聚类分析

什么是聚类分析

1. 聚类是一个将数据集划分为若干组(class)或类(cluster)的过程，并使得同一组内的数据对象具有较高的相似度，而不同组不相似。
2. 相似与否是基于数据描述属性的取值来确定的，通常利用各数据对象间的距离来进行表示。
3. 聚类分析适用于探讨样本间相互关联关系从而对一个样本结构做一个初步的评价。
4. 无监督学习。
5. 分为两种，Q型(核心)：对样品分类；R型：对变量(指标)分类，可以了解变量、变量组合间的亲疏程度，也即降维处理。

样本间相似度的度量

1.距离

知道欧式距离就行。k维下每个维度上距离平方的根号求和。
各距离及MATLAB命令

1.欧氏距离 $d(x_i,x_j)=[\sum_{k=1}^{p}(x_{ik}-x_{jk})^2]^{1/2}$ **pdist(x)**

2.绝对距离 $d(x_i,x_j)=\sum_{k=1}^{p}|x_{ik}-x_{jk}|$ **pdist(x,'cityblock')**

3.明氏距离 $d(x_i,x_j)=[\sum_{k=1}^{p}|x_{ik}-x_{jk}|^m]^{1/m}$ **pdist(x,'minkowski',r)**

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

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

将原数据标准化以后的欧氏距离

6.马氏距离 $d(x_i,x_j)=\sqrt{(x_i-x_j)^T\Sigma^{-1}(x_i-x_j)}$ **pdist(x,'mahal')**

式中Σ^{-1}为向量x和y的协方差矩阵的逆矩阵。

7. 兰氏距离 $d(x_i,x_j)=\frac{1}{p}\sum_{k=1}^{p}\frac{|x_{ik}-x_{jk}|}{x_{ik}+x_{jk}}$

8. 杰氏距离（Jffreys & Matusita）

$$d(x_i,x_j)=[\sum_{k=1}^{p}(\sqrt{x_{ik}}-\sqrt{x_{jk}})^2]^{1/2}$$

例如:

例1.为了研究辽宁、浙江、河南、甘肃、青海5省1991年城镇居民生活消费规律，需要利用调查资料对五个省进行分类，指标变量共8个，意义如下：x1:人均粮食支出，x2:人均副食支出;x3:人均烟酒茶支出，x4:人均其他副食支出,x5:人均衣着商品支出,x6:人均日用品支出，x7:人均燃料支出，x8人均非商品支出

表1 1991年五省城镇居民生活月均消费（元/人）

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8
辽宁	7.9	39.77	8.49	12.94	19.27	11.05	2.04	13.29
浙江	7.68	50.37	11.35	13.3	19.25	14.59	2.75	14.87
河南	9.42	27.93	8.2	8.14	16.17	9.42	1.55	9.76
甘肃	9.16	27.98	9.01	9.32	15.99	9.1	1.82	11.35
青海	10.06	28.64	10.52	10.05	16.18	8.39	1.96	10.81

计算各省之间的欧氏、绝对、明氏距离

解：a=[7.9 39.77 8.49 12.94 19.27 11.05 2.04 13.29
7.68 50.37 11.35 13.3 19.25 14.59 2.75 14.87
9.42 27.93 8.2 8.14 16.17 9.42 1.55 9.76
9.16 27.98 9.01 9.32 15.99 9.1 1.82 11.35
10.06 28.64 10.52 10.05 16.18 8.39 1.96 10.81];

d1=pdist(a);% 此时计算出各行之间的欧氏距离，

为了得到书中的距离矩阵，我们键入命令：

D= squareform(d1), % 注意此时d1必须是一个行向量，结果是实对称矩阵

若想得到书中的三角阵，则有命令：

S = tril(squareform(d1))

计算出各行间的距离后需转换成三角阵，便于查看两两间的距离。

```
S =     0        0        0        0        0
   11.6726       0        0        0        0
   13.8054  24.6353       0        0        0
   13.1278  24.0591   2.2033       0        0
   12.7983  23.5389   3.5037   2.2159       0
```

d2=pdist(a,'cityblock'); S2 = tril(squareform(d2))

```
S2 =       0        0        0        0        0
          19.89     0        0        0        0
          27.2    47.05      0        0        0
          24.58   43.39    4.66       0        0
          26.52   42.31    8.08     5.38       0
```

d3=pdist(a,'minkowski',3); S3 = tril(squareform(d3))

