

順位データの双対尺度法

青木繁伸

2020年3月17日

1 目的

順位データについて、双対尺度法による解析を行う。

2 使用法

```
import sys
sys.path.append("statlib")
from multi import ro_dual
ro_dual(F, verbose=True)
```

行スコアと列スコアのバイプロットを描く

```
import sys
sys.path.append("statlib")
from multi import ro_dual_plot
ro_dual_plot(obj, weighted=False, ax1=1, ax2=2, color="blue", color2="red", alpha=0.5)
```

2.1 引数

<code>F</code>	順位データを行列として与える
<code>col_names</code>	評価対象のラベル (デフォルトの <code>None</code> なら、便宜的な名前を仮定する)
<code>row_names</code>	評価者のラベル (デフォルトの <code>None</code> なら、便宜的な名前を仮定する)
<code>maxaxis</code>	解の数を制限する (デフォルトは 5)
<code>verbose</code>	必要最小限のプリント出力をする
<code>obj</code>	<code>ro_dual()</code> の戻り値
<code>weighted</code>	<code>True</code> を指定すれば重み付きの行・列スコアをプロットする
<code>ax1</code>	横軸にとる解の番号
<code>ax2</code>	縦軸にとる解の番号
<code>color</code>	列スコアに対する点とテキストの色
<code>color2</code>	行スコアに対する点とテキストの色
<code>alpha</code>	アルファチャンネル (デフォルトは 0.5)

2.2 戻り値の名前

"result"	結果表
"rs"	行スコア
"cs"	列スコア
"wrs"	重みつき行スコア
"wcs"	重みつき列スコア

3 使用例

```
F = [[6,1,5,3,2,8,4,7],
      [3,8,1,6,7,5,4,2],
      [5,7,1,6,8,2,4,3],
      [4,6,2,3,8,7,1,5],
      [2,4,6,3,7,5,1,8],
      [2,4,5,3,8,7,1,6],
      [1,7,6,3,8,5,2,4],
      [7,5,3,1,8,4,6,2],
      [4,2,7,3,8,6,5,1],
      [5,1,2,4,7,6,3,8],
      [6,4,3,2,8,7,5,1],
      [3,8,4,2,5,6,1,7],
      [3,2,1,6,4,7,5,8],
      [5,8,1,4,7,3,6,2]]

import sys
sys.path.append("statlib")
from multi import ro_dual

a = ro_dual(F)
```

Summary

	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
eta square	0.141472	0.122668	0.065831	0.055134	0.022531
correlation	0.376128	0.350240	0.256576	0.234807	0.150105
contribution	34.705409	30.092481	16.149470	13.525322	5.527319
cumu. contrib.	34.705409	64.797890	80.947360	94.472681	100.000000

Row score

	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
R1	1.083260	0.994792	1.010065	-0.672067	1.315833
R2	-1.401384	-0.437308	-0.834155	-0.645887	-0.606477
R3	-1.305682	-0.686448	-0.746310	-0.883252	-0.837312

R4	-1.308472	1.058171	-0.046030	-0.636846	0.698923
R5	-0.344224	1.667944	-0.322205	0.693907	-0.494508
R6	-0.801457	1.605872	0.247905	0.380521	-0.600201
R7	-1.223503	0.791443	-0.369551	1.455372	-0.523979
R8	-1.088726	-0.588526	1.448211	-0.366436	1.225776
R9	-0.546940	0.076377	2.061734	1.009645	-1.406954
R10	-0.054101	1.275397	0.597721	-1.707246	-0.661620
R11	-1.105210	-0.149283	1.824117	-0.473628	0.557675
R12	-0.717716	1.173728	-1.083847	0.399296	2.093881
R13	0.463729	0.898715	-0.310415	-2.059992	-0.991960
R14	-1.322475	-1.026901	-0.345257	-0.710888	0.482432

Column score

	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
C1	-0.350584	0.838875	-0.909848	0.764129	-1.298141
C2	1.255766	0.664667	1.665293	-0.660003	-1.241109
C3	-0.865885	-0.247532	-0.542972	-2.396498	0.226517
C4	-0.564459	0.597548	1.094065	0.526014	1.762960
C5	2.004001	-0.508396	-0.819158	0.114437	1.094249
C6	0.041102	-1.186677	-0.881640	0.483695	-0.509676
C7	-0.588290	1.462987	-0.633878	0.459257	0.263120
C8	-0.931652	-1.621472	1.028138	0.708968	-0.297920

Weighted row score

	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
R1	1.083260	0.994792	1.010065	-0.672067	1.315833
R2	-1.401384	-0.437308	-0.834155	-0.645887	-0.606477
R3	-1.305682	-0.686448	-0.746310	-0.883252	-0.837312
R4	-1.308472	1.058171	-0.046030	-0.636846	0.698923
R5	-0.344224	1.667944	-0.322205	0.693907	-0.494508
R6	-0.801457	1.605872	0.247905	0.380521	-0.600201
R7	-1.223503	0.791443	-0.369551	1.455372	-0.523979
R8	-1.088726	-0.588526	1.448211	-0.366436	1.225776
R9	-0.546940	0.076377	2.061734	1.009645	-1.406954
R10	-0.054101	1.275397	0.597721	-1.707246	-0.661620
R11	-1.105210	-0.149283	1.824117	-0.473628	0.557675
R12	-0.717716	1.173728	-1.083847	0.399296	2.093881
R13	0.463729	0.898715	-0.310415	-2.059992	-0.991960
R14	-1.322475	-1.026901	-0.345257	-0.710888	0.482432

