','weighted'};
R = zeros(6);
for i = 1:6
    for j = 1:6
        d = pdist(drinkzs,distance{i});  %欧式距离
        z= linkage(d,classdist{j});   %默认最短距离
        R(i,j) = cophenet(z,d);
    end
end
```

```
>> R
R =
    0.7679    0.6636    0.8190    0.6642    0.8190    0.6334
    0.6206    0.6023    0.6821    0.5340    0.6481    0.6755
    0.7733    0.6619    0.8222    0.6073    0.8284    0.6347
    0.6544    0.6705    0.7039    0.5369    0.6429    0.7012
    0.7721    0.6470    0.8207    0.6676    0.8207    0.6383
    0.8255    0.8378    0.9072    0.6052    0.0574    0.9055
```

注意：不要忽略警告信息，如ward、centroid需要指定为欧氏距离，非单调聚类树——centroid可能不合适。该例显示选择hamming和average最优。但仍需要聚类验证，进而选择是否最优。

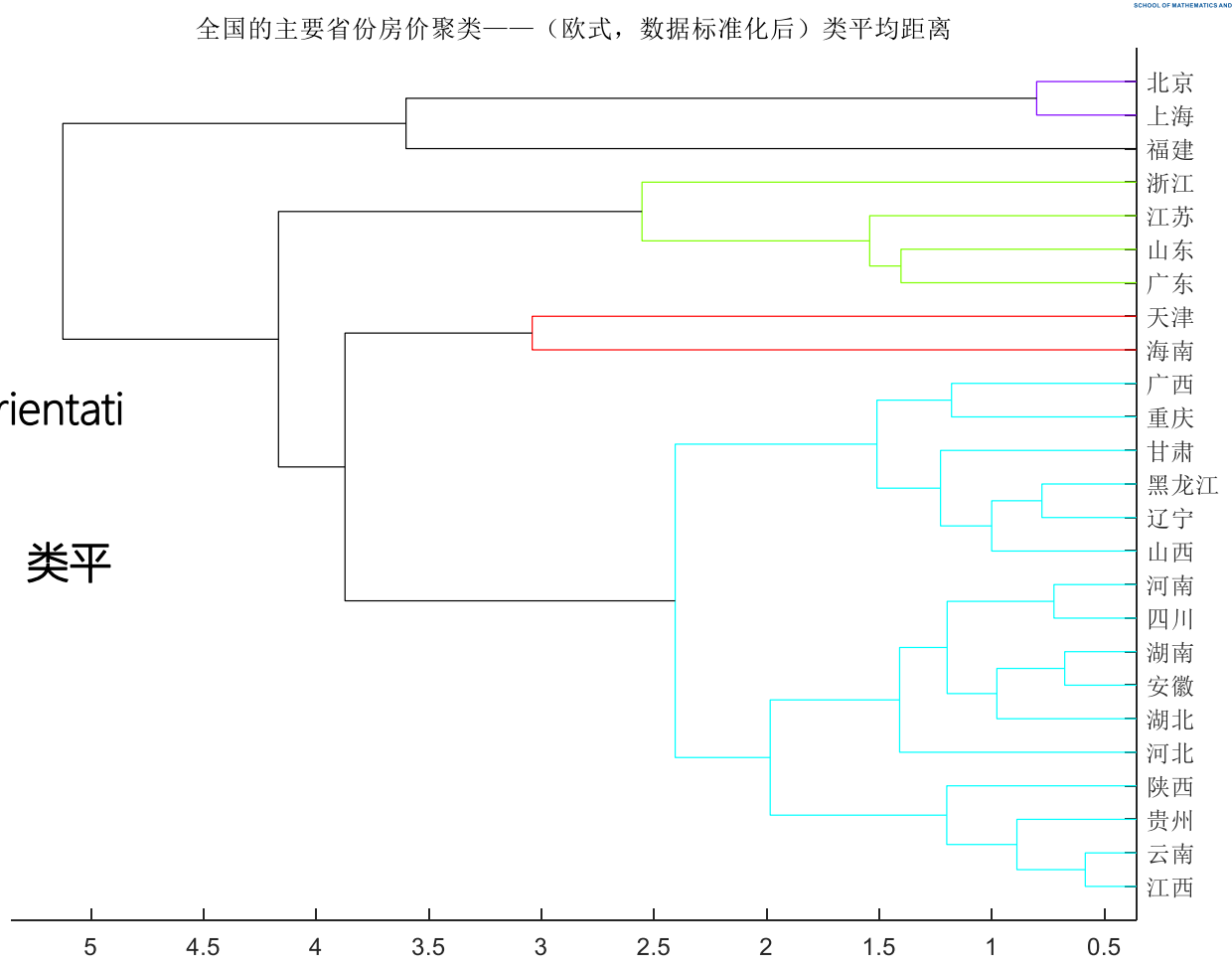
谱系聚类的MATLAB实现

例3：对全国的主要城市进行分类，从而帮助政府有针对性地对不同城市制定政策。以近年来最热门话题——房价为例，运用聚类分析根据房价及其他一些经济指标，对城市进行分类，了解不同类别城市的特点。本例选取25个省份的7项经济指标。部分数据如下表所示：

2018年上半年各省情况统计							
省份	GDP（万亿元）	人口（万人）	人均GDP（万元）	GDP增长率（%）	平均工资水平	省会城市房价均价	房价/工资
福建	1.484	3911	3.79	8.2	7215	45579	6.32
上海	1.556	2418.33	6.43	6.9	9796	55472	5.66
北京	1.405	2170.7	6.47	6.8	10531	54648	5.19
海南	0.243	925.76	2.62	5.8	7349	29614	4.03
天津	0.998	1556.87	6.41	3.4	6765	25610	3.79
广东	4.63	11169	4.15	7.1	8019	33981	4.24
山东	3.966	10005.83	3.96	6.6	7014	26407	3.76
江苏	4.486	8029.3	5.59	7	7660	29279	3.82

谱系聚类的MATLAB实现

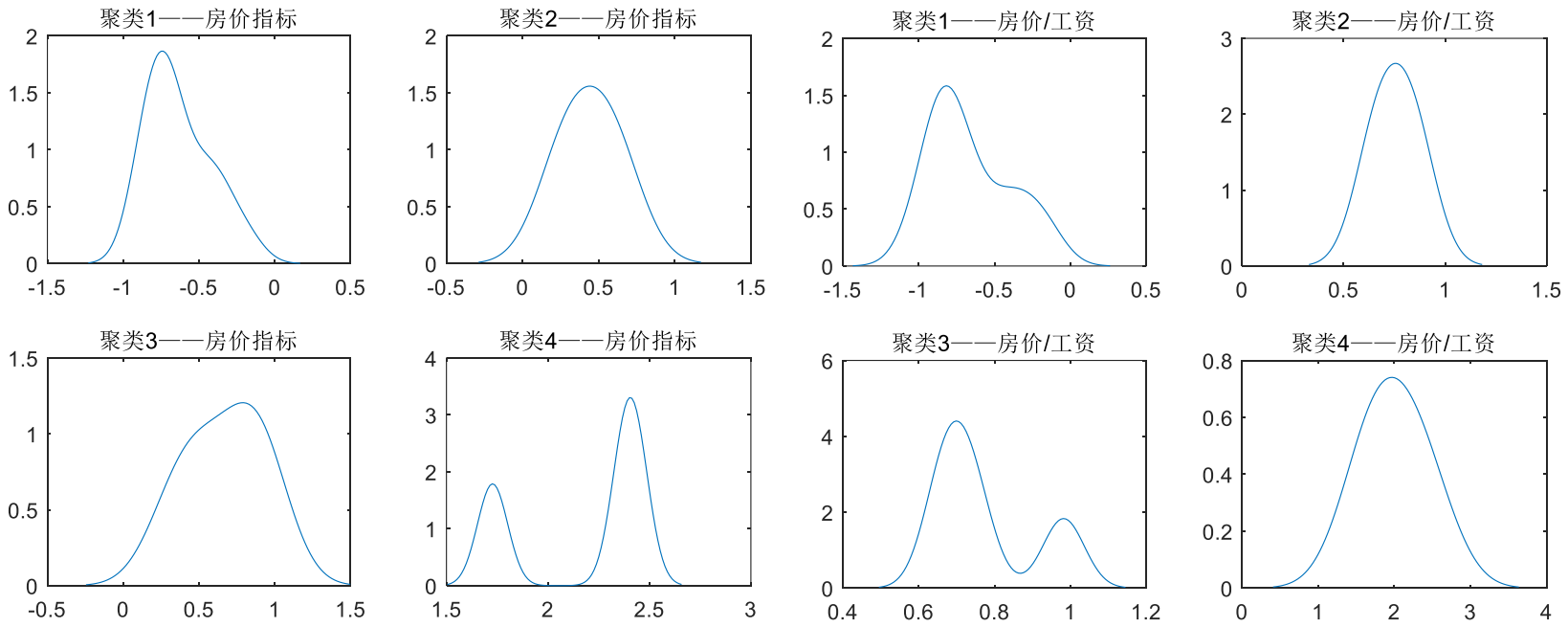
```
[sale,text] = xlsread('sale_houseprice.xlsx');
sale = zscore(sale);
d = pdist(sale);  %欧式距离
z= linkage(d,'average');   %类平均距离
labels = text(3:end,1);
dendrogram(z,'ColorThreshold','default','Labels',labels,'Orientati
on','left');  %谱系聚类图
title('全国的主要省份房价聚类——（欧式，数据标准化后）类平
均距离')
T=cluster(z,4);   %输出4类聚类结果
CA = cell(25,2);  %用元胞数组组合
CA(:,1) = labels;   CA(:,2) = num2cell(T);  %类别
CA = sortrows(CA,2)   %按类别扩展排序
```



明考斯基距离的思想就是尽量利用指标观测值之间的绝对差异，造成结果中方差大的指标权重更大一些，样本的指标观测值的方差越大，距离分析的结果意义越明确；

谱系聚类的MATLAB实现

```
s1 = sale(T == 1,:); s2 = sale(T == 2,:);
s3 = sale(T == 3,:); s4 = sale(T == 4,:);
subplot(2,2,1)
[f,xi]=ksdensity(s1(:,6));
plot(xi,f); title('聚类1——房价指标')
subplot(2,2,2)
[f,xi]=ksdensity(s2(:,6));
plot(xi,f); title('聚类2——房价指标')
subplot(2,2,3)
[f,xi]=ksdensity(s3(:,6));
plot(xi,f); title('聚类3——房价指标')
subplot(2,2,4)
[f,xi]=ksdensity(s4(:,6));
plot(xi,f); title('聚类4——房价指标')
```

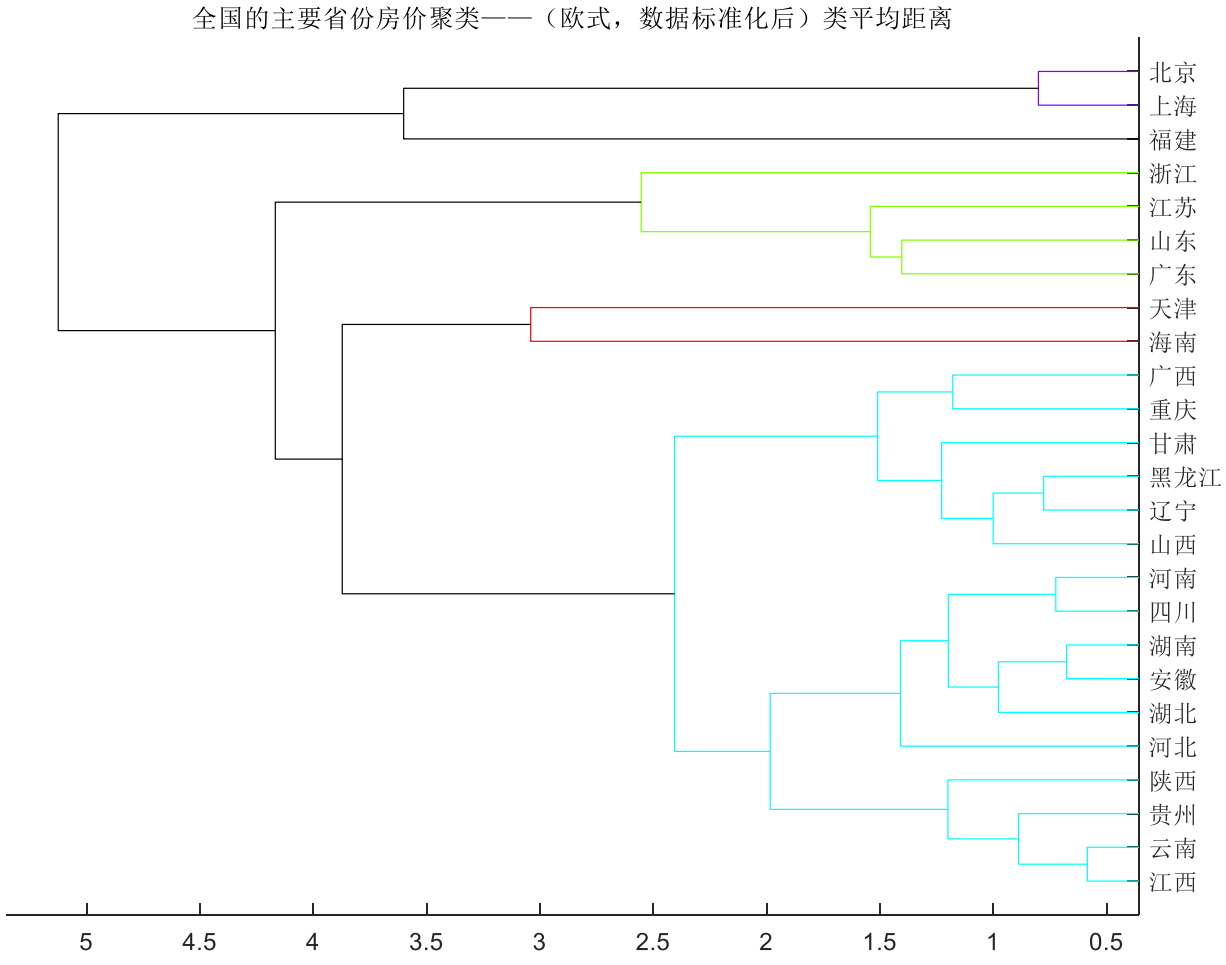


从4个聚类结果，分别绘制房价指标和房价/工资指标的核密度估计曲线图，从曲线上仍能分析出其区别，如聚类1房价范围在[-0.8,-0.2]左右，聚类2在[0.2,0.8]左右，聚类3在[0.3,1.2]左右，聚类4存在双峰，且与其他类别区别较大。此外，由于样本较少，核密度估计可能存在不准确。

谱系聚类的MATLAB实现

