

# 使用 Topsis 法分析河流的水质情况

## 一、问题的重述

有 20 条河流的水质情况需要评价，评价的指标有四个：含氧量(ppm)、PH 值、细菌总数(个/ml)和植物性营养物量(ppm),通过对其分析，对河流的水质情况进行评价排序。

## 二、模型假设

- 1、假设评价河流的水质情况时只考虑所给的四个指标
- 2、假设数据真实可靠

## 三、变量说明

符号	说明
$M$	目标层
$C$	准则层
$P$	方案层
$\lambda_{\max}$	最大特征值
$W_i$	特征向量
$\omega_i$	权重向量
$CI$	一致性指标
$CR$	一致性比例
$D_i^+$	第 i 个评价对象与最大值距离
$D_i^-$	第 i 个评价对象与最小值距离
$Z^+$	标准化矩阵中每一列的最大值组成的行向量
$Z^-$	标准化矩阵中每一列的最小值组成的行向量
$S_i$	第 i 个指标未归一化的得分

## 四、模型的建立与求解

### 4.1 问题分析

要评价 20 条河流的水质情况，首先需要使用层次分析法求出四个指标：含氧量、PH 值、细菌总数和植物性营养物量所占的权重。然后对四个指标所对应的数据进行标准化、正向化处理。最后使用 Topsis 法对每条河流进行评价，最终对评价结果进行排序即可求解问题。

### 4.2 模型的建立

#### 4.2.1 模型的原理

层次分析法，简称 AHP，是由美国匹兹堡大学教授 T.L.Satty 于 20 世纪 70 年代提出的一种多目标决策分析方法论<sup>[2]</sup>。其原理是将与决策有关的因素分解成目标层、准则层、方案层等若干层次，通过对各因素的计算和比较，得出不同因素的权重，为决策者选择最优方案提供参考依据。

Topsis 法，是 C.L.Hwang 和 K.Yoon 于 1981 年首次提出，根据有限个评价对象与理想化目标的接近程度进行排序的方法，是在现有的对象中进行相对优劣的评价。其基本原理，是通过检测评价对象与最优解、最劣解的距离来进行排序，若评价对象最靠近最优解同时又最远离最劣解，则为最好；否则不为最优。其中最优解的各指标值都达到各评价指标的最优值。最劣解的各指标值都达到各评价指标的最差值。

#### 4.2.2 模型的步骤

(1) 建立递阶的层次结构：根据对问题的分析，缕清问题所包含的因素，确定出各个因素之间的关联和隶属关系，按这些因素的共同特性，将它们分为目标层、准则层、方案层等多个层次。

(2) 对四个指标进行分析，建立判断矩阵。判断矩阵中一般采用九分制标度法（定义详见表 1），根据资料数据、专家意见或者系统分析人员的经验，经过反复研究后确定。

表 1：九分制标度法

标度 $a_{ij}$	定义
1	因素 i 比因素 j 同样重要
3	因素 i 比因素 j 稍微重要

5	因素 i 比因素 j 明显重要
7	因素 i 比因素 j 重要得多
9	因素 i 比因素 j 极端重要
2,4,6,8	因素 i 与因素 j 的重要性的标度值介于上述两个相邻的等级之间
标度值的倒数	因素 i 与因素 j 的反比较: $b_{ji} = 1/b_{ij}$

(3) 对判断矩阵进行一致性检验, 如果未通过一致性检验, 则修改判断矩阵, 直到通过一致性检验, 最终通过一致性检验的判断矩阵被称为一致矩阵。

(4) 使用 MATLAB 对一致矩阵使用算术平均法、几何平均法和特征值法求出四个指标的权重, 并求出其平均值。

(5) 正向化处理, 将极小型指标、中间型指标和区间型指标统一转化为极大型指标。

a. 极小型指标转换成极大型指标的公式为:

$$\max - x$$

b. 中间型指标转换成极大型指标的公式为:

$$M = \max \{ |x_i - x_{best}| \}, \tilde{x}_i = 1 - \frac{|x_i - x_{best}|}{M}$$

c. 区间型指标转换成极大型指标的公式为:

$$M = \max \{ a - \min \{ x_i \}, \max \{ x_i \} - b \}, \tilde{x}_i = \begin{cases} 1 - \frac{a - x}{M}, & x < a \\ 1, & a \leq x \leq b \\ 1 - \frac{x - b}{M}, & x > b \end{cases}$$

