VIKOR算法介绍

多属性决策算法，关键是区分出效益型和成本型的指标，尽量让效益型指标越大越好，成本型指标越小越好，从而选择出最优





import numpy as np  
import pandas as pd  
  
input\_df = pd.read\_csv("./dataset/vikor\_test.csv")  
# v4越小越优，其他变量越大越优  
add\_column = ["v1", "v2", "v3", "v5", "v6", "v7", "v8", "v9", "v10", "v11"]  
subtract\_column = ["v4"]  
select\_column = add\_column + subtract\_column  
(m, n) = input\_df.shape  
# step1.利用变异系数法计算指标权重w  
a = input\_df[select\_column].mean()  
s = input\_df[select\_column].std() # 样本标准差N-1  
delta = s / a  
w = delta / delta.sum()  
# step2.计算决策矩阵r，分为收益型和成本型  
r = pd.DataFrame(np.zeros([m, len(select\_column)]), columns=select\_column)  
r[add\_column] = (input\_df[add\_column] - input\_df[add\_column].min()) / (  
 input\_df[add\_column].max() - input\_df[add\_column].min())  
r[subtract\_column] = (input\_df[subtract\_column].max() - input\_df[subtract\_column]) / (  
 input\_df[subtract\_column].max() - input\_df[subtract\_column].min())  
# step3.确定理想点和负理想点  
r\_star = r.max()  
r\_star2 = r.min()  
# step4.计算群体效用值和个别遗憾值  
ss = np.zeros([m])  
rr = np.zeros([m])  
for i in range(m):  
 ss[i] = (w \* (r\_star - r.iloc[i][select\_column]) / (r\_star - r\_star2)).sum()  
 rr[i] = (w \* (r\_star - r.iloc[i][select\_column]) / (r\_star - r\_star2)).max()  
v = 0.5  
ss\_star = ss.min()  
# step5.计算综合排序指标，指标越小，信用等级越高  
q = np.zeros([m])  
for i in range(m):  
 q[i] = v \* (ss[i] - ss.min()) / (ss.max() - ss.min()) + (1 - v) \* (rr[i] - rr.min()) / (rr.max() - rr.min())  
# q.sort()  
print(q)