Lower (Anti-)Robinson Rank Representations for Symmetric Proximity Matrices

Lawrence J. Hubert[1] and Hans-Friedrich Köhn[2]

[1] Department of Psychology, University of Illinois
603 East Daniel Street, Champaign, Illinois 61820, *lhubert@cyrus.psych.uiuc.edu*
[2] Department of Psychology, University of Illinois
603 East Daniel Street, Champaign, Illinois 61820, *hkoehn@cyrus.psych.uiuc.edu*

Abstract. Edwin Diday, some two decades ago, was among the first few individuals to recognize the importance of the (anti-)Robinson form for representing a proximity matrix, and was the leader in suggesting how such matrices might be depicted graphically (as pyramids). We characterize the notions of an anti-Robinson (AR) and strongly anti-Robinson (SAR) matrix, and provide open-source M-files within a MATLAB environment to effect additive decompositions of a given proximity matrix into sums of AR (or SAR) matrices. We briefly introduce how the AR (or SAR) rank of a matrix might be specified.

1 Introduction

Various methods have been developed in the classification literature for representing the structure that may be present in a symmetric proximity matrix. The motivating bases for these strategies have been diverse, and include the reliance on spatial analogues (e.g., in multidimensional scaling), graph-theoretic concepts (e.g., in hierarchical clustering and the construction of additive trees), and order-constrained approximation matrices (e.g., matrices that satisfy the set of (anti-)Robinson (AR) order restrictions, characterized by a pattern of entries within each row and column never decreasing when moving away from the main diagonal in any direction; for historical precedents, see Robinson (1951)). It is within this last category of approximating a given proximity matrix by another that is order-constrained (and where, for convenience, proximity is now assumed keyed as a dissimilarity, so smaller values reflect more similar objects) in which Diday's contributions loam large. In the early 1980's and culminating in Diday (1986), he introduced the field to how (anti-)Robinson matrices may generally be represented through what are called pyramidal indices and their associated graphical display, or more broadly, to the relevance of the (graph-theoretic) literature on object seriation and its relation to the notion of an (anti-)Robinson form. We briefly review in this short paper a few of the advances in the last two decades, emphasizing, in particular, how sums of AR matrices might be identified and fitted through the minimization of a least-squares loss criterion. For a very

comprehensive and current review of the whole area of hierarchical representations and their various extensions, the reader is referred to Barthélemy, Brucker, and Osswald (2004).

2 Some definitions

Given an arbitrary symmetric $n \times n$ matrix, $\mathbf{A} = \{a_{ij}\}$, where the main diagonal entries are considered irrelevant and assumed to be zero (i.e., $a_{ii} = 0$ for $1 \leq i \leq n$), $\mathbf{A}$ is said to have an anti-Robinson (AR) form if after some reordering of the rows and columns of $\mathbf{A}$, the entries within each row and column have a distinctive pattern: moving away from the zero main diagonal entry within any row or any column, the entries never decrease. The entries in any AR matrix $\mathbf{A}$ can be reconstructed exactly through a collection of M subsets of the original object set $S = \{O_1, \ldots, O_n\}$, denoted by $S_1, \ldots, S_M$, and where M is determined by the particular pattern of tied entries, if any, in $\mathbf{A}$. These M subsets have the following characteristics:

(i) each S_m, $1 \leq m \leq M$, consists of a sequence of (two or more) consecutive integers so that $M \leq n(n-1)/2$. (This bound holds because the number of different subsets having consecutive integers for any given fixed ordering is $n(n-1)/2$, and will be achieved if all the entries in the AR matrix $\mathbf{A}$ are distinct).

(ii) each S_m, $1 \leq m \leq M$, has a diameter, denoted by $d(S_m)$, so that for all object pairs within S_m, the corresponding entries in $\mathbf{A}$ are less than or equal to the diameter. The subsets, $S_1, \ldots, S_M$, can be assumed ordered as $d(S_1) \leq d(S_2) \leq \cdots \leq d(S_M)$, and if $S_m \subseteq S_{m'}$, $d(S_m) \leq d(S_{m'})$.