(6) 将正向化矩阵进行标准化处理, 目的是消除量纲的影响, 标准化处理公式为:

$$z_{ij} = x_{ij} / \sqrt{\sum_{i=1}^n x_{ij}^2}$$

(7) 计算得分并归一化, 假设有 n 个要评价的对象, m 个评价指标的标准化矩阵:

$$z = \begin{pmatrix} z_{11} & z_{12} & \cdots & z_{1m} \\ z_{21} & z_{22} & \cdots & z_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{nm} \end{pmatrix}$$

$$\begin{aligned}
Z^+ &= (Z_1^+, Z_2^+, \dots, Z_m^+) \\
&= (\max\{z_{11}, z_{21}, \dots, z_{n1}\}, \max\{z_{12}, z_{22}, \dots, z_{n2}\}, \dots, \max\{z_{1m}, z_{2m}, \dots, z_{nm}\}) \\
Z^- &= (Z_1^-, Z_2^-, \dots, Z_m^-) \\
&= (\min\{z_{11}, z_{21}, \dots, z_{n1}\}, \min\{z_{12}, z_{22}, \dots, z_{n2}\}, \dots, \min\{z_{1m}, z_{2m}, \dots, z_{nm}\})
\end{aligned}$$

定义第*i* (*i* = 1, 2, ..., *n*) 个评价对象与最大值的距离:

$$D_i^+ = \sqrt{\sum_{j=1}^m \omega_j (Z_j^+ - z_{ij})^2}$$

定义第*i* (*i* = 1, 2, ..., *n*) 个评价对象与最小值的距离:

$$D_i^- = \sqrt{\sum_{j=1}^m \omega_j (Z_j^- - z_{ij})^2}$$

则, 我们就可以求出第*i* (*i* = 1, 2, ..., *n*) 个对象未归一化的得分:

$$S_i = \frac{D_i^-}{D_i^+ + D_i^-}$$

很明显,  $0 \leq S_i \leq 1$ , 且  $S_i$  越大  $D_i^+$  越小, 即越接近最大值

(8) 对评分后的结果进行排序, 得出结果。

### 4.2.3 模型在电脑选择中的应用

(1) 建立评价体系, 画出层次结构图, 如图 1 所示:

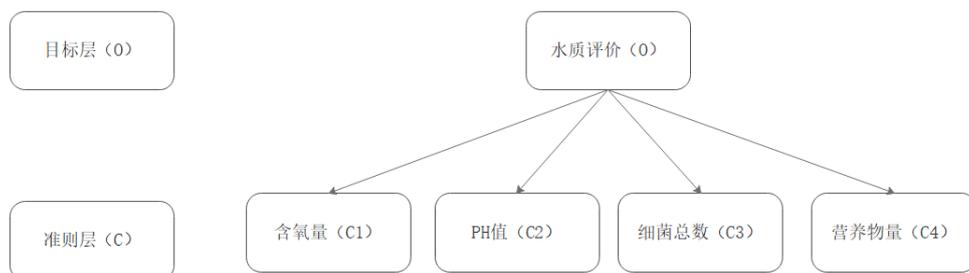


图 1: 层次结构图

(1) 建立判断矩阵并计算, 通过专家打分的方式对四个指标进行评价, 判断矩阵如表 1 所示:

表 1: 指标判断矩阵

O-C	C1	C2	C3	C4
C1	1	2	4	6

C2	1/2	1	1/4	1/6
C3	1/4	1/2	1	3/2
C4	1/6	1/3	2/3	1

(2) 对上述矩阵进行一致性检验:

step1. 计算一致性指标 CI:

$$CI = \frac{\lambda_{\max} - 1}{n - 1}$$

step2. 查找对应的平均随机一致性指标 CR:

