|  |  |
| --- | --- |
| 姓名 |  |
| 评分 |  |

## 实验报告

《数学建模》实验报告

课题名称： 数学建模层次分析法

专 业： 信息与计算科学

姓 名： 袁永润

班 级： 123212

完成日期： 2024年4月6日

### 实验报告

## 题目名称

评选中级职称——层次分析法

## 题目内容

## 现有一学校某系的三位助教(甲乙丙)，参加评选中级职称，评委由两位专家来组成，请你给出一种排序。 影响评审的因素有教龄、教学效果和科研成果。专家一认为，教龄的重要性比科研成果稍微高一些，教学效果的重要性比教龄高一些。专家二认为，教龄与教学效果同样重要，但都比科研成果高一些。 两位专家对三位助教各因素的评估如下： 教龄:专家一二一致认为甲比乙、丙高一些，乙、两一样。

## 教学效果:专家一认为乙比甲突出，丙比甲明显高而比乙稍高；专家二认为乙比甲明显高一一些，比丙稍高，丙比甲明显高一一点。

## 科研成果:专家一认为甲与丙一样，比乙稍高一些；专家二认为甲比乙稍高，比丙稍低一点，丙比乙明显高一些。

## 模型建立

3.1 明确问题

1.评价目标：对助教评职称进行排序；

2.评价准则：专家一、专家二；教龄、教学效果、科研成果；

3.选择方案：助教甲、助教乙、助教丙。

3.2 建立层次结构

三位助教的职称评估

目标层

专家一的判断

专家二的判断

准则层

科研成果

教学效果

教龄

助教丙

助教乙

助教甲

方案层

3.3 构建判断矩阵

构建的关系如下列表所示：

1.

|  |  |  |  |
| --- | --- | --- | --- |
| 专家一 | 教龄 | 教学效果 | 科研成果 |
| 教龄 | 1 | 1/4 | 4 |
| 教学成果 | 4 | 1 | 8 |
| 科研成果 | 1/4 | 1/8 | 1 |

**表1**

假设其为矩阵A1:

2.

|  |  |  |  |
| --- | --- | --- | --- |
| 专家二 | 教龄 | 教学效果 | 科研成果 |
| 教龄 | 1 | 1 | 4 |
| 教学成果 | 1 | 1 | 4 |
| 科研成果 | 1/4 | 1/4 | 1 |

**表2**

矩阵A2:

3.

|  |  |  |  |
| --- | --- | --- | --- |
| 教龄（专家一） | 甲 | 乙 | 丙 |
| 甲 | 1 | 4 | 4 |
| 乙 | 1/4 | 1 | 1 |
| 丙 | 1/4 | 1 | 1 |

**表3**

矩阵B1：

4.

|  |  |  |  |
| --- | --- | --- | --- |
| 教龄（专家二） | 甲 | 乙 | 丙 |
| 甲 | 1 | 4 | 4 |
| 乙 | 1/4 | 1 | 1 |
| 丙 | 1/4 | 1 | 1 |

**表4**

矩阵B2:

5.

|  |  |  |  |
| --- | --- | --- | --- |
| 教学成果（专家一） | 甲 | 乙 | 丙 |
| 甲 | 1 | 1/5 | 1/7 |
| 乙 | 5 | 1 | 1/3 |
| 丙 | 7 | 3 | 1 |

**表5**

矩阵B3:

6.

|  |  |  |  |
| --- | --- | --- | --- |
| 教学成果（专家二） | 甲 | 乙 | 丙 |
| 甲 | 1 | 1/7 | 1/6 |
| 乙 | 7 | 1 | 3 |
| 丙 | 6 | 1/3 | 1 |

**表6**

矩阵B4:

7.

|  |  |  |  |
| --- | --- | --- | --- |
| 科研成果（专家一） | 甲 | 乙 | 丙 |
| 甲 | 1 | 4 | 1 |
| 乙 | 1/4 | 1 | 1/4 |
| 丙 | 1 | 4 | 1 |

**表7**

矩阵B5:

8.

|  |  |  |  |
| --- | --- | --- | --- |
| 科研成果（专家二） | 甲 | 乙 | 丙 |
| 甲 | 1 | 3 | 1/3 |
| 乙 | 1/3 | 1 | 1/7 |
| 丙 | 3 | 7 | 1 |

**表8**

矩阵B6:

