

# 双対尺度法

青木繁伸

## 1 目的

一対比較データについて、双対尺度法による解析を行う。

参考文献

西里静彦「質的データの数量化－双対尺度法とその応用－」，朝倉書店，1982

## 2 使用法

```
import sys
sys.path.append("statlib")
from paired_comparison_dual import paired_comparison_dual
paired_comparison_dual(tbl, one_two=True, col_names=None, row_names=None, max_axis=5, 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 paired_comparison_dual import paired_comparison_dual_plot
paired_comparison_dual_plot(obj, weighted=False, ax1=1, ax2=2, color="blue", color2="red", alpha=0.5)
```

### 2.1 引数

|                        |  |
|------------------------|--|
| <code>tbl</code>       | 一対比較データを行列として与える   |
| <code>one_two</code>   | 一対比較のデータが $x_i > x_j$ なら 1, $x_i = x_j$ なら 0, $x_i < x_j$ なら -1 で入力されていれば <code>False</code><br>$x_i > x_j$ なら 1, $x_i < x_j$ なら 2, $x_i = x_j$ なら 0 で入力されていれば <code>True</code> |
| <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>paired_comparison_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 = [[1,1,2,1,1,2,1,2,2,2,2,2,2,1,1,2,1,1,1,2,1,1,2,1,2,1,2],
      [2,2,2,1,2,1,2,1,2,1,2,1,2,2,1,1,1,2,1,1,1,2,2,2,2,1,2,2],
      [1,1,1,1,1,2,1,1,1,1,1,2,1,1,1,1,2,1,1,1,2,1,2,2,2,2,1,1],
      [2,1,2,1,1,1,2,1,1,1,1,1,2,2,1,2,2,2,1,1,1,2,2,2,2,2,2,2],
      [2,2,2,1,2,1,2,2,2,1,2,2,2,2,1,2,1,2,1,1,1,1,2,2,2,1,2,2],
      [1,1,1,1,1,1,1,2,2,1,2,2,2,2,1,2,2,2,1,1,1,1,2,2,2,2,2,1],
      [1,1,1,1,1,2,1,1,2,1,1,2,1,2,1,1,2,1,1,1,2,1,2,2,2,2,2,1],
      [1,1,1,1,1,2,1,1,2,1,2,2,1,2,1,2,2,1,1,2,2,1,2,2,1,2,1,1],
      [1,2,2,1,1,2,1,2,2,1,1,2,2,1,1,1,2,1,1,1,2,1,2,2,2,2,2,1],
      [1,2,1,1,2,2,2,2,2,1,2,2,2,1,1,1,1,1,1,2,2,2,2,2,2,1,1,2],
      [1,2,1,1,1,1,1,2,2,2,2,2,2,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2],
      [2,2,2,2,1,2,2,1,2,1,1,1,1,2,1,1,1,1,1,1,1,1,1,1,1,2,2,1],
      [1,2,1,1,2,1,2,2,2,2,2,2,2,1,1,1,1,1,1,2,1,2,2,2,2,1,1,2],
      [2,2,2,2,1,2,1,2,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2,1]]

import sys
sys.path.append("statlib")
from paired_comparison_dual import paired_comparison_dual

