Multistructural Analysis: A (Nascent) Toolbox for MATLAB

From the toolbox construction team (listed in alphabetical order):

Lawrence Hubert; Hans-Friedrich Köhn; Douglas Steinley

M-files are available from:

`http://cda.psych.uiuc.edu/multistructuralanalysis_mfiles`

Version: April 27, 2007

Contents

List of Tables

List of Figures

1 Introduction

The two previous MATLAB Toolboxes in the series have dealt with Cluster Analysis and Unidimensional Scaling and the use of a single structure to help explain the patterning of data within a given proximity matrix. The present Multistructural Analysis Toolbox extends this interest in proximity matrix representation to the additive combination of two (or more) of these structures chosen from the earlier Toolboxes. As before, we work entirely within a MATLAB environment and provide our (open-source) M-files from the web site:

`http://cda.psych.uiuc.edu/multistructuralanalysis_mfiles`

The next few sections provide an explanation of the data sets used in our examples and discuss the various multistructural M-files that have been developed. Some of these M-files have been accelerated by using code conversions to Fortran callable subroutines through the use of the program `matlab2fmex.m` (written by Benjamin Barrowes). Various extensions and generalizations are then given in the remaining sections: the choice of representational metric (Euclidean or city-block); the related area of multifacility location (in the Euclidean, squared-Euclidean, or city-block contexts); and finally, incorporating the confirmatory fitting of tied coordinate patterns in city-block multidimensional scaling.

1.1 The Data Sets Used for Purposes of Illustration

There are four data sets used throughout this Toolbox to provide examples of use for the various M-files introduced. One is from Johnson and Tversky (1980) on perceptions of risk, and is in the form of a one-mode 18×18 symmetric proximity matrix. In our circular representations, we will again rely on the Morse Code digit data introduced in Section 5 of the Unidimensional Scaling Toolbox. The third proximity matrix for our discussion of multidimensional scaling (really, in two dimensions) based on the Euclidean and/or city-block metric, was derived from an inter-town distance matrix among twenty-seven Hampshire County cities (in England), published by John Norden in 1625.

Finally, we give six two-mode (10×15) proximity matrices from Osgood and Luria (1954) that provide semantic differential data on a well-known case of multiple personality (the *Three Faces of Eve*). There are two replications given for each of the separate personalities of Eve White, Eve Black, and Jane; we will chose one of these matrices for purposes of illustration.

1.1.1 Risk Perception

The first data set (an 18×18 proximity matrix) is generated from Johnson and Tversky (1984) and given in Table 1. It involves eighteen risks:

1: accidental falls
2: airplane accidents
3: electrocution
4: fire
5: flood
6: heart disease
7: homicide
8: leukaemia
9: lightning
10: lung cancer
11: nuclear accident
12: stomach cancer
13: stroke
14: terrorism
15: tornados
16: toxic chemical spill
17: traffic accidents
18: war

The original proximity matrix was obtained by averaging the ratings from a group of subjects who evaluated risk pairs on a scale from one (very dissimilar) to nine (very similar). To key these as dissimilarities, the similarities were subtracted from 10.0 to produce the entries given in Table 1. The `ascii` data file is called `risk_rate.dat` and has verbatim contents as follows:

```
0.0 7.6 6.9 5.4 6.2 7.5 6.4 7.1 6.2 7.5 7.1 8.2 6.0 7.7 6.8 6.9 4.8 7.5
```

```
7.6 0.0 7.0 7.0 5.8 8.1 6.9 7.9 5.8 7.9 6.5 8.0 7.6 5.7 6.6 6.4 4.7 6.2
6.9 7.0 0.0 3.0 5.9 7.6 7.4 7.9 3.3 8.3 6.5 7.9 7.7 7.0 6.1 6.7 7.0 7.1
5.4 7.0 3.0 0.0 7.0 8.3 6.1 7.4 3.8 7.3 6.5 7.7 7.2 6.3 5.4 5.7 5.8 6.1
6.2 5.8 5.9 7.0 0.0 8.5 5.3 6.5 3.3 7.7 6.5 8.2 7.9 7.0 2.3 6.9 7.2 7.3
7.5 8.1 7.6 8.3 8.5 0.0 7.5 4.4 5.8 6.9 8.1 4.1 3.0 8.2 8.1 7.6 6.7 8.0
6.4 6.9 7.4 6.1 5.3 7.5 0.0 7.7 8.0 8.0 4.0 8.2 7.5 3.8 7.5 6.8 5.3 3.0
7.1 7.9 7.9 7.4 6.5 4.4 7.7 0.0 8.1 6.1 6.9 3.8 4.8 8.4 8.0 6.7 7.7 7.4
6.2 5.8 3.3 3.8 3.3 5.8 8.0 8.1 0.0 7.8 6.5 7.1 6.3 7.0 2.4 6.5 7.1 7.0
7.5 7.9 8.3 7.3 7.7 6.9 8.0 6.1 7.8 0.0 8.0 4.3 4.1 7.7 8.6 7.0 6.9 7.4
7.1 6.5 6.5 6.5 6.5 8.1 4.0 6.9 6.5 8.0 0.0 6.9 8.3 7.1 6.3 2.7 6.8 4.3
8.2 8.0 7.9 7.7 8.2 4.1 8.2 3.8 7.1 4.3 6.9 0.0 7.1 8.1 7.9 7.0 7.8 7.9
6.0 7.6 7.7 7.2 7.9 3.0 7.5 4.8 6.3 4.1 8.3 7.1 0.0 7.1 4.9 7.4 5.7 7.8
7.7 5.7 7.0 6.3 7.0 8.2 3.8 8.4 7.0 7.7 7.1 8.1 7.1 0.0 7.1 6.4 7.1 2.5
6.8 6.6 6.1 5.4 2.3 8.1 7.5 8.0 2.4 8.6 6.3 7.9 4.9 7.1 0.0 5.5 7.5 7.8
6.9 6.4 6.7 5.7 6.9 7.6 6.8 6.7 6.5 7.0 2.7 7.0 7.4 6.4 5.5 0.0 5.8 7.8
4.8 4.7 7.0 5.8 7.2 6.7 5.3 7.7 7.1 6.9 6.8 7.8 5.7 7.1 7.5 5.8 0.0 7.0
7.5 6.2 7.1 6.1 7.3 8.0 3.0 7.4 7.0 7.4 4.3 7.9 7.8 2.5 7.8 7.8 7.0 0.0
```

1.1.2 Morse Code Digit Data

As in the Unidimensional Scaling Toolbox, a different data set is used for illustration of (multiple) circular structures, given in the form of a rather well-known proximity matrix in Table 2 (and called '`morse_digits`'). The later is a 10×10 proximity matrix for the ten Morse Code symbols that represent the first ten digits: (0: $- - - - -$; 1: $\bullet - - - -$; 2: $\bullet \bullet - - -$; 3: $\bullet \bullet \bullet - -$; 4: $\bullet \bullet \bullet \bullet -$; 5: $\bullet \bullet \bullet \bullet \bullet$; 6: $- \bullet \bullet \bullet \bullet$; 7: $- - \bullet \bullet \bullet$; 8: $- - - \bullet \bullet$; 9: $- - - - \bullet$). (Note that the labeling of objects in the output is from 1 to 10; thus, a translation back to the actual numbers corresponding to the Morse Code symbols requires a subtraction of one.) The entries in Table 2 have a dissimilarity interpretation and are defined for each object pair by 2.0 minus the sum of the two proportions for a group of subjects used by Rothkopf in the 1950's, representing 'same' judgments to the two symbols when given in the two possible presentation orders of the signals. Based on previous multidimensional scalings of the complete data set involving all of the Morse code symbols and in which the data of Table 2 are embedded, it might be expected that the symbols for the digits would form a clear linear unidimensional structure that would be interpretable according to a regular progression in the number of dots to

dashes. It turns out, as discussed in greater detail below, that a circular model (or better, the sum of two such circular models) is probably more consistent with the patterning of the proximities in Table 2 than are representations based on linear unidimensional scalings.

1.1.3 Hampshire County (in England) Inter-town Distances

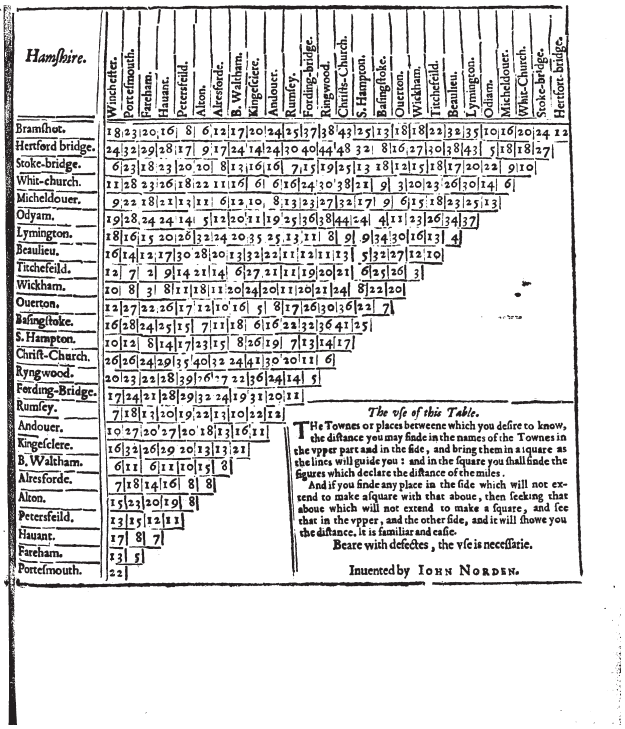
Table 3 provides a 27×27 dissimilarity matrix among twenty-seven Hampshire County towns (the first name listed is from Figure 1; current spellings are given in parentheses):

1: Bramfhot (Bramshott)
2: Hertford bridge (Hartfordbridge)
3: Stoke-bridge (Stockbridge)
4: Whit-church (Whitchurch)
5: Micheldouer (Micheldever)
6: Odyam (Odiham)
7: Lymington (Lymington)
8: Beaulieu (Beaulieu)
9: Titchefeild (Titchefield)
10: Wickham (Wickham)
11: Ouerton (Overton)
12: Bafingftoke (Basingstoke)
13: S. Hampton (Southhamption)
14: Chrift-Church (ChristChurch)
15: Ryngwood (Ringwood)
16: Fording-Bridge (Fordingbridge)
17: Rumfey (Romsey)
18: Andouer (Andover)
19: Kingefelere (Kingsclere)
20: B. Waltham (Bishops Waltham)
21: Alresforde (Alresford)
22: Alton (Alton)
23: Petersfeild (Petersfield)
24: Hauant (Havant)

25: Fareham (Fareham)
26: Portefmouth (Portesmouth)
27: Winchefter (Winchester)

These data are from John Norden's book (published 1625), *England, an Intended Guyde for English Travailers*. A facsimile page is given in Figure 1 (with its interesting reverse column order compared to Table 3); a (more or less current map) of the county appears in Figure 2 (Cobbett's Hampshire Map from 1830) that shows many of the twenty-seven county towns. Hampshire County is in south-central England and borders on the English Channel. It includes the port of Southampton (from whence the Titanic sailed on its maiden voyage (April 10, 1912; see Figure 3)). The `ascii` data file is called `hampshire_proximities.dat` and has verbatim contents as follows:

```
 0 12 24 20 16 10 35 32 22 18 18 13 25 43 38 37 25 24 20 17 12  6  8 16 20 23 18
12  0 27 18 18  5 43 38 30 27 16  8 32 48 44 40 30 24 14 24 17  9 17 28 29 32 24
24 27  0 10  9 22 20 17 18 15 12 18 13 25 19 15  7 16 16 13  8 20 20 23 18 23  6
20 18 10  0  6 14 30 26 23 20  3  9 21 38 30 24 16  6  6 16 11 22 18 26 23 28 11
16 18  9  6  0 13 25 23 18 15  6  9 17 32 27 23 13  8 10 12  6 11 13 21 18 22  9
10  5 22 14 13  0 37 34 26 23 11  4 24 44 38 36 25 19 11 20 12  5 14 24 24 28 19
35 43 20 30 25 37  0  4 13 16 30 34  9  9  8 11 13 25 35 20 24 32 26 20 15 16 18
32 38 17 26 23 34  4  0 10 12 27 32  5 13 11 12 11 22 32 13 20 28 30 17 12 14 16
22 30 18 23 18 26 13 10  0  3 26 25  6 21 20 19 11 21 27  6 14 21 14  9  2  7 12
18 27 15 20 15 23 16 12  3  0 20 22  8 24 21 20 11 20 24 20 11 18 11  8  3  8 10
18 16 12  3  6 11 30 27 26 20  0  7 22 36 30 26 17  8  5 16 10 12 17 26 22 27 12
13  8 18  9  9  4 34 32 25 22  7  0 25 41 36 32 22 16  6 18 11  7 15 25 24 28 16
25 32 13 21 17 24  9  5  6  8 22 25  0 17 14 13  7 19 26  8 15 23 17 14  8 12 10
43 48 25 38 32 44  9 13 21 24 36 41 17  0  6 11 20 30 41 24 32 40 35 29 24 26 26
38 44 19 30 27 38  8 11 20 21 30 36 14  6  0  5 14 24 36 22 27 26 39 28 22 23 20
37 40 15 24 23 36 11 12 19 20 26 32 13 11  5  0 11 20 31 19 24 32 29 28 21 24 17
25 30  7 16 13 25 13 11 11 11 17 22  7 20 14 11  0 12 22 10 13 22 19 20 13 18  7
24 24 16  6  8 19 25 22 21 20  8 16 19 30 24 20 12  0 11 16 13 18 20 27 20 27 10
20 14 16  6 10 11 35 32 27 24  5  6 26 41 36 31 22 11  0 21 13 13 20 29 26 32 16
17 24 13 16 12 20 20 13  6 20 16 18  8 24 22 19 10 16 21  0  8 15 10 11  6 11  6
12 17  8 11  6 12 24 20 14 11 10 11 15 32 27 24 13 13 13  8  0  8  8 16 14 18  7
 6  9 20 22 11  5 32 28 21 18 12  7 23 40 26 32 22 18 13 15  8  0  8 19 20 23 15
 8 17 20 18 13 14 26 30 14 11 17 15 17 35 39 29 19 20 20 10  8  8  0 11 12 15 13
16 28 23 26 21 24 20 17  9  8 26 25 14 29 28 28 20 27 29 11 16 19 11  0  7  8 17
20 29 18 23 18 24 15 12  2  3 22 24  8 24 22 21 13 20 26  6 14 20 12  7  0  5 13
23 32 23 28 22 28 16 14  7  8 27 28 12 26 23 24 18 27 32 11 18 23 15  8  5  0 22
18 24  6 11  9 19 18 16 12 10 12 16 10 26 20 17  7 10 16  6  7 15 13 17 13 22  0
```



Hamſhire.

Winchester. Portesmouth. Fareham. Hauant. Petersfeild. Alton. Alresforde. B. Waltham. Kingesclere. Andouer. Rumsey. Fording-bridge. Ringwood. Christs-Church. S. Hampton. Basingstoke. Ouerton. Wickham. Titchefeild. Beaulieu. Lymington. Odiam. Micheldouer. Whit-Church. Stoke-bridge. Herford-bridge.

Bramshot. 18 23 20 16 8 6 12 17 20 24 25 37 38 43 25 13 18 18 22 32 35 10 16 20 24 12
Hertford bridge. 24 32 29 28 17 9 17 24 14 24 30 40 44 48 32 8 16 27 30 38 43 5 18 18 27
Stoke-bridge. 6 23 18 23 20 20 8 13 16 16 7 15 19 25 13 18 12 15 18 17 20 22 9 10
Whit-church. 11 28 23 26 18 22 11 16 6 16 24 30 38 21 9 3 20 23 26 30 14 6
Micheldouer. 9 22 18 21 13 11 6 12 10 8 13 23 27 32 17 9 6 15 18 23 25 13
Odyam. 19 28 24 24 14 5 12 20 11 19 25 36 38 44 24 4 11 23 26 34 37
Lymington. 18 16 15 20 26 32 24 20 35 25 13 11 8 9 9 34 30 16 13 4
Beaulieu. 16 14 12 17 30 28 20 13 32 22 11 12 11 13 5 32 27 12 10
Titchefeild. 12 7 2 9 14 21 14 6 27 21 11 19 20 21 6 25 26 3
Wickham. 10 8 3 8 11 18 11 20 24 20 11 20 21 24 8 22 20
Ouerton. 12 27 22 26 17 12 10 16 5 8 17 26 30 36 22 7
Basingstoke. 16 28 24 25 15 7 11 18 6 16 22 32 36 41 25
S. Hampton. 10 12 8 14 17 23 15 8 26 19 7 13 14 17
Christ-Church. 26 26 24 29 35 40 32 24 41 30 20 11 6
Ryngwood. 20 23 22 28 39 46 27 22 36 24 14 5
Fording-Bridge. 17 24 21 28 29 32 24 19 31 20 11
Rumsey. 7 18 12 20 19 22 13 10 22 12
Andouer. 10 27 20 27 20 18 13 16 11
Kingesclere. 16 32 26 29 20 13 13 21
B. Waltham. 6 11 6 11 10 15 8
Alresforde. 7 18 14 16 8 8
Alton. 15 23 20 19 8
Petersfeild. 13 15 12 11
Hauant. 17 8 7
Fareham. 13 5
Portesmouth. 22

The vſe of this Table.

THe Townes or places betweene which you desire to know, the distance you may finde in the names of the Townes in the vpper part and in the side, and bring them in a square as the lines will guide you: and in the square you shall finde the figures which declare the distance of the miles.

And if you finde any place in the side which will not extend to make a square with that aboue, then seeking that aboue which will not extend to make a square, and see that in the vpper, and the other side, and it will showe you the distance, it is familiar and easie.

Beare with defectes, the vse is necessarie.

Inuented by IOHN NORDEN.

3 MR 62

Figure 1: Facsimile of John Norden's Distance Map for Hampshire County (1625).

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0.0	7.6	6.9	5.4	6.2	7.5	6.4	7.1	6.2	7.5	7.1	8.2	6.0	7.7	6.8	6.9	4.8	7.5
2	7.6	0.0	7.0	7.0	5.8	8.1	6.9	7.9	5.8	7.9	6.5	8.0	7.6	5.7	6.6	6.4	4.7	6.2
3	6.9	7.0	0.0	3.0	5.9	7.6	7.4	7.9	3.3	8.3	6.5	7.9	7.7	7.0	6.1	6.7	7.0	7.1
4	5.4	7.0	3.0	0.0	7.0	8.3	6.1	7.4	3.8	7.3	6.5	7.7	7.2	6.3	5.4	5.7	5.8	6.1
5	6.2	5.8	5.9	7.0	0.0	8.5	5.3	6.5	3.3	7.7	6.5	8.2	7.9	7.0	2.3	6.9	7.2	7.3
6	7.5	8.1	7.6	8.3	8.5	0.0	7.5	4.4	5.8	6.9	8.1	4.1	3.0	8.2	8.1	7.6	6.7	8.0
7	6.4	6.9	7.4	6.1	5.3	7.5	0.0	7.7	8.0	8.0	4.0	8.2	7.5	3.8	7.5	6.8	5.3	3.0
8	7.1	7.9	7.9	7.4	6.5	4.4	7.7	0.0	8.1	6.1	6.9	3.8	4.8	8.4	8.0	6.7	7.7	7.4
9	6.2	5.8	3.3	3.8	3.3	5.8	8.0	8.1	0.0	7.8	6.5	7.1	6.3	7.0	2.4	6.5	7.1	7.0
10	7.5	7.9	8.3	7.3	7.7	6.9	8.0	6.1	7.8	0.0	8.0	4.3	4.1	7.7	8.6	7.0	6.9	7.4
11	7.1	6.5	6.5	6.5	6.5	8.1	4.0	6.9	6.5	8.0	0.0	6.9	8.3	7.1	6.3	2.7	6.8	4.3
12	8.2	8.0	7.9	7.7	8.2	4.1	8.2	3.8	7.1	4.3	6.9	0.0	7.1	8.1	7.9	7.0	7.8	7.9
13	6.0	7.6	7.7	7.2	7.9	3.0	7.5	4.8	6.3	4.1	8.3	7.1	0.0	7.1	4.9	7.4	5.7	7.8
14	7.7	5.7	7.0	6.3	7.0	8.2	3.8	8.4	7.0	7.7	7.1	8.1	7.1	0.0	7.1	6.4	7.1	2.5
15	6.8	6.6	6.1	5.4	2.3	8.1	7.5	8.0	2.4	8.6	6.3	7.9	4.9	7.1	0.0	5.5	7.5	7.8
16	6.9	6.4	6.7	5.7	6.9	7.6	6.8	6.7	6.5	7.0	2.7	7.0	7.4	6.4	5.5	0.0	5.8	7.8
17	4.8	4.7	7.0	5.8	7.2	6.7	5.3	7.7	7.1	6.9	6.8	7.8	5.7	7.1	7.5	5.8	0.0	7.0
18	7.5	6.2	7.1	6.1	7.3	8.0	3.0	7.4	7.0	7.4	4.3	7.9	7.8	2.5	7.8	7.8	7.0	0.0

Table 1: Dissimilarities Among Eighteen Risks.

Table 2: A Proximity Matrix, morse_digits, for the Ten Morse Code Symbols Representing the First Ten Digits.

```
0.00  .75 1.69 1.87 1.76 1.77 1.59 1.26  .86  .95
 .75 0.00  .82 1.54 1.85 1.72 1.51 1.50 1.45 1.63
1.69  .82 0.00 1.25 1.47 1.33 1.66 1.57 1.83 1.81
1.87 1.54 1.25 0.00  .89 1.32 1.53 1.74 1.85 1.86
1.76 1.85 1.47  .89 0.00 1.41 1.64 1.81 1.90 1.90
1.77 1.72 1.33 1.32 1.41 0.00  .70 1.56 1.84 1.64
1.59 1.51 1.66 1.53 1.64  .70 0.00  .70 1.38 1.70
1.26 1.50 1.57 1.74 1.81 1.56  .70 0.00  .83 1.22
 .86 1.45 1.83 1.85 1.90 1.84 1.38  .83 0.00  .41
 .95 1.63 1.81 1.86 1.90 1.64 1.70 1.22  .41 0.00
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1	0	12	24	20	16	10	35	32	22	18	18	13	25	43	38	37	25	24	20	17	12	6	8	16	20	23	18
2	12	0	27	18	18	5	43	38	30	27	16	8	32	48	44	40	30	24	14	24	17	9	17	28	29	32	24
3	24	27	0	10	9	22	20	17	18	15	12	18	13	25	19	15	7	16	16	13	8	20	20	23	18	23	6
4	20	18	10	0	6	14	30	26	23	20	3	9	21	38	30	24	16	6	6	16	11	22	18	26	23	28	11
5	16	18	9	6	0	13	25	23	18	15	6	9	17	32	27	23	13	8	10	12	6	11	13	21	18	22	9
6	10	5	22	14	13	0	37	34	26	23	11	4	24	44	38	36	25	19	11	20	12	5	14	24	24	28	19
7	35	43	20	30	25	37	0	4	13	16	30	34	9	9	8	11	13	25	35	20	24	32	26	20	15	16	18
8	32	38	17	26	23	34	4	0	10	12	27	32	5	13	11	12	11	22	32	13	20	28	30	17	12	14	16
9	22	30	18	23	18	26	13	10	0	3	26	25	6	21	20	19	11	21	27	6	14	21	14	9	2	7	12
10	18	27	15	20	15	23	16	12	3	0	20	22	8	24	21	20	11	20	24	20	11	18	11	8	3	8	10
11	18	16	12	3	6	11	30	27	26	20	0	7	22	36	30	26	17	8	5	16	10	12	17	26	22	27	12
12	13	8	18	9	9	4	34	32	25	22	7	0	25	41	36	32	22	16	6	18	11	7	15	25	24	28	16
13	25	32	13	21	17	24	9	5	6	8	22	25	0	17	14	13	7	19	26	8	15	23	17	14	8	12	10
14	43	48	25	38	32	44	9	13	21	24	36	41	17	0	6	11	20	30	41	24	32	40	35	29	24	26	26
15	38	44	19	30	27	38	8	11	20	21	30	36	14	6	0	5	14	24	36	22	27	26	39	28	22	23	20
16	37	40	15	24	23	36	11	12	19	20	26	32	13	11	5	0	11	20	31	19	24	32	29	28	21	24	17
17	25	30	7	16	13	25	13	11	11	11	17	22	7	20	14	11	0	12	22	10	13	22	19	20	13	18	7
18	24	24	16	6	8	19	25	22	21	20	8	16	19	30	24	20	12	0	11	16	13	18	20	27	20	27	10
19	20	14	16	6	10	11	35	32	27	24	5	6	26	41	36	31	22	11	0	21	13	13	20	29	26	32	16
20	17	24	13	16	12	20	20	13	6	20	16	18	8	24	22	19	10	16	21	0	8	15	10	11	6	11	6
21	12	17	8	11	6	12	24	20	14	11	10	11	15	32	27	24	13	13	13	8	0	8	8	16	14	18	7
22	6	9	20	22	11	5	32	28	21	18	12	7	23	40	26	32	22	18	13	15	8	0	8	19	20	23	15
23	8	17	20	18	13	14	26	30	14	11	17	15	17	35	39	29	19	20	20	10	8	8	0	11	12	15	13
24	16	28	23	26	21	24	20	17	9	8	26	25	14	29	28	28	20	27	29	11	16	19	11	0	7	8	17
25	20	29	18	23	18	24	15	12	2	3	22	24	8	24	22	21	13	20	26	6	14	20	12	7	0	5	13
26	23	32	23	28	22	28	16	14	7	8	27	28	12	26	23	24	18	27	32	11	18	23	15	8	5	0	22
27	18	24	6	11	9	19	18	16	12	10	12	16	10	26	20	17	7	10	16	6	7	15	13	17	13	22	0

Table 3: A Dissimilarity Matrix Among Twenty-seven Hampshire County Towns.

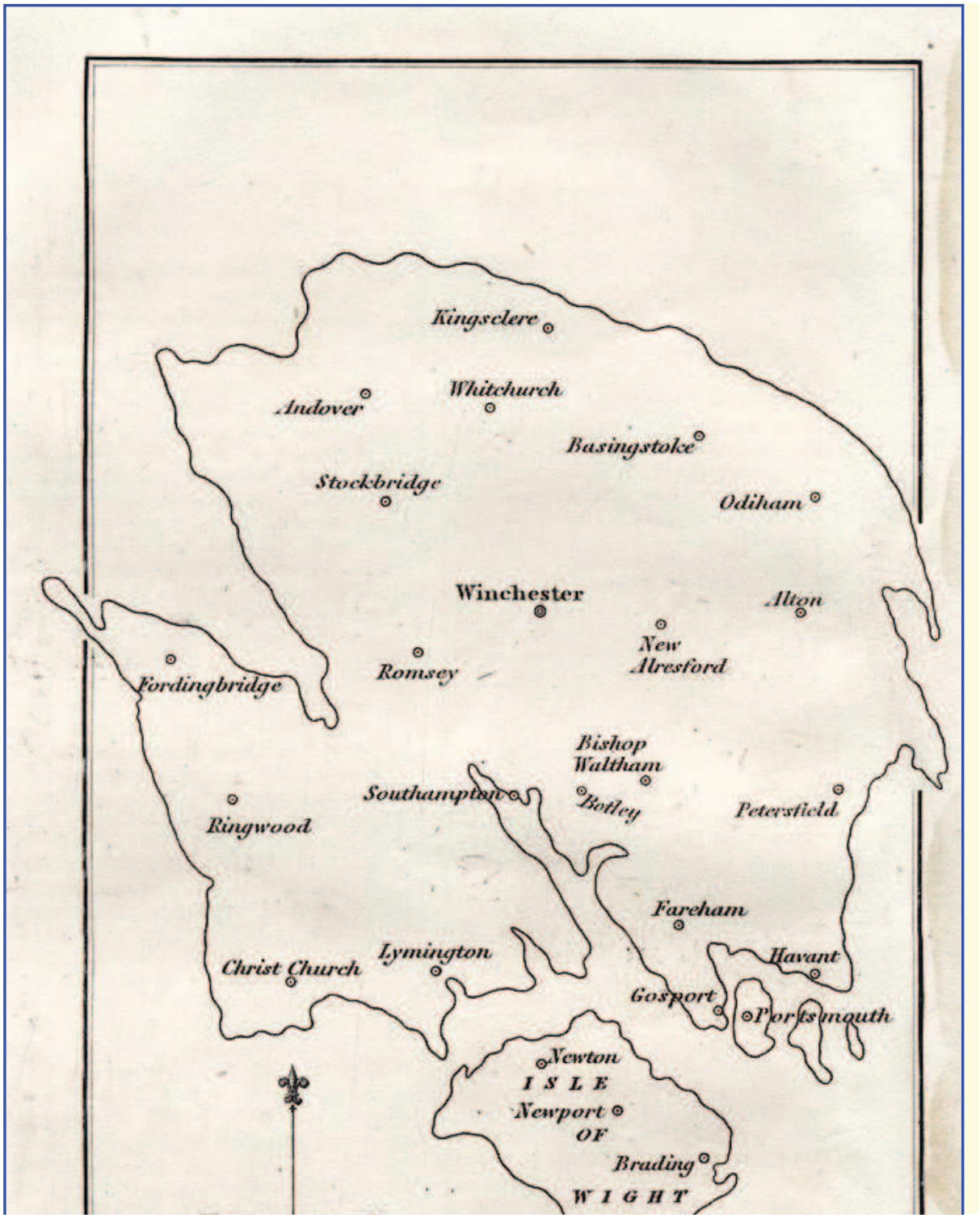


Figure 2: Cobbett's 1830 Map of Hampshire County.

http://www.maritimequest.com/liners/titanic/photos/art/04_titanic.jpg

http://www.maritimequest.com/liners/titanic/photos/art/04_titanic.jpg [3/28/2007 2:11:52 PM]

Figure 3: A Famous Sailing from Southampton.

concepts	1(a)	2(b)	3(c)	4(d)	5(e)	6(f)	7(g)	8(h)	9(i)	10(j)	11(k)	12(l)	13(m)	14(n)	15(o)
scales															
1	4.0	1.0	7.0	7.0	1.0	1.0	1.0	7.0	6.5	2.5	3.5	6.0	7.0	4.0	1.0
2	7.0	7.0	1.0	1.0	7.0	7.0	4.5	1.0	2.0	7.0	4.5	1.5	1.0	7.0	7.0
3	1.0	1.0	7.0	7.0	4.0	2.5	4.0	7.0	6.0	4.0	2.0	6.5	1.0	1.5	1.0
4	1.0	7.0	7.0	4.0	5.5	4.0	4.0	7.0	6.5	2.5	6.0	6.5	1.0	4.0	1.0
5	7.0	4.0	1.0	1.0	7.0	1.0	1.0	1.0	1.5	7.0	2.0	1.0	4.0	4.5	4.0
6	2.0	1.0	7.0	7.0	4.0	1.5	7.0	7.0	5.0	4.0	4.5	6.0	7.0	1.5	4.0
7	1.0	7.0	7.0	7.0	1.0	1.0	4.0	7.0	6.0	4.0	5.0	6.5	1.0	1.0	1.0
8	7.0	7.0	1.5	1.0	7.0	7.0	7.0	1.0	2.0	5.5	3.5	2.0	1.0	4.0	7.0
9	7.0	7.0	1.0	1.0	7.0	4.0	7.0	1.0	1.5	4.0	2.0	1.5	7.0	4.0	7.0
10	7.0	7.0	1.0	1.0	7.0	7.0	1.0	1.0	1.5	7.0	4.5	1.5	1.0	4.0	7.0

Table 4: A Two-Mode Dissimilarity Matrix for Eve Black Between Ten Scales and Fifteen Concepts.

1.1.4 Semantic Differential Data on the "Three Faces of Eve"

To have available a two-mode (10×15) matrix for purposes of illustration, we chose the first replication of Eve Black's Semantic Differential data given in Table 4 (`eve_black_one.dat`). There are six such exemplars available in the directory, `multistructuralanalysis_mfiles`:

`eve_white_one.dat`; `eve_white_two.dat`; `eve_black_one.dat` `eve_black_two.dat`; `jane_one.dat`; `jane_two.dat`

Readers are welcome to experiment with these as they see fit. All have the same ten rows (or scales) with the table entries being dissimilarities for the left-most member of the bipolar pair:

1) cold-hot
2) valuable-worthless
3) tense-relaxed
4) small-large
5) fast-slow
6) dirty-clean
7) weak-strong
8) tasty-distasteful
9) deep-shallow
10) active-passive

The fifteen columns correspond to the concepts being rated:

1(a): love
2(b): child
3(c): my doctor
4(d): me
5(e): my job
6(f): mental sickness
7(g): my mother
8(h): peace of mind
9(i): fraud
10(j): my spouse
11(k): self-control
12(l): hatred

13(m): my father
14(n): confusion
15(o): sex

The actual `ascii` data file, `eve_black_one.dat`, has verbatim contents as follows:

```
4.0 1.0 7.0 7.0 1.0 1.0 1.0 7.0 6.5 2.5 3.5 6.0 7.0 4.0 1.0
7.0 7.0 1.0 1.0 7.0 7.0 4.5 1.0 2.0 7.0 4.5 1.5 1.0 7.0 7.0
1.0 1.0 7.0 7.0 4.0 2.5 4.0 7.0 6.0 4.0 2.0 6.5 1.0 1.5 1.0
1.0 7.0 7.0 4.0 5.5 4.0 4.0 7.0 6.5 2.5 6.0 6.5 1.0 4.0 1.0
7.0 4.0 1.0 1.0 7.0 1.0 1.0 1.0 1.5 7.0 2.0 1.0 4.0 4.5 4.0
2.0 1.0 7.0 7.0 4.0 1.5 7.0 7.0 5.0 4.0 4.5 6.0 7.0 1.5 4.0
1.0 7.0 7.0 7.0 1.0 1.0 4.0 7.0 6.0 4.0 5.0 6.5 1.0 1.0 1.0
7.0 7.0 1.5 1.0 7.0 7.0 7.0 1.0 2.0 5.5 3.5 2.0 1.0 4.0 7.0
7.0 7.0 1.0 1.0 7.0 4.0 7.0 1.0 1.5 4.0 2.0 1.5 7.0 4.0 7.0
7.0 7.0 1.0 1.0 7.0 7.0 1.0 1.0 1.5 7.0 4.5 1.5 1.0 4.0 7.0
```

To provide some background for these data, we give a short plot summary for the 1957 movie, *The Three Faces of Eve*; Joanne Woodward won an Oscar for her portrayal of Eve White/Eve Black/Jane:

Eve White is a quite, mousy, unassuming wife and mother who keeps suffering from headaches and occasional black outs. Eventually she is sent to see psychiatrist Dr. Luther, and, while under hypnosis, a whole new personality emerges: the racy, wild, fun-loving Eve Black. Under continued therapy, yet a third personality appears, the relatively stable Jane. This film, based on the true-life case of a multiple personality, chronicles Dr. Luther's attempts to reconcile the three faces of Eve's multiple personalities.

The real reason, however, that this particular data matrix exemplar is used from among the six we have available, is to give two quotes from the movie:

Ralph White: I've never seen you take a drink in your life.
Eve Black: Honey, there are a lot of things you ain't never seen me do; that's no sign I don't do 'em.

The Soldier: When I spend eight bucks on a dame, I don't just go home with the morning paper, y'know what I mean?

2 The Primary Multistructural Analysis Routines

Each of the first seven subsections to follow discusses one of the major multistructural approaches for symmetric proximity matrices: anti-Robinson forms; ultrametrics; additive trees; metric and nonmetric city-block scaling; and finally, metric and nonmetric Euclidean scaling. In all cases, the example data set used is the 18×18 dissimilarity matrix among risks given in Table 1, so the various analyses can be compared. Generally, the structure of each of the first five subsections is the same:

(a) a running of a script that involves the raw M-file and 100 random starts (for: `biarobfnd.m`; `biultrafnd.m`; `biatreefnd.m`; `biscalqa.m`; and `bimonscalqa.m`). The VAFs are sorted and listed to indicate the range of local optimality achieved.

(b) a running of a script in which the major components of the M-file have been recompiled as Fortran executable subroutines (using `matlab2fmex`) and 1000 random starts (for comparable routines with an added suffix of `ms_`):
`ms_biarobfnd.m`; `ms_biultrafnd.m`; `ms_biatreefnd.m`; `ms_biscalqa.m`; and `ms_bimonscalqa.m`
Only the results for the best VAF achieved is given. Generally, based on the timings listed (using `tic;` and `toc;` in the various scripts), it takes a comparable time to run 1000 Fortran enhanced routines as it does to carry out 100 starts of the raw M-files.

(c) the script used in (b) is listed; the comparable script for (a) would involve the non-Fortran enhanced M-file (without the `ms_` suffix) and would not suppress the listing of the sorted VAFs. These scripts are given in the same directory where all the M-files are available for this Multistructural Toolbox (cited earlier).

The last two subsections incorporate the metric and nonmetric versions of the MATLAB routine, `mdscale.m`, from the Statistics Toolbox. Both of the two scripts used provide 100 random starts for these routines, and give the sorted VAFs.

2.1 (Multistructural) Anti-Robinson Forms

As can be seen from the results below, the best achievable VAF for the sum of two anti-Robinson (AR) forms is 96.44%, and (probably) represents some general upper-bound on how well one might do in accounting for the information in the `risk_rate` proximity matrix by the sum of two structures. Remember that both ultrametrics and linear unidimensional scales themselves have AR forms (albeit, much more restricted). The range of VAFs achieved over the 100 random starts is rather narrow (from .9373 to .9644; the latter value is also the best found over the 1000 random starts), and demonstrates the power of the combinatorial optimization routines being used (this, as we will see, is in contrast to gradient-based methods in the sections on metric and nonmetric Euclidean scaling). Also, the combinatorial optimization strategies implemented are remarkably fast.

In interpreting the results of a bistructural AR analysis, it generally seems easiest to interpret the first AR structure as being dominant and representing the major gradient to be interpreted in its row/column reordering. The second AR structure (which is also fit secondly in the optimization/residualization process) is typically more peripheral and reflects adjustments for some object pairs not fit as well by the dominant AR form. We first give below the labeled ordering of the objects for the dominant structure and an indication of meanings for object subsets that span a certain contiguous portion of the ordering; a few subsets are listed that are particularly salient with diameters at 3.00 or less.

10: lung cancer
12: stomach cancer
8: leukaemia
6: heart disease
13: stroke
17: traffic accidents
1: accidental falls
16: toxic chemical spill
4: fire
3: electrocution

9: lightning
15: tornados
5: flood
2: airplane accidents
11: nuclear accident
7: homicide
18: war
14: terrorism

{10,12,8}: cancer-like
{6,13}: coronary diseases (2.20)
{17,1,16}: labeled accidental
{4,3}: usually accidental (2.57)
{3,9}: electrical
{15,5}: water related
{9,15,5}: storm related; acts of nature; absence of group control (2.87)
{2,11}: accidental technological acts
{7,18,14}: nonaccidental violent acts
{18,14}: aggressive violent acts (2.92)
{10,12,8,6,13}: diseases; risks internal to person; health related; absence of control
{17,1,16,4,3,9}: risks external to person; accidental with possibility of avoidance by individual control
{2,11,7,18,14}: acts of people; possible avoidance by group control

The objects are labeled below according to the secondary structure, and when possible, labels are given for a few of the adjustment regions (i.e., contiguous object subsets in the ordering) with diameters less than -1.00).

15: tornados
13: stroke
10: lung cancer
1: accidental falls
5: flood
7: homicide

4: fire
14: terrorism
17: traffic accidents
2: airplane accidents
18: war
8: leukaemia
16: toxic chemical spill
11: nuclear accident
12: stomach cancer
9: lightning
6: heart disease
3: electrocution

{15,13}: (-2.12) apparent suddenness of onset
{13,10}: (-2.60) apparent suddenness of onset
{5,7}: (-1.65)
{7,4,14,17}: (-1.03) homicide and vehicle homicides
{17,2}: (-2.32) vehicle involvement
{16,11}: (-4.23) release of harmful material
{9,6}: (-1.69) apparent suddenness of onset

2.1.1 Basic Script Results for biarobfnd.m

```
>> script_biarobfnd

sorted_vafs =

  Columns 1 through 15

    0.9373    0.9406    0.9406    0.9430    0.9430    0.9430    0.9430    0.9447    0.9448    0.9448    0.9448    0.9448    0.9448    0.9448    0.9448

  Columns 16 through 30

    0.9448    0.9448    0.9452    0.9455    0.9455    0.9455    0.9455    0.9455    0.9455    0.9465    0.9465    0.9465    0.9476    0.9477    0.9477

  Columns 31 through 45

    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487    0.9487

  Columns 46 through 60

    0.9487    0.9487    0.9487    0.9487    0.9487    0.9504    0.9504    0.9504    0.9504    0.9504    0.9504    0.9504    0.9504    0.9504    0.9504

  Columns 61 through 75

    0.9504    0.9504    0.9504    0.9511    0.9511    0.9511    0.9511    0.9511    0.9511    0.9511    0.9511    0.9511    0.9511    0.9525    0.9525

  Columns 76 through 90

    0.9525    0.9525    0.9525    0.9525    0.9525    0.9527    0.9527    0.9527    0.9527    0.9527    0.9527    0.9548    0.9550    0.9550    0.9550

  Columns 91 through 100

    0.9550    0.9571    0.9571    0.9571    0.9613    0.9613    0.9644    0.9644    0.9644    0.9644
```

```
best_vaf =

    0.9644


best_find =

  Columns 1 through 15

         0    6.9841    6.8662    5.7461    6.2000    7.5028    6.4000    7.3388    6.4536    7.3340    7.2001    7.8591    6.0000    7.5581    6.8001
    6.9841         0    6.7783    6.5375    5.7669    7.9746    6.8613    7.7437    6.1409    7.8392    6.1865    8.0000    7.6005    5.9181    6.2920
    6.8662    6.7783         0    3.0001    5.9028    7.6000    7.4000    7.9076    3.3000    8.3338    6.7235    7.8528    7.7000    7.3728    6.1000
    5.7461    6.5375    3.0001         0    6.6539    7.9212    5.9181    7.3962    3.8002    7.3962    6.8671    7.7180    6.7754    6.2042    5.4006
    6.2000    5.7669    5.9028    6.6539         0    8.2944    5.3000    7.3962    3.3001    7.7797    6.4313    8.2000    7.5649    6.8613    2.3001
    7.5028    7.9746    7.6000    7.9212    8.2944         0    7.9882    4.4013    5.8000    6.9028    7.9197    4.1000    3.0028    8.2004    8.2944
    6.4000    6.8613    7.4000    5.9181    5.3000    7.9882         0    7.7437    7.3864    7.7797    4.0000    8.2000    7.6318    3.7997    7.3864
    7.3388    7.7437    7.9076    7.3962    7.3962    4.4013    7.7437         0    7.7702    6.1000    7.0049    3.7990    4.8000    7.9592    7.9212
    6.4536    6.1409    3.3000    3.8002    3.3001    5.8000    7.3864    7.7702         0    7.9212    6.5000    7.1000    7.1318    7.2353    2.4000
    7.3340    7.8392    8.3338    7.3962    7.7797    6.9028    7.7797    6.1000    7.9212         0    8.0551    4.3001    4.1000    7.9592    8.3047
    7.2001    6.1865    6.7235    6.8671    6.4313    7.9197    4.0000    7.0049    6.5000    8.0551         0    6.9000    7.9882    6.8920    6.6809
    7.8591    8.0000    7.8528    7.7180    8.2000    4.1000    8.2000    3.7990    7.1000    4.3001    6.9000         0    7.1001    8.2809    7.9212
    6.0000    7.6005    7.7000    6.7754    7.5649    3.0028    7.6318    4.8000    7.1318    4.1000    7.9882    7.1001         0    7.7268    4.9000
    7.5581    5.9181    7.3728    6.2042    6.8613    8.2004    3.7997    7.9592    7.2353    7.9592    6.8920    8.2809    7.7268         0    7.3864
    6.8001    6.2920    6.1000    5.4006    2.3001    8.2944    7.3864    7.9212    2.4000    8.3047    6.6809    7.9212    4.9000    7.3864         0
    6.2040    6.8364    6.2749    5.6996    6.9000    7.6013    6.8613    6.5951    6.1374    7.6095    2.7000    7.0000    7.1318    7.1474    6.2885
    4.7996    4.7000    7.1182    5.6635    6.9246    6.7004    5.9814    7.2793    6.9807    7.2793    6.9553    7.6011    5.7005    6.6149    7.4497
    7.4987    6.2000    7.3728    6.5625    6.8613    8.0013    2.9996    7.4000    7.2353    7.8392    4.2992    7.9010    7.8000    2.5000    7.3864

  Columns 16 through 18

    6.2040    4.7996    7.4987
    6.8364    4.7000    6.2000
    6.2749    7.1182    7.3728
    5.6996    5.6635    6.5625
    6.9000    6.9246    6.8613
    7.6013    6.7004    8.0013
    6.8613    5.9814    2.9996
    6.5951    7.2793    7.4000
    6.1374    6.9807    7.2353
    7.6095    7.2793    7.8392
    2.7000    6.9553    4.2992
    7.0000    7.6011    7.9010
    7.1318    5.7005    7.8000
    7.1474    6.6149    2.5000
    6.2885    7.4497    7.3864
         0    5.9286    7.1474
    5.9286         0    7.0000
    7.1474    7.0000         0


best_targone =

  Columns 1 through 15

         0    3.8662    6.0349    6.0958    6.7002    7.3705    7.4252    7.4252    7.4874    7.4874    7.4874    7.8709    7.8709    7.8709    7.8709
    3.8662         0    3.7330    4.2500    6.6662    7.3705    7.4252    7.4252    7.4874    7.4874    7.4874    7.4874    7.7868    7.8349    7.8349
    6.0349    3.7330         0    4.0359    4.6271    7.3705    7.3705    7.4252    7.4874    7.4874    7.4874    7.4874    7.4874    7.8349    7.8349
    6.0958    4.2500    4.0359         0    2.1958    6.2802    6.6958    7.2359    7.4874    7.4874    7.4874    7.4874    7.4874    7.5543    7.5543
    6.7002    6.6662    4.6271    2.1958         0    5.6229    5.9665    6.6979    6.6979    6.6979    6.6979    7.0158    7.4874    7.5229    7.5543
    7.3705    7.3705    7.3705    6.2802    5.6229         0    4.8908    6.0198    6.6979    6.6979    6.6979    7.0158    7.0158    7.0158    7.0158
    7.4252    7.4252    7.3705    6.6958    5.9665    4.8908         0    6.0198    6.0198    6.0198    6.0198    6.3662    6.9379    7.0158    7.0158
    7.4252    7.4252    7.4252    7.2359    6.6979    6.0198    6.0198         0    5.7908    5.8546    5.8546    5.8546    6.9276    6.9276    6.9276
    7.4874    7.4874    7.4874    7.4874    6.6979    6.6979    6.0198    5.7908         0    2.5662    3.5174    4.9667    6.9276    6.9276    6.9276
    7.4874    7.4874    7.4874    7.4874    6.6979    6.6979    6.0198    5.8546    2.5662         0    3.1167    4.9667    5.0958    6.3580    6.3580
    7.4874    7.4874    7.4874    7.4874    6.6979    6.6979    6.0198    5.8546    3.5174    3.1167         0    1.8234    2.8662    5.8581    6.3134
    7.8709    7.4874    7.4874    7.4874    7.0158    7.0158    6.3662    5.8546    4.9667    4.9667    1.8234         0    1.8662    5.8581    6.2470
    7.8709    7.7868    7.4874    7.4874    7.4874    7.0158    6.9379    6.9276    6.9276    5.0958    2.8662    1.8662         0    5.8581    6.2470
    7.8709    7.8349    7.8349    7.5543    7.5229    7.0158    7.0158    6.9276    6.9276    6.3580    5.8581    5.8581    5.8581         0    6.2470
    7.8709    7.8349    7.8349    7.5543    7.5543    7.0158    7.0158    6.9276    6.9276    6.3580    6.3134    6.2470    6.2470    6.2470         0
    7.8709    7.8349    7.8349    7.5543    7.5543    7.0158    7.0158    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525    3.9319
    7.8709    7.8349    7.8349    7.6359    7.6271    7.5304    7.5304    7.2386    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525    4.3598
    8.0504    8.0504    8.0504    7.7802    7.6493    7.6493    7.6493    7.2386    7.2386    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525

  Columns 16 through 18

    7.8709    7.8709    8.0504
    7.8349    7.8349    8.0504
    7.8349    7.8349    8.0504
    7.5543    7.6359    7.7802
    7.5543    7.6271    7.6493
    7.0158    7.5304    7.6493
    7.0158    7.5304    7.6493
    6.9525    7.2386    7.2386
    6.9525    6.9525    7.2386
    6.9525    6.9525    6.9525
```

```
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    3.9319    4.3598    6.9525
         0    3.0908    4.8341
    3.0908         0    2.9217
    4.8341    2.9217         0


best_targtwo =

  Columns 1 through 15

         0   -2.1158    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339
   -2.1158         0   -2.6002    0.0335    0.0775    0.0775    0.0775    0.0775    0.0775    0.0775    0.1729    0.1729    0.4339    0.4339    0.4339
    0.4339   -2.6002         0   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0317   -0.0317    0.0651    0.1843    0.1843    0.4339
    0.4339    0.0335   -0.0912         0   -0.7379   -0.6158   -0.2737   -0.0912   -0.0912   -0.0317   -0.0317   -0.0317    0.1843    0.1843    0.4339
    0.4339    0.0775   -0.0912   -0.7379         0   -1.6525   -0.2737   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0276    0.1843    0.4132
    0.4339    0.0775   -0.0912   -0.6158   -1.6525         0   -1.0344   -1.0344   -1.0344   -0.0912   -0.0912   -0.0912   -0.0912    0.0681    0.3651
    0.4339    0.0775   -0.0912   -0.2737   -0.2737   -1.0344         0   -1.0344   -1.0344   -0.3901   -0.3901   -0.0912   -0.0912   -0.0605    0.2306
    0.4339    0.0775   -0.0912   -0.0912   -0.0912   -1.0344   -1.0344         0   -1.0344   -1.0344   -0.4217   -0.0912   -0.0912   -0.0605    0.2306
    0.4339    0.0775   -0.0912   -0.0912   -0.0912   -1.0344   -1.0344   -1.0344         0   -2.3158   -0.5304   -0.0912   -0.0912   -0.0605    0.2306
    0.4339    0.0775   -0.0317   -0.0317   -0.0912   -0.0912   -0.3901   -1.0344   -2.3158         0   -0.7525   -0.0912   -0.0912   -0.0605    0.1651
    0.4339    0.1729   -0.0317   -0.0317   -0.0912   -0.0912   -0.3901   -0.4217   -0.5304   -0.7525         0   -0.4349   -0.0912   -0.0605    0.0661
    0.4339    0.1729    0.0651   -0.0317   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.4349         0   -0.8301   -0.8301    0.0661
    0.4339    0.4339    0.1843    0.1843   -0.0276   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.8301         0   -4.2276   -0.4252
    0.4339    0.4339    0.1843    0.1843    0.1843    0.0681   -0.0605   -0.0605   -0.0605   -0.0605   -0.0605   -0.8301   -4.2276         0   -0.9349
    0.4339    0.4339    0.4339    0.4339    0.4132    0.3651    0.2306    0.2306    0.2306    0.1651    0.0661    0.0661   -0.4252   -0.9349         0
    0.5766    0.4339    0.4339    0.4339    0.4339    0.4339    0.2828    0.2828    0.2828    0.2828    0.2828    0.2828    0.2828    0.1866   -0.3874
    0.8070    0.8070    0.8070    0.8070    0.8070    0.4339    0.4339    0.4203    0.4203    0.4203    0.3655    0.3655    0.3655    0.3655   -0.1500
    1.1333    1.0021    0.8464    0.8464    0.8070    0.4475    0.4339    0.4203    0.4203    0.4203    0.4203    0.4203    0.4203    0.3655    0.3655

