

THE STRUCTURAL REPRESENTATION OF THREE-WAY
PROXIMITY DATA

BY

HANS F. KOEHN

B.A. (Vordiplom), University of Cologne, Germany, 1984
M.A. (Diplom), University of Hamburg, Germany, 1988
M.S., University of Illinois, Urbana-Champaign, 2000

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Psychology
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2007

Urbana, Illinois

UMI Number: 3290276

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI[®]

UMI Microform 3290276

Copyright 2008 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

CERTIFICATE OF COMMITTEE APPROVAL

*University of Illinois at Urbana-Champaign
Graduate College*

May 7, 2007

We hereby recommend that the thesis by:

HANS F. KOEHN

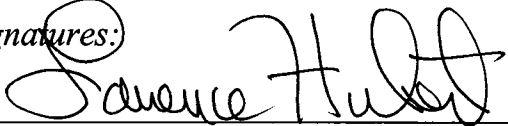

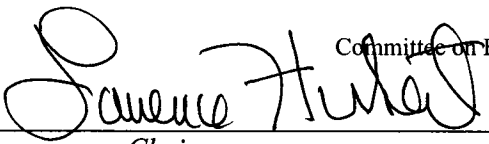
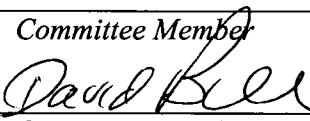
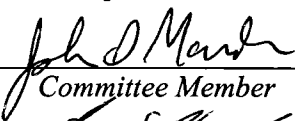
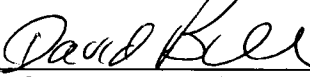
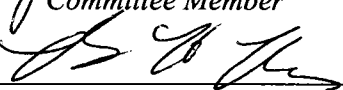
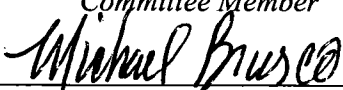
Entitled:

**THE STRUCTURAL REPRESENTATION OF THREE-WAY PROXIMITY
DATA**

Be accepted in partial fulfillment of the requirements for the degree of:

Doctor of Philosophy

Signatures:

 _____ Director of Research	 _____ Head of Department
 _____ Chairperson	 _____ Committee Member
 _____ Committee Member	 _____ Committee Member
 _____ Committee Member	 _____ Committee Member

Committee on Final Examination*

* Required for doctoral degree but not for master's degree

© 2007 by Hans F. Koehn. All rights reserved.

Abstract

Scaling and clustering techniques are well-established statistical methods for generating continuous and discrete structural representations of the relationships between the row and column objects of proximity matrices. Most commonly, the representational structure is fit to the observed data through minimizing the least-squares loss function; traditional implementations rely typically on gradient or sub-gradient optimization. Alternatively, scaling and clustering can be reformulated as combinatorial data analytic tasks, solvable through discrete optimization strategies. We develop generalizations of combinatorial algorithms for analyzing individual differences through scaling and clustering three-way data that consist of collections of proximity matrices observed on multiple sources. We propose an approach derived from a deviation-from-the-mean principle. Order-constrained matrix decomposition can be regarded as a combinatorial data analytic meta-technique, providing a unifying framework for evaluating the differential merits of continuous and discrete structural representations of proximity matrices. We introduce a generalization of order-constrained matrix decomposition to accommodate three-way proximity data. Multiobjective programming, as an alternative approach to modelling three-way data, is presented, accompanied by a survey of existing applications in the psychometric literature.

To my Grandmother, Hildegard Köhn, and Friedi, simply for being — und überhaupt.

Acknowledgments

I would like to express my deepest appreciation to Professor Lawrence Hubert for his inspiration, guidance, and support in making this thing really happen — Larry, I owe you for a life-time. Also, many, many thanks to my parents, Mechthild Köhn-Jonas and Paul-Werner Jonas, for all their help during the sometimes not so easy stages of this undertaking.

In addition, I would like to express my gratitude to my thesis committee members: Professors David Budescu, Mike Brusco, John Marden, and Brian Ross. They were of great encouragement in the completion of this project.

Table of Contents

List of Tables	ix
List of Figures	x
1 Introduction	1
2 Theoretical Preliminaries	6
2.1 Definitions and Concepts	6
2.1.1 Proximity Data	6
2.1.2 Structural Representation	6
2.1.3 Combinatorial Data Analysis and Structural Representations	8
2.1.4 Extensions of Combinatorial Structural Representations to Three-Way Proximity Matrices	9
2.2 Computational Devices	10
2.2.1 Quadratic Assignment	10
2.2.2 Iterative Projection	12
3 Additive Tree Representations	15
3.1 Additive Tree Structures	16
3.1.1 Definitions and Concepts	16
3.1.2 Additive Tree Representations as Constrained Least-Squares Minimization Problems	18
3.1.3 Multiple Additive Tree Structures	19
3.2 Locating Additive Trees Through the Penalty Function Algorithm	19
3.3 Locating Additive Trees Through Iterative Projection	20
3.3.1 Non-Heuristic Iterative Projection	21
3.3.2 Heuristic Iterative Projection	22
3.4 Locating Additive Trees for Three-Way Data	23
3.5 Applications	24
3.5.1 Three-Way Additive Tree Structures for Schematic Face Stimuli	25
3.6 Discussion and Conclusion	30
4 Multidimensional Scaling	33
4.1 The Weighted Euclidean and the City-Block Model for Individual Differences Scaling	34
4.2 Combinatorial Scaling	35
4.2.1 Combinatorial Unidimensional Scaling	36
4.2.2 Combinatorial Multidimensional Scaling	40
4.2.3 Combinatorial Individual Differences Scaling	41
4.3 Application: Scaling of Schematic Faces	41
4.4 Discussion and Conclusion	46
4.4.1 Discussion	46
4.4.2 Concluding Comments	48

5	Unfolding	55
5.1	Theory and Methods of Unfolding	56
5.1.1	Scalar Product versus Distance Models of Unfolding	57
5.1.2	Combinatorial Unfolding	60
5.2	Degeneracy	62
5.2.1	PREFSCAL: Avoiding Degeneracy by Penalizing the Loss Function	62
5.2.2	Evaluation: PREFSCAL versus Combinatorial Unfolding	63
5.3	Extensions of the Unfolding Model	66
5.3.1	Three-way Combinatorial Unfolding	68
5.3.2	Application: The Contraceptive Data from Weller and Romney (1990)	68
5.4	Discussion and Conclusion	70
6	Order-Constrained Matrix Decomposition	83
6.1	Order-Constrained Anti-Robinson Matrix Decomposition	86
6.1.1	Definitions and Formal Concepts	86
6.1.2	Optimal AR Decomposition: Algorithmic and Computational Details	87
6.1.3	Low AR Rank Approximation to a Proximity Matrix	89
6.1.4	Fit Measure	89
6.1.5	The Representation of AR Matrix Components Through Secondary Structures	89
6.1.6	Extending AR Matrix Decomposition to Three-Way Data	91
6.1.7	Application: Judgments of Schematic Face Stimuli	93
6.2	Order-Constrained (Anti-)Q Matrix Decomposition	107
6.2.1	Definitions and Formal Concepts	107
6.2.2	Optimal Anti-Q Decomposition: Algorithmic and Computational Details	108
6.2.3	Low-Anti-Q-Rank Approximation to a Proximity Matrix	110
6.2.4	Fit Measure	110
6.2.5	The Representation of Anti-Q Matrix Components Through Secondary Structures	111
6.2.6	Extending Anti-Q Matrix Decomposition to Three-Way Data	112
6.2.7	Application: The Contraceptive Data from Weller and Romney (1990)	113
6.3	Discussion and Conclusion	116
7	Multiobjective Programming	120
7.1	Theory and Concepts	120
7.1.1	Single- versus Multiobjective Programming	121
7.1.2	The Notion of 'Optimality'	122
7.1.3	Efficiency and Nondominance	124
7.1.4	Proper Efficiency and Nondominance	128
7.1.5	Properties of Efficient and Nondominated Sets	128
7.1.6	Solution Methods: Generation of Pareto-Optimal Sets	129
7.2	Applications of Multiobjective Programming in Quantitative Psychology	133
7.2.1	Combinatorial Optimization Problems	133
7.2.2	Combinatorial Multiobjective Programming	134
7.2.3	Applications	135
7.3	Conclusion and Outlook	137

Appendix I: Guttman's (1968) Gradient-Based Algorithm . . .	140
Appendix II: Re-Expressing the Least-Squares Loss Function .	144
Appendix III: Defays' (1978) Short Note on a Method of Seriation	152
References	160
Vita	175