(iii) each entry in $\mathbf{A}$ can be reconstructed from $d(S_1), \ldots, d(S_M)$, i.e., for $1 \leq i, j \leq n$,

$$a_{ij} = \min_{1 \leq m \leq M} \{d(S_m) \mid O_i, O_j \in S_m\},$$

so that the minimum diameter for subsets containing an object pair $O_i, O_j \in S$ is equal to a_{ij}. Given $\mathbf{A}$, the collection of subsets $S_1, \ldots, S_M$ and their diameters can be identified by inspection through the use of an increasing threshold that starts from the smallest entry in $\mathbf{A}$, and observing which subsets containing contiguous objects emerge from this process. The substantive interpretation of what $\mathbf{A}$ is depicting reduces to explaining why those subsets with the smallest diameters are so homogenous.

If the matrix $\mathbf{A}$ has a somewhat more restrictive form than just being AR, and is also *strongly* anti-Robinson (SAR), a convenient graphical representation can be given to the collection of AR reconstructive subsets $S_1, \ldots, S_M$ and their diameters, and how they can serve to retrieve $\mathbf{A}$. Specifically, $\mathbf{A}$ is said to be strongly anti-Robinson (SAR) if (considering the above-diagonal entries of $\mathbf{A}$) whenever two entries in adjacent columns are equal ($a_{ij} = a_{i(j+1)}$), those in the same two adjacent columns in the previous row

are also equal ($a_{(i-1)j} = a_{(i-1)(j+1)}$ for $1 \leq i-1 < j \leq n-1$); also, whenever two entries in adjacent rows are equal ($a_{ij} = a_{(i+1)j}$), those in the same two adjacent rows in the succeeding column are also equal ($a_{i(j+1)} = a_{(i+1)(j+1)}$ for $2 \leq i+1 < j \leq n-1$).

The reconstruction of an SAR matrix through the collection of consecutively defined object subsets, $S_1, \ldots, S_M$, and their diameters, and how these serve to reconstruct $\mathbf{A}$ can be modeled graphically (see Figure 1). Internal nodes would be at a height equal to the diameter of the respective subset; the consecutive objects forming that subset are identifiable by downward paths from the internal nodes to the terminal nodes corresponding to the objects in $S = \{O_1, \ldots, O_n\}$. An entry a_{ij} in $\mathbf{A}$ can be reconstructed as the minimum node height of a subset for which a path can be constructed from O_i up to that internal node and then back down to O_j.

As a few final introductory historical notes, there is now a rather extensive literature on graphically representing a matrix having an AR or SAR form. The reader interested in pursuing some of the relevant literature might begin with the earlier cited reference by Diday (1986) and his introduction to graphically representing an AR matrix by a 'pyramid', and then continue with the review by Durand and Fichet (1988), who point out the necessity of strengthening the AR condition to one that is SAR if a consistent graphical (pyramidal) representation is to be possible with no unresolvable graphical anomalies. For further discussion and development of some of these representations issues, the reader is referred to Diatta and Fichet (1998), Critchley (1994), Critchley and Fichet (1994), and Mirkin (1996, Chapter 7).

2.1 An illustrative numerical example

The proximity matrix given in Table 1 was published by *The New York Times* (July 2, 2005), and contains the percentages of non-unanimous cases in which the U.S. Supreme Court Justices *dis*agreed from the 1994/95 term through 2003/04 (known as the Rehnquist Court). The (upper-triangular portion of the) dissimilarity matrix is given in the same row and column order as the *Times* data set, with the justices ordered from "liberal" to "conservative":

1: John Paul Stevens (St)
2: Stephen G. Breyer (Br)
3: Ruth Bader Ginsberg (Gi)
4: David Souter (So)
5: Sandra Day O'Connor (Oc)
6: Anthony M. Kennedy (Ke)
7: William H. Rehnquist (Re)
8: Antonin Scalia (Sc)
9: Clarence Thomas (Th)

The lower-triangular portion of Table 1 is a best-fitting (least-squares) SAR matrix obtained with the MATLAB M-file `sarobfnd.m` mentioned in the

next section. The variance-accounted-for is 98.62%, so there is little residual variability left. A graphical representation is given in Figure 1; the 'pyramidal' structure would be more apparent if the vertical lines were tilted slightly inward toward the internal nodes.