  Columns 16 through 18

    0.5766    0.8070    1.1333
    0.4339    0.8070    1.0021
    0.4339    0.8070    0.8464
    0.4339    0.8070    0.8464
    0.4339    0.8070    0.8070
    0.4339    0.4339    0.4475
    0.2828    0.4339    0.4339
    0.2828    0.4203    0.4203
    0.2828    0.4203    0.4203
    0.2828    0.4203    0.4203
    0.2828    0.3655    0.4203
    0.2828    0.3655    0.4203
    0.2828    0.3655    0.4203
    0.1866    0.3655    0.3655
   -0.3874   -0.1500    0.3655
         0   -1.6874    0.1833
   -1.6874         0    0.1126
    0.1833    0.1126         0


best_outpermone =

    10    12     8     6    13    17     1    16     4     3     9    15     5     2    11     7    18    14


best_outpermtwo =

    15    13    10     1     5     7     4    14    17     2    18     8    16    11    12     9     6     3

Elapsed time is 256.630044 seconds.
```

2.1.2 The Enhanced Script for ms_biarobfnd.m

```
load risk_rate.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(1000,1);

for i = 1:1000

    [find,vaf,targone,targtwo,outpermone,outpermtwo] = ...
        ms_biarobfnd(risk_rate,randperm(18),3);

    store_vaf(i) = vaf;
```

```
    if (vaf > best_vaf)

        best_vaf = vaf;
        best_find = find;
        best_targone = targone;
        best_targtwo = targtwo;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs;

best_vaf

best_find

best_targone

best_targtwo

best_outpermone

best_outpermtwo

toc;
```

2.1.3 Enhanced Script Results for ms_biarobfnd.m

```
>> ms_script_biarobfnd

best_vaf =

    0.9644


best_find =

  Columns 1 through 15

         0    6.9841    6.8662    5.7461    6.2000    7.5028    6.4000    7.3388    6.4536    7.3340    7.2001    7.8591    6.0000    7.5581    6.8001
    6.9841         0    6.7783    6.5375    5.7669    7.9746    6.8613    7.7437    6.1409    7.8392    6.1865    8.0000    7.6005    5.9181    6.2920
    6.8662    6.7783         0    3.0001    5.9028    7.6000    7.4000    7.9076    3.3000    8.3338    6.7235    7.8528    7.7000    7.3728    6.1000
    5.7461    6.5375    3.0001         0    6.6539    7.9212    5.9181    7.3962    3.8002    7.3962    6.8671    7.7180    6.7754    6.2042    5.4006
    6.2000    5.7669    5.9028    6.6539         0    8.2944    5.3000    7.3962    3.3001    7.7797    6.4313    8.2000    7.5649    6.8613    2.3001
    7.5028    7.9746    7.6000    7.9212    8.2944         0    7.9882    4.4013    5.8000    6.9028    7.9197    4.1000    3.0028    8.2004    8.2944
    6.4000    6.8613    7.4000    5.9181    5.3000    7.9882         0    7.7437    7.3864    7.7797    4.0000    8.2000    7.6318    3.7997    7.3864
    7.3388    7.7437    7.9076    7.3962    7.3962    4.4013    7.7437         0    7.7702    6.1000    7.0049    3.7990    4.8000    7.9592    7.9212
    6.4536    6.1409    3.3000    3.8002    3.3001    5.8000    7.3864    7.7702         0    7.9212    6.5000    7.1000    7.1318    7.2353    2.4000
    7.3340    7.8392    8.3338    7.3962    7.7797    6.9028    7.7797    6.1000    7.9212         0    8.0551    4.3001    4.1000    7.9592    8.3047
    7.2001    6.1865    6.7235    6.8671    6.4313    7.9197    4.0000    7.0049    6.5000    8.0551         0    6.9000    7.9882    6.8920    6.6809
    7.8591    8.0000    7.8528    7.7180    8.2000    4.1000    8.2000    3.7990    7.1000    4.3001    6.9000         0    7.1001    8.2809    7.9212
    6.0000    7.6005    7.7000    6.7754    7.5649    3.0028    7.6318    4.8000    7.1318    4.1000    7.9882    7.1001         0    7.7268    4.9000
    7.5581    5.9181    7.3728    6.2042    6.8613    8.2004    3.7997    7.9592    7.2353    7.9592    6.8920    8.2809    7.7268         0    7.3864
    6.8001    6.2920    6.1000    5.4006    2.3001    8.2944    7.3864    7.9212    2.4000    8.3047    6.6809    7.9212    4.9000    7.3864         0
    6.2040    6.8364    6.2749    5.6996    6.9000    7.6013    6.8613    6.5951    6.1374    7.6095    2.7000    7.0000    7.1318    7.1474    6.2885
    4.7996    4.7000    7.1182    5.6635    6.9246    6.7004    5.9814    7.2793    6.9807    7.2793    6.9553    7.6011    5.7005    6.6149    7.4497
    7.4987    6.2000    7.3728    6.5625    6.8613    8.0013    2.9996    7.4000    7.2353    7.8392    4.2992    7.9010    7.8000    2.5000    7.3864

  Columns 16 through 18

    6.2040    4.7996    7.4987
    6.8364    4.7000    6.2000
    6.2749    7.1182    7.3728
    5.6996    5.6635    6.5625
    6.9000    6.9246    6.8613
    7.6013    6.7004    8.0013
    6.8613    5.9814    2.9996
    6.5951    7.2793    7.4000
    6.1374    6.9807    7.2353
    7.6095    7.2793    7.8392
    2.7000    6.9553    4.2992
    7.0000    7.6011    7.9010
    7.1318    5.7005    7.8000
    7.1474    6.6149    2.5000
    6.2885    7.4497    7.3864
         0    5.9286    7.1474
    5.9286         0    7.0000
    7.1474    7.0000         0
```

```
best_targone =

  Columns 1 through 15

         0    3.8662    6.0349    6.0958    6.7002    7.3705    7.4252    7.4252    7.4874    7.4874    7.4874    7.8709    7.8709    7.8709    7.8709
    3.8662         0    3.7330    4.2500    6.6662    7.3705    7.4252    7.4252    7.4874    7.4874    7.4874    7.4874    7.7868    7.8349    7.8349
    6.0349    3.7330         0    4.0359    4.6271    7.3705    7.3705    7.4252    7.4874    7.4874    7.4874    7.4874    7.4874    7.8349    7.8349
    6.0958    4.2500    4.0359         0    2.1958    6.2802    6.6958    7.2359    7.4874    7.4874    7.4874    7.4874    7.4874    7.5543    7.5543
    6.7002    6.6662    4.6271    2.1958         0    5.6229    5.9665    6.6979    6.6979    6.6979    6.6979    6.6979    7.0158    7.4874    7.5229    7.5543
    7.3705    7.3705    7.3705    6.2802    5.6229         0    4.8908    6.0198    6.6979    6.6979    6.6979    7.0158    7.0158    7.0158    7.0158
    7.4252    7.4252    7.3705    6.6958    5.9665    4.8908         0    6.0198    6.0198    6.0198    6.0198    6.3662    6.9379    7.0158    7.0158
    7.4252    7.4252    7.4252    7.2359    6.6979    6.0198    6.0198         0    5.7908    5.8546    5.8546    5.8546    6.9276    6.9276    6.9276
    7.4874    7.4874    7.4874    7.4874    6.6979    6.6979    6.0198    5.7908         0    2.5662    3.5174    4.9667    6.9276    6.9276    6.9276
    7.4874    7.4874    7.4874    7.4874    6.6979    6.6979    6.0198    5.8546    2.5662         0    3.1167    4.9667    5.0958    6.3580    6.3580
    7.4874    7.4874    7.4874    7.4874    6.6979    6.6979    6.0198    5.8546    3.5174    3.1167         0    1.8234    2.8662    5.8581    6.3134
    7.8709    7.4874    7.4874    7.4874    7.0158    7.0158    6.3662    5.8546    4.9667    4.9667    1.8234         0    1.8662    5.8581    6.2470
    7.8709    7.7868    7.4874    7.4874    7.4874    7.0158    6.9379    6.9276    6.9276    5.0958    2.8662    1.8662         0    5.8581    6.2470
    7.8709    7.8349    7.8349    7.5543    7.5229    7.0158    7.0158    6.9276    6.9276    6.3580    5.8581    5.8581    5.8581         0    6.2470
    7.8709    7.8349    7.8349    7.5543    7.5543    7.0158    7.0158    6.9276    6.9276    6.3580    6.3134    6.2470    6.2470    6.2470         0
    7.8709    7.8349    7.8349    7.5543    7.5543    7.0158    7.0158    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525    3.9319
    7.8709    7.8349    7.8349    7.6359    7.6271    7.5304    7.5304    7.2386    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525    4.3598
    8.0504    8.0504    8.0504    7.7802    7.6493    7.6493    7.6493    7.2386    7.2386    6.9525    6.9525    6.9525    6.9525    6.9525    6.9525

  Columns 16 through 18

    7.8709    7.8709    8.0504
    7.8349    7.8349    8.0504
    7.8349    7.8349    8.0504
    7.5543    7.6359    7.7802
    7.5543    7.6271    7.6493
    7.0158    7.5304    7.6493
    7.0158    7.5304    7.6493
    6.9525    7.2386    7.2386
    6.9525    6.9525    7.2386
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    6.9525    6.9525    6.9525
    3.9319    4.3598    6.9525
         0    3.0908    4.8341
    3.0908         0    2.9217
    4.8341    2.9217         0


best_targtwo =

  Columns 1 through 15

         0   -2.1158    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339    0.4339
   -2.1158         0   -2.6002    0.0335    0.0775    0.0775    0.0775    0.0775    0.0775    0.0775    0.1729    0.1729    0.4339    0.4339    0.4339
    0.4339   -2.6002         0   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0317   -0.0317    0.0651    0.1843    0.1843    0.4339
    0.4339    0.0335   -0.0912         0   -0.7379   -0.6158   -0.2737   -0.0912   -0.0912   -0.0317   -0.0317   -0.0317    0.1843    0.1843    0.4339
    0.4339    0.0775   -0.0912   -0.7379         0   -1.6525   -0.2737   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0276    0.1843    0.4132
    0.4339    0.0775   -0.0912   -0.6158   -1.6525         0   -1.0344   -1.0344   -1.0344   -0.0912   -0.0912   -0.0912   -0.0912    0.0681    0.3651
    0.4339    0.0775   -0.0912   -0.2737   -0.2737   -1.0344         0   -1.0344   -1.0344   -0.3901   -0.3901   -0.0912   -0.0912   -0.0605    0.2306
    0.4339    0.0775   -0.0912   -0.0912   -0.0912   -1.0344   -1.0344         0   -1.0344   -1.0344   -0.4217   -0.0912   -0.0912   -0.0605    0.2306
    0.4339    0.0775   -0.0912   -0.0912   -0.0912   -1.0344   -1.0344   -1.0344         0   -2.3158   -0.5304   -0.0912   -0.0912   -0.0605    0.2306
    0.4339    0.0775   -0.0317   -0.0317   -0.0912   -0.0912   -0.3901   -1.0344   -2.3158         0   -0.7525   -0.0912   -0.0912   -0.0605    0.1651
    0.4339    0.1729   -0.0317   -0.0317   -0.0912   -0.0912   -0.3901   -0.4217   -0.5304   -0.7525         0   -0.4349   -0.0912   -0.0605    0.0661
    0.4339    0.1729    0.0651   -0.0317   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.4349         0   -0.8301   -0.8301    0.0661
    0.4339    0.4339    0.1843    0.1843   -0.0276   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.0912   -0.8301         0   -4.2276   -0.4252
    0.4339    0.4339    0.1843    0.1843    0.1843    0.0681   -0.0605   -0.0605   -0.0605   -0.0605   -0.0605   -0.8301   -4.2276         0   -0.9349
    0.4339    0.4339    0.4339    0.4339    0.4132    0.3651    0.2306    0.2306    0.2306    0.1651    0.0661    0.0661   -0.4252   -0.9349         0
    0.5766    0.4339    0.4339    0.4339    0.4339    0.4339    0.2828    0.2828    0.2828    0.2828    0.2828    0.2828    0.2828    0.1866   -0.3874
    0.8070    0.8070    0.8070    0.8070    0.8070    0.4339    0.4339    0.4203    0.4203    0.4203    0.3655    0.3655    0.3655    0.3655   -0.1500
    1.1333    1.0021    0.8464    0.8464    0.8070    0.4475    0.4339    0.4203    0.4203    0.4203    0.4203    0.4203    0.4203    0.3655    0.3655

  Columns 16 through 18

    0.5766    0.8070    1.1333
    0.4339    0.8070    1.0021
    0.4339    0.8070    0.8464
    0.4339    0.8070    0.8464
    0.4339    0.8070    0.8070
    0.4339    0.4339    0.4475
    0.2828    0.4339    0.4339
    0.2828    0.4203    0.4203
    0.2828    0.4203    0.4203
    0.2828    0.4203    0.4203
    0.2828    0.3655    0.4203
    0.2828    0.3655    0.4203
    0.2828    0.3655    0.4203
    0.1866    0.3655    0.3655
   -0.3874   -0.1500    0.3655
```

```
         0   -1.6874    0.1833
   -1.6874         0    0.1126
    0.1833    0.1126         0


best_outpermone =

    10    12     8     6    13    17     1    16     4     3     9    15     5     2    11     7    18    14


best_outpermtwo =

    15    13    10     1     5     7     4    14    17     2    18     8    16    11    12     9     6     3

Elapsed time is 444.395907 seconds.
```

2.2 (Multistructural) Ultrametric Structures

Although the bistructural AR representation may generally be considered some type of upper constraint on how well one might do with the use of any two (additively combined) structures, the more restrictive bistructural ultrametric identified below (with a best VAF of .8472 found in a 1000 random starts but not for the lesser set of 100), is not too far behind. In fact, among all of the remaining representations, it is probably the analysis of choice. There is not much more VAF gained in moving to the more general additive trees with the concomitant increase in number of fitted weights, and none of the scaling routines (metric and nonmetric in the city-block and Euclidean implementations) do nearly as well in terms of VAF. In other words, the risk data of Table 1 appear to be better represented through discrete (or categorical) structures than by any type of more continuous representational device.

As noted in the scripts used in fitting the bistructural ultrametric representation, the M-file, `ultraorder.m`, is relied on for interpretive purposes to reorder the various ultrametric components to *nonuniquely* display AR forms. Although some of a dominant/secondary distinction may still be present in our interpretations, it generally seems lessened as compared to using AR forms per se. The range of local optima found is again fairly restricted in VAF (for the 100 random starts, a range from .8297 to .8408), and also in terms of the actual structures identified. From our informal explorations, there are only small variations in the branching of the ultrametric and when certain subsets might emerge; but really major differences in structural interpretations do not appear to be present in any of the local optima identified.

We present the (nonunique) orderings identified by `ultraorder.m` for the

two ultrametric components below. Also, we list the seventeen new subsets defined by each ultrametric along with their diameters (i.e., the maximum values within the subsets), which represent when they were formed in the hierarchy. For the second structures and because of the residualization fitting process, some of these diameters (and fitted values, for that matter) will be negative. Because of an invariance (or insensitivity) to fitting linear transformations of the proximities instead of the original values, an arbitrary additive constant may be added, without loss of any generality, to the fitted second ultrametric structure to make the diameter values, for example, positive (and to make one feel better about the analysis).

Primary:
1: accidental falls
4: fire
3: electrocution
15: tornados
5: flood
9: lightning
17: traffic accidents
2: airplane accidents
16: toxic chemical spill
11: nuclear accident
18: war
14: terrorism
7: homicide
6: heart disease
12: stomach cancer
8: leukaemia
10: lung cancer
13: stroke

{5,15}: (2.02) – storms with water
{18,14}: (2.28) – violent acts (by groups)
{16,11}: (2.42) – discharge of dangerous material
{15,5,9}: (2.91) – storm related

{12,8}: (3.42) – cancers
{10,13}: (3.72) – diseases
{6,12}: (3.92) – diseases
{18,14,7}: (4.10) – violent acts
{17,2}: (4.42) – vehicle accidents
{4,3}: (5.36) – hot hazards
{6,12,8,10,13}: (5.87) – diseases generally
{4,3,15,5,9}: (6.16) – hazards generally
{17,2,16,11}: (6.22) – equipment related
{1,4,3,15,5,9}: (6.36) –
{17,2,16,11,18,14,7}: (6.43) –
{1,4,3,5,9,17,2,16,11,18,14,7}: (6.65) – nondisease risks
{all}: (7.40) –

Secondary:
12: stomach cancer
10: lung cancer
15: tornados
16: toxic chemical spill
9: lightning
3: electrocution
4: fire
13: stroke
6: heart disease
1: accidental falls
17: traffic accidents
18: war
11: nuclear accident
7: homicide
8: leukaemia
5: flood
14: terrorism
2: airplane accidents

{13,6}: (-2.87) – coronary related

{9,3}: (-2.86) – electrical
{11,7}: (-2.43) – resulting in deaths
{9,13,4}: (-2.36) – electrical storm related
{1,17}: (-1.85) – labeled accidental
{8,11,7}: (-1.61) – also, resulting in deaths
{12,10}: (-1.57) – cancers
{13,6,1,17}: (-.93) –
{8,5}: (-.90) –
{14,2}: (-.73) –
{15,16,9,3,4}: (-.40) –
{18,11,7,8,5}: (-.18) –
{15,16,9,3,4,13,6,1,17}: (-.13) –
{18,11,7,8,5,14,2}: (.21) –
{15,16,9,3,4,13,6,1,17,18,11,7,8,5,14,2}: (.29) –
{all}: (.38) –

2.2.1 Basic Script Results for biultrafnd.m

```
>> script_biultrafnd

sorted_vafs =

  Columns 1 through 15

    0.8297    0.8308    0.8317    0.8317    0.8318    0.8324    0.8326    0.8326    0.8326    0.8326    0.8326    0.8326    0.8326    0.8326    0.8326

  Columns 16 through 30

    0.8326    0.8326    0.8326    0.8326    0.8327    0.8328    0.8329    0.8329    0.8329    0.8329    0.8329    0.8329    0.8329    0.8337    0.8338

  Columns 31 through 45

    0.8338    0.8345    0.8346    0.8346    0.8346    0.8346    0.8346    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347

  Columns 46 through 60

    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347

  Columns 61 through 75

    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347    0.8347

  Columns 76 through 90

    0.8347    0.8347    0.8347    0.8347    0.8357    0.8371    0.8373    0.8373    0.8373    0.8374    0.8374    0.8374    0.8374    0.8374    0.8374

  Columns 91 through 100

    0.8374    0.8374    0.8374    0.8375    0.8376    0.8376    0.8376    0.8377    0.8377    0.8408

best_vaf =

    0.8408

best_find =

  Columns 1 through 15

         0    6.9518    6.2773    6.2773    6.8820    6.9913    6.9518    7.7909    6.2773    7.7331    6.8820    7.7331    6.9913    6.9518    6.0824
```

```
    6.9518         0    6.9518    6.9518    6.7923    7.7909    6.4037    7.6314    6.9518    7.7909    6.7923    7.7909    7.7909    5.7000    6.9518
    6.2773    6.9518         0    2.9970    6.4357    7.1862    6.9518    7.7909    3.3000    7.7331    6.3790    7.7331    7.1862    6.9518    5.8310
    6.2773    6.9518    2.9970         0    6.4357    7.1862    6.9518    7.7909    3.8030    7.7331    6.3790    7.7331    7.1862    6.9518    5.8310
    6.8820    6.7923    6.4357    6.4357         0    7.7909    6.5165    6.5000    3.1527    7.7909    6.0004    7.7909    7.7909    6.7923    2.3002
    6.9913    7.7909    7.1862    7.1862    7.7909         0    7.7909    4.2790    7.1862    5.7994    7.7909    4.2212    4.5832    7.7909    6.5168
    6.9518    6.4037    6.9518    6.9518    6.5165    7.7909         0    7.3556    6.9518    7.7909    4.0000    7.7909    7.7909    4.3618    6.9518
    7.7909    7.6314    7.7909    7.7909    6.5000    4.2790    7.3556         0    7.7909    5.8572    7.3556    3.8002    5.8572    7.6314    7.7909
    6.2773    6.9518    3.3000    3.8030    3.1527    7.1862    6.9518    7.7909         0    7.7331    6.4357    7.7331    7.1862    6.9518    2.5480
    7.7331    7.7909    7.7331    7.7331    7.7909    5.7994    7.7909    5.8572    7.7331         0    7.7909    4.3000    4.1001    7.7909    7.7331
    6.8820    6.7923    6.3790    6.3790    6.0004    7.7909    4.0000    7.3556    6.4357    7.7909         0    7.7909    7.7909    6.7923    6.4357
    7.7331    7.7909    7.7331    7.7331    7.7909    4.2212    7.7909    3.8002    7.7331    4.3000    7.7909         0    5.7994    7.7909    7.7331
    6.9913    7.7909    7.1862    7.1862    7.7909    4.5832    7.7909    5.8572    7.1862    4.1001    7.7909    5.7994         0    7.7909    4.9000
    6.9518    5.7000    6.9518    6.9518    6.7923    7.7909    4.3618    7.6314    6.9518    7.7909    6.7923    7.7909    7.7909         0    6.9518
    6.0824    6.9518    5.8310    5.8310    2.3002    6.5168    6.9518    7.7909    2.5480    7.7331    6.4357    7.7331    4.9000    6.9518         0
    6.8242    6.9518    6.3212    6.3212    6.4357    7.7331    6.9518    7.7909    6.3779    7.0000    2.7002    7.0000    7.7331    6.9518    6.3779
    4.8000    4.7002    6.3471    6.3471    6.9518    6.9913    6.5631    7.7909    6.3471    7.7331    6.9518    7.7331    6.9913    6.5631    6.1522
    6.9518    6.4037    6.9518    6.9518    6.5165    7.7909    2.4348    7.3556    6.9518    7.7909    4.8652    7.7909    7.7909    2.4990    6.9518

  Columns 16 through 18

    6.8242    4.8000    6.9518
    6.9518    4.7002    6.4037
    6.3212    6.3471    6.9518
    6.3212    6.3471    6.9518
    6.4357    6.9518    6.5165
    7.7331    6.9913    7.7909
    6.9518    6.5631    2.4348
    7.7909    7.7909    7.3556
    6.3779    6.3471    6.9518
    7.0000    7.7331    7.7909
    2.7002    6.9518    4.8652
    7.0000    7.7331    7.7909
    7.7331    6.9913    7.7909
    6.9518    6.5631    2.4990
    6.3779    6.1522    6.9518
         0    6.8940    6.9518
    6.8940         0    6.5631
    6.9518    6.5631         0


best_targone =

  Columns 1 through 15

         0    6.6389    6.5691    6.5691    6.5691    7.4780    6.6389    7.4780    6.5691    7.4780    6.5691    7.4780    7.4780    6.6389    6.5691
    6.6389         0    6.6389    6.6389    6.6389    7.4780    6.2503    7.4780    6.6389    7.4780    6.6389    7.4780    7.4780    6.2503    6.6389
    6.5691    6.6389         0    5.3168    6.1228    7.4780    6.6389    7.4780    6.1228    7.4780    6.0661    7.4780    7.4780    6.6389    6.1228
    6.5691    6.6389    5.3168         0    6.1228    7.4780    6.6389    7.4780    6.1228    7.4780    6.0661    7.4780    7.4780    6.6389    6.1228
    6.5691    6.6389    6.1228    6.1228         0    7.4780    6.6389    7.4780    2.8398    7.4780    6.1228    7.4780    7.4780    6.6389    1.9873
    7.4780    7.4780    7.4780    7.4780    7.4780         0    7.4780    3.9661    7.4780    5.5443    7.4780    3.9661    5.5443    7.4780    7.4780
    6.6389    6.2503    6.6389    6.6389    6.6389    7.4780         0    7.4780    6.6389    7.4780    6.6389    7.4780    7.4780    4.2084    6.6389
    7.4780    7.4780    7.4780    7.4780    7.4780    3.9661    7.4780         0    7.4780    5.5443    7.4780    3.4873    5.5443    7.4780    7.4780
    6.5691    6.6389    6.1228    6.1228    2.8398    7.4780    6.6389    7.4780         0    7.4780    6.1228    7.4780    7.4780    6.6389    2.8398
    7.4780    7.4780    7.4780    7.4780    7.4780    5.5443    7.4780    5.5443    7.4780         0    7.4780    5.5443    3.8450    7.4780    7.4780
    6.5691    6.6389    6.0661    6.0661    6.1228    7.4780    6.6389    7.4780    6.1228    7.4780         0    7.4780    7.4780    6.6389    6.1228
    7.4780    7.4780    7.4780    7.4780    7.4780    3.9661    7.4780    3.4873    7.4780    5.5443    7.4780         0    5.5443    7.4780    7.4780
    7.4780    7.4780    7.4780    7.4780    7.4780    5.5443    7.4780    5.5443    7.4780    3.8450    7.4780    5.5443         0    7.4780    7.4780
    6.6389    6.2503    6.6389    6.6389    6.6389    7.4780    4.2084    7.4780    6.6389    7.4780    6.6389    7.4780    7.4780         0    6.6389
    6.5691    6.6389    6.1228    6.1228    1.9873    7.4780    6.6389    7.4780    2.8398    7.4780    6.1228    7.4780    7.4780    6.6389         0
    6.5691    6.6389    6.0661    6.0661    6.1228    7.4780    6.6389    7.4780    6.1228    7.4780    2.3873    7.4780    7.4780    6.6389    6.1228
    6.6389    4.3873    6.6389    6.6389    6.6389    7.4780    6.2503    7.4780    6.6389    7.4780    6.6389    7.4780    7.4780    6.2503    6.6389
    6.6389    6.2503    6.6389    6.6389    6.6389    7.4780    4.2084    7.4780    6.6389    7.4780    6.6389    7.4780    7.4780    2.3456    6.6389

  Columns 16 through 18

    6.5691    6.6389    6.6389
    6.6389    4.3873    6.2503
    6.0661    6.6389    6.6389
    6.0661    6.6389    6.6389
    6.1228    6.6389    6.6389
    7.4780    7.4780    7.4780
    6.6389    6.2503    4.2084
    7.4780    7.4780    7.4780
    6.1228    6.6389    6.6389
    7.4780    7.4780    7.4780
    2.3873    6.6389    6.6389
    7.4780    7.4780    7.4780
    7.4780    7.4780    7.4780
    6.6389    6.2503    2.3456
    6.1228    6.6389    6.6389
         0    6.6389    6.6389
    6.6389         0    6.2503
    6.6389    6.2503         0


best_targtwo =
```

```
  Columns 1 through 15

         0    0.3129   -0.2918   -0.2918    0.3129   -0.4867    0.3129    0.3129   -0.2918    0.2551    0.3129    0.2551   -0.4867    0.3129   -0.4867
    0.3129         0    0.3129    0.3129    0.1534    0.3129    0.1534    0.1534    0.3129    0.3129    0.1534    0.3129    0.3129   -0.5503    0.3129
   -0.2918    0.3129         0   -2.3198    0.3129   -0.2918    0.3129    0.3129   -2.8228    0.2551    0.3129    0.2551   -0.2918    0.3129   -0.2918
   -0.2918    0.3129   -2.3198         0    0.3129   -0.2918    0.3129    0.3129   -2.3198    0.2551    0.3129    0.2551   -0.2918    0.3129   -0.2918
    0.3129    0.1534    0.3129    0.3129         0    0.3129   -0.1224   -0.9780    0.3129    0.3129   -0.1224    0.3129    0.3129    0.1534    0.3129
   -0.4867    0.3129   -0.2918   -0.2918    0.3129         0    0.3129    0.3129   -0.2918    0.2551    0.3129    0.2551   -0.9612    0.3129   -0.9612
    0.3129    0.1534    0.3129    0.3129   -0.1224    0.3129         0   -0.1224    0.3129    0.3129   -2.6389    0.3129    0.3129    0.1534    0.3129
    0.3129    0.1534    0.3129    0.3129   -0.9780    0.3129   -0.1224         0    0.3129    0.3129   -0.1224    0.3129    0.3129    0.1534    0.3129
   -0.2918    0.3129   -2.8228   -2.3198    0.3129   -0.2918    0.3129    0.3129         0    0.2551    0.3129    0.2551   -0.2918    0.3129   -0.2918
    0.2551    0.3129    0.2551    0.2551    0.3129    0.2551    0.3129    0.3129    0.2551         0    0.3129   -1.2443    0.2551    0.3129    0.2551
    0.3129    0.1534    0.3129    0.3129   -0.1224    0.3129   -2.6389   -0.1224    0.3129    0.3129         0    0.3129    0.3129    0.1534    0.3129
    0.2551    0.3129    0.2551    0.2551    0.3129    0.2551    0.3129    0.3129    0.2551   -1.2443    0.3129         0    0.2551    0.3129    0.2551
   -0.4867    0.3129   -0.2918   -0.2918    0.3129   -0.9612    0.3129    0.3129   -0.2918    0.2551    0.3129    0.2551         0    0.3129   -2.5780
    0.3129   -0.5503    0.3129    0.3129    0.1534    0.3129    0.1534    0.1534    0.3129    0.3129    0.1534    0.3129    0.3129         0    0.3129
   -0.4867    0.3129   -0.2918   -0.2918    0.3129   -0.9612    0.3129    0.3129   -0.2918    0.2551    0.3129    0.2551   -2.5780    0.3129         0
    0.2551    0.3129    0.2551    0.2551    0.3129    0.2551    0.3129    0.3129    0.2551   -0.4780    0.3129   -0.4780    0.2551    0.3129    0.2551
   -1.8389    0.3129   -0.2918   -0.2918    0.3129   -0.4867    0.3129    0.3129   -0.2918    0.2551    0.3129    0.2551   -0.4867    0.3129   -0.4867
    0.3129    0.1534    0.3129    0.3129   -0.1224    0.3129   -1.7737   -0.1224    0.3129    0.3129   -1.7737    0.3129    0.3129    0.1534    0.3129

  Columns 16 through 18

    0.2551   -1.8389    0.3129
    0.3129    0.3129    0.1534
    0.2551   -0.2918    0.3129
    0.2551   -0.2918    0.3129
    0.3129    0.3129   -0.1224
    0.2551   -0.4867    0.3129
    0.3129    0.3129   -1.7737
    0.3129    0.3129   -0.1224
    0.2551   -0.2918    0.3129
   -0.4780    0.2551    0.3129
    0.3129    0.3129   -1.7737
   -0.4780    0.2551    0.3129
    0.2551   -0.4867    0.3129
    0.3129    0.3129    0.1534
    0.2551   -0.4867    0.3129
         0    0.2551    0.3129
    0.2551         0    0.3129
    0.3129    0.3129         0


orderproxone =

  Columns 1 through 15

         0    3.8450    5.5443    5.5443    5.5443    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780
    3.8450         0    5.5443    5.5443    5.5443    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780
    5.5443    5.5443         0    3.9661    3.9661    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780
    5.5443    5.5443    3.9661         0    3.4873    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780
    5.5443    5.5443    3.9661    3.4873         0    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780    7.4780
    7.4780    7.4780    7.4780    7.4780    7.4780         0    2.3456    4.2084    6.2503    6.2503    6.6389    6.6389    6.6389    6.6389    6.6389
    7.4780    7.4780    7.4780    7.4780    7.4780    2.3456         0    4.2084    6.2503    6.2503    6.6389    6.6389    6.6389    6.6389    6.6389
    7.4780    7.4780    7.4780    7.4780    7.4780    4.2084    4.2084         0    6.2503    6.2503    6.6389    6.6389    6.6389    6.6389    6.6389
    7.4780    7.4780    7.4780    7.4780    7.4780    6.2503    6.2503    6.2503         0    4.3873    6.6389    6.6389    6.6389    6.6389    6.6389
    7.4780    7.4780    7.4780    7.4780    7.4780    6.2503    6.2503    6.2503    4.3873         0    6.6389    6.6389    6.6389    6.6389    6.6389
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389         0    6.5691    6.5691    6.5691    6.5691
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691         0    2.8398    2.8398    6.1228
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691    2.8398         0    1.9873    6.1228
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691    2.8398    1.9873         0    6.1228
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691    6.1228    6.1228    6.1228         0
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691    6.1228    6.1228    6.1228    2.3873
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691    6.1228    6.1228    6.1228    6.0661
    7.4780    7.4780    7.4780    7.4780    7.4780    6.6389    6.6389    6.6389    6.6389    6.6389    6.5691    6.1228    6.1228    6.1228    6.0661

  Columns 16 through 18

    7.4780    7.4780    7.4780
    7.4780    7.4780    7.4780
    7.4780    7.4780    7.4780
    7.4780    7.4780    7.4780
    7.4780    7.4780    7.4780
    6.6389    6.6389    6.6389
    6.6389    6.6389    6.6389
    6.6389    6.6389    6.6389
    6.6389    6.6389    6.6389
    6.6389    6.6389    6.6389
    6.5691    6.5691    6.5691
    6.1228    6.1228    6.1228
    6.1228    6.1228    6.1228
    6.1228    6.1228    6.1228
    2.3873    6.0661    6.0661
         0    6.0661    6.0661
    6.0661         0    5.3168
    6.0661    5.3168         0
```

```
orderpermone =

    13    10     6    12     8    14    18     7    17     2     1     9     5    15    16    11     3     4


orderproxtwo =

  Columns 1 through 15

         0   -0.4780   -0.4780    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.3129    0.3129    0.3129    0.3129
   -0.4780         0   -1.2443    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.3129    0.3129    0.3129    0.3129
   -0.4780   -1.2443         0    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.2551    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551         0   -1.8389   -0.4867   -0.4867   -0.4867   -0.2918   -0.2918   -0.2918    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -1.8389         0   -0.4867   -0.4867   -0.4867   -0.2918   -0.2918   -0.2918    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -0.4867   -0.4867         0   -0.9612   -0.9612   -0.2918   -0.2918   -0.2918    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -0.4867   -0.4867   -0.9612         0   -2.5780   -0.2918   -0.2918   -0.2918    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -0.4867   -0.4867   -0.9612   -2.5780         0   -0.2918   -0.2918   -0.2918    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -0.2918   -0.2918   -0.2918   -0.2918   -0.2918         0   -2.8228   -2.3198    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -0.2918   -0.2918   -0.2918   -0.2918   -0.2918   -2.8228         0   -2.3198    0.3129    0.3129    0.3129    0.3129
    0.2551    0.2551    0.2551   -0.2918   -0.2918   -0.2918   -0.2918   -0.2918   -2.3198   -2.3198         0    0.3129    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129         0   -0.5503    0.1534    0.1534
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129   -0.5503         0    0.1534    0.1534
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.1534    0.1534         0   -2.6389
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.1534    0.1534   -2.6389         0
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.1534    0.1534   -1.7737   -1.7737
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.1534    0.1534   -0.1224   -0.1224
    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.3129    0.1534    0.1534   -0.1224   -0.1224

  Columns 16 through 18

    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.3129    0.3129    0.3129
    0.1534    0.1534    0.1534
    0.1534    0.1534    0.1534
   -1.7737   -0.1224   -0.1224
   -1.7737   -0.1224   -0.1224
         0   -0.1224   -0.1224
   -0.1224         0   -0.9780
   -0.1224   -0.9780         0


orderpermtwo =

    16    12    10    17     1     6    13    15     9     3     4    14     2    11     7    18     5     8

Elapsed time is 1634.839380 seconds.
```

2.2.2 The Enhanced Script for ms_biultrafnd.m

```
load risk_rate.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(1000,1);

for i = 1:1000

    [find,vaf,targone,targtwo] = ...
        ms_biultrafnd(risk_rate,randperm(18));

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_find = find;
        best_targone = targone;
        best_targtwo = targtwo;
```

```
    end
end
sorted_vafs = sort(store_vaf');
sorted_vafs;
best_vaf
best_find
best_targone
best_targtwo
[orderproxone,orderpermone] = ultraorder(best_targone)
[orderproxtwo,orderpermtwo] = ultraorder(best_targtwo)

toc;
```

2.2.3 Enhanced Script Results for ms_biultrafnd.m

```
>> ms_script_biultrafnd

best_vaf =

    0.8472


best_find =

  Columns 1 through 15

         0    6.9375    6.2309    6.2309    6.6474    6.4750    6.9375    7.6853    6.2309    7.7756    6.9375    7.7756    6.4750    6.9375    6.2309
    6.9375         0    6.9375    6.9375    6.8645    7.6853    6.6400    7.6123    6.9375    7.7756    6.4272    7.7756    7.6853    5.7000    6.9375
    6.2309    6.9375         0    2.9997    6.4421    7.2688    6.9375    7.6853    3.3000    7.7756    6.9375    7.7756    7.2688    6.9375    5.7606
    6.2309    6.9375    2.9997         0    6.4421    7.2688    6.9375    7.6853    3.8003    7.7756    6.9375    7.7756    7.2688    6.9375    5.7606
    6.6474    6.8645    6.4421    6.4421         0    7.6853    6.4761    6.5000    3.1925    7.7756    6.4761    7.7756    7.6853    6.8645    2.3010
    6.4750    7.6853    7.2688    7.2688    7.6853         0    7.6853    4.2050    7.2688    6.2473    7.6853    4.2952    3.0000    7.6853    7.2688
    6.9375    6.6400    6.9375    6.9375    6.4761    7.6853         0    7.2239    6.9375    7.7756    4.0000    7.7756    7.6853    4.3074    6.9375
    7.6853    7.6123    7.6853    7.6853    6.5000    4.2050    7.2239         0    7.6853    6.2473    7.2239    3.7992    6.1570    7.6123    7.6853
    6.2309    6.9375    3.3000    3.8003    3.1925    7.2688    6.9375    7.6853         0    7.7756    6.9375    7.7756    7.2688    6.9375    2.5110
    7.7756    7.7756    7.7756    7.7756    7.7756    6.2473    7.7756    6.2473    7.7756         0    7.7756    4.3000    4.0992    7.7756    7.7756
    6.9375    6.4272    6.9375    6.9375    6.4761    7.6853    4.0000    7.2239    6.9375    7.7756         0    7.7756    7.6853    6.6400    6.9375
    7.7756    7.7756    7.7756    7.7756    7.7756    4.2952    7.7756    3.7992    7.7756    4.3000    7.7756         0    6.2473    7.7756    7.7756
    6.4750    7.6853    7.2688    7.2688    7.6853    3.0000    7.6853    6.1570    7.2688    4.0992    7.6853    6.2473         0    7.6853    7.2688
    6.9375    5.7000    6.9375    6.9375    6.8645    7.6853    4.3074    7.6123    6.9375    7.7756    6.6400    7.7756    7.6853         0    6.9375
    6.2309    6.9375    5.7606    5.7606    2.3010    7.2688    6.9375    7.6853    2.5110    7.7756    6.9375    7.7756    7.2688    6.9375         0
    6.5210    6.5002    6.2560    6.2560    6.9375    7.2688    6.7129    7.6853    6.2560    7.7756    2.7010    7.7756    7.2688    6.7129    5.5000
    4.8000    4.7010    6.5210    6.5210    6.9375    6.4750    6.7129    7.6853    6.5210    7.7756    6.5002    7.7756    6.4750    6.7129    6.5210
    6.9375    6.6400    6.9375    6.9375    6.4761    7.6853    2.4837    7.2239    6.9375    7.7756    4.8163    7.7756    7.6853    2.4918    6.9375

  Columns 16 through 18

    6.5210    4.8000    6.9375
    6.5002    4.7010    6.6400
    6.2560    6.5210    6.9375
    6.2560    6.5210    6.9375
    6.9375    6.9375    6.4761
    7.2688    6.4750    7.6853
    6.7129    6.7129    2.4837
    7.6853    7.6853    7.2239
    6.2560    6.5210    6.9375
    7.7756    7.7756    7.7756
    2.7010    6.5002    4.8163
    7.7756    7.7756    7.7756
    7.2688    6.4750    7.6853
    6.7129    6.7129    2.4918
    5.5000    6.5210    6.9375
         0    6.0837    6.7129
    6.0837         0    6.7129
    6.7129    6.7129         0


best_targone =

  Columns 1 through 15

         0    6.6526    6.3625    6.3625    6.3625    7.4004    6.6526    7.4004    6.3625    7.4004    6.6526    7.4004    7.4004    6.6526    6.3625
    6.6526         0    6.6526    6.6526    6.6526    7.4004    6.4280    7.4004    6.6526    7.4004    6.2153    7.4004    7.4004    6.4280    6.6526
```

```
    6.3625    6.6526         0    5.3566    6.1572    7.4004    6.6526    7.4004    6.1572    7.4004    6.6526    7.4004    7.4004    6.6526    6.1572
    6.3625    6.6526    5.3566         0    6.1572    7.4004    6.6526    7.4004    6.1572    7.4004    6.6526    7.4004    7.4004    6.6526    6.1572
    6.3625    6.6526    6.1572    6.1572         0    7.4004    6.6526    7.4004    2.9076    7.4004    6.6526    7.4004    7.4004    6.6526    2.0161
    7.4004    7.4004    7.4004    7.4004    7.4004         0    7.4004    3.9201    7.4004    5.8721    7.4004    3.9201    5.8721    7.4004    7.4004
    6.6526    6.4280    6.6526    6.6526    6.6526    7.4004         0    7.4004    6.6526    7.4004    6.4280    7.4004    7.4004    4.0955    6.6526
    7.4004    7.4004    7.4004    7.4004    7.4004    3.9201    7.4004         0    7.4004    5.8721    7.4004    3.4240    5.8721    7.4004    7.4004
    6.3625    6.6526    6.1572    6.1572    2.9076    7.4004    6.6526    7.4004         0    7.4004    6.6526    7.4004    7.4004    6.6526    2.9076
    7.4004    7.4004    7.4004    7.4004    7.4004    5.8721    7.4004    5.8721    7.4004         0    7.4004    5.8721    3.7240    7.4004    7.4004
    6.6526    6.2153    6.6526    6.6526    6.6526    7.4004    6.4280    7.4004    6.6526    7.4004         0    7.4004    7.4004    6.4280    6.6526
    7.4004    7.4004    7.4004    7.4004    7.4004    3.9201    7.4004    3.4240    7.4004    5.8721    7.4004         0    5.8721    7.4004    7.4004
    7.4004    7.4004    7.4004    7.4004    7.4004    5.8721    7.4004    5.8721    7.4004    3.7240    7.4004    5.8721         0    7.4004    7.4004
    6.6526    6.4280    6.6526    6.6526    6.6526    7.4004    4.0955    7.4004    6.6526    7.4004    6.4280    7.4004    7.4004         0    6.6526
    6.3625    6.6526    6.1572    6.1572    2.0161    7.4004    6.6526    7.4004    2.9076    7.4004    6.6526    7.4004    7.4004    6.6526         0
    6.6526    6.2153    6.6526    6.6526    6.6526    7.4004    6.4280    7.4004    6.6526    7.4004    2.4161    7.4004    7.4004    6.4280    6.6526
    6.6526    4.4161    6.6526    6.6526    6.6526    7.4004    6.4280    7.4004    6.6526    7.4004    6.2153    7.4004    7.4004    6.4280    6.6526
    6.6526    6.4280    6.6526    6.6526    6.6526    7.4004    4.0955    7.4004    6.6526    7.4004    6.4280    7.4004    7.4004    2.2799    6.6526

  Columns 16 through 18

    6.6526    6.6526    6.6526
    6.2153    4.4161    6.4280
    6.6526    6.6526    6.6526
    6.6526    6.6526    6.6526
    6.6526    6.6526    6.6526
    7.4004    7.4004    7.4004
    6.4280    6.4280    4.0955
    7.4004    7.4004    7.4004
    6.6526    6.6526    6.6526
    7.4004    7.4004    7.4004
    2.4161    6.2153    6.4280
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    6.4280    6.4280    2.2799
    6.6526    6.6526    6.6526
         0    6.2153    6.4280
    6.2153         0    6.4280
    6.4280    6.4280         0


best_targtwo =

  Columns 1 through 15

         0    0.2849   -0.1316   -0.1316    0.2849   -0.9254    0.2849    0.2849   -0.1316    0.3752    0.2849    0.3752   -0.9254    0.2849   -0.1316
    0.2849         0    0.2849    0.2849    0.2119    0.2849    0.2119    0.2119    0.2849    0.3752    0.2119    0.3752    0.2849   -0.7280    0.2849
   -0.1316    0.2849         0   -2.3569    0.2849   -0.1316    0.2849    0.2849   -2.8572    0.3752    0.2849    0.3752   -0.1316    0.2849   -0.3966
   -0.1316    0.2849   -2.3569         0    0.2849   -0.1316    0.2849    0.2849   -2.3569    0.3752    0.2849    0.3752   -0.1316    0.2849   -0.3966
    0.2849    0.2119    0.2849    0.2849         0    0.2849   -0.1765   -0.9004    0.2849    0.3752   -0.1765    0.3752    0.2849    0.2119    0.2849
   -0.9254    0.2849   -0.1316   -0.1316    0.2849         0    0.2849    0.2849   -0.1316    0.3752    0.2849    0.3752   -2.8721    0.2849   -0.1316
    0.2849    0.2119    0.2849    0.2849   -0.1765    0.2849         0   -0.1765    0.2849    0.3752   -2.4280    0.3752    0.2849    0.2119    0.2849
    0.2849    0.2119    0.2849    0.2849   -0.9004    0.2849   -0.1765         0    0.2849    0.3752   -0.1765    0.3752    0.2849    0.2119    0.2849
   -0.1316    0.2849   -2.8572   -2.3569    0.2849   -0.1316    0.2849    0.2849         0    0.3752    0.2849    0.3752   -0.1316    0.2849   -0.3966
    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752         0    0.3752   -1.5721    0.3752    0.3752    0.3752
    0.2849    0.2119    0.2849    0.2849   -0.1765    0.2849   -2.4280   -0.1765    0.2849    0.3752         0    0.3752    0.2849    0.2119    0.2849
    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752   -1.5721    0.3752         0    0.3752    0.3752    0.3752
   -0.9254    0.2849   -0.1316   -0.1316    0.2849   -2.8721    0.2849    0.2849   -0.1316    0.3752    0.2849    0.3752         0    0.2849   -0.1316
    0.2849   -0.7280    0.2849    0.2849    0.2119    0.2849    0.2119    0.2119    0.2849    0.3752    0.2119    0.3752    0.2849         0    0.2849
   -0.1316    0.2849   -0.3966   -0.3966    0.2849   -0.1316    0.2849    0.2849   -0.3966    0.3752    0.2849    0.3752   -0.1316    0.2849         0
   -0.1316    0.2849   -0.3966   -0.3966    0.2849   -0.1316    0.2849    0.2849   -0.3966    0.3752    0.2849    0.3752   -0.1316    0.2849   -1.1526
   -1.8526    0.2849   -0.1316   -0.1316    0.2849   -0.9254    0.2849    0.2849   -0.1316    0.3752    0.2849    0.3752   -0.9254    0.2849   -0.1316
    0.2849    0.2119    0.2849    0.2849   -0.1765    0.2849   -1.6118   -0.1765    0.2849    0.3752   -1.6118    0.3752    0.2849    0.2119    0.2849

  Columns 16 through 18

   -0.1316   -1.8526    0.2849
    0.2849    0.2849    0.2119
   -0.3966   -0.1316    0.2849
   -0.3966   -0.1316    0.2849
    0.2849    0.2849   -0.1765
   -0.1316   -0.9254    0.2849
    0.2849    0.2849   -1.6118
    0.2849    0.2849   -0.1765
   -0.3966   -0.1316    0.2849
    0.3752    0.3752    0.3752
    0.2849    0.2849   -1.6118
    0.3752    0.3752    0.3752
   -0.1316   -0.9254    0.2849
    0.2849    0.2849    0.2119
   -1.1526   -0.1316    0.2849
         0   -0.1316    0.2849
   -0.1316         0    0.2849
    0.2849    0.2849         0


orderproxone =

  Columns 1 through 15
```

```
         0    6.3625    6.3625    6.3625    6.3625    6.3625    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    7.4004    7.4004
    6.3625         0    5.3566    6.1572    6.1572    6.1572    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    7.4004    7.4004
    6.3625    5.3566         0    6.1572    6.1572    6.1572    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    7.4004    7.4004
    6.3625    6.1572    6.1572         0    2.0161    2.9076    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    7.4004    7.4004
    6.3625    6.1572    6.1572    2.0161         0    2.9076    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    7.4004    7.4004
    6.3625    6.1572    6.1572    2.9076    2.9076         0    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526         0    4.4161    6.2153    6.2153    6.4280    6.4280    6.4280    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    4.4161         0    6.2153    6.2153    6.4280    6.4280    6.4280    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.2153    6.2153         0    2.4161    6.4280    6.4280    6.4280    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.2153    6.2153    2.4161         0    6.4280    6.4280    6.4280    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.4280    6.4280    6.4280    6.4280         0    2.2799    4.0955    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.4280    6.4280    6.4280    6.4280    2.2799         0    4.0955    7.4004    7.4004
    6.6526    6.6526    6.6526    6.6526    6.6526    6.6526    6.4280    6.4280    6.4280    6.4280    4.0955    4.0955         0    7.4004    7.4004
    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004         0    3.9201
    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    3.9201         0
    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    3.9201    3.4240
    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    5.8721    5.8721
    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    7.4004    5.8721    5.8721

