四　川　大　学

博士研究生课程考试试卷

姓名　王霞　　学号　2021326200014

学院　灾后重建与管理学院　　专业　安全科学与减灾

考试课程名称 灾害信息与大数据科学	
考试方式 □笔试 □口试 ☑撰写论文	考试成绩
任课教师 田兵伟	考试时间

四川大学研究生院制

目录

一、代码示例

```
library(tibble)
library(dplyr)
library(readr)
install.packages("devtools")
library(devtools)
install_github("Youxiaaaa/R_Package/wxTOPSIS")
library(wxTOPSIS)

weight <- c(0.1,0.1,0.3,0.2,0.2,0.1)
attribute <- c(1,0,1,0,0,0)
dat <- usetopsis("dataset.csv", weight, attribute)
print(dat)
```

二、结果分析

1. 示例题目

例：假设发生了一起山体滑坡事故，并造成了人员伤亡和经济损失。经初步调查，确定了 4 个备选应急方案：(A_1)，(A_2)，(A_3)，(A_4)。选择时，决策者需要考虑：救援成本(R_1)，直接财产损失(R_2)，间接财产损失(R_3)，救援时效性(R_4)，人员伤亡(R_5)，可操作性(R_6)。

其决策矩阵如下：

	R_1	R_2	R_3	R_4	R_5	R_6
A_1	5	1.4	6	差（0.3）	中（0.5）	好（0.7）
A_2	9	2	30	好（0.7）	中（0.5）	很好（0.9）
A_3	8	1.8	11	中（0.5）	高（0.7）	中（0.5）
A_4	12	2.5	18	好（0.7）	中（0.5）	中（0.5）
权重	0.1	0.1	0.3	0.2	0.2	0.1

2. 运行结果

```
> print(dat)
# A tibble: 4 x 10
  NAME      R1      R2      R3      R4      R5      R6      du      dn      Ri
  <chr>  <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
1 A3    0.0451  0.0458  0.0888  0.0870  0.126   0.0373  0.0658  0.163   0.713
2 A1    0.0282  0.0356  0.0484  0.0522  0.0898  0.0522  0.0845  0.198   0.701
3 A4    0.0677  0.0636  0.145   0.122   0.0898  0.0373  0.115   0.123   0.517
4 A2    0.0508  0.0509  0.242   0.122   0.0898  0.0671  0.199   0.0791  0.285
>
```

3. 结果分析

应用 TOPSIS 方法评选方案，如下：

Step1：构造决策矩阵：

$$M=\begin{pmatrix}5 & 1.4 & 6 & 0.3 & 0.5 & 0.7\\ 9 & 2 & 30 & 0.7 & 0.5 & 0.9\\ 8 & 1.8 & 11 & 0.5 & 0.7 & 0.5\\ 12 & 2.5 & 18 & 0.7 & 0.5 & 0.5\end{pmatrix}$$

Step2：规范化决策矩阵

$$N=\begin{pmatrix}0.2822 & 0.3562 & 0.1615 & 0.2611 & 0.4490 & 0.5217\\ 0.5079 & 0.5088 & 0.8037 & 0.6039 & 0.4490 & 0.6708\\ 0.4515 & 0.4579 & 0.2960 & 0.4352 & 0.6286 & 0.3727\\ 0.6772 & 0.6360 & 0.4844 & 0.6093 & 0.4490 & 0.3727\end{pmatrix}$$

Step3：计算加权规范决策矩阵

$$V=\begin{pmatrix}0.0282 & 0.0356 & 0.0484 & 0.0522 & 0.0898 & 0.0522\\ 0.0508 & 0.0509 & 0.2422 & 0.1219 & 0.0898 & 0.0671\\ 0.0452 & 0.0458 & 0.0888 & 0.0870 & 0.1257 & 0.0373\\ 0.0677 & 0.0636 & 0.1453 & 0.1219 & 0.0898 & 0.0373\end{pmatrix}$$

Step4：取 R_1， R_3 为成本型属性， R_2， R_4， R_5， R_6 为效益型属性。

正理想解： $A^+=(0.0282,0.0636,0.0484,0.1219,0.1257,0.0671)$

负理想解： $A^-=(0.0677,0.0356,0.2422,0.0522,0.0898,0.0373)$

Step5：确定某个方案与正理想解和负理想解的分离度

$d_1^+=0.0848$　　$d_1^-=0.1983$　　$d_2^+=0.1987$　　$d_2^-=0.0791$

$d_3^+=0.0658$　　$d_3^-=0.1632$　　$d_4^+=0.1146$　　$d_4^-=0.1225$

Step5：确定相对接近度

$$r_1^*=\frac{d_1^-}{d_1^++d_1^-}=0.7011 \qquad r_2^*=0.2847 \qquad r_3^*=0.7126 \qquad r_4^*=0.5168$$

排序 $A_3>A_1>A_4>A_2$，所以选择应急方案。结果分析和运行结果一致。

4. 源代码分析

```
# 标准化变量值
z_value <- function(x){
  x / sqrt(sum(x^2))
}

# 加权标准化变量值
w_value <- function(x,y){
  for(i in 1:ncol(x)) {
    x[,i]=x[,i]*y[i]
  }
  return (x)
}

#根据属性得到正理想型(1 是成本型，0 是效益型)
en_positive <- function(x,y) {
  z <- c()
  for(i in 1:ncol(x)) {
    if(y[i]==1) {
      z[i]=min(x[,i])
    }
    else {
      z[i]=max(x[,i])
    }
  }
  return (z)
}

#根据属性得到负理想型(1 是成本型，0 是效益型)
en_negative <- function(x,y) {
  z <- c()
  for(i in 1:ncol(x)) {
    if(y[i]==1) {
      z[i]=max(x[,i])
    }
    else {
      z[i]=min(x[,i])
    }
  }
  return (z)
}

# 计算最优距离
dist <-function(x, std){
  res <- c()
```

```
  for ( i in 1 : nrow(x)) {
    res[i] = sqrt(sum((unlist(x[i,-1])-std)^2))
  }
  return(res)
}
# 主函数
usetopsis <- function(content, weight, attribute) {
  # load sample data
  dat <- read_csv(content)

  # 按列对数据进行规范化
  dat_f <- dat %>% mutate(across(c(2:ncol(dat)), z_value))

  # 按列对数据进行加权规范化
  dat_z <-mutate(dat_f, w_value(dat_f[,c(2:ncol(dat))],weight))

  # unlist 转换tibble为vector
  z_positive <- en_positive(dat_z[,c(2:ncol(dat_z))],attribute)
  z_negative <- en_negative(dat_z[,c(2:ncol(dat_z))],attribute)

  # dat_z %>% select(2:ncol(dat_z))) %>% rowwise() %>% mutate(du = dist(., z_
    max), dn= dist(., z_min))
  du <- dist(dat_z, z_positive)
  dn <- dist(dat_z, z_negative)

  # 计算RI并按照降序排序
  return (dat_z %>% add_column(du = du, dn = dn) %>%
            mutate(Ri= dn/(du+dn)) %>%
            arrange(-Ri))
}
```