2.相似系数

用于对指标进行聚类。最常用的是相关系数和夹角余弦。相似系数需满足：

若用 $C_{\alpha,\beta}$ 表示变量之间的相似系数，则应满足：

$$|C_{\alpha\beta}| \leq 1, \text{且 } C_{\alpha\alpha} = 1$$

$$C_{\alpha\beta} = \pm 1, \text{当且仅当 } \boldsymbol{\alpha} = k\boldsymbol{\beta}, k \neq 0$$

$$C_{\alpha\beta} = C_{\beta\alpha}$$

1. 夹角余弦

两变量的夹角余弦定义为：

$$C_{ij}(1) = \cos\alpha_{ij} = \frac{\sum_{t=1}^{n} x_{ti}x_{tj}}{\sqrt{\sum_{t=1}^{n} x_{ti}^2}\sqrt{\sum_{t=1}^{n} x_{tj}^2}}$$

2. 相关系数

两变量的相关系数定义为：

$$C_{ij}(2) = \frac{\sum_{t=1}^{n} (x_{ti} - \bar{x}_i)(x_{tj} - \bar{x}_j)}{\sqrt{\sum_{t=1}^{n} (x_{ti} - \bar{x}_i)^2}\sqrt{\sum_{t=1}^{n} (x_{tj} - \bar{x}_j)^2}}$$

二者MATLAB命令：

例3.计算例1中各指标之间的相关系数与夹角余弦

```
解：a=[7.9      39.77   8.49    12.94   19.27   11.05   2.04 13.29
7.68    50.37   11.35   13.3    19.25   14.59   2.75    14.87
9.42    27.93   8.2     8.14    16.17   9.42    1.55  9.76
9.16    27.98   9.01    9.32    15.99   9.1     1.82    11.35
10.06   28.64   10.52   10.05   16.18   8.39    1.96    10.81];
```

```
R=corrcoef(a);% 指标之间的相关系数
a1=normc(a); % 将a的各列化为单位向量
J=a1'*a1  % 计算a中各列之间的夹角余弦
```

```
J =
   1.0000    0.9410    0.9847    0.9613    0.9824    0.9546    0.9620    0.9695
   0.9410    1.0000    0.9782    0.9939    0.9853    0.9977    0.9947    0.9935
   0.9847    0.9782    1.0000    0.9859    0.9911    0.9840    0.9931    0.9909
   0.9613    0.9939    0.9859    1.0000    0.9944    0.9919    0.9947    0.9981
   0.9824    0.9853    0.9911    0.9944    1.0000    0.9901    0.9901    0.9968
   0.9546    0.9977    0.9840    0.9919    0.9901    1.0000    0.9952    0.9953
   0.9620    0.9947    0.9931    0.9947    0.9901    0.9952    1.0000    0.9968
   0.9695    0.9935    0.9909    0.9981    0.9968    0.9953    0.9968    1.0000
```

谱系聚类法

1. 步骤

1. 选择样本间距离的定义及类间距离的定义；
2. 计算n个样本两两之间的距离，得到距离矩阵 $D=(d_{ij})$
3. 构造个类，每类只含有一个样本；
4. 合并符合类间距离定义要求的两类为一个新类；
5. 计算新类与当前各类的距离。若类的个数为1，则转到步骤6，否则回到步骤4；
6. 画出聚类图；
7. 决定类的个数和类。

（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

$$D_0 = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nn} \end{bmatrix}$$

（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Gp,Gq合并成一个新类Gr=（Gp,Gq），在D_0中去掉Gp,Gq所在的两行、两列，并加上新类与其余各类之间的距离(或相似系数)，得到n-1阶矩阵D1。

（3）从D1出发重复步骤（2）的做法得到D2，再由D2出发重复上述步骤，直到所有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。

例子:

例：为了研究辽宁等5省1991年城镇居民生活消费情况的分布规律，根据调查资料做类型分类，用最短距离做类间分类。数据如下：

	x1	x2	x3	x4	x5	x6	x7	x8
辽宁1	7.90	39.77	8.49	12.94	19.27	11.05	2.04	13.29
浙江2	7.68	50.37	11.35	13.30	19.25	14.59	2.75	14.87
河南3	9.42	27.93	8.20	8.14	16.17	9.42	1.55	9.76
甘肃4	9.16	27.98	9.01	9.32	15.99	9.10	1.82	11.35
青海5	10.06	28.64	10.52	10.05	16.18	8.39	1.96	10.81

将每一个省区视为一个样品，先计算5个省区之间的欧式距离，用D_0表示距离矩阵（对称阵，故给出下三角阵）

$$D_0 = \begin{array}{c} \\ \text{辽宁}\ 1 \\ \text{浙江}\ 2 \\ \text{河南}\ 3 \\ \text{甘肃}\ 4 \\ \text{青海}\ 5 \end{array} \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & & & & \\ 11.67 & 0 & & & \\ 13.80 & 24.63 & 0 & & \\ 13.12 & 24.06 & 2.20 & 0 & \\ 12.80 & 23.54 & 3.51 & 2.21 & 0 \end{pmatrix}$$

