2018年12月5日

实验二任务说明：

1. 采用K均值算法对给定数据集进行聚类，给出聚类结果。改变K的不同取值，研究K值改变给聚类结果所带来的变化。改变初始簇心，研究簇心变化给聚类结果所带来的变化。SPSS和WEKA中均有K-均值算法的实现，可自由选择你所熟悉的工具完成本实验。
2. 请将你的聚类结果和相应分析说明写在实验报告中（Word，格式不限），并在今天下课前发至邮箱：[buptssedwkd2018@163.com](mailto:buptssedwkd2018@163.com)

# 使用python实现聚类实验

1. 数据归一化

def normalizations(df):

return df.apply(lambda x: (x - np.min(x)) / (np.max(x) - np.min(x)))

1. 同时发现数据中592行数据有缺失，将它删去
2. 使用scikit-learn 机器学习包中的kmeans算法计算

def k\_means(df):

max = 0

index = 0

df = normalizations(df)

for i in range(3, 16):

y\_pred = KMeans(n\_clusters=i).fit\_predict(df)

r= metrics.calinski\_harabaz\_score(df, y\_pred)

print(f'k为{i},评分为{r}')

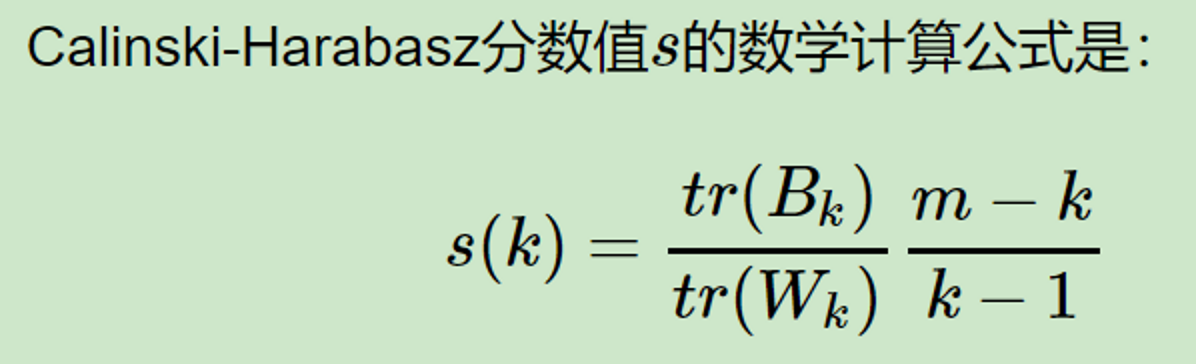
if r > max:

max = r

index = i

print(f'最优时，k为{index},评分为{max}')

1. 使用calinski\_harabaz评价聚类结果他可以从簇内的稠密程度和簇间的离散程度来评估聚类的效果。



1. 输出结果
2. 因此，当k为4时，聚类效果最好

k为3,评分为3473.5486295563533

k为4,评分为3730.5767271906952

k为5,评分为3354.9089637074076

k为6,评分为3409.4564710202917

k为7,评分为3393.4306292366223

k为8,评分为3387.0017586198787

k为9,评分为3373.187488926511

k为10,评分为3360.211786616531

k为11,评分为3172.4831732169378

k为12,评分为3061.7701496409136

k为13,评分为2962.7315137480814

k为14,评分为2883.4755385386284

k为15,评分为2804.833517635444

最优时，k为4,评分为3730.5767271906952

1. 通过修改聚类初始点，观测对于calinski\_harabaz分数的变化。
2. 输出结果

def change\_init(df):

y\_pred = KMeans(n\_clusters=4, random\_state=9,init='k-means++').fit\_predict(df)

r = metrics.calinski\_harabaz\_score(df, y\_pred)

print(f'中心点方法：k-means++ 分数：{r}')

y\_pred = KMeans(n\_clusters=4, random\_state=9, init='random').fit\_predict(df)

r = metrics.calinski\_harabaz\_score(df, y\_pred)

print(f'中心点方法：random 分数：{r}')

中心点方法：k-means++ 分数：4563.351498217524

中心点方法：random 分数：2104.7693650359834

1. 观察到初始中心点选区方法不同，对于结果的影响也是巨大的。

传统的k-means算法随机选取中心点，而k-means算法对于初始点十分敏感。随机选取导致不同的结果。

而K-means++算法使初始的聚类中心之间的相互距离要尽可能的远，符合直观的感觉。

算法描述如下：

步骤一：随机选取一个样本作为第一个聚类中心 c1；

步骤二：

计算每个样本与当前已有类聚中心最短距离（即与最近一个聚类中心的距离），用 D(x)表示；

这个值越大，表示被选取作为聚类中心的概率较大；

最后，用轮盘法选出下一个聚类中心；

步骤三：重复步骤二，知道选出 k 个聚类中心。

1. 根据分类结果，给出若干图像：