	St	Br	Gi	So	Oc	Ke	Re	Sc	Th
1 St	.00	.38	.34	.37	.67	.64	.75	.86	.85
2 Br	.36	.00	.28	.29	.45	.53	.57	.75	.76
3 Gi	.36	.28	.00	.22	.53	.51	.57	.72	.74
4 So	.37	.29	.22	.00	.45	.50	.56	.69	.71
5 Oc	.66	.49	.49	.45	.00	.33	.29	.46	.46
6 Ke	.70	.55	.55	.53	.31	.00	.23	.42	.41
7 Re	.70	.55	.55	.53	.31	.23	.00	.34	.32
8 Sc	.86	.74	.74	.70	.46	.42	.33	.00	.21
9 Th	.86	.74	.74	.70	.46	.42	.33	.21	.00

Table 1. Dissimilarities among the nine Supreme Court justices above the diagonal; best-fitting SAR values below the diagonal.

3 Computational procedures within MATLAB

The recent monograph by Hubert, Arabie, and Meulman (2006) provides a collection of open-source M-files (i.e., the code is freely available) within a MATLAB environment to effect a variety of least-squares structural representations for a proximity matrix. Among these are strategies to search for good-fitting AR and SAR forms, including additive decompositions of up to two such structures for a single given proximity matrix. We do not give the algorithmic details here on how these M-files are built, and instead, refer the reader to the Hubert et. al (2006) monograph. We have collected all the relevant M-files together at `http://cda.psych.uiuc.edu/diday_mfiles`. The three M-files, `arobfnd.m`, `biarobfnd.m`, `triarobfnd.m`, fit respectively, one, two, and three AR matrices to a given input proximity matrix; the three M-files, `sarobfnd.m`, `bisarobfnd.m`, `trisarobfnd.m`, are for the strengthened SAR forms. The two files, `triarobfnd.m` and `trisarobfnd.m`, are unique to this site, and should provide a programming template to extend easily, when needed, the additive decomposition to four or more matrices.

We give the help header for the representative file `triarobfnd.m` below, along with an application to a randomly constructed 10×10 proximity matrix (obtained from the contributed M-file `randprox.m`). As can be seen, the (random) matrix is perfectly reconstructed by the three AR matrices (a variance-accounted-for of 1.0 is achieved). For example, the (4,6) entry in `prox` of .7948 is reconstructed based on the given output permutations,