```
CA =
  25×2 cell 数组
    '河北'      [1]
    '湖北'      [1]
    '四川'      [1]
    '河南'      [1]
    '安徽'      [1]
    '陕西'      [1]
    '山西'      [1]
    '辽宁'      [1]
    '江西'      [1]
    '黑龙江'    [1]
    '甘肃'      [1]
    '重庆'      [1]
    '云南'      [1]
    '湖南'      [1]
    '广西'      [1]
    '贵州'      [1]
    '海南'      [2]
    '天津'      [2]
    '广东'      [3]
    '山东'      [3]
    '江苏'      [3]
    '浙江'      [3]
    '福建'      [4]
    '上海'      [4]
    '北京'      [4]
```

第一组	第二组	第三组	第四组		
福建	广东	海南	子小组	子小组	
上海	山东	天津	广西	河南	河北
北京	江苏		重庆	四川	陕西
	浙江		甘肃	湖南	贵州
			黑龙江	安徽	云南
			辽宁	湖北	江西
			山西		



谱系聚类的MATLAB实现

d = pdist(sale,'mahalanobis'); %马氏距离

T=cluster(z,6); %输出6类聚类结果

CA = cell(25,2); %用元胞数组组合

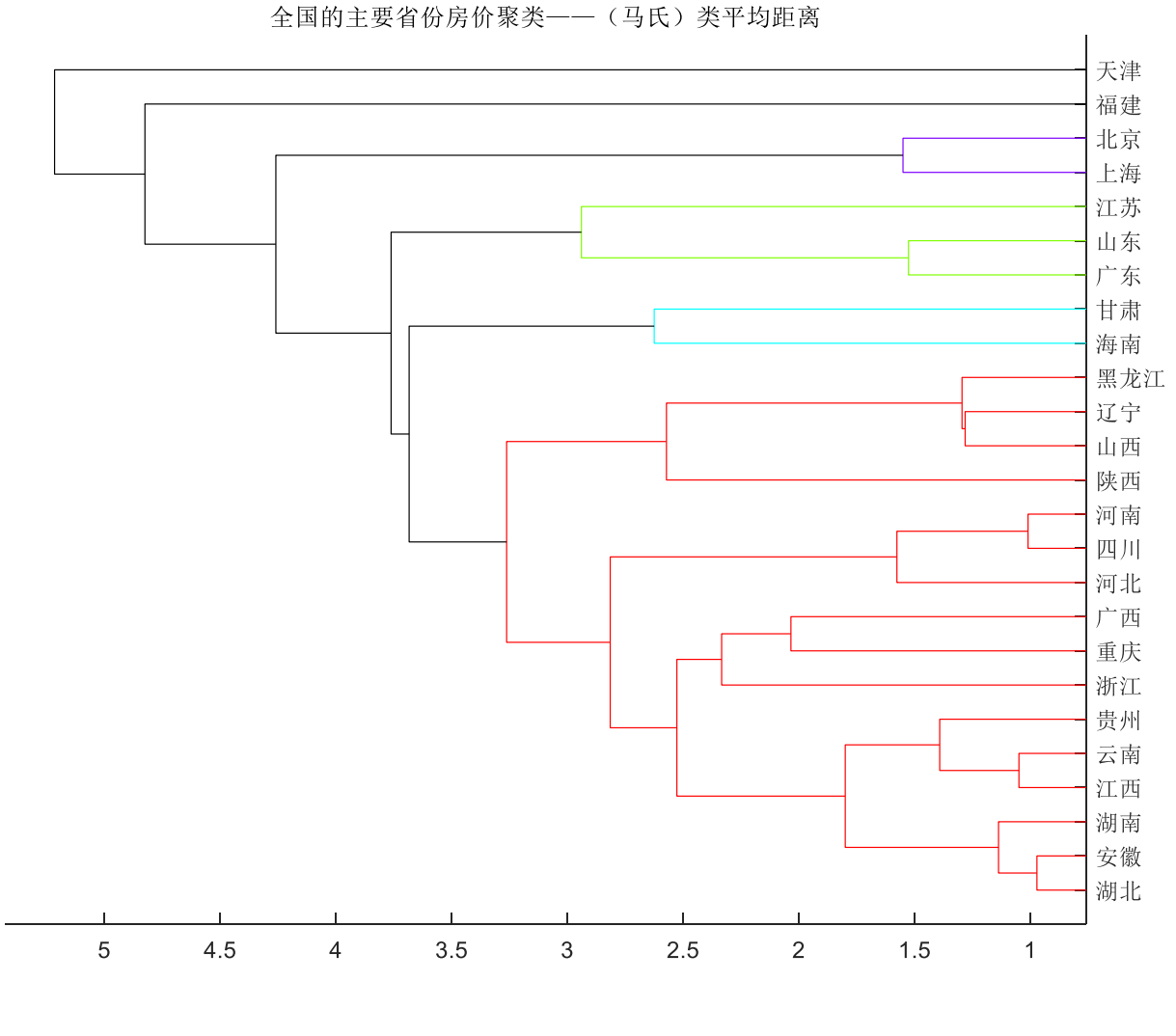
CA(:,1) = labels; %城市名称

CA(:,2) = num2cell(T); %类别

CA = sortrows(CA,2) %按类别扩展排序

马氏距离，考虑到了指标变量的协方差矩阵结构（相关性）对分类的影响，但忽略了各指标变量相对重要程度的差异。

```
CA =
  25×2 cell 数组
    '海南'      [1]
    '甘肃'      [1]
    '浙江'      [2]
    '河北'      [2]
    '湖北'      [2]
    '四川'      [2]
    '河南'      [2]
    '安徽'      [2]
    '陕西'      [2]
    '山西'      [2]
    '辽宁'      [2]
    '江西'      [2]
    '黑龙江'    [2]
    '重庆'      [2]
    '云南'      [2]
    '湖南'      [2]
    '广西'      [2]
    '贵州'      [2]
    '广东'      [3]
    '山东'      [3]
    '江苏'      [3]
    '上海'      [4]
    '北京'      [4]
    '福建'      [5]
    '天津'      [6]
```



谱系聚类的MATLAB实现

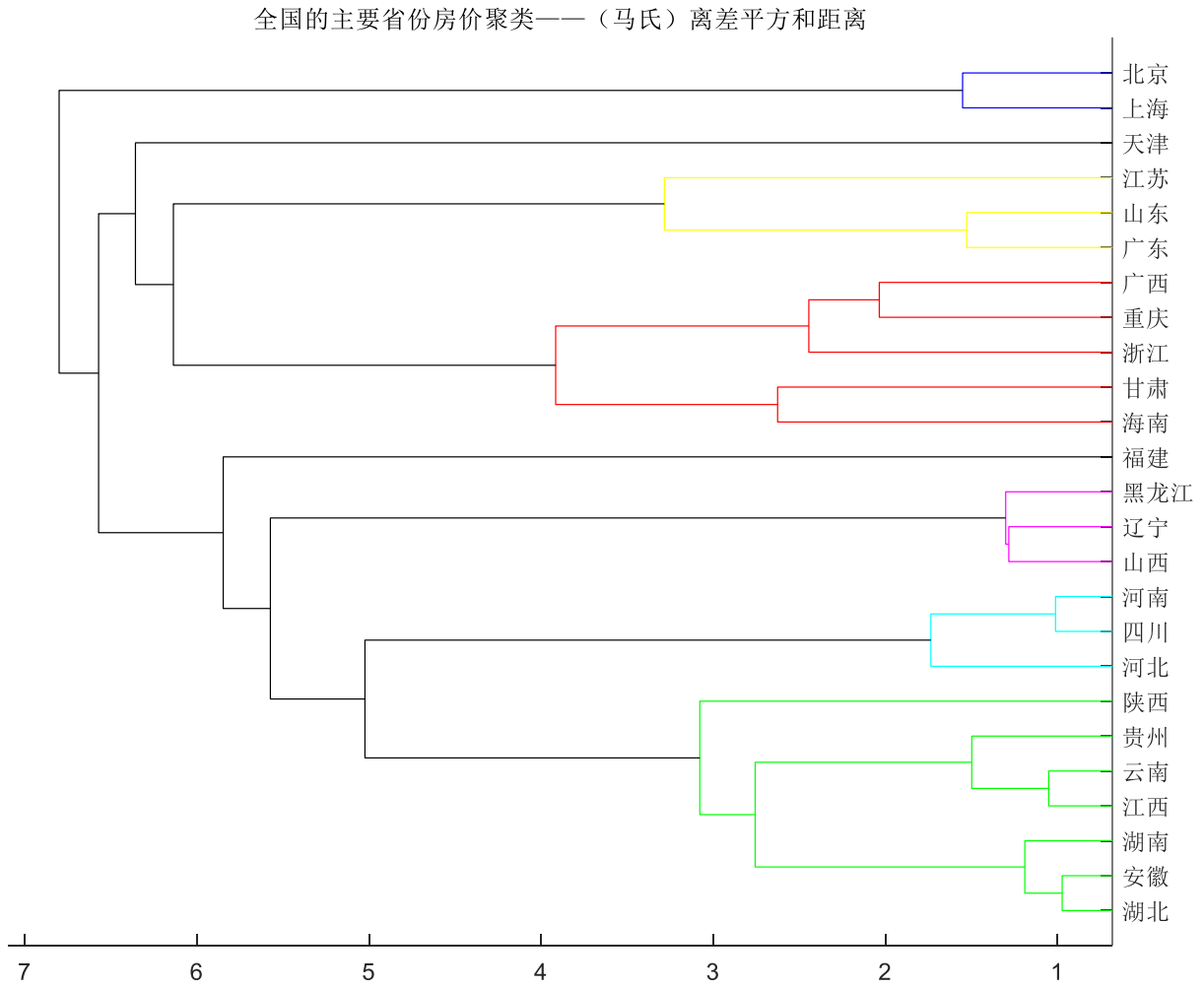
d = pdist(sale,'mahalanobis'); %马氏距离

z= linkage(d,'ward'); %离差平方和距离

```
CAWard =
  25×2 cell 数组
    '福建'      [1]
    '河北'      [2]
    '湖北'      [2]
    '四川'      [2]
    '河南'      [2]
    '安徽'      [2]
    '陕西'      [2]
    '山西'      [2]
    '辽宁'      [2]
    '江西'      [2]
    '黑龙江'    [2]
    '云南'      [2]
    '湖南'      [2]
    '贵州'      [2]
    '广东'      [3]
    '山东'      [3]
    '江苏'      [3]
    '海南'      [4]
    '浙江'      [4]
    '甘肃'      [4]
    '重庆'      [4]
    '广西'      [4]
    '天津'      [5]
    '上海'      [6]
    '北京'      [6]
```

离差平方和与类平均距离聚类结果近似

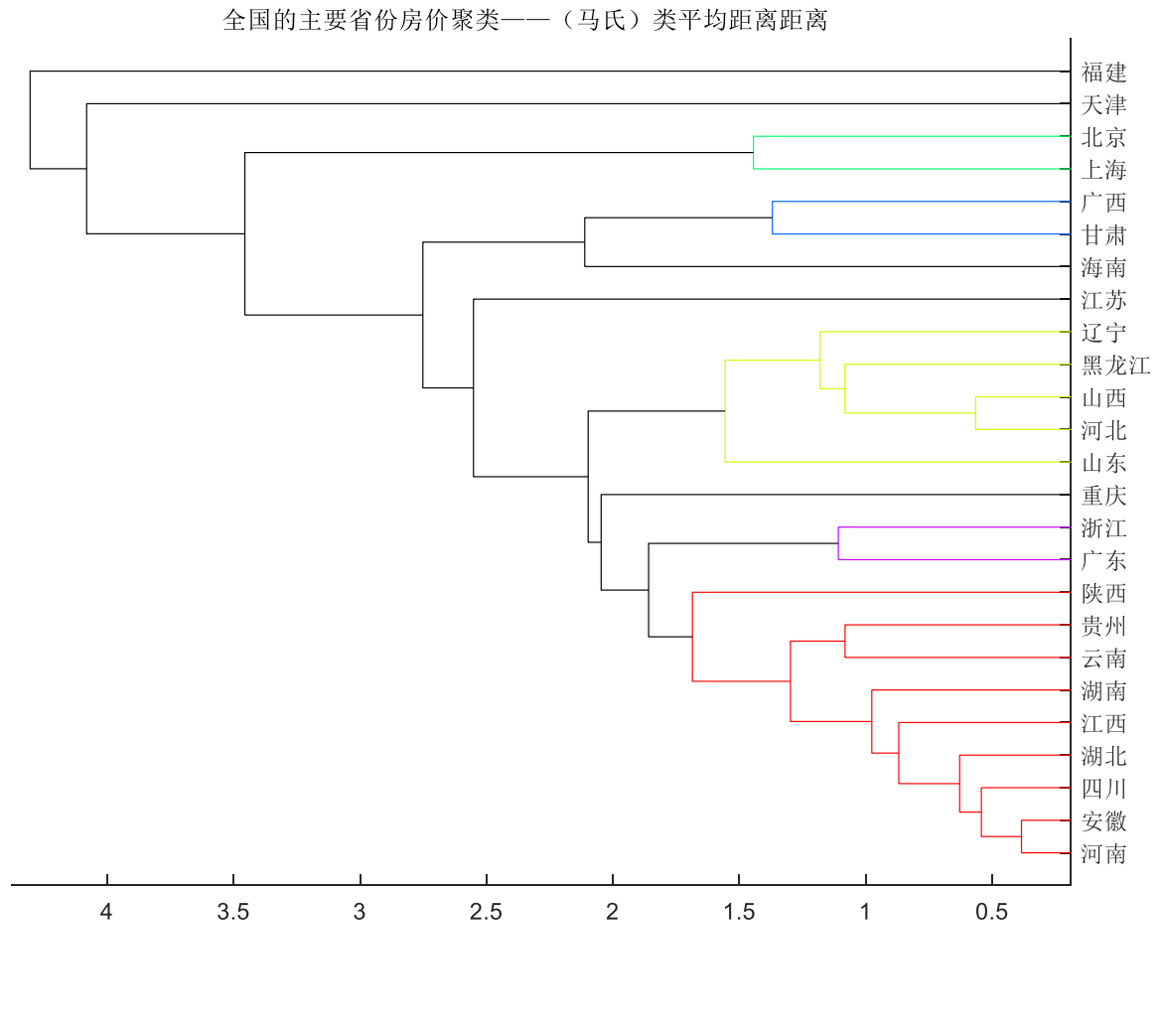
```
CA =
  25×2 cell 数组
    '海南'      [1]
    '甘肃'      [1]
    '浙江'      [2]
    '河北'      [2]
    '湖北'      [2]
    '四川'      [2]
    '河南'      [2]
    '安徽'      [2]
    '陕西'      [2]
    '山西'      [2]
    '辽宁'      [2]
    '江西'      [2]
    '黑龙江'    [2]
    '重庆'      [2]
    '云南'      [2]
    '湖南'      [2]
    '广西'      [2]
    '贵州'      [2]
    '广东'      [3]
    '山东'      [3]
    '江苏'      [3]
    '上海'      [4]
    '北京'      [4]
    '福建'      [5]
    '天津'      [6]
```



谱系聚类的MATLAB实现

