中国多个城市空气质量的数据分析

1. 系统聚类

1.1. 读入数据

In [ ]:

```
####读入数据####
x=read.table("C:/Users/bff/Desktop/me-ppt/air.txt",header=T,fileEncoding = "GB18030")
x=as.matrix(x)
rownames(x)=c("北京","天津","石家庄","太原","呼和浩特","沈阳","长春","哈尔滨",
"上海","南京","杭州","合肥","福州","南昌","济南","郑州","武汉","长沙","广州",
"南宁","海口","重庆","成都","贵阳","昆明","拉萨","西安","兰州","西宁","银川","乌鲁木齐")

#注意此处特别给数据加行名称，而不是直接读入所有变量信息，是希望后面展示聚类结果时，可以直接展示

####描述分析####
summary(x)
dim(x)

```

注意：

该数据中包含DUST,SO2,NO2,DAYS,变量含义如下：单位面积内空气中可吸入颗粒物多少(μg/m3)，二氧化硫(μg/m3)、二氧化氮含量(μg/m3)，空气质量达到及好于二级的天数四个变量，注意影响一个城市空气质量分组的因素有很多，这里简单的列出了四个，特别是最后一个，与前三列变量相比，量纲差异较大。遇到数据量纲差异较大的问题时，需要对数据进行标准化

1.2 计算距离

In [ ]:

```
d=dist(scale(x)) # dist默认欧氏距离，这里用欧氏距离，是因为数据已经实施了标准化处理
```

1.3 系统聚类

In [ ]:

```
####用不同联接方法聚类####
hc1=hclust(d,"single")   #最短距离法
hc2=hclust(d,"complete")#最长距离法，R中默认的联接方法
hc3=hclust(d,"ward.D")   #ward法
hc4=hclust(d,"centroid")#重心法
```

1.4. 绘制谱系图

In [ ]:

```
####绘制谱四种联结方法得到的谱系图####
par(mfrow=c(1,1))#设置画布，如可以绘制2*2的组图。
par(family='STKaiti')#设置字体
plot(hc1,hang=-1)#最短距离法的谱系图
#cutree(hc1, k=3)
rect.hclust(hc1,k=3)#画出对应的图
```

In [ ]:

```
plot(hc2,hang=-1,family='STKaiti')#最长距离法的谱系图
plot(hc3,hang=-1)#ward法的谱系图
plot(hc4,hang=-1)#重心法的谱系图
```

2. k-mean聚类

2.1 读入数据

In [ ]:

```
#这里使用的数据和系统聚类的案例数据一致，因此读入数据、描述数据的步骤可省略

#x=read.table("air.txt",header=T,fileEncoding = "GB18030")
#...
#summary(x)
```

2.2 k均值聚类

In [ ]:

```
cl=kmeans(x,3,20) # 聚为3类，最大迭代次数20
```

2.3 聚类结果展示

In [ ]:

```
cl # 展示K均值聚类的主要结果
```

In [ ]:

```
#### 也可以单独展示部分结果####
cl$cluster #展示每个样本观测属于哪一类
```

In [ ]:

```
cl$size# 每一类大小都是多少
```

In [ ]:

```
#par(family='STKaiti')#设置字体
plot(x,col=cl$cluster,pch=2,lwd=1) ### 用前两个变量“可吸入颗粒物”、“二氧化硫”绘制散点图，并
```

In [ ]:

```
pairs(x,col=cl$cluster,pch=2,lwd=1)#绘制两两不同的散点图矩阵
```

2.4 聚类结果评价---SSE （Sum of squared errors）平方误差和

In [ ]:

```
# 开始与结果边界
begin = 1;
length = 15;
count = 50;
end = begin + length - 1;

# 结果容器
result = c();
result[begin:end] = 0;

# 遍历计算kmeans的SSE
for(i in begin:end) {
    # 计算SSE
    tmp = c();
    tmp[1:count] = 0;
    for(j in 1:count) {
        kcluster = kmeans(x, i);
        tmp[j] = kcluster$tot.withinss;
    }
    result[i] = mean(tmp);
}

# 绘制结果
plot(result, type="o", xlab="Number of Cluster", ylab="Sum of Squer Error");
```

此脚本按照K从1到15，计算不同的聚类的SSE，由于kmeans算法中的随机因数，每次结果都不一样，为了减少时间结果的偶然性，对于每个k值，都重复运行50次，求出平均的SSE，最后绘制出SSE曲线，如下所示：

2.5 聚类结果评价---计算Silhouette Coefficient

In [ ]:

```
# 开始与结果边界
begin = 2;
length = 15;
count = 50;
end = begin + length - 1;

# 结果容器
result = c();
result[begin:end] = -1;

# 遍历计算kmeans的SSE
library(cluster);
for(i in begin:end) {
    # Silhouette coefficient
    tmp = c();
    tmp[1:count] = 0;
    for(j in 1:count) {
        kcluster = pam(x, i);
        tmp[j] = kcluster$silinfo$avg.width;
    }
    result[i] = mean(tmp);
}

# 绘制结果
plot(result, type="o", xlab="Number of Cluster", ylab="Silhouette Coefficient");

```

K从2到15（k=1时无法计算），重复执行50次，得到结果如上图。

In [ ]:

```

```