因此将3.4合并为一类，为类6，替代了3、4两类

类6与剩余的1、2、5之间的距离分别为：

$d(3,4)_1=\min(d31,d41)=\min(13.80,13.12)=13.12$

$d(3,4)_2=\min(d32,d42)=\min(24.63,24.06)=24.06$

$d(3,4)_5=\min(d35,d45)=\min(3.51,2.21)=2.21$

注意，合并两类为新类时，更新新类与剩余类的距离的方式。即取最小类间距离。

- **得到新矩阵**

$$D_1 = \begin{array}{c} \\ G6 \\ G1 \\ G2 \\ G5 \end{array} \begin{pmatrix} G6 & G1 & G2 & G5 \\ 0 & & & \\ 13.12 & 0 & & \\ 24.06 & 11.67 & 0 & \\ 2.21 & 12.80 & 23.54 & 0 \end{pmatrix}$$

合并类6和类5，得到新类7

- **类7与剩余的1、2之间的距离分别为：**

  $d(5,6)_1=\min(d51,d61)=\min(12.80,13.12)=12.80$

  $d(5,6)_2=\min(d52,d62)=\min(23.54,24.06)=23.54$

得到新矩阵

$$D_2 = \begin{Bmatrix} & G7 & G1 & G2 \\ G7 & 0 & & \\ G1 & 12.80 & 0 & \\ G2 & 23.54 & 11.67 & 0 \end{Bmatrix}$$

合并类1和类2，得到新类8

此时，我们有两个不同的类：类7和类8。

它们的最近距离

d(7,8) =min(d71,d72)=min(12.80,23.54)=12.80

得到矩阵

$$D_3 = \begin{Bmatrix} & G7 & G8 \\ G7 & 0 & \\ G8 & 12.80 & 0 \end{Bmatrix}$$

最后合并为一个大类。这就是按最短距离定义类间距离的系统聚类方法。最长距离法类似！

MATLAB命令

1. 输入矩阵，用上述距离命令计算各样品间的距离。

2. 选择类间距离进行聚类

(3) 选择不同的类间距离进行聚类

最短距离：z1= linkage(d)　% 此处及以下的d都（2）中算出的距离行向量
最长距离：z2= linkage(d,'complete')
中间距离：z3= linkage(d,'centroid')
重心距离：z4= linkage(d,'average')
离差平方和：z5= linkage(d,'ward')

3. 根据类间距离做出谱系聚类图，并根据图得出聚类结果。

（4）作出谱系聚类图

H=dendrogram(z,d) % 注意若样本少于30，可以省去d，否则必须填写.

（5）根据分类数目，输出聚类结果

T=cluster(z,k) % 注意k是分类数目，z是（3）中的结果

Find(T==k0) % 找出属于第k0类的样品编号

例如：

作谱系聚类图：H= dendrogram(z1)

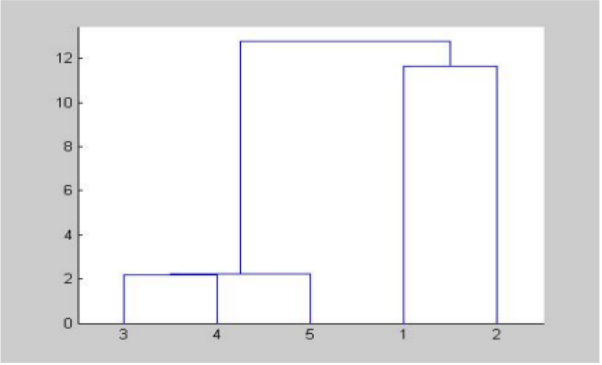


图2.最短距离聚类图

% 输出分类结果

T=cluster(z1,3)

结果表明：若分为三类，则辽宁是一类，浙江是一类，河南、青海和甘肃是另一类。

K-评价聚类(k-means)