List of Tables

1.1	The 15 contraceptive methods.	4
1.2	The contraceptive data.	5
1.3	The contraceptive data matrices aggregated for the female, male, and total sample.	5
2.1	IP Sequences of $\mathbf{x}^{(t)}$ and $\mathbf{e}^{(t)}$ for $W = 2$ constraint sets, \mathcal{C}_1 and \mathcal{C}_2	14
3.1	Heuristic additive tree fitting to the aggregated proximity data, \mathbf{P}_a : frequency distributions of the VAF scores, as obtained from 10,000 random starts.	25
3.2	Ranking of individual fit measures.	28
4.1	Ranking of individual fit measures.	45
5.1	Summary of unfolding models fit to data matrices generated from the original Coombs and Kao (1960) random coordinates of 30 row and 15 column objects.	64
6.1	Frequency distributions of the VAF scores as obtained from 10,000 random starts for $K = \{1, 2, 3, 4\}$	93
6.2	Ranking of Individual VAF scores plus various fit measures quan- tifying the separate contributions of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$	98
6.3	Individual VAF scores for the unidimensional scale (US) and ul- trametric tree (UT) representations of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$	99
6.4	Independent biadditive AR decomposition of selected sources: In- dividual VAF scores plus various fit measures quantifying the separate contributions of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$	103
6.5	Individual VAF scores for the unidimensional scale (US) and ul- trametric tree (UT) representations of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$ from the independent analysis of selected sources.	103
6.6	Frequency distributions of the VAF scores as obtained from 10,000 random starts on <i>TOTAL</i> for $K = \{1, 2\}$	113
6.7	VAF scores for the unidimensional unfolding (UU) and two-mode ultrametric tree (UT) representations of the fitted anti-Q com- ponents of <i>TOTAL</i> , <i>FEMALE</i> and <i>MALE</i>	114
6.8	VAF scores for the unidimensional unfolding (UU) and two-mode ultrametric tree (UT) representations of the fitted anti-Q com- ponents from the independent analysis of <i>FEMALE</i> and <i>MALE</i>	115

List of Figures

1.1	The construction of schematic face stimuli.	3
1.2	Dissimilarity ratings of schematic face stimuli aggregated across 22 subjects.	4
3.1	Biadditive reference tree representation (VAF=.8521) of the aggregated proximity matrix, \mathbf{P}_a	27
3.2	Individual biadditive tree representations for selected subjects: first structure.	29
3.3	Independent biadditive tree representations for the bottom three subjects: first structure.	31
4.1	Group Spaces	44
4.2	Private spaces for subject 12.	47
4.3	Private spaces for subject 1.	48
4.4	Private spaces for subject 14.	49
4.5	Private spaces for subject 2.	50
4.6	Exploratory spaces for subject 1.	51
4.7	Exploratory spaces for subject 14.	52
4.8	Exploratory spaces for subject 2.	53
5.1	Metric and non-metric combinatorial unfolding analysis of Euclidean and city-block distance matrices computed from the Coombs-Kao (1960) standard-normal random coordinates — unperturbed.	73
5.2	Metric and non-metric combinatorial unfolding analysis of Euclidean and city-block distance matrices computed from the Coombs-Kao (1960) standard-normal random coordinates — 25% error-perturbed.	74
5.3	Metric and non-metric combinatorial unfolding analysis of the Brewery Data (Borg & Bergermaier, 1982).	75
5.4	Metric and non-metric combinatorial unfolding analysis of the Breakfast Data (Green & Rao, 1972): Toast Pop-up (TP), Buttered Toast (BT), English Muffin and Margarine (EMM), Jelly Donut (JD), Cinnamon Toast (CT), Blueberry Muffin and Margarine (BMM), Hard Rolls and Butter (HRB), Toast and Marmalade (TMd), Buttered Toast and Jelly (BTj), Toast and Margarine (TMn), Cinnamon Bun (CB), Danish Pastry (DP), Glazed Donut (GD), Coffee Cake (CC), Corn Muffin and Butter (CMB).	76
5.5	Metric Unfolding: reference space based on matrix <i>TOTAL</i> , and corresponding confirmatory spaces of matrices <i>FEMALE</i> and <i>MALE</i>	77
5.6	Nonmetric Unfolding: reference space based on matrix <i>TOTAL</i> , and corresponding confirmatory spaces of matrices <i>FEMALE</i> and <i>MALE</i>	78

5.7	Metric Unfolding: exploratory spaces of matrices <i>FEMALE</i> and <i>MALE</i>	79
5.8	Nonmetric Unfolding: exploratory spaces of matrices <i>FEMALE</i> and <i>MALE</i>	80
5.9	Metric Unfolding: reference space based on matrix <i>FEMALE</i> , and corresponding confirmatory space of matrix <i>MALE</i>	81
5.10	Nonmetric Unfolding: reference space based on matrix <i>FEMALE</i> , and corresponding confirmatory space of matrix <i>MALE</i>	82
6.1	The orders of schematic face stimuli obtained for the various Z_k components of the AR decompositions into K components with maximum VAF score.	94
6.2	Biadditive reference AR decomposition of P_a (VAF=.959): unidimensional scale and ultrametric tree representations.	96
6.3	Confirmatory biadditive AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the first AR component $Z_{1(s)}$	101
6.4	Confirmatory biadditive reference AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the second AR component $Z_{2(s)}$	102
6.5	Independent biadditive AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the first AR component $Z_{1(s)}$	105
6.6	Independent biadditive AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the second AR component $Z_{2(s)}$	106
6.7	Unidimensional unfolding and two-mode ultrametric tree graphs of the reference anti-Q component extracted from <i>TOTAL</i> , with confirmatory counterparts for <i>FEMALE</i> and <i>MALE</i>	118
6.8	Unidimensional unfolding and two-mode ultrametric tree graphs of the anti-Q components extracted through the independent analyses of <i>FEMALE</i> and <i>MALE</i> and confirmatory anti-Q component from fitting the male data set against the female reference structure.	119
7.1	Example: Notation	125

1 Introduction

In his seminal paper, Tversky (1977) presents a thorough theoretical treatment of the concept of similarity, emphasizing its eminent role “in theories of knowledge and behavior. It serves as an organizing principle by which individuals classify objects, form concepts, and make generalizations. Indeed, the concept of similarity is ubiquitous in psychological theory. It underlies the accounts of stimulus and response generalization in learning, it is employed to explain errors in memory and pattern recognition, and it is central to the analysis of connotative meaning” (p. 327). Similarity and its complement, dissimilarity, are typically subsumed under the notion of proximity.

In its broadest sense, the term ‘proximity’, refers to any numerical measure of relationship between the elements of a pair from two (possibly distinct) sets of entities or objects. Proximities are typically collected into a matrix, with rows and columns representing the respective sets of objects, and the numerical cell values the observed pairwise proximity scores. Proximity data possess the attractive feature that their collection is straightforward, while, at the same time — despite their holistic nature — they yield rich and detailed information concerning the constituting features of a proximity evaluation. In fact, the application of appropriate statistical techniques allows to infer (post hoc) a set of perceptual criteria (and their relative weights) that informed a subject’s proximity judgments, and determine the cognitive structure underlying the perception of an object domain. As our ultimate analytic goal, we seek to generate a structural representation of the relationship between the objects under study, either in the form of a discrete network or a continuous spatial configuration. Substantively, the two models address distinct perspectives on how objects are mentally represented.

In most practical applications, the proximity data are aggregated into a single matrix across multiple data sources such as subjects, scenarios, experimental conditions, time points, and so on. However, if the focus of a study lies on individual variability among observational units, we may want to analyze the data at a non-aggregated level, where the entire body of data consists of a cube, with multiple individual proximity matrices stacked as slices along the third dimension. The complexity of the non-aggregated data layout prevents the immediate application of analytic methods suitable for a single proximity matrix and calls for appropriate technical adjustments (see Coppi & Bolasco, 1989; Law, Snyder, Hattie, & McDonald, 1984; Smilde, Bro, & Geladi,

2004). Loosely speaking, their common principle consists in modelling individual variation against a reference structure derived from the entire data set. The INDSCAL implementation (Carroll & Chang, 1970) of the weighted Euclidean model, a generalization of multidimensional scaling for analyzing individual differences as scaled representations of a (continuous) group space obtained from the total data set, presumably represents the most famous instance of such a methodological transfer from a single proximity matrix to a setting characterized by multiple data matrices.

In the dissertation presented in the following chapters, we develop combinatorial data analytic techniques for the structural representation of individual differences based on multiple data sources. The terms ‘structural representation’ and ‘combinatorial data analysis’ warrant further clarification, to be offered in Chapter 2. We also introduce the general framework developed for accommodating extensions of combinatorial representation techniques to multiple proximity matrices. Lastly, Chapter 2 provides a technical description of the two main computational devices employed for solving combinatorial data analytic tasks, namely, quadratic assignment and iterative projection. Chapter 3 introduces additive tree representations for three-way data. Chapter 4 presents an extension of (city-block) multidimensional scaling to three-way data. Chapter 5 develops a distance-based unfolding model for three-way data. So far, distance-based unfolding models cannot necessarily be regarded as an overwhelming success due to the unresolved issue of degeneracy. However, combinatorial unfolding promises to successfully remedy the problem of degenerate solutions. Therefore, we decided to include a section solely devoted to degeneracy — although the relation to three-way analysis at first might appear remote. Order-constrained matrix decomposition can be regarded as a combinatorial data-analytic meta-technique for constructing and evaluating structural representations. Chapter 6 develops order-constrained matrix decomposition for three-way data. The total variability of a given proximity matrix is represented by a minimal number of additive components, each determined by a distinct object ordering that translates into discrete and continuous representations immediately comparable in terms of formal fit as well as substantive interpretability. The extension of order-constrained matrix decomposition to three-way data provides the analyst with an instrument to explore complex hypotheses concerning the appropriateness of continuous versus discrete stimuli representations from an interindividual as well as intraindividual perspective. Chapter 7, as conclusion and outlook on future research directions, introduces multiobjective programming as a viable alternative for the analysis of individual differences based on three-way data.