表 7: 平均随机一致性指标 CR:

N	1	2	3	4	5	6
RI	0	0	0.52	0.89	0.12	1.26

step3. 计算一致性比例 CR:

$$CR = \frac{CI}{RI}$$

使用 MATLAB 计算得到一致性比例  $CR = -0.1960 < 0.1$  则说明通过一致性检验。

(4) 在 MATLAB 中, 使用三种计算权重的方式对一致矩阵进行计算, 使计算的结果更具有稳健性:

表 3: 四个指标的权重

算术平均法	几何平均法	特征值法	平均值
0.60	0.64	0.60	0.62
0.15	0.09	0.14	0.13
0.15	0.16	0.15	0.15
0.10	0.11	0.10	0.10



图 2: 指标权重图

(5) 使用 MATLAB 对原始矩阵进行正向化和标准化，结果在附录中。

(6) 使用 MATLAB 计算得分并进行归一化，并使用 sort 函数进行排序，得到最终的结果如下：

表 4：水质情况排名表

名称	排名
A	11
B	8
C	3
D	4
E	7
F	20
G	10
H	13
I	9
J	18
K	16
L	17
M	5
N	12
O	1
P	19
Q	6
R	15
S	2
T	14

## 五、模型的评价、改进与推广

- 1、较为方便地解决我们所遇到的问题。
- 2、层次分析法的主观性较强，后序可以使用熵权法进行修正

## 六、参考文献

- [1] 小区开放对道路通行的影响 2016 年国赛一等奖

## 七、附录

表 5：正向化后的结果为：

正向化	含氧量	PH 值	细菌总数	植物性营养物量
A	4.69	0.72	3	1.00
B	2.03	0.41	35	0.69
C	9.11	0.52	8	0.91
D	8.61	0.97	8	0.44
E	7.13	0.66	4	0.69
F	2.39	0.84	16	0.60
G	7.69	0.86	16	0.66
H	9.3	0.87	27	0.00
I	5.45	0.57	49	1.00
J	6.19	0.81	37	0.78
K	7.93	0.63	45	0.70
L	4.4	0.81	37	0.54
M	7.46	0.14	31	1.00
N	2.01	0.00	7	0.45
O	2.04	0.59	31	1.00
P	7.73	0.41	2	1.00
Q	6.35	0.60	29	0.18
R	8.29	0.03	15	1.00
S	3.54	0.81	0	0.41
T	7.44	0.49	46	0.27

表 6：正向化后标准化后的结果

标准化	含氧量	PH 值	细菌总数	植物性营养物量
A	0.16	0.25	0.02	0.31
B	0.07	0.14	0.29	0.21
C	0.32	0.18	0.07	0.28
D	0.30	0.33	0.07	0.14
E	0.25	0.23	0.03	0.21
F	0.08	0.29	0.13	0.18
G	0.27	0.30	0.13	0.20
H	0.32	0.30	0.22	0.00
I	0.19	0.20	0.40	0.31
J	0.21	0.28	0.30	0.24
K	0.27	0.22	0.37	0.21
L	0.15	0.28	0.30	0.17
M	0.26	0.05	0.25	0.31
N	0.07	0.00	0.06	0.14
O	0.07	0.20	0.25	0.31