  Columns 16 through 18

    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    7.4004    7.4004    7.4004
    3.9201    5.8721    5.8721
    3.4240    5.8721    5.8721
         0    5.8721    5.8721
    5.8721         0    3.7240
    5.8721    3.7240         0


orderpermone =

     1     4     3    15     5     9    17     2    16    11    18    14     7     6    12     8    10    13


orderproxtwo =

  Columns 1 through 15

         0   -1.5721    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752
   -1.5721         0    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752    0.3752
    0.3752    0.3752         0   -1.1526   -0.3966   -0.3966   -0.3966   -0.1316   -0.1316   -0.1316   -0.1316    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -1.1526         0   -0.3966   -0.3966   -0.3966   -0.1316   -0.1316   -0.1316   -0.1316    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.3966   -0.3966         0   -2.8572   -2.3569   -0.1316   -0.1316   -0.1316   -0.1316    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.3966   -0.3966   -2.8572         0   -2.3569   -0.1316   -0.1316   -0.1316   -0.1316    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.3966   -0.3966   -2.3569   -2.3569         0   -0.1316   -0.1316   -0.1316   -0.1316    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.1316   -0.1316   -0.1316   -0.1316   -0.1316         0   -2.8721   -0.9254   -0.9254    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.1316   -0.1316   -0.1316   -0.1316   -0.1316   -2.8721         0   -0.9254   -0.9254    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.1316   -0.1316   -0.1316   -0.1316   -0.1316   -0.9254   -0.9254         0   -1.8526    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752   -0.1316   -0.1316   -0.1316   -0.1316   -0.1316   -0.9254   -0.9254   -1.8526         0    0.2849    0.2849    0.2849    0.2849
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849         0   -1.6118   -1.6118   -0.1765
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849   -1.6118         0   -2.4280   -0.1765
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849   -1.6118   -2.4280         0   -0.1765
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849   -0.1765   -0.1765   -0.1765         0
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849   -0.1765   -0.1765   -0.1765   -0.9004
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2119    0.2119    0.2119    0.2119
    0.3752    0.3752    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2849    0.2119    0.2119    0.2119    0.2119

  Columns 16 through 18

    0.3752    0.3752    0.3752
    0.3752    0.3752    0.3752
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
    0.2849    0.2849    0.2849
   -0.1765    0.2119    0.2119
   -0.1765    0.2119    0.2119
   -0.1765    0.2119    0.2119
         0    0.2119    0.2119
```

```
    0.2119         0   -0.7280
    0.2119   -0.7280         0

orderpermtwo =

    12    10    15    16     9     3     4    13     6     1    17    18    11     7     8     5    14     2

Elapsed time is 2250.681747 seconds.
```

2.3 (Multistructural) Additive Trees

We will not give a detailed interpretation for the additive tree results that follow. They would generally mirror those proposed for `biultrafnd.m`, with the additional decompositions of each additive tree into an ultrametric and centroid (metric) component plus a subsequent reordering of the ultrametric with `ultraorder.m`. There is little to be gained in best VAFs observed for the increased (about double) number of weights that must be used over that for a biultrametric representation (the best VAFs change about 3% from .8472 (biultrametric) to .8768 (biadditive tree); there is also the matter of a three-fold increase in computation times.

2.3.1 Basic Script Results for biatreefnd.m

```
>> script_biatreefnd

sorted_vafs =

  Columns 1 through 15

    0.8577    0.8580    0.8580    0.8580    0.8580    0.8592    0.8593    0.8597    0.8598    0.8598    0.8598    0.8599    0.8599    0.8602    0.8602

  Columns 16 through 30

    0.8602    0.8603    0.8603    0.8603    0.8603    0.8603    0.8603    0.8603    0.8603    0.8603    0.8603    0.8610    0.8610    0.8610    0.8610

  Columns 31 through 45

    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610    0.8610

  Columns 46 through 60

    0.8624    0.8626    0.8629    0.8637    0.8637    0.8659    0.8662    0.8662    0.8681    0.8685    0.8689    0.8689    0.8690    0.8690    0.8690

  Columns 61 through 75

    0.8691    0.8691    0.8691    0.8691    0.8691    0.8691    0.8691    0.8691    0.8704    0.8707    0.8707    0.8707    0.8707    0.8708    0.8715

  Columns 76 through 90

    0.8721    0.8721    0.8722    0.8722    0.8722    0.8722    0.8722    0.8722    0.8722    0.8729    0.8729    0.8733    0.8734    0.8734    0.8734

  Columns 91 through 100

    0.8734    0.8735    0.8735    0.8735    0.8736    0.8741    0.8741    0.8741    0.8767    0.8768

best_vaf =

    0.8768

best_find =

  Columns 1 through 15
```

```
         0    7.1981    6.4445    5.9240    7.2328    7.0879    6.9879    7.3750    6.1315    7.5163    7.0599    7.6330    6.3899    7.1743    6.6711
    7.1981         0    6.8226    6.3022    6.0488    8.2088    6.7589    7.7492    5.9176    7.8674    6.5075    7.9841    7.5108    6.5208    6.2337
    6.4445    6.8226         0    2.9934    6.8573    7.6786    6.9754    7.9657    3.3000    8.1070    6.7240    8.2237    6.9806    7.1618    6.2956
    5.9240    6.3022    2.9934         0    6.3368    7.1582    6.4550    7.4452    3.8066    7.5865    6.2036    7.7033    6.4602    6.6414    5.7751
    7.2328    6.0488    6.8573    6.3368         0    8.2434    6.3927    6.5000    3.1313    7.9251    6.1413    8.0418    7.5454    6.9800    2.2998
    7.0879    8.2088    7.6786    7.1582    8.2434         0    7.9985    4.2006    7.3656    6.6947    8.0706    4.0999    3.0000    8.1849    6.8490
    6.9879    6.7589    6.9754    6.4550    6.3927    7.9985         0    7.1381    6.6624    7.6802    4.5547    7.7969    7.3005    4.3542    6.9785
    7.3750    7.7492    7.9657    7.4452    6.5000    4.2006    7.1381         0    7.6527    6.2351    7.2101    3.9991    5.8555    7.7253    7.9688
    6.1315    5.9176    3.3000    3.8066    3.1313    7.3656    6.6624    7.6527         0    7.7940    6.4110    7.9107    6.6676    6.8488    2.5696
    7.5163    7.8674    8.1070    7.5865    7.9251    6.6947    7.6802    6.2351    7.7940         0    7.7523    4.3000    4.0999    7.8435    8.1101
    7.0599    6.5075    6.7240    6.2036    6.1413    8.0706    4.5547    7.2101    6.4110    7.7523         0    7.8690    7.3726    6.8071    6.7271
    7.6330    7.9841    8.2237    7.7033    8.0418    4.0999    7.7969    3.9991    7.9107    4.3000    7.8690         0    6.1135    7.9602    8.2268
    6.3899    7.5108    6.9806    6.4602    7.5454    3.0000    7.3005    5.8555    6.6676    4.0999    7.3726    6.1135         0    7.4869    6.1510
    7.1743    6.5208    7.1618    6.6414    6.9800    8.1849    4.3542    7.7253    6.8488    7.8435    6.8071    7.9602    7.4869         0    7.1649
    6.6711    6.2337    6.2956    5.7751    2.2998    6.8490    6.9785    7.9688    2.5696    8.1101    6.7271    8.2268    6.1510    7.1649         0
    6.8427    6.2671    6.5068    5.9863    6.3249    7.8533    6.4035    7.3937    6.1937    6.9416    2.6999    7.0584    7.1554    6.5668    6.5098
    4.7998    4.7000    6.9441    6.4236    6.7622    7.3640    6.5173    6.9044    6.6311    7.0226    6.5894    7.1393    6.6660    6.2792    6.9472
    7.3314    7.1024    7.3189    6.7985    6.7362    8.3420    2.4453    7.4816    7.0059    8.0237    4.3000    8.1404    7.6440    2.4999    7.3220

  Columns 16 through 18

    6.8427    4.7998    7.3314
    6.2671    4.7000    7.1024
    6.5068    6.9441    7.3189
    5.9863    6.4236    6.7985
    6.3249    6.7622    6.7362
    7.8533    7.3640    8.3420
    6.4035    6.5173    2.4453
    7.3937    6.9044    7.4816
    6.1937    6.6311    7.0059
    6.9416    7.0226    8.0237
    2.6999    6.5894    4.3000
    7.0584    7.1393    8.1404
    7.1554    6.6660    7.6440
    6.5668    6.2792    2.4999
    6.5098    6.9472    7.3220
         0    6.3490    6.7470
    6.3490         0    6.8608
    6.7470    6.8608         0


best_targone =

  Columns 1 through 15

         0    7.0386    6.8480    6.3918    7.0755    7.3401    6.6776    7.2176    6.5350    7.3358    6.7123    7.4526    6.6421    7.1150    7.0317
    7.0386         0    6.4203    5.9641    6.0558    7.8786    6.6130    7.7562    5.5153    7.8744    6.3241    7.9911    7.1806    7.0503    6.0120
    6.8480    6.4203         0    4.6472    6.4572    7.6880    6.4223    7.5656    5.9167    7.6838    6.1335    7.8005    6.9900    6.8597    6.4134
    6.3918    5.9641    4.6472         0    6.0009    7.2317    5.9661    7.1093    5.4604    7.2275    5.6773    7.3442    6.5338    6.4035    5.9572
    7.0755    6.0558    6.4572    6.0009         0    7.9154    6.6498    7.7930    2.7312    7.9112    6.3610    8.0279    7.2175    7.0872    2.0803
    7.3401    7.8786    7.6880    7.2317    7.9154         0    7.5176    3.8727    7.3750    6.3437    7.5522    3.7489    5.6499    7.9549    7.8717
    6.6776    6.6130    6.4223    5.9661    6.6498    7.5176         0    7.3951    6.1093    7.5133    6.2866    7.6301    6.8196    4.3084    6.6060
    7.2176    7.7562    7.5656    7.1093    7.7930    3.8727    7.3951         0    7.2525    6.2212    7.4298    3.9852    5.5275    7.8325    7.7493
    6.5350    5.5153    5.9167    5.4604    2.7312    7.3750    6.1093    7.2525         0    7.3707    5.8205    7.4875    6.6770    6.5467    2.6874
    7.3358    7.8744    7.6838    7.2275    7.9112    6.3437    7.5133    6.2212    7.3707         0    7.5480    6.4562    3.7489    7.9507    7.8675
    6.7123    6.3241    6.1335    5.6773    6.3610    7.5522    6.2866    7.4298    5.8205    7.5480         0    7.6647    6.8543    6.7240    6.3172
    7.4526    7.9911    7.8005    7.3442    8.0279    3.7489    7.6301    3.9852    7.4875    6.4562    7.6647         0    5.7624    8.0674    7.9842
    6.6421    7.1806    6.9900    6.5338    7.2175    6.4499    6.8196    5.5275    6.6770    3.7489    6.8543    5.7624         0    7.2570    7.1737
    7.1150    7.0503    6.8597    6.4035    7.0872    7.9549    4.3084    7.8325    6.5467    7.9507    6.7240    8.0674    7.2570         0    7.0434
    7.0317    6.0120    6.4134    5.9572    2.0803    7.8717    6.6060    7.7493    2.6874    7.8675    6.3172    7.9842    7.1737    7.0434         0
    6.7623    6.3741    6.1835    5.7273    6.4110    7.6022    6.3366    7.4798    5.8705    7.5980    2.5956    7.7147    6.9043    6.7740    6.3672
    4.6403    6.7324    6.5418    6.0855    6.7692    7.0338    6.3714    6.9114    6.2288    7.0296    6.4060    7.1463    6.3358    6.8087    6.7255
    6.9837    6.9191    6.7285    6.2722    6.9559    7.8237    4.1772    7.7013    6.4154    7.8195    6.5927    7.9362    7.1257    2.4167    6.9122

  Columns 16 through 18

    6.7623    4.6403    6.9837
    6.3741    6.7324    6.9191
    6.1835    6.5418    6.7285
    5.7273    6.0855    6.2722
    6.4110    6.7692    6.9559
    7.6022    7.0338    7.8237
    6.3366    6.3714    4.1772
    7.4798    6.9114    7.7013
    5.8705    6.2288    6.4154
    7.5980    7.0296    7.8195
    2.5956    6.4060    6.5927
    7.7147    7.1463    7.9362
    6.9043    6.3358    7.1257
    6.7740    6.8087    2.4167
    6.3672    6.7255    6.9122
         0    6.4560    6.6427
    6.4560         0    6.6775
    6.6427    6.6775         0
```

```
best_targtwo =

  Columns 1 through 15

         0    0.1595   -0.4035   -0.4677    0.1573   -0.2522    0.3103    0.1573   -0.4035    0.1804    0.3477    0.1804   -0.2522    0.0593   -0.3607
    0.1595         0    0.4023    0.3381   -0.0070    0.3302    0.1460   -0.0070    0.4023   -0.0070    0.1833   -0.0070    0.3302   -0.5295    0.2217
   -0.4035    0.4023         0   -1.6538    0.4001   -0.0094    0.5531    0.4001   -2.6167    0.4232    0.5905    0.4232   -0.0094    0.3021   -0.1178
   -0.4677    0.3381   -1.6538         0    0.3359   -0.0736    0.4889    0.3359   -1.6538    0.3590    0.5263    0.3590   -0.0736    0.2379   -0.1820
    0.1573   -0.0070    0.4001    0.3359         0    0.3280   -0.2571   -1.2930    0.4001    0.0139   -0.2197    0.0139    0.3280   -0.1072    0.2195
   -0.2522    0.3302   -0.0094   -0.0736    0.3280         0    0.4809    0.3280   -0.0094    0.3511    0.5183    0.3511   -2.6499    0.2300   -1.0227
    0.3103    0.1460    0.5531    0.4889   -0.2571    0.4809         0   -0.2571    0.5531    0.1669   -1.7319    0.1669    0.4809    0.0458    0.3725
    0.1573   -0.0070    0.4001    0.3359   -1.2930    0.3280   -0.2571         0    0.4001    0.0139   -0.2197    0.0139    0.3280   -0.1072    0.2195
   -0.4035    0.4023   -2.6167   -1.6538    0.4001   -0.0094    0.5531    0.4001         0    0.4232    0.5905    0.4232   -0.0094    0.3021   -0.1178
    0.1804   -0.0070    0.4232    0.3590    0.0139    0.3511    0.1669    0.0139    0.4232         0    0.2042   -2.1562    0.3511   -0.1072    0.2426
    0.3477    0.1833    0.5905    0.5263   -0.2197    0.5183   -1.7319   -0.2197    0.5905    0.2042         0    0.2042    0.5183    0.0832    0.4098
    0.1804   -0.0070    0.4232    0.3590    0.0139    0.3511    0.1669    0.0139    0.4232   -2.1562    0.2042         0    0.3511   -0.1072    0.2426
   -0.2522    0.3302   -0.0094   -0.0736    0.3280   -2.6499    0.4809    0.3280   -0.0094    0.3511    0.5183    0.3511         0    0.2300   -1.0227
    0.0593   -0.5295    0.3021    0.2379   -0.1072    0.2300    0.0458   -0.1072    0.3021   -0.1072    0.0832   -0.1072    0.2300         0    0.1215
   -0.3607    0.2217   -0.1178   -0.1820    0.2195   -1.0227    0.3725    0.2195   -0.1178    0.2426    0.4098    0.2426   -1.0227    0.1215         0
    0.0804   -0.1070    0.3232    0.2590   -0.0861    0.2511    0.0669   -0.0861    0.3232   -0.6564    0.1043   -0.6564    0.2511   -0.2072    0.1426
    0.1595   -2.0324    0.4023    0.3381   -0.0070    0.3302    0.1460   -0.0070    0.4023   -0.0070    0.1833   -0.0070    0.3302   -0.5295    0.2217
    0.3477    0.1833    0.5905    0.5263   -0.2197    0.5183   -1.7319   -0.2197    0.5905    0.2042   -2.2927    0.2042    0.5183    0.0832    0.4098

  Columns 16 through 18

    0.0804    0.1595    0.3477
   -0.1070   -2.0324    0.1833
    0.3232    0.4023    0.5905
    0.2590    0.3381    0.5263
   -0.0861   -0.0070   -0.2197
    0.2511    0.3302    0.5183
    0.0669    0.1460   -1.7319
   -0.0861   -0.0070   -0.2197
    0.3232    0.4023    0.5905
   -0.6564   -0.0070    0.2042
    0.1043    0.1833   -2.2927
   -0.6564   -0.0070    0.2042
    0.2511    0.3302    0.5183
   -0.2072   -0.5295    0.0832
    0.1426    0.2217    0.4098
         0   -0.1070    0.1043
   -0.1070         0    0.1833
    0.1043    0.1833         0


ulmetricone =

  Columns 1 through 15

         0   -8.4050   -8.4050   -8.4050   -8.4050   -8.0674   -8.4050   -8.0674   -8.4050   -8.0674   -8.4050   -8.0674   -8.0674   -8.4050   -8.4050
   -8.4050         0   -9.3713   -9.3713   -9.9633   -8.0674   -9.0082   -8.0674   -9.9633   -8.0674   -9.3317   -8.0674   -8.0674   -9.0082   -9.9633
   -8.4050   -9.3713         0  -10.4975   -9.3713   -8.0674   -9.0082   -8.0674   -9.3713   -8.0674   -9.3317   -8.0674   -8.0674   -9.0082   -9.3713
   -8.4050   -9.3713  -10.4975         0   -9.3713   -8.0674   -9.0082   -8.0674   -9.3713   -8.0674   -9.3317   -8.0674   -8.0674   -9.0082   -9.3713
   -8.4050   -9.9633   -9.3713   -9.3713         0   -8.0674   -9.0082   -8.0674  -12.7842   -8.0674   -9.3317   -8.0674   -8.0674   -9.0082  -13.9318
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674         0   -8.0674  -11.9148   -8.0674   -9.5620   -8.0674  -12.2735   -9.5620   -8.0674   -8.0674
   -8.4050   -9.0082   -9.0082   -9.0082   -9.0082   -8.0674         0   -8.0674   -9.0082   -8.0674   -9.0082   -8.0674   -8.0674  -11.3891   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674  -11.9148   -8.0674         0   -8.0674   -9.5620   -8.0674  -11.9148   -9.5620   -8.0674   -8.0674
   -8.4050   -9.9633   -9.3713   -9.3713  -12.7842   -8.0674   -9.0082   -8.0674         0   -8.0674   -9.3317   -8.0674   -8.0674   -9.0082  -12.7842
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.5620   -8.0674   -9.5620   -8.0674         0   -8.0674   -9.5620  -11.4588   -8.0674   -8.0674
   -8.4050   -9.3317   -9.3317   -9.3317   -9.3317   -8.0674   -9.0082   -8.0674   -9.3317   -8.0674         0   -8.0674   -8.0674   -9.0082   -9.3317
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674  -12.2735   -8.0674  -11.9148   -8.0674   -9.5620   -8.0674         0   -9.5620   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.5620   -8.0674   -9.5620   -8.0674  -11.4588   -8.0674   -9.5620         0   -8.0674   -8.0674
   -8.4050   -9.0082   -9.0082   -9.0082   -9.0082   -8.0674  -11.3891   -8.0674   -9.0082   -8.0674   -9.0082   -8.0674   -8.0674         0   -9.0082
   -8.4050   -9.9633   -9.3713   -9.3713  -13.9318   -8.0674   -9.0082   -8.0674  -12.7842   -8.0674   -9.3317   -8.0674   -8.0674   -9.0082         0
   -8.4050   -9.3317   -9.3317   -9.3317   -9.3317   -8.0674   -9.0082   -8.0674   -9.3317   -8.0674  -12.7838   -8.0674   -8.0674   -9.0082   -9.3317
   -9.9586   -8.4050   -8.4050   -8.4050   -8.4050   -8.0674   -8.4050   -8.0674   -8.4050   -8.0674   -8.4050   -8.0674   -8.0674   -8.4050   -8.4050
   -8.4050   -9.0082   -9.0082   -9.0082   -9.0082   -8.0674  -11.3891   -8.0674   -9.0082   -8.0674   -9.0082   -8.0674   -8.0674  -13.5869   -9.0082

  Columns 16 through 18

   -8.4050   -9.9586   -8.4050
   -9.3317   -8.4050   -9.0082
   -9.3317   -8.4050   -9.0082
   -9.3317   -8.4050   -9.0082
   -9.3317   -8.4050   -9.0082
   -8.0674   -8.0674   -8.0674
   -9.0082   -8.4050  -11.3891
   -8.0674   -8.0674   -8.0674
   -9.3317   -8.4050   -9.0082
   -8.0674   -8.0674   -8.0674
  -12.7838   -8.4050   -9.0082
   -8.0674   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674
   -9.0082   -8.4050  -13.5869
   -9.3317   -8.4050   -9.0082
         0   -8.4050   -9.0082
   -8.4050         0   -8.4050
```

```
   -9.0082   -8.4050         0
```

```
ctmetricone =

    7.4526
    7.9911
    7.8005
    7.3442
    8.0279
    7.9549
    7.6301
    7.8325
    7.4875
    7.9507
    7.6647
    8.0674
    7.2570
    8.0674
    7.9842
    7.7147
    7.1463
    7.9362
```

```
ulmetrictwo =

  Columns 1 through 15

         0   -0.5905   -1.3417   -1.3417   -0.5905   -1.1182   -0.5905   -0.5905   -1.3417   -0.5905   -0.5905   -0.5905   -1.1182   -0.5905   -1.1182
   -0.5905         0   -0.5905   -0.5905   -0.8094   -0.5905   -0.8094   -0.8094   -0.5905   -0.8325   -0.8094   -0.8325   -0.5905   -1.2339   -0.5905
   -1.3417   -0.5905         0   -2.7706   -0.5905   -1.1182   -0.5905   -0.5905   -3.7976   -0.5905   -0.5905   -0.5905   -1.1182   -0.5905   -1.1182
   -1.3417   -0.5905   -2.7706         0   -0.5905   -1.1182   -0.5905   -0.5905   -2.7706   -0.5905   -0.5905   -0.5905   -1.1182   -0.5905   -1.1182
   -0.5905   -0.8094   -0.5905   -0.5905         0   -0.5905   -1.2103   -2.0933   -0.5905   -0.8094   -1.2103   -0.8094   -0.5905   -0.8094   -0.5905
   -1.1182   -0.5905   -1.1182   -1.1182   -0.5905         0   -0.5905   -0.5905   -1.1182   -0.5905   -0.5905   -0.5905   -3.6866   -0.5905   -1.9509
   -0.5905   -0.8094   -0.5905   -0.5905   -1.2103   -0.5905         0   -1.2103   -0.5905   -0.8094   -2.8754   -0.8094   -0.5905   -0.8094   -0.5905
   -0.5905   -0.8094   -0.5905   -0.5905   -2.0933   -0.5905   -1.2103         0   -0.5905   -0.8094   -1.2103   -0.8094   -0.5905   -0.8094   -0.5905
   -1.3417   -0.5905   -3.7976   -2.7706   -0.5905   -1.1182   -0.5905   -0.5905         0   -0.5905   -0.5905   -0.5905   -1.1182   -0.5905   -1.1182
   -0.5905   -0.8325   -0.5905   -0.5905   -0.8094   -0.5905   -0.8094   -0.8094   -0.5905         0   -0.8094   -3.0026   -0.5905   -0.8325   -0.5905
   -0.5905   -0.8094   -0.5905   -0.5905   -1.2103   -0.5905   -2.8754   -1.2103   -0.5905   -0.8094         0   -0.8094   -0.5905   -0.8094   -0.5905
   -0.5905   -0.8325   -0.5905   -0.5905   -0.8094   -0.5905   -0.8094   -0.8094   -0.5905   -3.0026   -0.8094         0   -0.5905   -0.8325   -0.5905
   -1.1182   -0.5905   -1.1182   -1.1182   -0.5905   -3.6866   -0.5905   -0.5905   -1.1182   -0.5905   -0.5905   -0.5905         0   -0.5905   -1.9509
   -0.5905   -1.2339   -0.5905   -0.5905   -0.8094   -0.5905   -0.8094   -0.8094   -0.5905   -0.8325   -0.8094   -0.8325   -0.5905         0   -0.5905
   -1.1182   -0.5905   -1.1182   -1.1182   -0.5905   -1.9509   -0.5905   -0.5905   -1.1182   -0.5905   -0.5905   -0.5905   -1.9509   -0.5905         0
   -0.5905   -0.8325   -0.5905   -0.5905   -0.8094   -0.5905   -0.8094   -0.8094   -0.5905   -1.4028   -0.8094   -1.4028   -0.5905   -0.8325   -0.5905
   -0.5905   -2.8370   -0.5905   -0.5905   -0.8094   -0.5905   -0.8094   -0.8094   -0.5905   -0.8325   -0.8094   -0.8325   -0.5905   -1.2339   -0.5905
   -0.5905   -0.8094   -0.5905   -0.5905   -1.2103   -0.5905   -2.8754   -1.2103   -0.5905   -0.8094   -3.4736   -0.8094   -0.5905   -0.8094   -0.5905

  Columns 16 through 18

   -0.5905   -0.5905   -0.5905
   -0.8325   -2.8370   -0.8094
   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
   -0.8094   -0.8094   -1.2103
   -0.5905   -0.5905   -0.5905
   -0.8094   -0.8094   -2.8754
   -0.8094   -0.8094   -1.2103
   -0.5905   -0.5905   -0.5905
   -1.4028   -0.8325   -0.8094
   -0.8094   -0.8094   -3.4736
   -1.4028   -0.8325   -0.8094
   -0.5905   -0.5905   -0.5905
   -0.8325   -1.2339   -0.8094
   -0.5905   -0.5905   -0.5905
         0   -0.8325   -0.8094
   -0.8325         0   -0.8094
   -0.8094   -0.8094         0
```

```
ctmetrictwo =

    0.3477
    0.4023
    0.5905
    0.5263
    0.4001
    0.5183
    0.5531
    0.4001
    0.5905
    0.4232
    0.5905
    0.4232
    0.5183
    0.3021
    0.4098
```

```
    0.3232
    0.4023
    0.5905


orderproxone =

  Columns 1 through 15

         0  -11.9148  -11.9148   -9.5620   -9.5620   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674
  -11.9148         0  -12.2735   -9.5620   -9.5620   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674
  -11.9148  -12.2735         0   -9.5620   -9.5620   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674
   -9.5620   -9.5620   -9.5620         0  -11.4588   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674
   -9.5620   -9.5620   -9.5620  -11.4588         0   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674         0  -12.7838   -9.3317   -9.3317   -9.3317   -9.3317   -9.3317   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674  -12.7838         0   -9.3317   -9.3317   -9.3317   -9.3317   -9.3317   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.3317   -9.3317         0   -9.9633   -9.9633   -9.9633   -9.3713   -9.3713   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.3317   -9.3317   -9.9633         0  -13.9318  -12.7842   -9.3713   -9.3713   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.3317   -9.3317   -9.9633  -13.9318         0  -12.7842   -9.3713   -9.3713   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.3317   -9.3317   -9.9633  -12.7842  -12.7842         0   -9.3713   -9.3713   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.3317   -9.3317   -9.3713   -9.3713   -9.3713   -9.3713         0  -10.4975   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.3317   -9.3317   -9.3713   -9.3713   -9.3713   -9.3713  -10.4975         0   -9.0082   -9.0082
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082         0  -13.5869
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082  -13.5869         0
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082   -9.0082  -11.3891  -11.3891
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050
   -8.0674   -8.0674   -8.0674   -8.0674   -8.0674   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050   -8.4050

  Columns 16 through 18

   -8.0674   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674
   -8.0674   -8.0674   -8.0674
   -9.0082   -8.4050   -8.4050
   -9.0082   -8.4050   -8.4050
   -9.0082   -8.4050   -8.4050
   -9.0082   -8.4050   -8.4050
   -9.0082   -8.4050   -8.4050
   -9.0082   -8.4050   -8.4050
   -9.0082   -8.4050   -8.4050
  -11.3891   -8.4050   -8.4050
  -11.3891   -8.4050   -8.4050
         0   -8.4050   -8.4050
   -8.4050         0   -9.9586
   -8.4050   -9.9586         0


orderpermone =

     8     6    12    10    13    11    16     2     5    15     9     3     4    18    14     7    17     1


orderproxtwo =

  Columns 1 through 15

         0   -1.3417   -1.3417   -1.3417   -1.1182   -1.1182   -1.1182   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -1.3417         0   -2.7706   -2.7706   -1.1182   -1.1182   -1.1182   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -1.3417   -2.7706         0   -3.7976   -1.1182   -1.1182   -1.1182   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -1.3417   -2.7706   -3.7976         0   -1.1182   -1.1182   -1.1182   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -1.1182   -1.1182   -1.1182   -1.1182         0   -3.6866   -1.9509   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -1.1182   -1.1182   -1.1182   -1.1182   -3.6866         0   -1.9509   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -1.1182   -1.1182   -1.1182   -1.1182   -1.9509   -1.9509         0   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905         0   -1.4028   -1.4028   -0.8325   -0.8325   -0.8325   -0.8094   -0.8094
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -1.4028         0   -3.0026   -0.8325   -0.8325   -0.8325   -0.8094   -0.8094
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -1.4028   -3.0026         0   -0.8325   -0.8325   -0.8325   -0.8094   -0.8094
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8325   -0.8325   -0.8325         0   -1.2339   -1.2339   -0.8094   -0.8094
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8325   -0.8325   -0.8325   -1.2339         0   -2.8370   -0.8094   -0.8094
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8325   -0.8325   -0.8325   -1.2339   -2.8370         0   -0.8094   -0.8094
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094         0   -3.4736
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -3.4736         0
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -2.8754   -2.8754
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -1.2103   -1.2103
   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.5905   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -0.8094   -1.2103   -1.2103

  Columns 16 through 18

   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
   -0.5905   -0.5905   -0.5905
```

```
   -0.8094   -0.8094   -0.8094
   -0.8094   -0.8094   -0.8094
   -0.8094   -0.8094   -0.8094
   -0.8094   -0.8094   -0.8094
   -0.8094   -0.8094   -0.8094
   -0.8094   -0.8094   -0.8094
   -2.8754   -1.2103   -1.2103
   -2.8754   -1.2103   -1.2103
         0   -1.2103   -1.2103
   -1.2103         0   -2.0933
   -1.2103   -2.0933         0


orderpermtwo =

     1     4     3     9     6    13    15    16    10    12    14    17     2    18    11     7     8     5

Elapsed time is 10398.474202 seconds.
```

2.3.2 The Enhanced Script for ms_biatreefnd.m

```
load risk_rate.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(1000,1);

for i = 1:1000

    [find,vaf,targone,targtwo] = ...
        ms_biatreefnd(risk_rate,randperm(18));

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_find = find;
        best_targone = targone;
        best_targtwo = targtwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs;

best_vaf

best_find

best_targone

best_targtwo

[ulmetricone,ctmetricone] = atreedec(best_targone,0.0)

 [ulmetrictwo,ctmetrictwo] = atreedec(best_targtwo,0.0)

[orderproxone,orderpermone] = ultraorder(ulmetricone)

[orderproxtwo,orderpermtwo] = ultraorder(ulmetrictwo)


toc;
```

2.3.3 Enhanced Script Results for ms_biatreefnd.m

```
>> ms_script_biatreefnd

best_vaf =

    0.8768


best_find =
```

```
  Columns 1 through 15

       0    7.1980    6.4443    5.9240    7.2327    7.0879    6.9879    7.3750    6.1317    7.5163    7.0600    7.6330    6.3899    7.1743    6.6709
  7.1980         0    6.8231    6.3028    6.0487    8.2087    6.7588    7.7490    5.9178    7.8673    6.5074    7.9840    7.5107    6.5207    6.2336
  6.4443    6.8231         0    2.9955    6.8578    7.6783    6.9751    7.9654    3.3000    8.1067    6.7237    8.2234    6.9803    7.1615    6.2960
  5.9240    6.3028    2.9955         0    6.3375    7.1580    6.4548    7.4451    3.8045    7.5864    6.2034    7.7031    6.4600    6.6412    5.7757
  7.2327    6.0487    6.8578    6.3375         0    8.2433    6.3927    6.5000    3.1312    7.9250    6.1412    8.0417    7.5454    6.9799    2.2999
  7.0879    8.2087    7.6783    7.1580    8.2433         0    7.9985    4.2007    7.3657    6.6949    8.0706    4.1000    3.0000    8.1849    6.8490
  6.9879    6.7588    6.9751    6.4548    6.3927    7.9985         0    7.1381    6.6626    7.6802    4.5547    7.7970    7.3006    4.3543    6.9784
  7.3750    7.7490    7.9654    7.4451    6.5000    4.2007    7.1381         0    7.6528    6.2352    7.2102    3.9991    5.8556    7.7253    7.9687
  6.1317    5.9178    3.3000    3.8045    3.1312    7.3657    6.6626    7.6528         0    7.7941    6.4112    7.9109    6.6678    6.8490    2.5694
  7.5163    7.8673    8.1067    7.5864    7.9250    6.6949    7.6802    6.2352    7.7941         0    7.7523    4.3000    4.1000    7.8435    8.1100
  7.0600    6.5074    6.7237    6.2034    6.1412    8.0706    4.5547    7.2102    6.4112    7.7523         0    7.8691    7.3727    6.8072    6.7270
  7.6330    7.9840    8.2234    7.7031    8.0417    4.1000    7.7970    3.9991    7.9109    4.3000    7.8691         0    6.1136    7.9603    8.2267
  6.3899    7.5107    6.9803    6.4600    7.5454    3.0000    7.3006    5.8556    6.6678    4.1000    7.3727    6.1136         0    7.4870    6.1510
  7.1743    6.5207    7.1615    6.6412    6.9799    8.1849    4.3543    7.7253    6.8490    7.8435    6.8072    7.9603    7.4870         0    7.1648
  6.6709    6.2336    6.2960    5.7757    2.2999    6.8490    6.9784    7.9687    2.5694    8.1100    6.7270    8.2267    6.1510    7.1648         0
  6.8427    6.2671    6.5065    5.9862    6.3248    7.8534    6.4035    7.3938    6.1939    6.9416    2.6999    7.0584    7.1554    6.5668    6.5098
  4.7999    4.7000    6.9438    6.4235    6.7621    7.3641    6.5173    6.9044    6.6312    7.0226    6.5894    7.1394    6.6661    6.2793    6.9471
  7.3314    7.1023    7.3186    6.7983    6.7362    8.3421    2.4453    7.4816    7.0061    8.0237    4.3000    8.1405    7.6441    2.4999    7.3219

  Columns 16 through 18

  6.8427    4.7999    7.3314
  6.2671    4.7000    7.1023
  6.5065    6.9438    7.3186
  5.9862    6.4235    6.7983
  6.3248    6.7621    6.7362
  7.8534    7.3641    8.3421
  6.4035    6.5173    2.4453
  7.3938    6.9044    7.4816
  6.1939    6.6312    7.0061
  6.9416    7.0226    8.0237
  2.6999    6.5894    4.3000
  7.0584    7.1394    8.1405
  7.1554    6.6661    7.6441
  6.5668    6.2793    2.4999
  6.5098    6.9471    7.3219
       0    6.3491    6.7470
  6.3491         0    6.8608
  6.7470    6.8608         0


best_targone =

  Columns 1 through 15

       0    7.0386    6.8475    6.3913    7.0754    7.3402    6.6777    7.2178    6.5350    7.3360    6.7124    7.4527    6.6422    7.1150    7.0317
  7.0386         0    6.4206    5.9644    6.0557    7.8785    6.6129    7.7561    5.5153    7.8743    6.3241    7.9910    7.1806    7.0503    6.0120
  6.8475    6.4206         0    4.6518    6.4575    7.6875    6.4219    7.5651    5.9170    7.6833    6.1330    7.8000    6.9895    6.8592    6.4137
  6.3913    5.9644    4.6518         0    6.0012    7.2312    5.9656    7.1088    5.4608    7.2270    5.6768    7.3437    6.5333    6.4030    5.9575
  7.0754    6.0557    6.4575    6.0012         0    7.9154    6.6498    7.7930    2.7309    7.9112    6.3610    8.0279    7.2174    7.0871    2.0805
  7.3402    7.8785    7.6875    7.2312    7.9154         0    7.5176    3.8728    7.3750    6.3438    7.5523    3.7489    5.6501    7.9550    7.8716
  6.6777    6.6129    6.4219    5.9656    6.6498    7.5176         0    7.3952    6.1093    7.5134    6.2867    7.6301    6.8197    4.3086    6.6060
  7.2178    7.7561    7.5651    7.1088    7.7930    3.8728    7.3952         0    7.2525    6.2214    7.4299    3.9853    5.5276    7.8326    7.7492
  6.5350    5.5153    5.9170    5.4608    2.7309    7.3750    6.1093    7.2525         0    7.3707    5.8205    7.4875    6.6770    6.5467    2.6871
  7.3360    7.8743    7.6833    7.2270    7.9112    6.3438    7.5134    6.2214    7.3707         0    7.5481    6.4563    3.7489    7.9508    7.8674
  6.7124    6.3241    6.1330    5.6768    6.3610    7.5523    6.2867    7.4299    5.8205    7.5481         0    7.6648    6.8544    6.7241    6.3172
  7.4527    7.9910    7.8000    7.3437    8.0279    3.7489    7.6301    3.9853    7.4875    6.4563    7.6648         0    5.7626    8.0675    7.9841
  6.6422    7.1806    6.9895    6.5333    7.2174    5.6501    6.8197    5.5276    6.6770    3.7489    6.8544    5.7626         0    7.2570    7.1737
  7.1150    7.0503    6.8592    6.4030    7.0871    7.9550    4.3086    7.8326    6.5467    7.9508    6.7241    8.0675    7.2570         0    7.0434
  7.0317    6.0120    6.4137    5.9575    2.0805    7.8716    6.6060    7.7492    2.6871    7.8674    6.3172    7.9841    7.1737    7.0434         0
  6.7624    6.3741    6.1830    5.7268    6.4110    7.6023    6.3367    7.4799    5.8705    7.5981    2.5957    7.7148    6.9044    6.7741    6.3672
  4.6404    6.7323    6.5413    6.0850    6.7692    7.0339    6.3714    6.9115    6.2287    7.0297    6.4061    7.1464    6.3360    6.8088    6.7254
  6.9838    6.9190    6.7280    6.2717    6.9559    7.8238    4.1774    7.7013    6.4154    7.8195    6.5928    7.9363    7.1258    2.4168    6.9121

  Columns 16 through 18

  6.7624    4.6404    6.9838
  6.3741    6.7323    6.9190
  6.1830    6.5413    6.7280
  5.7268    6.0850    6.2717
  6.4110    6.7692    6.9559
  7.6023    7.0339    7.8238
  6.3367    6.3714    4.1774
  7.4799    6.9115    7.7013
  5.8705    6.2287    6.4154
  7.5981    7.0297    7.8195
  2.5957    6.4061    6.5928
  7.7148    7.1464    7.9363
  6.9044    6.3360    7.1258
  6.7741    6.8088    2.4168
  6.3672    6.7254    6.9121
       0    6.4561    6.6428
  6.4561         0    6.6775
  6.6428    6.6775         0
```

```
best_targtwo =

  Columns 1 through 15

         0    0.1594   -0.4033   -0.4673    0.1572   -0.2523    0.3102    0.1572   -0.4033    0.1804    0.3476    0.1804   -0.2523    0.0592   -0.3608
    0.1594         0    0.4025    0.3384   -0.0071    0.3301    0.1459   -0.0071    0.4025   -0.0071    0.1833   -0.0071    0.3301   -0.5295    0.2216
   -0.4033    0.4025         0   -1.6563    0.4003   -0.0092    0.5533    0.4003   -2.6170    0.4234    0.5907    0.4234   -0.0092    0.3023   -0.1177
   -0.4673    0.3384   -1.6563         0    0.3362   -0.0733    0.4892    0.3362   -1.6563    0.3594    0.5266    0.3594   -0.0733    0.2383   -0.1818
    0.1572   -0.0071    0.4003    0.3362         0    0.3279   -0.2571   -1.2930    0.4003    0.0138   -0.2197    0.0138    0.3279   -0.1073    0.2194
   -0.2523    0.3301   -0.0092   -0.0733    0.3279         0    0.4809    0.3279   -0.0092    0.3510    0.5183    0.3510   -2.6501    0.2299   -1.0227
    0.3102    0.1459    0.5533    0.4892   -0.2571    0.4809         0   -0.2571    0.5533    0.1668   -1.7320    0.1668    0.4809    0.0457    0.3724
    0.1572   -0.0071    0.4003    0.3362   -1.2930    0.3279   -0.2571         0    0.4003    0.0138   -0.2197    0.0138    0.3279   -0.1073    0.2194
   -0.4033    0.4025   -2.6170   -1.6563    0.4003   -0.0092    0.5533    0.4003         0    0.4234    0.5907    0.4234   -0.0092    0.3023   -0.1177
    0.1804   -0.0071    0.4234    0.3594    0.0138    0.3510    0.1668    0.0138    0.4234         0    0.2042   -2.1563    0.3510   -0.1072    0.2426
    0.3476    0.1833    0.5907    0.5266   -0.2197    0.5183   -1.7320   -0.2197    0.5907    0.2042         0    0.2042    0.5183    0.0831    0.4098
    0.1804   -0.0071    0.4234    0.3594    0.0138    0.3510    0.1668    0.0138    0.4234   -2.1563    0.2042         0    0.3510   -0.1072    0.2426
   -0.2523    0.3301   -0.0092   -0.0733    0.3279   -2.6501    0.4809    0.3279   -0.0092    0.3510    0.5183    0.3510         0    0.2299   -1.0227
    0.0592   -0.5295    0.3023    0.2383   -0.1073    0.2299    0.0457   -0.1073    0.3023   -0.1072    0.0831   -0.1072    0.2299         0    0.1214
   -0.3608    0.2216   -0.1177   -0.1818    0.2194   -1.0227    0.3724    0.2194   -0.1177    0.2426    0.4098    0.2426   -1.0227    0.1214         0
    0.0804   -0.1071    0.3234    0.2594   -0.0861    0.2511    0.0668   -0.0861    0.3234   -0.6565    0.1042   -0.6565    0.2511   -0.2072    0.1426
    0.1594   -2.0323    0.4025    0.3384   -0.0071    0.3301    0.1459   -0.0071    0.4025   -0.0071    0.1833   -0.0071    0.3301   -0.5295    0.2216
    0.3476    0.1833    0.5907    0.5266   -0.2197    0.5183   -1.7320   -0.2197    0.5907    0.2042   -2.2928    0.2042    0.5183    0.0831    0.4098

  Columns 16 through 18

    0.0804    0.1594    0.3476
   -0.1071   -2.0323    0.1833
    0.3234    0.4025    0.5907
    0.2594    0.3384    0.5266
   -0.0861   -0.0071   -0.2197
    0.2511    0.3301    0.5183
    0.0668    0.1459   -1.7320
   -0.0861   -0.0071   -0.2197
    0.3234    0.4025    0.5907
   -0.6565   -0.0071    0.2042
    0.1042    0.1833   -2.2928
   -0.6565   -0.0071    0.2042
    0.2511    0.3301    0.5183
   -0.2072   -0.5295    0.0831
    0.1426    0.2216    0.4098
         0   -0.1071    0.1042
   -0.1071         0    0.1833
    0.1042    0.1833         0


ulmetricone =

  Columns 1 through 15

         0   -8.4051   -8.4051   -8.4051   -8.4051   -8.0675   -8.4051   -8.0675   -8.4051   -8.0675   -8.4051   -8.0675   -8.0675   -8.4051   -8.4051
   -8.4051         0   -9.3704   -9.3704   -9.9632   -8.0675   -9.0083   -8.0675   -9.9632   -8.0675   -9.3318   -8.0675   -8.0675   -9.0083   -9.9632
   -8.4051   -9.3704         0  -10.4919   -9.3704   -8.0675   -9.0083   -8.0675   -9.3704   -8.0675   -9.3318   -8.0675   -8.0675   -9.0083   -9.3704
   -8.4051   -9.3704  -10.4919         0   -9.3704   -8.0675   -9.0083   -8.0675   -9.3704   -8.0675   -9.3318   -8.0675   -8.0675   -9.0083   -9.3704
   -8.4051   -9.9632   -9.3704   -9.3704         0   -8.0675   -9.0083   -8.0675  -12.7845   -8.0675   -9.3318   -8.0675   -8.0675   -9.0083  -13.9316
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675         0   -8.0675  -11.9148   -8.0675   -9.5620   -8.0675  -12.2736   -9.5620   -8.0675   -8.0675
   -8.4051   -9.0083   -9.0083   -9.0083   -9.0083   -8.0675         0   -8.0675   -9.0083   -8.0675   -9.0083   -8.0675   -8.0675  -11.3890   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675  -11.9148   -8.0675         0   -8.0675   -9.5620   -8.0675  -11.9148   -9.5620   -8.0675   -8.0675
   -8.4051   -9.9632   -9.3704   -9.3704  -12.7845   -8.0675   -9.0083   -8.0675         0   -8.0675   -9.3318   -8.0675   -8.0675   -9.0083  -12.7845
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.5620   -8.0675   -9.5620   -8.0675         0   -8.0675   -9.5620  -11.4589   -8.0675   -8.0675
   -8.4051   -9.3318   -9.3318   -9.3318   -9.3318   -8.0675   -9.0083   -8.0675   -9.3318   -8.0675         0   -8.0675   -8.0675   -9.0083   -9.3318
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675  -12.2736   -8.0675  -11.9148   -8.0675   -9.5620   -8.0675         0   -9.5620   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.5620   -8.0675   -9.5620   -8.0675  -11.4589   -8.0675   -9.5620         0   -8.0675   -8.0675
   -8.4051   -9.0083   -9.0083   -9.0083   -9.0083   -8.0675  -11.3890   -8.0675   -9.0083   -8.0675   -9.0083   -8.0675   -8.0675         0   -9.0083
   -8.4051   -9.9632   -9.3704   -9.3704  -13.9316   -8.0675   -9.0083   -8.0675  -12.7845   -8.0675   -9.3318   -8.0675   -8.0675   -9.0083         0
   -8.4051   -9.3318   -9.3318   -9.3318   -9.3318   -8.0675   -9.0083   -8.0675   -9.3318   -8.0675  -12.7840   -8.0675   -8.0675   -9.0083   -9.3318
   -9.9587   -8.4051   -8.4051   -8.4051   -8.4051   -8.0675   -8.4051   -8.0675   -8.4051   -8.0675   -8.4051   -8.0675   -8.0675   -8.4051   -8.4051
   -8.4051   -9.0083   -9.0083   -9.0083   -9.0083   -8.0675  -11.3890   -8.0675   -9.0083   -8.0675   -9.0083   -8.0675   -8.0675  -13.5870   -9.0083

  Columns 16 through 18

   -8.4051   -9.9587   -8.4051
   -9.3318   -8.4051   -9.0083
   -9.3318   -8.4051   -9.0083
   -9.3318   -8.4051   -9.0083
   -9.3318   -8.4051   -9.0083
   -8.0675   -8.0675   -8.0675
   -9.0083   -8.4051  -11.3890
   -8.0675   -8.0675   -8.0675
   -9.3318   -8.4051   -9.0083
   -8.0675   -8.0675   -8.0675
  -12.7840   -8.4051   -9.0083
   -8.0675   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675
   -9.0083   -8.4051  -13.5870
   -9.3318   -8.4051   -9.0083
```

```
         0   -8.4051   -9.0083
   -8.4051         0   -8.4051
   -9.0083   -8.4051         0


ctmetricone =

    7.4527
    7.9910
    7.8000
    7.3437
    8.0279
    7.9550
    7.6301
    7.8326
    7.4875
    7.9508
    7.6648
    8.0675
    7.2570
    8.0675
    7.9841
    7.7148
    7.1464
    7.9363


ulmetrictwo =

  Columns 1 through 15

         0   -0.5907   -1.3415   -1.3415   -0.5907   -1.1182   -0.5907   -0.5907   -1.3415   -0.5907   -0.5907   -0.5907   -1.1182   -0.5907   -1.1182
   -0.5907         0   -0.5907   -0.5907   -0.8099   -0.5907   -0.8099   -0.8099   -0.5907   -0.8330   -0.8099   -0.8330   -0.5907   -1.2343   -0.5907
   -1.3415   -0.5907         0   -2.7736   -0.5907   -1.1182   -0.5907   -0.5907   -3.7984   -0.5907   -0.5907   -0.5907   -1.1182   -0.5907   -1.1182
   -1.3415   -0.5907   -2.7736         0   -0.5907   -1.1182   -0.5907   -0.5907   -2.7736   -0.5907   -0.5907   -0.5907   -1.1182   -0.5907   -1.1182
   -0.5907   -0.8099   -0.5907   -0.5907         0   -0.5907   -1.2107   -2.0936   -0.5907   -0.8099   -1.2107   -0.8099   -0.5907   -0.8099   -0.5907
   -1.1182   -0.5907   -1.1182   -1.1182   -0.5907         0   -0.5907   -0.5907   -1.1182   -0.5907   -0.5907   -0.5907   -3.6867   -0.5907   -1.9508
   -0.5907   -0.8099   -0.5907   -0.5907   -1.2107   -0.5907         0   -1.2107   -0.5907   -0.8099   -2.8760   -0.8099   -0.5907   -0.8099   -0.5907
   -0.5907   -0.8099   -0.5907   -0.5907   -2.0936   -0.5907   -1.2107         0   -0.5907   -0.8099   -1.2107   -0.8099   -0.5907   -0.8099   -0.5907
   -1.3415   -0.5907   -3.7984   -2.7736   -0.5907   -1.1182   -0.5907   -0.5907         0   -0.5907   -0.5907   -0.5907   -1.1182   -0.5907   -1.1182
   -0.5907   -0.8330   -0.5907   -0.5907   -0.8099   -0.5907   -0.8099   -0.8099   -0.5907         0   -0.8099   -3.0031   -0.5907   -0.8330   -0.5907
   -0.5907   -0.8099   -0.5907   -0.5907   -1.2107   -0.5907   -2.8760   -1.2107   -0.5907   -0.8099         0   -0.8099   -0.5907   -0.8099   -0.5907
   -0.5907   -0.8330   -0.5907   -0.5907   -0.8099   -0.5907   -0.8099   -0.8099   -0.5907   -3.0031   -0.8099         0   -0.5907   -0.8330   -0.5907
   -1.1182   -0.5907   -1.1182   -1.1182   -0.5907   -3.6867   -0.5907   -0.5907   -1.1182   -0.5907   -0.5907   -0.5907         0   -0.5907   -1.9508
   -0.5907   -1.2343   -0.5907   -0.5907   -0.8099   -0.5907   -0.8099   -0.8099   -0.5907   -0.8330   -0.8099   -0.8330   -0.5907         0   -0.5907
   -1.1182   -0.5907   -1.1182   -1.1182   -0.5907   -1.9508   -0.5907   -0.5907   -1.1182   -0.5907   -0.5907   -0.5907   -1.9508   -0.5907         0
   -0.5907   -0.8330   -0.5907   -0.5907   -0.8099   -0.5907   -0.8099   -0.8099   -0.5907   -1.4033   -0.8099   -1.4033   -0.5907   -0.8330   -0.5907
   -0.5907   -2.8373   -0.5907   -0.5907   -0.8099   -0.5907   -0.8099   -0.8099   -0.5907   -0.8330   -0.8099   -0.8330   -0.5907   -1.2343   -0.5907
   -0.5907   -0.8099   -0.5907   -0.5907   -1.2107   -0.5907   -2.8760   -1.2107   -0.5907   -0.8099   -3.4741   -0.8099   -0.5907   -0.8099   -0.5907