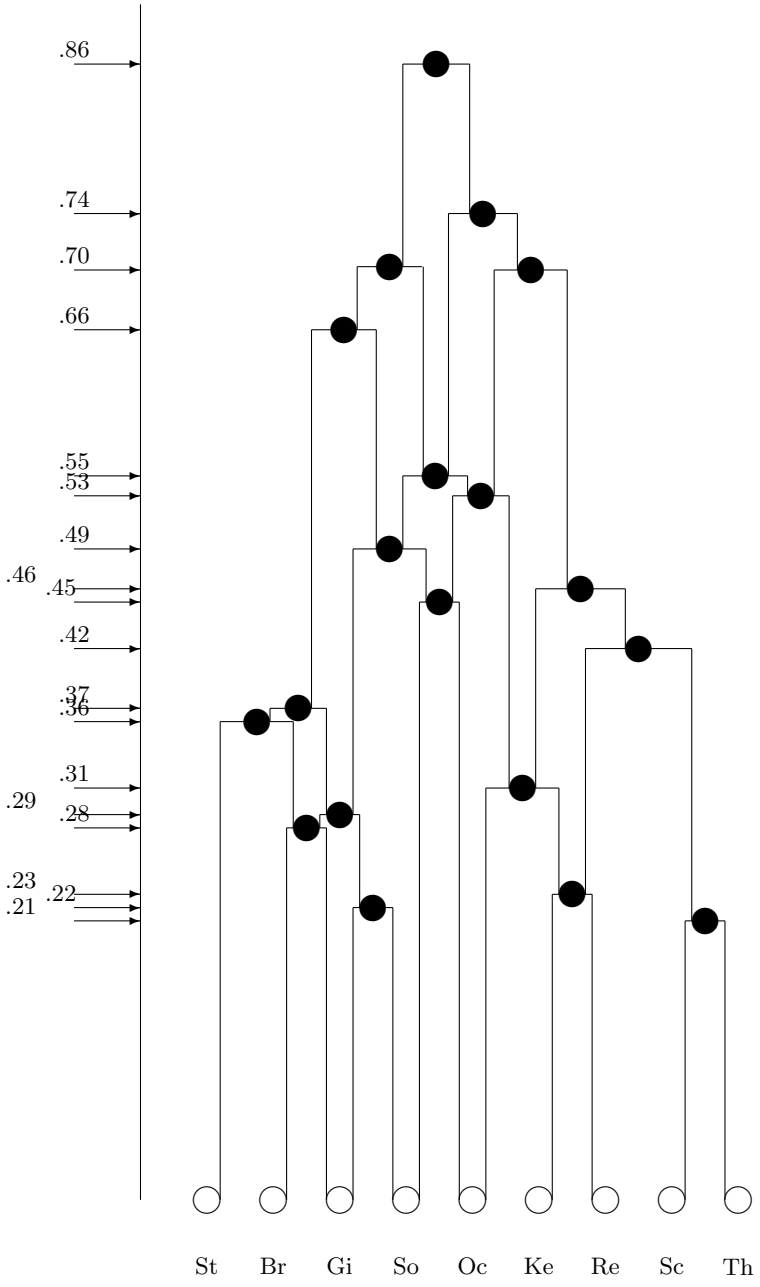

Fig. 1. A 'pyramidal' representation for the SAR matrix given in Table 1 having VAF of 98.62%