```
[sale,text] = xlsread('sale_houseprice.xlsx');
%选择四个指标：人均GDP，GDP增长率，平均工资，平均房价
sale = sale(:,3:6);
d = pdist(sale,'mahalanobis');  %马氏距离
z= linkage(d,'average');   %类平均距离
labels = text(3:end,1);
dendrogram(z,'ColorThreshold',0.4*max(z(:,3)),'Labels',labels,'Orientation','left');  %谱系聚类图
title('全国的主要省份房价聚类——（马氏）类平均距离距离')
```

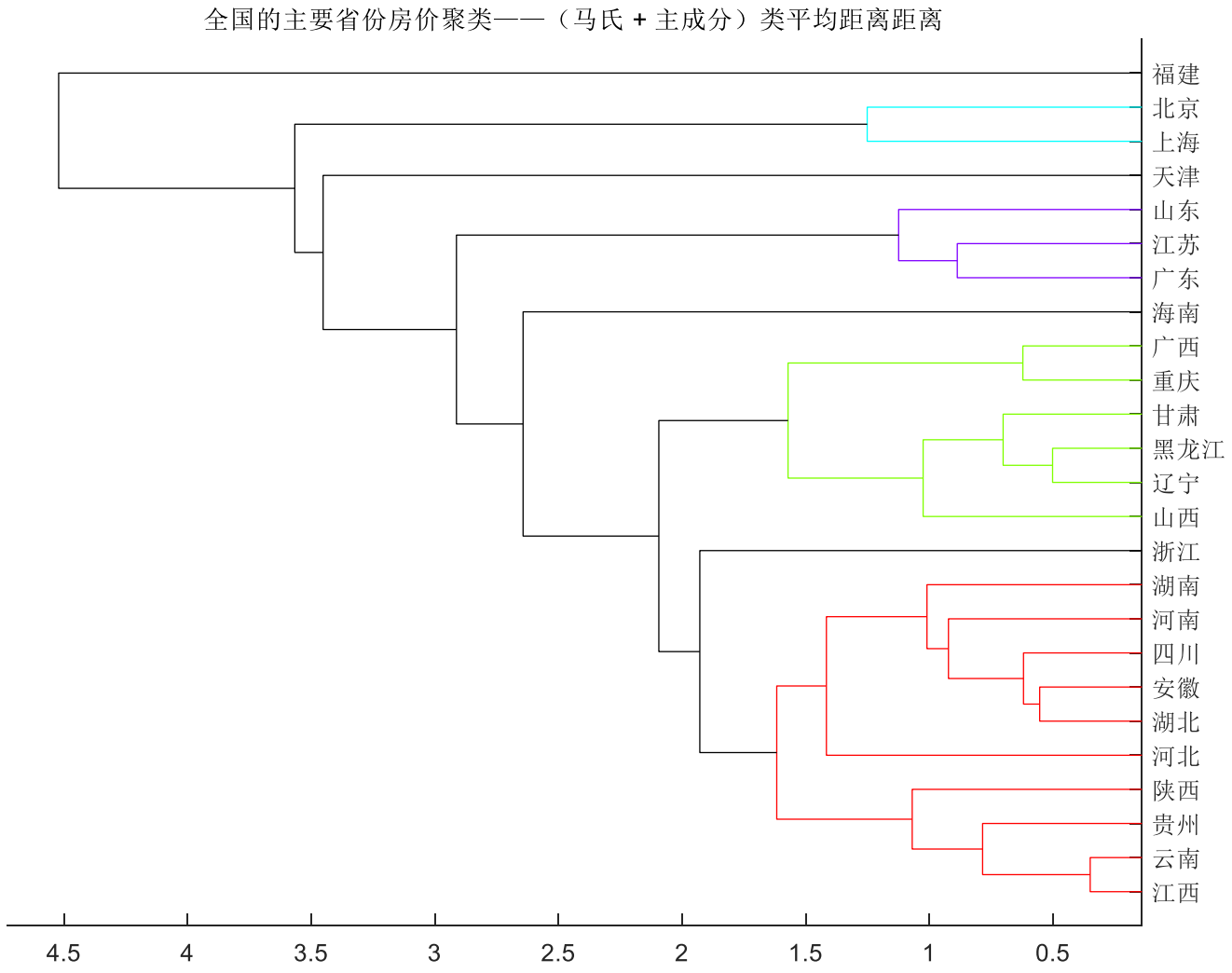
```
CA =
  25×2 cell 数组
    '江苏'    [1]
    '广东'    [2]
    '山东'    [2]
    '浙江'    [2]
    '河北'    [2]
    '湖北'    [2]
    '四川'    [2]
    '河南'    [2]
    '安徽'    [2]
    '陕西'    [2]
    '山西'    [2]
    '辽宁'    [2]
    '江西'    [2]
    '黑龙江'    [2]
    '重庆'    [2]
    '云南'    [2]
    '湖南'    [2]
    '贵州'    [2]
    '海南'    [3]
    '甘肃'    [3]
    '广西'    [3]
    '上海'    [4]
    '北京'    [4]
    '天津'    [5]
    '福建'    [6]
```



谱系聚类的MATLAB实现

```
[sale,text] = xlsread('sale_houseprice.xlsx');
sale = zscore(sale);
[coeff,score,latent,~,explained] = pca(sale);
% 选取前四个主成分，解释方差度为%96.6454
salepca = sale*coeff(:,1:4);
d = pdist(salepca,'mahalanobis');  %马氏距离
z= linkage(d,'average');   %类平均距离
labels = text(3:end,1);
dendrogram(z,'ColorThreshold',0.4*max(z(:,3)),'La
bels',labels,'Orientation','left');  %谱系聚类图
title('全国的主要省份房价聚类——（马氏 + 主成
分）类平均距离距离')
```

与前期分析差异不大，且主成分降维后，聚类效果比较合理。



谱系聚类的MATLAB实现

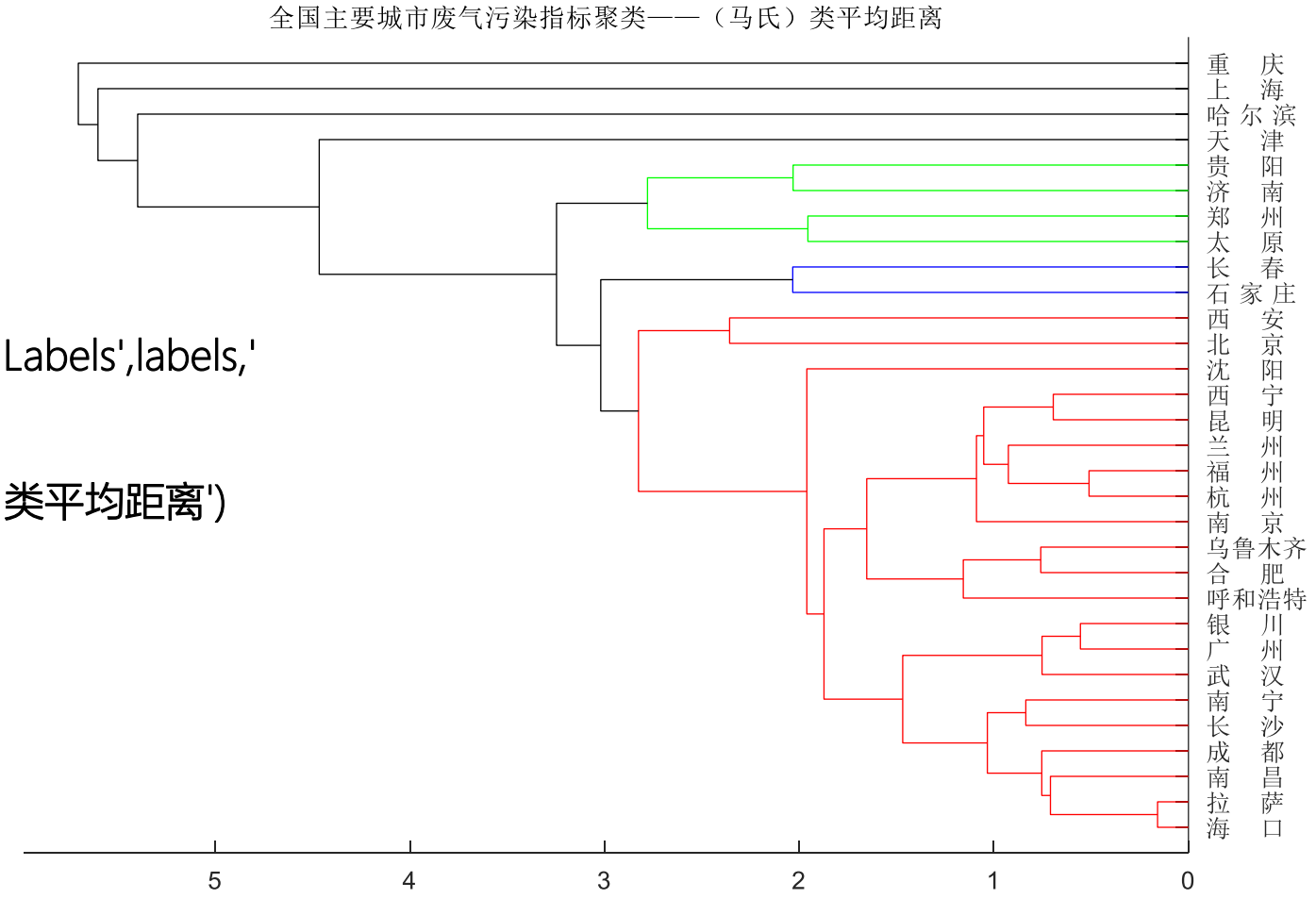
信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

例4：主要城市废气中主要污染物排放情况 (2013年)

城市	工业二氧化硫	工业氮氧化物	工业烟(粉)尘	生活二氧化硫	生活氮氧化物	生活烟尘	城市	工业二氧化硫	工业氮氧化物	工业烟(粉)尘	生活二氧化硫	生活氮氧化物	生活烟尘
北京	52041	75927	27182	34967	13638	28258	武汉	96222	95612	20020	5720	1416	1001
天津	207793	250646	62766	8959	5221	18400	长沙	21173	15951	19545	2366	153	2946
石家庄	176469	200301	99806	9564	2802	6635	广州	65589	57164	16660	663	276	214
太原	88880	96018	37003	33396	6738	26727	南宁	33045	34797	20950	8748	1068	4631
呼和浩特	96190	131665	48822	4257	665	3763	海口	1798	86	1149	11	17	5
沈阳	130672	83348	60425	14389	5154	15276	重庆	494415	247905	179842	53261	4487	4401
长春	57246	95190	72970	7344	1545	7919	成都	52040	44411	21452	4891	2109	661
哈尔滨	65987	85515	82323	50012	22985	80792	贵阳	70603	30450	24233	35493	1753	5530
上海	172867	262346	67174	42947	23474	6451	昆明	102842	68213	57366	5263	970	328
南京	110665	109693	65256	1750	400	1000	拉萨	930	2016	538	678	40	199
杭州	82021	67283	40243	633	335	135	西安	69103	34917	15893	23831	10951	14012
合肥	41483	70311	42387	2710	130	3188	兰州	72148	79915	40109	7413	1950	1088
福州	76043	72284	43483	1279	169	547	西宁	71839	53280	52765	7129	1419	4793
南昌	40756	18597	11413	641	58	254	银川	92369	84321	27170	5697	1237	3016
济南	81118	72969	47117	26087	3629	8355	乌鲁木齐	74216	113803	52441	6691	1425	4920
郑州	106123	134120	33828	11975	1780	9150							

谱系聚类的MATLAB实现

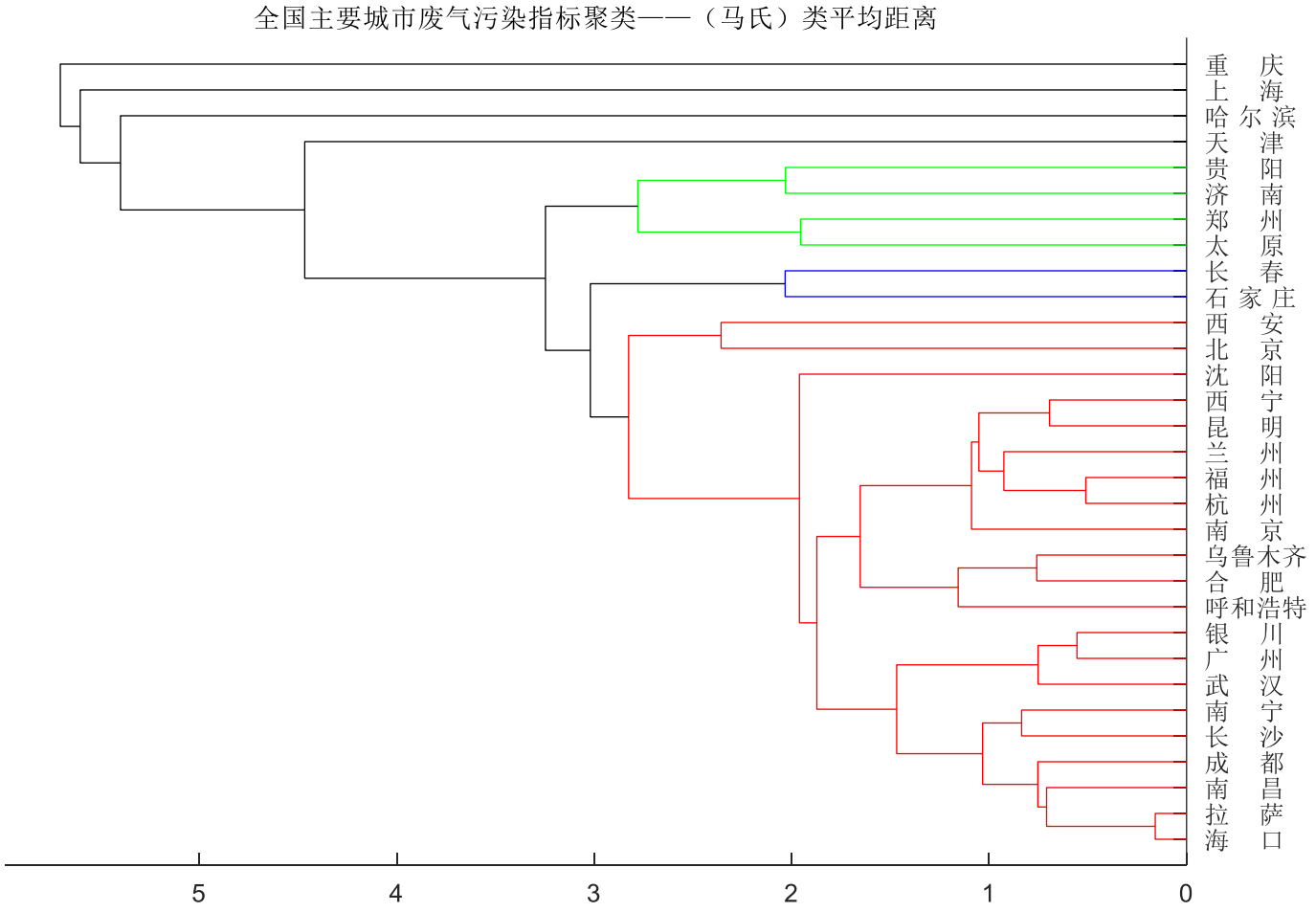
```
[orig,text] = xlsread('Z0816C-2013Orig');
md = pdist(orig,'mahalanobis');  %马氏距离
Z = linkage(md,'average');   %类平均距离
labels = text(7:end,1);
Ha=
dendrogram(Z,0,'ColorThreshold',0.5*max(Z(:,3)),'Labels',labels,'
Orientation','left');  %谱系聚类图
title('全国主要城市废气污染指标聚类——（马氏）类平均距离')