  Columns 16 through 18

   -0.5907   -0.5907   -0.5907
   -0.8330   -2.8373   -0.8099
   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907
   -0.8099   -0.8099   -1.2107
   -0.5907   -0.5907   -0.5907
   -0.8099   -0.8099   -2.8760
   -0.8099   -0.8099   -1.2107
   -0.5907   -0.5907   -0.5907
   -1.4033   -0.8330   -0.8099
   -0.8099   -0.8099   -3.4741
   -1.4033   -0.8330   -0.8099
   -0.5907   -0.5907   -0.5907
   -0.8330   -1.2343   -0.8099
   -0.5907   -0.5907   -0.5907
         0   -0.8330   -0.8099
   -0.8330         0   -0.8099
   -0.8099   -0.8099         0


ctmetrictwo =

    0.3476
    0.4025
    0.5907
    0.5266
    0.4003
    0.5183
    0.5533
    0.4003
    0.5907
    0.4234
    0.5907
    0.4234
    0.5183
```

```
    0.3023
    0.4098
    0.3234
    0.4025
    0.5907


orderproxone =

  Columns 1 through 15

         0  -11.9148  -11.9148   -9.5620   -9.5620   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675
  -11.9148         0  -12.2736   -9.5620   -9.5620   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675
  -11.9148  -12.2736         0   -9.5620   -9.5620   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675
   -9.5620   -9.5620   -9.5620         0  -11.4589   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675
   -9.5620   -9.5620   -9.5620  -11.4589         0   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675         0  -12.7840   -9.3318   -9.3318   -9.3318   -9.3318   -9.3318   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675  -12.7840         0   -9.3318   -9.3318   -9.3318   -9.3318   -9.3318   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.3318   -9.3318         0   -9.9632   -9.9632   -9.9632   -9.3704   -9.3704   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.3318   -9.3318   -9.9632         0  -13.9316  -12.7845   -9.3704   -9.3704   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.3318   -9.3318   -9.9632  -13.9316         0  -12.7845   -9.3704   -9.3704   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.3318   -9.3318   -9.9632  -12.7845  -12.7845         0   -9.3704   -9.3704   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.3318   -9.3318   -9.3704   -9.3704   -9.3704   -9.3704         0  -10.4919   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.3318   -9.3318   -9.3704   -9.3704   -9.3704   -9.3704  -10.4919         0   -9.0083   -9.0083
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083         0  -13.5870
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083  -13.5870         0
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083   -9.0083  -11.3890  -11.3890
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051
   -8.0675   -8.0675   -8.0675   -8.0675   -8.0675   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051   -8.4051

  Columns 16 through 18

   -8.0675   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675
   -8.0675   -8.0675   -8.0675
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
   -9.0083   -8.4051   -8.4051
  -11.3890   -8.4051   -8.4051
  -11.3890   -8.4051   -8.4051
         0   -8.4051   -8.4051
   -8.4051         0   -9.9587
   -8.4051   -9.9587         0


orderpermone =

     8     6    12    10    13    11    16     2     5    15     9     3     4    18    14     7    17     1


orderproxtwo =

  Columns 1 through 15

         0   -1.3415   -1.3415   -1.3415   -1.1182   -1.1182   -1.1182   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -1.3415         0   -2.7736   -2.7736   -1.1182   -1.1182   -1.1182   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -1.3415   -2.7736         0   -3.7984   -1.1182   -1.1182   -1.1182   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -1.3415   -2.7736   -3.7984         0   -1.1182   -1.1182   -1.1182   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -1.1182   -1.1182   -1.1182   -1.1182         0   -3.6867   -1.9508   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -1.1182   -1.1182   -1.1182   -1.1182   -3.6867         0   -1.9508   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -1.1182   -1.1182   -1.1182   -1.1182   -1.9508   -1.9508         0   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907         0   -1.4033   -1.4033   -0.8330   -0.8330   -0.8330   -0.8099   -0.8099
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -1.4033         0   -3.0031   -0.8330   -0.8330   -0.8330   -0.8099   -0.8099
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -1.4033   -3.0031         0   -0.8330   -0.8330   -0.8330   -0.8099   -0.8099
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8330   -0.8330   -0.8330         0   -1.2343   -1.2343   -0.8099   -0.8099
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8330   -0.8330   -0.8330   -1.2343         0   -2.8373   -0.8099   -0.8099
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8330   -0.8330   -0.8330   -1.2343   -2.8373         0   -0.8099   -0.8099
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099         0   -3.4741
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -3.4741         0
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -2.8760   -2.8760
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -1.2107   -1.2107
   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.5907   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -0.8099   -1.2107   -1.2107

  Columns 16 through 18

   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907
```

```
   -0.5907   -0.5907   -0.5907
   -0.5907   -0.5907   -0.5907
   -0.8099   -0.8099   -0.8099
   -0.8099   -0.8099   -0.8099
   -0.8099   -0.8099   -0.8099
   -0.8099   -0.8099   -0.8099
   -0.8099   -0.8099   -0.8099
   -0.8099   -0.8099   -0.8099
   -2.8760   -1.2107   -1.2107
   -2.8760   -1.2107   -1.2107
         0   -1.2107   -1.2107
   -1.2107         0   -2.0936
   -1.2107   -2.0936         0


orderpermtwo =

     1     4     3     9     6    13    15    16    10    12    14    17     2    18    11     7     8     5

Elapsed time is 7080.033220 seconds.
```

2.4 (Multistructural) Metric City-Block Scaling

The various multidimensional scaling results presented in the next several sections, generally don't do as well in terms of VAFs as the corresponding biultrametric representation given earlier. Using the city-block distance measure we have VAFs of .6611 and .7721 for the metric and nonmetric instances, respectively; for the Euclidean case, the VAFs are .5492 and .7412 over the metric and nonmetric instances. Several points can be made about these representations:

(a) the city-block metric seems generally better than the Euclidean in VAFs, and irrespective of the use of a monotone transformation;

(b) there are general similarities in the corresponding graphical representations given in Figures 4 through 7 to the biultrametric results, although at least the one particular grouping of {1:accidental falls, 2:airplane accidents; 17:traffic accidents} seems to wander a bit in the figures, and is not well-placed at all in the Euclidean scalings;

(c) the range of VAFs achieved for the gradient-based Euclidean representations are very wide compared to the combinatorially-based city-block results, suggesting a greater proneness of the gradient optimization strategy to be trapped earlier at less well-fitting local optima;

(d) in the Euclidean scalings, the use of the default classical multidimensional scaling starting point does not necessarily do better than choosing the best VAF from a number of random starts. For example, in the metric case,

the classical multidimensional scaling (Torgerson) start produced a VAF of .5282, which was bested 33 out of 100 times by merely using a random initialization;

(e) generally, results do not seem to vary dramatically over either the choice of representational metric (city-block or Euclidean) or the imposition of a monotonic transformation.

2.4.1 Basic Script Results for biscalqa.m

```
>> script_biscalqa

sorted_vafs =

  Columns 1 through 15

    0.6051    0.6051    0.6303    0.6303    0.6303    0.6303    0.6303    0.6303    0.6303    0.6303    0.6303    0.6303    0.6303    0.6385    0.6385

  Columns 16 through 30

    0.6385    0.6385    0.6385    0.6390    0.6390    0.6390    0.6390    0.6390    0.6390    0.6419    0.6419    0.6419    0.6419    0.6419    0.6442

  Columns 31 through 45

    0.6443    0.6490    0.6492    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500    0.6500

  Columns 46 through 60

    0.6500    0.6500    0.6503    0.6503    0.6503    0.6503    0.6504    0.6504    0.6504    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556

  Columns 61 through 75

    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556    0.6556

  Columns 76 through 90

    0.6556    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611

  Columns 91 through 100

    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611    0.6611


best_vaf =

    0.6611


best_outpermone =

    18    14     7    11    16     2    17     3     4     9     5    15     1    13     6     8    12    10


best_outpermtwo =

     1    17     2     6    13    14    18    10     7    12     8    16    11     4     9    15     5     3


best_coordone =

   -1.8720
   -1.8720
   -1.7221
   -1.3087
   -0.8809
   -0.7399
   -0.3500
   -0.3472
   -0.3472
    0.0256
    0.0256
    0.0256
    0.0256
    1.3495
    1.7439
```

```
    1.8168
    2.1778
    2.2498


best_coordtwo =

   -1.5592
   -1.2448
   -1.0848
   -0.5365
   -0.5365
   -0.2780
   -0.2780
   -0.2780
   -0.2780
   -0.0267
   -0.0267
    0.2467
    0.2467
    0.8058
    1.1358
    1.1358
    1.1358
    1.4203


best_fitone =

  Columns 1 through 15

         0         0    0.1499    0.5634    0.9911    1.1321    1.5220    1.5248    1.5248    1.8976    1.8976    1.8976    1.8976    3.2215    3.6159
         0         0    0.1499    0.5634    0.9911    1.1321    1.5220    1.5248    1.5248    1.8976    1.8976    1.8976    1.8976    3.2215    3.6159
    0.1499    0.1499         0    0.4134    0.8412    0.9822    1.3721    1.3749    1.3749    1.7477    1.7477    1.7477    1.7477    3.0716    3.4660
    0.5634    0.5634    0.4134         0    0.4277    0.5687    0.9587    0.9615    0.9615    1.3342    1.3342    1.3342    1.3342    2.6582    3.0526
    0.9911    0.9911    0.8412    0.4277         0    0.1410    0.5309    0.5338    0.5338    0.9065    0.9065    0.9065    0.9065    2.2305    2.6248
    1.1321    1.1321    0.9822    0.5687    0.1410         0    0.3899    0.3927    0.3927    0.7655    0.7655    0.7655    0.7655    2.0894    2.4838
    1.5220    1.5220    1.3721    0.9587    0.5309    0.3899         0    0.0028    0.0028    0.3756    0.3756    0.3756    0.3756    1.6995    2.0939
    1.5248    1.5248    1.3749    0.9615    0.5338    0.3927    0.0028         0         0    0.3728    0.3728    0.3728    0.3728    1.6967    2.0911
    1.5248    1.5248    1.3749    0.9615    0.5338    0.3927    0.0028         0         0    0.3728    0.3728    0.3728    0.3728    1.6967    2.0911
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0         0         0    1.3239    1.7183
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0         0         0    1.3239    1.7183
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0         0    0.0000    1.3239    1.7183
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0    0.0000         0    1.3239    1.7183
    3.2215    3.2215    3.0716    2.6582    2.2305    2.0894    1.6995    1.6967    1.6967    1.3239    1.3239    1.3239    1.3239         0    0.3944
    3.6159    3.6159    3.4660    3.0526    2.6248    2.4838    2.0939    2.0911    2.0911    1.7183    1.7183    1.7183    1.7183    0.3944         0
    3.6888    3.6888    3.5389    3.1254    2.6977    2.5567    2.1668    2.1639    2.1639    1.7912    1.7912    1.7912    1.7912    0.4672    0.0728
    4.0498    4.0498    3.8999    3.4865    3.0588    2.9178    2.5278    2.5250    2.5250    2.1522    2.1522    2.1522    2.1522    0.8283    0.4339
    4.1218    4.1218    3.9719    3.5584    3.1307    2.9897    2.5998    2.5970    2.5970    2.2242    2.2242    2.2242    2.2242    0.9003    0.5059

  Columns 16 through 18

    3.6888    4.0498    4.1218
    3.6888    4.0498    4.1218
    3.5389    3.8999    3.9719
    3.1254    3.4865    3.5584
    2.6977    3.0588    3.1307
    2.5567    2.9178    2.9897
    2.1668    2.5278    2.5998
    2.1639    2.5250    2.5970
    2.1639    2.5250    2.5970
    1.7912    2.1522    2.2242
    1.7912    2.1522    2.2242
    1.7912    2.1522    2.2242
    1.7912    2.1522    2.2242
    0.4672    0.8283    0.9003
    0.0728    0.4339    0.5059
         0    0.3611    0.4330
    0.3611         0    0.0720
    0.4330    0.0720         0


best_fittwo =

  Columns 1 through 15

         0    0.3144    0.4744    1.0227    1.0227    1.2812    1.2812    1.2812    1.2812    1.5325    1.5325    1.8059    1.8059    2.3650    2.6950
    0.3144         0    0.1600    0.7083    0.7083    0.9668    0.9668    0.9668    0.9668    1.2181    1.2181    1.4915    1.4915    2.0505    2.3806
    0.4744    0.1600         0    0.5483    0.5483    0.8068    0.8068    0.8068    0.8068    1.0581    1.0581    1.3315    1.3315    1.8906    2.2206
    1.0227    0.7083    0.5483         0         0    0.2585    0.2585    0.2585    0.2585    0.5098    0.5098    0.7832    0.7832    1.3423    1.6723
    1.0227    0.7083    0.5483         0         0    0.2585    0.2585    0.2585    0.2585    0.5098    0.5098    0.7832    0.7832    1.3423    1.6723
    1.2812    0.9668    0.8068    0.2585    0.2585         0    0.0000         0         0    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.2812    0.9668    0.8068    0.2585    0.2585    0.0000         0         0         0    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.2812    0.9668    0.8068    0.2585    0.2585         0         0         0    0.0000    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.2812    0.9668    0.8068    0.2585    0.2585         0         0    0.0000         0    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.5325    1.2181    1.0581    0.5098    0.5098    0.2513    0.2513    0.2513    0.2513         0         0    0.2734    0.2734    0.8325    1.1625
    1.5325    1.2181    1.0581    0.5098    0.5098    0.2513    0.2513    0.2513    0.2513         0         0    0.2734    0.2734    0.8325    1.1625
```

```
    1.8059    1.4915    1.3315    0.7832    0.7832    0.5247    0.5247    0.5247    0.5247    0.2734    0.2734         0         0    0.5590    0.8891
    1.8059    1.4915    1.3315    0.7832    0.7832    0.5247    0.5247    0.5247    0.5247    0.2734    0.2734         0         0    0.5590    0.8891
    2.3650    2.0505    1.8906    1.3423    1.3423    1.0838    1.0838    1.0838    1.0838    0.8325    0.8325    0.5590    0.5590         0    0.3301
    2.6950    2.3806    2.2206    1.6723    1.6723    1.4138    1.4138    1.4138    1.4138    1.1625    1.1625    0.8891    0.8891    0.3301         0
    2.6950    2.3806    2.2206    1.6723    1.6723    1.4138    1.4138    1.4138    1.4138    1.1625    1.1625    0.8891    0.8891    0.3301         0
    2.6950    2.3806    2.2206    1.6723    1.6723    1.4138    1.4138    1.4138    1.4138    1.1625    1.1625    0.8891    0.8891    0.3301         0
    2.9795    2.6651    2.5051    1.9568    1.9568    1.6983    1.6983    1.6983    1.6983    1.4470    1.4470    1.1736    1.1736    0.6145    0.2844

  Columns 16 through 18

    2.6950    2.6950    2.9795
    2.3806    2.3806    2.6651
    2.2206    2.2206    2.5051
    1.6723    1.6723    1.9568
    1.6723    1.6723    1.9568
    1.4138    1.4138    1.6983
    1.4138    1.4138    1.6983
    1.4138    1.4138    1.6983
    1.4138    1.4138    1.6983
    1.1625    1.1625    1.4470
    1.1625    1.1625    1.4470
    0.8891    0.8891    1.1736
    0.8891    0.8891    1.1736
    0.3301    0.3301    0.6145
         0         0    0.2844
         0         0    0.2844
         0         0    0.2844
    0.2844    0.2844         0


best_addconone =

   -5.0503


best_addcontwo =

    1.0000

Elapsed time is 3489.366559 seconds.
```

2.4.2 The Enhanced Script for ms_biscalqa.m

```
load risk_rate.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(1000,1);

for i = 1:1000

    [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,...
        addcontwo,vaf] = ms_biscalqa(risk_rate,targlin(18),targlin(18), ...
        randperm(18),randperm(18),3,1);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_fitone = fitone;
        best_fittwo = fittwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs;

        best_vaf
        best_outpermone
        best_outpermtwo
```

```
        best_coordone
        best_coordtwo
        best_fitone
        best_fittwo
        best_addconone
        best_addcontwo

        axes = zeros(18,2);

        for i = 1:18

            axes(best_outpermone(i),1) = best_coordone(i);
            axes(best_outpermtwo(i),2) = best_coordtwo(i);
        end

        plot(axes(1:18,1),axes(1:18,2),'ko')

        hold on

        for i = 1:18

            objectlabels{i,1} = int2str(i);

        end

        text(axes(1:18,1),axes(1:18,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

toc;
```

2.4.3 Enhanced Script Results for ms_biscalqa.m

```
>> ms_script_biscalqa

best_vaf =

    0.6611


best_outpermone =

    18    14     7    11    16     2    17     3     4     9     5    15     1    13     6     8    12    10


best_outpermtwo =

     1    17     2     6    13    14    18    10     7    12     8    16    11     4     9    15     5     3


best_coordone =

   -1.8720
   -1.8720
   -1.7221
   -1.3087
   -0.8809
   -0.7399
   -0.3500
   -0.3472
   -0.3472
    0.0256
    0.0256
    0.0256
    0.0256
    1.3495
    1.7439
    1.8168
    2.1778
    2.2498


best_coordtwo =

   -1.5592
   -1.2448
   -1.0848
   -0.5365
   -0.5365
   -0.2780
   -0.2780
   -0.2780
   -0.2780
   -0.0267
```

```
   -0.0267
    0.2467
    0.2467
    0.8058
    1.1358
    1.1358
    1.1358
    1.4203
```

```
best_fitone =

  Columns 1 through 15

         0         0    0.1499    0.5634    0.9911    1.1321    1.5220    1.5248    1.5248    1.8976    1.8976    1.8976    1.8976    3.2215    3.6159
         0         0    0.1499    0.5634    0.9911    1.1321    1.5220    1.5248    1.5248    1.8976    1.8976    1.8976    1.8976    3.2215    3.6159
    0.1499    0.1499         0    0.4134    0.8412    0.9822    1.3721    1.3749    1.3749    1.7477    1.7477    1.7477    1.7477    3.0716    3.4660
    0.5634    0.5634    0.4134         0    0.4277    0.5687    0.9587    0.9615    0.9615    1.3342    1.3342    1.3342    1.3342    2.6582    3.0526
    0.9911    0.9911    0.8412    0.4277         0    0.1410    0.5309    0.5338    0.5338    0.9065    0.9065    0.9065    0.9065    2.2305    2.6248
    1.1321    1.1321    0.9822    0.5687    0.1410         0    0.3899    0.3927    0.3927    0.7655    0.7655    0.7655    0.7655    2.0894    2.4838
    1.5220    1.5220    1.3721    0.9587    0.5309    0.3899         0    0.0028    0.0028    0.3756    0.3756    0.3756    0.3756    1.6995    2.0939
    1.5248    1.5248    1.3749    0.9615    0.5338    0.3927    0.0028         0         0    0.3728    0.3728    0.3728    0.3728    1.6967    2.0911
    1.5248    1.5248    1.3749    0.9615    0.5338    0.3927    0.0028         0         0    0.3728    0.3728    0.3728    0.3728    1.6967    2.0911
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0         0         0    1.3239    1.7183
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0         0         0    1.3239    1.7183
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0         0    0.0000    1.3239    1.7183
    1.8976    1.8976    1.7477    1.3342    0.9065    0.7655    0.3756    0.3728    0.3728         0         0    0.0000         0    1.3239    1.7183
    3.2215    3.2215    3.0716    2.6582    2.2305    2.0894    1.6995    1.6967    1.6967    1.3239    1.3239    1.3239    1.3239         0    0.3944
    3.6159    3.6159    3.4660    3.0526    2.6248    2.4838    2.0939    2.0911    2.0911    1.7183    1.7183    1.7183    1.7183    0.3944         0
    3.6888    3.6888    3.5389    3.1254    2.6977    2.5567    2.1668    2.1639    2.1639    1.7912    1.7912    1.7912    1.7912    0.4672    0.0728
    4.0498    4.0498    3.8999    3.4865    3.0588    2.9178    2.5278    2.5250    2.5250    2.1522    2.1522    2.1522    2.1522    0.8283    0.4339
    4.1218    4.1218    3.9719    3.5584    3.1307    2.9897    2.5998    2.5970    2.5970    2.2242    2.2242    2.2242    2.2242    0.9003    0.5059

  Columns 16 through 18

    3.6888    4.0498    4.1218
    3.6888    4.0498    4.1218
    3.5389    3.8999    3.9719
    3.1254    3.4865    3.5584
    2.6977    3.0588    3.1307
    2.5567    2.9178    2.9897
    2.1668    2.5278    2.5998
    2.1639    2.5250    2.5970
    2.1639    2.5250    2.5970
    1.7912    2.1522    2.2242
    1.7912    2.1522    2.2242
    1.7912    2.1522    2.2242
    1.7912    2.1522    2.2242
    0.4672    0.8283    0.9003
    0.0728    0.4339    0.5059
         0    0.3611    0.4330
    0.3611         0    0.0720
    0.4330    0.0720         0
```

```
best_fittwo =

  Columns 1 through 15

         0    0.3144    0.4744    1.0227    1.0227    1.2812    1.2812    1.2812    1.2812    1.5325    1.5325    1.8059    1.8059    2.3650    2.6950
    0.3144         0    0.1600    0.7083    0.7083    0.9668    0.9668    0.9668    0.9668    1.2181    1.2181    1.4915    1.4915    2.0505    2.3806
    0.4744    0.1600         0    0.5483    0.5483    0.8068    0.8068    0.8068    0.8068    1.0581    1.0581    1.3315    1.3315    1.8906    2.2206
    1.0227    0.7083    0.5483         0         0    0.2585    0.2585    0.2585    0.2585    0.5098    0.5098    0.7832    0.7832    1.3423    1.6723
    1.0227    0.7083    0.5483         0         0    0.2585    0.2585    0.2585    0.2585    0.5098    0.5098    0.7832    0.7832    1.3423    1.6723
    1.2812    0.9668    0.8068    0.2585    0.2585         0    0.0000         0         0    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.2812    0.9668    0.8068    0.2585    0.2585    0.0000         0         0         0    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.2812    0.9668    0.8068    0.2585    0.2585         0         0         0    0.0000    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.2812    0.9668    0.8068    0.2585    0.2585         0         0    0.0000         0    0.2513    0.2513    0.5247    0.5247    1.0838    1.4138
    1.5325    1.2181    1.0581    0.5098    0.5098    0.2513    0.2513    0.2513    0.2513         0         0    0.2734    0.2734    0.8325    1.1625
    1.5325    1.2181    1.0581    0.5098    0.5098    0.2513    0.2513    0.2513    0.2513         0         0    0.2734    0.2734    0.8325    1.1625
    1.8059    1.4915    1.3315    0.7832    0.7832    0.5247    0.5247    0.5247    0.5247    0.2734    0.2734         0         0    0.5590    0.8891
    1.8059    1.4915    1.3315    0.7832    0.7832    0.5247    0.5247    0.5247    0.5247    0.2734    0.2734         0         0    0.5590    0.8891
    2.3650    2.0505    1.8906    1.3423    1.3423    1.0838    1.0838    1.0838    1.0838    0.8325    0.8325    0.5590    0.5590         0    0.3301
    2.6950    2.3806    2.2206    1.6723    1.6723    1.4138    1.4138    1.4138    1.4138    1.1625    1.1625    0.8891    0.8891    0.3301         0
    2.6950    2.3806    2.2206    1.6723    1.6723    1.4138    1.4138    1.4138    1.4138    1.1625    1.1625    0.8891    0.8891    0.3301         0
    2.6950    2.3806    2.2206    1.6723    1.6723    1.4138    1.4138    1.4138    1.4138    1.1625    1.1625    0.8891    0.8891    0.3301         0
    2.9795    2.6651    2.5051    1.9568    1.9568    1.6983    1.6983    1.6983    1.6983    1.4470    1.4470    1.1736    1.1736    0.6145    0.2844

  Columns 16 through 18

    2.6950    2.6950    2.9795
    2.3806    2.3806    2.6651
    2.2206    2.2206    2.5051
    1.6723    1.6723    1.9568
    1.6723    1.6723    1.9568
    1.4138    1.4138    1.6983
    1.4138    1.4138    1.6983
```

```
    1.4138    1.4138    1.6983
    1.4138    1.4138    1.6983
    1.1625    1.1625    1.4470
    1.1625    1.1625    1.4470
    0.8891    0.8891    1.1736
    0.8891    0.8891    1.1736
    0.3301    0.3301    0.6145
         0         0    0.2844
         0         0    0.2844
         0         0    0.2844
    0.2844    0.2844         0


best_addconone =

   -5.0503


best_addcontwo =

    1.0000

Elapsed time is 3311.755736 seconds.
```

2.5 (Multistructural) Nonmetric City-block Scaling

2.5.1 Basic Script Results for bimonscalqa.m

```
>> script_bimonscalqa

sorted_vafs =

  Columns 1 through 15

    0.7307    0.7307    0.7381    0.7391    0.7399    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408

  Columns 16 through 30

    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7408    0.7421    0.7451    0.7451    0.7451    0.7451

  Columns 31 through 45

    0.7457    0.7457    0.7457    0.7457    0.7457    0.7457    0.7457    0.7457    0.7457    0.7458    0.7470    0.7481    0.7481    0.7481    0.7492

  Columns 46 through 60

    0.7492    0.7492    0.7492    0.7492    0.7492    0.7492    0.7492    0.7492    0.7492    0.7492    0.7494    0.7494    0.7494    0.7498    0.7498

  Columns 61 through 75

    0.7518    0.7525    0.7534    0.7588    0.7588    0.7588    0.7600    0.7603    0.7603    0.7603    0.7603    0.7603    0.7603    0.7603    0.7603

  Columns 76 through 90

    0.7603    0.7603    0.7638    0.7638    0.7638    0.7638    0.7638    0.7641    0.7641    0.7641    0.7641    0.7646    0.7646    0.7646    0.7646

  Columns 91 through 100

    0.7646    0.7690    0.7721    0.7721    0.7721    0.7721    0.7721    0.7721    0.7721    0.7721


best_vaf =

    0.7721


best_outpermone =

     5    15     9     3     2     4    16    11     1     7    14    17    18    13     6     8    10    12


best_outpermtwo =

     1    17     6    13     3    15     9     4    10    16     8    12     5    11     2     7    14    18


best_coordone =

   -0.2416
   -0.2416
   -0.1863
```

```
   -0.1863
   -0.1727
   -0.1387
   -0.0767
   -0.0767
   -0.0584
   -0.0347
   -0.0347
   -0.0003
   -0.0003
    0.2050
    0.2820
    0.2820
    0.3358
    0.3442


best_coordtwo =

   -0.2044
   -0.1367
   -0.1287
   -0.1287
   -0.0897
   -0.0834
   -0.0834
   -0.0460
   -0.0286
    0.0162
    0.0195
    0.0195
    0.0195
    0.0706
    0.1355
    0.1992
    0.2247
    0.2247


best_fitone =

  Columns 1 through 15

         0         0    0.0553    0.0553    0.0689    0.1029    0.1649    0.1649    0.1832    0.2069    0.2069    0.2413    0.2413    0.4466    0.5236
         0         0    0.0553    0.0553    0.0689    0.1029    0.1649    0.1649    0.1832    0.2069    0.2069    0.2413    0.2413    0.4466    0.5236
    0.0553    0.0553         0         0    0.0137    0.0477    0.1096    0.1096    0.1279    0.1517    0.1517    0.1861    0.1861    0.3913    0.4683
    0.0553    0.0553         0         0    0.0137    0.0477    0.1096    0.1096    0.1279    0.1517    0.1517    0.1861    0.1861    0.3913    0.4683
    0.0689    0.0689    0.0137    0.0137         0    0.0340    0.0960    0.0960    0.1142    0.1380    0.1380    0.1724    0.1724    0.3776    0.4546
    0.1029    0.1029    0.0477    0.0477    0.0340         0    0.0620    0.0620    0.0802    0.1040    0.1040    0.1384    0.1384    0.3436    0.4207
    0.1649    0.1649    0.1096    0.1096    0.0960    0.0620         0         0    0.0183    0.0420    0.0420    0.0764    0.0764    0.2817    0.3587
    0.1649    0.1649    0.1096    0.1096    0.0960    0.0620         0         0    0.0183    0.0420    0.0420    0.0764    0.0764    0.2817    0.3587
    0.1832    0.1832    0.1279    0.1279    0.1142    0.0802    0.0183    0.0183         0    0.0238    0.0238    0.0582    0.0582    0.2634    0.3404
    0.2069    0.2069    0.1517    0.1517    0.1380    0.1040    0.0420    0.0420    0.0238         0         0    0.0344    0.0344    0.2396    0.3167
    0.2069    0.2069    0.1517    0.1517    0.1380    0.1040    0.0420    0.0420    0.0238         0         0    0.0344    0.0344    0.2396    0.3167
    0.2413    0.2413    0.1861    0.1861    0.1724    0.1384    0.0764    0.0764    0.0582    0.0344    0.0344         0         0    0.2053    0.2823
    0.2413    0.2413    0.1861    0.1861    0.1724    0.1384    0.0764    0.0764    0.0582    0.0344    0.0344         0         0    0.2053    0.2823
    0.4466    0.4466    0.3913    0.3913    0.3776    0.3436    0.2817    0.2817    0.2634    0.2396    0.2396    0.2053    0.2053         0    0.0770
    0.5236    0.5236    0.4683    0.4683    0.4546    0.4207    0.3587    0.3587    0.3404    0.3167    0.3167    0.2823    0.2823    0.0770         0
    0.5236    0.5236    0.4683    0.4683    0.4546    0.4207    0.3587    0.3587    0.3404    0.3167    0.3167    0.2823    0.2823    0.0770         0
    0.5774    0.5774    0.5221    0.5221    0.5084    0.4744    0.4125    0.4125    0.3942    0.3704    0.3704    0.3360    0.3360    0.1308    0.0538
    0.5858    0.5858    0.5306    0.5306    0.5169    0.4829    0.4209    0.4209    0.4027    0.3789    0.3789    0.3445    0.3445    0.1393    0.0622

  Columns 16 through 18

    0.5236    0.5774    0.5858
    0.5236    0.5774    0.5858
    0.4683    0.5221    0.5306
    0.4683    0.5221    0.5306
    0.4546    0.5084    0.5169
    0.4207    0.4744    0.4829
    0.3587    0.4125    0.4209
    0.3587    0.4125    0.4209
    0.3404    0.3942    0.4027
    0.3167    0.3704    0.3789
    0.3167    0.3704    0.3789
    0.2823    0.3360    0.3445
    0.2823    0.3360    0.3445
    0.0770    0.1308    0.1393
         0    0.0538    0.0622
         0    0.0538    0.0622
    0.0538         0    0.0085
    0.0622    0.0085         0


best_fittwo =

  Columns 1 through 15
```

```
         0    0.0676    0.0757    0.0757    0.1147    0.1210    0.1210    0.1583    0.1758    0.2205    0.2239    0.2239    0.2239    0.2749    0.3399
    0.0676         0    0.0080    0.0080    0.0470    0.0534    0.0534    0.0907    0.1082    0.1529    0.1563    0.1563    0.1563    0.2073    0.2722
    0.0757    0.0080         0         0    0.0390    0.0453    0.0453    0.0827    0.1001    0.1449    0.1482    0.1482    0.1482    0.1993    0.2642
    0.0757    0.0080         0         0    0.0390    0.0453    0.0453    0.0827    0.1001    0.1449    0.1482    0.1482    0.1482    0.1993    0.2642
    0.1147    0.0470    0.0390    0.0390         0    0.0063    0.0063    0.0437    0.0611    0.1059    0.1092    0.1092    0.1092    0.1603    0.2252
    0.1210    0.0534    0.0453    0.0453    0.0063         0         0    0.0373    0.0548    0.0995    0.1029    0.1029    0.1029    0.1539    0.2189
    0.1210    0.0534    0.0453    0.0453    0.0063         0         0    0.0373    0.0548    0.0995    0.1029    0.1029    0.1029    0.1539    0.2189
    0.1583    0.0907    0.0827    0.0827    0.0437    0.0373    0.0373         0    0.0175    0.0622    0.0656    0.0656    0.0656    0.1166    0.1815
    0.1758    0.1082    0.1001    0.1001    0.0611    0.0548    0.0548    0.0175         0    0.0447    0.0481    0.0481    0.0481    0.0991    0.1641
    0.2205    0.1529    0.1449    0.1449    0.1059    0.0995    0.0995    0.0622    0.0447         0    0.0034    0.0034    0.0034    0.0544    0.1193
    0.2239    0.1563    0.1482    0.1482    0.1092    0.1029    0.1029    0.0656    0.0481    0.0034         0         0         0    0.0511    0.1160
    0.2239    0.1563    0.1482    0.1482    0.1092    0.1029    0.1029    0.0656    0.0481    0.0034         0         0    0.0000    0.0511    0.1160
    0.2239    0.1563    0.1482    0.1482    0.1092    0.1029    0.1029    0.0656    0.0481    0.0034         0    0.0000         0    0.0511    0.1160
    0.2749    0.2073    0.1993    0.1993    0.1603    0.1539    0.1539    0.1166    0.0991    0.0544    0.0511    0.0511    0.0511         0    0.0649
    0.3399    0.2722    0.2642    0.2642    0.2252    0.2189    0.2189    0.1815    0.1641    0.1193    0.1160    0.1160    0.1160    0.0649         0
    0.4036    0.3360    0.3279    0.3279    0.2889    0.2826    0.2826    0.2453    0.2278    0.1831    0.1797    0.1797    0.1797    0.1286    0.0637
    0.4291    0.3615    0.3534    0.3534    0.3144    0.3081    0.3081    0.2708    0.2533    0.2086    0.2052    0.2052    0.2052    0.1542    0.0892
    0.4291    0.3615    0.3534    0.3534    0.3144    0.3081    0.3081    0.2708    0.2533    0.2086    0.2052    0.2052    0.2052    0.1542    0.0892

  Columns 16 through 18

    0.4036    0.4291    0.4291
    0.3360    0.3615    0.3615
    0.3279    0.3534    0.3534
    0.3279    0.3534    0.3534
    0.2889    0.3144    0.3144
    0.2826    0.3081    0.3081
    0.2826    0.3081    0.3081
    0.2453    0.2708    0.2708
    0.2278    0.2533    0.2533
    0.1831    0.2086    0.2086
    0.1797    0.2052    0.2052
    0.1797    0.2052    0.2052
    0.1797    0.2052    0.2052
    0.1286    0.1542    0.1542
    0.0637    0.0892    0.0892
         0    0.0255    0.0255
    0.0255         0         0
    0.0255         0         0


best_addconone =

   -6.5376


best_addcontwo =

    0.1490


best_monprox =

  Columns 1 through 15

         0    6.9486    6.6446    6.6049    6.6446    6.9017    6.6446    6.9017    6.6446    6.9486    6.7605    7.0413    6.6049    6.9486    6.6446
    6.9486         0    6.6813    6.6813    6.6049    7.0413    6.6446    7.0167    6.6049    7.0413    6.6446    7.0413    6.9486    6.6049    6.6446
    6.6446    6.6813         0    6.3551    6.6049    6.9486    6.9017    7.0255    6.3551    7.0413    6.6446    7.0413    6.9486    6.7605    6.6049
    6.6049    6.6813    6.3551         0    6.6813    7.0413    6.6446    6.9017    6.3869    6.9017    6.6446    6.9486    6.9017    6.6446    6.6049
    6.6446    6.6049    6.6049    6.6813         0    7.1219    6.6049    6.6446    6.3832    6.9486    6.6446    7.0413    7.0413    6.7605    6.3551
    6.9017    7.0413    6.9486    7.0413    7.1219         0    6.9486    6.5222    6.6049    6.6446    7.0413    6.4933    6.3551    7.0413    7.0413
    6.6446    6.6446    6.9017    6.6446    6.6049    6.9486         0    6.9486    7.0413    7.0413    6.4933    7.0413    6.9486    6.3832    6.9120
    6.9017    7.0167    7.0255    6.9017    6.6446    6.5222    6.9486         0    7.0413    6.6049    6.6813    6.3832    6.6049    7.0413    7.0413
    6.6446    6.6049    6.3551    6.3869    6.3832    6.6049    7.0413    7.0413         0    7.0069    6.6446    6.9017    6.6446    6.7605    6.3551
    6.9486    7.0413    7.0413    6.9017    6.9486    6.6446    7.0413    6.6049    7.0069         0    7.0413    6.4933    6.4933    6.9486    7.1219
    6.7605    6.6446    6.6446    6.6446    6.6446    7.0413    6.4933    6.6813    6.6446    7.0413         0    6.6813    7.0413    6.7605    6.6446
    7.0413    7.0413    7.0413    6.9486    7.0413    6.4933    7.0413    6.3832    6.9017    6.4933    6.6813         0    6.7605    7.0413    7.0413
    6.6049    6.9486    6.9486    6.9017    7.0413    6.3551    6.9486    6.6049    6.6446    6.4933    7.0413    6.7605         0    6.9017    6.6049
    6.9486    6.6049    6.7605    6.6446    6.7605    7.0413    6.3832    7.0413    6.7605    6.9486    6.7605    7.0413    6.9017         0    6.9017
    6.6446    6.6446    6.6049    6.6049    6.3551    7.0413    6.9120    7.0413    6.3551    7.1219    6.6446    7.0413    6.6049    6.9017         0
    6.6446    6.6446    6.6446    6.6049    6.6446    6.9486    6.6446    6.6446    6.6446    6.7605    6.3551    6.7605    6.9017    6.6446    6.6049
    6.6049    6.6049    6.6813    6.6049    6.9017    6.6446    6.6049    6.9486    6.7605    6.6813    6.6446    6.9486    6.6049    6.7903    6.9017
    6.9088    6.6446    6.9017    6.6446    6.9017    7.0413    6.3551    6.9017    6.7605    6.9017    6.5222    7.0069    7.0012    6.3551    6.9892

  Columns 16 through 18

    6.6446    6.6049    6.9088
    6.6446    6.6049    6.6446
    6.6446    6.6813    6.9017
    6.6049    6.6049    6.6446
    6.6446    6.9017    6.9017
    6.9486    6.6446    7.0413
    6.6446    6.6049    6.3551
    6.6446    6.9486    6.9017
    6.6446    6.7605    6.7605
    6.7605    6.6813    6.9017
```

```
    6.3551    6.6446    6.5222
    6.7605    6.9486    7.0069
    6.9017    6.6049    7.0012
    6.6446    6.7903    6.3551
    6.6049    6.9017    6.9892
         0    6.6049    6.9486
    6.6049         0    6.7460
    6.9486    6.7460         0

Elapsed time is 26397.414245 seconds.
```

2.5.2 The Enhanced Script for ms_bimonscalqa.m

```
load risk_rate.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(1000,1);

for i = 1:1000

    [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,...
        addcontwo,vaf,monprox] = ms_bimonscalqa(risk_rate,targlin(18),targlin(18), ...
        randperm(18),randperm(18),3,1);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_fitone = fitone;
        best_fittwo = fittwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;
        best_monprox = monprox;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs;

        best_vaf
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo
        best_fitone
        best_fittwo
        best_addconone
        best_addcontwo
        best_monprox

         axes = zeros(18,2);

        for i = 1:18

            axes(best_outpermone(i),1) = best_coordone(i);
            axes(best_outpermtwo(i),2) = best_coordtwo(i);
        end

        plot(axes(1:18,1),axes(1:18,2),'ko')

        hold on

        for i = 1:18

            objectlabels{i,1} = int2str(i);

        end

        text(axes(1:18,1),axes(1:18,2),objectlabels,'fontsize',10,'verticalalignment','bottom')
```

```
toc;
```

2.5.3 Enhanced Script Results for ms_bimonscalqa.m

```
>> ms_script_bimonscalqa

best_vaf =

    0.7721


best_outpermone =

     5    15     9     3     2     4    16    11     1     7    14    17    18    13     6     8    10    12


best_outpermtwo =

     1    17     6    13     3    15     9     4    10    16     8    12     5    11     2     7    14    18


best_coordone =

   -0.2416
   -0.2416
   -0.1863
   -0.1863
   -0.1727
   -0.1387
   -0.0767
   -0.0767
   -0.0584
   -0.0347
   -0.0347
   -0.0003
   -0.0003
    0.2050
    0.2820
    0.2820
    0.3358
    0.3442


best_coordtwo =

   -0.2044
   -0.1367
   -0.1287
   -0.1287
   -0.0897
   -0.0834
   -0.0834
   -0.0460
   -0.0286
    0.0162
    0.0195
    0.0195
    0.0195
    0.0706
    0.1355
    0.1992
    0.2247
    0.2247


best_fitone =

  Columns 1 through 15

         0         0    0.0553    0.0553    0.0689    0.1029    0.1649    0.1649    0.1832    0.2069    0.2069    0.2413    0.2413    0.4466    0.5236
         0         0    0.0553    0.0553    0.0689    0.1029    0.1649    0.1649    0.1832    0.2069    0.2069    0.2413    0.2413    0.4466    0.5236
    0.0553    0.0553         0         0    0.0137    0.0477    0.1096    0.1096    0.1279    0.1517    0.1517    0.1861    0.1861    0.3913    0.4683
    0.0553    0.0553         0         0    0.0137    0.0477    0.1096    0.1096    0.1279    0.1517    0.1517    0.1861    0.1861    0.3913    0.4683
    0.0689    0.0689    0.0137    0.0137         0    0.0340    0.0960    0.0960    0.1142    0.1380    0.1380    0.1724    0.1724    0.3776    0.4546
    0.1029    0.1029    0.0477    0.0477    0.0340         0    0.0620    0.0620    0.0802    0.1040    0.1040    0.1384    0.1384    0.3436    0.4207
    0.1649    0.1649    0.1096    0.1096    0.0960    0.0620         0         0    0.0183    0.0420    0.0420    0.0764    0.0764    0.2817    0.3587
    0.1649    0.1649    0.1096    0.1096    0.0960    0.0620         0         0    0.0183    0.0420    0.0420    0.0764    0.0764    0.2817    0.3587
    0.1832    0.1832    0.1279    0.1279    0.1142    0.0802    0.0183    0.0183         0    0.0238    0.0238    0.0582    0.0582    0.2634    0.3404
    0.2069    0.2069    0.1517    0.1517    0.1380    0.1040    0.0420    0.0420    0.0238         0         0    0.0344    0.0344    0.2396    0.3167
    0.2069    0.2069    0.1517    0.1517    0.1380    0.1040    0.0420    0.0420    0.0238         0         0    0.0344    0.0344    0.2396    0.3167
    0.2413    0.2413    0.1861    0.1861    0.1724    0.1384    0.0764    0.0764    0.0582    0.0344    0.0344         0         0    0.2053    0.2823
    0.2413    0.2413    0.1861    0.1861    0.1724    0.1384    0.0764    0.0764    0.0582    0.0344    0.0344         0         0    0.2053    0.2823
    0.4466    0.4466    0.3913    0.3913    0.3776    0.3436    0.2817    0.2817    0.2634    0.2396    0.2396    0.2053    0.2053         0    0.0770
```

```
    0.5236    0.5236    0.4683    0.4683    0.4546    0.4207    0.3587    0.3587    0.3404    0.3167    0.3167    0.2823    0.2823    0.0770         0
    0.5236    0.5236    0.4683    0.4683    0.4546    0.4207    0.3587    0.3587    0.3404    0.3167    0.3167    0.2823    0.2823    0.0770         0
    0.5774    0.5774    0.5221    0.5221    0.5084    0.4744    0.4125    0.4125    0.3942    0.3704    0.3704    0.3360    0.3360    0.1308    0.0538
    0.5858    0.5858    0.5306    0.5306    0.5169    0.4829    0.4209    0.4209    0.4027    0.3789    0.3789    0.3445    0.3445    0.1393    0.0622

  Columns 16 through 18

    0.5236    0.5774    0.5858
    0.5236    0.5774    0.5858
    0.4683    0.5221    0.5306
    0.4683    0.5221    0.5306
    0.4546    0.5084    0.5169
    0.4207    0.4744    0.4829
    0.3587    0.4125    0.4209
    0.3587    0.4125    0.4209
    0.3404    0.3942    0.4027
    0.3167    0.3704    0.3789
    0.3167    0.3704    0.3789
    0.2823    0.3360    0.3445
    0.2823    0.3360    0.3445
    0.0770    0.1308    0.1393
         0    0.0538    0.0622
         0    0.0538    0.0622
    0.0538         0    0.0085
    0.0622    0.0085         0


best_fittwo =

  Columns 1 through 15

         0    0.0676    0.0757    0.0757    0.1147    0.1210    0.1210    0.1583    0.1758    0.2205    0.2239    0.2239    0.2239    0.2749    0.3399
    0.0676         0    0.0080    0.0080    0.0470    0.0534    0.0534    0.0907    0.1082    0.1529    0.1563    0.1563    0.1563    0.2073    0.2722
    0.0757    0.0080         0         0    0.0390    0.0453    0.0453    0.0827    0.1001    0.1449    0.1482    0.1482    0.1482    0.1993    0.2642
    0.0757    0.0080         0         0    0.0390    0.0453    0.0453    0.0827    0.1001    0.1449    0.1482    0.1482    0.1482    0.1993    0.2642
    0.1147    0.0470    0.0390    0.0390         0    0.0063    0.0063    0.0437    0.0611    0.1059    0.1092    0.1092    0.1092    0.1603    0.2252
    0.1210    0.0534    0.0453    0.0453    0.0063         0         0    0.0373    0.0548    0.0995    0.1029    0.1029    0.1029    0.1539    0.2189
    0.1210    0.0534    0.0453    0.0453    0.0063         0         0    0.0373    0.0548    0.0995    0.1029    0.1029    0.1029    0.1539    0.2189
    0.1583    0.0907    0.0827    0.0827    0.0437    0.0373    0.0373         0    0.0175    0.0622    0.0656    0.0656    0.0656    0.1166    0.1815
    0.1758    0.1082    0.1001    0.1001    0.0611    0.0548    0.0548    0.0175         0    0.0447    0.0481    0.0481    0.0481    0.0991    0.1641
    0.2205    0.1529    0.1449    0.1449    0.1059    0.0995    0.0995    0.0622    0.0447         0    0.0034    0.0034    0.0034    0.0544    0.1193
    0.2239    0.1563    0.1482    0.1482    0.1092    0.1029    0.1029    0.0656    0.0481    0.0034         0         0         0    0.0511    0.1160
    0.2239    0.1563    0.1482    0.1482    0.1092    0.1029    0.1029    0.0656    0.0481    0.0034         0         0    0.0000    0.0511    0.1160
    0.2239    0.1563    0.1482    0.1482    0.1092    0.1029    0.1029    0.0656    0.0481    0.0034         0    0.0000         0    0.0511    0.1160
    0.2749    0.2073    0.1993    0.1993    0.1603    0.1539    0.1539    0.1166    0.0991    0.0544    0.0511    0.0511    0.0511         0    0.0649
    0.3399    0.2722    0.2642    0.2642    0.2252    0.2189    0.2189    0.1815    0.1641    0.1193    0.1160    0.1160    0.1160    0.0649         0
    0.4036    0.3360    0.3279    0.3279    0.2889    0.2826    0.2826    0.2453    0.2278    0.1831    0.1797    0.1797    0.1797    0.1286    0.0637
    0.4291    0.3615    0.3534    0.3534    0.3144    0.3081    0.3081    0.2708    0.2533    0.2086    0.2052    0.2052    0.2052    0.1542    0.0892
    0.4291    0.3615    0.3534    0.3534    0.3144    0.3081    0.3081    0.2708    0.2533    0.2086    0.2052    0.2052    0.2052    0.1542    0.0892

  Columns 16 through 18

    0.4036    0.4291    0.4291
    0.3360    0.3615    0.3615
    0.3279    0.3534    0.3534
    0.3279    0.3534    0.3534
    0.2889    0.3144    0.3144
    0.2826    0.3081    0.3081
    0.2826    0.3081    0.3081
    0.2453    0.2708    0.2708
    0.2278    0.2533    0.2533
    0.1831    0.2086    0.2086
    0.1797    0.2052    0.2052
    0.1797    0.2052    0.2052
    0.1797    0.2052    0.2052
    0.1286    0.1542    0.1542
    0.0637    0.0892    0.0892
         0    0.0255    0.0255
    0.0255         0         0
    0.0255         0         0


best_addconone =

   -6.5376


best_addcontwo =

    0.1490


best_monprox =

  Columns 1 through 15

         0    6.9486    6.6446    6.6049    6.6446    6.9017    6.6446    6.9017    6.6446    6.9486    6.7605    7.0413    6.6049    6.9486    6.6446
```

```
    6.9486         0    6.6813    6.6813    6.6049    7.0413    6.6446    7.0167    6.6049    7.0413    6.6446    7.0413    6.9486    6.6049    6.6446
    6.6446    6.6813         0    6.3551    6.6049    6.9486    6.9017    7.0255    6.3551    7.0413    6.6446    7.0413    6.9486    6.7605    6.6049
    6.6049    6.6813    6.3551         0    6.6813    7.0413    6.6446    6.9017    6.3869    6.9017    6.6446    6.9486    6.9017    6.6446    6.6049
    6.6446    6.6049    6.6049    6.6813         0    7.1219    6.6049    6.6446    6.3832    6.9486    6.6446    7.0413    7.0413    6.7605    6.3551
    6.9017    7.0413    6.9486    7.0413    7.1219         0    6.9486    6.5222    6.6049    6.6446    7.0413    6.4933    6.3551    7.0413    7.0413
    6.6446    6.6446    6.9017    6.6446    6.6049    6.9486         0    6.9486    7.0413    7.0413    6.4933    7.0413    6.9486    6.3832    6.9120
    6.9017    7.0167    7.0255    6.9017    6.6446    6.5222    6.9486         0    7.0413    6.6049    6.6813    6.3832    6.6049    7.0413    7.0413
    6.6446    6.6049    6.3551    6.3869    6.3832    6.6049    7.0413    7.0413         0    7.0069    6.6446    6.9017    6.6446    6.7605    6.3551
    6.9486    7.0413    7.0413    6.9017    6.9486    6.6446    7.0413    6.6049    7.0069         0    7.0413    6.4933    6.4933    6.9486    7.1219
    6.7605    6.6446    6.6446    6.6446    6.6446    7.0413    6.4933    6.6813    6.6446    7.0413         0    6.6813    7.0413    6.7605    6.6446
    7.0413    7.0413    7.0413    6.9486    7.0413    6.4933    7.0413    6.3832    6.9017    6.4933    6.6813         0    6.7605    7.0413    7.0413
    6.6049    6.9486    6.9486    6.9017    7.0413    6.3551    6.9486    6.6049    6.6446    6.4933    7.0413    6.7605         0    6.9017    6.6049
    6.9486    6.6049    6.7605    6.6446    6.7605    7.0413    6.3832    7.0413    6.7605    6.9486    6.7605    7.0413    6.9017         0    6.9017
    6.6446    6.6446    6.6049    6.6049    6.3551    7.0413    6.9120    7.0413    6.3551    7.1219    6.6446    7.0413    6.6049    6.9017         0
    6.6446    6.6446    6.6446    6.6049    6.6446    6.9486    6.6446    6.6446    6.6446    6.7605    6.3551    6.7605    6.9017    6.6446    6.6049
    6.6049    6.6049    6.6813    6.6049    6.9017    6.6446    6.6049    6.9486    6.7605    6.6813    6.6446    6.9486    6.6049    6.7903    6.9017
    6.9088    6.6446    6.9017    6.6446    6.9017    7.0413    6.3551    6.9017    6.7605    6.9017    6.5222    7.0069    7.0012    6.3551    6.9892