Two data sets will be used repeatedly as numerical examples. The Face Data consist of a collection of 22 square-symmetric proximity matrices, containing the pairwise dissimilarity ratings of 12 schematic faces observed on 22 graduate students in the Psychology department of the University of Illinois. The 66 pairs of schematic faces were displayed on a computer screen and had to be evaluated

Krantz (1969), the twelve face stimuli were generated by completely crossing the three factors Facial Shape, Eyes, and Mouth (see Figure 1.1).

Figure 1.1: The construction of schematic face stimuli.













Facial Shape	Eyes	Mouth		
		smile	flat	frown
circle	open	 ₁	 ₂	 ₃
	solid	 ₄	 ₅	 ₆
oval	open	 ₇	 ₈	 ₉
	solid	 ₁₀	 ₁₁	 ₁₂

Figure 1.2 displays the matrix of mean dissimilarity ratings derived from the 22 individual data matrices. Each individual's ratings were first converted into z-scores (i.e., subtracting a person's mean rating from each of his or her 66 dissimilarity ratings, and dividing these differences by his or her standard deviation). The z-scores were then averaged across the 22 respondents for each pair of faces. Lastly, to eliminate negative numbers, and to put the aggregated dissimilarities back onto a scale similar to the original nine-point scale, the average z-score ratings were multiplied by 2, and a constant value of 4 was added.

The contraceptive data from Weller and Romney (1990) were collected from two female (F1 and F2) and two male samples (M1 and M2), each consisting of seven individuals who ranked fifteen contraceptive methods (see Table 1.1) from 'least' (= rank 1) to 'most' (= rank 15) with respect to the four criteria of perceived effectiveness (E), safety (S), availability (A), and convenience (C). The rankings were aggregated into a rectangular 16×15 two-way two-mode data matrix, with rows containing the sum of all rankings for a specific criterion within a particular sample (i.e., EF1 = rankings with respect to effectiveness from the first female sample; EF2 = second female sample; EM1 = first male sample, and so on). The original data were transformed into dissimilarities by subtracting each cell entry from $15(7) = 105$, and are given in Table 1.2. By

Figure 1.2: Dissimilarity ratings of schematic face stimuli aggregated across 22 subjects.













											
0.00	2.21	3.45	2.04	4.19	4.88	1.64	4.29	4.89	4.75	5.92	6.22
2.21	0.00	2.43	3.55	2.08	4.49	4.08	2.05	3.98	5.48	4.06	6.30
3.45	2.43	0.00	4.89	4.76	1.97	5.24	4.50	2.61	6.77	5.79	3.93
2.04	3.55	4.89	0.00	2.14	3.37	4.31	5.82	6.17	1.64	3.78	4.92
4.19	2.08	4.76	2.14	0.00	2.90	5.85	3.82	6.08	4.42	2.24	4.80
4.88	4.49	1.97	3.37	2.90	0.00	6.76	6.08	3.87	4.81	5.03	2.16
1.64	4.08	5.24	4.31	5.85	6.76	0.00	2.50	3.00	1.80	4.30	4.87
4.29	2.05	4.50	5.82	3.82	6.08	2.50	0.00	3.11	3.94	1.90	4.60
4.89	3.98	2.61	6.17	6.08	3.87	3.00	3.11	0.00	5.02	4.43	1.93
4.75	5.48	6.77	1.64	4.42	4.81	1.80	3.94	5.02	0.00	2.27	3.33
5.92	4.06	5.79	3.78	2.24	5.03	4.30	1.90	4.43	2.27	0.00	2.64
6.22	6.30	3.93	4.92	4.80	2.16	4.87	4.60	1.93	3.33	2.64	0.00

Table 1.1: The 15 contraceptive methods.

	Letter Code	Contraceptive Method
1	Pil	pill
2	Con	condom
3	Iud	intrauterine device
4	Dia	diaphragm
5	Foa	foam
6	Rhy	rhythm
7	Abs	abstinence
8	Wit	withdrawal
9	Vas	vasectomy
10	Tub	tubal ligation
11	Abo	abortion
12	Dou	douche
13	Ora	oral sex
14	Spe	spermicide
15	Hys	hysterectomy

collapsing rows of the original data matrix of Table 1.2, the data set can be re-arranged into three-way three-mode format; for our illustrative purpose, we choose the split according to gender, female versus male (see Table 1.3), yielding a $4 \times 15 \times 2$ data cube (of course, the *FEMALE* and *MALE* submatrices sum

Table 1.2: The contraceptive data.

	Pil	Con	Iud	Dia	Foa	Rhy	Abs	Wit	Vas	Tub	Abo	Dou	Ora	Spe	Hys
EF1	31	58	43	49	66	64	24	90	18	23	57	90	21	71	30
EF2	28	48	45	42	61	82	13	81	22	22	77	84	39	66	25
EM1	26	45	66	50	52	65	9	75	31	43	70	73	33	52	45
EM2	54	43	24	37	43	66	61	48	36	29	60	69	64	49	52
SF1	77	35	71	47	43	10	2	17	72	78	94	35	16	48	90
SF2	77	25	79	44	51	20	13	22	44	64	89	45	23	62	77
SM1	52	33	64	36	38	32	8	26	66	71	93	51	27	59	79
SM2	31	20	27	35	48	43	34	43	66	67	75	65	64	54	63
AF1	62	35	63	47	35	13	21	34	85	82	90	29	12	36	91
AF2	51	34	68	57	36	30	13	12	79	84	79	36	19	46	91
AM1	31	11	57	53	21	39	52	42	66	68	61	36	66	47	85
AM2	51	16	49	16	45	37	60	38	65	60	63	56	51	58	70
CF1	31	24	35	40	40	52	18	54	72	76	92	42	34	40	85
CF2	39	39	45	55	53	63	25	36	49	56	91	52	29	50	53
CM1	6	26	27	36	38	57	56	74	69	56	75	50	35	60	70
CM2	60	49	26	29	42	43	44	56	54	40	69	49	58	50	66

into a single 4×15 dissimilarity matrix, *TOTAL* — see Table 1.3).

Table 1.3: The contraceptive data matrices aggregated for the female, male, and total sample.

FEMALE															
	Pil	Con	Iud	Dia	Foa	Rhy	Abs	Wit	Vas	Tub	Abo	Dou	Ora	Spe	Hys
E	59	106	88	91	127	146	37	171	40	45	134	174	60	137	55
S	154	60	150	91	94	30	15	39	116	142	183	80	39	110	167
A	113	69	131	104	71	43	34	46	164	166	169	65	31	82	182
C	70	63	80	95	93	115	43	90	121	132	183	94	63	90	138

MALE															
	Pil	Con	Iud	Dia	Foa	Rhy	Abs	Wit	Vas	Tub	Abo	Dou	Ora	Spe	Hys
E	80	88	90	87	95	131	70	123	67	72	130	142	97	101	97
S	83	53	91	71	86	75	42	69	132	138	168	116	91	113	142
A	82	27	106	69	66	76	112	80	131	128	124	92	117	105	155
C	66	75	53	65	80	100	100	130	123	96	144	99	93	110	136

TOTAL															
	Pil	Con	Iud	Dia	Foa	Rhy	Abs	Wit	Vas	Tub	Abo	Dou	Ora	Spe	Hys
E	139	194	178	178	222	277	107	294	107	117	264	316	157	238	152
S	237	113	241	162	180	105	57	108	248	280	351	196	130	223	309
A	195	96	237	173	137	119	146	126	295	294	293	157	148	187	337
C	136	138	133	160	173	215	143	220	244	228	327	193	156	200	274

2 Theoretical Preliminaries

2.1 Definitions and Concepts

2.1.1 Proximity Data

By assumption, proximities are restricted to be nonnegative, and are, henceforth, consistently interpreted as dissimilarities so that larger numerical indices pertain to less similar pairs of objects. Inspired by concepts originally introduced by Tucker (1964), Carroll and Arabie (1980) established a widely accepted taxonomy of proximity matrices based on the number of ‘ways’ and ‘modes’. The former relates to the dimensionality of a matrix and the ‘mode’ to the number of sets of entities the ways correspond to. Formally, given a set of N objects, $\mathcal{O} = \{O_1, \dots, O_N\}$, the pairwise proximities observed on objects O_i, O_j are collected into an $N \times N$ square-symmetric two-way one-mode proximity matrix $\mathbf{P} = \{p_{ij}\}$, with $p_{ij} = p_{ji}$ and $p_{ii} = 0$ for $1 \leq i, j \leq N$. In contrast, a two-way two-mode proximity matrix $\mathbf{Q} = \{q_{rc}\}$ consists of a rectangular $N_R \times N_C$ data set, with $r = 1, \dots, N_R$ and $c = 1, \dots, N_C$, where rows and columns represent two distinct sets of entities, $\mathcal{O}_R = \{O_{1R}, \dots, O_{N_R R}\}$ and $\mathcal{O}_C = \{O_{1C}, \dots, O_{N_C C}\}$, containing N_R and N_C elements, respectively. The proximities q_{rc} between objects $O_{rR} \in \mathcal{O}_R$ and $O_{cC} \in \mathcal{O}_C$ can represent preference scores obtained from subjects judging objects, ratings of objects on attributes, and so on.