T=cluster(Z,6);   %输出4类聚类结果
CA = cell(31,2);  %用元胞数组组合
CA(:,1) = labels; %城市名称
CA(:,2) = num2cell(T);  %类别
CA = sortrows(CA,2)   %按类别扩展排序
```



谱系聚类的MATLAB实现

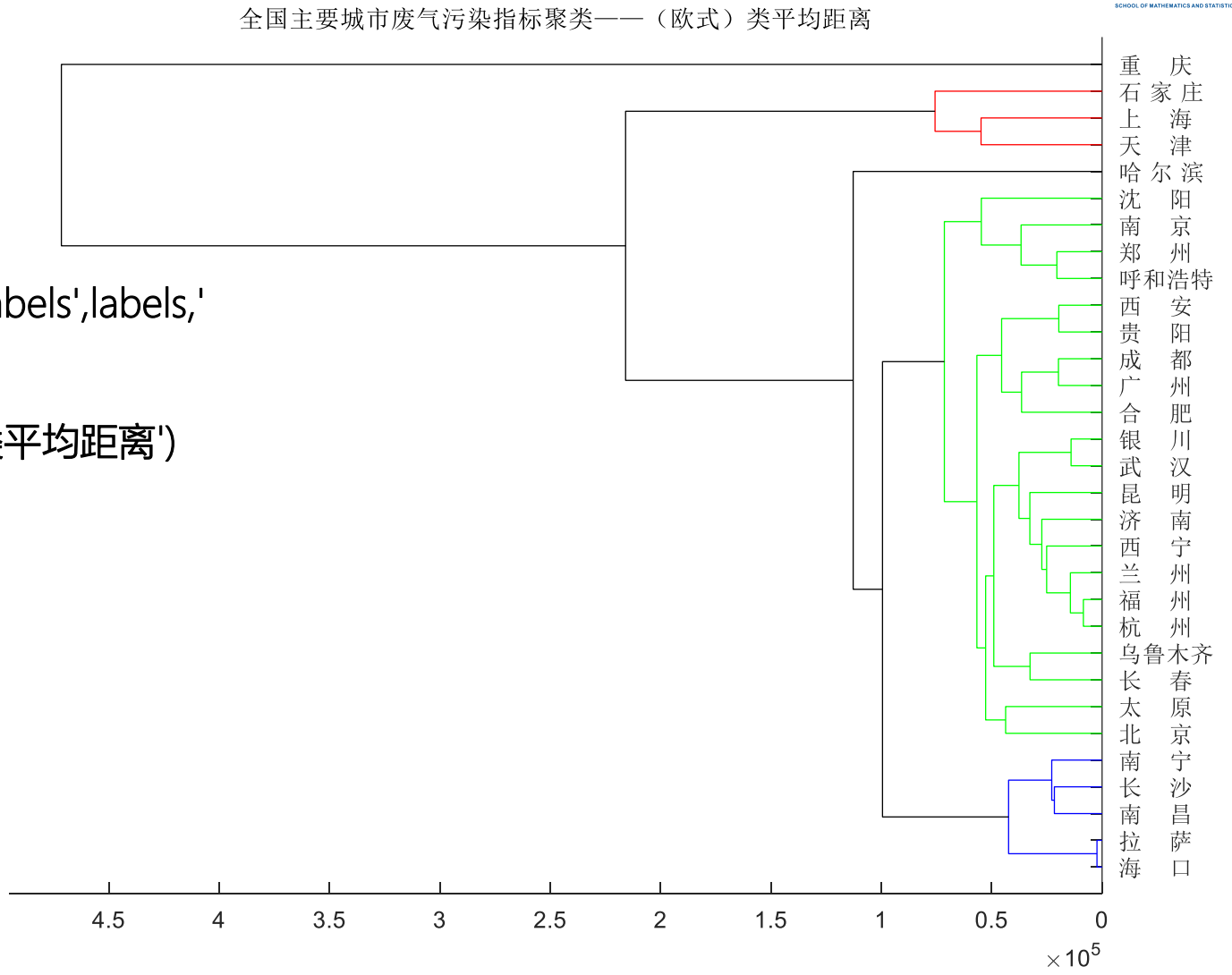
信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

```
CA =
  31×2 cell 数组
    '太　　原'　　[1]
    '济　　南'　　[1]
    '郑　　州'　　[1]
    '贵　　阳'　　[1]
    '北　　京'　　[2]
    '石 家 庄'　　[2]
    '呼和浩特'　　[2]
    '沈　　阳'　　[2]
    '长　　春'　　[2]
    '南　　京'　　[2]
    '杭　　州'　　[2]
    '合　　肥'　　[2]
    '福　　州'　　[2]
    '南　　昌'　　[2]
    '武　　汉'　　[2]
    '长　　沙'　　[2]
    '广　　州'　　[2]
    '南　　宁'　　[2]
    '海　　口'　　[2]
    '成　　都'　　[2]
    '昆　　明'　　[2]
    '拉　　萨'　　[2]
    '西　　安'　　[2]
    '兰　　州'　　[2]
    '西　　宁'　　[2]
    '银　　川'　　[2]
    '乌鲁木齐'　　[2]
    '天　　津'　　[3]
    '哈 尔 滨'　　[4]
    '上　　海'　　[5]
    '重　　庆'　　[6]
```



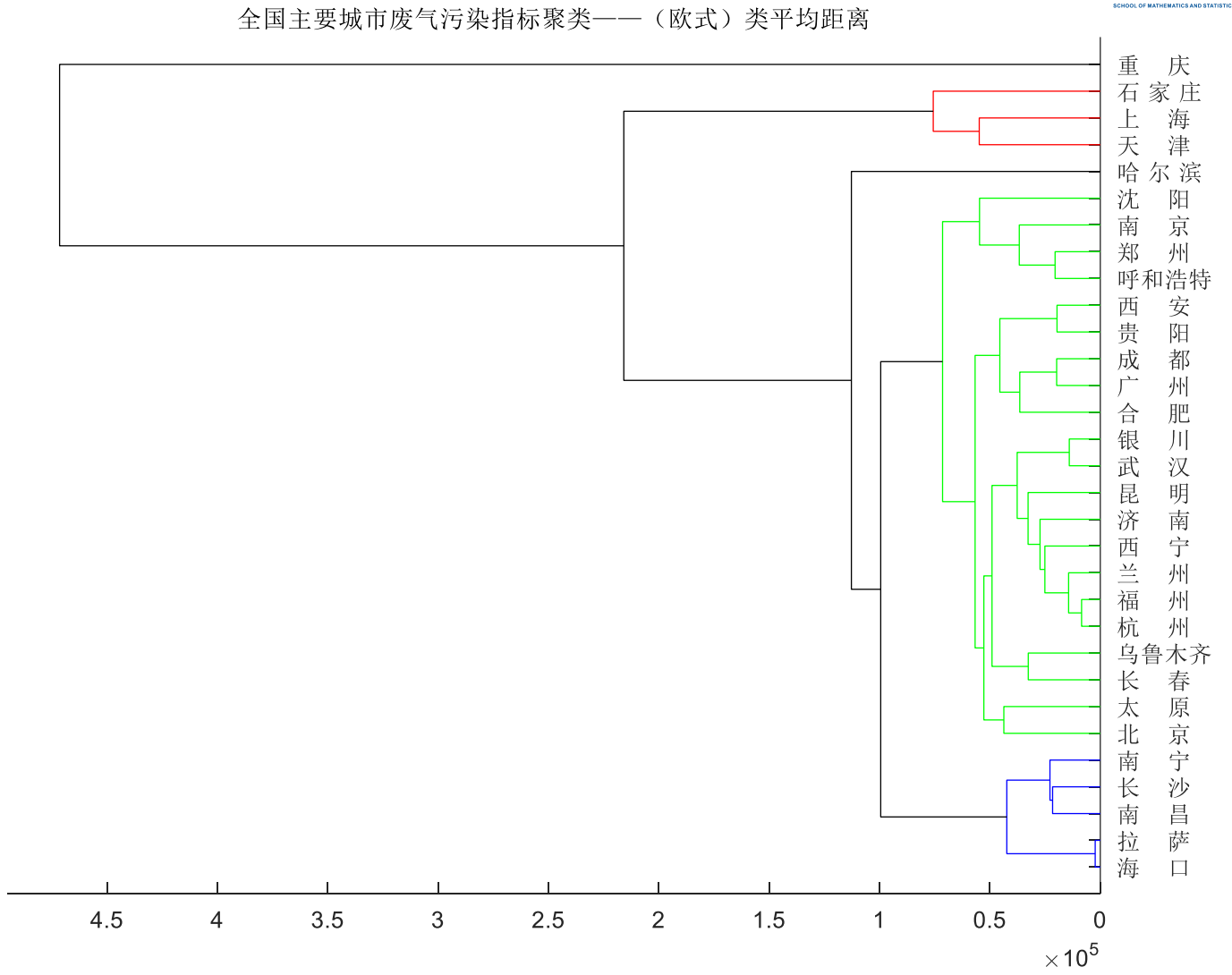
谱系聚类的MATLAB实现

```
md = pdist(orig);  %欧式距离
Z = linkage(md,'average');   %类平均距离
labels = text(7:end,1);
Ha=
dendrogram(Z,0,'ColorThreshold',0.2*max(Z(:,3)),'Labels',labels,'
Orientation','left');  %谱系聚类图
title('全国主要城市废气污染指标聚类——（欧式）类平均距离')

T=cluster(Z,6);   %输出4类聚类结果
CA = cell(31,2);  %用元胞数组组合
CA(:,1) = labels; %城市名称
CA(:,2) = num2cell(T);  %类别
CA = sortrows(CA,2)   %按类别扩展排序
```



谱系聚类的MATLAB实现

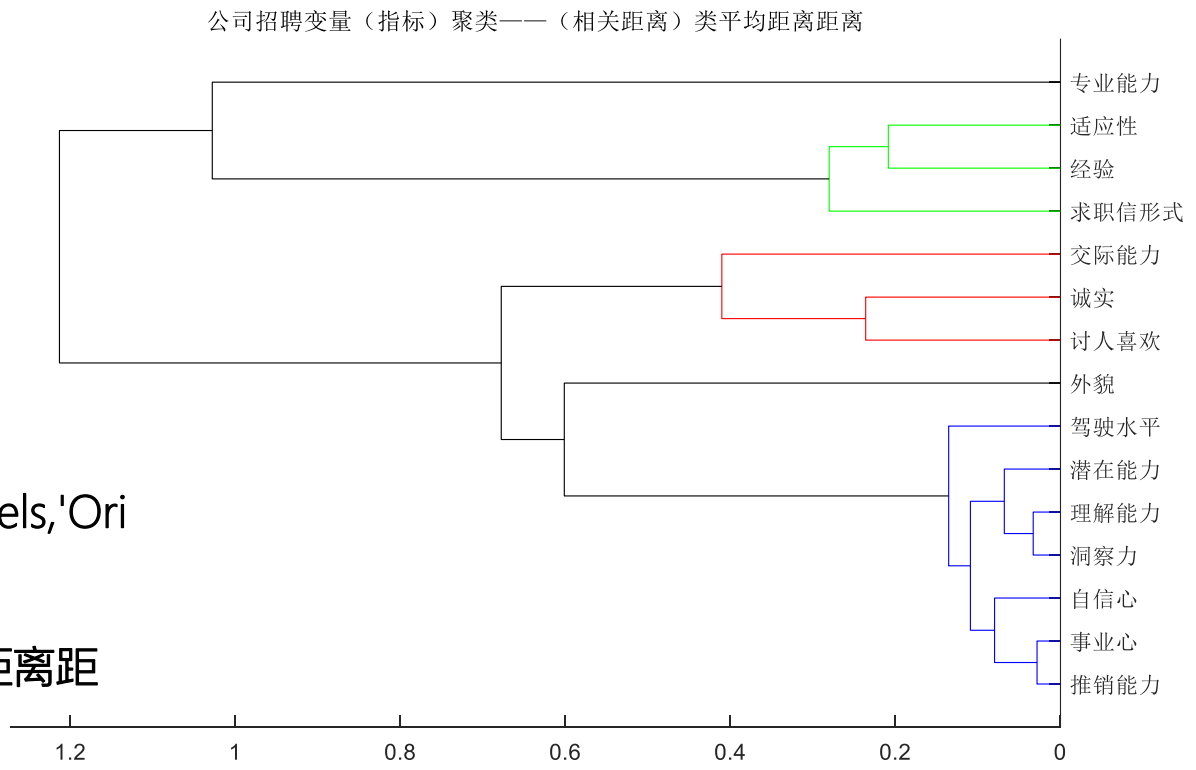
```
CA =
  31×2 cell 数组
    '石家庄'      [1]
    '天　津'      [2]
    '上　海'      [2]
    '南　昌'      [3]
    '长　沙'      [3]
    '南　宁'      [3]
    '海　口'      [3]
    '拉　萨'      [3]
    '北　京'      [4]
    '太　原'      [4]
    '呼和浩特'    [4]
    '沈　阳'      [4]
    '长　春'      [4]
    '南　京'      [4]
    '杭　州'      [4]
    '合　肥'      [4]
    '福　州'      [4]
    '济　南'      [4]
    '郑　州'      [4]
    '武　汉'      [4]
    '广　州'      [4]
    '成　都'      [4]
    '贵　阳'      [4]
    '昆　明'      [4]
    '西　安'      [4]
    '兰　州'      [4]
    '西　宁'      [4]
    '银　川'      [4]
    '乌鲁木齐'    [4]
    '哈尔滨'      [5]
    '重　庆'      [6]
```



谱系聚类——R型聚类

例5：48名应聘者应聘某公司的某职位，公司为这些应聘者的15项指标打分，15项指标说明：求职信形式：FL，外貌：APP，专业能力：AA，讨人喜欢：LA，自信心：SC，洞察力：LC，诚实：HON，推销能力：SMS，经验：EXP，驾驭水平：DRV，事业心：AMB，理解能力：GSP，潜在能力:POT，交际能力:KJ，适应性:SUIT。试对变量（指标）进行聚类分析。

