

# Principal Component Analysis

## Steps

1. normalized:

$$\begin{aligned}\bar{x}_j &= \frac{\sum_{i=1}^n x_{ij}}{n} \text{ (第} i \text{个指标的平均值)} \\ s_j^2 &= \frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}{n-1} \text{ (关于第} j \text{个指标的方差)} \\ z_{ij} &= \frac{x_{ij} - \bar{x}_j}{s_j} \text{ (} i=1 \dots n, j=1 \dots p \text{) ( 标准化的矩阵)}\end{aligned}$$

2. correlation coefficient matrix:

根据标准化矩阵求得相关系数矩阵：R=corrcoef(Z);

3. solve correlation coefficient matrix eigenvector and eigenvalue

(求出eigenvalue  $\lambda_i$  的值，同时将  $\lambda_i$  从大到小排序),然后利用：

$$\frac{\sum_{j=1}^m \lambda_j}{\sum_{j=1}^p \lambda_j} \geq 0.85$$

其中，0.85是贡献率，可以随意变动的值。

4. 再将eigenvector进行单位化：MATLAB中运用orth()即可。

5. 求出主成分：

$$Y = \text{成分}^T \times \text{单位特征向量}$$

6. 对m个主成分进行加权求和，然后进行分析。

## code

```
clc;
clear ;
close all;
houseFee_data=xlsread('C:\Users\subin\Desktop\houseFee.xlsx','C2:J31');
[city_number,variable_number]=size(houseFee_data);%求得所得矩阵的行数和列数
normalized_houseFee_data=zeros(city_number,variable_number);%预定义标准矩阵
%对数据进行了标准化
for i=1:variable_number
    normalized_houseFee_data(:,i) =(houseFee_data(:,i)-...
        mean(houseFee_data(:,i)))/std(houseFee_data(:,i));
end
```

```

%根据标准化矩阵求得相关系数矩阵
houseFee_corrcoef_matrix=corrcoef(normalized_houseFee_data);
%求出特征向量eigenvectors and eigenvalues
[eigenVectors,eigenvalues]=eig(houseFee_corrcoef_matrix);
%将eigenvalues 放到一个 vector
eigenValues=size(variable_number,1);
for i=1:variable_number
    eigenValues(i,1)=eigenvalues(i,i);
end
%将eigenVector 进行标准正交化
eigenVectors=orth(eigenVectors);
%将eigenvalues进行降序排列
eigenValues=sort(eigenValues,'descend');
%计算排序后的主成分的贡献率和累计的贡献率
contribution_Rate_EachEigenValue=zeros(variable_number,1);
accumulate_contribution_rate=zeros(variable_number,1);
for k=1:variable_number
    contribution_Rate_EachEigenValue(k,1)=eigenValues(k,1)/sum(eigenValues);
    accumulate_contribution_rate(k,1)=sum(eigenValues(1:k,1))/sum(eigenValues);
end
%主成分信息的保留率
praInformationRatio=0.85;
%统计在满足主成分分析的数量个数
for i=1:variable_number
    if accumulate_contribution_rate(i,1)>=praInformationRatio
        main_component_number=i;
        break;
    end
end
%截取主成分对应的特征向量
mainComponentVector=zeros(variable_number,main_component_number);
for i=1:main_component_number
    mainComponentVector(:,i)=eigenVectors(:,variable_number+1-i);
end
%计算打分矩阵
scoreMatrix=normalized_houseFee_data*mainComponentVector;
%计算各个公司的总得分
for i=1:city_number
    for j=1:main_component_number
        city_sum_score(i,1)=scoreMatrix(i,j)*contribution_Rate_EachEigenValue(j,1);
        city_sum_score(i,2)=i;
    end
end
%按第一列总分数进行排序
sort_city_sum_score=sortrows(city_sum_score,-1);

```

**注：**

1. 对于像贡献率这样的常用数字，最好使用一个单词常量来进行表示。
2. 截取对应主成分的单位特征向量：`eigenVectors(:,variable_number+1-i);`
3. 标准化过程时既可以像上边那样进行，也可以用函数`zscore(x)`来进行标准化。