The cube formed by stacking proximity matrices from multiple data sources is often referred to as three-way two-mode or three-way three-mode proximities, depending on the data format of the individual sources, denoted $\mathbf{P}^{(S)} = \{p_{ijs}\}$ or $\mathbf{Q}^{(S)} = \{q_{rcs}\}$, respectively, by introducing the source index $s = 1, \dots, S$. The layers of individual sources are indicated by $\mathbf{P}_s = \{p_{ij(s)}\}$ and $\mathbf{Q}_s = \{q_{rc(s)}\}$.

2.1.2 Structural Representation

In pre-scientific jargon, the essence of what we will formalize in more rigorous terms as structural representation is best captured by colloquialisms such as ‘what goes with what’ or ‘what is greater or less than’. For our purposes, we choose the mathematical explication of the concept of structural representation in terms of a discrete non-spatial or continuous spatial model. The former specifies the relationship between row and column objects by a graph-theoretic network, with objects represented as nodes and proximities approximated by

paths of different length connecting the object nodes. Depending on the specific nature of constraints imposed on the path length estimates, the resulting graph either represents an ultrametric or additive tree structure, arranging objects into (nested) categories — commonly and less accurately, simply referred to as a hierarchical cluster diagram. In case of a continuous spatial model, the relationship between objects is represented through a (possibly multidimensional) geometric configuration of objects, ordering them along the axes of the representational space; the proximities are translated into interobject distances. Concretely, we will consider uni- and multidimensional scaling and their companion, distance-based unfolding.

As a common denominator, all structural representations can be treated as least-squares minimization problems. By introducing generic notation \mathbf{S} for any kind of structural representation, in case of a square-symmetric two-way one-mode dissimilarity matrix we express the least-squares loss function as

$$L(\mathbf{S}) = \sum_{i < j} (p_{ij} - s_{ij})^2 = \frac{1}{2} \text{tr}(\mathbf{P} - \mathbf{S})(\mathbf{P} - \mathbf{S})',$$

where tr denotes the trace function, defined for an arbitrary matrix, \mathbf{M} , as $\text{tr } \mathbf{M} = \sum_i m_{ii}$. For a rectangular two-way two-mode data matrix, the least-squares loss function is given by

$$L(\mathbf{S}) = \sum_r \sum_c (q_{rc} - s_{rc})^2 = \text{tr}(\mathbf{Q} - \mathbf{S})(\mathbf{Q} - \mathbf{S})'.$$

The fit of a particular structural representation is assessed through the variance-accounted-for (VAF) criterion, modelled after the least-squares loss function and equivalent to the coefficient of determination in regression, $R^2 = 1 - \text{SSE}/\text{SSTO}$:

$$\text{VAF} = 1 - \frac{\sum_{i < j} (p_{ij} - s_{ij})^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} = 1 - \frac{\text{tr}(\mathbf{P} - \mathbf{S})(\mathbf{P} - \mathbf{S})'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}})(\mathbf{P} - \bar{\mathbf{P}})'},$$

with $\bar{\mathbf{P}}$ denoting a square-symmetric matrix having zero entries along the main diagonal and all off-diagonal values equal to $\bar{p} = \frac{1}{N(N-1)} \sum_{i \neq j} p_{ij}$. For a rectangular dissimilarity matrix, the VAF criterion becomes

$$\text{VAF} = 1 - \frac{\sum_r \sum_c (q_{rc} - s_{rc})^2}{\sum_r \sum_c (q_{rc} - \bar{q})^2} = 1 - \frac{\text{tr}(\mathbf{Q} - \mathbf{S})(\mathbf{Q} - \mathbf{S})'}{\text{tr}(\mathbf{Q} - \bar{\mathbf{Q}})(\mathbf{Q} - \bar{\mathbf{Q}})'},$$

where $\bar{\mathbf{Q}}$ represents a rectangular matrix with each cell containing $\bar{q} = \frac{1}{N_R N_C} \sum_r \sum_c q_{rc}$.

Spectral decomposition of a matrix into a sum of equally-sized matrices allows for approximating the total variability of the original data matrix by a small number of additive components (known as low-rank approximation). In an analogous manner, multiple discrete or continuous structures can be fit to a given proximity matrix to increase the overall VAF score vis-à-vis a solution

featuring only a single representational structure (see Carroll, 1976). Multiple representational structures are fit to the proximity data matrix through successive residualization. Let \mathbf{S}_k denote the k^{th} representational structure, with $k = 1, \dots, K$. The initial structure \mathbf{S}_1 is fit to \mathbf{P} or \mathbf{Q} . The residual matrix $\mathbf{P} - \mathbf{S}_1 = \mathbf{R}_1$ (or $\mathbf{Q} - \mathbf{S}_1$) is subsequently fit by a second structure, \mathbf{S}_2 , bearing residual matrix $\mathbf{R}_2 = \mathbf{R}_1 - \mathbf{S}_2$ that, in turn, is fit by a third structure, \mathbf{S}_3 . The resulting residual matrix $\mathbf{R}_3 = \mathbf{R}_2 - \mathbf{S}_3$ is then approximated by a fourth structure, \mathbf{S}_4 , and so on. In case of multiple representational structures, the least-squares loss functions are given by

$$L(\mathbf{S}_1, \dots, \mathbf{S}_K) = \sum_{i < j} (p_{ij} - \sum_k s_{(k)ij})^2 = \frac{1}{2} \text{tr}(\mathbf{P} - \sum_k \mathbf{S}_k)(\mathbf{P} - \sum_k \mathbf{S}_k)',$$

and

$$L(\mathbf{S}_1, \dots, \mathbf{S}_K) = \sum_r \sum_c (q_{rc} - \sum_k s_{(k)rc})^2 = \text{tr}(\mathbf{Q} - \sum_k \mathbf{S}_k)(\mathbf{Q} - \sum_k \mathbf{S}_k)'.$$

The VAF criteria become:

$$\text{VAF} = 1 - \frac{\sum_{i < j} (p_{ij} - \sum_k s_{(k)ij})^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} = 1 - \frac{\text{tr}(\mathbf{P} - \sum_k \mathbf{S}_k)(\mathbf{P} - \sum_k \mathbf{S}_k)'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}})(\mathbf{P} - \bar{\mathbf{P}})'},$$

and

$$\text{VAF} = 1 - \frac{\sum_r \sum_c (q_{rc} - \sum_k s_{(k)rc})^2}{\sum_r \sum_c (q_{rc} - \bar{q})^2} = 1 - \frac{\text{tr}(\mathbf{Q} - \sum_k \mathbf{S}_k)(\mathbf{Q} - \sum_k \mathbf{S}_k)'}{\text{tr}(\mathbf{Q} - \bar{\mathbf{Q}})(\mathbf{Q} - \bar{\mathbf{Q}})'}$$

2.1.3 Combinatorial Data Analysis and Structural Representations

Scaling, unfolding and (hierarchical) tree representations are familiar, well-established techniques in quantitative psychology and statistics. Traditional implementations rely on calculus-based or related numerical methods for minimizing the least-squares loss function. They work well with multidimensional scaling and tree fitting; in instances such as unfolding or unidimensional scaling, their performance is often disastrous. In fact, during the past several decades, quantitative psychologists have become increasingly aware of the combinatorial nature of these representation techniques (for reviews on combinatorial data analysis, see Arabie & Hubert, 1992, 1996). As the most prominent example, proximity-based unidimensional scaling is today widely recognized as a combinatorial optimization problem more appropriately addressed by discrete rather than continuous optimization techniques (see Brusco, 2002b; Defays, 1978; De Leeuw & Heiser, 1977; Hubert & Arabie, 1986, 1988; Hubert, Arabie, & Meulman, 2002, 2006).