a = paired_comparison_dual(F)
```

### Summary

|               | Axis 1    | Axis 2    | Axis 3    | Axis 4    | Axis 5     |
|---------------|-----------|-----------|-----------|-----------|------------|
| eta square    | 0.141067  | 0.109784  | 0.065227  | 0.055123  | 0.028448   |
| correlation   | 0.375589  | 0.331337  | 0.255396  | 0.234782  | 0.168665   |
| contribution  | 35.297781 | 27.470102 | 16.321095 | 13.792770 | 7.118252   |
| cum. contrib. | 35.297781 | 62.767883 | 79.088978 | 92.881748 | 100.000000 |

### Row score

|    | Axis 1   | Axis 2    | Axis 3    | Axis 4    | Axis 5    |
|----|----------|-----------|-----------|-----------|-----------|
| R1 | 1.101257 | -0.416655 | -0.435636 | -0.155752 | 1.989033  |
| R2 | 0.334417 | 1.473295  | 1.417472  | -0.449350 | -0.520176 |
| R3 | 1.210731 | -1.159970 | 0.104690  | 0.158204  | -1.526930 |
| R4 | 0.418405 | 0.506687  | 2.144601  | 0.782704  | -0.897664 |
| R5 | 0.668767 | 1.480685  | 0.872008  | -0.332109 | 0.255117  |
| R6 | 1.497937 | 0.195057  | 0.345398  | 1.128946  | 0.983613  |

|     |           |           |           |           |           |
|-----|-----------|-----------|-----------|-----------|-----------|
| R7  | 1.422351  | -0.967869 | 0.693198  | 0.294025  | -0.298941 |
| R8  | 1.085625  | -0.926942 | -0.518151 | 1.193154  | -0.500687 |
| R9  | 1.527787  | -0.366311 | 0.095199  | -1.011463 | -0.122307 |
| R10 | 0.808443  | 1.148995  | -1.172902 | -0.617681 | -1.593369 |
| R11 | 1.295815  | 0.613090  | -0.545908 | -0.938717 | 1.036729  |
| R12 | -0.120340 | -0.651086 | 1.512945  | -1.906429 | 0.797150  |
| R13 | 0.678091  | 1.453348  | -1.257221 | -0.585373 | -0.387083 |
| R14 | -0.024482 | -1.288676 | -0.098144 | -1.994588 | -0.921600 |

Column score

|    | Axis 1    | Axis 2    | Axis 3    | Axis 4    | Axis 5    |
|----|-----------|-----------|-----------|-----------|-----------|
| C1 | 1.041721  | -0.412330 | -0.487567 | 1.244424  | 0.372903  |
| C2 | -0.923079 | -0.949302 | 1.504153  | -0.140475 | -1.562700 |
| C3 | 0.506228  | 0.360627  | -1.016581 | -2.246284 | -0.468372 |
| C4 | 0.795922  | 0.247724  | 1.388135  | -0.443439 | 1.566724  |
| C5 | -1.949742 | -0.720476 | -0.720803 | -0.076646 | 1.402748  |