  Columns 16 through 18

    6.6446    6.6049    6.9088
    6.6446    6.6049    6.6446
    6.6446    6.6813    6.9017
    6.6049    6.6049    6.6446
    6.6446    6.9017    6.9017
    6.9486    6.6446    7.0413
    6.6446    6.6049    6.3551
    6.6446    6.9486    6.9017
    6.6446    6.7605    6.7605
    6.7605    6.6813    6.9017
    6.3551    6.6446    6.5222
    6.7605    6.9486    7.0069
    6.9017    6.6049    7.0012
    6.6446    6.7903    6.3551
    6.6049    6.9017    6.9892
         0    6.6049    6.9486
    6.6049         0    6.7460
    6.9486    6.7460         0

Elapsed time is 26000.419536 seconds.
```

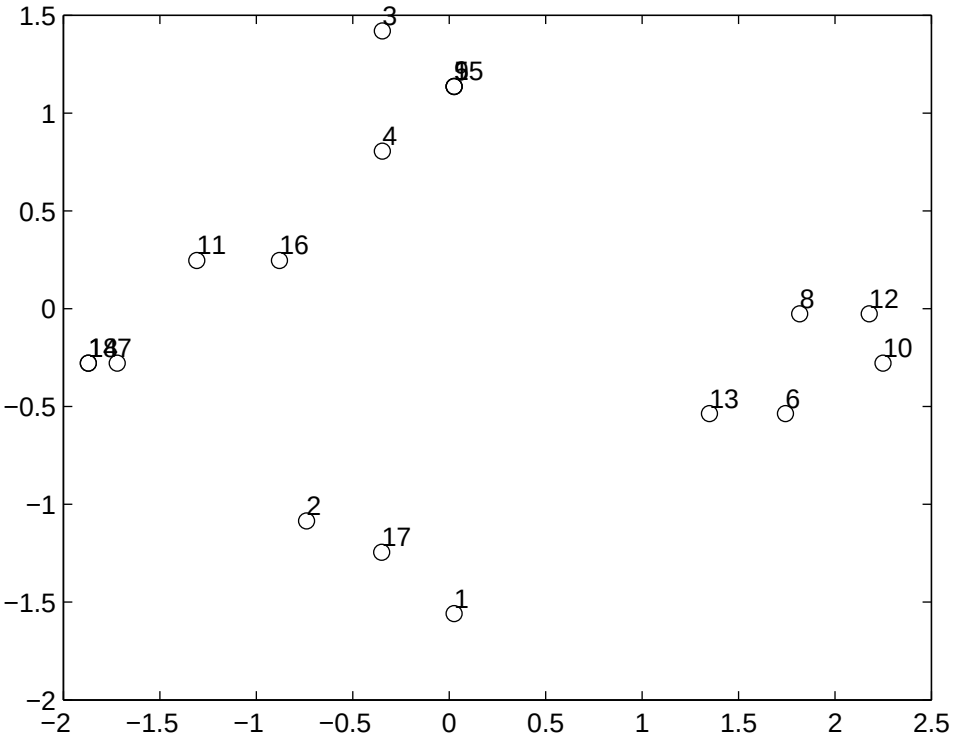


Figure 4: Best Metric City-Block Scaling for the risk_rate Data Based on One-thousand Random Starts.

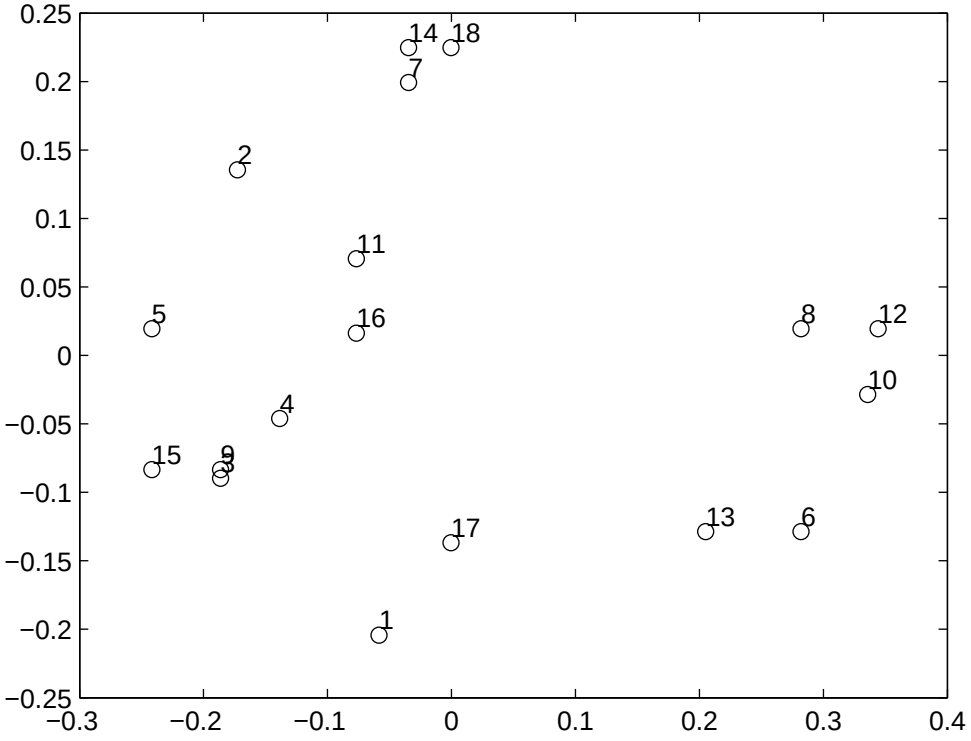


Figure 5: Best Nonmetric City-Block Scaling for the risk_rate Data Based on One-thousand Random Starts.

2.6 (Multistructural) Metric Euclidean Scaling

2.6.1 The Script for Metric Euclidean Scaling Using the MATLAB M-file, mdscal.m

```
load risk_rate.dat

tic;

opts = statset('Maxiter',1000);

best_vaf = 0.0;

store_vaf = zeros(100,1);

for k = 1:100

[coords,stress] = mdscale(risk_rate,2,'Criterion','metricstress', 'Start',...
    'random','Replicates',1,'Options',opts);

n = size(coords,1);

distance_matrix = zeros(n,n);

for i = 1:n
    for j = 1:n

        distance_matrix(i,j) = sqrt(((coords(i,1) - coords(j,1))^2) + ...
            ((coords(i,2) - coords(j,2))^2));
    end
```

```
end

ave_diff = (sum(sum(risk_rate - distance_matrix)))/(n*(n-1));

ave_proximity = (sum(sum(risk_rate)))/(n*(n-1));

risk_rate_vec = squareform(risk_rate);

distance_vec = squareform(distance_matrix);

r = corrcoef(risk_rate_vec',distance_vec');

vaf = r(1,2)^2;

store_vaf(k) = vaf;

if(vaf > best_vaf)

    best_vaf = vaf;
    best_coords = coords;

end

end

store_vaf;

sorted_vafs = sort(store_vaf');

sorted_vafs

best_vaf

best_coords

axis equal

plot(best_coords(:,1),best_coords(:,2),'ko')

hold on

for i = 1:18

    objectlabels{i,1} = int2str(i);

end

text(best_coords(:,1),best_coords(:,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

toc;
```

2.6.2 The Script Results for Metric Euclidean Scaling Using the MATLAB M-file, mdscal.m

```
>> script_metric_mds

sorted_vafs =

  Columns 1 through 15

    0.2646    0.4169    0.4264    0.4558    0.4597    0.4633    0.4659    0.4691    0.4722    0.4739    0.4758    0.4775    0.4777    0.4785    0.4792

  Columns 16 through 30

    0.4795    0.4812    0.4873    0.4881    0.4886    0.4929    0.4941    0.4949    0.4962    0.4971    0.4987    0.4999    0.5000    0.5001    0.5002

  Columns 31 through 45

    0.5013    0.5021    0.5031    0.5042    0.5042    0.5042    0.5044    0.5056    0.5066    0.5082    0.5083    0.5084    0.5085    0.5085    0.5089

  Columns 46 through 60

    0.5101    0.5101    0.5104    0.5106    0.5107    0.5116    0.5117    0.5119    0.5119    0.5124    0.5127    0.5128    0.5130    0.5131    0.5133

  Columns 61 through 75

    0.5138    0.5144    0.5159    0.5173    0.5176    0.5186    0.5186    0.5192    0.5196    0.5202    0.5206    0.5217    0.5226    0.5229    0.5232

  Columns 76 through 90

    0.5232    0.5249    0.5250    0.5253    0.5256    0.5270    0.5287    0.5292    0.5309    0.5323    0.5360    0.5370    0.5372    0.5386    0.5398
```

```
  Columns 91 through 100

    0.5398    0.5457    0.5486    0.5486    0.5486    0.5492    0.5492    0.5492    0.5492    0.5492


best_vaf =

    0.5492


best_coords =

    1.6742    4.2039
   -4.8684   -0.3476
   -4.0371    2.5216
   -2.4668    1.0908
   -2.2080    4.2680
    5.0997    1.7239
   -2.2123   -3.4356
    4.9497   -0.1492
   -1.6114    3.2053
    4.2708   -3.3949
   -0.3570   -4.1987
    5.1830   -1.8821
    3.6586    2.3156
   -4.1534   -2.9553
   -1.0932    4.4162
    0.8473   -3.1382
    0.4704   -0.2157
   -3.1462   -4.0280

Elapsed time is 25.522540 seconds.
```

2.7 (Multistructual) Nonmetric Euclidean Scaling

2.7.1 The Script for Nonmetric Euclidean Scaling Using the MATLAB M-file, mdscal.m

```
load risk_rate.dat

tic;

opts = statset('Maxiter',1000);

best_vaf = 0.0;

store_vaf = zeros(100,1);

store_stress = zeros(100,1);

for k = 1:100

[coords,stress,disparities] = mdscale(risk_rate,2,'Criterion','stress', 'Start',...
    'random','Replicates',1,'Options',opts);

n = size(coords,1);

distance_matrix = zeros(n,n);

for i = 1:n
    for j = 1:n

        distance_matrix(i,j) = sqrt(((coords(i,1) - coords(j,1))^2) + ...
            ((coords(i,2) - coords(j,2))^2));
    end
end

ave_diff = (sum(sum(disparities - distance_matrix)))/(n*(n-1));

ave_proximity = (sum(sum(disparities)))/(n*(n-1));

disparities_vec = squareform(disparities);

distance_vec = squareform(distance_matrix);

r = corrcoef(disparities_vec',distance_vec');

vaf = r(1,2)^2;

store_vaf(k) = vaf;
```

```
store_stress(k) = stress;

if(vaf > best_vaf)

    best_vaf = vaf;
    best_coords = coords;
    best_disparities = disparities;

end

end

store_vaf;

sorted_vafs = sort(store_vaf');

sorted_stress = sort(store_stress');

sorted_vafs sorted_stress;

best_vaf

best_coords

best_disparities

lossvalues = [store_vaf,store_stress];

corrcoef(lossvalues);

axis equal

plot(best_coords(:,1),best_coords(:,2),'ko')

hold on

for i = 1:18

    objectlabels{i,1} = int2str(i);

end

text(best_coords(:,1),best_coords(:,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

toc;
```

2.7.2 The Script Results for Nonmetric Euclidean Scaling Using the MATLAB M-file, mdscal.m

```
>> script_nonmetric_mds

sorted_vafs =

  Columns 1 through 15

    0.0364    0.3629    0.3731    0.3928    0.4326    0.5293    0.5989    0.6278    0.6285    0.6321    0.6369    0.6384    0.6397    0.6401    0.6406

  Columns 16 through 30

    0.6444    0.6451    0.6464    0.6498    0.6501    0.6502    0.6510    0.6609    0.6669    0.6695    0.6845    0.6851    0.6888    0.6891    0.6924

  Columns 31 through 45

    0.6925    0.6925    0.6926    0.6978    0.6994    0.6994    0.6996    0.7006    0.7006    0.7102    0.7134    0.7135    0.7139    0.7139    0.7140

  Columns 46 through 60

    0.7199    0.7199    0.7239    0.7239    0.7239    0.7239    0.7240    0.7241    0.7276    0.7277    0.7277    0.7284    0.7284    0.7285    0.7286

  Columns 61 through 75

    0.7293    0.7300    0.7300    0.7300    0.7300    0.7300    0.7300    0.7304    0.7313    0.7313    0.7338    0.7338    0.7347    0.7347    0.7348

  Columns 76 through 90

    0.7352    0.7359    0.7360    0.7360    0.7362    0.7396    0.7396    0.7396    0.7396    0.7396    0.7396    0.7411    0.7411    0.7411    0.7411

  Columns 91 through 100

    0.7411    0.7411    0.7411    0.7412    0.7412    0.7412    0.7412    0.7412    0.7412    0.7412
```

```
best_vaf =

    0.7412


best_coords =

    0.5999    3.1164
   -3.6062   -0.8089
   -3.1989    1.8110
   -2.7214    0.6281
   -1.9069    2.7765
    4.6957    0.8933
   -1.8474   -3.0223
    3.9270    0.0940
   -1.3809    2.5193
    4.2506   -2.4861
   -0.9232   -0.9511
    4.6869   -1.0454
    3.5638    2.0335
   -2.8797   -3.0564
   -1.6568    3.7219
   -0.2457   -0.0397
    0.7224   -2.4338
   -2.0792   -3.7503


best_disparities =

  Columns 1 through 15

         0    6.5558    3.7203    3.6876    3.6876    6.3731    3.7203    4.7875    3.6876    6.5558    4.7875    7.2167    3.6876    6.5558    3.7203
    6.5558         0    3.7203    3.7203    3.6876    7.2167    3.7203    7.0596    3.6876    7.0596    3.7203    7.0596    6.5558    3.6876    3.7203
    3.7203    3.7203         0    1.1559    3.6876    6.5558    6.3731    7.0596    1.4512    7.6555    3.7203    7.0596    6.5558    4.7875    3.6876
    3.6876    3.7203    1.1559         0    3.7203    7.4219    3.6876    6.3731    2.1750    6.3731    3.7203    6.5558    6.3731    3.6879    3.6876
    3.6876    3.6876    3.6876    3.7203         0    7.6555    3.6876    3.7203    1.1559    6.5558    3.7203    7.2167    7.0596    4.7875    0.9779
    6.3731    7.2167    6.5558    7.4219    7.6555         0    6.5558    2.5536    3.6876    3.7203    7.0596    2.1750    1.1559    7.2167    7.0596
    3.7203    3.7203    6.3731    3.6876    3.6876    6.5558         0    6.5558    7.0596    7.0596    2.1750    7.2167    6.5558    1.4512    6.5558
    4.7875    7.0596    7.0596    6.3731    3.7203    2.5536    6.5558         0    7.0596    3.6876    3.7203    1.4512    3.2984    7.6555    7.0596
    3.6876    3.6876    1.4512    2.1750    1.1559    3.6876    7.0596    7.0596         0    7.0596    3.7203    6.3731    3.7203    4.7875    1.0482
    6.5558    7.0596    7.6555    6.3731    6.5558    3.7203    7.0596    3.6876    7.0596         0    7.0596    2.5536    2.5536    6.5558    8.5695
    4.7875    3.7203    3.7203    3.7203    3.7203    7.0596    2.1750    3.7203    3.7203    7.0596         0    3.7203    7.2167    4.7875    3.7203
    7.2167    7.0596    7.0596    6.5558    7.2167    2.1750    7.2167    1.4512    6.3731    2.5536    3.7203         0    4.7875    7.2167    7.0596
    3.6876    6.5558    6.5558    6.3731    7.0596    1.1559    6.5558    3.2984    3.7203    2.5536    7.2167    4.7875         0    6.3731    3.6876
    6.5558    3.6876    4.7875    3.6879    4.7875    7.2167    1.4512    7.6555    4.7875    6.5558    4.7875    7.2167    6.3731         0    6.3731
    3.7203    3.7203    3.6876    3.6876    0.9779    7.0596    6.5558    7.0596    1.0482    8.5695    3.7203    7.0596    3.6876    6.3731         0
    3.7203    3.7203    3.7203    3.6876    3.7203    6.5558    3.7203    3.7203    3.7203    4.7875    1.0482    4.7875    6.3731    3.7203    3.6876
    3.6876    3.2984    4.7875    3.6876    6.3731    3.7203    3.6876    6.5558    5.3811    3.7203    3.7203    6.5558    3.6876    4.7875    6.5558
    6.5558    3.6876    5.6728    3.6876    6.3731    7.0596    1.0482    6.3731    4.7875    6.3731    2.5536    7.0596    7.0596    1.0482    7.0596

  Columns 16 through 18

    3.7203    3.6876    6.5558
    3.7203    3.2984    3.6876
    3.7203    4.7875    5.6728
    3.6876    3.6876    3.6876
    3.7203    6.3731    6.3731
    6.5558    3.7203    7.0596
    3.7203    3.6876    1.0482
    3.7203    6.5558    6.3731
    3.7203    5.3811    4.7875
    4.7875    3.7203    6.3731
    1.0482    3.7203    2.5536
    4.7875    6.5558    7.0596
    6.3731    3.6876    7.0596
    3.7203    4.7875    1.0482
    3.6876    6.5558    7.0596
         0    3.6876    6.5558
    3.6876         0    3.7203
    6.5558    3.7203         0

Elapsed time is 19.854091 seconds.
```

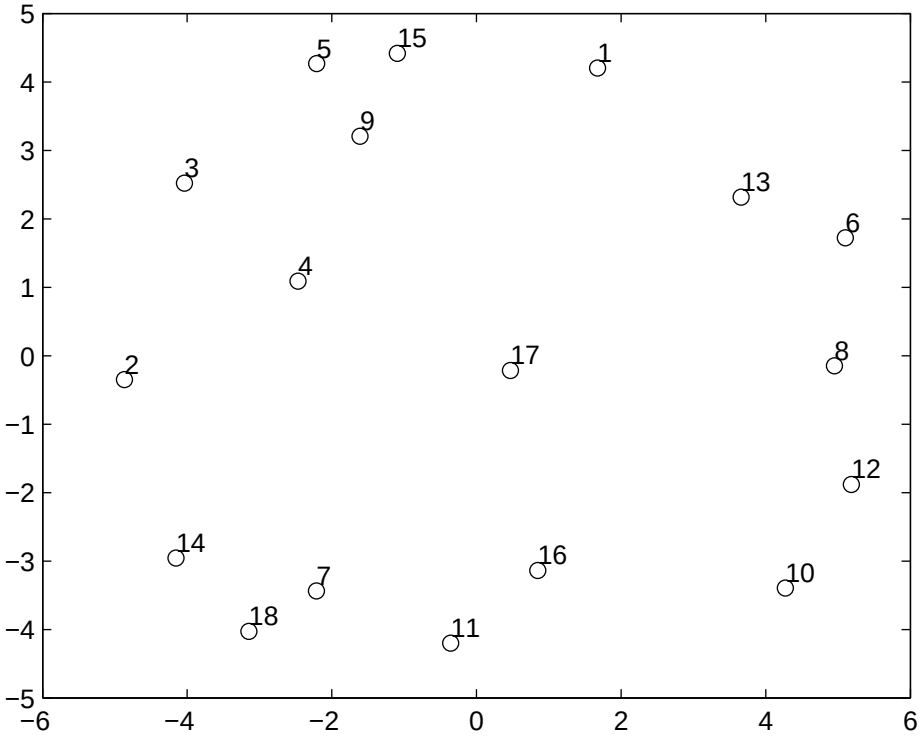


Figure 6: Best Metric Euclidean Scaling for the risk_rate Data Based on One-hundred Random Starts.

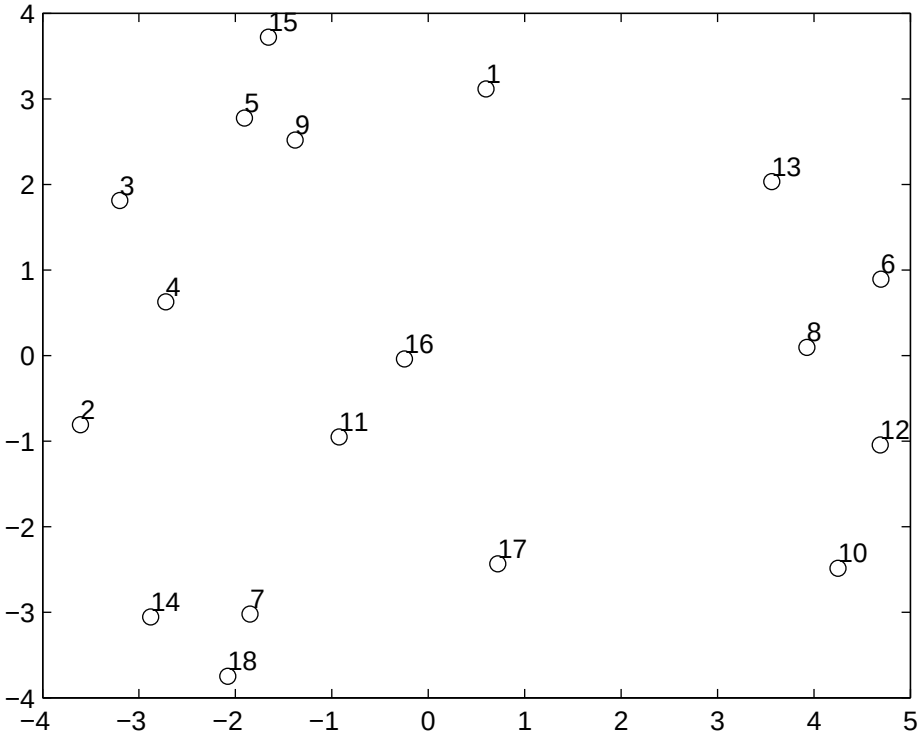


Figure 7: Best Nonmetric Euclidean Scaling for the risk_rate Data Based on One-hundred Random Starts.

3 Other Multistructural Analysis Routines

3.1 (Multistructural) Circular Structures

Just as the linear unidimensional scaling task from the Unidimensional Scaling Toolbox was extended to two dimensions (through `biscalqa.m`), so can the circular scaling task (through `bicirac.m` and the Fortran-enhanced routine, `ms_bicirac.m`). As an illustration of the results obtainable using the Morse code digit data, the MATLAB output below gives the best (according to a VAF of 92.18%) two-CUS (Circular Unidimensional Scaling) representation obtained from 1000 random starting permutations for each of the circular components. The two CUS structures (given in Figures 8 and 9 and generated with `circularplot.m`) have rather clear substantive interpretations: as with our example using `unicirac.m` in the Unidimensional Scaling Toolbox,

the first shows the regular replacement of dots by dashes moving around the closed continuum; the second provides a perfect ordering around the closed continuum according to ratios of dots to dashes or of dashes to dots and where adjacent pairs of stimuli have dashes and dots exchanged one-for-one; i.e., for the adjacent stimuli pairs moving clockwise, we have:

0:5 for {− − − − −; • • • • ••} (0,5); 1:4 for {• − − − −; − • • • •} (1,6); 2:3 for {• • − − −; − − • • •} (2,7); 3:2 for {• • • − −; − − − • •} (3,8); and 1:4 for {− − − − •; • • • • −}(9,4).

The two additive constants have values of -.7007 and .3526, respectively. (As mentioned, the output given below represents the best two-CUS structures obtained for 1000 random starting permutations, but as might be expected given the earlier computational results in the Unidimensional Scaling Toolbox, the same type of local optima were observed here as found in the fitting of a single CUS structure, i.e., several local optima were generated from small differences in the estimation of inflection points and the adjacent object spacings but with the identical object orderings around the closed continua)

3.1.1 Basic Script Results for bicirac.m

```
>> script_bicirac

sorted_vafs =

  Columns 1 through 15

    0.8153    0.8190    0.8379    0.8384    0.8392    0.8488    0.8576    0.8577    0.8593    0.8606    0.8606    0.8606    0.8623    0.8637    0.8742

  Columns 16 through 30

    0.8758    0.8761    0.8795    0.8802    0.8802    0.8809    0.8809    0.8824    0.8824    0.8832    0.8835    0.8835    0.8858    0.8867    0.8889

  Columns 31 through 45

    0.8893    0.8893    0.8900    0.8901    0.8902    0.8904    0.8907    0.8911    0.8917    0.8920    0.8932    0.8932    0.8946    0.8946    0.8946

  Columns 46 through 60

    0.8959    0.8977    0.8989    0.9005    0.9015    0.9022    0.9024    0.9075    0.9079    0.9079    0.9079    0.9080    0.9095    0.9141    0.9148

  Columns 61 through 75

    0.9156    0.9187    0.9190    0.9190    0.9191    0.9192    0.9192    0.9192    0.9192    0.9192    0.9192    0.9196    0.9201    0.9201    0.9204

  Columns 76 through 90

    0.9204    0.9204    0.9204    0.9206    0.9206    0.9206    0.9206    0.9206    0.9206    0.9206    0.9206    0.9206    0.9211    0.9211    0.9213

  Columns 91 through 100

    0.9214    0.9218    0.9218    0.9218    0.9218    0.9218    0.9218    0.9218    0.9218    0.9218

best_vaf =

    0.9218
```

```
best_outpermone =

     6     5     4     3     2     1    10     9     8     7


best_outpermtwo =

    10     9     4     8     3     7     2     6     1     5


best_targone =

         0    0.4249    0.5300    0.9195    1.3215    1.1359    0.9187    0.8897    0.6528    0.3078
    0.4249         0    0.1051    0.4946    0.8966    1.2017    1.3436    1.3146    1.0777    0.7327
    0.5300    0.1051         0    0.3895    0.7915    1.0966    1.3139    1.3429    1.1828    0.8378
    0.9195    0.4946    0.3895         0    0.4020    0.7071    0.9243    0.9533    1.1902    1.2274
    1.3215    0.8966    0.7915    0.4020         0    0.3051    0.5224    0.5514    0.7882    1.1332
    1.1359    1.2017    1.0966    0.7071    0.3051         0    0.2173    0.2463    0.4831    0.8281
    0.9187    1.3436    1.3139    0.9243    0.5224    0.2173         0    0.0290    0.2659    0.6109
    0.8897    1.3146    1.3429    0.9533    0.5514    0.2463    0.0290         0    0.2369    0.5819
    0.6528    1.0777    1.1828    1.1902    0.7882    0.4831    0.2659    0.2369         0    0.3450
    0.3078    0.7327    0.8378    1.2274    1.1332    0.8281    0.6109    0.5819    0.3450         0


best_targtwo =

         0    0.0977    0.2286    0.4321    0.5652    0.6139    0.6173    0.5611    0.2915    0.0785
    0.0977         0    0.1310    0.3345    0.4675    0.5162    0.5775    0.6336    0.3891    0.1761
    0.2286    0.1310         0    0.2035    0.3365    0.3853    0.4466    0.5027    0.5201    0.3071
    0.4321    0.3345    0.2035         0    0.1330    0.1818    0.2430    0.2992    0.5688    0.5106
    0.5652    0.4675    0.3365    0.1330         0    0.0487    0.1100    0.1661    0.4358    0.6436
    0.6139    0.5162    0.3853    0.1818    0.0487         0    0.0613    0.1174    0.3871    0.6000
    0.6173    0.5775    0.4466    0.2430    0.1100    0.0613         0    0.0561    0.3258    0.5388
    0.5611    0.6336    0.5027    0.2992    0.1661    0.1174    0.0561         0    0.2697    0.4827
    0.2915    0.3891    0.5201    0.5688    0.4358    0.3871    0.3258    0.2697         0    0.2130
    0.0785    0.1761    0.3071    0.5106    0.6436    0.6000    0.5388    0.4827    0.2130         0


best_find =

         0    0.9797    1.4916    1.9654    1.7634    1.7543    1.5639    1.4007    0.9841    0.8574
    0.9797         0    0.8607    1.5868    1.7841    1.7264    1.5432    1.3800    1.4776    1.4884
    1.4916    0.8607         0    1.0748    1.4870    1.4344    1.6248    1.6720    1.7696    1.8382
    1.9654    1.5868    1.0748         0    0.7609    1.3814    1.5719    1.7351    1.8225    1.8912
    1.7634    1.7841    1.4870    0.7609         0    1.2563    1.6815    1.9371    1.8395    1.7708
    1.7543    1.7264    1.4344    1.3814    1.2563         0    0.7739    1.3007    1.8720    1.8285
    1.5639    1.5432    1.6248    1.5719    1.6815    0.7739         0    0.8755    1.4468    1.5735
    1.4007    1.3800    1.6720    1.7351    1.9371    1.3007    0.8755         0    0.9201    1.0467
    0.9841    1.4776    1.7696    1.8225    1.8395    1.8720    1.4468    0.9201         0    0.4754
    0.8574    1.4884    1.8382    1.8912    1.7708    1.8285    1.5735    1.0467    0.4754         0


best_addconone =

   -0.7000


best_addcontwo =

    0.3512

Elapsed time is 2291.309111 seconds.
```

3.1.2 The Enhanced Script for ms_bicirac.m

```
load morse_digits.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(1000,1);

for i = 1:1000

    [find,vaf,targone,targtwo,outpermone,outpermtwo,addconone,addcontwo] = ...
        ms_bicirac(morse_digits,randperm(10),3);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
```

```
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_find = find;
        best_targone = targone;
        best_targtwo = targtwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs;

        best_vaf
        best_outpermone
        best_outpermtwo
        best_targone
        best_targtwo
        best_find

        best_addconone
        best_addcontwo


toc;

figure(1)

[circum_one,radius_one,coord_one,degrees_one,cumdegrees_one] ...
    = circularplot(best_targone,best_outpermone)

figure(2)

[circum_two,radius_two,coord_two,degrees_two,cumdegrees_two] ...
    = circularplot(best_targtwo,best_outpermtwo)
```

3.1.3 Enhanced Script Results for ms_bicirac.m

```
>> ms_script_bicirac

best_vaf =

    0.9218


best_outpermone =

     9    10     1     2     3     4     5     6     7     8


best_outpermtwo =

     6     1     5    10     9     4     8     3     7     2


best_targone =

         0    0.0354    0.2464    0.5505    0.9516    1.3433    1.3087    0.8833    0.5762    0.2324
    0.0354         0    0.2110    0.5151    0.9161    1.3079    1.3442    0.9187    0.6117    0.2678
    0.2464    0.2110         0    0.3041    0.7051    1.0969    1.2058    1.1297    0.8227    0.4788
    0.5505    0.5151    0.3041         0    0.4011    0.7928    0.9018    1.3272    1.1268    0.7829
    0.9516    0.9161    0.7051    0.4011         0    0.3917    0.5007    0.9262    1.2332    1.1839
    1.3433    1.3079    1.0969    0.7928    0.3917         0    0.1090    0.5344    0.8415    1.1853
    1.3087    1.3442    1.2058    0.9018    0.5007    0.1090         0    0.4255    0.7325    1.0764
    0.8833    0.9187    1.1297    1.3272    0.9262    0.5344    0.4255         0    0.3070    0.6509
    0.5762    0.6117    0.8227    1.1268    1.2332    0.8415    0.7325    0.3070         0    0.3439
    0.2324    0.2678    0.4788    0.7829    1.1839    1.1853    1.0764    0.6509    0.3439         0


best_targtwo =

         0    0.2738    0.4798    0.5659    0.6364    0.5018    0.3010    0.1587    0.1184    0.0534
    0.2738         0    0.2060    0.2921    0.3881    0.5226    0.5748    0.4325    0.3922    0.3272
    0.4798    0.2060         0    0.0861    0.1820    0.3166    0.5174    0.6385    0.5982    0.5332
    0.5659    0.2921    0.0861         0    0.0959    0.2305    0.4313    0.5736    0.6139    0.6193
    0.6364    0.3881    0.1820    0.0959         0    0.1346    0.3354    0.4776    0.5180    0.5829
    0.5018    0.5226    0.3166    0.2305    0.1346         0    0.2008    0.3431    0.3834    0.4484
    0.3010    0.5748    0.5174    0.4313    0.3354    0.2008         0    0.1423    0.1826    0.2476
    0.1587    0.4325    0.6385    0.5736    0.4776    0.3431    0.1423         0    0.0403    0.1053
    0.1184    0.3922    0.5982    0.6139    0.5180    0.3834    0.1826    0.0403         0    0.0650
```

```
    0.0534    0.3272    0.5332    0.6193    0.5829    0.4484    0.2476    0.1053    0.0650         0


best_find =

         0    0.9793    1.4857    1.9676    1.7600    1.7516    1.5630    1.4017    0.9826    0.8512
    0.9793         0    0.8544    1.5892    1.7831    1.7287    1.5398    1.3785    1.4815    1.4825
    1.4857    0.8544         0    1.0829    1.4873    1.4330    1.6216    1.6743    1.7773    1.8378
    1.9676    1.5892    1.0829         0    0.7736    1.3843    1.5729    1.7342    1.8260    1.8865
    1.7600    1.7831    1.4873    0.7736         0    1.2533    1.6788    1.9418    1.8388    1.7783
    1.7516    1.7287    1.4330    1.3843    1.2533         0    0.7735    1.3000    1.8677    1.8327
    1.5630    1.5398    1.6216    1.5729    1.6788    0.7735         0    0.8745    1.4423    1.5737
    1.4017    1.3785    1.6743    1.7342    1.9418    1.3000    0.8745         0    0.9158    1.0472
    0.9826    1.4815    1.7773    1.8260    1.8388    1.8677    1.4423    0.9158         0    0.4795
    0.8512    1.4825    1.8378    1.8865    1.7783    1.8327    1.5737    1.0472    0.4795         0


best_addconone =

   -0.7007


best_addcontwo =

    0.3526

Elapsed time is 2188.250924 seconds.

circum_one =

    2.7610


radius_one =

    0.4394


coord_one =

         0    0.4394
    0.0354    0.4380
    0.2337    0.3721
    0.4174    0.1374
    0.3640   -0.2462
    0.0372   -0.4379
   -0.0715   -0.4336
   -0.3977   -0.1869
   -0.4247    0.1127
   -0.2217    0.3794


degrees_one =

    0.0807
    0.4802
    0.6919
    0.9127
    0.8915
    0.2480
    0.9682
    0.6987
    0.7825
    0.5288


cumdegrees_one =

    0.0807
    0.5608
    1.2528
    2.1654
    3.0569
    3.3049
    4.2731
    4.9718
    5.7544
    6.2832


circum_two =

    1.2982


radius_two =
```

```
    0.2066


coord_two =

         0    0.2066
    0.2004    0.0503
    0.1510   -0.1411
    0.0810   -0.1901
   -0.0127   -0.2062
   -0.1351   -0.1563
   -0.2053    0.0235
   -0.1436    0.1486
   -0.1120    0.1736
   -0.0528    0.1997


degrees_two =

    1.3250
    0.9972
    0.4166
    0.4644
    0.6514
    0.9718
    0.6885
    0.1951
    0.3145
    0.2586


cumdegrees_two =

    1.3250
    2.3222
    2.7388
    3.2032
    3.8545
    4.8264
    5.5149
    5.7100
    6.0246
    6.2832
```

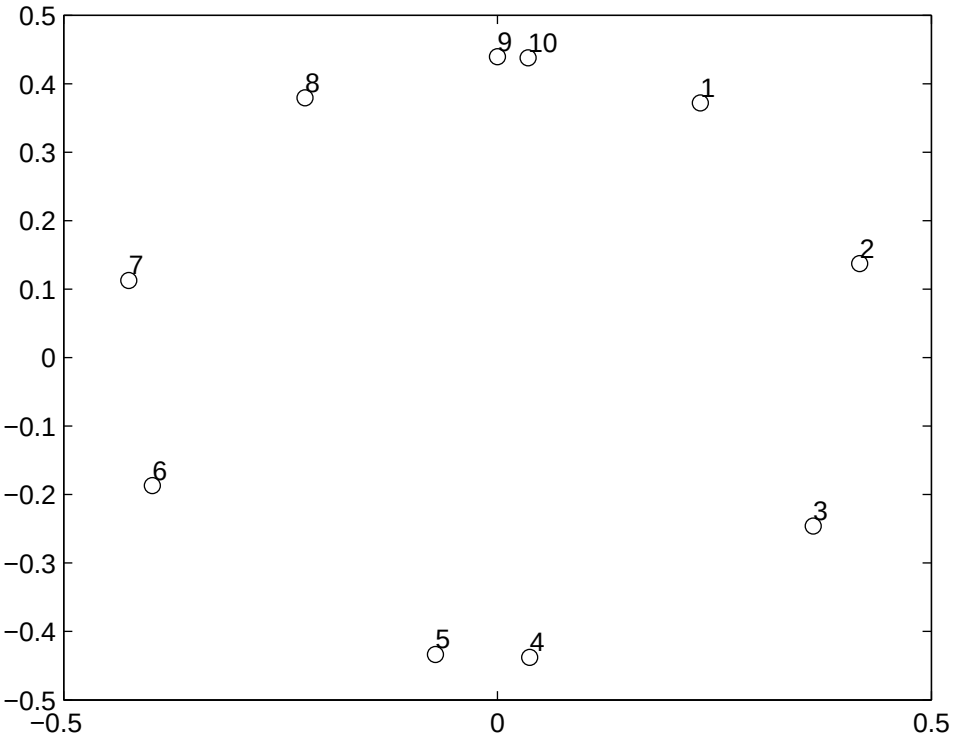


Figure 8: Best First Circular Structure for the Morse Code Digit Data Based on One-thousand Random Starts.

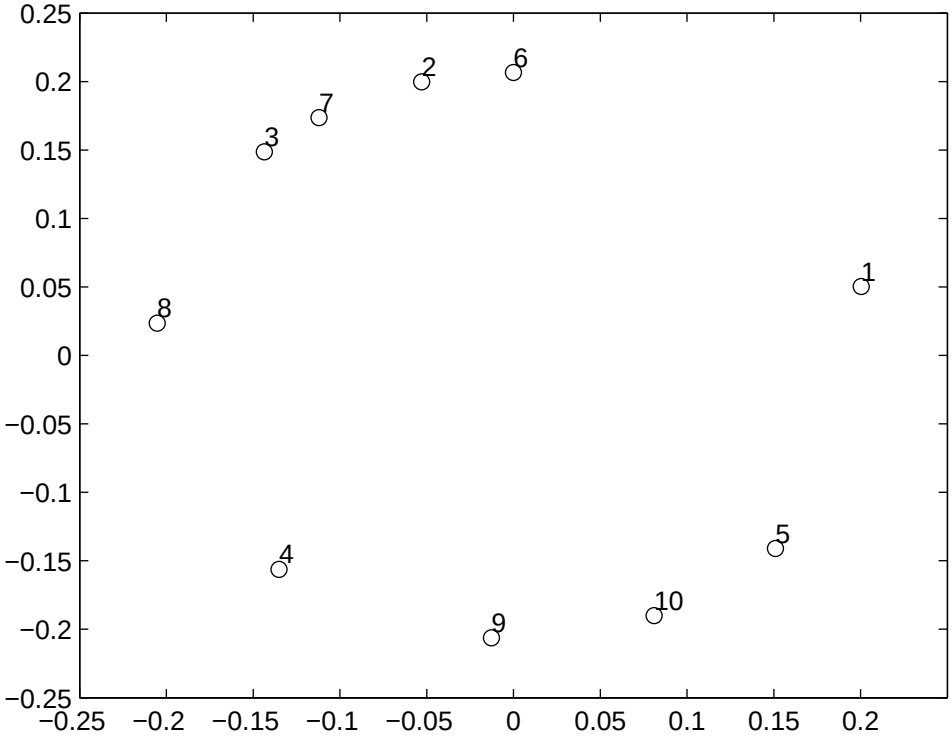


Figure 9: Best Second Circular Structure for the Morse Code Digit Data Based on One-thousand Random Starts.

3.2 (Multistructural) Two-Mode Metric City-Block Scaling

In illustrating the two-mode metric and nonmetric two-dimensional city-block scaling routine (`biscaltmac.m` and `bimonscaltmac.m`), we use the 10×15 data matrix, `eve_black_one.dat`, augmented by a sixteenth "average" concept defined by a column vector containing all fours. There are a number of points to be made about these routines and the resultant output:

a) `biscaltmac.m` and `bimonscaltmac.m` are *very* computationally time consuming to run, and only the enhanced versions with one-hundred random starts of each are even illustrated;

b) both `biscaltmac.m` and `bimonscaltmac.m` include two controls for ensuring convergence. One is `addcontol` which regulates when the change in the additive constant is deemed completed. If set too small, the routines

may labor for a very long time trying to reach it. In the two enhanced scripts below, `addcontol` was set at .01. The other convergence control is the maximum number of iterations (`maxiteration`) allowed internally to obtain the coordinates with the accuracy desired (and before reiterating on the estimation of the additive constant). We have set `maxiteration` at 1000 in our two scripts. The reader is welcome to experiment with setting these two values to evaluate the consequences in the identified structures. Generally, the object permutations stay the same; only the additive constant, coordinates, and VAFs vary, depending on how tight or loose `addcontol` and `maxiteration` are set.

c) the spatial structures plotted with `biplottm.m` in Figures 10 and 11 (metric and nonmetric, respectively) are very similar. This seems to be true generally — a nonmetric solution rarely (if ever) gives something substantively different from a metric alternative.

d) interpreting the spatial structures can be difficult (to say the least), and we quote Osgood and Luria below on the configuration they found for Eve Black. If one looks closely, and if one reads carefully, Figures 10 and 11 are very compatible:

"The most general characterization here would be that *Eve Black has achieved a violent kind of adjustment in which she perceives herself as literally perfect, but, to accomplish this break, her way of perceiving "the world" becomes completely disoriented from the norm.* The only exceptions to this dictum are MY DOCTOR and PEACE OF MIND, which maintain their *good* and *strong* characteristics, ... But if Eve Black perceives herself as being *good*, then she also has to accept HATRED and FRAUD as positive values, since (we assume) she has strong hatreds and is socially fraudulent. So we find a tight, but very un-normal, favorable cluster of ME, MY DOCTOR, PEACE OF MIND, HATRED, and FRAUD. What are positive values for most people — CHILD, MY SPOUSE, MY JOB, LOVE, and SEX — are completely rejected as *bad* and *passive*, and all of these except CHILD are also *weak* (this may be because CHILD was weak in Eve White and much of the change here is a simple "flip-flop" of meanings). Note that it is MOTHER in this personality that becomes relatively meaningless; FATHER, on the other hand, stays

good but shifts completely from *strong* (in Eve White) to *weak* — possible implications of these familial identifications will be considered later. Note also that in this personality LOVE and SEX are closely identified, both as *bad*, *weak*, *passive* things."

3.2.1 The Enhanced Script for ms_biscaltmac.m

```
load eve_black_one.dat


best_vaf = 0.0;

store_vaf = zeros(100,1);

core = [4,4,4,4,4,4,4,4,4,4];

proxtm = [eve_black_one,core'];

maxiteration = 1000;

addcontol = .01;

for i = 1:100

    [find,vaf,targone,targtwo,outpermone,outpermtwo,rowpermone,colpermone, ...
        rowpermtwo,colpermtwo,addconone,addcontwo,coordone,coordtwo,axes] = ...
        ms_biscaltmac(proxtm,randperm(26),randperm(26),3,1,addcontol,maxiteration);


    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_find = find;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_targone = targone;
        best_targtwo = targtwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;
        best_rowpermone = rowpermone;
        best_colpermone = colpermone;
        best_rowpermtwo = rowpermtwo;
        best_colpermtwo = colpermtwo;
        best_axes = axes;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs

        best_vaf
        best_find
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo
        best_targone
        best_targtwo
        best_addconone
        best_addcontwo
        best_rowpermone
        best_colpermone
        best_rowpermtwo
        best_colpermtwo
        best_axes
```

3.2.2 Enhanced Script Results for ms_biscaltmac.m

```
ms_script_biscaltmac

sorted_vafs =

  Columns 1 through 15

    0.7367    0.7469    0.7497    0.7518    0.7527    0.7545    0.7564    0.7564    0.7571    0.7571    0.7600    0.7600    0.7614    0.7622    0.7635

  Columns 16 through 30

    0.7656    0.7661    0.7663    0.7664    0.7670    0.7670    0.7670    0.7670    0.7671    0.7676    0.7680    0.7690    0.7727    0.7754    0.7754

  Columns 31 through 45

    0.7754    0.7754    0.7757    0.7762    0.7762    0.7777    0.7792    0.7792    0.7796    0.7801    0.7801    0.7801    0.7801    0.7801    0.7802

  Columns 46 through 60

    0.7808    0.7809    0.7809    0.7809    0.7809    0.7810    0.7816    0.7817    0.7817    0.7817    0.7817    0.7817    0.7817    0.7819    0.7821

  Columns 61 through 75

    0.7821    0.7826    0.7828    0.7830    0.7830    0.7833    0.7833    0.7833    0.7834    0.7834    0.7834    0.7834    0.7838    0.7838    0.7838

  Columns 76 through 90

    0.7838    0.7838    0.7838    0.7838    0.7845    0.7845    0.7846    0.7846    0.7846    0.7846    0.7855    0.7896    0.7896    0.7896    0.7896

  Columns 91 through 100

    0.7896    0.7896    0.7896    0.7896    0.7896    0.7896    0.7896    0.7896    0.7897    0.7897


best_vaf =

    0.7897


best_find =

  Columns 1 through 15

    3.4255    0.5227    6.3560    6.8903   -0.1528    1.0370    2.8512    6.4138    6.0640    2.3559    3.4496    5.9718    5.7659    2.9199    1.3095
    6.2535    8.5900    1.4056    0.8712    7.9144    6.7246    4.9104    1.3478    1.6976    7.1975    4.3120    1.7898    2.5381    4.8417    6.4521
    1.0842    3.9038    6.5749    6.0387    3.2283    2.1340    3.9482    6.5171    5.7985    3.6919    4.6418    6.1907    2.3845    2.0683    1.7700
    0.9573    5.2112    6.7018    6.1655    4.5356    3.3458    4.0750    6.6440    5.9254    3.8187    4.7687    6.3176    2.4524    2.1951    3.0733
    7.4794    5.8209    1.0578    1.5921    5.1453    3.9555    2.1413    1.1156    0.7658    4.4284    1.5429    0.6736    3.7640    4.0213    4.3192
    3.0072    0.9410    7.0082    6.4720    1.6165    2.5674    4.3815    6.9504    6.2318    4.1252    5.0751    6.6241    5.3476    2.5016    2.2037
    1.0568    3.9361    6.6023    6.0661    3.2606    2.1613    3.9756    6.5445    5.8259    3.7193    4.6692    6.2181    2.3527    2.0956    1.7980
    6.3771    8.4645    1.2820    0.7458    7.7889    6.5991    4.7850    1.2242    1.5721    7.0720    4.1865    1.6643    2.6617    4.7162    6.3266
    8.9996    7.1021    1.0367    1.5710    6.4265    5.4757    3.6615    1.0945    1.8112    4.4495    2.9679    1.4190    5.2842    5.5415    5.8394
    6.2400    7.9476    1.4191    0.8829    7.2721    6.0823    4.2681    1.3613    1.0552    6.5552    3.6696    1.1475    2.5246    4.1994    5.8098

  Column 16

    3.6284
    4.1332
    3.5123
    3.6391
    2.5773
    3.9456
    3.5397
    4.0077
    4.0974
    3.4909


best_outpermone =

     2    13    18     8     9    14    10    22    19     5    21    23    26    17    24     4    20    16    25     7     3    11     1    15     6    12


best_outpermtwo =

     4    11    23     7     3     2     8    10    14    24     6    19    26    18    22    13    25    16     5    17    21     1    12    15     9    20


best_coordone =

         0
         0
         0
    0.0009
```

```
    0.0009
    0.0009
    0.3279
    0.3842
    0.5342
    0.7716
    2.4197
    2.4197
    2.8951
    3.0658
    3.9714
    4.3375
    4.3375
    4.8799
    4.9255
    4.9252
    4.9277
    5.4477
    5.9503
    5.9503
    6.6258
    6.6258


best_coordtwo =

         0
         0
    0.6875
    0.6873
    0.7171
    0.9587
    1.0832
    1.2731
    1.9819
    1.9819
    1.9819
    2.2749
    2.3496
    2.4593
    2.5171
    2.5171
    2.6381
    2.9562
    2.9562
    2.9562
    3.0038
    3.0757
    3.0757
    3.0757
    3.7058
    3.9716


best_targone =

  Columns 1 through 15

         0         0    0.0009    0.3842    0.5342    2.4197    2.4197    2.8951    3.0658    3.9714    4.3375    4.8799    4.9255    5.4477    5.9503
    0.0010    0.0009         0    0.3832    0.5332    2.4188    2.4188    2.8942    3.0648    3.9704    4.3365    4.8790    4.9246    5.4467    5.9493
    0.0010    0.0009         0    0.3832    0.5332    2.4188    2.4188    2.8942    3.0648    3.9704    4.3365    4.8790    4.9246    5.4467    5.9493
    0.3279    0.3279    0.3270    0.0563    0.2063    2.0918    2.0918    2.5672    2.7379    3.6435    4.0096    4.5520    4.5976    5.1198    5.6224
    0.7716    0.7716    0.7706    0.3874    0.2374    1.6482    1.6482    2.1235    2.2942    3.1998    3.5659    4.1084    4.1539    4.6761    5.1787
    4.3375    4.3375    4.3365    3.9533    3.8033    1.9177    1.9177    1.4424    1.2717    0.3661         0    0.5425    0.5880    1.1102    1.6128
    4.9252    4.9252    4.9243    4.5411    4.3911    2.5055    2.5055    2.0301    1.8595    0.9539    0.5878    0.0452         0    0.5224    1.0250
    4.9277    4.9277    4.9268    4.5436    4.3936    2.5080    2.5080    2.0326    1.8620    0.9564    0.5903    0.0478    0.0018    0.5199    1.0225
    5.9503    5.9503    5.9493    5.5661    5.4161    3.5306    3.5306    3.0552    2.8845    1.9789    1.6128    1.0703    1.0248    0.5026         0
    6.6258    6.6258    6.6249    6.2416    6.0916    4.2061    4.2061    3.7307    3.5600    2.6544    2.2883    1.7459    1.7003    1.1782    0.6755