Distinct from continuous, real-valued optimization problems, combinatorial optimization is characterized by non-smooth functions, because additional inte-

ger (particularly, binary) restrictions have been placed on at least some of the decision variables. Combinatorial optimization problems are discrete, the set of feasible solutions is finite, and an optimal solution always exists, implying the misrepresentation that these problems are “easy” and solvable by complete enumeration. The number of feasible solutions, however, grows exponentially with problem size, and nice optimality conditions are not in existence to verify, analytically, guaranteed optimality. Instead, discrete optimization problems require the explicit or implicit search of the entire solution space to locate the global optimum. Even for small-scale problems, the computational effort of an exhaustive enumeration of all feasible solutions is prohibitive. Because of this, most current algorithms for solving combinatorial optimization problems of substantial size are heuristic, with no guarantee of identifying a global optimum but often producing solutions at least within a close neighborhood of the desired global optimum. Partial enumeration strategies such as dynamic programming (see Hubert et al., 2001a) and branch-and-bound (see Brusco & Stahl, 2005a) can often provide guaranteed globally optimal solutions without the need for explicit enumeration of the entire feasible solution set but do face serious limitations on the sizes of problems that can be handled.

Constructing combinatorial continuous spatial or discrete non-spatial object representations rests on identifying optimal object permutations, where ‘optimal’ is operationalized within the context of a specific representation. Defays (1978), for instance, demonstrates that the task of finding a best-fitting unidimensional scale for given inter-object proximities can be solved solely by permuting the rows and columns of the data matrix such that a certain ordinal patterning among cell entries is satisfied, corresponding to an arrangement of objects along a dimension. The desired numerical scale values can immediately be inferred from the reordered matrix. Similarly, for hierarchical clustering problems, in their more refined guise as searches for ultrametric or additive tree representations, optimal solutions agree with a characteristic ordinal patterning of the input proximity matrix, implied by the constraints deducible from the additive or ultrametric inequality defining the respective tree structure.

2.1.4 Extensions of Combinatorial Structural Representations to Three-Way Proximity Matrices

The approach for modelling three-way data adopted here for all structural representations is guided by a principle common in statistics as well as of immediate intuitive appeal, namely, to analyze individual variability within a deviation-from-the-mean framework. Based on the aggregated body of individual proximity data, a combinatorially optimal structural representation is identified and then used in a subsequent step as a frame of reference for analyzing the individual proximity matrices. Recall that each discrete non-spatial as well as continuous spatial representation of the aggregated data matrix is characterized

by a particular permutation of its row and column objects. Their order serves as a blueprint for imposing the respective reference structure in a confirmatory manner on the individual proximity matrices: the individual source matrices are directly fit against the reference permutation by estimating the representational structure through a constrained least-squares algorithm. The computation of individual fit measures allows for an interindividual assessment quantifying the amount of agreement (or deviation) between the individual data representations and the chosen reference structure.

2.2 Computational Devices

As will be described in much greater detail in Chapter 4, constructing a combinatorial continuous spatial representation requires a two-step procedure. First, an optimal ordering of objects along the continuum is obtained by permuting the rows and columns of the input proximity matrix through a heuristic search termed quadratic assignment. The identified object order translates into linear inequality constraints to be satisfied by the actual numerical estimates of the interobject distances minimizing the least-squares loss function. For estimation, we use a constrained least-squares iterative projection algorithm proposed by Dykstra (1983). Identifying discrete non-spatial representations does not involve quadratic assignment, but solely relies on iterative projection for estimating the length of the paths connecting objects. The next two sections provide a technical description of quadratic assignment and iterative projection.

2.2.1 Quadratic Assignment

Matrix permutation analysis falls into a class of NP-hard combinatorial problems (Garey & Johnson, 1979), and guaranteed optimal solutions are difficult if not impossible to obtain in a reasonable amount of time. In fact, from a certain object set size onward, obtaining guaranteed optimal solutions is downright infeasible (as the number of distinct permutations to be evaluated equals $N!/2$, in not counting reverse object permutations). Although complete enumeration of all object orderings is not computationally feasible for matrix permutation problems where $N > 13$, partial enumeration techniques such as dynamic programming and branch-and-bound methods can often facilitate optimal solutions of larger problems. Hubert et al. (2001a, Chapter 4) advocate the implementation of dynamic programming to guarantee optimal solutions for combinatorial data analysis problems related to matrix permutation. Depending on available computer resources, especially random-access-memory (RAM) storage, dynamic programming is a reliable and efficient method for finding globally optimal permutations for a variety of important problems related to the seriation and scaling of proximity data, provided that the number of objects is 30 or fewer. Branch-and-bound methods (Brusco & Stahl, 2005a, Chapters 7-11) can sometimes

provide globally-optimal solutions for slightly larger matrices, but can also require more computation time than dynamic programming for some matrices. Despite significant advancement in the development of optimal solution procedures, heuristic algorithms remain necessary for matrix permutation problems of practical size. These algorithms typically fall into one of two categories: (1) iterative improvement strategies, and (2) metaheuristics such as tabu search (Glover and Laguna, 1993), genetic algorithms (Goldberg, 1989), simulated annealing (Aarts & Korst, 1989), and variable neighborhood search (Hansen & Mladenovic, 2003, 2005) designed to facilitate escape from local optima. We focus on iterative improvement strategies. They require the generation of an initial permutation, which is usually constructed via random assignment of objects to sequence positions. Neighborhood search operations, which include pairwise interchange of objects, relocation of object blocks, and object-block reversals, are subsequently applied to refine the initial permutation, resulting in iterative improvement of the objective function index corresponding to the matrix permutation problem. The iterative improvement algorithm terminates when no more neighborhood moves will further improve the objective function. The resulting solution is, therefore, locally optimal with respect to the class of neighborhood search operations used in the iterative improvement algorithm.

Quadratic assignment (QA) represents an especially efficient iterative improvement strategy for matrix permutation problems (see Hubert et al. 1997, 2006). In general, solving the QA problem for two given $N \times N$ matrices $\mathbf{P} = \{p_{ij}\}$ and $\mathbf{B} = \{b_{ij}\}$, requires finding a permutation of the rows and columns of \mathbf{P} for \mathbf{B} fixed, such that the Hadamard product, the sum of the products of corresponding matrix cells is maximized:

$$\max_{\rho} \Gamma(\rho) = \max_{\rho} \left\{ \sum_{i,j} (p_{\rho(i)\rho(j)} b_{ij}) \right\},$$

with $\rho(\cdot) \equiv \rho$ denoting a feasible permutation of the first N integers (see Hubert et al. 1997, 2006; Rendl, 2002). Let \mathbf{P} denote any input proximity matrix to be permuted into a distinctive form most closely approximating the desired ordinal patterning associated with a specific structural representation. Matrix \mathbf{B} serves as target and embodies the intended characteristic patterning. As the algorithm iteratively tries to maximize $\Gamma(\rho)$ by re-ordering \mathbf{P} against \mathbf{B} , those corresponding cells of the two matrices will be matched that maximally contribute to an increase of the QA criterion. The resulting permutation will render \mathbf{P} in a form as close as possible to the target patterning of \mathbf{B} , denoted \mathbf{P}_{ρ} . From the identified permutation, ρ , linear inequality constraints are deduced on the numerical estimates of the intended structural representation, \mathbf{S} , that is fit to \mathbf{P} through iterative projection (described in detail in the next section). Subsequently, target matrix \mathbf{B} can be updated by \mathbf{S} obtained, initializing a second search cycle for a permutation possibly superior to the previously identified. The process iterates until convergence, when updating \mathbf{B} does not result in any further increase of

$\Gamma(\rho)$. Clearly, the solution depends on the initial random arrangement of rows and columns of \mathbf{P} , for which it is passed to the first QA search cycle. Therefore, to reduce the risk of detecting a purely locally optimal solution, a common nuisance to any heuristic procedure, utilization of the algorithm with multiple starts on the randomly permuted input matrix \mathbf{P} is strongly recommended.

2.2.2 Iterative Projection

The numerical estimates of structural representations, \mathbf{S} , must conform to certain constraints, either deducible from the specific permutation of the row and column objects of \mathbf{P} as found through QA in case of continuous geometric representations, or directly imposed by the defining properties of discrete structures. The least-squares approximation \mathbf{S} to \mathbf{P} , is estimated by iterative projection (IP), a computational strategy for solving constrained least-squares minimization problems proposed by Dykstra (1983; see also Deutsch, 2001). Let \mathbf{p} and \mathbf{s} indicate vectorizations of matrices \mathbf{P} and \mathbf{S} ; IP minimizes $(\mathbf{p} - \mathbf{s})'(\mathbf{p} - \mathbf{s})$ subject to \mathbf{s} conforming to characteristic linear inequality constraints.

In general, let $X \in \mathbb{R}^n$ (with $n \in \mathbb{N}_+$) denote a complete inner product space, also called a Hilbert space. The least-squares approximation to an arbitrary vector $\mathbf{x} \in X$ is sought from a closed convex set $\mathcal{C} \in X$ in the form of a vector $\hat{\mathbf{x}}^* \in \mathcal{C}$ minimizing the least-squares criterion, $(\mathbf{x} - \hat{\mathbf{x}})'(\mathbf{x} - \hat{\mathbf{x}})$, subject to linear constraints imposed on $\hat{\mathbf{x}}$ as defined by \mathcal{C} . The characterization of \mathcal{C} as a convex set generally implies that for arbitrary vectors $\mathbf{x}, \mathbf{x}^o \in \mathcal{C}$ the line segment joining the two points defined by them is also contained in \mathcal{C} ; algebraically $\alpha\mathbf{x} + (1 - \alpha)\mathbf{x}^o \in \mathcal{C}$, with $0 \leq \alpha \leq 1$; a set is closed if for any convergent sequence $\mathbf{x}^{(t)}$, $\lim_t \mathbf{x}^{(t)}$ also belongs to the set.