P	0.27	0.14	0.02	0.31
Q	0.22	0.21	0.24	0.06
R	0.29	0.01	0.12	0.31
S	0.12	0.28	0.00	0.13
T	0.26	0.17	0.38	0.08

代码:

```

1. clc;clear
2. load data_water_quality.mat
3. load panduan.mat
4. %% 输入矩阵
5. disp('请输入判断矩阵: ')
6. A = input('判断矩阵 A = ')
7. n = size(A,1)
8. %% 求特征值和特征向量 V 表示特征向量 D 表示特征值
9. [V,D] = eig(A)
10. Max_eig = max(max(D))
11. D = Max_eig
12. [r,c] = find(D == Max_eig , 1)
13. %% 计算一致性比例
14. clc
15. CI = (Max_eig - n) / (n-1);
16. RI=[0 0 0.52 0.89 1.12 1.26 1.36 1.41 1.46 1.49 1.52 1.54 1.56 1.58 1.59];
17. CR=CI/RI(n);
18. disp('一致性指标 CI=');disp(CI);
19. disp('一致性比例 CR=');disp(CR);
20. if CR<0.10
21.     disp('因为 CR < 0.10, 所以该判断矩阵 A 的一致性可以接受!');
22. else
23.     disp('注意: CR >= 0.10, 因此该判断矩阵 A 需要进行修改!');
24. end
25. %% 权重计算
26. Sum_A = sum(A);
27. SUM_A = repmat(Sum_A,n,1);
28. Stand_A = A ./ SUM_A;
29.
30. disp('算术平均法求权重的结果为: ');
31. disp(sum(Stand_A,2)./n)
32. w1 = sum(Stand_A,2)./n;
33. % 方法 2: 几何平均法求权重%
34. Prduct_A = prod(A,2);
35. Prduct_n_A = Prduct_A .^ (1/n);

```

```

36. disp('几何平均法求权重的结果为: ');
37. disp(Product_n_A ./ sum(Product_n_A))
38. w2 = Product_n_A ./ sum(Product_n_A);
39. % 方法3: 特征值法求权重%
40. [V,D] = eig(A);
41. Max_eig = max(max(D));
42. [r,c]=find(D == Max_eig , 1);
43. disp('特征值法求权重的结果为: ');
44. disp( V(:,c) ./ sum(V(:,c)) )
45. w3 = V(:,c) ./ sum(V(:,c));
46. disp('平均后的权重为: ')
47. w = (w1 + w2 + w3)/3
48. disp(w1);
49. %% topsis 法来进行计算
50. [r,c] = size(X)
51. disp(['当前共有' num2str(r) '个评价对象和' num2str(c) '个指标'])
52. %%正向化
53. judge = input('是否需要正向化处理, 需要按 1, 不需要按 0: ')
54. if judge == 1
55.     Position = input('需要对哪几个指标进行正向化? ')
56.     Type = input('分别对应的类型, 输入 1 表示极小型, 2 表示中间型, 3 表示区间型: ')
57.
58.     for i = 1:size(Position , 2)
59.         X(:,Position(i)) = Positivition(X(:,Position(i)),Type(i),Position(i)
        );
60.     end
61.     disp('正向化矩阵后的结果为: ');
62.     disp(X);
63. end
64. %% 标准化
65. n = size(X,1)
66. Z = X./repmat(sum(X.*X).^0.5 , n , 1)
67. disp('标准化矩阵 Z = ')
68. disp(Z)
69. %% 计算得分
70. D_P = sum(repmat(w',n,1) .* (repmat(max(Z) , n , 1) - Z).^ 0.5 , 2)
71. D_N = sum(repmat(w',n,1) .* (repmat(min(Z) , n , 1) - Z).^ 0.5 , 2)
72.
73. S = D_N / (D_P + D_N)
74. Stand_s = S/sum(S) % 归一化
75. [sorted_s , index] = sort(Stand_s , 'descend')
76. %%min2max
77. function [output] = min2max(x)

```

```

78.     output = max(x) - x
79. end
80. %%mid2max
81. function [output] = mid2max(x)
82.     best = input('请输入最佳指标的数值: ')
83.     M = max(abs(x - best))
84.     output = 1 - abs(x - best)/M
85. end
86. %%inter2max
87. function [posit_x] = inter2max(x,a,b)
88.     disp('输入区间型变量的下界: ')
89.     a = input('a = ');
90.     disp('输入区间型变量的上界: ')
91.     b = input('b = ');
92.     r_x = size(x,1); % row of x
93.     M = max([a-min(x),max(x)-b]);
94.     posit_x = zeros(r_x,1);
95.     for i = 1: r_x
96.         if x(i) < a
97.             posit_x(i) = 1-(a-x(i))/M;
98.         elseif x(i) > b
99.             posit_x(i) = 1-(x(i)-b)/M;
100.        else
101.            posit_x(i) = 1;
102.        end
103.    end
104. end

```