  Column 16

    6.6258
    6.6249
    6.6249
    6.2979
    5.8542
    2.2883
    1.7006
    1.6981
    0.6755
         0


best_targtwo =

  Columns 1 through 15
```

```
         0    0.6875    1.9819    1.9819    2.2749    2.3496    2.4593    2.5171    2.5171    2.6381    2.9562    2.9562    3.0038    3.0757    3.0757
    0.6873         0    1.2946    1.2946    1.5877    1.6624    1.7721    1.8299    1.8299    1.9509    2.2689    2.2689    2.3165    2.3884    2.3884
    0.7171    0.0293    1.2647    1.2647    1.5578    1.6325    1.7422    1.8000    1.8000    1.9210    2.2391    2.2391    2.2866    2.3586    2.3586
    0.9587    0.2712    1.0232    1.0232    1.3162    1.3909    1.5007    1.5584    1.5584    1.6794    1.9975    1.9975    2.0451    2.1170    2.1170
    1.0832    0.3957    0.8986    0.8986    1.1917    1.2664    1.3761    1.4339    1.4339    1.5549    1.8730    1.8730    1.9205    1.9925    1.9925
    1.2731    0.5856    0.7088    0.7088    1.0018    1.0765    1.1862    1.2440    1.2440    1.3650    1.6831    1.6831    1.7306    1.8026    1.8026
    1.9819    1.2944         0         0    0.2931    0.3678    0.4775    0.5353    0.5353    0.6562    0.9743    0.9743    1.0219    1.0938    1.0938
    2.9562    2.2687    0.9743    0.9743    0.6813    0.6066    0.4969    0.4391    0.4391    0.3181         0         0    0.0476    0.1195    0.1195
    3.0757    2.3882    1.0938    1.0938    0.8008    0.7261    0.6163    0.5586    0.5586    0.4376    0.1195    0.1195    0.0719         0         0
    3.7058    3.0183    1.7239    1.7239    1.4308    1.3561    1.2464    1.1886    1.1886    1.0676    0.7496    0.7496    0.7020    0.6301    0.6301

  Column 16

    3.9716
    3.2843
    3.2545
    3.0129
    2.8884
    2.6985
    1.9897
    1.0154
    0.8959
    0.2658


best_addconone =

   -1.2587


best_addcontwo =

    1.4116


best_rowpermone =

     2
     8
     9
    10
     5
     4
     7
     3
     1
     6


best_colpermone =

     3
     8
     4
    12
     9
    11
    13
    16
     7
    14
    10
     6
    15
     1
     5
     2


best_rowpermtwo =

     4
     7
     3
     2
     8
    10
     6
     5
     1
     9


best_colpermtwo =

     1
    13
```

```
     4
    14
     9
    16
     8
    12
     3
    15
     6
     7
    11
     2
     5
    10


best_axes =

    5.9503    3.0757
         0    0.9587
    4.9277    0.7171
    4.3375         0
    0.7716    2.9562
    6.6258    1.9819
    4.9252    0.6873
    0.0009    1.0832
    0.0009    3.7058
    0.3279    1.2731
    5.4477         0
    6.6258    3.0757
         0    2.5171
    0.0009    1.9819
    5.9503    3.0757
    4.8799    2.9562
    3.0658    2.9562
         0    2.4593
    0.5342    2.2749
    4.3375    3.9716
    2.4197    3.0038
    0.3842    2.5171
    2.4197    0.6875
    3.9714    1.9819
    4.9255    2.6381
    2.8951    2.3496
```

3.3 (Multistructural) Two-Mode Nonmetric City-Block Scaling

3.3.1 The Enhanced Script for ms_bimonscaltmac.m

```
load eve_black_one.dat


best_vaf = 0.0;

store_vaf = zeros(100,1);

core = [4,4,4,4,4,4,4,4,4,4];

proxtm = [eve_black_one,core'];

maxiteration = 1000;

addcontol = .01;

for i = 1:100

    [find,vaf,targone,targtwo,outpermone,outpermtwo,rowpermone,colpermone, ...
        rowpermtwo,colpermtwo,addconone,addcontwo,coordone,coordtwo,axes,monproxtm] = ...
        ms_bimonscaltmac(proxtm,randperm(26),randperm(26),3,1,addcontol,maxiteration);



    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_find = find;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
```

```
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_targone = targone;
        best_targtwo = targtwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;
        best_rowpermone = rowpermone;
        best_colpermone = colpermone;
        best_rowpermtwo = rowpermtwo;
        best_colpermtwo = colpermtwo;
        best_axes = axes;
        best_monprox = monproxtm;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs

        best_vaf
        best_find
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo
        best_targone
        best_targtwo
        best_addconone
        best_addcontwo
        best_rowpermone
        best_colpermone
        best_rowpermtwo
        best_colpermtwo
        best_axes
        best_monprox
```

3.3.2 Enhanced Script Results for ms_bimonscaltmac.m

```
ms_script_bimonscaltmac

sorted_vafs =

  Columns 1 through 15

    0.9127    0.9127    0.9331    0.9342    0.9375    0.9379    0.9399    0.9406    0.9410    0.9419    0.9420    0.9422    0.9427    0.9427    0.9428

  Columns 16 through 30

    0.9428    0.9430    0.9430    0.9431    0.9434    0.9442    0.9447    0.9456    0.9463    0.9463    0.9467    0.9467    0.9469    0.9470    0.9470

  Columns 31 through 45

    0.9475    0.9475    0.9477    0.9482    0.9482    0.9484    0.9486    0.9489    0.9491    0.9491    0.9495    0.9495    0.9500    0.9500    0.9500

  Columns 46 through 60

    0.9500    0.9501    0.9502    0.9510    0.9510    0.9510    0.9511    0.9513    0.9514    0.9515    0.9515    0.9516    0.9516    0.9516    0.9516

  Columns 61 through 75

    0.9517    0.9517    0.9517    0.9517    0.9517    0.9517    0.9517    0.9517    0.9517    0.9528    0.9529    0.9529    0.9529    0.9529    0.9529

  Columns 76 through 90

    0.9529    0.9531    0.9532    0.9549    0.9549    0.9549    0.9549    0.9549    0.9549    0.9549    0.9549    0.9549    0.9549    0.9549    0.9551

  Columns 91 through 100

    0.9551    0.9551    0.9559    0.9559    0.9559    0.9560    0.9560    0.9560    0.9560    0.9560

best_vaf =

    0.9560

best_find =

  Columns 1 through 15
```

```
    3.3987    2.5101    6.1830    6.7296    2.0699    2.6032    3.6336    6.6069    5.3106    3.0835    3.8420    5.1782    5.0118    3.4806    2.0940
    6.8888    8.1447    2.6308    2.0843    7.7046    6.2106    5.1802    2.2070    3.5033    6.6909    4.9719    3.6357    4.0593    5.3333    6.7199
    2.6247    3.8807    6.8949    5.3590    3.4405    3.3151    4.3455    6.4711    4.9099    3.7954    4.5538    5.0424    3.6412    3.0799    2.8058
    2.5722    5.3017    6.9474    5.4115    4.8616    3.3676    4.3980    6.5236    4.9625    3.8479    4.6064    5.0949    3.4365    3.1324    3.8769
    6.5013    5.6748    3.0183    3.5648    5.2347    3.7407    2.7103    3.4421    2.1459    4.2210    2.5020    2.0134    3.6718    3.9759    4.2500
    3.4378    2.4709    7.3347    6.7687    3.2216    3.7549    4.7853    6.9109    5.3498    4.2352    4.9937    5.4822    5.0509    3.5197    3.2457
    2.6712    4.1843    6.8485    5.3125    3.7441    3.2687    4.2991    6.4246    4.8635    3.7490    4.5074    4.9959    3.3376    3.0335    2.7594
    6.3941    7.6500    3.1255    1.5896    7.2099    5.7159    4.6855    2.7017    3.0085    6.1962    4.4772    3.1410    3.5646    4.8386    6.2252
    7.9156    7.0891    2.5646    3.1111    5.6883    5.1550    4.1246    2.9885    3.5602    4.6747    3.9163    3.4277    5.0861    5.3902    5.6643
    6.3941    7.6500    3.1255    1.5896    7.2099    5.7159    4.6855    2.7017    3.0085    6.1962    4.4772    3.1410    3.5646    4.8386    6.2252

  Column 16

    4.1899
    4.6240
    3.7892
    3.8418
    3.2666
    4.2291
    3.7428
    4.1292
    4.6809
    4.1292


best_outpermone =

     2    13    18     8     9    14    10    22    19     5    21    23    26    17    24     4    20    16     7    25     3    11     1    15     6    12


best_outpermtwo =

     4    11    23     7     3     2     8    10    14    24     6    19    26    18    22    13    25    16     5    17    21     1    12    15     9    20


best_coordone =

         0
         0
         0
    0.4947
    0.4947
    0.4947
    0.4947
    1.4287
    1.4287
    1.4287
    2.3411
    2.3411
    2.5494
    2.5494
    3.2587
    3.5798
    3.5798
    3.5798
    4.0891
    4.0891
    4.2641
    4.5624
    4.5934
    4.5934
    5.1889
    5.5139


best_coordtwo =

         0
         0
    0.6082
    0.6082
    0.7368
    0.7368
    0.7368
    0.7368
    0.7368
    1.2217
    1.2217
    1.2217
    1.2217
    1.3542
    1.3542
    1.7780
    1.7780
    1.7780
    1.7780
```

```
    1.7780
    1.7780
    1.7780
    1.7780
    2.2583
    2.2583
    2.2583


best_targone =

  Columns 1 through 15

         0         0    0.4947    1.4287    1.4287    2.3411    2.3411    2.5494    2.5494    3.2587    3.5798    3.5798    4.0891    4.5624    4.5934
    0.4947    0.4947    0.0000    0.9340    0.9340    1.8464    1.8464    2.0547    2.0547    2.7640    3.0851    3.0851    3.5943    4.0677    4.0987
    0.4947    0.4947    0.0000    0.9340    0.9340    1.8464    1.8464    2.0547    2.0547    2.7640    3.0851    3.0851    3.5943    4.0677    4.0987
    0.4947    0.4947         0    0.9340    0.9340    1.8464    1.8464    2.0547    2.0547    2.7640    3.0851    3.0851    3.5943    4.0677    4.0987
    1.4287    1.4287    0.9340         0         0    0.9124    0.9124    1.1207    1.1207    1.8300    2.1511    2.1511    2.6604    3.1337    3.1647
    3.5798    3.5798    3.0851    2.1511    2.1511    1.2387    1.2387    1.0304    1.0304    0.3211         0    0.0000    0.5092    0.9826    1.0136
    4.0891    4.0891    3.5943    2.6604    2.6604    1.7480    1.7480    1.5397    1.5397    0.8303    0.5092    0.5092         0    0.4733    0.5044
    4.2641    4.2641    3.7694    2.8354    2.8354    1.9230    1.9230    1.7147    1.7147    1.0053    0.6843    0.6843    0.1750    0.2983    0.3294
    4.5934    4.5934    4.0987    3.1647    3.1647    2.2524    2.2524    2.0440    2.0440    1.3347    1.0136    1.0136    0.5044    0.0310    0.0000
    5.1889    5.1889    4.6942    3.7602    3.7602    2.8478    2.8478    2.6395    2.6395    1.9301    1.6091    1.6091    1.0998    0.6265    0.5954

  Column 16

    5.5139
    5.0192
    5.0192
    5.0192
    4.0852
    1.9341
    1.4248
    1.2498
    0.9205
    0.3250


best_targtwo =

  Columns 1 through 15

         0    0.6082    0.7368    1.2217    1.2217    1.2217    1.3542    1.3542    1.7780    1.7780    1.7780    1.7780    1.7780    1.7780    2.2583
    0.6082         0    0.1286    0.6135    0.6135    0.6135    0.7460    0.7460    1.1698    1.1698    1.1698    1.1698    1.1698    1.1698    1.6501
    0.7368    0.1286         0    0.4849    0.4849    0.4849    0.6174    0.6174    1.0412    1.0412    1.0412    1.0412    1.0412    1.0412    1.5215
    0.7368    0.1286         0    0.4849    0.4849    0.4849    0.6174    0.6174    1.0412    1.0412    1.0412    1.0412    1.0412    1.0412    1.5215
    0.7368    0.1286         0    0.4849    0.4849    0.4849    0.6174    0.6174    1.0412    1.0412    1.0412    1.0412    1.0412    1.0412    1.5215
    0.7368    0.1286         0    0.4849    0.4849    0.4849    0.6174    0.6174    1.0412    1.0412    1.0412    1.0412    1.0412    1.0412    1.5215
    1.2217    0.6135    0.4849         0         0         0    0.1324    0.1324    0.5563    0.5563    0.5563    0.5563    0.5563    0.5563    1.0366
    1.7780    1.1698    1.0412    0.5563    0.5563    0.5563    0.4238    0.4238    0.0000    0.0000         0         0    0.0000    0.0000    0.4803
    1.7780    1.1698    1.0412    0.5563    0.5563    0.5563    0.4238    0.4238    0.0000    0.0000    0.0000    0.0000         0    0.0000    0.4803
    2.2583    1.6501    1.5215    1.0366    1.0366    1.0366    0.9041    0.9041    0.4803    0.4803    0.4803    0.4803    0.4803    0.4803         0

  Column 16

    2.2583
    1.6501
    1.5215
    1.5215
    1.5215
    1.5215
    1.0366
    0.4803
    0.4803
    0.0000


best_addconone =

   -2.3698


best_addcontwo =

    0.7802


best_rowpermone =

     2
     8
     9
    10
     5
     4
     7
     3
```

```
     1
     6


best_colpermone =

     3
     8
     4
    12
     9
    11
    13
    16
     7
    14
    10
     6
    15
     1
     5
     2


best_rowpermtwo =

     4
     7
     3
     2
     8
    10
     6
     5
     1
     9


best_colpermtwo =

     1
    13
     4
    14
     9
    16
     8
    12
     3
    15
     6
     7
    11
     2
     5
    10


best_axes =

    4.5934    1.7780
         0    0.7368
    4.2641    0.7368
    3.5798         0
    1.4287    1.7780
    5.1889    1.2217
    4.0891    0.6082
    0.4947    0.7368
    0.4947    2.2583
    0.4947    0.7368
    4.5624         0
    5.5139    1.7780
         0    1.7780
    0.4947    0.7368
    4.5934    2.2583
    3.5798    1.7780
    2.5494    1.7780
         0    1.3542
    1.4287    1.2217
    3.5798    2.2583
    2.3411    1.7780
    1.4287    1.3542
    2.3411    0.6082
    3.2587    1.2217
    4.0891    1.7780
    2.5494    1.2217
```

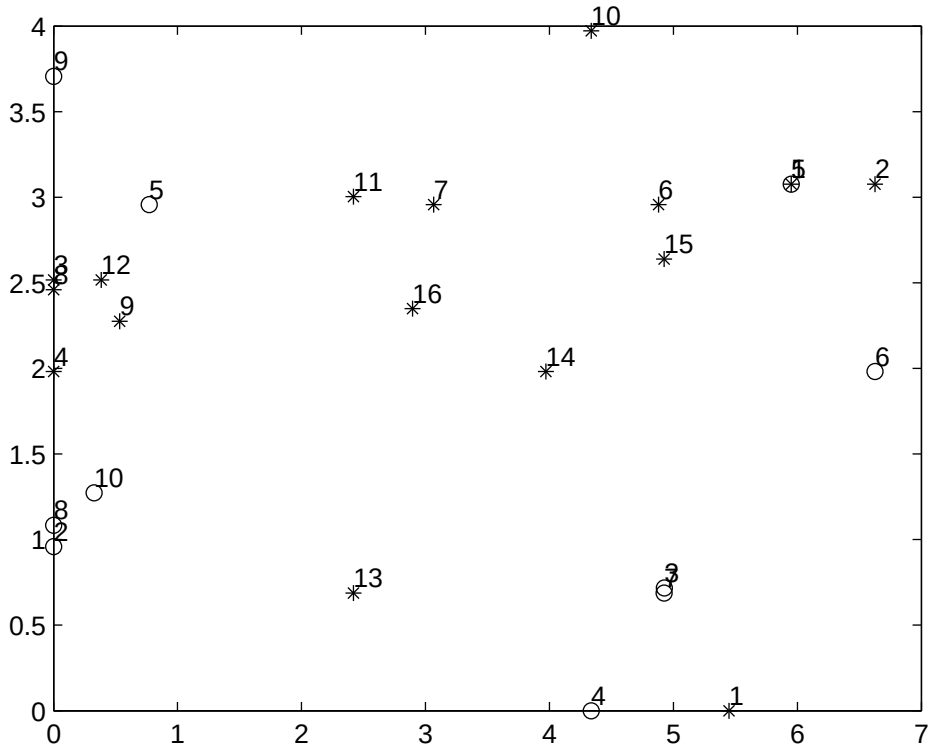


Figure 10: Best Two-Mode Metric Scaling for the eve_black_one Data Based on One-hundred Random Starts.

```
best_monprox =

  Columns 1 through 15

    3.5454    2.4076    6.2199    6.8344    1.9725    2.5176    3.4175    6.6639    4.9773    3.5454    3.5454    4.9773    5.0515    3.5454    1.9947
    6.9396    8.2678    2.5690    1.9545    7.8327    6.2713    4.9773    2.1250    3.4396    6.7795    4.9773    3.4175    3.4175    5.3535    6.7942
    2.5265    3.4175    6.9821    5.3873    3.5454    3.5454    4.3508    6.5381    4.9773    3.7879    3.5454    4.9773    3.4175    3.4175    2.7569
    2.4727    5.3301    7.0359    4.9773    4.9773    3.5454    4.4046    6.5919    4.9773    3.5454    4.9773    4.9773    3.3764    3.5454    3.4175
    6.5892    4.9773    2.9194    3.4175    5.2730    3.4175    2.6406    3.3634    3.4175    4.9773    3.4175    1.9084    3.6688    4.9773    4.2345
    3.4239    2.3622    7.4342    6.8798    3.5454    3.4175    4.9773    6.9902    4.9773    4.2401    4.9773    4.9773    5.0968    3.4175    3.5454
    2.5735    4.9773    6.9351    5.3403    3.4175    3.2327    4.3038    6.4910    4.9773    3.7409    4.9773    4.9773    3.2756    2.9816    2.7099
    6.4495    7.7777    3.4175    1.4643    7.3426    5.7812    4.9773    2.6151    3.4175    4.9773    3.5454    3.4175    3.4175    4.8633    6.3041
    8.0623    7.1812    2.4626    3.0771    5.7298    4.9773    4.9773    2.9067    3.4175    4.6765    3.5454    3.4175    5.1419    4.9773    5.7076
    6.4495    7.7777    3.0591    1.4643    7.3426    5.7812    3.4175    2.6151    3.4175    6.2893    4.9773    3.4175    3.4175    4.8633    6.3041

  Column 16

    4.1731
    4.6158
    3.7663
    3.8202
    3.5454
    4.2185
    3.7193
    4.1257
    4.6982
    4.1257
```

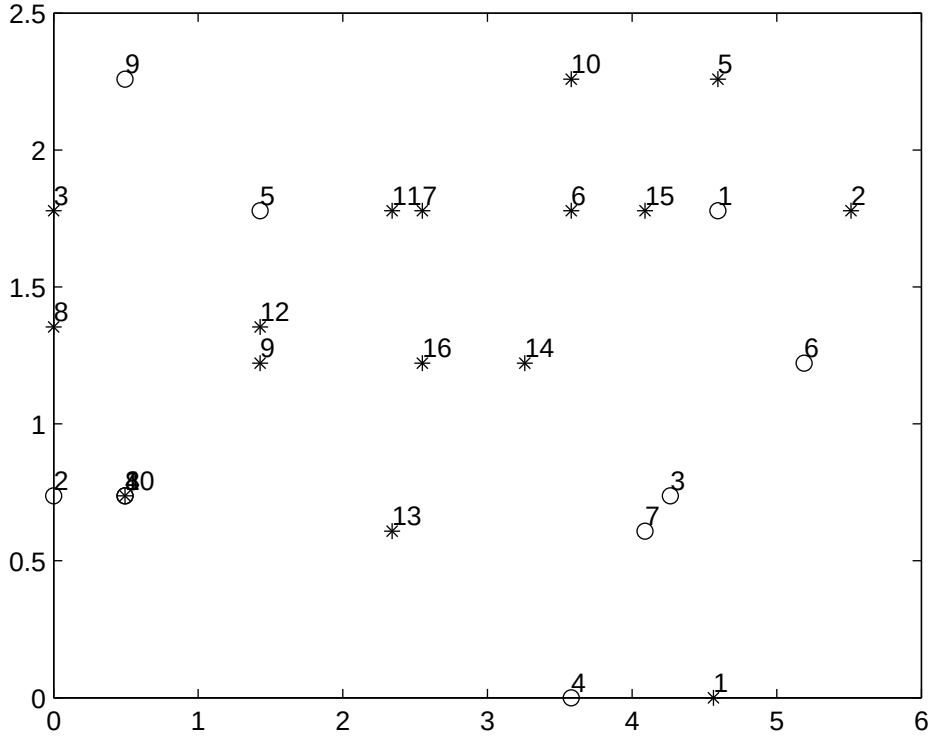


Figure 11: Best Two-Mode Nonmetric Scaling for the eve_black_one Data Based on One-hundred Random Starts.

4 Choice of the City-Block or Euclidean Representational Metric

As suggested already in our earlier scripts for carrying out metric and nonmetric scaling in the city-block and Euclidean distance measure, the choice of representational metric may not really make that much substantive difference. This section explores this dichotomy further (between the use of the city-block and Euclidean representational metric): a first script is implemented for comparing representations for the Hampshire proximities; a second script includes a data generation M-file for producing proximity matrices reflecting a noise-corrupted underlying city-block or Euclidean structure. This latter script has a number of usable variations to study the city-block versus Euclidean scaling task, although here only one run will be given for

an instance of data conforming to a perfect (error-free) city-block and/or Euclidean structure.

4.1 The Hampshire Proximity Matrix

As can be seen in the output below for the Hampshire proximities, the Euclidean and city-block representations have respective VAFs of .9755 and .9499 (both values are the best over 100 random starts, with only a very few local optima observed in either instance). It is somewhat ambiguous as to where the original Norden Hampshire distances actually came from, but presumably by inspection of some crude county map of the time. Thus, it may note be surprising that the Euclidean representation has a higher VAF; but it is somewhat surprising how little is lost (about 2.5%) in using a city-block distance measure.

Figure 12 gives the city-block and Euclidean scalings of the Hampshire proximities and a Procrustes transformed set of Euclidean coordinates to conform to the city-block target. All of the details are given in the output obtained from the given script. The two representations are remarkably the same. An inspection of the Cobbett 1930 Map of Hampshire County given in Figure 2 shows clearly an interpretation of the axes structure in Figure 12: a south-east to north-west axis is from ChristChurch (14) and Ringwood (15) to Odiham (6) and Alton (22); a north-east to south-west axis is from Portesmouth (26) and Havant (24) to Kingsclere (19) and Andover (18).

4.1.1 The Script for Comparing Representations for the Hampshire Proximities: ms_script_hampshire_cityblock_versus_euclidean

```
load hampshire_proximities.dat

tic;

n = 27;

best_vaf = 0.0;

store_vaf = zeros(100,1);

for i = 1:100

    [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,...
        addcontwo,vaf] = ms_biscalqa(hampshire_proximities,targlin(n),targlin(n), ...
        randperm(n),randperm(n),3,1);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)
```

```
        best_vaf = vaf;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_fitone = fitone;
        best_fittwo = fittwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs

        best_vaf
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo

        best_addconone
        best_addcontwo

        axes = zeros(n,2);

        for i = 1:n

            axes(best_outpermone(i),1) = best_coordone(i);
            axes(best_outpermtwo(i),2) = best_coordtwo(i);
        end

        figure(1)

        plot(axes(1:n,1),axes(1:n,2),'ko')

        hold on

        for i = 1:n

            objectlabels{i,1} = int2str(i);

        end

        text(axes(1:n,1),axes(1:n,2),objectlabels,'fontsize',10,'verticalalignment','bottom')


toc;

cityblock_coordinates = axes;



tic;

opts = statset('Maxiter',1000);

best_vaf = 0.0;

store_vaf = zeros(100,1);

for k = 1:100

[coords,stress] = mdscale(hampshire_proximities,2,'Criterion','metricstress', ...
'Start', 'random','Replicates',1,'Options',opts);

n = size(coords,1);

distance_matrix = zeros(n,n);

for i = 1:n
    for j = 1:n

        distance_matrix(i,j) = sqrt(((coords(i,1) - coords(j,1))^2) + ...
            ((coords(i,2) - coords(j,2))^2));
    end
end

ave_diff = (sum(sum(hampshire_proximities - distance_matrix)))/(n*(n-1));

ave_proximity = (sum(sum(hampshire_proximities)))/(n*(n-1));

hampshire_proximities_vec = squareform(hampshire_proximities);
```

```
distance_vec = squareform(distance_matrix);

r = corrcoef(hampshire_proximities_vec',distance_vec');

vaf = r(1,2)^2;

store_vaf(k) = vaf;

if(vaf > best_vaf)

    best_vaf = vaf;
    best_coords = coords;

end end

store_vaf;

sorted_vafs = sort(store_vaf');

sorted_vafs

best_vaf

best_coords

figure(2)

axis equal

plot(best_coords(:,1),best_coords(:,2),'ko')

hold on

for i = 1:n

    objectlabels{i,1} = int2str(i);

end

text(best_coords(:,1),best_coords(:,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

toc;

euclidean_coordinates = [best_coords(:,1),best_coords(:,2)];

[d,z,transform] = procrustes(cityblock_coordinates,euclidean_coordinates);

figure(3)

plot(cityblock_coordinates(:,1),cityblock_coordinates(:,2),'rx',...
    euclidean_coordinates(:,1),euclidean_coordinates(:,2),'b.',...
    z(:,1),z(:,2),'ko')

hold on

text(cityblock_coordinates(:,1),cityblock_coordinates(:,2),objectlabels,'fontsize',8,'verticalalignment','bottom')

text(z(:,1),z(:,2),objectlabels,'fontsize',8,'verticalalignment','bottom')

transform(1).b

transform(1).T

transform(1).c
```

4.1.2 The Results for the Script, ms_script_hampshire_cityblock_versus_euclidean

```
>> ms_script_hampshire_cityblock_versus_euclidean

sorted_vafs =

  Columns 1 through 15

    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486

  Columns 16 through 30

    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486

  Columns 31 through 45
```

```
    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486

  Columns 46 through 60

    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9486    0.9499    0.9499    0.9499    0.9499    0.9499

  Columns 61 through 75

    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499

  Columns 76 through 90

    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499

  Columns 91 through 100

    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499    0.9499

best_vaf =

    0.9499

best_outpermone =

  Columns 1 through 26

     2     6    12    19    22     1    11     4     5    23    21    18     3    27    20    24    17    10    25     9    26    13     8    16     7    15

  Column 27

    14

best_outpermtwo =

  Columns 1 through 26

    18     4    19    16     3    11    15    17     5    14    27     7     8    13    12    21    20     6     2    22     9    10    25    23     1    26

  Column 27

    24

best_coordone =

  -18.7774
  -14.5546
  -13.3532
  -12.7321
  -10.7176
  -10.4929
   -9.1592
   -7.5277
   -6.3345
   -5.0574
   -4.5962
   -3.9197
   -0.5995
   -0.5995
    1.0962
    2.9431
    4.1366
    4.3894
    5.0841
    6.1710
    6.7689
    7.1528
   12.0962
   13.8566
   15.3812
   17.2920
   22.0535

best_coordtwo =

   -7.1710
   -5.9032
   -5.3954
   -4.9343
   -4.9031
   -4.3382
   -3.5487
```

```
   -3.5487
   -2.3513
   -2.1245
   -1.8605
   -0.8060
   -0.7641
   -0.6959
   -0.0770
    0.8202
    0.9491
    1.8868
    2.3597
    2.7219
    3.3432
    3.5636
    4.0583
    5.9585
    6.8939
    7.4466
    8.4201


best_addconone =

   -6.4552


best_addcontwo =

    5.0638

Elapsed time is 1485.404266 seconds.

sorted_vafs =

  Columns 1 through 15

    0.8420    0.8432    0.8593    0.8608    0.8754    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755

  Columns 16 through 30

    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755

  Columns 31 through 45

    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755

  Columns 46 through 60

    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755

  Columns 61 through 75

    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755

  Columns 76 through 90

    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755

  Columns 91 through 100

    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755    0.9755


best_vaf =

    0.9755


best_coords =

  -14.4603    9.7080
  -22.7771    1.8147
    1.8802   -8.3178
   -8.6078  -10.2091
   -6.5298   -4.5963
  -17.8177    0.7080
   18.6371    1.1553
   15.2906    1.7609
    7.0855    7.1429
    4.1702    7.4095
  -10.3665   -7.9665
  -15.6237   -3.0821
    9.1125    1.6918
   25.8294   -3.7077
   20.1378   -7.1154
   16.0074   -9.5425
```

```
    6.5181   -4.4801
   -3.8430  -11.9393
  -14.7009   -9.4021
    1.0779    2.4427
   -5.5290    0.5170
  -13.1083    4.1042
   -7.5749    8.9328
    1.9970   14.2860
    5.2753    8.3908
    7.6876   13.3285
    0.2322   -3.0344

Elapsed time is 10.098065 seconds.

ans =

    0.7678


ans =

    0.9977   -0.0679
    0.0679    0.9977


ans =

  1.0e-014 *

   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
   -0.1201    0.0424
```

4.2 A Data Generation Mechanism for Comparing Representations of Proximities Having an Underlying City-Block or Euclidean Structure

To exert more control over the conduct of studies that compare representational distance measures, the help-header is given below for the M-file, `data_generate_cityblock_and_euclidean.m`:

```
% DATA_GENERATE_CITYBLOCK_AND_EUCLIDEAN produces data having an underlying
% city-block and Euclidean structure.
%
% syntax: [datamatrix_cb,datamatrix_eu,corr_cb_eu,gamma_cb_eu,datavectors,coord] ...
```

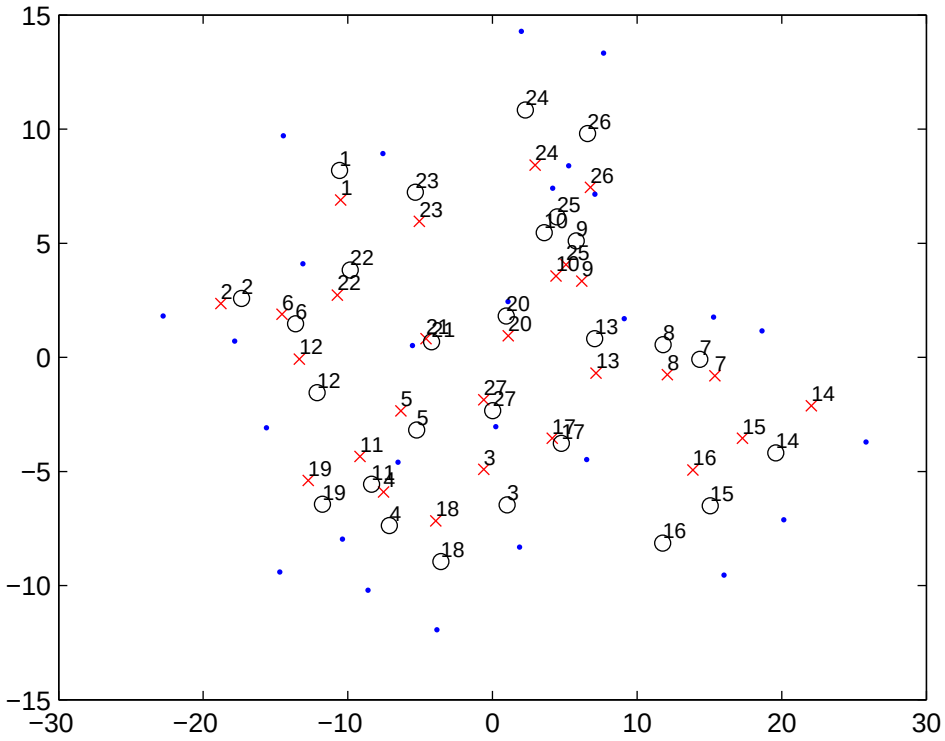


Figure 12: The City-Block and Euclidean Scalings of the Hampshire Proximities (Distances): x: City-Block Scaling Coordinates; .: Euclidean Scaling Coordinates; o: Procrustes Transformed Euclidean Coordinates.

```
%    = data_generate_cityblock_and_euclidean(n,error,elongation_factor)
%
% The input parameters are the number of objects, $n$; ERROR is in the form of
% a multiplier used on a randomly (entry) permutated proximity matrix with
% perfect city-block or Euclidean structure that is then added to these
% latter matrices; the ELONGATION_FACTOR refers to a multiplier used on the
% randomly generated first dimension coordinates (in case a dominant
% dimension is desired).
% The outputs are the cityblock and Euclidean $n \times n$ proximity
% matrices, DATAMATRIX_CB and DATAMATRIX_EU, respectively; CORR_CB_EU and
% GAMMA_CB_EU are the Pearson and Goodman-Kruskal coefficients of correlation
% between the entries in DATAMATRIX_CB and DATAMATRIX_EU; DATAVECTORS is
% $n*(n-1)/2 \times 2$ and contains the entries in DATAMATRIX_CB and
% DATAMATRIX_EU in vector form; COORD contains the randomly generated
% coordinates.
```

The results of running a script with error set at 0.0 (i.e., perfect city-block and Euclidean data) and the elongation factor set at 1.0, are reproduced below. As might be expected, `ms_biscalqa.m` retrieves the city-block structure (VAF of .9999) and `mdscale.m` retrieves the Euclidean (VAF of 1.000). What may be more interesting is how well both routines do on "each other's data": `ms_biscalqa.m` on the Euclidean data gives a VAF of .9804; `mdscale.m` on the city-block data gives a VAF of .9736. Graphically, this is demonstrated in Figures 13 and 14 where is is obvious that either `ms_biscalqa.m` or `mdscale.m` can essentially retrieve the underlying coordinate structure irrespective of whether the generated data used is city-block or Euclidean to begin with. Maybe this is not very surprising given the very high Pearson correlation reported of .9724 (`corr_cb_eu`) between the two generated data sets obtained from the same object (coordinate) placement.

4.2.1 The Script for Comparing Representations of Proximities Having an Underlying City-Block or Euclidean Structure: ms_script_cityblock_versus_euclidean

```
[datamatrix_cb,datamatrix_eu,corr_cb_eu,gamma_cb_eu,datavectors,coord] = ...
    data_generate_cityblock_and_euclidean(20,0.0,1.0);

datamatrix_cb

datamatrix_eu

corr_cb_eu

gamma_cb_eu

coord
```

```
tic;

n = 20;

best_vaf = 0.0;

store_vaf = zeros(100,1);


for i = 1:100

    [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,...
        addcontwo,vaf] = ms_biscalqa(datamatrix_cb,targlin(n),targlin(n), ...
        randperm(n),randperm(n),3,1);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_fitone = fitone;
        best_fittwo = fittwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs

        best_vaf
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo

        best_addconone
        best_addcontwo

        axes = zeros(n,2);

        for i = 1:n

            axes(best_outpermone(i),1) = best_coordone(i);
            axes(best_outpermtwo(i),2) = best_coordtwo(i);
        end

        figure(2)

        plot(axes(1:n,1),axes(1:n,2),'ko')

        hold on

        for i = 1:n

            objectlabels{i,1} = int2str(i);

        end

        text(axes(1:n,1),axes(1:n,2),objectlabels,'fontsize',10,'verticalalignment','bottom')


toc;

cityblock_coordinates = axes;



tic;

opts = statset('Maxiter',1000);

best_vaf = 0.0;

store_vaf = zeros(100,1);

for k = 1:100

[coords,stress] = mdscale(datamatrix_cb,2,'Criterion','metricstress', 'Start',...
    'random','Replicates',1,'Options',opts);
```

```
n = size(coords,1);

distance_matrix = zeros(n,n);

for i = 1:n
    for j = 1:n

        distance_matrix(i,j) = sqrt(((coords(i,1) - coords(j,1))^2) + ...
            ((coords(i,2) - coords(j,2))^2));
    end
end

ave_diff = (sum(sum(datamatrix_cb - distance_matrix)))/(n*(n-1));

ave_proximity = (sum(sum(datamatrix_cb)))/(n*(n-1));

datamatrix_cb_vec = squareform(datamatrix_cb);

distance_vec = squareform(distance_matrix);

r = corrcoef(datamatrix_cb_vec',distance_vec');

vaf = r(1,2)^2;

store_vaf(k) = vaf;

if(vaf > best_vaf)

    best_vaf = vaf;
    best_coords = coords;

end end

store_vaf;

sorted_vafs = sort(store_vaf');

sorted_vafs

best_vaf

best_coords

figure(3)

axis equal

plot(best_coords(:,1),best_coords(:,2),'ko')

hold on

for i = 1:n

    objectlabels{i,1} = int2str(i);

end

text(best_coords(:,1),best_coords(:,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

toc;

euclidean_coordinates = [best_coords(:,1),best_coords(:,2)];

[d,z,transform] = procrustes(cityblock_coordinates,euclidean_coordinates);

figure(4)

plot(cityblock_coordinates(:,1),cityblock_coordinates(:,2),'rx',...
    euclidean_coordinates(:,1),euclidean_coordinates(:,2),'b.',...
    z(:,1),z(:,2),'go')

hold on

text(cityblock_coordinates(:,1),cityblock_coordinates(:,2),objectlabels,'fontsize',8,'verticalalignment','bottom')

text(z(:,1),z(:,2),objectlabels,'fontsize',8,'verticalalignment','bottom')

transform(1).b

transform(1).T

transform(1).c

%-------------------------------------
```

```
tic;

n = 20;

best_vaf = 0.0;

store_vaf = zeros(100,1);

for i = 1:100

    [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,...
        addcontwo,vaf] = ms_biscalqa(datamatrix_eu,targlin(n),targlin(n), ...
        randperm(n),randperm(n),3,1);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_fitone = fitone;
        best_fittwo = fittwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs

        best_vaf
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo

        best_addconone
        best_addcontwo

        axes = zeros(n,2);

        for i = 1:n

            axes(best_outpermone(i),1) = best_coordone(i);
            axes(best_outpermtwo(i),2) = best_coordtwo(i);
        end

        figure(5)

        plot(axes(1:n,1),axes(1:n,2),'ko')

        hold on

        for i = 1:n

            objectlabels{i,1} = int2str(i);

        end

        text(axes(1:n,1),axes(1:n,2),objectlabels,'fontsize',10,'verticalalignment','bottom')


toc;

cityblock_coordinates = axes;



tic;

opts = statset('Maxiter',1000);

best_vaf = 0.0;

store_vaf = zeros(100,1);

for k = 1:100

[coords,stress] = mdscale(datamatrix_eu,2,'Criterion','metricstress', 'Start',...
    'random','Replicates',1,'Options',opts);
```

```
n = size(coords,1);

distance_matrix = zeros(n,n);

for i = 1:n
    for j = 1:n

        distance_matrix(i,j) = sqrt(((coords(i,1) - coords(j,1))^2) + ...
            ((coords(i,2) - coords(j,2))^2));
    end
end

ave_diff = (sum(sum(datamatrix_eu - distance_matrix)))/(n*(n-1));

ave_proximity = (sum(sum(datamatrix_eu)))/(n*(n-1));

datamatrix_eu_vec = squareform(datamatrix_eu);

distance_vec = squareform(distance_matrix);

r = corrcoef(datamatrix_eu_vec',distance_vec');

vaf = r(1,2)^2;

store_vaf(k) = vaf;

if(vaf > best_vaf)

    best_vaf = vaf;
    best_coords = coords;

end end

store_vaf;

sorted_vafs = sort(store_vaf');

sorted_vafs

best_vaf

best_coords

figure(6)

axis equal

plot(best_coords(:,1),best_coords(:,2),'ko')

hold on

for i = 1:n

    objectlabels{i,1} = int2str(i);

end

text(best_coords(:,1),best_coords(:,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

toc;

euclidean_coordinates = [best_coords(:,1),best_coords(:,2)];

[d,z,transform] = procrustes(cityblock_coordinates,euclidean_coordinates);

figure(7)

plot(cityblock_coordinates(:,1),cityblock_coordinates(:,2),'rx',...
    euclidean_coordinates(:,1),euclidean_coordinates(:,2),'b.',...
    z(:,1),z(:,2),'go')

hold on

text(cityblock_coordinates(:,1),cityblock_coordinates(:,2),objectlabels,'fontsize',8,'verticalalignment','bottom')

text(z(:,1),z(:,2),objectlabels,'fontsize',8,'verticalalignment','bottom')

transform(1).b

transform(1).T

transform(1).c
```

4.2.2 The Results for the Script, ms_script_cityblock_versus_euclidean

```
>> ms_script_cityblock_versus_euclidean

datamatrix_cb =

  Columns 1 through 15

         0    0.5859    0.9075    0.4091    0.6652    0.2607    0.6057    0.6926    0.5730    0.7578    0.4458    0.9221    0.8482    0.9291    0.8942
    0.5859         0    0.9852    0.1768    1.0623    0.4509    0.6833    0.9799    0.6506    1.0799    0.4793    0.9997    0.9259    0.5890    0.9718
    0.9075    0.9852         0    1.0496    0.5970    1.1682    0.3018    0.5147    0.3345    0.6146    1.3533    0.5219    0.3923    0.8084    0.2268
    0.4091    0.1768    1.0496         0    0.8855    0.2741    0.7478    0.8347    0.7151    0.9031    0.3037    1.0642    0.9903    0.6534    1.0363
    0.6652    1.0623    0.5970    0.8855         0    0.9259    0.8343    0.0823    0.4399    0.0926    1.1109    0.2569    0.2047    1.4054    0.8239
    0.2607    0.4509    1.1682    0.2741    0.9259         0    0.8664    0.9533    0.8337    1.0185    0.1850    1.1828    1.1089    0.7940    1.1549
    0.6057    0.6833    0.3018    0.7478    0.8343    0.8664         0    0.7519    0.3944    0.8518    1.0515    0.7591    0.6296    0.5711    0.2885
    0.6926    0.9799    0.5147    0.8347    0.0823    0.9533    0.7519         0    0.3575    0.0999    1.1384    0.2295    0.1556    1.3231    0.7415
    0.5730    0.6506    0.3345    0.7151    0.4399    0.8337    0.3944    0.3575         0    0.4574    1.0188    0.3647    0.2752    0.9655    0.3840
    0.7578    1.0799    0.6146    0.9031    0.0926    1.0185    0.8518    0.0999    0.4574         0    1.2036    0.1643    0.2223    1.4230    0.8414
    0.4458    0.4793    1.3533    0.3037    1.1109    0.1850    1.0515    1.1384    1.0188    1.2036         0    1.3678    1.2940    0.9571    1.3399
    0.9221    0.9997    0.5219    1.0642    0.2569    1.1828    0.7591    0.2295    0.3647    0.1643    1.3678         0    0.1296    1.3303    0.7487
    0.8482    0.9259    0.3923    0.9903    0.2047    1.1089    0.6296    0.1556    0.2752    0.2223    1.2940    0.1296         0    1.2007    0.6191
    0.9291    0.5890    0.8084    0.6534    1.4054    0.7940    0.5711    1.3231    0.9655    1.4230    0.9571    1.3303    1.2007         0    0.5816
    0.8942    0.9718    0.2268    1.0363    0.8239    1.1549    0.2885    0.7415    0.3840    0.8414    1.3399    0.7487    0.6191    0.5816         0
    0.3037    0.2912    1.2112    0.1616    0.9689    0.1596    0.9094    0.9963    0.8767    1.0615    0.1880    1.2258    1.1519    0.8150    1.1978
    0.3639    0.6259    0.5436    0.5060    0.4363    0.6246    0.3979    0.3540    0.2091    0.4539    0.8097    0.5581    0.4843    0.9691    0.5302
    0.6768    1.2627    0.7975    1.0859    0.2998    0.8119    1.0347    0.3273    0.6403    0.3925    0.8111    0.5567    0.4829    1.6058    1.0243
    0.6137    0.6914    0.2938    0.7558    0.6488    0.8744    0.1854    0.5665    0.2090    0.6664    1.0595    0.5737    0.4441    0.7566    0.2804
    0.6888    1.2747    0.8095    1.0979    0.2124    0.8239    1.0467    0.2948    0.6523    0.2121    0.9914    0.3764    0.4172    1.6178    1.0363

  Columns 16 through 20

    0.3037    0.3639    0.6768    0.6137    0.6888
    0.2912    0.6259    1.2627    0.6914    1.2747
    1.2112    0.5436    0.7975    0.2938    0.8095
    0.1616    0.5060    1.0859    0.7558    1.0979
    0.9689    0.4363    0.2998    0.6488    0.2124
    0.1596    0.6246    0.8119    0.8744    0.8239
    0.9094    0.3979    1.0347    0.1854    1.0467
    0.9963    0.3540    0.3273    0.5665    0.2948
    0.8767    0.2091    0.6403    0.2090    0.6523
    1.0615    0.4539    0.3925    0.6664    0.2121
    0.1880    0.8097    0.8111    1.0595    0.9914
    1.2258    0.5581    0.5567    0.5737    0.3764
    1.1519    0.4843    0.4829    0.4441    0.4172
    0.8150    0.9691    1.6058    0.7566    1.6178
    1.1978    0.5302    1.0243    0.2804    1.0363
         0    0.6676    0.9715    0.9174    0.9835
    0.6676         0    0.6368    0.2498    0.6488
    0.9715    0.6368         0    0.8493    0.1803
    0.9174    0.2498    0.8493         0    0.8613
    0.9835    0.6488    0.1803    0.8613         0


datamatrix_eu =

  Columns 1 through 15

         0    0.4179    0.6474    0.3063    0.5785    0.2076    0.4975    0.5635    0.4060    0.6396    0.3241    0.7111    0.6298    0.7499    0.6783
    0.4179         0    0.7704    0.1331    0.9166    0.3441    0.5094    0.8813    0.6367    0.9654    0.4273    0.9934    0.8924    0.4820    0.7075
    0.6474    0.7704         0    0.7697    0.4556    0.8261    0.2715    0.3944    0.2613    0.4473    0.9570    0.3692    0.2806    0.6366    0.1607
    0.3063    0.1331    0.7697         0    0.8473    0.2112    0.5300    0.8191    0.5958    0.9015    0.3031    0.9458    0.8498    0.5905    0.7340
    0.5785    0.9166    0.4556    0.8473         0    0.7846    0.5914    0.0614    0.3178    0.0667    0.8872    0.1893    0.1942    0.9997    0.6046
    0.2076    0.3441    0.8261    0.2112    0.7846         0    0.6326    0.7710    0.6019    0.8465    0.1324    0.9183    0.8352    0.7831    0.8303
    0.4975    0.5094    0.2715    0.5300    0.5914    0.6326         0    0.5352    0.2798    0.6119    0.7644    0.5815    0.4770    0.4093    0.2041
    0.5635    0.8813    0.3944    0.8191    0.0614    0.7710    0.5352         0    0.2666    0.0844    0.8794    0.1623    0.1400    0.9443    0.5434
    0.4060    0.6367    0.2613    0.5958    0.3178    0.6019    0.2798    0.2666         0    0.3489    0.7275    0.3570    0.2560    0.6841    0.3540
    0.6396    0.9654    0.4473    0.9015    0.0667    0.8465    0.6119    0.0844    0.3489         0    0.9511    0.1334    0.1697    1.0212    0.6027
    0.3241    0.4273    0.9570    0.3031    0.8872    0.1324    0.7644    0.8794    0.7275    0.9511         0    1.0317    0.9533    0.8923    0.9627
    0.7111    0.9934    0.3692    0.9458    0.1893    0.9183    0.5815    0.1623    0.3570    0.1334    1.0317         0    0.1055    0.9845    0.5298
    0.6298    0.8924    0.2806    0.8498    0.1942    0.8352    0.4770    0.1400    0.2560    0.1697    0.9533    0.1055         0    0.8818    0.4390
    0.7499    0.4820    0.6366    0.5905    0.9997    0.7831    0.4093    0.9443    0.6841    1.0212    0.8923    0.9845    0.8818         0    0.4923
    0.6783    0.7075    0.1607    0.7340    0.6046    0.8303    0.2041    0.5434    0.3540    0.6027    0.9627    0.5298    0.4390    0.4923         0
    0.2992    0.2607    0.8653    0.1400    0.8756    0.1169    0.6446    0.8564    0.6627    0.9351    0.1667    0.9967    0.9076    0.7303    0.8478
    0.2589    0.5417    0.4008    0.4784    0.3749    0.4587    0.3292    0.3416    0.1479    0.4249    0.5819    0.4701    0.3796    0.6931    0.4644
    0.5438    0.9462    0.6819    0.8476    0.2550    0.7249    0.7511    0.3058    0.4760    0.3062    0.7974    0.4393    0.4457    1.1402    0.8151
    0.4508    0.5696    0.2110    0.5600    0.4602    0.6189    0.1312    0.4044    0.1505    0.4821    0.7506    0.4605    0.3551    0.5399    0.2337
    0.6214    1.0033    0.6270    0.9161    0.1724    0.8152    0.7413    0.2329    0.4617    0.2037    0.8988    0.3350    0.3644    1.1442    0.7731

  Columns 16 through 20

    0.2992    0.2589    0.5438    0.4508    0.6214
    0.2607    0.5417    0.9462    0.5696    1.0033
    0.8653    0.4008    0.6819    0.2110    0.6270
    0.1400    0.4784    0.8476    0.5600    0.9161
```

```
    0.8756    0.3749    0.2550    0.4602    0.1724
    0.1169    0.4587    0.7249    0.6189    0.8152
    0.6446    0.3292    0.7511    0.1312    0.7413
    0.8564    0.3416    0.3058    0.4044    0.2329
    0.6627    0.1479    0.4760    0.1505    0.4617
    0.9351    0.4249    0.3062    0.4821    0.2037
    0.1667    0.5819    0.7974    0.7506    0.8988
    0.9967    0.4701    0.4393    0.4605    0.3350
    0.9076    0.3796    0.4457    0.3551    0.3644
    0.7303    0.6931    1.1402    0.5399    1.1442
    0.8478    0.4644    0.8151    0.2337    0.7731
         0    0.5281    0.8341    0.6544    0.9189
    0.5281         0    0.4503    0.2319    0.4764
    0.8341    0.4503         0    0.6257    0.1278
    0.6544    0.2319    0.6257         0    0.6109
    0.9189    0.4764    0.1278    0.6109         0


corr_cb_eu =

    1.0000    0.9724
    0.9724    1.0000


gamma_cb_eu =

    0.8751


coord =

    0.6656    0.7807
    0.9973    0.5265
    0.2721    0.2666
    0.9412    0.6472
    0.0948    0.6863
    0.8635    0.8435
    0.5417    0.2989
    0.1222    0.6314
    0.3608    0.5124
    0.0397    0.6487
    0.9418    0.9502
    0.0039    0.5202
    0.1056    0.4924
    0.8744    0.0605
    0.3922    0.1598
    0.9648    0.7852
    0.4636    0.6187
    0.1445    0.9364
    0.4449    0.3876
    0.0483    0.8522


sorted_vafs =

  Columns 1 through 15

    0.9951    0.9951    0.9951    0.9951    0.9951    0.9951    0.9951    0.9951    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961

  Columns 16 through 30

    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961

  Columns 31 through 45

    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961

  Columns 46 through 60

    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961

  Columns 61 through 75

    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961

  Columns 76 through 90

    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9961    0.9999

  Columns 91 through 100

    0.9999    0.9999    0.9999    0.9999    0.9999    0.9999    0.9999    0.9999    0.9999    0.9999


best_vaf =

    0.9999
```

```
best_outpermone =

    14    15     3     7    19    13     9    12     2     8    17    10     4     5     1    16     6    20    18    11


best_outpermtwo =

    12    10    20     5    13     8    18     3     9    15    19    17     7     1     6    14    11     4    16     2


best_coordone =

   -0.5190
   -0.4204
   -0.3137
   -0.2813
   -0.1927
   -0.0884
   -0.0680
   -0.0610
   -0.0528
    0.0445
    0.0445
    0.0675
    0.0677
    0.1053
    0.2005
    0.2060
    0.2635
    0.2707
    0.3550
    0.3721


best_coordtwo =

   -0.4597
   -0.4244
   -0.4165
   -0.3693
   -0.3580
   -0.3429
   -0.3208
   -0.1908
   -0.1022
   -0.0707
   -0.0181
   -0.0009
    0.0780
    0.2012
    0.3989
    0.4111
    0.4756
    0.4768
    0.4997
    0.5330


best_addconone =

   -0.4184


best_addcontwo =

    0.4181

Elapsed time is 421.884150 seconds.

sorted_vafs =

  Columns 1 through 15

    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736

  Columns 16 through 30

    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736

  Columns 31 through 45

    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736

  Columns 46 through 60
```

```
    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736

  Columns 61 through 75

    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736

  Columns 76 through 90

    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736

  Columns 91 through 100

    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736    0.9736


best_vaf =

    0.9736


best_coords =

   -0.2496    0.2481
   -0.5944   -0.0768
    0.3126   -0.4369
   -0.5338    0.0950
    0.4147    0.1920
   -0.4974    0.3421
   -0.0712   -0.4461
    0.3947    0.1127
    0.1643   -0.1526
    0.4842    0.1595
   -0.6468    0.4404
    0.5633   -0.0169
    0.4582   -0.0594
   -0.6408   -0.6070
    0.1400   -0.5945
   -0.5980    0.2649
    0.0160   -0.0247
    0.3627    0.4811
    0.0614   -0.3247
    0.4599    0.4037

Elapsed time is 5.338398 seconds.

ans =

    0.7809


ans =

   -0.0390   -0.9992
    0.9992   -0.0390


ans =

  1.0e-016 *

    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873
    0.1890    0.1873


sorted_vafs =

  Columns 1 through 15

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804
```

```
  Columns 16 through 30

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804

  Columns 31 through 45

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804

  Columns 46 through 60

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804

  Columns 61 through 75

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804

  Columns 76 through 90

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804

  Columns 91 through 100

    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804    0.9804


best_vaf =

    0.9804


best_outpermone =

    18    20    12    10     5     8    13     3     9    17    15    19     1     7     6    11    16     4     2    14


best_outpermtwo =

    11    16     6     1     4     2    18    20    17     5     8    10     9    13    12    19     7     3    14    15


best_coordone =

   -0.3723
   -0.3723
   -0.2981
   -0.2981
   -0.2740
   -0.2408
   -0.2142
   -0.0673
   -0.0274
    0.0117
    0.0248
    0.0477
    0.0886
    0.1205
    0.2164
    0.2285
    0.2817
    0.3150
    0.3922
    0.4377


best_coordtwo =

   -0.3549
   -0.2578
   -0.2546
   -0.1900
   -0.1673
   -0.1315
   -0.0968
   -0.0362
   -0.0254
    0.0230
    0.0403
    0.0478
    0.0607
    0.1117
    0.1191
    0.1358
    0.1871
    0.2270
    0.2632
    0.2988
```

```
best_addconone =

   -0.2533


best_addcontwo =

    0.2167

Elapsed time is 502.951259 seconds.

sorted_vafs =

  Columns 1 through 15

    0.8762    0.8762    0.8762    0.8762    0.8762    0.8762    0.8762    0.8786    0.8786    0.8786    0.8786    0.8786    0.8786    0.8786    0.8786

  Columns 16 through 30

    0.8786    0.8786    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

  Columns 31 through 45

    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

  Columns 46 through 60

    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

  Columns 61 through 75

    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

  Columns 76 through 90

    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

  Columns 91 through 100

    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000


best_vaf =

    1.0000


best_coords =

   -0.2077    0.1939
   -0.5312   -0.0707
    0.2019   -0.3074
   -0.4789    0.0517
    0.3658    0.1177
   -0.4075    0.2504
   -0.0685   -0.2837
    0.3401    0.0620
    0.1055   -0.0645
    0.4221    0.0820
   -0.4892    0.3546
    0.4619   -0.0453
    0.3611   -0.0764
   -0.3935   -0.5326
    0.0853   -0.4179
   -0.5069    0.1888
   -0.0007    0.0385
    0.3081    0.3661
    0.0253   -0.1920
    0.4069    0.2851

Elapsed time is 7.523088 seconds.

ans =

    0.7230


ans =

   -0.9244    0.3813
   -0.3813   -0.9244


ans =
```

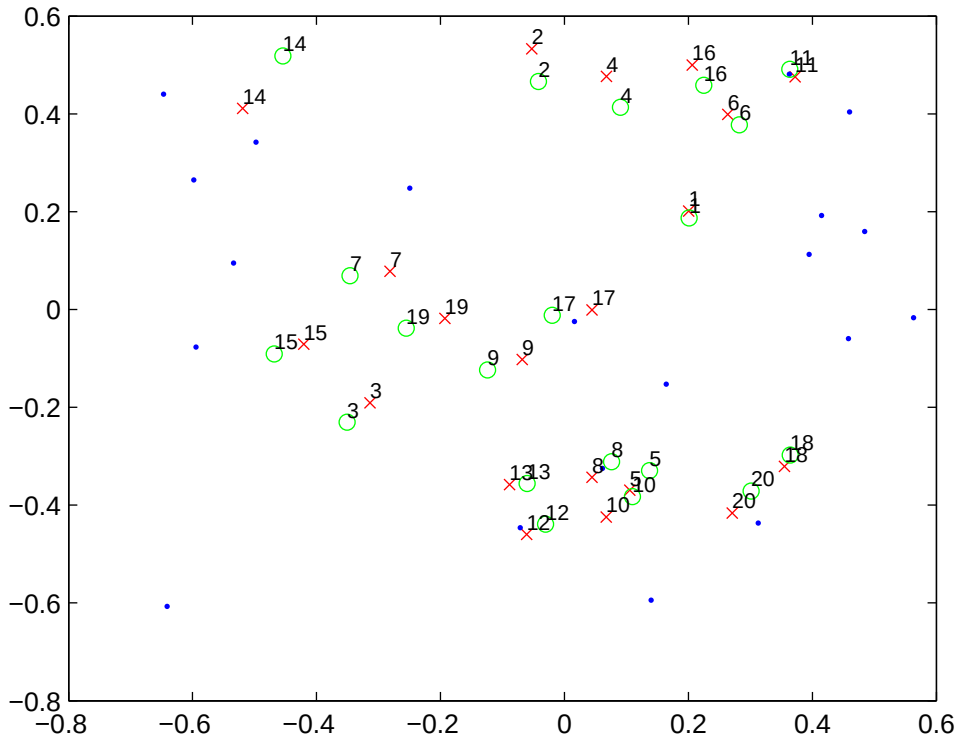


Figure 13: The City-Block and Euclidean Scalings of the (Perfect) City-Block Data (Distances): x: City-Block Scaling Coordinates; .: Euclidean Scaling Coordinates; o: Procrustes Transformed Euclidean Coordinates.