```
[cand,text] = xlsread('candidate.xlsx');
candcoef = corrcoef(cand);
d = pdist(candcoef,'correlation');  %相关距离
z= linkage(d,'average');   %类平均距离
labels = text(2,2:end);
dendrogram(z,'ColorThreshold',0.4*max(z(:,3)),'Labels',labels,'Ori
entation','left');  %谱系聚类图
title('公司招聘变量（指标）聚类——（相关距离）类平均距离距
离')
```



该R型聚类结果与因子分析结果基本一致。

3. K-均值（K-Means）聚类

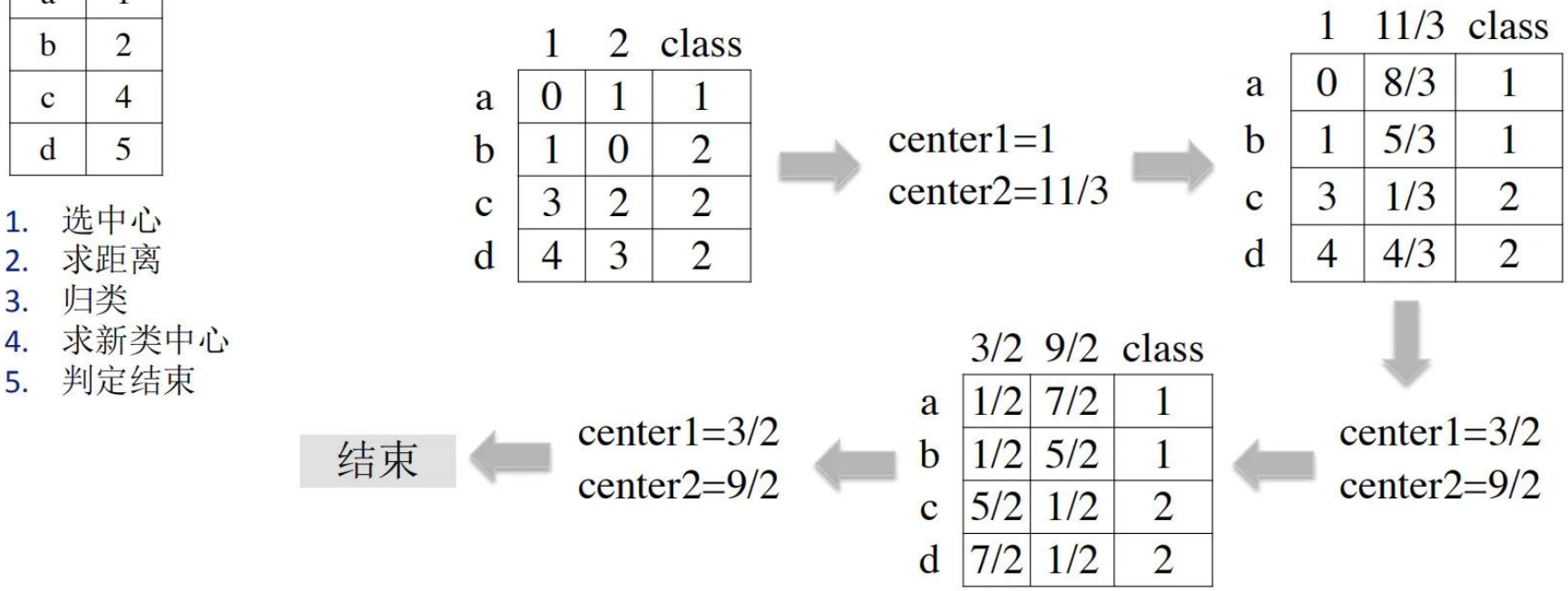
- 谱系聚类法是先将每个样品看成一类，通过比较距离的大小逐步扩充类。因此，对于给定的数据，谱系聚类一定能够将样品合并为一类，分类的结果唯一。
- 但是谱系聚类有一个缺点，样品一旦被分到某一类中就不能改变，且当样本容量较大时，计算量也相应地变大。克服此缺点的一个方法就是K均值聚类法，又称快速聚类法或动态聚类法。
- K-均值聚类算法（k-means clustering algorithm）是一种迭代求解的聚类分析算法，其步骤：预将数据分为K组，则随机选取K个对象作为初始的聚类中心，然后计算每个对象与各个种子聚类中心之间的距离，把每个对象分配给距离它最近的聚类中心。聚类中心以及分配给它们的对象就代表一个聚类。每分配一个样本，聚类的聚类中心会根据聚类中现有的对象被重新计算。这个过程将不断重复直到满足某个终止条件。终止条件可以是没有（或最小数目）对象被重新分配给不同的聚类，没有（或最小数目）聚类中心再发生变化，误差平方和局部最小。
- 在运用K均值聚类法之前，要根据实际问题先确定分类数k， 在每一类中选择有代表性的样品，这样的样品称为聚点。选择聚点的方法通常有最小最大原则。

3. K-均值（K-Means）聚类

- K-均值聚类算法步骤：预将数据分为K组，则随机选取K个对象作为初始的聚类中心，然后计算每个对象与各个种子聚类中心之间的距离，把每个对象分配给距离它最近的聚类中心。聚类中心以及分配给它们的对象就代表一个聚类。每分配一个样本，聚类的聚类中心会根据聚类中现有的对象被重新计算。这个过程将不断重复直到满足某个终止条件。终止条件可以是没有（或最小数目）对象被重新分配给不同的聚类，没有（或最小数目）聚类中心再发生变化，误差平方和局部最小。

a	1
b	2
c	4
d	5

1. 选中心
2. 求距离
3. 归类
4. 求新类中心
5. 判定结束



3. K-均值（K-Means）聚类

优点：

1、原理比较简单，实现也是很容易，收敛速度快。

2、当结果簇是密集的，而簇与簇之间区别明显时，它的效果较好。

3、主要需要调参的参数仅仅是簇数k。

缺点：

1、K值需要预先给定，很多情况下K值的估计是非常困难的。

2、K-Means算法对初始选取的质心点是敏感的，不同的随机种子点得到的聚类结果完全不同，对结果影响很大。

3、对噪音和异常点比较的敏感。用来检测异常值。

4、采用迭代方法，可能只能得到局部的最优解，而无法得到全局的最优解。

3. K-均值（K-Means）聚类

1、K值的选定：分几类主要取决于经验与实际样本信息，通常的做法是多尝试几个K值，看分成几类的结果更好解释，更符合分析目的等。或者可以把各种K值算出的E做比较，取最小的E的K值。

2、初始的K个质心怎么选？最常用的方法是随机选，初始质心的选取对最终聚类结果有影响，因此算法一定要多执行几次，哪个结果更reasonable，就用哪个结果。 当然也有一些优化的方法，第一种是选择彼此距离最远的点，具体来说就是先选第一个点，然后选离第一个点最远的第二个点，然后选第三个点，第三个点到第一、第二两点的距离之和最小，以此类推。第二种是先根据其他聚类算法（如层次聚类）得到聚类结果，从结果中每个分类选一个点。

3、关于离群值？离群值就是远离整体的，非常异常、非常特殊的数据点，在聚类之前应该将这些“极大”“极小”之类的离群数据都去掉，否则会对于聚类的结果有影响。但是，离群值往往自身就很有分析的价值，可以把离群值单独作为一类来分析。

3. K-均值（K-Means）聚类

4、单位要一致！比如X的单位是米，Y也是米，那么距离算出来的单位还是米，是有意义的。但是如果X是米，Y是吨，用距离公式计算就会出现“米的平方”加上“吨的平方”再开平方，最后算出的东西没有数学意义，这就有问题了。

5、标准化：如果数据中X整体都比较小，比如都是1到10之间的数，Y很大，比如都是1000以上的数，那么，在计算距离的时候Y起到的作用就比X大很多，X对于距离的影响几乎可以忽略，这也有问题。因此，如果K-Means聚类中选择欧几里德距离计算距离，数据集又出现了上面所述的情况，就一定要进行数据的标准化（normalization），即将数据按比例缩放，使之落入一个小的特定区间。

https://www.jianshu.com/p/4f032dccdcef

K-均值聚类

若将n个样品分成k类，则先选择所有样品中距离最远的两个样品x_{i1}, x_{i2}为前两个聚点，即选择x_{i1}, x_{i2}，使得

$$d(x_{i1}, x_{i2}) = d_{i1i2} = \max\{d_{ij}\}$$

然后选择第3个聚点x_{i3}，使得x_{i3}与前两个聚点的距离最小者等于所有其余的与x_{i3}的较小距离中最大的，即

$$\min\{d(x_{i3}, x_{ir}), r = 1, 2\} = \max\{\min\{d(x_j, x_{ir}), r = 1, 2\}, j \neq i1, i2\}$$

然后按相同的原则选取x_{ik}，重复前面的步骤，直至确定k个聚点 $x_{i1}, x_{i2}, \ldots, x_{ik}$

K-均值聚类的步骤

样品之间的距离采用欧氏距离。

(1) 设第k个初始聚点的集合是：$L^{(0)}=\{x_1^{(0)},x_2^{(0)},\ldots,x_k^{(0)}\}$.

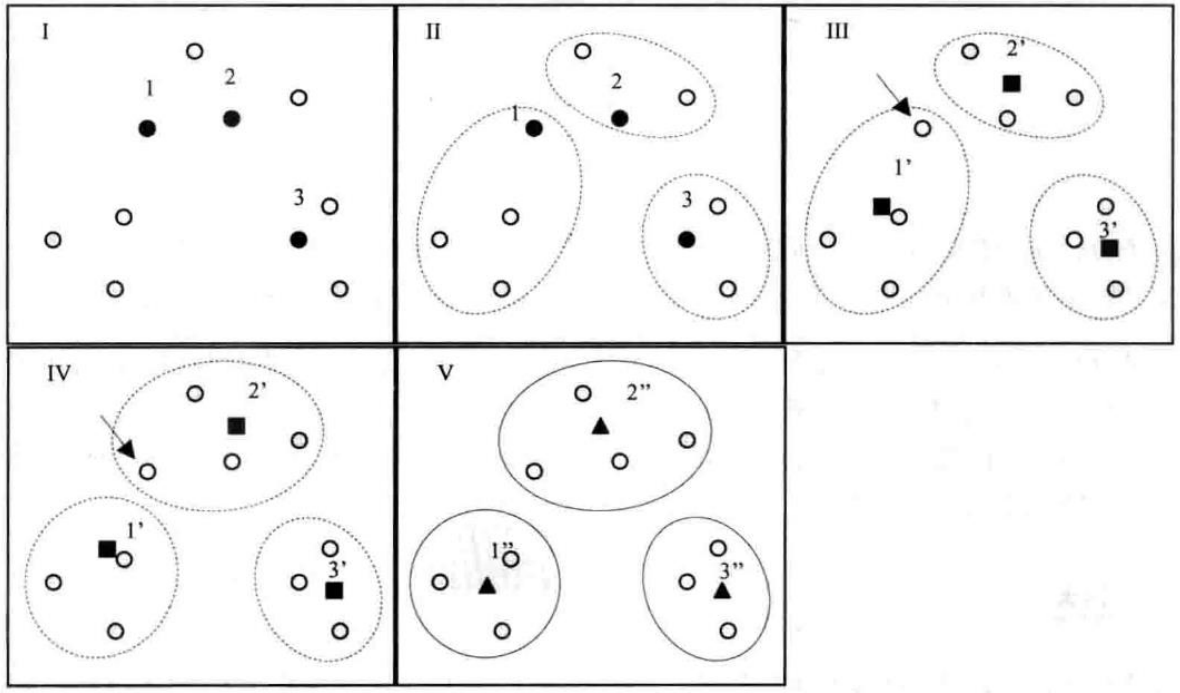
记 $G_i^{(0)}=\{x:d(x,x_i^{(0)})\le d(x,x_j^{(0)}),\ j=1,2,\ldots,k,j\ne i\},\ i=1,2,\ldots,k$

于是，将样品分成不相交的k类，得到一个初始分类 $G^{(0)}=\{G_1^{(0)},G_2^{(0)},\ldots,G_k^{(0)}\}$

(2) 从G^0出发，计算新的聚点集合$L^{(1)}$，计算

$$x_i^{(1)}=\frac{1}{n_i}\sum_{x_l\in G_i^{(0)}}x_l,\ i=1,2\ldots,k.$$

其中n_i是类G^0中的样品数，得到一个新的集合 $L^{(1)}=\{x_1^{(1)},x_2^{(1)},\ldots,x_k^{(1)}\}$.



K-均值聚类的步骤

从$L^{(1)}$开始再进行分类，将样品作新的分类，记

$$G_i^{(1)}=\{x:d(x,x_i^{(1)})\le d(x,x_j^{(1)}),\ j=1,2,\ldots,k,\ j\ne i\},\ i=1,2,\ldots,k$$

得到一个新的分类 $G^{(1)}=\{G_1^{(1)},G_2^{(1)},\ldots,G_k^{(1)}\}$，依次重复计算下去.

(3) 重复上述步骤m次得 $G^{(m)}=\{G_1^{(m)},G_2^{(m)},\ldots,G_k^{(m)}\}$

其中$x_i^{(m)}$是类$G_i^{(m-1)}$的重心。$x_i^{(m)}$不一定是样品，当m逐渐增大时，分类趋于稳定，同时$x_i^{(m)}$可以近似地看作$G_i^{(m)}$的重心。即 $x_i^{(m+1)}\approx x_i^{(m)}, G_i^{(m+1)}\approx G_i^{(m)}$，此时结束计算。

实际计算时，若对某个 m，$G^{(m+1)}=\{G_1^{(m+1)},G_2^{(m+1)},\ldots,G_k^{(m+1)}\}$ 与 $G^{(m)}=\{G_1^{(m)},G_2^{(m)},\ldots,G_k^{(m)}\}$ 相同，则结束计算。

K-均值聚类的MATLAB命令

`idx = kmeans(X,k)` 执行 k 均值聚类，以将 n×p 数据矩阵 X 的观测值划分为 k 个聚类，并返回包含每个观测值的簇索引的 n×1 向量 (idx)。X 的行对应于点，列对应于变量。

默认情况下，kmeans 使用欧几里德距离平方度量，并用 k-means++ 算法进行簇中心初始化。

`idx = kmeans(X,k,Name,Value)` 进一步按一个或多个 Name,Value 对组参数所指定的附加选项返回簇索引。

例如，指定余弦距离、使用新初始值重复聚类的次数或使用并行计算的次数。

`[idx,C] = kmeans(___)` 在 k×p 矩阵 C 中返回 k 个簇质心的位置。

`[idx,C,sumd] = kmeans(___)` 在 k×1 向量 sumd 中返回簇内的点到质心距离的总和。

`[idx,C,sumd,D] = kmeans(___)` 在 n×k 矩阵 D 中返回每个点到每个质心的距离。

名称-值对组参数： 'Distance','cosine','Replicates',10,'Options',statset('UseParallel',1) 指定使用余弦距离，以不同起始值重复 10 次聚类，并使用并行计算。

'Distance'参数：p 维空间中的距离度量，用于最小化距离和，指定为以逗号分隔的对组，其中包含 'Distance' 和 'sqeuclidean'、'cityblock'、'cosine'、'correlation' 或 'hamming'。

案例分析

信阳师范学院
数学与统计学院

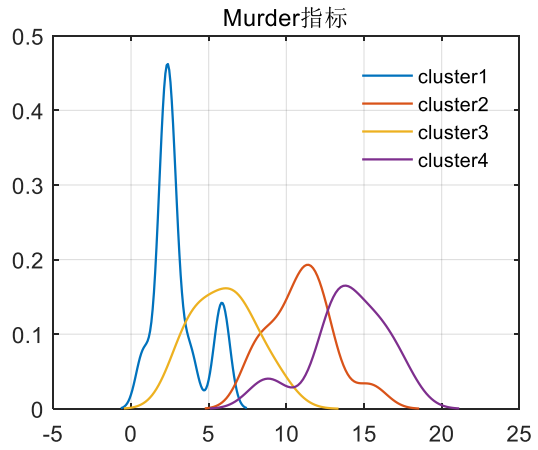
例6：1973年美国50个州每10万居民因袭击、谋杀和强奸被捕人数的统计数据。同时给出了居住在城市地区的人口百分比。各变量含义：Murder：谋杀数字，谋杀逮捕（每100000人）；Assault：攻击性数字，攻击逮捕（每100000人）；UrbanPop：城市人口百分比；Rape：强奸数字，强奸逮捕（每100000人）。

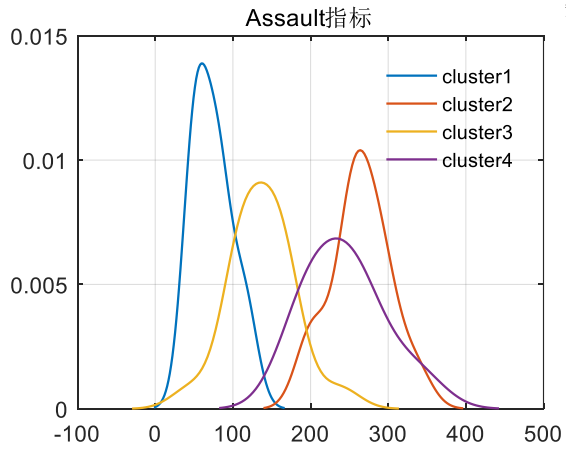
```
[ar,text] = xlsread('USArrests.xlsx');
arzs = zscore(ar);
[ind,C]=kmeans(arzs,4);
rownames = text(2:51,1);
result = cell(50,2);
result(:,1) = rownames;
result(:,2) = num2cell(ind);
result = sortrows(result,2)
ar1 = ar(ind == 1,:);
ar2 = ar(ind == 2,:);
```

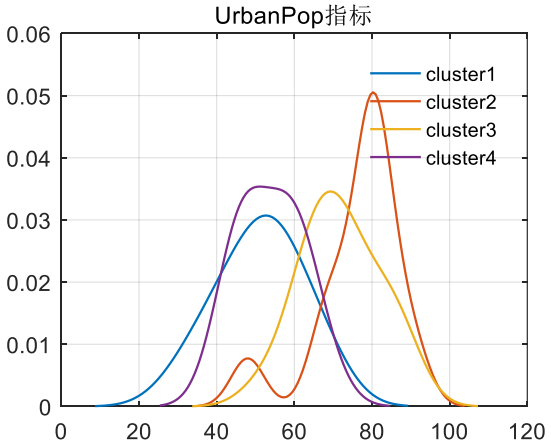
```
result =
  50×2 cell 数组
    'Idaho'            [1]
    'Iowa'             [1]
    'Kentucky'         [1]
    'Maine'            [1]
    'Minnesota'        [1]
    'Montana'          [1]
    'Nebraska'         [1]
    'New Hampshire'    [1]
    'North Dakota'     [1]
    'South Dakota'     [1]
    'Vermont'          [1]
    'West Virginia'    [1]
    'Wisconsin'        [1]
    'Arizona'          [2]
    'California'       [2]
    'Colorado'         [2]
    'Florida'          [2]
    'Illinois'         [2]
    'Maryland'         [2]
    'Michigan'         [2]
    'Nevada'           [2]
    'New Mexico'       [2]
    'New York'         [2]
    'Texas'            [2]
    'Alabama'          [3]
    'Alaska'           [3]
    'Arkansas'         [3]
    'Georgia'          [3]
    'Louisiana'        [3]
    'Mississippi'      [3]
    'North Carolina'   [3]
    'South Carolina'   [3]
    'Tennessee'        [3]
    'Connecticut'      [4]
    'Delaware'         [4]
    'Hawaii'           [4]
    'Indiana'          [4]
    'Kansas'           [4]
    'Massachusetts'    [4]
    'Missouri'         [4]
    'New Jersey'       [4]
    'Ohio'             [4]
    'Oklahoma'         [4]
    'Oregon'           [4]
    'Pennsylvania'     [4]
    'Rhode Island'     [4]
    'Utah'             [4]
    'Virginia'         [4]
    'Washington'       [4]
    'Wyoming'          [4]
```

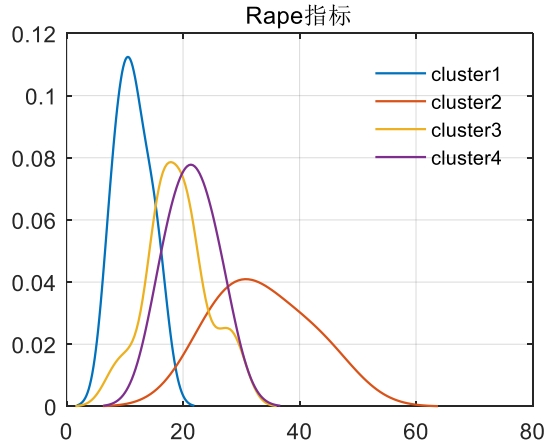
案例分析