3.4 一致性检验

建立判断矩阵的维数为n=3,查表可得n=3时随机一致性指标RI=0.58，通过计算矩阵A1、A2、B1、B2、B3、B4、B5、B6的最大特征值，可得到该矩阵的一致性指标，以及随机一致性比率，通过判断是否小于0.1，由此判断矩阵是否通过一致性检验，计算结果如下表9：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A1 | A2 | B1 | B2 | B3 | B4 | B5 | B6 |
| CI | 0.0268 | -4.4409e-16 | -4.4409e-16 | -4.4409e-16 | 0.0324 | 0.0500 | -4.4409e-16 | 0.0035 |
| RI | 0.58 | 0.58 | 0.58 | 0.58 | 0.58 | 0.58 | 0.58 | 0.58 |
| CR | 0.0462 | -7.6567e-16 | -7.6567e-16 | -7.6567e-16 | 0.0559 | 0.0861 | -7.6567e-16 | 0.0061 |

**表9 甲乙丙职称评估的各个判断矩阵的一致性指标CI以及随机一致性比率RI**

CI：一致性指标CI

RI：随机一致性指标RI

CR：随机一致性比率CR

经过计算其中以上八个矩阵计算得出的CR都小于0.1，可判断以上矩阵均通过一致性检验。

3.5 计算权重

根据其方案计算出各个判断矩阵的最大特征值和特征系数如下表10：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A1 | A2 | B1 | B2 | B3 | B4 | B5 | B6 |
| 最大特征值 | 3.0542 | 3 | 3 | 3 | 3.0655 | 3.1013 | 3 | 3.0070 |
| 权重系数 | 0.2267  0.7015  0.0718 | 0.4444  0.4444  0.1111 | 0.6667  0.1667  0.1667 | 0.6667  0.1667  0.1667 | 0.0738  0.2828  0.6434 | 0.0694  0.6325  0.2981 | 0.4444  0.1111  0.4444 | 0.2431  0.0882  0.6687 |

**表10 计算三个准则分别对专家一二以及甲乙丙分别对三个准则的权重系数**

最后根据准则层权值以及方案层权值相乘相加得到如下表11：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 专家一准则层 | 教龄 | 教学效果 | 科研成果 | 总排序权值 |
| 准则层权值 | 0.2267 | 0.7015 | 0.0718 |  |
| 甲 | 0.6667 | 0.0738 | 0.4444 | 0.2348 |
| 乙 | 0.1667 | 0.2828 | 0.1111 | 0.2441 |
| 丙 | 0.1667 | 0.6434 | 0.4444 | 0.5210 |

**表11 专家一准则层综合权重计算**

再得到关于专家二的评估权重如表12;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 专家二准则层 | 教龄 | 教学效果 | 科研成果 | 总排序权值 |
| 准则层权值 | 0.4444 | 0.4444 | 0.1111 |  |
| 甲 | 0.6667 | 0.0694 | 0.2431 | 0.3541 |
| 乙 | 0.1667 | 0.6325 | 0.0882 | 0.3650 |
| 丙 | 0.1667 | 0.2981 | 0.6687 | 0.2808 |

**表12 专家二准则层综合权重计算**

3.6 结论

根据上述表格综合专家一和专家二的综合权重，计算得甲乙丙总排序权值分别为0.5889、0.6091、0.8018，由此可进行评估，助教甲要比助教乙好，助教乙比助教丙好。

## 四、 附录

|  |
| --- |
| 1.代码 |
| import numpy as np  # Input matrix  OC = np.array([[1, 1/4, 4], [4, 1, 8],[1/4, 1/8, 1]])  # Calculate eigenvalues and eigenvectors  D, V = np.linalg.eig(OC)  # Calculate the eigenvalue  lamda = D[0]  # Calculate consistency index (CI)  n = OC.shape[0]  CI = (lamda - n) / (n - 1)  # Random index (RI)  RI = 0.58  # Calculate consistency ratio (CR)  CR = CI / RI  # Print CI and CR  print("Consistency Index (CI):", CI)  print("Consistency Ratio (CR):", CR)  # Calculate weight vector  A = OC  n = A.shape[0]  # Step 1: Normalize the matrix by columns  A\_sum = np.sum(A, axis=0)  A\_sum = np.tile(A\_sum, (n, 1))  A\_ave = A / A\_sum  # Step 2: Sum the normalized matrix by rows  Weight = np.sum(A\_ave, axis=1)  # Normalize the summed vector to get the weight vector  Weight /= np.sum(Weight)  # Calculate the largest eigenvalue  Lamda = np.sum(np.dot(A, Weight) / Weight) / n  # Print weight vector and the largest eigenvalue  print("Weight Vector:", Weight)  print("Largest Eigenvalue (Lambda):", Lamda) |