Conceptually, $\hat{\mathbf{x}}^*$ can be found directly by projecting \mathbf{x} onto \mathcal{C} ; in practice, however, this may pose extreme computational demands. Dykstra's solution to this problem is as simple as it is elegant: based on the decomposition of the constraint set $\mathcal{C} = \bigcap_{w=1}^W \mathcal{C}_w \neq \emptyset$, the (presumably) difficult calculation of $\hat{\mathbf{x}}^* \in \mathcal{C}$ is broken down into the easier task of constructing a sequence $\mathbf{x}^{(t)}$, with $t = 0, 1, 2, \dots$, by way of iterative projections of \mathbf{x} onto the W closed convex subsets of restrictions, $\mathcal{C}_1, \dots, \mathcal{C}_W$, which, as was proven by Boyle and Dykstra (1985), converges to $\hat{\mathbf{x}}^* \in \mathcal{C}$:

$$\lim_t \|\mathbf{x}^{(t)} - P_{\mathcal{C}}(\mathbf{x})\| = 0,$$

with $P_{\mathcal{C}}(\mathbf{x})$ indicating the projection of \mathbf{x} onto \mathcal{C} . Note that $\mathcal{C}_1, \dots, \mathcal{C}_W$ need not necessarily be subspaces of X ; they represent linear inequality constraints of the general form

$$\mathcal{C}_w = \{\hat{\mathbf{x}} \in \mathcal{C}_w \mid \mathbf{a}'_w \hat{\mathbf{x}} \leq b_w \Leftrightarrow \mathbf{a}'_w \hat{\mathbf{x}} - b_w \leq 0\}.$$

The sequence $\mathbf{x}^{(t)}$ is initialized by setting $\mathbf{x}^{(0)} := \mathbf{x}$, followed by the projec-

tion of $\mathbf{x}^{(0)}$ onto \mathcal{C}_1 , resulting in $\mathbf{x}^{(1)}$ that in turn is projected onto \mathcal{C}_2 , yielding $\mathbf{x}^{(2)}$ to be projected onto \mathcal{C}_3 , and so on. The difference between consecutive projections $\mathbf{x}^{(t-1)}$ and $\mathbf{x}^{(t)}$ is often called the increment, or residual. The algorithm concludes its first cycle of projections onto sets $\mathcal{C}_1, \dots, \mathcal{C}_W$, with the projection of $\mathbf{x}^{(W-1)}$ onto \mathcal{C}_W producing $\mathbf{x}^{(W)}$, the input vector for the second projection cycle through $\mathcal{C}_1, \dots, \mathcal{C}_W$. However, from the second cycle on, each time $\mathcal{C}_1, \dots, \mathcal{C}_W$ are revisited in subsequent cycles, the increment from the previous cycle associated with that particular set has to be subtracted from the vector before actually proceeding with the projection. Introducing notation $[t]$, sometimes referred to as “ $t \bmod W$ ”, allows for tying together the number of subsequent projections and the index of constraint sets involved (see Deutsch, 2001; Dykstra, 1983):

$$[t] := \{1, 2, \dots, W\} \cap \{t - gW \mid g = 0, 1, 2, \dots\};$$

for example, $[1] = 1$, $[2] = 2$, $[3] = 3$, ..., $[W] = W$, then $[W + 1] = 1$, $[W + 2] = 2$, ..., $[2W] = W$, and so on. Han (1988) demonstrates that the projection of $P_{\mathcal{C}_{[t]}}(\mathbf{x}) = \hat{\mathbf{x}}^{(t)}$ for locally minimizing $(\mathbf{x} - \hat{\mathbf{x}}^{(t)})'(\mathbf{x} - \hat{\mathbf{x}}^{(t)})$, with $\mathbf{x} \in \mathbb{R}^n$ fixed, has the closed form solution

$$\hat{\mathbf{x}}^{(t)} = \mathbf{x} - (\mathbf{a}'_w \mathbf{a}_w)^{-1} (\max\{\mathbf{a}'_w \mathbf{x} - b_w, 0\}) \mathbf{a}_w.$$

If $\mathcal{C}_1, \dots, \mathcal{C}_W$ are closed subspaces of X , or, more generally, if the $\mathcal{C}_1, \dots, \mathcal{C}_W$ are closed affine sets (i.e., translations of closed subspaces), then the computation of the residuals can be omitted, and each $\mathbf{x}^{(t)}$ can be directly computed as the projection of its predecessor onto $\mathcal{C}_{[t]}$ (i.e., $\mathbf{x}^{(t)} = P_{\mathcal{C}_{[t]}}(\mathbf{x}^{(t-1)})$), because

$$\lim_{t \rightarrow \infty} (P_{\mathcal{C}_W} P_{\mathcal{C}_{W-1}} \dots P_{\mathcal{C}_1})^{(t)}(\mathbf{x}) = P_{\mathcal{C}}(\mathbf{x})$$

(for proofs see Deutsch, 2001; Dykstra, 1983). Hubert and Arabie (1995b) refer to this special case as IP without augmentation, whereas the general case, with each projection entailing the preceding removal of the increments from the past projection onto the same set, is called IP with augmentation.

In conclusion, we present a detailed account of the intricate computation of the increments based on a recursion formula, including a small-scale example. After initializing $\mathbf{x}^{(0)} := \mathbf{x}$, and associated residuals as $\mathbf{e}^{(0)} = \mathbf{0}$, the recursion

$$\mathbf{x}^{(t)} = P_{\mathcal{C}_{[t]}}(\mathbf{x}^{(t-1)} + \mathbf{e}^{(t-W)}),$$

allows for construction of the sequence $\mathbf{x}^{(t)}$ as consecutive projections. The corresponding increments, $\mathbf{e}^{(t)}$, when moving from $\mathbf{x}^{(t-1)}$ to $\mathbf{x}^{(t)}$ are also defined recursively, however, not as plain differences between $\mathbf{x}^{(t-1)}$ and $\mathbf{x}^{(t)}$, but rather as corrected residuals, already incorporating the necessary removal of the increment associated with a particular \mathcal{C}_w from the preceding cycle, namely as the

difference between the corrected projection from the previous step and the one onto \mathcal{C}_{w+1} to follow:

$$\mathbf{e}^{(t)} := \mathbf{x}^{(t-1)} + \mathbf{e}^{(t-W)} - \mathbf{x}^{(t)} = \mathbf{x}^{(t-1)} + \mathbf{e}^{(t-W)} - P_{\mathcal{C}_{[t]}}(\mathbf{x}^{(t-1)} + \mathbf{e}^{(t-W)}). \quad (2.1)$$

As an illustration of the underlying mechanics, we will consider the simplest case with $W = 2$. The resulting sequences $\mathbf{x}^{(t)}$ and $\mathbf{e}^{(t)}$ are given in Table 2.1; note that the residuals, as defined in (2.1), are strictly negative. To summarize, determining $\mathbf{x}^{(t)}$ first requires to add $\mathbf{e}^{(t-W)}$ to $\mathbf{x}^{(t-1)}$ and then project the sum $\mathbf{x}^{(t-1)} + \mathbf{e}^{(t-W)}$ onto $\mathcal{C}_{[t]}$.

Table 2.1: IP Sequences of $\mathbf{x}^{(t)}$ and $\mathbf{e}^{(t)}$ for $W = 2$ constraint sets, \mathcal{C}_1 and \mathcal{C}_2 .