1. 以k为参数，将n个对象分为k个簇，使得簇内对象具有较高的相似度。相似度的计算根据一个簇中对象的平均值(重心)。

K–平均聚类算法

（1）*k–means* 算法

算法 6.1：根据聚类中的均值进行聚类划分的 *k–means* 算法。

输入：聚类个数 k，以及包含 n 个数据对象的数据库。

输出：满足方差最小标准的 k 个聚类。

处理流程：

（1） 从 n 个数据对象任意选择 k 个对象作为初始聚类中心；

（2） 循环（3）到（4）直到每个聚类不再发生变化为止

（3） 根据每个聚类对象的均值（中心对象），计算每个对象与这些中心对象的距离；并根据最小距离重新对相应对象进行划分；

（4） 重新计算每个（有变化）聚类的均值（中心对象）

2. 特点

- 只适合聚类均值有意义的场合；
- 用户须事先指定k的个数；
- 对噪声和孤立点数据敏感，少量的该类数据能对聚类均值造成很大影响。

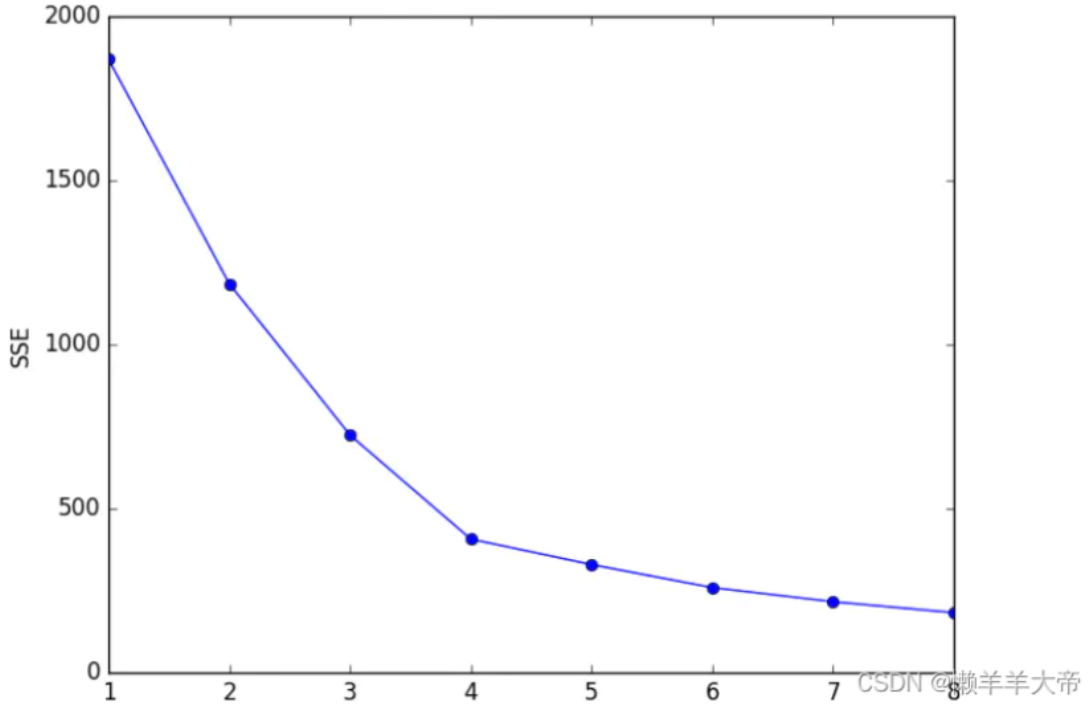
k-means的优化

1. 使用多次的随机初始化，计算每一次的代价函数值，取最小代价为类结果。

2. 肘部分析法选择k的值：

k-means是以最小化样本与质点平方误差作为目标函数，将每个簇的质点与簇内样本点的平方距离误差和**(sum of the squared errors，SSE,误差平方和)**称为畸变程度(distortions)，那么，对于一个簇，它的畸变程度越低，代表簇内成员越紧密，畸变程度越高，代表簇内结构越松散。

当k小于真实聚类数时，由于k的增大会大幅增加每个簇的聚合程度，故SSE的下降幅度会很大，而当k到达真实聚类数时，**再增加k所得到的聚合程度回报会迅速变小**，所以SSE的下降幅度会骤减，然后随着k值的继续增大而趋于平缓，也就是说SSE和k的关系图是一个手肘的形状，而这个肘部对应的k值就是数据的真实聚类数.



例如上图，在k=5时，相较于k=4时，畸变程度的变化急剧减小，所以，在k=4这点就是最佳的k值

matlab代码：

```
clc;
clear;
load fisheriris

data = meas;
%对原始数据进行归一化处理
data=mapminmax(meas,0,1);
%n是样本数，p为特征维数，k为分类数
[n,p]=size(data);

K=8;D=zeros(K,2);
for k=2:K

[lable,c,sumd,d]=kmeans(data,k,'dist','sqeuclidean');
% data，n×p原始数据向量
% lable，n×1向量，聚类结果标签；
% c，k×p向量，k个聚类质心的位置
% sumd，k×1向量，类间所有点与该类质心点距离之和
% d，n×k向量，每个点与聚类质心的距离
sse1 = sum(sumd.^2);
D(k,1) = k;
D(k,2) = sse1;
end

plot(D(2:end,1),D(2:end,2))
hold on;
plot(D(2:end,1),D(2:end,2),'or');

title('不同K值聚类偏差图')
xlabel('分类数(K值)')
ylabel('簇内误差平方和')

```