```
ar3 = ar(ind == 3,:);
ar4 = ar(ind == 4,:);
varnames = text(1,2:5);
for i = 1:4
    subplot(2,2,i)
    [f1,xi1]=ksdensity(ar1(:,i));
    [f2,xi2]=ksdensity(ar2(:,i));
    [f3,xi3]=ksdensity(ar3(:,i));
    [f4,xi4]=ksdensity(ar4(:,i));
    plot(xi1,f1,xi2,f2,xi3,f3,xi4,f4,'LineWidth',1)
    grid on
    legend('cluster1','cluster2','cluster3','cluster4')
    legend('boxoff')
    title(strcat(varnames{i},'指标'))
end
```





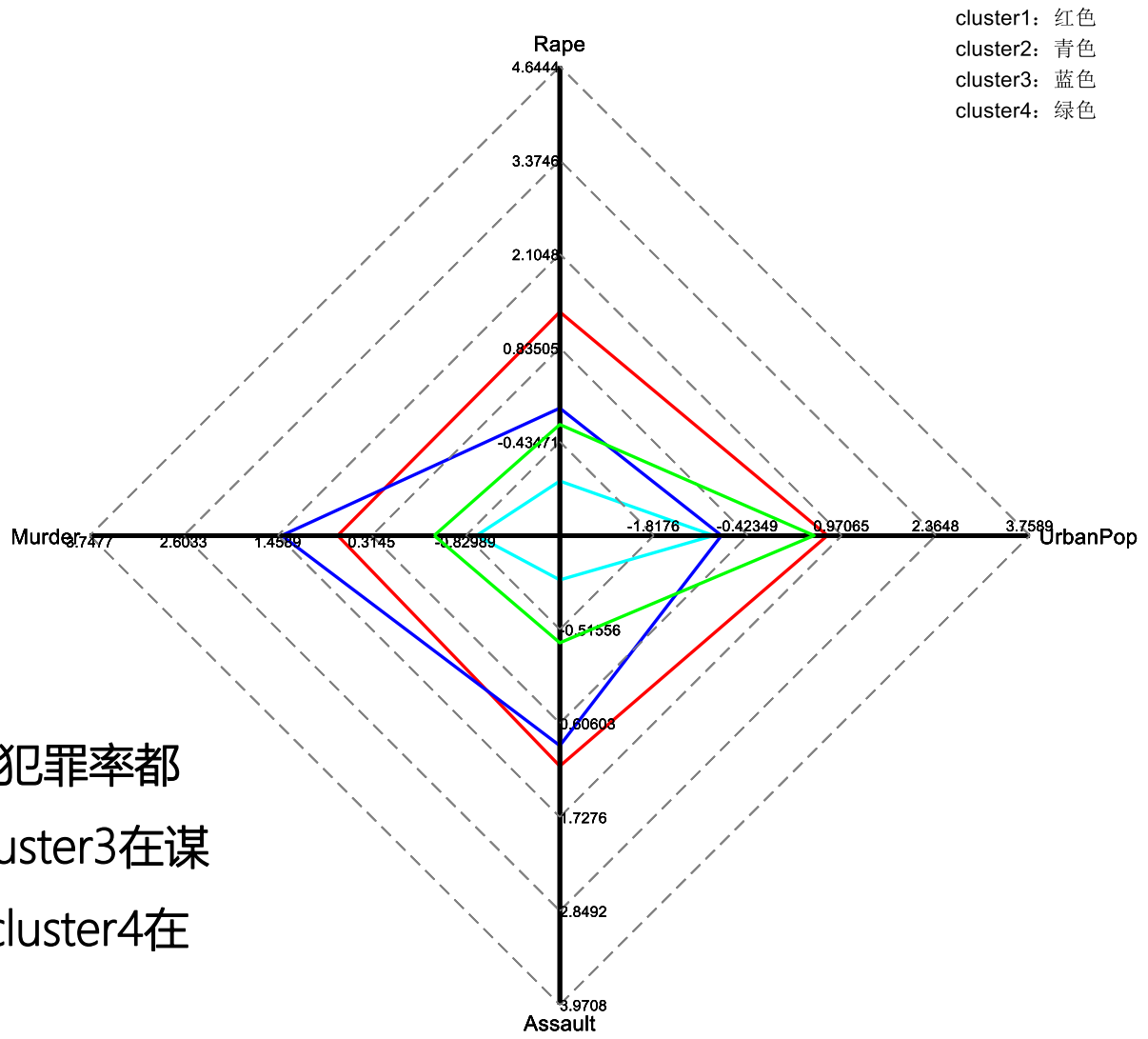




从划分4个类别的角度看，各指标分类结果较好，谋杀指标聚类最好。

案例分析

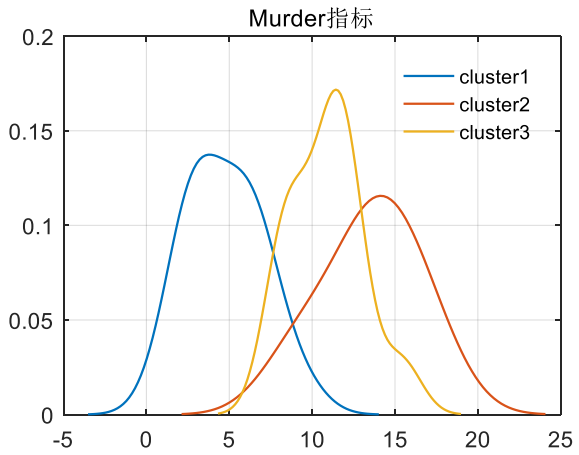
```
ar1 = arzs(ind == 1,:); ar2 = arzs(ind == 2,:);
ar3 = arzs(ind == 3,:); ar4 = arzs(ind == 4,:);
min1 = min(ar1);   max1 = max(ar1);
lim = [min1 - 2;max1+2]';
prefer_range = [min1 - 1;max1 + 1]';
%对每个cluster每个指标取均值绘制雷达图
draw_radar(mean(ar1),lim,prefer_range,varnames,'r')
draw_radar(mean(ar2),lim,prefer_range,varnames,'c')
draw_radar(mean(ar3),lim,prefer_range,varnames,'b')
draw_radar(mean(ar4),lim,prefer_range,varnames,'g')
```

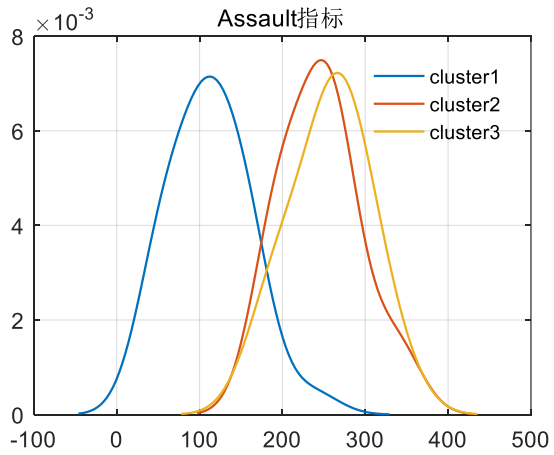


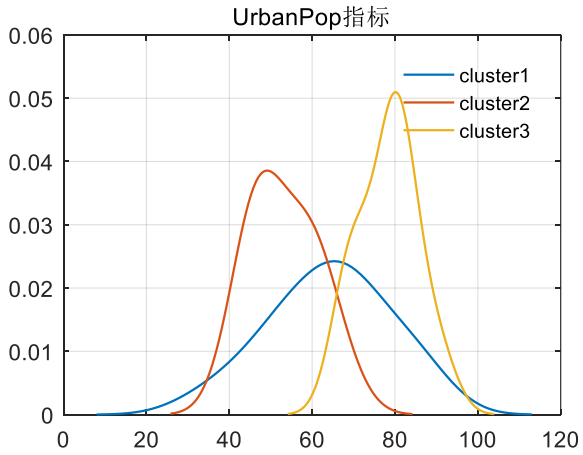
从雷达图上看出，cluster1在各方面值都较大，各种犯罪率都较高。cluster2在正好相反，各种犯罪率都较低；cluster3在谋杀犯罪率上最高，在攻击性、强奸率上较为突出；cluster4在城市人口百分比上较为突出，其他犯罪率方面较低。

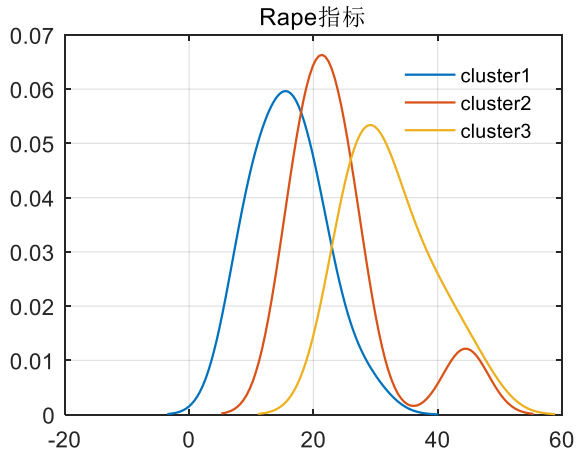
案例分析