outpermone, outpermtwo, and outpermthree; explicitly, we use the (4,10) entry in targone (.8290), the (8,9) entry in targtwo (−.0515), and the (3,9) entry in targthree (.0173): .7948 = .8290 + (−.0515) + (.0173).

```
>> help triarobfnd

  TRIAROBFND finds and fits the sum of three 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,targthree,outpermone, ...
      outpermtwo,outpermthree] = triarobfnd(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
  three anti-Robinson matrices TARGONE, TARGTWO, and TARGTHREE
  based on the three row and column object orderings given by
  the ending permutations OUTPERMONE, OUTPERMTWO, and
  OUTPERMTHREE. KBLOCK defines the block size in the use of
  the iterative quadratic assignment routine.

>> prox = randprox(10)

prox =

         0    0.6979    0.3784    0.8600    0.8537    0.5936    0.4966    0.8998    0.8216    0.6449
    0.6979         0    0.8180    0.6602    0.3420    0.2897    0.3412    0.5341    0.7271    0.3093
    0.3784    0.8180         0    0.8385    0.5681    0.3704    0.7027    0.5466    0.4449    0.6946
    0.8600    0.6602    0.8385         0    0.6213    0.7948    0.9568    0.5226    0.8801    0.1730
    0.8537    0.3420    0.5681    0.6213         0    0.9797    0.2714    0.2523    0.8757    0.7373
    0.5936    0.2897    0.3704    0.7948    0.9797         0    0.1365    0.0118    0.8939    0.1991
    0.4966    0.3412    0.7027    0.9568    0.2714    0.1365         0    0.2987    0.6614    0.2844
    0.8998    0.5341    0.5466    0.5226    0.2523    0.0118    0.2987         0    0.4692    0.0648
    0.8216    0.7271    0.4449    0.8801    0.8757    0.8939    0.6614    0.4692         0    0.9883
    0.6449    0.3093    0.6946    0.1730    0.7373    0.1991    0.2844    0.0648    0.9883         0

>> [find,vaf,targone,targtwo,targthree, ...
    outpermone,outpermtwo,outpermthree] = ...
    triarobfnd(prox,randperm(10),2)

find =

         0    0.6979    0.3784    0.8600    0.8536    0.5936    0.4966    0.8998    0.8216    0.6449
    0.6979         0    0.8180    0.6602    0.3420    0.2897    0.3412    0.5341    0.7271    0.3093
    0.3784    0.8180         0    0.8385    0.5681    0.3704    0.7027    0.5466    0.4449    0.6946
    0.8600    0.6602    0.8385         0    0.6213    0.7948    0.9568    0.5226    0.8801    0.1730
    0.8536    0.3420    0.5681    0.6213         0    0.9797    0.2714    0.2523    0.8757    0.7373
    0.5936    0.2897    0.3704    0.7948    0.9797         0    0.1365    0.0118    0.8939    0.1991
    0.4966    0.3412    0.7027    0.9568    0.2714    0.1365         0    0.2987    0.6614    0.2844
    0.8998    0.5341    0.5466    0.5226    0.2523    0.0118    0.2987         0    0.4692    0.0648
    0.8216    0.7271    0.4449    0.8801    0.8757    0.8939    0.6614    0.4692         0    0.9883
```

```
   0.6449   0.3093   0.6946   0.1730   0.7373   0.1991   0.2844   0.0648   0.9883        0

vaf =

    1.0000

targone =

        0   0.6591   0.6591   0.6601   0.6601   0.7509   0.7754   0.7755   0.8757   0.8801
   0.6591        0   0.3569   0.5849   0.6601   0.7509   0.7509   0.7755   0.8290   0.8290
   0.6591   0.3569        0   0.3704   0.6601   0.6720   0.6851   0.7755   0.7840   0.8290
   0.6601   0.5849   0.3704        0   0.1030   0.2063   0.2661   0.3883   0.7840   0.8290
   0.6601   0.6601   0.6601   0.1030        0   0.2063   0.2418   0.3883   0.4269   0.8290
   0.7509   0.7509   0.6720   0.2063   0.2063        0   0.0283   0.3290   0.3290   0.6651
   0.7754   0.7509   0.6851   0.2661   0.2418   0.0283        0   0.2702   0.3290   0.5290
   0.7755   0.7755   0.7755   0.3883   0.3883   0.3290   0.2702        0   0.2963   0.5263
   0.8757   0.8290   0.7840   0.7840   0.4269   0.3290   0.3290   0.2963        0   0.5263
   0.8801   0.8290   0.8290   0.8290   0.8290   0.6651   0.5290   0.5263   0.5263        0

targtwo =

        0  -0.1489   0.0312   0.0312   0.0312   0.0492   0.0578   0.1813   0.2296   0.4148
  -0.1489         0  -0.1392  -0.0471  -0.0333   0.0492   0.0578   0.0578   0.1344   0.1344
   0.0312  -0.1392         0  -0.0537  -0.0333   0.0281   0.0376   0.0376   0.0376   0.0620
   0.0312  -0.0471  -0.0537         0  -0.2446   0.0281   0.0376   0.0376   0.0376   0.0620
   0.0312  -0.0333  -0.0333  -0.2446         0  -0.2488  -0.1600   0.0376   0.0376   0.0620
   0.0492   0.0492   0.0281   0.0281  -0.2488         0  -0.1600  -0.0080   0.0160   0.0160
   0.0578   0.0578   0.0376   0.0376  -0.1600  -0.1600         0  -0.3058  -0.0080        0
   0.1813   0.0578   0.0376   0.0376   0.0376  -0.0080  -0.3058         0  -0.0515  -0.0426
   0.2296   0.1344   0.0376   0.0376   0.0376   0.0160  -0.0080  -0.0515         0  -0.3495
   0.4148   0.1344   0.0620   0.0620   0.0620   0.0160         0  -0.0426  -0.3495        0

targthree =

        0  -0.1217  -0.0376  -0.0312   0.0346   0.0346   0.1510   0.1958   0.1962   0.1962
  -0.1217         0  -0.1345  -0.1345   0.0346   0.0346   0.0364   0.1113   0.1113   0.1675
  -0.0376  -0.1345         0  -0.1345  -0.0065  -0.0065  -0.0065  -0.0065   0.0173   0.0964
  -0.0312  -0.1345  -0.1345         0  -0.2651  -0.0065  -0.0065  -0.0065   0.0145   0.0145
   0.0346   0.0346  -0.0065  -0.2651         0  -0.0065  -0.0065  -0.0065   0.0080   0.0145
   0.0346   0.0346  -0.0065  -0.0065  -0.0065         0  -0.0917  -0.0243  -0.0243        0
   0.1510   0.0364  -0.0065   0.0065  -0.0065  -0.0917         0  -0.1680  -0.0243  -0.0229
   0.1958   0.1113  -0.0065  -0.0065  -0.0065  -0.0243  -0.1680         0   0.0289  -0.0239
   0.1962   0.1113   0.0173   0.0145   0.0080  -0.0243  -0.0243  -0.0289         0  -0.1362
   0.1962   0.1675   0.0964   0.0145   0.0145         0  -0.0229  -0.0239  -0.1362        0

outpermone =

     9     1     3     6     7     8    10     2     5     4

outpermtwo =

     5     7     1     2     9     3     8     6     4    10

outpermthree =

     9     8     4     5     3     7    10     1     6     2
```