```
1.0e-016 *

  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
  -0.1003    0.0696
```

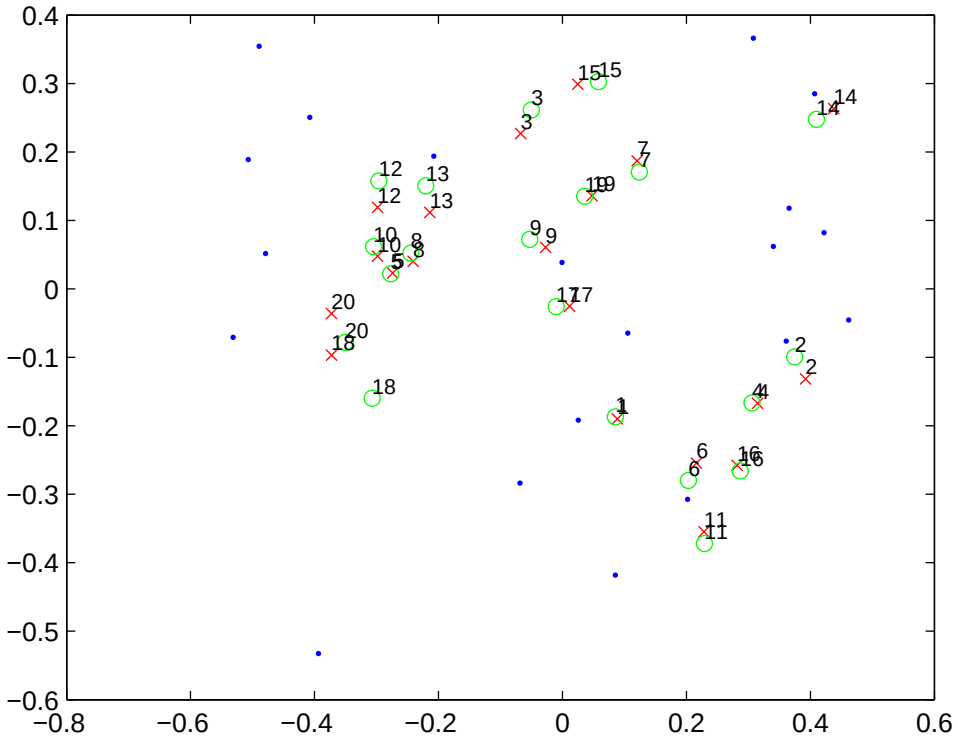


Figure 14: The City-Block and Euclidean Scalings of the (Perfect) Euclidean Data (Distances): x: City-Block Scaling Coordinates; .: Euclidean Scaling Coordinates; o: Procrustes Transformed Euclidean Coordinates.

5 Multifacility Location Routines

A multifacility location problem is usually stated in the following form: let $\mathbf{a}_1, \ldots, \mathbf{a}_m$ be m points (existing facilities) in d-dimensional space, and suppose $\mathbf{x}_1, \ldots, \mathbf{x}_n$ represent the to-be-found locations of n additional points (new facilities) in this same space. Two matrices containing nonnegative entries represent costs (and, thus, could be considered keyed as similarities): $\mathbf{W}_{n \times m} = \{w_{ji}\}$, the weights between a new facility j and an existing facility i, $1 \leq j \leq n$, $1 \leq i \leq m$; $\mathbf{V}_{n \times n} = \{v_{jk}\}$, the weights between new facilities j and k, $1 \leq j, k \leq n$. We wish to find the n locations, $\mathbf{x}_1, \ldots, \mathbf{x}_n$, to minimize the weighted costs:

$$\sum_{j=1}^{n} \sum_{i=1}^{m} w_{ji} d(\mathbf{x}_j, \mathbf{a}_i) + \sum_{j<k} v_{jk} d(\mathbf{x}_j, \mathbf{x}_k) \ ,$$

where $d(\cdot, \cdot)$ is some "distance-like" measure: Euclidean, city-block, or squared Euclidean.

When $d(\cdot, \cdot)$ has either a Euclidean or city-block specification, one standard solution method is to use what is called a hyperboloid approximation procedure (hap) (see Eyster, White, and Wierwille, 1973, for all the algorithmic details), which augments the distance measure by a small constant, ϵ (for us and in our scripts below, this is set at $1.0e{-}012$; by so doing, derivatives exist, when necessary, and the iterative optimization strategy can proceed without numerical anomaly. For the specification of $d(\cdot, \cdot)$ as squared-Euclidean, a closed-form solution exists (so no iterative process is needed, nor the hap approximation). The necessary formulas for this case are given in Eyster and White (1973).

Three M-file are available to carry out the multifacility location task:

`cityblock_multifacility_hap`, `euclidean_multifacility_hap.m`, and `squared_euclidean_multifacility.m`.

We provide the header help comments for `euclidean_multifacility_hap.m` below; those for `cityblock_multifacility_hap.m` are the same, and `squared_euclidea` excludes mention of `epsil` because a closed-form solution is provided:

```
% EUCLIDEAN_MULTIFACILITY_HAP carries out the Euclidean multifacility
% location problem using a hyperboloid approximation procedure.
```

```
%
% syntax: [new_facility_location,distance_new_to_old,...
%    distance_among_new,objective_function] = ...
%    euclidean_multifacility_hap(weight_new_to_old,...
%    weight_among_new,old_facility_location,epsil)
%
% There are n new facilities to place among m existing facilities in d
% dimensions.  WEIGHT_NEW_TO_OLD (n x m) and WEIGHT_AMONG_NEW (n x n) are
% both weight matrices (with similarity interpretations);
% OLD_FACILITY_LOCATION (d x m) contains the coordinates for the existing
% facilities; and EPSIL (e.g., 1.0e-012) produces the hyperboloid
% approximation allowing the objective function to be differentiable
% everywhere.
% As output, there are the NEW_FACILITY_LOCATIONS (d x n);
% DISTANCE_NEW_TO_OLD (n x m); DISTANCE_AMONG_NEW (n x n); and the
% OBJECTIVE_FUNCTION (i.e., the weighted sum of the distances among the new
% locations plus of the new locations to those existing).
```

5.1 A Data Analysis Example of Using the Multifacility Location Routines

To give one illustration of how the multifacility location routines might be used, the script given below first scales the sixteen concepts for Eve Black (including the one defined by the average) based on `ms_biscalqa.m` and the 16×16 proximity matrix defined by the Euclidean distance between the columns (augmented by the average concept) of the `eve_black_one` two-mode data matrix. This scaling serves as the locations of the existing facilities. The ten scales form the new facilities to be placed based on the similarities (8 - `eve_black_one`) between the new (rows) facilities and the old (columns). The weight matrix between the new facilities is set to all zeros. Figures 15, 16, and 17 show the embeddings of the ten scales for the various distance-like measures. One can see that the five scales of cold, tense, small, dirty, and weak, are close to the concepts of love, child, my job, mental sickness, my spouse, confusion, and sex. The remaining five scales of valuable, fast, tasty, deep, and active, are generally close to the concepts of my doctor, me, peace of mind, fraud, and my spouse. This is similar to our previous two-mode `biscaltmac.m` and `bimonscaltmac.m` results.

5.1.1 A Script Using the eve_black_one Data, ms_script_multifacility_twomode_sqeuclidean_biscalqa, for Embedding the Scales (Rows) into a Scaling Representation for the Concepts (Columns) using the Multifacility Location Routines

```
load eve_black_one.dat

tic;

best_vaf = 0.0;

store_vaf = zeros(100,1);

core = [4,4,4,4,4,4,4,4,4,4];

datamatrix = [eve_black_one,core'];

prox = zeros(16,16);

for i = 1:16
    for j=1:16

        for k = 1:10

            prox(i,j) = prox(i,j) + ((datamatrix(k,i) - datamatrix(k,j))^2);

        end

        prox(i,j) = sqrt(prox(i,j));

    end
end

prox


for i = 1:100

    [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,...
        addcontwo,vaf] = ms_biscalqa(prox,targlin(16),targlin(16), ...
        randperm(16),randperm(16),3,1);

    store_vaf(i) = vaf;

    if (vaf > best_vaf)

        best_vaf = vaf;
        best_outpermone = outpermone;
        best_outpermtwo = outpermtwo;
        best_coordone = coordone;
        best_coordtwo = coordtwo;
        best_fitone = fitone;
        best_fittwo = fittwo;
        best_addconone = addconone;
        best_addcontwo = addcontwo;

    end

end

sorted_vafs = sort(store_vaf');

sorted_vafs;

        best_vaf
        best_outpermone
        best_outpermtwo
        best_coordone
        best_coordtwo

        best_addconone
        best_addcontwo

        axes = zeros(16,2);

        for i = 1:16

            axes(best_outpermone(i),1) = best_coordone(i);
            axes(best_outpermtwo(i),2) = best_coordtwo(i);
        end

        figure(1)
```

```
plot(axes(1:16,1),axes(1:16,2),'ko')

hold on

for i = 1:16

    objectlabels{i,1} = int2str(i);

end

text(axes(1:16,1),axes(1:16,2),objectlabels,'fontsize',10,'verticalalignment','bottom')

weight_new_to_old = 8 - datamatrix;

old_facility_location = axes';

for i = 1:10

    newobjectlabels{i,1} = int2str(i);

end

[new_facility_location,distance_new_to_old,distance_among_new,objective_function] = ...
    cityblock_multifacility_hap(weight_new_to_old,zeros(10,10),old_facility_location,1.0e-012)


newaxes = new_facility_location';

plot(newaxes(1:10,1),newaxes(1:10,2),'rx')

text(newaxes(1:10,1),newaxes(1:10,2),newobjectlabels,'fontsize',10,'verticalalignment','bottom')

figure(2)

plot(axes(1:16,1),axes(1:16,2),'ko')

hold on

[new_facility_location,distance_new_to_old,distance_among_new,objective_function] = ...
    euclidean_multifacility_hap(weight_new_to_old,zeros(10,10),old_facility_location,1.0e-012)



text(axes(1:16,1),axes(1:16,2),objectlabels,'fontsize',10,'verticalalignment','bottom')



newaxes = new_facility_location';

plot(newaxes(1:10,1),newaxes(1:10,2),'rx')

text(newaxes(1:10,1),newaxes(1:10,2),newobjectlabels,'fontsize',10,'verticalalignment','bottom')

figure(3)

plot(axes(1:16,1),axes(1:16,2),'ko')

hold on

[new_facility_location,distance_new_to_old,distance_among_new,objective_function] = ...
    squared_euclidean_multifacility(weight_new_to_old,zeros(10,10),old_facility_location)



text(axes(1:16,1),axes(1:16,2),objectlabels,'fontsize',10,'verticalalignment','bottom')



newaxes = new_facility_location';

plot(newaxes(1:10,1),newaxes(1:10,2),'rx')

text(newaxes(1:10,1),newaxes(1:10,2),newobjectlabels,'fontsize',10,'verticalalignment','bottom')


toc;
```

5.1.2 The Results for the Script, ms_script_multifacility_twomode_sqeuclidean_biscalqa

```
>> ms_script_multifacility_twomode_sqeuclidean_biscalqa

prox =

  Columns 1 through 15

         0    9.5394   17.7834   17.1756    6.5000    8.0932   11.8004   17.9444   15.3704    6.1441   11.1018   16.2019   12.2882    6.5383    4.6904
    9.5394         0   15.9765   16.4317    8.0777    8.0932   10.6888   16.1555   13.7568    8.3516    8.8459   14.3353   15.8745    9.0416    9.0000
   17.7834   15.9765         0    3.0414   16.3248   15.5804   11.8954    0.5000    2.9155   14.1598    8.8882    1.9365   12.3794   13.6748   17.3277
   17.1756   16.4317    3.0414         0   16.5000   15.4758   11.7580    3.0000    3.9051   13.6657    9.1788    3.2404   11.2250   13.4443   16.7033
    6.5000    8.0777   16.3248   16.5000         0    7.4666    9.9247   16.5000   14.2127    5.6125   10.0747   14.7733   13.8293    7.5498    6.1847
    8.0932    8.0932   15.5804   15.4758    7.4666         0    9.6307   15.7639   13.1434    7.7621    8.0777   13.7659   14.2653    6.3443    5.9582
   11.8004   10.6888   11.8954   11.7580    9.9247    9.6307         0   12.1347   10.5119   10.1489    8.4853   10.4523   10.9659   10.0000    9.3408
   17.9444   16.1555    0.5000    3.0000   16.5000   15.7639   12.1347         0    3.0414   14.3091    9.0139    2.1213   12.3693   13.7750   17.4929
   15.3704   13.7568    2.9155    3.9051   14.2127   13.1434   10.5119    3.0414         0   12.0416    6.6708    1.5000   11.1018   10.9772   15.1079
    6.1441    8.3516   14.1598   13.6657    5.6125    7.7621   10.1489   14.3091   12.0416         0    8.0000   12.6392   12.6392    6.5955    6.5383
   11.1018    8.8459    8.8882    9.1788   10.0747    8.0777    8.4853    9.0139    6.6708    8.0000         0    7.2629   10.9659    6.8191   10.1119
   16.2019   14.3353    1.9365    3.2404   14.7733   13.7659   10.4523    2.1213    1.5000   12.6392    7.2629         0   11.5542   12.0312   15.6365
   12.2882   15.8745   12.3794   11.2250   13.8293   14.2653   10.9659   12.3693   11.1018   12.6392   10.9659   11.5542         0   10.5712   12.3693
    6.5383    9.0416   13.6748   13.4443    7.5498    6.3443   10.0000   13.7750   10.9772    6.5955    6.8191   12.0312   10.5712         0    7.1937
    4.6904    9.0000   17.3277   16.7033    6.1847    5.9582    9.3408   17.4929   15.1079    6.5383   10.1119   15.6365   12.3693    7.1937         0
    8.7178    9.0000    9.3408    9.0000    8.0777    7.9057    7.3655    9.4868    6.9462    5.8095    4.2720    7.6485    9.0000    5.5453    8.4853

  Column 16

    8.7178
    9.0000
    9.3408
    9.0000
    8.0777
    7.9057
    7.3655
    9.4868
    6.9462
    5.8095
    4.2720
    7.6485
    9.0000
    5.5453
    8.4853
         0


best_vaf =

    0.9441


best_outpermone =

     3     8     4    12     9    13    11    16     7    14    10     1    15     6     5     2


best_outpermtwo =

    13     1    15    14     5    10    16     6    11     9     2    12     8     3     4     7


best_coordone =

   -6.2144
   -6.2144
   -6.0620
   -5.1142
   -4.7814
   -2.2748
   -0.3474
    0.0118
    0.9474
    1.8426
    3.0818
    3.9629
    4.2927
    4.7835
    5.3291
    6.7569


best_coordtwo =

   -4.3195
   -2.9674
```

```
   -2.1531
   -1.2480
   -0.8006
   -0.5333
    0.2227
    0.3731
    0.9795
    1.1379
    1.1379
    1.2318
    1.2318
    1.2318
    1.2727
    3.2028


best_addconone =

   -4.9030


best_addcontwo =

    2.1120


new_facility_location =

    3.0818   -4.7814    3.0817    1.8426   -1.3547    3.0818    3.0818   -4.7814   -4.7814   -4.7814
    0.2227    1.2318   -0.5333   -0.5333    1.1379    0.2227   -0.8006    1.1379    1.1379    1.2318


distance_new_to_old =

  Columns 1 through 15

    4.0713    4.5903   10.3052   10.1938    3.2706    1.8521    5.1144   10.3052    8.7783    0.7560    4.1859    9.2050    9.8988    2.7099    3.5868
   12.9434   11.6321    1.4330    1.3216   12.1428   10.4235    7.6999    1.4330    0.0939    9.6282    4.6863    0.3329    8.0578    9.1037   12.4590
    3.3153    5.3464   11.0612   10.9498    2.5147    2.6082    5.8704   11.0612    9.5343    0.0000    4.9419    9.9610    9.1428    1.9539    2.8308
    4.5544    6.5855    9.8220    9.7106    3.7538    3.8474    4.6312    9.8220    8.2951    1.2392    3.7027    8.7219    7.9036    0.7147    4.0699
    9.4229    8.1116    4.9535    4.8421    8.6223    6.9030    4.3671    4.9535    3.4267    6.1077    1.1658    3.8534    6.3775    5.5832    8.9384
    4.0713    4.5903   10.3052   10.1938    3.2706    1.8521    5.1144   10.3052    8.7783    0.7560    4.1859    9.2050    9.8988    2.7099    3.5868
    3.0479    5.6136   11.3285   11.2171    2.2473    2.8754    6.1377   11.3285    9.8016    0.2673    5.2092   10.2283    8.8755    1.6866    2.5635
   12.8496   11.5383    1.5269    1.4155   12.0490   10.3297    7.7937    1.5269    0.0000    9.5343    4.5924    0.4267    7.9640    9.0098   12.3651
   12.8496   11.5383    1.5269    1.4155   12.0490   10.3297    7.7937    1.5269    0.0000    9.5343    4.5924    0.4267    7.9640    9.0098   12.3651
   12.9434   11.6321    1.4330    1.3216   12.1428   10.4235    7.6999    1.4330    0.0939    9.6282    4.6863    0.3329    8.0578    9.1037   12.4590

  Column 16

    3.0699
    5.8023
    3.8259
    2.5867
    2.2817
    3.0699
    4.0932
    5.7084
    5.7084
    5.8023


distance_among_new =

         0    8.8722    0.7561    1.9952    5.3516    0.0000    1.0233    8.7783    8.7783    8.8722
    8.8722         0    9.6282    8.3890    3.5205    8.8722    9.8955    0.0939    0.0939    0.0000
    0.7561    9.6282         0    1.2392    6.1076    0.7561    0.2673    9.5343    9.5343    9.6282
    1.9952    8.3890    1.2392         0    4.8685    1.9952    1.5065    8.2951    8.2951    8.3890
    5.3516    3.5205    6.1076    4.8685         0    5.3516    6.3749    3.4267    3.4267    3.5205
    0.0000    8.8722    0.7561    1.9952    5.3516         0    1.0233    8.7783    8.7783    8.8722
    1.0233    9.8955    0.2673    1.5065    6.3749    1.0233         0    9.8016    9.8016    9.8955
    8.7783    0.0939    9.5343    8.2951    3.4267    8.7783    9.8016         0    0.0000    0.0939
    8.7783    0.0939    9.5343    8.2951    3.4267    8.7783    9.8016    0.0000         0    0.0939
    8.8722    0.0000    9.6282    8.3890    3.5205    8.8722    9.8955    0.0939    0.0939         0


objective_function =

  2.8025e+003


new_facility_location =

    3.0877   -4.8914    2.5753    2.0453   -1.4583    3.0802    2.6117   -4.8025   -4.7815   -4.7821
   -0.5262    1.1027   -0.8215   -1.0655    0.7434   -0.5359   -0.9779    1.0958    1.1374    1.1336


distance_new_to_old =
```

```
  Columns 1 through 15

    2.5933    4.0290    9.4667    9.3249    2.2582    1.9195    4.2996    9.4667    8.0431    0.0092    3.7506    8.3882    6.5685    1.4392    2.0246
    9.7449   11.6483    1.3293    1.1829   10.3962    9.7023    6.2050    1.3293    0.1155    8.1392    4.5457    0.2575    6.0205    7.1324    9.7441
    2.5554    4.6178    9.0263    8.8876    2.7538    2.5106    4.3411    9.0263    7.6132    0.5827    3.4330    7.9590    5.9800    0.8479    2.1731
    2.7008    5.2014    8.5732    8.4377    3.2945    3.0932    4.4073    8.5732    7.1734    1.1652    3.1475    7.5190    5.4085    0.2727    2.4968
    6.5696    8.2247    4.7810    4.6340    6.9608    6.2528    3.4404    4.7810    3.3464    4.7162    1.1357    3.6884    5.1284    3.8551    6.4393
    2.5868    4.0397    9.4612    9.3194    2.2644    1.9306    4.3043    9.4612    8.0378    0.0030    3.7477    8.3830    6.5569    1.4279    2.0213
    2.4050    4.6539    9.0985    8.9610    2.7231    2.5576    4.4998    9.0985    7.6899    0.6469    3.5479    8.0357    5.9199    0.8153    2.0511
    9.6613   11.5595    1.4184    1.2719   10.3076    9.6132    6.1239    1.4184    0.0472    8.0508    4.4567    0.3401    5.9761    7.0463    9.6581
    9.6599   11.5384    1.4360    1.2876   10.2947    9.5955    6.0899    1.4360    0.0005    8.0388    4.4369    0.3458    6.0051    7.0405    9.6524
    9.6588   11.5390    1.4356    1.2875   10.2945    9.5958    6.0917    1.4356    0.0044    8.0386    4.4374    0.3463    6.0019    7.0398    9.6517

  Column 16

    3.1657
    4.9816
    2.7680
    2.4071
    1.5597
    3.1608
    2.8637
    4.8929
    4.8799
    4.8797


distance_among_new =

         0    8.1436    0.5913    1.1737    4.7200    0.0122    0.6561    8.0552    8.0431    8.0429
    8.1436         0    7.7107    7.2676    3.4518    8.1383    7.7862    0.0891    0.1152    0.1135
    0.5913    7.7107         0    0.5836    4.3266    0.5801    0.1606    7.6229    7.6132    7.6128
    1.1737    7.2676    0.5836         0    3.9430    1.1626    0.5732    7.1808    7.1734    7.1728
    4.7200    3.4518    4.3266    3.9430         0    4.7154    4.4191    3.3627    3.3464    3.3466
    0.0122    8.1383    0.5801    1.1626    4.7154         0    0.6441    8.0499    8.0379    8.0376
    0.6561    7.7862    0.1606    0.5732    4.4191    0.6441         0    7.6988    7.6899    7.6894
    8.0552    0.0891    7.6229    7.1808    3.3627    8.0499    7.6988         0    0.0467    0.0430
    8.0431    0.1152    7.6132    7.1734    3.3464    8.0379    7.6899    0.0467         0    0.0039
    8.0429    0.1135    7.6128    7.1728    3.3466    8.0376    7.6894    0.0430    0.0039         0


objective_function =

  2.2928e+003


new_facility_location =

    2.4071   -2.9930    1.8558    1.0004   -1.3547    2.1513    1.7319   -2.6776   -2.2250   -2.5747
    0.0219    0.3434   -0.6142   -0.8177    0.5710   -0.3001   -0.8419    0.1204    0.5934    0.4294


distance_new_to_old =

  Columns 1 through 15

   11.3564   20.1663   75.7932   73.2902    9.2146    5.7707   12.2489   75.7932   52.9195    0.7634    8.5040   58.0339   40.7682    1.9313    8.2862
   59.3462   95.6923   11.1661   10.2823   70.5665   60.4754   23.7037   11.1661    3.8294   37.6718    7.4041    5.2886   22.2585   25.9156   59.3146
    9.9775   27.0907   68.5347   66.2518   12.0986    9.5463   15.3946   68.5348   47.1216    1.5096    7.3936   51.9885   30.7915    0.4019    8.3069
   13.3979   36.9617   56.2525   54.2469   18.7381   15.7300   16.1670   56.2525   37.2528    4.4130    5.0461   41.5884   22.9900    0.8945   12.6229
   40.7967   66.1194   24.0530   22.6512   46.5543   37.7166   12.2266   24.0530   12.0635   20.9016    1.1816   14.5707   24.7634   13.5311   39.3138
   10.3959   23.2790   72.3320   69.9333   10.3485    7.3815   13.7202   72.3320   50.1307    0.9200    7.8810   55.1354   35.7465    0.9938    8.0189
    9.4951   29.1705   67.4428   65.2163   12.9417   10.7887   16.9752   67.4428   46.3420    1.9175    7.6407   51.1692   28.1474    0.1772    8.2771
   53.6305   90.0447   13.7439   12.7821   64.9552   55.7316   22.6420   13.7439    5.4612   33.5974    6.1679    7.1723   19.8752   22.3043   53.7540
   50.9689   80.9702   16.3228   15.1844   59.0070   49.1671   16.8732   16.3228    6.8317   29.4308    3.6744    8.7553   24.1394   19.9356   50.0235
   54.2774   87.5797   13.8914   12.8729   63.9819   54.1455   20.0972   13.8914    5.3717   32.9217    5.2634    7.0933   22.6418   22.3253   53.8298

  Column 16

    5.7775
    9.0440
    4.1005
    2.0596
    1.9888
    4.8508
    4.0919
    7.2436
    5.1408
    6.7327


distance_among_new =

         0   29.2647    0.7085    2.6837   14.4525    0.1691    1.2021   25.8636   21.7825   24.9837
   29.2647         0   24.4280   17.2953    2.7359   26.8789   23.7297    0.1492    0.6524    0.1824
    0.7085   24.4280         0    0.7731   11.7118    0.1860    0.0672   21.0911   18.1108   20.7177
```

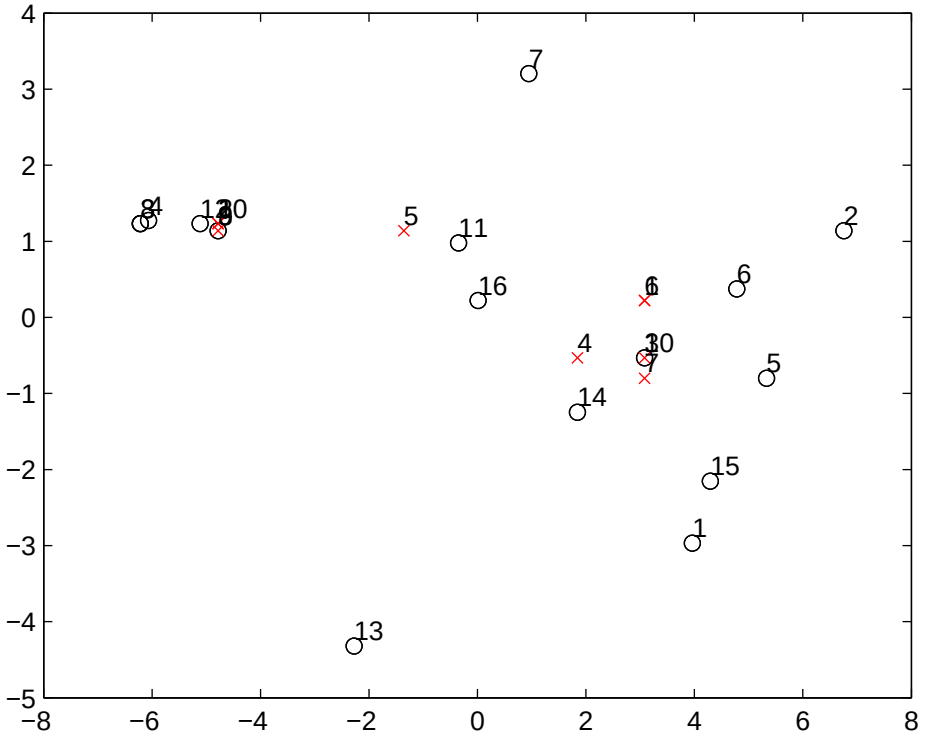


Figure 15: Multifacility (City-Block) Embedding of Scales for Eve Black: x: Embedded Scales; o: Scaled Concepts.

```
    2.6837   17.2953    0.7731         0    7.4746    1.5926    0.5357   14.4074   12.3939   14.3358
   14.4525    2.7359   11.7118    7.4746         0   13.0512   11.5231    1.9530    0.7579    1.5083
    0.1691   26.8789    0.1860    1.5926   13.0512         0    0.4695   23.4955   19.9506   22.8673
    1.2021   23.7297    0.0672    0.5357   11.5231    0.4695         0   20.3694   17.7167   20.1623
   25.8636    0.1492   21.0911   14.4074    1.9530   23.4955   20.3694         0    0.4286    0.1060
   21.7825    0.6524   18.1108   12.3939    0.7579   19.9506   17.7167    0.4286         0    0.1492
   24.9837    0.1824   20.7177   14.3358    1.5083   22.8673   20.1623    0.1060    0.1492         0

objective_function =

  1.1164e+004

Elapsed time is 355.969496 seconds.
```

6 The Confirmatory Fitting of Tied Coordinate Patterns in Bidimensional City-Block Scaling

One of the options in the M-file, `biscalqa.m`, allows the confirmatory fitting of two given input permutations by setting the `nopt` switch to zero. An

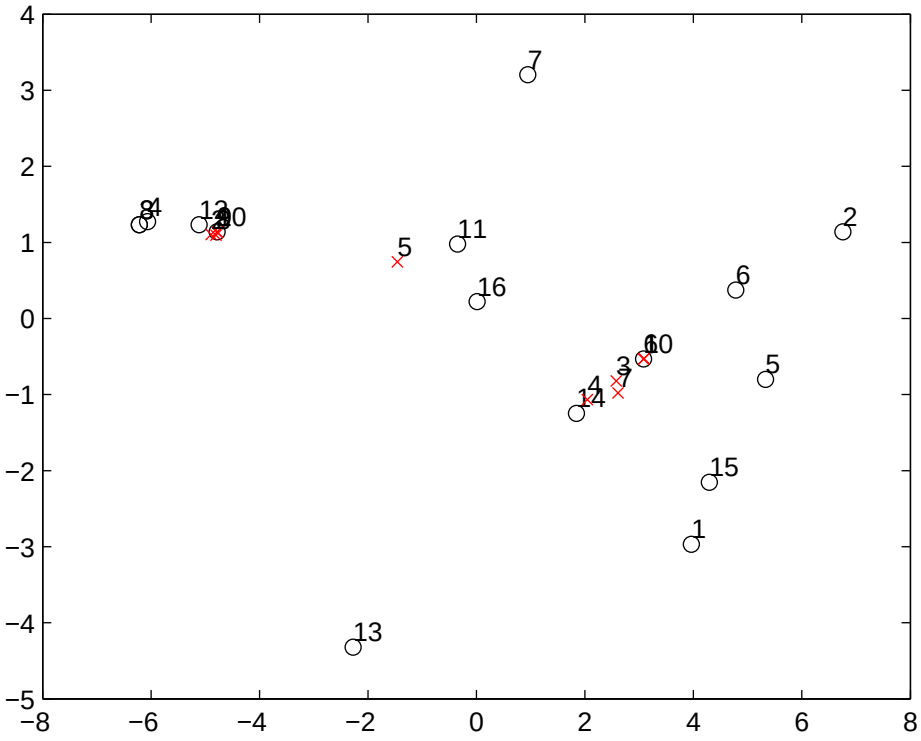


Figure 16: Multifacility (Euclidean) Embedding of Scales for Eve Black: x: Embedded Scales; o: Scaled Concepts.

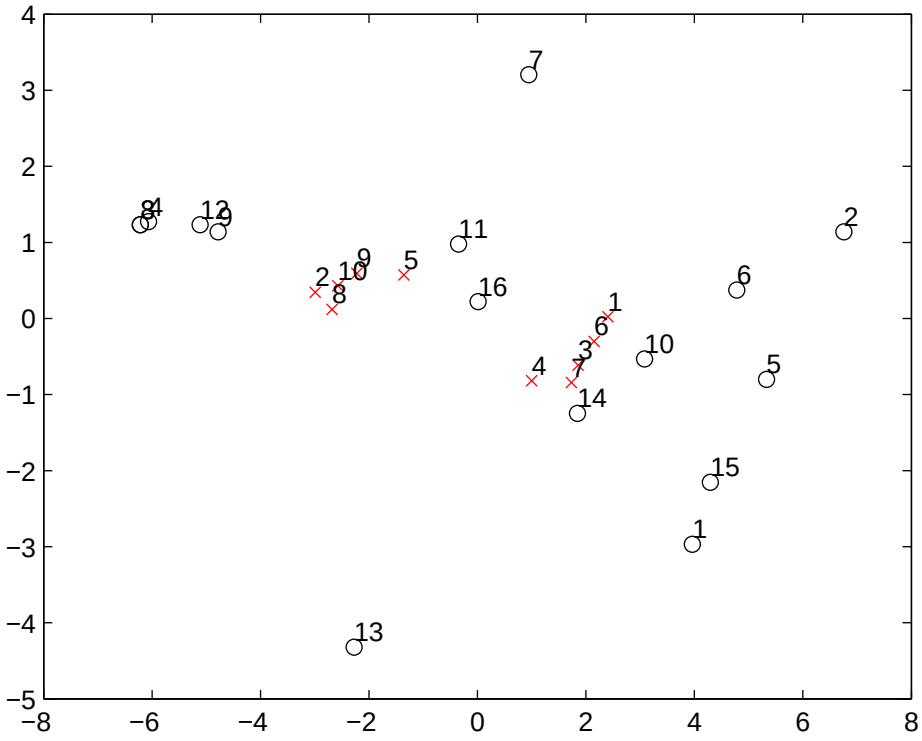


Figure 17: Multifacility (Squared-Euclidean) Embedding of Scales for Eve Black: x: Embedded Scales; o: Scaled Concepts.

extension of this M-file to `biscalqa_tied.m` is presented here that further allows the imposition of given patterns of tied coordinates along the two axes. Equal integer values that are present in `tiedcoordone` and `tiedcoordtwo` impose equal coordinate values in the optimization process.

We give two examples below of using `biscalqa_tied.m` on the Hampshire proximities. The first illustration does a confirmatory fitting of the two object permutations that led to the best VAF identified earlier of .9499 and with no imposed tied coordinates (so both `tiedcoordone` and `tiedcoordtwo` consist of the integers from 1 to 27). The exact same VAF is obtained in the confirmatory fit, but interestingly, there are slight variations in the additive constants and the coordinates at the level of the third or fourth decimal places. So we have small variations that apparently "cancel" and lead to the same best VAF as before. The second illustration imposes tied coordinates along both axes in groups of threes, so `tiedcoordone` and `tiedcoordtwo` both have the pattern of `[1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6 7 7 7 8 8 8 9 9 9]`. There is an expected drop in the VAF to .8734 (but not too severe given the major reduction in model complexity). The resulting configuration of Hampshire towns with these tied coordinate structures imposed is given in Figure 18.

```
load hampshire_proximities.dat

inpermone = [2 6 12 19 22 1 11 4 5 23 21 18 3 27 20 24 17 10 25 9 26 13 8 16 7 15 14];
inpermtwo = [18 4 19 16 3 11 15 17 5 14 27 7 8 13 12 21 20 6 2 22 9 10 25 23 1 26 24];

[outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,addcontwo,vaf] = ...
    biscalqa_tied(hampshire_proximities,targlin(27),targlin(27),inpermone,inpermtwo,1:27,1:27,3,0)

tiedcoordone = [1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6 7 7 7 8 8 8 9 9 9];

 tiedcoordtwo = [1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6 7 7 7 8 8 8 9 9 9];

[outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,addconone,addcontwo,vaf] = ...
    biscalqa_tied(hampshire_proximities,targlin(27),targlin(27),inpermone,inpermtwo,...
    tiedcoordone,tiedcoordtwo,3,0)

    axes = zeros(27,2);

        for i = 1:27

            axes(outpermone(i),1) = coordone(i);
            axes(outpermtwo(i),2) = coordtwo(i);
        end

        plot(axes(1:27,1),axes(1:27,2),'ko')

        hold on

        for i = 1:27

            objectlabels{i,1} = int2str(i);

        end

        text(axes(1:27,1),axes(1:27,2),objectlabels,'fontsize',10,'verticalalignment','bottom')


>> script_tied_coordinates_biscalqa
```

```
outpermone =

  Columns 1 through 26

     2     6    12    19    22     1    11     4     5    23    21    18     3    27    20    24    17    10    25     9    26    13     8    16     7    15

  Column 27

    14


outpermtwo =

  Columns 1 through 26

    18     4    19    16     3    11    15    17     5    14    27     7     8    13    12    21    20     6     2    22     9    10    25    23     1    26

  Column 27

    24


coordone =

  -18.7759
  -14.5533
  -13.3525
  -12.7337
  -10.7167
  -10.4930
   -9.1591
   -7.5284
   -6.3342
   -5.0568
   -4.5961
   -3.9200
   -0.5998
   -0.5998
    1.0959
    2.9431
    4.1361
    4.3892
    5.0837
    6.1704
    6.7689
    7.1508
   12.0935
   13.8611
   15.3806
   17.2922
   22.0538


coordtwo =

   -7.1706
   -5.9026
   -5.3941
   -4.9316
   -4.9034
   -4.3384
   -3.5486
   -3.5486
   -2.3517
   -2.1241
   -1.8610
   -0.8060
   -0.7654
   -0.6970
   -0.0768
    0.8199
    0.9487
    1.8872
    2.3601
    2.7219
    3.3431
    3.5634
    4.0580
    5.9584
    6.8936
    7.4461
    8.4196


fitone =

  Columns 1 through 15
```

```
       0    4.2226    5.4234    6.0422    8.0592    8.2830    9.6168   11.2475   12.4418   13.7191   14.1799   14.8559   18.1761   18.1761   19.8719
  4.2226         0    1.2008    1.8196    3.8366    4.0604    5.3942    7.0249    8.2192    9.4965    9.9573   10.6333   13.9535   13.9535   15.6493
  5.4234    1.2008         0    0.6188    2.6358    2.8595    4.1934    5.8241    7.0183    8.2957    8.7564    9.4325   12.7527   12.7527   14.4484
  6.0422    1.8196    0.6188         0    2.0170    2.2407    3.5746    5.2053    6.3995    7.6769    8.1376    8.8137   12.1339   12.1339   13.8296
  8.0592    3.8366    2.6358    2.0170         0    0.2237    1.5576    3.1883    4.3825    5.6599    6.1206    6.7967   10.1169   10.1169   11.8126
  8.2830    4.0604    2.8595    2.2407    0.2237         0    1.3339    2.9645    4.1588    5.4361    5.8969    6.5730    9.8932    9.8932   11.5889
  9.6168    5.3942    4.1934    3.5746    1.5576    1.3339         0    1.6307    2.8249    4.1023    4.5631    5.2391    8.5593    8.5593   10.2550
 11.2475    7.0249    5.8241    5.2053    3.1883    2.9645    1.6307         0    1.1943    2.4716    2.9324    3.6084    6.9286    6.9286    8.6243
 12.4418    8.2192    7.0183    6.3995    4.3825    4.1588    2.8249    1.1943         0    1.2773    1.7381    2.4142    5.7344    5.7344    7.4301
 13.7191    9.4965    8.2957    7.6769    5.6599    5.4361    4.1023    2.4716    1.2773         0    0.4608    1.1368    4.4570    4.4570    6.1527
 14.1799    9.9573    8.7564    8.1376    6.1206    5.8969    4.5631    2.9324    1.7381    0.4608         0    0.6761    3.9963    3.9963    5.6920
 14.8559   10.6333    9.4325    8.8137    6.7967    6.5730    5.2391    3.6084    2.4142    1.1368    0.6761         0    3.3202    3.3202    5.0159
 18.1761   13.9535   12.7527   12.1339   10.1169    9.8932    8.5593    6.9286    5.7344    4.4570    3.9963    3.3202         0         0    1.6957
 18.1761   13.9535   12.7527   12.1339   10.1169    9.8932    8.5593    6.9286    5.7344    4.4570    3.9963    3.3202         0         0    1.6957
 19.8719   15.6493   14.4484   13.8296   11.8126   11.5889   10.2550    8.6243    7.4301    6.1527    5.6920    5.0159    1.6957    1.6957         0
 21.7191   17.4965   16.2956   15.6768   13.6598   13.4361   12.1022   10.4715    9.2773    7.9999    7.5392    6.8631    3.5429    3.5429    1.8472
 22.9120   18.6894   17.4885   16.8697   14.8528   14.6290   13.2952   11.6645   10.4702    9.1929    8.7321    8.0560    4.7358    4.7358    3.0401
 23.1651   18.9425   17.7416   17.1228   15.1059   14.8821   13.5483   11.9176   10.7233    9.4460    8.9852    8.3091    4.9889    4.9889    3.2932
 23.8596   19.6370   18.4362   17.8173   15.8004   15.5766   14.2428   12.6121   11.4178   10.1405    9.6797    9.0036    5.6835    5.6835    3.9877
 24.9464   20.7238   19.5229   18.9041   16.8871   16.6634   15.3295   13.6989   12.5046   11.2273   10.7665   10.0904    6.7702    6.7702    5.0745
 25.5448   21.3222   20.1214   19.5026   17.4856   17.2618   15.9280   14.2973   13.1030   11.8257   11.3649   10.6889    7.3687    7.3687    5.6729
 25.9267   21.7041   20.5033   19.8845   17.8675   17.6438   16.3099   14.6792   13.4850   12.2076   11.7468   11.0708    7.7506    7.7506    6.0549
 30.8695   26.6469   25.4460   24.8272   22.8102   22.5865   21.2526   19.6219   18.4277   17.1503   16.6896   16.0135   12.6933   12.6933   10.9976
 32.6370   28.4144   27.2136   26.5948   24.5778   24.3540   23.0202   21.3895   20.1952   18.9179   18.4571   17.7811   14.4609   14.4609   12.7651
 34.1565   29.9339   28.7330   28.1142   26.0972   25.8735   24.5397   22.9090   21.7147   20.4374   19.9766   19.3005   15.9803   15.9803   14.2846
 36.0681   31.8455   30.6447   30.0258   28.0089   27.7851   26.4513   24.8206   23.6263   22.3490   21.8882   21.2121   17.8919   17.8919   16.1962
 40.8297   36.6071   35.4063   34.7875   32.7705   32.5468   31.2129   29.5822   28.3880   27.1106   26.6499   25.9738   22.6536   22.6536   20.9579