t	$\mathbf{x}^{(t)}$	$\mathbf{e}^{(t)}$
0	$\mathbf{x}^{(0)} := \mathbf{x}$	$\mathbf{e}^{(0)} = \mathbf{0}$
1^{st} cycle through constraint sets \mathcal{C}_1 and \mathcal{C}_2		
1	$\mathbf{x}^{(1)} = P_{\mathcal{C}_{[1]}}(\mathbf{x}^{(0)} + \mathbf{e}^{(-1)})$	$\mathbf{e}^{(1)} = \mathbf{x}^{(0)} + \mathbf{e}^{(-1)} - P_{\mathcal{C}_{[1]}}(\mathbf{x}^{(0)})$
	$= P_{\mathcal{C}_1}(\mathbf{x}^{(0)})$	$= \mathbf{x}^{(0)} - \mathbf{x}^{(1)}$
2	$\mathbf{x}^{(2)} = P_{\mathcal{C}_{[2]}}(\mathbf{x}^{(1)} + \mathbf{e}^{(0)})$	$\mathbf{e}^{(2)} = \mathbf{x}^{(1)} + \mathbf{e}^{(0)} - P_{\mathcal{C}_{[2]}}(\mathbf{x}^{(1)} + \mathbf{e}^{(0)})$
	$= P_{\mathcal{C}_2}(\mathbf{x}^{(1)})$	$= \mathbf{x}^{(1)} - \mathbf{x}^{(2)}$
2^{nd} cycle through constraint sets \mathcal{C}_1 and \mathcal{C}_2		
3	$\mathbf{x}^{(3)} = P_{\mathcal{C}_{[3]}}(\mathbf{x}^{(2)} + \mathbf{e}^{(1)})$	$\mathbf{e}^{(3)} = \mathbf{x}^{(2)} + \mathbf{e}^{(1)} - P_{\mathcal{C}_{[3]}}(\mathbf{x}^{(2)} + \mathbf{e}^{(1)})$
	$= P_{\mathcal{C}_1}(\mathbf{x}^{(2)} + \mathbf{e}^{(1)})$	$= \mathbf{x}^{(2)} + \mathbf{e}^{(1)} - \mathbf{x}^{(3)}$
4	$\mathbf{x}^{(4)} = P_{\mathcal{C}_{[4]}}(\mathbf{x}^{(3)} + \mathbf{e}^{(2)})$	$\mathbf{e}^{(4)} = \mathbf{x}^{(3)} + \mathbf{e}^{(2)} - P_{\mathcal{C}_{[4]}}(\mathbf{x}^{(3)} + \mathbf{e}^{(2)})$
	$= P_{\mathcal{C}_2}(\mathbf{x}^{(3)} + \mathbf{e}^{(2)})$	$= \mathbf{x}^{(3)} + \mathbf{e}^{(2)} - \mathbf{x}^{(4)}$
3^{rd} cycle through constraint sets \mathcal{C}_1 and \mathcal{C}_2		
5	$\mathbf{x}^{(5)} = P_{\mathcal{C}_{[5]}}(\mathbf{x}^{(4)} + \mathbf{e}^{(3)})$	$\mathbf{e}^{(5)} = \mathbf{x}^{(4)} + \mathbf{e}^{(3)} - P_{\mathcal{C}_{[5]}}(\mathbf{x}^{(4)} + \mathbf{e}^{(3)})$
	$= P_{\mathcal{C}_1}(\mathbf{x}^{(4)} + \mathbf{e}^{(3)})$	$= \mathbf{x}^{(4)} + \mathbf{e}^{(3)} - \mathbf{x}^{(5)}$
6	$\mathbf{x}^{(6)} = P_{\mathcal{C}_{[6]}}(\mathbf{x}^{(5)} + \mathbf{e}^{(4)})$	$\mathbf{e}^{(6)} = \mathbf{x}^{(5)} + \mathbf{e}^{(4)} - P_{\mathcal{C}_{[6]}}(\mathbf{x}^{(5)} + \mathbf{e}^{(4)})$
	$= P_{\mathcal{C}_2}(\mathbf{x}^{(5)} + \mathbf{e}^{(4)})$	$= \mathbf{x}^{(5)} + \mathbf{e}^{(4)} - \mathbf{x}^{(6)}$
\vdots		

3 Additive Tree Representations

Additive and ultrametric trees represent two of the more prominent graph theoretic models for discrete nonspatial representations of two-way one-mode proximity data. Typically, a single best-fitting tree structure for a given proximity matrix is sought such that the discrepancies between the observed interobject proximities and corresponding estimates of tree distances are minimized. However, in some instances a data representation through multiple tree structures might be preferred to attain a higher overall fit.

Extensions to three-way two-mode data, as might arise in the context of cross-sectional or longitudinal studies, to model individual differences, similar to the INDSCAL generalization of multidimensional scaling by Carroll and Chang (1970), have been suggested for locating additive as well as ultrametric tree structures. The INDTREES algorithm for identifying individualized tree structures (Carroll, Clark, & DeSarbo, 1984; De Soete & Carroll, 1989), estimates differential interobject tree distances through conjugate gradient minimization of a least-squares loss function, augmented by penalty terms to account for violations of the constraints as imposed by the underlying tree model (i.e., the ultrametric inequality in case of ultrametric trees, and the four-point condition for additive trees). Unfortunately, INDTREES is not available as a computer program.

We propose an alternative method for finding and fitting additive tree structures to three-way two-mode proximity data that in estimating the tree distances neither relies on gradient-based optimization nor penalty functions. Instead, the method presented minimizes the least-squares loss criterion by using IP of the vector of observed proximities onto closed convex sets representing the constraints defined by the four-point condition specifying the additive tree model (see Hubert & Arabie, 1995b; Hubert et al., 2006).

The next section summarizes the essential definitions and concepts related to additive tree representations. We proceed with a review of the penalty function algorithm in the third section; the fourth presents IP as a tool for heuristic and non-heuristic identification of best-fitting additive tree structures. Constructing additive tree representations to accommodate three-way two-mode data is introduced in section five. An application of IP-based additive tree representations for analyzing individual differences in judgments of schematic face stimuli illustrates the method.

3.1 Additive Tree Structures

3.1.1 Definitions and Concepts

A brief review of a few essential graph-theoretic definitions might be helpful in clarifying the concept of an additive tree structure:

1. A graph G is defined as a pair $G = (V, E)$, where V denotes a finite set of nodes or vertices, $V = \{v_1, v_2, v_3, \dots\}$. Set E has as its elements subsets of V of cardinality two called edges, $E = \{e_1, e_2, e_3, \dots\} = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}, \dots\}$.
2. In a graph $G = (V, E)$ with $e_1 = \{v_1, v_2\} \in E$, v_1 is said to be adjacent to v_2 (and vice-versa), and e is incident upon v_1 (and v_2).
3. The degree of a vertex v of G corresponds to the number of edges incident upon v .
4. The sequence s of vertices $s = (v_1, v_2, v_3, \dots, v_k)$, with $k \geq 1$, such that $\{v_j, v_{j+1}\} \in E$, for $j = 1, \dots, k-1$, defines a walk in G . If s does not contain any repeated nodes it is called a path.
5. A path with coinciding endpoints, $v_k = v_1$, for $k > 1$, is called a circuit or cycle.
6. A graph is called connected if there is at least one path between any pair of distinct vertices.
7. A weighted graph $G = (V, E)$ involves a function $f : E \mapsto \mathbb{R}^+$ with $w_e = f(e) = f(\{v_j, v_{j'}\})$, with $e \in E$ and $v_j, v_{j'} \in V$, $j \neq j'$ that assigns a weight to edge e . Often weights are interpreted in terms of distances between vertices v_j and $v_{j'}$.
8. A tree is a connected graph G with no cycles; vertices with degree 1 are called leaves or terminal nodes, and the other (interior) vertices are termed nodes. The distance between leaves and subsequent internal nodes is called node height; paths connecting leaves with the first layer of internal nodes sometimes are referred to as branches.
9. A rooted tree consists of a tree G and a node r of G , its root, denoted by (G, r) .

An additive tree can be characterized as a weighted acyclic connected graph without a natural root (i.e., an additive tree is not necessarily organized into a vertical hierarchy), and thus, can be rooted at any node, or point on any path connecting two objects. The terminal nodes of an additive tree represent a set of N objects $\mathcal{O} = \{O_1, \dots, O_N\} \subset V$. The weights along the paths connecting objects O_i, O_j , with $1 \leq i, j \leq N$, typically with a distance interpretation, can be collected into an $N \times N$ matrix $\Delta = \{\delta_{ij}\}$. As a necessary and sufficient

condition for a unique additive tree representation of Δ , the δ_{ij} have to satisfy the additive inequality or four-point condition (see Barthélemy & Guénoche, 1991; De Soete & Carroll, 1996):

$$\delta_{ij} + \delta_{kl} \leq \max \{ \delta_{ik} + \delta_{jl}, \delta_{jk} + \delta_{il} \} \quad \text{for } 1 \leq i, j, k, l \leq N; \quad (3.1)$$

or equivalently, for any object quadruple O_i, O_j, O_k , and O_l the two largest sums of path length distances among $\delta_{ij} + \delta_{kl}$, $\delta_{ik} + \delta_{jl}$, and $\delta_{jk} + \delta_{il}$ must be equal. The additive tree metric, as defined by the four-point condition, represents a generalization of an ultrametric on Δ , defined by the ultrametric inequality (see Barthélemy & Guénoche, 1991; De Soete & Carroll, 1996):

$$\delta_{ij} \leq \max \{ \delta_{ik}, \delta_{jk} \} \quad \text{for } 1 \leq i, j, k \leq N;$$

in words, for any object triple O_i, O_j , and O_k , the largest two path length distances among δ_{ij} , δ_{ik} , and δ_{jk} must be equal (for greater clarity, we denote distances δ_{ij} conforming to the ultrametric condition as u_{ij}). Unlike an additive tree, an ultrametric tree has a unique root, equidistant to all leaves or terminal nodes.