4 The concept of minimum AR (or SAR) matrix rank

Based on the type of M-file (`triarobfnd.m`) illustrated in the previous section, a rather natural question arises as to the number of AR (or SAR) components necessary to exhaust perfectly any given proximity matrix. The minimum such number will be referred to as the AR (or SAR) rank of a symmetric proximity matrix. As we saw for the random 10×10 matrix in the example of the last section, we usually can do quite well with many fewer components than the order of the matrix. Although we might expect this to be true for a data matrix that is well-structured (and where two or three AR or SAR components are all that is needed to effectively exhaust the given proximity matrix), the same also appears to hold for merely randomly structured matrices.

To make this last point even more clear, a small Monte Carlo analysis was carried out in which 1000 random proximity matrices (with entries uniform on (0,1)), of sizes 10, 20, 30, 40, and 50, were approximated by sums of AR matrices to the point where at least a VAF of 99% was achieved. The frequency results (out of 1000 such randomly generated matrices) are tabulated below:

	Number AR Components Needed								
Matrix Size	2	3	4	5	6	7	8	9	10
10	37	959	4						
20			316	684					
30					994	6			
40						205	795		
50								995	5

Figure 2 illustrates, by means of box-and-whisker plots, the incremental gain in VAF as a function of the number of fitted AR components.