| C6 | -0.389380 | 1.077688  | -1.113201 | 0.971608  | -0.775117 |
| C7 | 1.090912  | -1.379694 | -0.360709 | 0.326212  | -0.395884 |
| C8 | -0.172581 | 1.775763  | 0.806574  | 0.364600  | -0.140301 |

Weighted row score

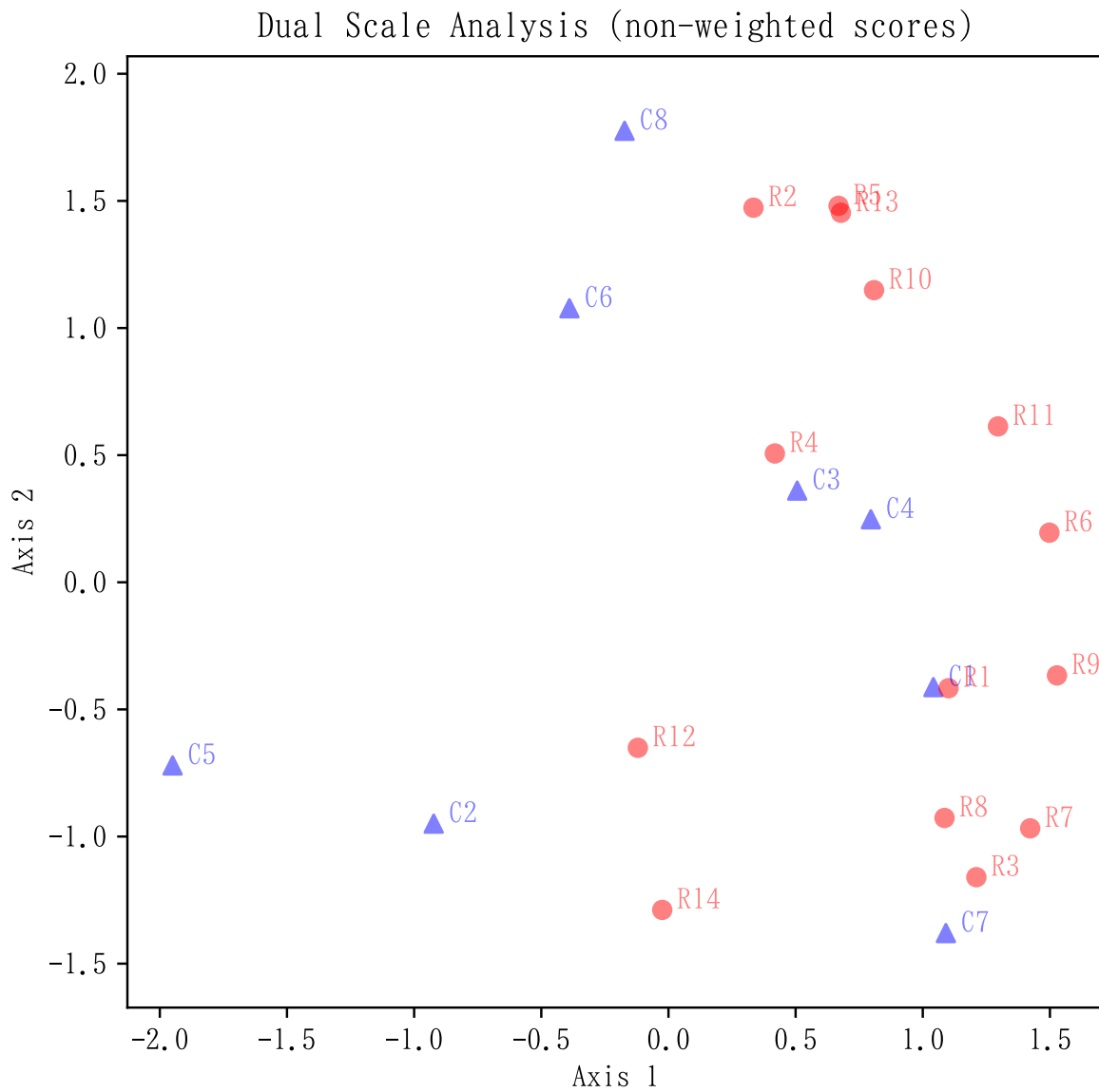
|     | Axis 1    | Axis 2    | Axis 3    | Axis 4    | Axis 5    |
|-----|-----------|-----------|-----------|-----------|-----------|
| R1  | 1.101257  | -0.416655 | -0.435636 | -0.155752 | 1.989033  |
| R2  | 0.334417  | 1.473295  | 1.417472  | -0.449350 | -0.520176 |
| R3  | 1.210731  | -1.159970 | 0.104690  | 0.158204  | -1.526930 |
| R4  | 0.418405  | 0.506687  | 2.144601  | 0.782704  | -0.897664 |
| R5  | 0.668767  | 1.480685  | 0.872008  | -0.332109 | 0.255117  |
| R6  | 1.497937  | 0.195057  | 0.345398  | 1.128946  | 0.983613  |
| R7  | 1.422351  | -0.967869 | 0.693198  | 0.294025  | -0.298941 |
| R8  | 1.085625  | -0.926942 | -0.518151 | 1.193154  | -0.500687 |
| R9  | 1.527787  | -0.366311 | 0.095199  | -1.011463 | -0.122307 |
| R10 | 0.808443  | 1.148995  | -1.172902 | -0.617681 | -1.593369 |
| R11 | 1.295815  | 0.613090  | -0.545908 | -0.938717 | 1.036729  |
| R12 | -0.120340 | -0.651086 | 1.512945  | -1.906429 | 0.797150  |
| R13 | 0.678091  | 1.453348  | -1.257221 | -0.585373 | -0.387083 |
| R14 | -0.024482 | -1.288676 | -0.098144 | -1.994588 | -0.921600 |

Weighted column score

|    | Axis 1    | Axis 2    | Axis 3    | Axis 4    | Axis 5    |
|----|-----------|-----------|-----------|-----------|-----------|
| C1 | 1.041721  | -0.412330 | -0.487567 | 1.244424  | 0.372903  |
| C2 | -0.923079 | -0.949302 | 1.504153  | -0.140475 | -1.562700 |
| C3 | 0.506228  | 0.360627  | -1.016581 | -2.246284 | -0.468372 |
| C4 | 0.795922  | 0.247724  | 1.388135  | -0.443439 | 1.566724  |
| C5 | -1.949742 | -0.720476 | -0.720803 | -0.076646 | 1.402748  |
| C6 | -0.389380 | 1.077688  | -1.113201 | 0.971608  | -0.775117 |

```
C7 1.090912 -1.379694 -0.360709 0.326212 -0.395884
C8 -0.172581 1.775763 0.806574 0.364600 -0.140301
```

```
from paired_comparison_dual import paired_comparison_dual_plot
paired_comparison_dual_plot(a)
```



```
from paired_comparison_dual import paired_comparison_dual_plot
paired_comparison_dual_plot(a, weighted=True)
```

Dual Scale Analysis (weighted scores)