```
[ind,C]=kmeans(arzs,3);
ar1 = ar(ind == 1,:); ar2 = ar(ind == 2,:);
ar3 = ar(ind == 3,:); varnames = text(1,2:5);
for i = 1:4
    subplot(2,2,i)
    [f1,xi1]=ksdensity(ar1(:,i));
    [f2,xi2]=ksdensity(ar2(:,i));
    [f3,xi3]=ksdensity(ar3(:,i));
    plot(xi1,f1,xi2,f2,xi3,f3,'LineWidth',1)
    grid on
    legend('cluster1','cluster2','cluster3')
    legend('boxoff')
    title(strcat(varnames{i},'指标'))
end
```









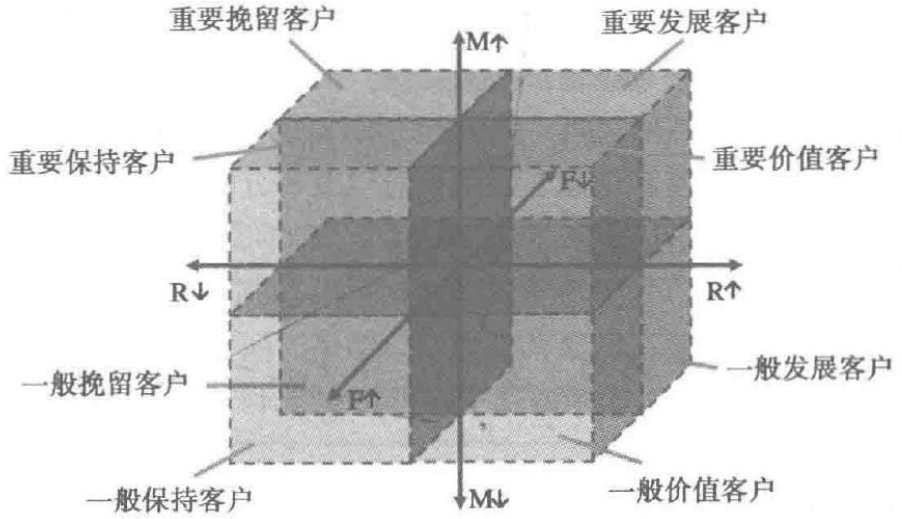
从划分3个类别的角度看，各指标分类结果比较好，聚类效果非常明显，故可聚类为三类。

案例分析

例6：部分餐饮客户的消费行为特征数据，共有940个样本数据，部分数据如下表所示。其中指标R表示最近一次消费时间间隔，F表示消费频率，M表示消费总金额。通过建立合理的客户价值评估模型，对客户进行分群，分析比较不同客户群的客户价值，并制定相应的营销策略，对不同的客户群提供个性化的客户服务是非常必要的。

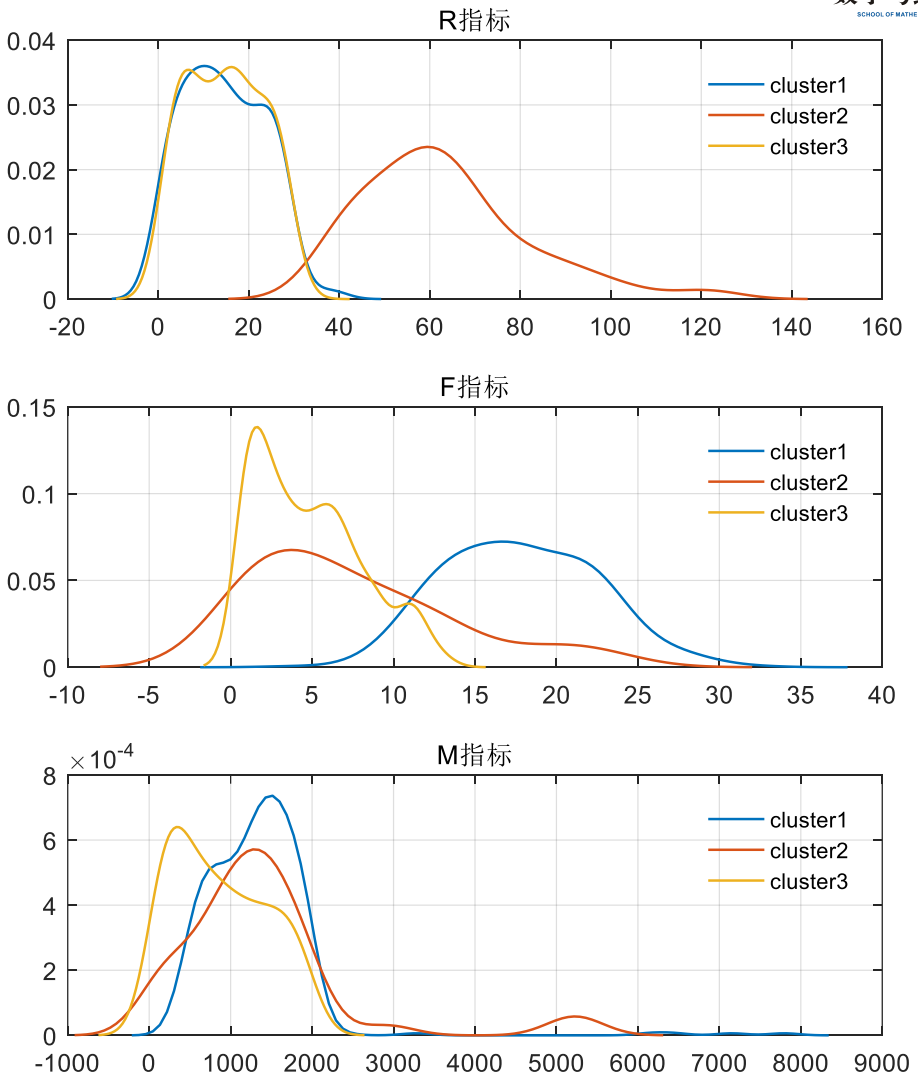
RFM模型是衡量客户价值和客户创利能力的重要工具和手段，它通过一个客户的近期购买行为、购买的总体频次以及购买的总体金额三个指标来描述客户的价值状况。分别为：最近消费时间间隔(Recently)、消费频率(Frequency)、消费金额(Money)。

Id	R	F	M	Id	R	F	M
1	27	6	232.61	7	5	2	615.83
2	3	5	1507.11	8	26	2	1059.66
3	4	16	817.62	9	21	9	304.82
4	3	11	232.81	10	2	21	1227.96
5	14	7	1913.05	11	15	2	521.02
6	19	6	220.07	12	26	3	438.22



案例分析

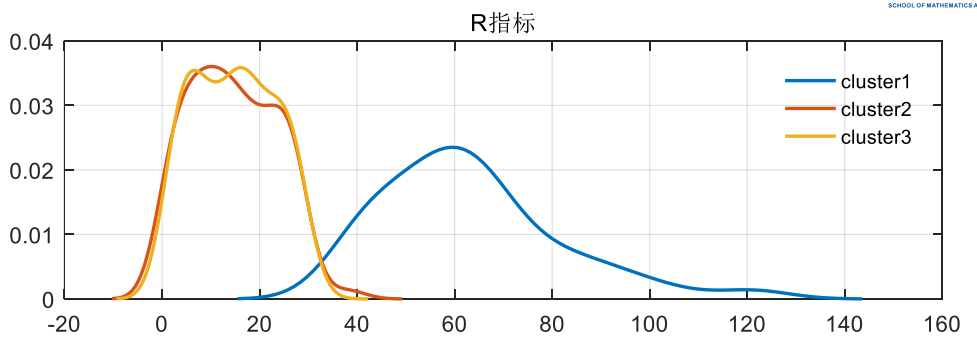
```
[customer,text] = xlsread('consumption_data.xls');
customer(:,1) = [];
cons = zscore(customer);
opts = statset('Display','final');
[idx,C] = kmeans(cons,3,'Replicates',5,'Options',opts); %欧氏距离
cus1 = customer(idx == 1,:);  cus2 = customer(idx == 2,:);
cus3 = customer(idx == 3,:);
[varnames] = text(1,2:4);
for i = 1:3
    subplot(3,1,i);  [f1,xi1]=ksdensity(cus1(:,i));
    [f2,xi2]=ksdensity(cus2(:,i));  [f3,xi3]=ksdensity(cus3(:,i));
    plot(xi1,f1,xi2,f2,xi3,f3,'LineWidth',1);  grid on
    legend('cluster1','cluster2','cluster3'); legend('boxoff')
    title(strcat(varnames{i},'指标'))
end
```

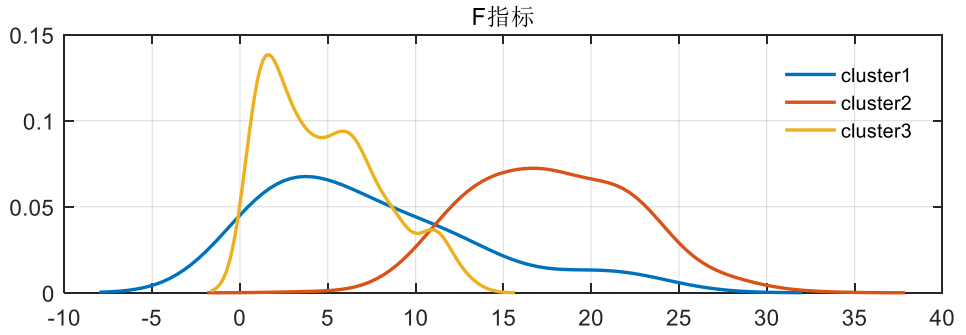


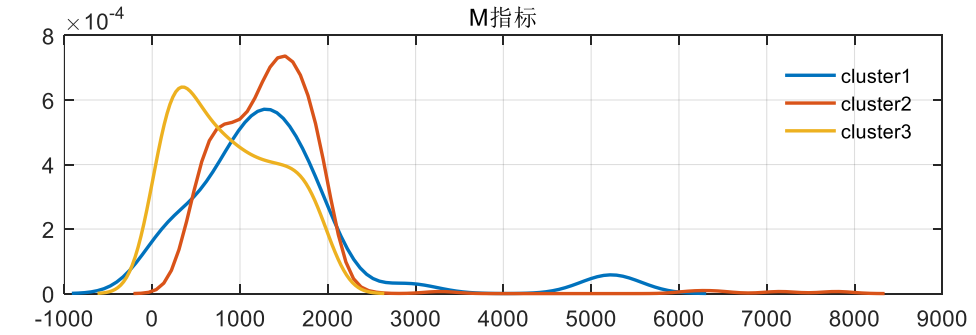
程序运行两次，得到的结果基本一致.

案例分析

- 客户价值分析如下：
  - 分群1特点：R间隔相对较大，主要集中在30~80天；消费次数集中在0~15次，消费金额在0~2000；
  - 分群2特点：R间隔相对较小，主要集中在0~30天；消费次数集中在10~25次，消费金额在500~2000；
  - 分群3特点：R间隔相对较小，主要集中在0~30天；消费次数集中在0~12次，消费金额在0~1800；
- 对比分析
  - 分群1的时间间隔较长，消费次数较少，消费金额也不是特别高，是价值较低的客户群体。
  - 分群2时间间隔较短，消费次数多，而且消费金额较大，是高消费、高价值人群；
  - 分群3时间间隔、消费次数和消费金额处于中等水平，代表着一般客户；







4. K-Mediods （K中心点）聚类

K-中心点算法与K-均值算法在原理上十分相近，它是针对于K-均值算法易受极值影响这一缺点的改进算法，即在含有离群点的情况下，K-Mediods的鲁棒性（稳定性）要更好。在原理上的差异在于选择各类别中心点时不取样本均值点，而是类别内选取到其余样本距离之和最小的样本为中心。

K-mediods算法描述

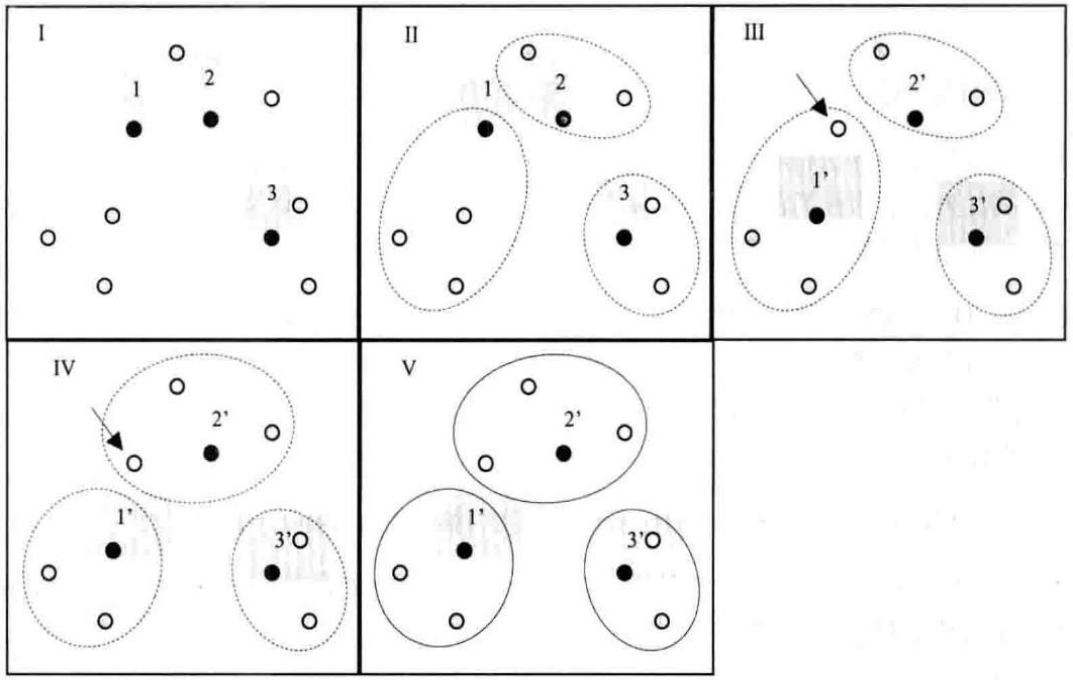
① 首先随机选取一组聚类样本作为中心点集

② 每个中心点对应一个簇

③ 计算各样本点到各个中心点的距离（如欧几里德距离），将样本点放入距离中心点最短的那个簇中

④ 计算各簇中，距簇内各样本点距离的绝度误差最小的点，作为新的中心点

⑤ 如果新的中心点集与原中心点集相同，算法终止；如果新的中心点集与原中心点集不完全相同，返回②。



4. K-Mediods （K中心点）聚类

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

`idx = kmedoids(X,k)` performs k-medoids Clustering to partition the observations of the n-by-p matrix `X` into `k` clusters, and returns an n-by-1 vector `idx` containing cluster indices of each observation. Rows of `X` correspond to points and columns correspond to variables. By default, `kmedoids` uses squared Euclidean distance metric and the k-means++ algorithm for choosing initial cluster medoid positions.

`idx = kmedoids(X,k,Name,Value)` uses additional options specified by one or more `Name,Value` pair arguments.

`[idx,C] = kmedoids( ___ )` returns the `k` cluster medoid locations in the k-by-p matrix `C`.

`[idx,C,sumd] = kmedoids( ___ )` returns the within-cluster sums of point-to-medoid distances in the k-by-1 vector `sumd`.

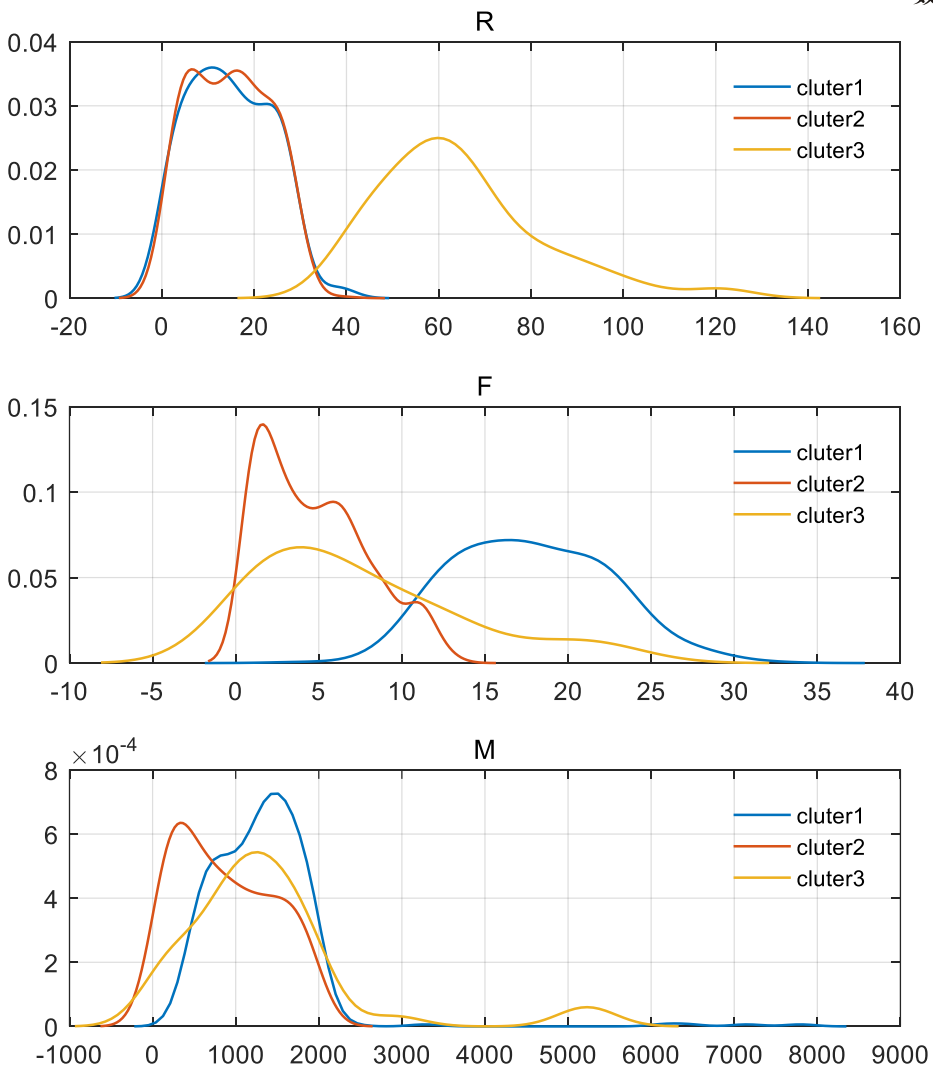
`[idx,C,sumd,D] = kmedoids( ___ )` returns distances from each point to every medoid in the n-by-k matrix `D`.

`[idx,C,sumd,D,midx] = kmedoids( ___ )` returns the indices `midx` such that `C = X(midx,:)`. `midx` is a k-by-1 vector.

`[idx,C,sumd,D,midx,info] = kmedoids( ___ )` returns a structure `info` with information about the options used by the algorithm when executed.

4. K-Mediods （K中心点）聚类

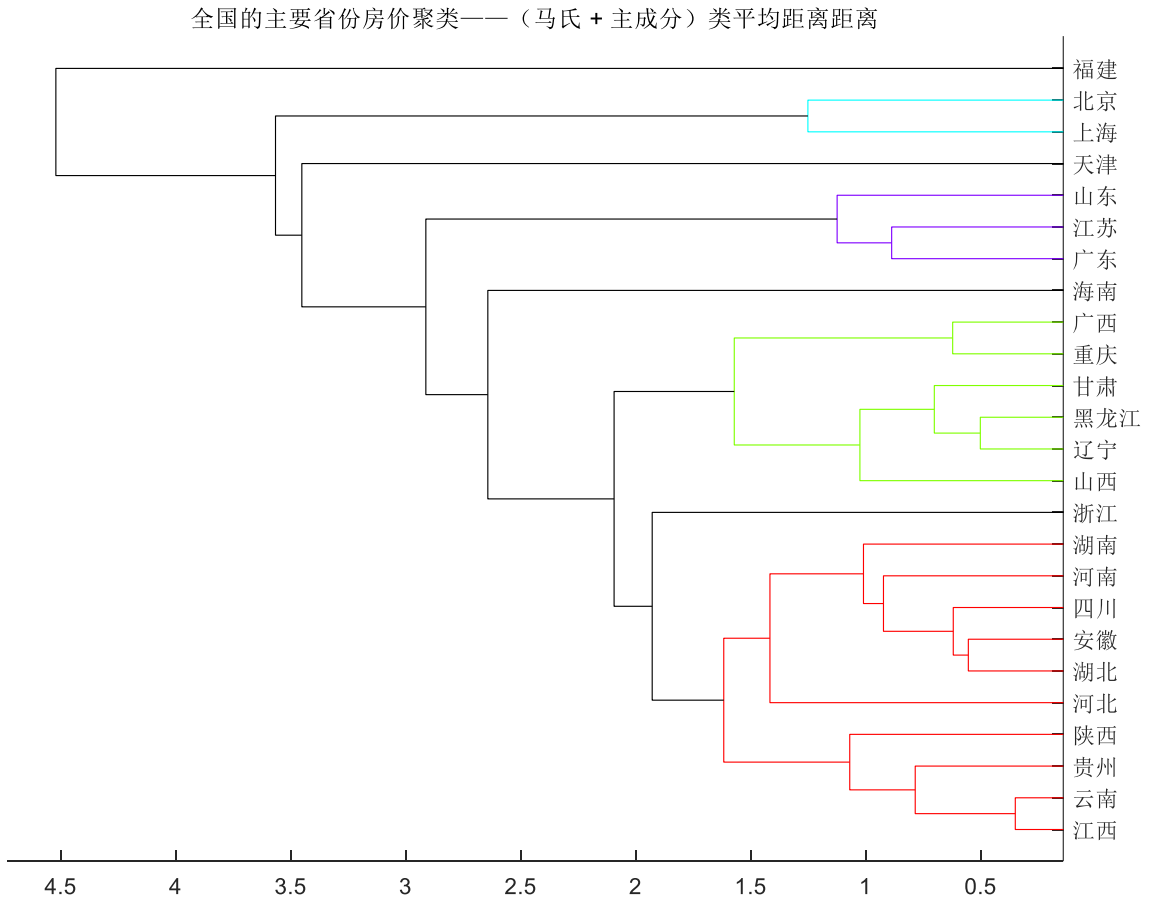
```
[cons,text] = xlsread('consumption_data.xls');
cons(:,1) = [];  czs = zscore(cons);
[ind,C,sumD,D,midx,info] = kmedoids(czs,3);
cus1 = cons(ind == 1,:);  cus2 = cons(ind == 2,:);
cus3 = cons(ind == 3,:);  varnames = text(1,2:4);
for i = 1:3
    subplot(3,1,i)
    [f1,xi1] = ksdensity(cus1(:,i));
    [f2,xi2] = ksdensity(cus2(:,i));
    [f3,xi3] = ksdensity(cus3(:,i));
    plot(xi1,f1,xi2,f2,xi3,f3,'LineWidth',1); grid on
    legend('cluter1','cluter2','cluter3');  legend('boxoff')
    title(varnames{i})
end
```



4. K-Mediods （K中心点）聚类

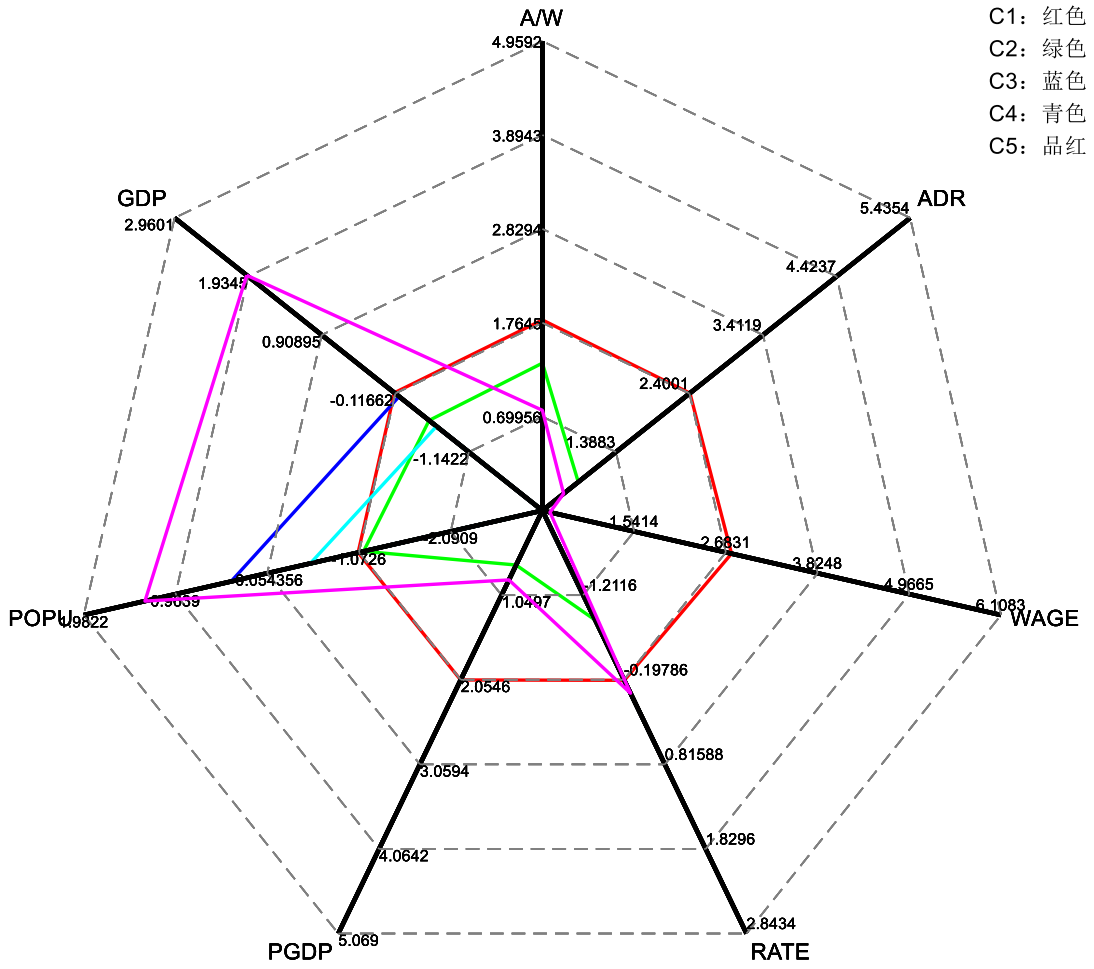
```
[sale,text] = xlsread('sale_houseprice.xlsx');
salezs = zscore(sale);
[ind,C,sumD,D] = kmedoids(salezs,5);
sa1 = sale(ind == 1,:);
sa2 = sale(ind == 2,:);
sa3 = sale(ind == 3,:);
sa4 = sale(ind == 4,:);
sa5 = sale(ind == 5,:);
varnames = text(2,2:8);
rownames = text(3:end,1);
res = cell(25,1);
res(:,1) = rownames;
res(:,2) = num2cell(ind);
res = sortrows(res,2)
```