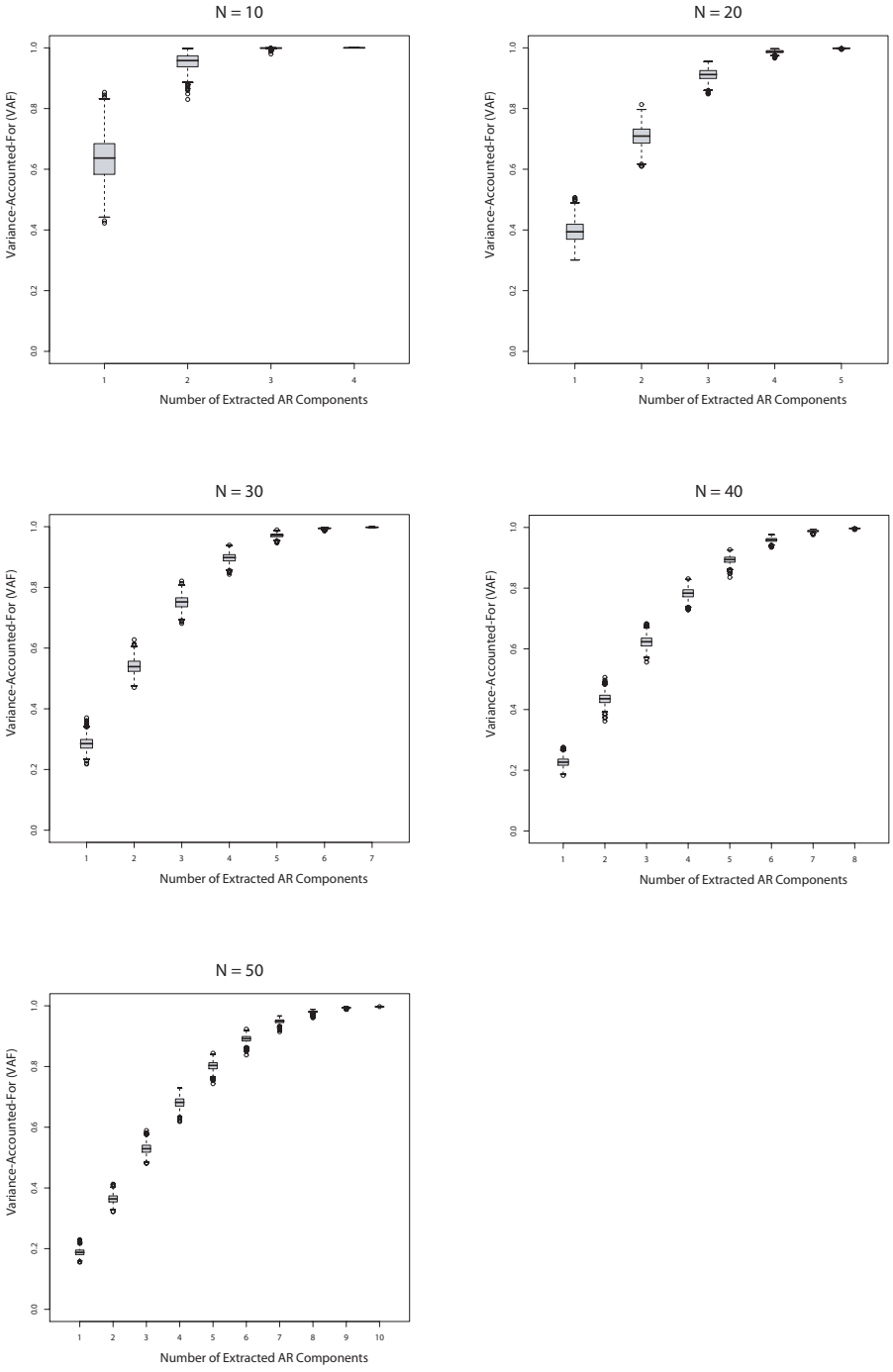

Fig. 2. Incremental VAF Gains for Differing Numbers of AR Components

References

BARTHÉLEMY, J.-P., BRUCKER, F. and OSSWALD, C. (2004): Combinatorial optimization and hierarchical classifications. *4OR: A Quarterly Journal of Operations Research 2 (3), 179–219.*

CRITCHLEY, R. (1994): On exchangeability-based equivalence relations induced by strongly Robinson and, in particular, by quadripolar Robinson dissimilarity matrices. In: B. van Cutsem (Ed.): *Classification and Dissimilarity Analysis.* Springer-Verlag, New York, 173–199.

CRITCHLEY, R. and FICHET, B. (1994): The partial order by inclusion of the principal classes of dissimilarity on a finite set, and some of their basic properties. In: B. van Cutsem (Ed.): *Classification and Dissimilarity Analysis.* Springer-Verlag, New York, 5–65.

DIATTA, J. and FICHET, B. (1998): Quasi-ultrametrics and their 2-ball hypergraphs. *Discrete Mathematics 192 (1-3), 87–102.*

DIDAY, E. (1986): Orders and overlapping clusters by pyramids. In: J. De Leeuw, W. Heiser, J. Meulman and F. Critchley (Eds.): *Multidimensional Data Analysis.* DSWO Press, Leiden, 201–234.

DURAND, C. and FICHET, B. (1988): One-to-one correspondences in pyramidal representations: A unified approach. In: H. H. Bock (Ed.): *Classification and Related Methods of Data Analysis.* North-Holland, Amsterdam, 85–90.

HUBERT, L., ARABIE, P. and MEULMAN, J. (2006): *The Structural Representation of Proximity Matrices with MATLAB.* SIAM, Philadelphia.

MIRKIN, B. (1996): *Mathematical Classification and Clustering.* Kluwer, Dordrecht.

ROBINSON, W. S. (1951): A method for chronologically ordering archaeological deposits. *American Antiquity 19 (4), 293–301.*