  Columns 16 through 27

  21.7191   22.9120   23.1651   23.8596   24.9464   25.5448   25.9267   30.8695   32.6370   34.1565   36.0681   40.8297
  17.4965   18.6894   18.9425   19.6370   20.7238   21.3222   21.7041   26.6469   28.4144   29.9339   31.8455   36.6071
  16.2956   17.4885   17.7416   18.4362   19.5229   20.1214   20.5033   25.4460   27.2136   28.7330   30.6447   35.4063
  15.6768   16.8697   17.1228   17.8173   18.9041   19.5026   19.8845   24.8272   26.5948   28.1142   30.0258   34.7875
  13.6598   14.8528   15.1059   15.8004   16.8871   17.4856   17.8675   22.8102   24.5778   26.0972   28.0089   32.7705
  13.4361   14.6290   14.8821   15.5766   16.6634   17.2618   17.6438   22.5865   24.3540   25.8735   27.7851   32.5468
  12.1022   13.2952   13.5483   14.2428   15.3295   15.9280   16.3099   21.2526   23.0202   24.5397   26.4513   31.2129
  10.4715   11.6645   11.9176   12.6121   13.6989   14.2973   14.6792   19.6219   21.3895   22.9090   24.8206   29.5822
   9.2773   10.4702   10.7233   11.4178   12.5046   13.1030   13.4850   18.4277   20.1952   21.7147   23.6263   28.3880
   7.9999    9.1929    9.4460   10.1405   11.2273   11.8257   12.2076   17.1503   18.9179   20.4374   22.3490   27.1106
   7.5392    8.7321    8.9852    9.6797   10.7665   11.3649   11.7468   16.6896   18.4571   19.9766   21.8882   26.6499
   6.8631    8.0560    8.3091    9.0036   10.0904   10.6889   11.0708   16.0135   17.7811   19.3005   21.2121   25.9738
   3.5429    4.7358    4.9889    5.6835    6.7702    7.3687    7.7506   12.6933   14.4609   15.9803   17.8919   22.6536
   3.5429    4.7358    4.9889    5.6835    6.7702    7.3687    7.7506   12.6933   14.4609   15.9803   17.8919   22.6536
   1.8472    3.0401    3.2932    3.9877    5.0745    5.6729    6.0549   10.9976   12.7651   14.2846   16.1962   20.9579
        0    1.1929    1.4460    2.1405    3.2273    3.8257    4.2077    9.1504   10.9179   12.4374   14.3490   19.1107
   1.1929         0    0.2531    0.9476    2.0344    2.6328    3.0147    7.9575    9.7250   11.2445   13.1561   17.9178
   1.4460    0.2531         0    0.6945    1.7813    2.3797    2.7616    7.7044    9.4719   10.9914   12.9030   17.6647
   2.1405    0.9476    0.6945         0    1.0868    1.6852    2.0671    7.0099    8.7774   10.2969   12.2085   16.9702
   3.2273    2.0344    1.7813    1.0868         0    0.5984    0.9804    5.9231    7.6906    9.2101   11.1217   15.8834
   3.8257    2.6328    2.3797    1.6852    0.5984         0    0.3819    5.3246    7.0922    8.6117   10.5233   15.2849
   4.2077    3.0147    2.7616    2.0671    0.9804    0.3819         0    4.9427    6.7103    8.2298   10.1414   14.9030
   9.1504    7.9575    7.7044    7.0099    5.9231    5.3246    4.9427         0    1.7676    3.2870    5.1986    9.9603
  10.9179    9.7250    9.4719    8.7774    7.6906    7.0922    6.7103    1.7676         0    1.5195    3.4311    8.1927
  12.4374   11.2445   10.9914   10.2969    9.2101    8.6117    8.2298    3.2870    1.5195         0    1.9116    6.6733
  14.3490   13.1561   12.9030   12.2085   11.1217   10.5233   10.1414    5.1986    3.4311    1.9116         0    4.7617
  19.1107   17.9178   17.6647   16.9702   15.8834   15.2849   14.9030    9.9603    8.1927    6.6733    4.7617         0


fittwo =

  Columns 1 through 15

       0    1.2681    1.7765    2.2390    2.2672    2.8322    3.6220    3.6220    4.8189    5.0465    5.3097    6.3646    6.4053    6.4736    7.0938
  1.2681         0    0.5084    0.9710    0.9992    1.5641    2.3540    2.3540    3.5509    3.7785    4.0416    5.0966    5.1372    5.2056    5.8258
  1.7765    0.5084         0    0.4625    0.4908    1.0557    1.8455    1.8455    3.0424    3.2700    3.5332    4.5881    4.6288    4.6971    5.3173
  2.2390    0.9710    0.4625         0    0.0282    0.5932    1.3830    1.3830    2.5799    2.8075    3.0706    4.1256    4.1663    4.2346    4.8548
  2.2672    0.9992    0.4908    0.0282         0    0.5649    1.3548    1.3548    2.5517    2.7793    3.0424    4.0974    4.1380    4.2064    4.8266
  2.8322    1.5641    1.0557    0.5932    0.5649         0    0.7898    0.7898    1.9867    2.2143    2.4775    3.5324    3.5731    3.6414    4.2616
  3.6220    2.3540    1.8455    1.3830    1.3548    0.7898         0         0    1.1969    1.4245    1.6876    2.7426    2.7832    2.8516    3.4718
  3.6220    2.3540    1.8455    1.3830    1.3548    0.7898         0         0    1.1969    1.4245    1.6876    2.7426    2.7832    2.8516    3.4718
  4.8189    3.5509    3.0424    2.5799    2.5517    1.9867    1.1969    1.1969         0    0.2276    0.4907    1.5457    1.5864    1.6547    2.2749
  5.0465    3.7785    3.2700    2.8075    2.7793    2.2143    1.4245    1.4245    0.2276         0    0.2631    1.3181    1.3587    1.4271    2.0473
  5.3097    4.0416    3.5332    3.0706    3.0424    2.4775    1.6876    1.6876    0.4907    0.2631         0    1.0550    1.0956    1.1640    1.7842
  6.3646    5.0966    4.5881    4.1256    4.0974    3.5324    2.7426    2.7426    1.5457    1.3181    1.0550         0    0.0407    0.1090    0.7292
  6.4053    5.1372    4.6288    4.1663    4.1380    3.5731    2.7832    2.7832    1.5864    1.3587    1.0956    0.0407         0    0.0684    0.6885
  6.4736    5.2056    4.6971    4.2346    4.2064    3.6414    2.8516    2.8516    1.6547    1.4271    1.1640    0.1090    0.0684         0    0.6202
  7.0938    5.8258    5.3173    4.8548    4.8266    4.2616    3.4718    3.4718    2.2749    2.0473    1.7842    0.7292    0.6885    0.6202         0
  7.9906    6.7225    6.2141    5.7516    5.7233    5.1584    4.3685    4.3685    3.1717    2.9440    2.6809    1.6259    1.5853    1.5169    0.8968
  8.1194    6.8513    6.3429    5.8803    5.8521    5.2872    4.4973    4.4973    3.3004    3.0728    2.8097    1.7547    1.7141    1.6457    1.0255
  9.0578    7.7898    7.2813    6.8188    6.7906    6.2257    5.4358    5.4358    4.2389    4.0113    3.7482    2.6932    2.6526    2.5842    1.9640
  9.5307    8.2626    7.7542    7.2917    7.2635    6.6985    5.9087    5.9087    4.7118    4.4842    4.2210    3.1661    3.1254    3.0571    2.4369
  9.8926    8.6245    8.1161    7.6535    7.6253    7.0604    6.2705    6.2705    5.0736    4.8460    4.5829    3.5279    3.4873    3.4189    2.7987
 10.5137    9.2456    8.7372    8.2747    8.2465    7.6815    6.8917    6.8917    5.6948    5.4672    5.2040    4.1491    4.1084    4.0401    3.4199
 10.7340    9.4660    8.9575    8.4950    8.4668    7.9018    7.1120    7.1120    5.9151    5.6875    5.4244    4.3694    4.3288    4.2604    3.6402
 11.2286    9.9606    9.4521    8.9896    8.9614    8.3964    7.6066    7.6066    6.4097    6.1821    5.9190    4.8640    4.8234    4.7550    4.1348
```

```
   13.1291   11.8610   11.3526   10.8900   10.8618   10.2969    9.5070    9.5070    8.3101    8.0825    7.8194    6.7644    6.7238    6.6554    6.0352
   14.0642   12.7961   12.2877   11.8252   11.7969   11.2320   10.4422   10.4422    9.2453    9.0177    8.7545    7.6996    7.6589    7.5906    6.9704
   14.6168   13.3487   12.8403   12.3777   12.3495   11.7846   10.9947   10.9947    9.7978    9.5702    9.3071    8.2521    8.2115    8.1431    7.5229
   15.5903   14.3222   13.8138   13.3513   13.3230   12.7581   11.9683   11.9683   10.7714   10.5438   10.2806    9.2257    9.1850    9.1166    8.4965

  Columns 16 through 27

    7.9906    8.1194    9.0578    9.5307    9.8926   10.5137   10.7340   11.2286   13.1291   14.0642   14.6168   15.5903
    6.7225    6.8513    7.7898    8.2626    8.6245    9.2456    9.4660    9.9606   11.8610   12.7961   13.3487   14.3222
    6.2141    6.3429    7.2813    7.7542    8.1161    8.7372    8.9575    9.4521   11.3526   12.2877   12.8403   13.8138
    5.7516    5.8803    6.8188    7.2917    7.6535    8.2747    8.4950    8.9896   10.8900   11.8252   12.3777   13.3513
    5.7233    5.8521    6.7906    7.2635    7.6253    8.2465    8.4668    8.9614   10.8618   11.7969   12.3495   13.3230
    5.1584    5.2872    6.2257    6.6985    7.0604    7.6815    7.9018    8.3964   10.2969   11.2320   11.7846   12.7581
    4.3685    4.4973    5.4358    5.9087    6.2705    6.8917    7.1120    7.6066    9.5070   10.4422   10.9947   11.9683
    4.3685    4.4973    5.4358    5.9087    6.2705    6.8917    7.1120    7.6066    9.5070   10.4422   10.9947   11.9683
    3.1717    3.3004    4.2389    4.7118    5.0736    5.6948    5.9151    6.4097    8.3101    9.2453    9.7978   10.7714
    2.9440    3.0728    4.0113    4.4842    4.8460    5.4672    5.6875    6.1821    8.0825    9.0177    9.5702   10.5438
    2.6809    2.8097    3.7482    4.2210    4.5829    5.2040    5.4244    5.9190    7.8194    8.7545    9.3071   10.2806
    1.6259    1.7547    2.6932    3.1661    3.5279    4.1491    4.3694    4.8640    6.7644    7.6996    8.2521    9.2257
    1.5853    1.7141    2.6526    3.1254    3.4873    4.1084    4.3288    4.8234    6.7238    7.6589    8.2115    9.1850
    1.5169    1.6457    2.5842    3.0571    3.4189    4.0401    4.2604    4.7550    6.6554    7.5906    8.1431    9.1166
    0.8968    1.0255    1.9640    2.4369    2.7987    3.4199    3.6402    4.1348    6.0352    6.9704    7.5229    8.4965
         0    0.1288    1.0673    1.5401    1.9020    2.5231    2.7435    3.2381    5.1385    6.0736    6.6262    7.5997
    0.1288         0    0.9385    1.4114    1.7732    2.3943    2.6147    3.1093    5.0097    5.9448    6.4974    7.4709
    1.0673    0.9385         0    0.4729    0.8347    1.4559    1.6762    2.1708    4.0712    5.0063    5.5589    6.5324
    1.5401    1.4114    0.4729         0    0.3619    0.9830    1.2033    1.6979    3.5984    4.5335    5.0860    6.0596
    1.9020    1.7732    0.8347    0.3619         0    0.6211    0.8415    1.3361    3.2365    4.1716    4.7242    5.6977
    2.5231    2.3943    1.4559    0.9830    0.6211         0    0.2203    0.7149    2.6154    3.5505    4.1030    5.0766
    2.7435    2.6147    1.6762    1.2033    0.8415    0.2203         0    0.4946    2.3950    3.3302    3.8827    4.8563
    3.2381    3.1093    2.1708    1.6979    1.3361    0.7149    0.4946         0    1.9004    2.8356    3.3881    4.3617
    5.1385    5.0097    4.0712    3.5984    3.2365    2.6154    2.3950    1.9004         0    0.9351    1.4877    2.4612
    6.0736    5.9448    5.0063    4.5335    4.1716    3.5505    3.3302    2.8356    0.9351         0    0.5526    1.5261
    6.6262    6.4974    5.5589    5.0860    4.7242    4.1030    3.8827    3.3881    1.4877    0.5526         0    0.9735
    7.5997    7.4709    6.5324    6.0596    5.6977    5.0766    4.8563    4.3617    2.4612    1.5261    0.9735         0


addconone =

   -6.4555


addcontwo =

    5.0634


vaf =

    0.9499


outpermone =

  Columns 1 through 26

     2     6    12    19    22     1    11     4     5    23    21    18     3    27    20    24    17    10    25     9    26    13     8    16     7    15

  Column 27

    14


outpermtwo =

  Columns 1 through 26

    18     4    19    16     3    11    15    17     5    14    27     7     8    13    12    21    20     6     2    22     9    10    25    23     1    26

  Column 27

    24


coordone =

  -14.9857
  -14.9857
  -14.9857
  -10.6768
  -10.6768
  -10.6768
   -7.2769
   -7.2769
   -7.2769
   -4.6149
   -4.6149
   -4.6149
```

```
   -0.0553
   -0.0553
   -0.0553
    3.6327
    3.6327
    3.6327
    5.6893
    5.6893
    5.6893
   10.6332
   10.6332
   10.6332
   17.6544
   17.6544
   17.6544


coordtwo =

   -6.1101
   -6.1101
   -6.1101
   -4.9751
   -4.9751
   -4.9751
   -2.6275
   -2.6275
   -2.6275
   -1.5461
   -1.5461
   -1.5461
   -0.2714
   -0.2714
   -0.2714
    0.9779
    0.9779
    0.9779
    3.0910
    3.0910
    3.0910
    4.2537
    4.2537
    4.2537
    7.2075
    7.2075
    7.2075


fitone =

  Columns 1 through 15

         0         0         0    4.3089    4.3089    4.3089    7.7088    7.7088    7.7088   10.3707   10.3707   10.3707   14.9304   14.9304   14.9304
         0         0         0    4.3089    4.3089    4.3089    7.7088    7.7088    7.7088   10.3707   10.3707   10.3707   14.9304   14.9304   14.9304
         0         0         0    4.3089    4.3089    4.3089    7.7088    7.7088    7.7088   10.3707   10.3707   10.3707   14.9304   14.9304   14.9304
    4.3089    4.3089    4.3089         0         0         0    3.3999    3.3999    3.3999    6.0619    6.0619    6.0619   10.6215   10.6215   10.6215
    4.3089    4.3089    4.3089         0         0         0    3.3999    3.3999    3.3999    6.0619    6.0619    6.0619   10.6215   10.6215   10.6215
    4.3089    4.3089    4.3089         0         0         0    3.3999    3.3999    3.3999    6.0619    6.0619    6.0619   10.6215   10.6215   10.6215
    7.7088    7.7088    7.7088    3.3999    3.3999    3.3999         0         0         0    2.6620    2.6620    2.6620    7.2216    7.2216    7.2216
    7.7088    7.7088    7.7088    3.3999    3.3999    3.3999         0         0         0    2.6620    2.6620    2.6620    7.2216    7.2216    7.2216
    7.7088    7.7088    7.7088    3.3999    3.3999    3.3999         0         0         0    2.6620    2.6620    2.6620    7.2216    7.2216    7.2216
   10.3707   10.3707   10.3707    6.0619    6.0619    6.0619    2.6620    2.6620    2.6620         0         0         0    4.5596    4.5596    4.5596
   10.3707   10.3707   10.3707    6.0619    6.0619    6.0619    2.6620    2.6620    2.6620         0         0         0    4.5596    4.5596    4.5596
   10.3707   10.3707   10.3707    6.0619    6.0619    6.0619    2.6620    2.6620    2.6620         0         0         0    4.5596    4.5596    4.5596
   14.9304   14.9304   14.9304   10.6215   10.6215   10.6215    7.2216    7.2216    7.2216    4.5596    4.5596    4.5596         0         0         0
   14.9304   14.9304   14.9304   10.6215   10.6215   10.6215    7.2216    7.2216    7.2216    4.5596    4.5596    4.5596         0         0         0
   14.9304   14.9304   14.9304   10.6215   10.6215   10.6215    7.2216    7.2216    7.2216    4.5596    4.5596    4.5596         0         0         0
   18.6184   18.6184   18.6184   14.3096   14.3096   14.3096   10.9097   10.9097   10.9097    8.2477    8.2477    8.2477    3.6881    3.6881    3.6881
   18.6184   18.6184   18.6184   14.3096   14.3096   14.3096   10.9097   10.9097   10.9097    8.2477    8.2477    8.2477    3.6881    3.6881    3.6881
   18.6184   18.6184   18.6184   14.3096   14.3096   14.3096   10.9097   10.9097   10.9097    8.2477    8.2477    8.2477    3.6881    3.6881    3.6881
   20.6750   20.6750   20.6750   16.3661   16.3661   16.3661   12.9663   12.9663   12.9663   10.3043   10.3043   10.3043    5.7447    5.7447    5.7447
   20.6750   20.6750   20.6750   16.3661   16.3661   16.3661   12.9663   12.9663   12.9663   10.3043   10.3043   10.3043    5.7447    5.7447    5.7447
   20.6750   20.6750   20.6750   16.3661   16.3661   16.3661   12.9663   12.9663   12.9663   10.3043   10.3043   10.3043    5.7447    5.7447    5.7447
   25.6189   25.6189   25.6189   21.3100   21.3100   21.3100   17.9101   17.9101   17.9101   15.2482   15.2482   15.2482   10.6885   10.6885   10.6885
   25.6189   25.6189   25.6189   21.3100   21.3100   21.3100   17.9101   17.9101   17.9101   15.2482   15.2482   15.2482   10.6885   10.6885   10.6885
   25.6189   25.6189   25.6189   21.3100   21.3100   21.3100   17.9101   17.9101   17.9101   15.2482   15.2482   15.2482   10.6885   10.6885   10.6885
   32.6400   32.6400   32.6400   28.3312   28.3312   28.3312   24.9313   24.9313   24.9313   22.2693   22.2693   22.2693   17.7097   17.7097   17.7097
   32.6400   32.6400   32.6400   28.3312   28.3312   28.3312   24.9313   24.9313   24.9313   22.2693   22.2693   22.2693   17.7097   17.7097   17.7097
   32.6400   32.6400   32.6400   28.3312   28.3312   28.3312   24.9313   24.9313   24.9313   22.2693   22.2693   22.2693   17.7097   17.7097   17.7097

  Columns 16 through 27

   18.6184   18.6184   18.6184   20.6750   20.6750   20.6750   25.6189   25.6189   25.6189   32.6400   32.6400   32.6400
   18.6184   18.6184   18.6184   20.6750   20.6750   20.6750   25.6189   25.6189   25.6189   32.6400   32.6400   32.6400
   18.6184   18.6184   18.6184   20.6750   20.6750   20.6750   25.6189   25.6189   25.6189   32.6400   32.6400   32.6400
   14.3096   14.3096   14.3096   16.3661   16.3661   16.3661   21.3100   21.3100   21.3100   28.3312   28.3312   28.3312
   14.3096   14.3096   14.3096   16.3661   16.3661   16.3661   21.3100   21.3100   21.3100   28.3312   28.3312   28.3312
```

```
   14.3096   14.3096   14.3096   16.3661   16.3661   16.3661   21.3100   21.3100   21.3100   28.3312   28.3312   28.3312
   10.9097   10.9097   10.9097   12.9663   12.9663   12.9663   17.9101   17.9101   17.9101   24.9313   24.9313   24.9313
   10.9097   10.9097   10.9097   12.9663   12.9663   12.9663   17.9101   17.9101   17.9101   24.9313   24.9313   24.9313
   10.9097   10.9097   10.9097   12.9663   12.9663   12.9663   17.9101   17.9101   17.9101   24.9313   24.9313   24.9313
    8.2477    8.2477    8.2477   10.3043   10.3043   10.3043   15.2482   15.2482   15.2482   22.2693   22.2693   22.2693
    8.2477    8.2477    8.2477   10.3043   10.3043   10.3043   15.2482   15.2482   15.2482   22.2693   22.2693   22.2693
    8.2477    8.2477    8.2477   10.3043   10.3043   10.3043   15.2482   15.2482   15.2482   22.2693   22.2693   22.2693
    3.6881    3.6881    3.6881    5.7447    5.7447    5.7447   10.6885   10.6885   10.6885   17.7097   17.7097   17.7097
    3.6881    3.6881    3.6881    5.7447    5.7447    5.7447   10.6885   10.6885   10.6885   17.7097   17.7097   17.7097
    3.6881    3.6881    3.6881    5.7447    5.7447    5.7447   10.6885   10.6885   10.6885   17.7097   17.7097   17.7097
         0         0         0    2.0566    2.0566    2.0566    7.0005    7.0005    7.0005   14.0216   14.0216   14.0216
         0         0         0    2.0566    2.0566    2.0566    7.0005    7.0005    7.0005   14.0216   14.0216   14.0216
         0         0         0    2.0566    2.0566    2.0566    7.0005    7.0005    7.0005   14.0216   14.0216   14.0216
    2.0566    2.0566    2.0566         0         0         0    4.9439    4.9439    4.9439   11.9650   11.9650   11.9650
    2.0566    2.0566    2.0566         0         0         0    4.9439    4.9439    4.9439   11.9650   11.9650   11.9650
    2.0566    2.0566    2.0566         0         0         0    4.9439    4.9439    4.9439   11.9650   11.9650   11.9650
    7.0005    7.0005    7.0005    4.9439    4.9439    4.9439         0         0         0    7.0211    7.0211    7.0211
    7.0005    7.0005    7.0005    4.9439    4.9439    4.9439         0         0         0    7.0211    7.0211    7.0211
    7.0005    7.0005    7.0005    4.9439    4.9439    4.9439         0         0         0    7.0211    7.0211    7.0211
   14.0216   14.0216   14.0216   11.9650   11.9650   11.9650    7.0211    7.0211    7.0211         0         0         0
   14.0216   14.0216   14.0216   11.9650   11.9650   11.9650    7.0211    7.0211    7.0211         0         0         0
   14.0216   14.0216   14.0216   11.9650   11.9650   11.9650    7.0211    7.0211    7.0211         0         0         0


fittwo =

  Columns 1 through 15

         0         0         0    1.1350    1.1350    1.1350    3.4826    3.4826    3.4826    4.5639    4.5639    4.5639    5.8387    5.8387    5.8387
         0         0         0    1.1350    1.1350    1.1350    3.4826    3.4826    3.4826    4.5639    4.5639    4.5639    5.8387    5.8387    5.8387
         0         0         0    1.1350    1.1350    1.1350    3.4826    3.4826    3.4826    4.5639    4.5639    4.5639    5.8387    5.8387    5.8387
    1.1350    1.1350    1.1350         0         0         0    2.3476    2.3476    2.3476    3.4290    3.4290    3.4290    4.7037    4.7037    4.7037
    1.1350    1.1350    1.1350         0         0         0    2.3476    2.3476    2.3476    3.4290    3.4290    3.4290    4.7037    4.7037    4.7037
    1.1350    1.1350    1.1350         0         0         0    2.3476    2.3476    2.3476    3.4290    3.4290    3.4290    4.7037    4.7037    4.7037
    3.4826    3.4826    3.4826    2.3476    2.3476    2.3476         0         0         0    1.0813    1.0813    1.0813    2.3561    2.3561    2.3561
    3.4826    3.4826    3.4826    2.3476    2.3476    2.3476         0         0         0    1.0813    1.0813    1.0813    2.3561    2.3561    2.3561
    3.4826    3.4826    3.4826    2.3476    2.3476    2.3476         0         0         0    1.0813    1.0813    1.0813    2.3561    2.3561    2.3561
    4.5639    4.5639    4.5639    3.4290    3.4290    3.4290    1.0813    1.0813    1.0813         0         0         0    1.2748    1.2748    1.2748
    4.5639    4.5639    4.5639    3.4290    3.4290    3.4290    1.0813    1.0813    1.0813         0         0         0    1.2748    1.2748    1.2748
    4.5639    4.5639    4.5639    3.4290    3.4290    3.4290    1.0813    1.0813    1.0813         0         0         0    1.2748    1.2748    1.2748
    5.8387    5.8387    5.8387    4.7037    4.7037    4.7037    2.3561    2.3561    2.3561    1.2748    1.2748    1.2748         0         0         0
    5.8387    5.8387    5.8387    4.7037    4.7037    4.7037    2.3561    2.3561    2.3561    1.2748    1.2748    1.2748         0         0         0
    5.8387    5.8387    5.8387    4.7037    4.7037    4.7037    2.3561    2.3561    2.3561    1.2748    1.2748    1.2748         0         0         0
    7.0880    7.0880    7.0880    5.9530    5.9530    5.9530    3.6054    3.6054    3.6054    2.5240    2.5240    2.5240    1.2493    1.2493    1.2493
    7.0880    7.0880    7.0880    5.9530    5.9530    5.9530    3.6054    3.6054    3.6054    2.5240    2.5240    2.5240    1.2493    1.2493    1.2493
    7.0880    7.0880    7.0880    5.9530    5.9530    5.9530    3.6054    3.6054    3.6054    2.5240    2.5240    2.5240    1.2493    1.2493    1.2493
    9.2010    9.2010    9.2010    8.0660    8.0660    8.0660    5.7184    5.7184    5.7184    4.6371    4.6371    4.6371    3.3623    3.3623    3.3623
    9.2010    9.2010    9.2010    8.0660    8.0660    8.0660    5.7184    5.7184    5.7184    4.6371    4.6371    4.6371    3.3623    3.3623    3.3623
    9.2010    9.2010    9.2010    8.0660    8.0660    8.0660    5.7184    5.7184    5.7184    4.6371    4.6371    4.6371    3.3623    3.3623    3.3623
   10.3638   10.3638   10.3638    9.2288    9.2288    9.2288    6.8812    6.8812    6.8812    5.7998    5.7998    5.7998    4.5251    4.5251    4.5251
   10.3638   10.3638   10.3638    9.2288    9.2288    9.2288    6.8812    6.8812    6.8812    5.7998    5.7998    5.7998    4.5251    4.5251    4.5251
   10.3638   10.3638   10.3638    9.2288    9.2288    9.2288    6.8812    6.8812    6.8812    5.7998    5.7998    5.7998    4.5251    4.5251    4.5251
   13.3176   13.3176   13.3176   12.1826   12.1826   12.1826    9.8350    9.8350    9.8350    8.7536    8.7536    8.7536    7.4789    7.4789    7.4789
   13.3176   13.3176   13.3176   12.1826   12.1826   12.1826    9.8350    9.8350    9.8350    8.7536    8.7536    8.7536    7.4789    7.4789    7.4789
   13.3176   13.3176   13.3176   12.1826   12.1826   12.1826    9.8350    9.8350    9.8350    8.7536    8.7536    8.7536    7.4789    7.4789    7.4789

  Columns 16 through 27

    7.0880    7.0880    7.0880    9.2010    9.2010    9.2010   10.3638   10.3638   10.3638   13.3176   13.3176   13.3176
    7.0880    7.0880    7.0880    9.2010    9.2010    9.2010   10.3638   10.3638   10.3638   13.3176   13.3176   13.3176
    7.0880    7.0880    7.0880    9.2010    9.2010    9.2010   10.3638   10.3638   10.3638   13.3176   13.3176   13.3176
    5.9530    5.9530    5.9530    8.0660    8.0660    8.0660    9.2288    9.2288    9.2288   12.1826   12.1826   12.1826
    5.9530    5.9530    5.9530    8.0660    8.0660    8.0660    9.2288    9.2288    9.2288   12.1826   12.1826   12.1826
    5.9530    5.9530    5.9530    8.0660    8.0660    8.0660    9.2288    9.2288    9.2288   12.1826   12.1826   12.1826
    3.6054    3.6054    3.6054    5.7184    5.7184    5.7184    6.8812    6.8812    6.8812    9.8350    9.8350    9.8350
    3.6054    3.6054    3.6054    5.7184    5.7184    5.7184    6.8812    6.8812    6.8812    9.8350    9.8350    9.8350
    3.6054    3.6054    3.6054    5.7184    5.7184    5.7184    6.8812    6.8812    6.8812    9.8350    9.8350    9.8350
    2.5240    2.5240    2.5240    4.6371    4.6371    4.6371    5.7998    5.7998    5.7998    8.7536    8.7536    8.7536
    2.5240    2.5240    2.5240    4.6371    4.6371    4.6371    5.7998    5.7998    5.7998    8.7536    8.7536    8.7536
    2.5240    2.5240    2.5240    4.6371    4.6371    4.6371    5.7998    5.7998    5.7998    8.7536    8.7536    8.7536
    1.2493    1.2493    1.2493    3.3623    3.3623    3.3623    4.5251    4.5251    4.5251    7.4789    7.4789    7.4789
    1.2493    1.2493    1.2493    3.3623    3.3623    3.3623    4.5251    4.5251    4.5251    7.4789    7.4789    7.4789
    1.2493    1.2493    1.2493    3.3623    3.3623    3.3623    4.5251    4.5251    4.5251    7.4789    7.4789    7.4789
         0         0         0    2.1130    2.1130    2.1130    3.2758    3.2758    3.2758    6.2296    6.2296    6.2296
         0         0         0    2.1130    2.1130    2.1130    3.2758    3.2758    3.2758    6.2296    6.2296    6.2296
         0         0         0    2.1130    2.1130    2.1130    3.2758    3.2758    3.2758    6.2296    6.2296    6.2296
    2.1130    2.1130    2.1130         0         0         0    1.1627    1.1627    1.1627    4.1166    4.1166    4.1166
    2.1130    2.1130    2.1130         0         0         0    1.1627    1.1627    1.1627    4.1166    4.1166    4.1166
    2.1130    2.1130    2.1130         0         0         0    1.1627    1.1627    1.1627    4.1166    4.1166    4.1166
    3.2758    3.2758    3.2758    1.1627    1.1627    1.1627         0         0         0    2.9538    2.9538    2.9538
    3.2758    3.2758    3.2758    1.1627    1.1627    1.1627         0         0         0    2.9538    2.9538    2.9538
    3.2758    3.2758    3.2758    1.1627    1.1627    1.1627         0         0         0    2.9538    2.9538    2.9538
    6.2296    6.2296    6.2296    4.1166    4.1166    4.1166    2.9538    2.9538    2.9538         0         0         0
    6.2296    6.2296    6.2296    4.1166    4.1166    4.1166    2.9538    2.9538    2.9538         0         0         0
    6.2296    6.2296    6.2296    4.1166    4.1166    4.1166    2.9538    2.9538    2.9538         0         0         0
```

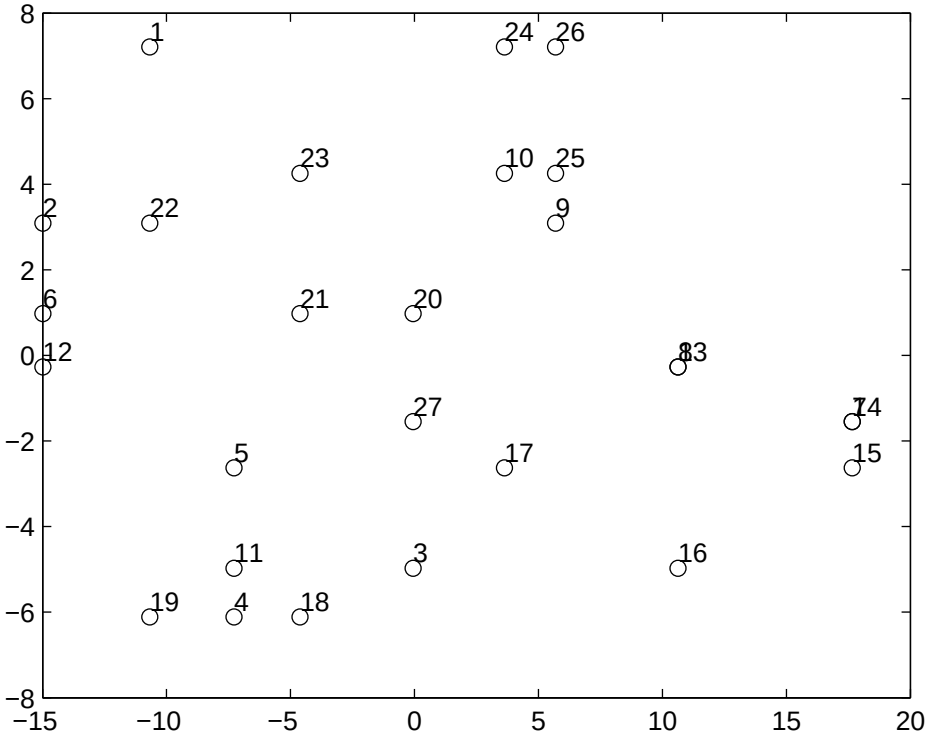


Figure 18: The City-Block Scaling of the Hampshire Towns Based on the Given Tied Coordinate Patterns (in Groups of Three) Obtained with biscalqa_tied.m.

```
addconone =

   -7.1024

addcontwo =

    4.8676

vaf =

    0.8734
```

References

[1] Eyster, J. W., & White, J. A. (1973). Some properties of the squared Euclidean distance location problem. *AIIE Transactions*, *5*, 275–280.

[2] Eyster, J. W., White, J. A., & Wierwille, W. W. (1973). On solving multifacility location problems using a hyperboloid approximation procedure. *AIIE Transactions*, *5*, 1–6.

[3] Johnson, E. J., & Tversky, A. (1984). Representations of perceptions of risks. *Journal of Experimental Psychology: General*, *113*, 55–70.

[4] Osgood, C. E., & Luria, Z. (1954). A blind analysis of a case of multiple personality using the Sematic Differential. *Journal of Abnormal and Social Psychology*, *49*, 579–591.

A Header Comments for the M-files Mentioned in the Text or Used Internally by Other M-files; Given in Alphabetical Order

atreedec.m

```
% ATREEDEC decomposes a given additive tree matrix into an
% ultrametric and a centroid metric matrix (where the root is
% halfway along the longest path).
%
% syntax: [ulmetric,ctmetric] = atreedec(prox,constant)
%
% PROX is the input proximity matrix (with a zero main diagonal
% and a dissimilarity interpretation);
% CONSTANT is a nonnegative number (less than or equal to the
% maximum proximity value) that controls the
% positivity of the constructed ultrametric values;
% ULMETRIC is the ultrametric component of the decomposition;
% CTMETRIC is the centroid metric component of the decomposition
% (given by values $g_{1},...,g_{n}$ for each of the objects,
% some of which may actually be negative depending on the input
% proximity matrix used).
```

biplottm.m

```
% BIPLOTTM plots the combined row and column object set using
% coordinates given in the $n \times 2$ matrix AXES; here the
% number of rows is NROW and the number of columns is NCOL,
% and $n$ is the sum of NROW and NCOL.
%
% syntax: [] = biplottm(axes,nrow,ncol)
%
% The first NROW rows of AXES give the row object coordinates;
% the last NCOL rows of AXES give the column object coordinates.
% The plotting symbol for rows is a circle (o);
% for columns it is an asterisk (*).
% The labels for rows are from 1 to NROW;
% those for columns are from 1 to NCOL.
```

biscalqa_tied

```
%  BISCALQA_TIED carries out a bidimensional scaling of a symmetric
```

```
%  proximity matrix using iterative quadratic assignment.
%
%  syntax: [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,...
%    addconone,addcontwo,vaf] = ...
%     biscalqa_tied(prox,targone,targtwo,inpermone,inpermtwo, ...
%     tiedcoordone,tiedcoordtwo,kblock,nopt)
%
%  PROX is the input proximity matrix (with a zero main diagonal
%  and a dissimilarity interpretation);
%  TARGONE is the input target matrix for the first dimension
%  (usually with a zero main diagonal and a dissimilarity
%  interpretation representing equally spaced locations along
%  a continuum); TARGTWO is the input target
%  matrix for the second dimension;
%  INPERMONE is the input beginning permutation for the first
%  dimension (a permutation of the first $n$ integers);
%  INPERMTWO is the input beginning
%  permutation for the second dimension;
%  the insertion and rotation routines use from 1 to KBLOCK
%  (which is less than or equal to $n-1$) consecutive objects in
%  the permutation defining the row and column orders of the data
%  matrix. NOPT controls the confirmatory or exploratory fitting
%  of the unidimensional scales; a value of NOPT = 0 will fit in a
%  confirmatory manner the two scales
%  indicated by INPERMONE and INPERMTWO;
%  a value of NOPT = 1 uses iterative QA
%  to locate the better permutations to fit;
%  TIEDCOORDONE and TIEDCOORDTWO specify the pattern of
%  tied coordinates (using integers from 1 up to n);
%  OUTPERMONE is the final object permutation for the
%  first dimension; OUTPERMTWO is the final object permutation
%  for the second dimension;
%  COORDONE is the set of first dimension coordinates
%  in ascending order; COORDTWO is the set of second dimension
%  coordinates in ascending order;
%  ADDCONONE is the additive constant for the first
%  dimensional model; ADDCONTWO is the additive constant for
%  the second dimensional model;
%  VAF is the variance-accounted-for in PROX by
%  the bidimensional scaling.
```

circularplot.m

```
% CIRCULARPLOT plots the object set using the coordinates
```

```
% around a circular structure derived from the $n \times n$
% interpoint distance matrix around a circle given by CIRC.
% The positions are labeled by the order of objects
% given in INPERM.
%
% syntax: [circum,radius,coord,degrees,cumdegrees] = ...
%   circularplot(circ,inperm)
%
% The output consists of a plot, the circumference of the
% circle (CIRCUM) and radius (RADIUS); the coordinates of
% the plot positions (COORD), and the degrees and cumulative
% degrees induced between the plot positions
% (in DEGREES and CUMDEGREES).
% The positions around the circle are numbered from 1
% (at the "noon" position) to $n$, moving
% clockwise around the circular structure.
```

cityblock_multifacility_hap.m

```
% CITYBLOCK_MULTIFACILITY_HAP carries out the Cityblock multifacility
% location problem using a hyperboloid approximation procedure.
%
% syntax: [new_facility_location,distance_new_to_old,...
%    distance_among_new,objective_function] = ...
%    cityblock_multifacility_hap(weight_new_to_old,...
%    weight_among_new,old_facility_location,epsil)
%
% There are n new facilities to place among m existing facilities in d
% dimensions.  WEIGHT_NEW_TO_OLD (n x m) and WEIGHT_AMONG_NEW (n x n) are
% both weight matrices (with similarity interpretations);
% OLD_FACILITY_LOCATION (d x m) contains the coordinates for the existing
% facilities; and EPSIL (e.g., 1.0e-012) produces the hyperboloid
% approximation allowing the objective function to be differentiable
% everywhere.
% As output, there are the NEW_FACILITY_LOCATIONS (d x n);
% DISTANCE_NEW_TO_OLD (n x m); DISTANCE_AMONG_NEW (n x n); and the
% OBJECTIVE_FUNCTION (i.e., the weighted sum of the distances among the new
% locations plus of the new locations to those existing).
```

data_generate_cityblock_and_euclidean.m

```
% DATA_GENERATE_CITYBLOCK_AND_EUCLIDEAN produces data having an underlying
% city-block and Euclidean structure.
```

```
%
% syntax: [datamatrix_cb,datamatrix_eu,corr_cb_eu,gamma_cb_eu,datavectors,coord] ...
%    = data_generate_cityblock_and_euclidean(n,error,elongation_factor)
%
% The input parameters are the number of objects, $n$; ERROR is in the form of
% a multiplier used on a randomly (entry) permutated proximity matrix with
% perfect city-block or Euclidean structure that is then added to these
% latter matrices; the ELONGATION_FACTOR refers to a multiplier used on the
% randomly generated first dimension coordinates (in case a dominant
% dimension is desired).
% The outputs are the cityblock and Euclidean $n \times n$ proximity
% matrices, DATAMATRIX_CB and DATAMATRIX_EU, respectively; CORR_CB_EU and
% GAMMA_CB_EU are the Pearson and Goodman-Kruskal coefficients of correlation
% between the entries in DATAMATRIX_CB and DATAMATRIX_EU; DATAVECTORS is
% $n*(n-1)/2 \times 2$ and contrains the entries in DATAMATRIX_CB and
% DATAMATRIX_EU in vector form; COORD contains the randomly generated
% coordinates.
```

euclidean_multifacility_hap.m

```
% EUCLIDEAN_MULTIFACILITY_HAP carries out the Euclidean multifacility
% location problem using a hyperboloid approximation procedure.
%
% syntax: [new_facility_location,distance_new_to_old,...
%    distance_among_new,objective_function] = ...
%    euclidean_multifacility_hap(weight_new_to_old,...
%    weight_among_new,old_facility_location,epsil)
%
% There are n new facilities to place among m existing facilities in d
% dimensions.  WEIGHT_NEW_TO_OLD (n x m) and WEIGHT_AMONG_NEW (n x n) are
% both weight matrices (with similarity interpretations);
% OLD_FACILITY_LOCATION (d x m) contains the coordinates for the existing
% facilities; and EPSIL (e.g., 1.0e-012) produces the hyperboloid
% approximation allowing the objective function to be differentiable
% everywhere.
% As output, there are the NEW_FACILITY_LOCATIONS (d x n);
% DISTANCE_NEW_TO_OLD (n x m); DISTANCE_AMONG_NEW (n x n); and the
% OBJECTIVE_FUNCTION (i.e., the weighted sum of the distances among the new
% locations plus of the new locations to those existing).
```

ms_biarobfnd.m

```
% BIAROBFND finds and fits the sum of two
% anti-Robinson matrices using iterative projection to
```

```
% a symmetric proximity matrix in the $L_{2}$-norm
% based on permutations identified through
% the use of iterative quadratic assignment.
%
% syntax: [find,vaf,targone,targtwo,outpermone, ...
%     outpermtwo] = biarobfnd(prox,inperm,kblock)
%
% PROX is the input proximity matrix ($n \times n$ with a zero
% main diagonal and a dissimilarity interpretation);
% INPERM is a given starting permutation of the first $n$ integers;
% FIND is the least-squares optimal matrix (with
% variance-accounted-for of VAF)
% to PROX and is the sum of the two anti-Robinson matrices
% TARGONE and TARGTWO based on the two row and column
% object orderings given by the ending permutations OUTPERMONE
% and OUTPERMTWO. KBLOCK defines the block size in the use of the
% iterative quadratic assignment routine.
```

ms_biatreefnd.m

```
% BIATREEFND finds and fits the sum
% of two additive trees using iterative projection
% heuristically on a symmetric proximity matrix in the $L_{2}$-norm.
%
% syntax: [find,vaf,targone,targtwo] = biatreefnd(prox,inperm)
%
% PROX is the input proximity matrix (with a zero main diagonal
% and a dissimilarity interpretation);
% INPERM is a permutation that determines the order in which the
% inequality constraints are considered;
% FIND is the found least-squares matrix (with variance-accounted-for
% of VAF) to PROX and is the sum of
% the two additive tree matrices TARGONE and TARGTWO.
```

ms_bicirac.m

```
% BICIRAC finds and fits the sum of two circular
% unidimensional scales using iterative projection to
% a symmetric proximity matrix in the $L_{2}$-norm based on
% permutations identified through the use
% of iterative quadratic assignment.
%
% syntax: [find,vaf,targone,targtwo,outpermone,outpermtwo, ...
%     addconone,addcontwo] = bicirac(prox,inperm,kblock)
```

```
%
% PROX is the input proximity matrix ($n \times n$ with a zero
% main diagonal and a dissimilarity interpretation);
% INPERM is a given starting permutation of the first $n$ integers;
% FIND is the least-squares optimal matrix (with variance-
% accounted-for of VAF) to PROX and is the sum of the two
% circular anti-Robinson matrices;
% TARGONE and TARGTWO are based on the two row and column
% object orderings given by the ending permutations OUTPERMONE
% and OUTPERMTWO. KBLOCK defines the block size in the use of the
% iterative quadratic assignment routine and ADDCONONE and ADDCONTWO
% are the two additive constants for the two model components.
```

ms_bimonscalqa.m

```
%  BIMONSCALQA carries out a bidimensional scaling of a symmetric
%  proximity matrix using iterative quadratic assignment, plus
%  it provides an optimal monotonic transformation (MONPROX) of
%  the original input proximity matrix.
%
%  syntax: [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo, ...
%       addconone,addcontwo,vaf,monprox] = ...
%    bimonscalqa(prox,targone,targtwo,inpermone,inpermtwo,kblock,nopt)
%
%  PROX is the input proximity matrix (with a zero main diagonal
%  and a dissimilarity interpretation);
%  TARGONE is the input target matrix for the
%  first dimension (usually with
%  a zero main diagonal and with a
%  dissimilarity interpretation representing
%  equally spaced locations along a continuum);
%  TARGTWO is the input target
%  matrix for the second dimension;
%  INPERMONE is the input beginning permutation for the first
%  dimension (a permutation of the first $n$ integers);
%  INPERMTWO is the input beginning
%  permutation for the second dimension;
%  the insertion and rotation routines use from 1 to KBLOCK
%  (which is less than or equal to $n-1$) consecutive objects in
%  the permutation defining the row and column orders of the data
%  matrix; NOPT controls the confirmatory or exploratory fitting of
%  the unidimensional scales; a value of NOPT = 0 will fit in a
%  confirmatory manner the two scales indicated by INPERMONE
%  and INPERMTWO; a value of NOPT = 1 uses iterative QA
```

```
%  to locate the better permutations to fit;
%  OUTPERMONE is the final object permutation for the first
%  dimension; OUTPERMTWO is the final object
%  permutation for the second dimension;
%  COORDONE is the set of first dimension coordinates
%  in ascending order; COORDTWO is the set of second
%  dimension coordinates in ascending order;
%  ADDCONONE is the additive constant for the first dimensional
%  model; ADDCONTWO is the additive constant for the second
%  dimensional model; VAF is the variance-accounted-for
%  in MONPROX by the bidimensional scaling.
```

ms_bimonscaltmac.m

```
% BIMONSCALTMAC finds and fits the sum of two linear unidimensional
% scales using iterative projection to
% a two-mode proximity matrix in the $L_{2}$-norm based on
% permutations identified through the use of iterative quadratic
% assignment.  It also provides an optimal monotonic transformation
% (MONPROX) of the original input proximity matrix.
%
% syntax: [find,vaf,targone,targtwo,outpermone,outpermtwo, ...
%        rowpermone,colpermone,rowpermtwo,colpermtwo,addconone,...
%        addcontwo,coordone,coordtwo,axes,monproxtm] = ...
%    bimonscaltmac(proxtm,inpermone,inpermtwo,kblock,nopt)
%
% PROXTM is the input two-mode proximity matrix ($nrow \times ncol$
% with a dissimilarity interpretation);
% FIND is the least-squares optimal matrix (with variance-
% accounted-for of VAF) to the monotonic transformation MONPROXTM of
% the input proximity matrix and is the sum of the two matrices
% TARGONE and TARGTWO based on the two row and column
% object orderings given by the ending permutations OUTPERMONE
% and OUTPERMTWO, and in turn ROWPERMONE and ROWPERMTWO and
% COLPERMONE and COLPERMTWO. KBLOCK defines the block size in
% the use of the iterative quadratic assignment routine and ADDCONONE
% and ADDCONTWO are the two additive constants for the two model
% components; The $n$ coordinates are in COORDONE and COORDTWO.
% The input permutations are INPERMONE and INPERMTWO.  The
% $n \times 2$ matrix AXES gives the plotting coordinates for the
% combined row and column object set.
% NOPT controls the confirmatory or exploratory fitting of
% the unidimensional scales; a value of NOPT = 0 will fit in a
% confirmatory manner the two scales
```

```
% indicated by INPERMONE and INPERMTWO;
% a value of NOPT = 1 uses iterative QA
% to locate the better permutations to fit.
```

ms_biscalqa.m

```
%  BISCALQA carries out a bidimensional scaling of a symmetric
%  proximity matrix using iterative quadratic assignment.
%
%  syntax: [outpermone,outpermtwo,coordone,coordtwo,fitone,fittwo,...
%    addconone,addcontwo,vaf] = ...
%     biscalqa(prox,targone,targtwo,inpermone,inpermtwo,kblock,nopt)
%
%  PROX is the input proximity matrix (with a zero main diagonal
%  and a dissimilarity interpretation);
%  TARGONE is the input target matrix for the first dimension
%  (usually with a zero main diagonal and a dissimilarity
%  interpretation representing equally spaced locations along
%  a continuum); TARGTWO is the input target
%  matrix for the second dimension;
%  INPERMONE is the input beginning permutation for the first
%  dimension (a permutation of the first $n$ integers);
%  INPERMTWO is the input beginning
%  permutation for the second dimension;
%  the insertion and rotation routines use from 1 to KBLOCK
%  (which is less than or equal to $n-1$) consecutive objects in
%  the permutation defining the row and column orders of the data
%  matrix. NOPT controls the confirmatory or exploratory fitting
%  of the unidimensional scales; a value of NOPT = 0 will fit in a
%  confirmatory manner the two scales
%  indicated by INPERMONE and INPERMTWO;
%  a value of NOPT = 1 uses iterative QA
%  to locate the better permutations to fit;
%  OUTPERMONE is the final object permutation for the
%  first dimension; OUTPERMTWO is the final object permutation
%  for the second dimension;
%  COORDONE is the set of first dimension coordinates
%  in ascending order; COORDTWO is the set of second dimension
%  coordinates in ascending order;
%  ADDCONONE is the additive constant for the first
%  dimensional model; ADDCONTWO is the additive constant for
%  the second dimensional model;
%  VAF is the variance-accounted-for in PROX by
%  the bidimensional scaling.
```

ms_biscaltmac.m

```
% BISCALTMAC finds and fits the sum of two linear
% unidimensional scales using iterative projection to
% a two-mode proximity matrix in the $L_{2}$-norm based on
% permutations identified through the use of iterative quadratic
% assignment.
%
% syntax: [find,vaf,targone,targtwo,outpermone,outpermtwo, ...
%        rowpermone,colpermone,rowpermtwo,colpermtwo,addconone,...
%        addcontwo,coordone,coordtwo,axes] = ...
%    biscaltmac(proxtm,inpermone,inpermtwo,kblock,nopt)
%
% PROXTM is the input two-mode proximity matrix ($nrow \times ncol$
% with a dissimilarity interpretation);
% FIND is the least-squares optimal matrix (with variance-accounted-
% for of VAF) to PROXTM and is the sum of the two matrices
% TARGONE and TARGTWO based on the two row and column
% object orderings given by the ending permutations OUTPERMONE
% and OUTPERMTWO, and in turn ROWPERMONE and ROWPERMTWO and
% COLPERMONE and COLPERMTWO. KBLOCK defines the block size
% in the use of the iterative quadratic assignment routine and
% ADDCONONE and ADDCONTWO are
% the two additive constants for the two model components;
% The $n$ coordinates
% are in COORDONE and COORDTWO.  The input permutations are INPERMONE
% and INPERMTWO.  The $n \times 2$ matrix AXES gives the
% plotting coordinates for the
% combined row and column object set.
% NOPT controls the confirmatory or
% exploratory fitting of the unidimensional
% scales; a value of NOPT = 0 will
% fit in a confirmatory manner the two scales
% indicated by INPERMONE and INPERMTWO;
% a value of NOPT = 1 uses iterative QA
% to locate the better permutations to fit.
```

ms_biultrafnd.m

```
% BIULTRAFND finds and fits the sum
% of two ultrametrics using iterative projection
% heuristically on a symmetric proximity matrix in the $L_{2}$-norm.
%
% syntax: [find,vaf,targone,targtwo] = biultrafnd(prox,inperm)
```

```
%
% PROX is the input proximity matrix (with a zero main diagonal
% and a dissimilarity interpretation);
% INPERM is a permutation that determines the order in which the
% inequality constraints are considered;
% FIND is the found least-squares matrix (with variance-accounted-for
% of VAF) to PROX and is the sum
% of the two ultrametric matrices TARGONE and TARGTWO.
```

squared_euclidean_multifacility.m

```
% SQUARED_EUCLIDEAN_MULTIFACILITY carries out the squared Euclidean multifacility
% location problem using closed form expressions for the results.
%
% syntax: [new_facility_location,distance_new_to_old,...
%    distance_among_new,objective_function] = ...
%    squared_euclidean_multifacility(weight_new_to_old,...
%    weight_among_new,old_facility_location)
%
% There are n new facilities to place among m existing facilities in d
% dimensions.  WEIGHT_NEW_TO_OLD (n x m) and WEIGHT_AMONG_NEW (n x n) are
% both weight matrices (with similarity interpretations);
% OLD_FACILITY_LOCATION (d x m) contains the coordinates for the existing
% facilities.
% As output, there are the NEW_FACILITY_LOCATIONS (d x n);
% DISTANCE_NEW_TO_OLD (n x m); DISTANCE_AMONG_NEW (n x n); and the
% OBJECTIVE_FUNCTION (i.e., the weighted sum of the distances among the new
% locations plus of the new locations to those existing).
```

ultraorder.m

```
% ULTRAORDER finds for the input proximity matrix PROX
% (assumed to be ultrametric with a zero main diagonal)
% a permutation ORDERPERM that displays the anti-
% Robinson form in the reordered proximity matrix
% ORDERPROX; thus, prox(orderperm,orderperm) = orderprox.
%
% syntax:  [orderprox,orderperm] = ultraorder(prox)
```