```
res =
  25×2 cell 数组
    '上海'      [1]
    '北京'      [1]
    '福建'      [2]
    '海南'      [2]
    '天津'      [2]
    '河北'      [3]
    '湖北'      [3]
    '四川'      [3]
    '河南'      [3]
    '安徽'      [3]
    '陕西'      [3]
    '江西'      [3]
    '云南'      [3]
    '湖南'      [3]
    '贵州'      [3]
    '山西'      [4]
    '辽宁'      [4]
    '黑龙江'    [4]
    '甘肃'      [4]
    '重庆'      [4]
    '广西'      [4]
    '广东'      [5]
    '山东'      [5]
    '江苏'      [5]
    '浙江'      [5]
```



4. K-Mediods （K中心点）聚类

```
varnames = {'GDP','POPU','PGDP','RATE','WAGE','ADR','A/W'};
sa1 = salezs(ind == 1,:);
sa2 = salezs(ind == 2,:);
sa3 = salezs(ind == 3,:);
sa4 = salezs(ind == 4,:);
sa5 = salezs(ind == 5,:);
min1 = min(sa1);
max1 = max(sa1);
lim = [min1-2;max1+3]';
prefer_range = [min1-1; max1+1]';
draw_radar(mean(sa1),lim,prefer_range,varnames,'r')
draw_radar(mean(sa2),lim,prefer_range,varnames,'g')
draw_radar(mean(sa3),lim,prefer_range,varnames,'b')
draw_radar(mean(sa4),lim,prefer_range,varnames,'c')
draw_radar(mean(sa5),lim,prefer_range,varnames,'m')
```




01
IDEA
02
LEARNING
03
CREATIVE
INSPIRATION
EDUCATION
BUSINESS
PHOTO BOOK

感谢聆听