Weighted column score

	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
--	--------	--------	--------	--------	--------

```

C1 -0.350584  0.838875 -0.909848  0.764129 -1.298141
C2  1.255766  0.664667  1.665293 -0.660003 -1.241109
C3 -0.865885 -0.247532 -0.542972 -2.396498  0.226517
C4 -0.564459  0.597548  1.094065  0.526014  1.762960
C5  2.004001 -0.508396 -0.819158  0.114437  1.094249
C6  0.041102 -1.186677 -0.881640  0.483695 -0.509676
C7 -0.588290  1.462987 -0.633878  0.459257  0.263120
C8 -0.931652 -1.621472  1.028138  0.708968 -0.297920

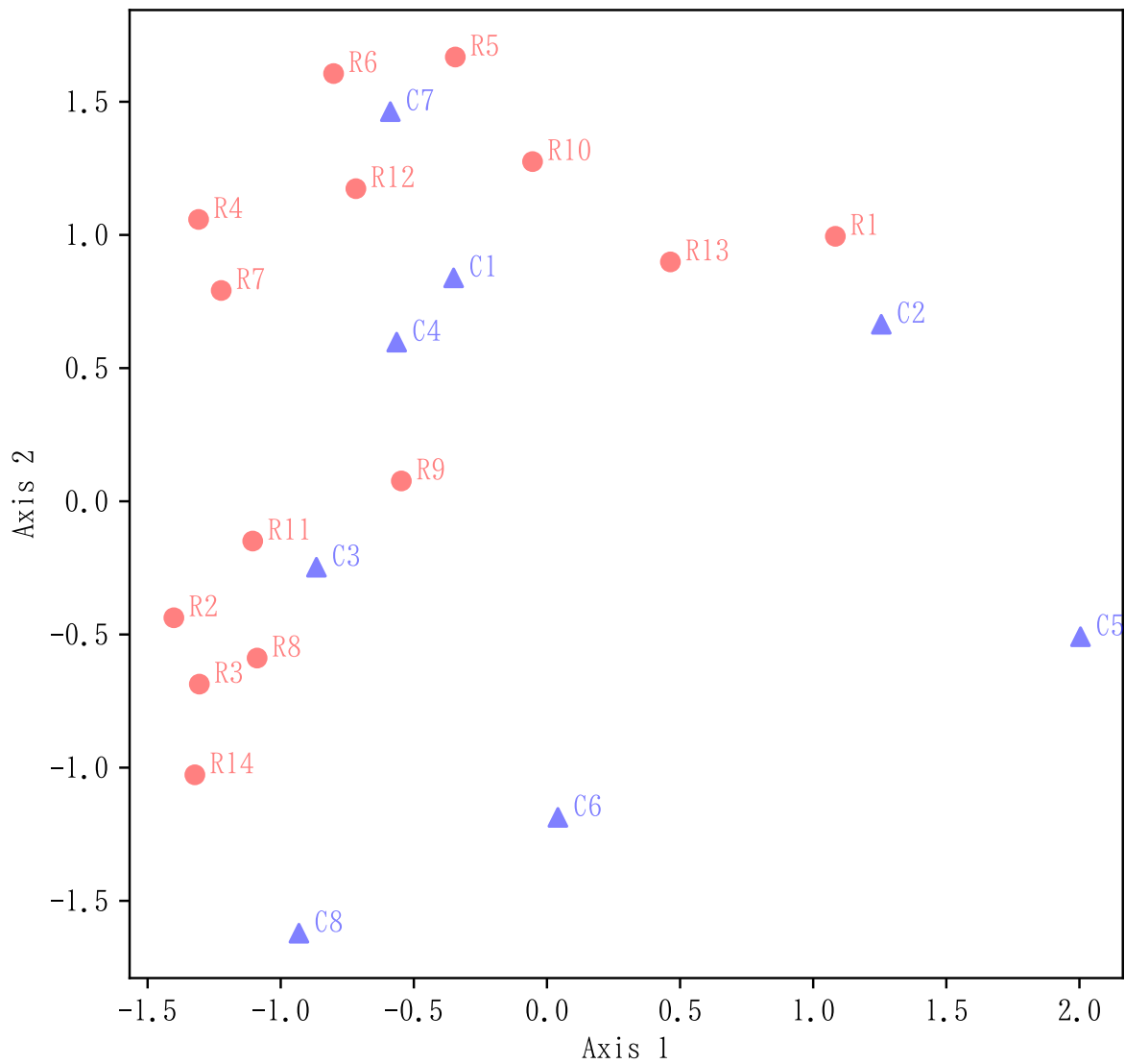
```

```

from multi import ro_dual_plot
ro_dual_plot(a)

```

Dual Scale Analysis (non-weighted scores)



```

ro_dual_plot(a, weighted=True)

```

Dual Scale Analysis (weighted scores)