Carroll (1976) discusses the decomposition of an additive tree distance into the sum of an ultrametric and a centroid distance. The latter induces a star as its tree representation, where one of the vertices, the center v_c , is fixed. Distances for all $v_j, v_{j'}$, with $j, j' \neq c$, are obtained by passing through v_c . Formally, centroid distances have the following properties:

1. $\delta_{jj} = 0 \quad \forall v_j \in V, j \neq c \text{ and } \delta_{jc} > 0$
2. $\delta_{jj'} = \delta_{jc} + \delta_{cj'} \quad \forall v_j \neq v_{j'} \in V$
3. $\delta_{cj} = \delta_{jc} \quad \forall v_j \in V$

(subsequently, we use c_{ij} as notation for the centroid distance between objects O_i, O_j). Any additive tree matrix can be represented as the sum of an ultrametric and a centroid matrix. A decomposition is not unique due to the location indeterminacy of the root of an additive tree, leaving an infinite number of choices for rooting the tree, and as an immediate consequence, allowing for infinitely many decompositions.

Numerous conventions exist for determining the root of an additive tree. For example, the procedure proposed by Barthélemy and Guénoche (1991, Ch. 3.3.3) makes explicit use of the ultrametric-centroid decomposition as a means to determine the root of an additive tree. Let O_{i^*} and O_{j^*} denote the two objects in \mathcal{O} farthest apart, defining the largest path in the additive tree: $\delta_{i^*j^*} = \max\{\delta_{ij}\}$. Consider a point x on the path $\delta_{i^*j^*}$ with $x \notin \mathcal{O}$; calculate $\delta_{i^*x} = \delta_{j^*x} = \delta_{i^*j^*}/2$. For any $O_i, O_j \in \mathcal{O}$, with $i, j \neq i^*, j^*$, compute $c(i) = \max\{\delta_{ii^*}, \delta_{ij^*}\} - \delta_{i^*j^*}/2$, and $c(j) = \max\{\delta_{ji^*}, \delta_{jj^*}\} - \delta_{i^*j^*}/2$ — note that $c(i) = \delta_{ic} = c_{ic}$ and $c(j) = \delta_{jc} = c_{jc}$ such that the centroid distance between O_i and O_j , c_{ij} , corresponds

to the sum of $c(i)$ and $c(j)$, $c_{ij} = c(i) + c(j)$. Let the ultrametric distance between O_i and O_j , u_{ij} , be given by $u_{ij} = \delta_{ij} - c(i) - c(j)$. In case of possible negative values of u_{ij} , a constant k needs to be determined such that $u_{ij} \geq 0$, $\forall i, j$, to recompute $c(i), c(j)$ as $c(i) := c(i) - k/2$. The process is continued until all $u_{ij} \geq 0$. The ultrametric path length distances, u_{ij} , between leaves $O_i, O_j \in \mathcal{O}$ are then lengthened or shortened by attaching the centroid components, $c(i)$ and $c(j)$, resulting in the final rooted additive tree representation of Δ . Lastly, it should be noted that due to the indeterminacy of the ultrametric-centroid decomposition no closed-form computational procedure is available for determining k ; rather, k has to be identified through trial and error.

The approach we will adopt henceforth for rooting an additive tree was suggested by Sattath and Tversky (1977) and simply identifies the root as the particular node, not necessarily in V , that minimizes the variance of the distances from the root to the leaves. This operationalization seems particularly appealing due to its emphasis on additive trees representing generalizations of ultrametric trees as reflected by the minimum variance criterion attaining zero in that special instance. The Sattath-Tversky procedure tends to produce balanced tree diagrams as it attempts to impose a bifurcated branching pattern such that at each level of the hierarchy, whenever possible vis-à-vis the actual distances, at most two previously disjoint branches are merged.

3.1.2 Additive Tree Representations as Constrained Least-Squares Minimization Problems

Observed data typically never satisfy the four-point condition. Thus, the task of finding an additive tree representation of a given data, often in the format of a square symmetric proximity matrix, $\mathbf{P} = \{p_{ij}\}$, observed on \mathcal{O} , requires the estimation of path length distances $\hat{\delta}_{ij} =: d_{ij}$, optimally approximating the proximities, at the same time satisfying the constraints as defined by the four-point condition in (3.1). Usually, fit is quantified by the least-squares criterion. Locating an additive tree structure to \mathbf{P} can be formalized as a constrained optimization problem (see Barthélemy & Guénoche, 1991; De Soete & Carroll, 1996):

$$\min_{\mathbf{D}} \{L(\mathbf{D})\} = \min_{\mathbf{D}} \left\{ \sum_{i < j} (p_{ij} - d_{ij})^2 \right\} = \min_{\mathbf{D}} \left\{ \frac{1}{2} \text{tr}(\mathbf{P} - \mathbf{D})(\mathbf{P} - \mathbf{D})' \right\}, \quad (3.2)$$

subject to

$$d_{ij} + d_{kl} \leq \max \{d_{ik} + d_{jl}, d_{jk} + d_{il}\} \quad \text{for } 1 \leq i, j, k, l \leq N,$$

with $\mathbf{D} = \{d_{ij}\}$ denoting the matrix of estimated path length distances.

Conceptually, locating an additive tree for a given proximity matrix separates into the distinct tasks of construction and fitting. The former seeks

to identify a particular additive tree structure most appropriate for the given data through establishing constraints as defined by (3.1). Fitting is concerned with estimating tree distances, conforming to the constraints of the particular additive tree structure identified such that $L(\mathbf{D})$ is minimized. As Křivánek (1986) demonstrates, locating an additive tree poses an NP-hard optimization problem, rendering all currently available algorithms for finding additive tree structures as heuristics, with no guarantee of obtaining a globally optimal solution (of course, with the possible exception of a reasonably sized problem that allows for complete enumeration). If the constraints characterizing the desired tree structure are known in advance, either from research or suggested by theory (i.e., the construction step can be skipped), (3.2) reduces to solving a least-squares problem with given linear inequality constraints, typically termed non-heuristic tree-fitting.

3.1.3 Multiple Additive Tree Structures

Carroll and Pruzansky (1980; see also, Carroll, 1976; De Soete & Carroll, 1996) propose locating multiple additive tree structures to a given data matrix by means of successive residualizations of the input proximity matrix. As an example, consider locating a biadditive tree representation of \mathbf{P} : an initial tree structure $\mathbf{D}_1 = \{d_{(1)ij}\}$ is identified for \mathbf{P} , and a second structure $\mathbf{D}_2 = \{d_{(2)ij}\}$ located for the residual matrix $\mathbf{P} - \mathbf{D}_1$. In an attempt to further improve the fit of the biadditive structure, the residuals $\mathbf{P} - \mathbf{D}_1 - \mathbf{D}_2$ are added back to \mathbf{D}_1 , followed by re-estimating \mathbf{D}_1 , potentially better fitting $\mathbf{P} - \mathbf{D}_2$, and producing a revised residual matrix $(\mathbf{P} - \mathbf{D}_2) - \mathbf{D}_1$ to be restituted to \mathbf{D}_2 , followed by refitting \mathbf{D}_2 . The process continues by repetitively fitting the residuals from the second additive tree structure by the first, and the residuals from the first additive tree structure by the second, until the sequence converges (i.e., no further improvement in fit can be attained). Note that possibly negative residuals can be amended by an additive constant of sufficient size because additive (and also ultrametric) path length distances are translation invariant.

3.2 Locating Additive Trees Through the Penalty Function Algorithm

The mathematical programming algorithm for finding additive tree structures, devised by Carroll and Pruzansky (1980), employs the decomposition of an additive tree into an ultrametric and a star tree as a vehicle for identifying an additive tree structure by cycling interchangeably through fitting either of the two structures while the other is kept fixed, until the process converges. Conformity of the distance estimates to the ultrametric inequality is enforced through inclusion of a penalty term in the loss function. De Soete (1983) suggests a refinement of the penalty algorithm that operates directly on the observed proxim-

ities, without relying on an intermediate decomposition step. De Soete's penalty algorithm utilizes a relaxation of the constrained minimization problem in (3.2), obtained by translating the restrictions imposed by the four-point condition into a penalty function and shifting the latter into the objective function, yielding the unconstrained minimization problem:

$$\min_{\mathbf{D}} \{F(\mathbf{D}, \lambda)\} = \min_{\mathbf{D}} \{L(\mathbf{D}) + \lambda P(\mathbf{D})\},$$

with the penalty function reflecting violations of the four-point condition

$$P(\mathbf{D}) = \sum_{\mathcal{Q}} (d_{ik} + d_{jl} - d_{il} - d_{jk})^2,$$

where the set

$$\mathcal{Q} = \left\{ (i, j, k, l) \mid i \neq j \neq k \neq l, \ d_{ij} + d_{kl} \leq \min \{d_{ik} + d_{jl}, d_{jk} + d_{il}\}, \ d_{ik} + d_{jl} \neq d_{jk} + d_{il} \right\}$$

contains the ordered quadruples of objects for which the distance estimates infringe the additive inequality. The tuning parameter λ is initialized as the ratio of $L(\mathbf{D})/P(\mathbf{D})$, and updated from iteration $t - 1$ to t by $\lambda^{(t)} = 10\lambda^{(t-1)}$, thereby steadily increasing the impact of the penalty term for each iteration. Computationally, the minimization of $F(\mathbf{D}, \lambda)$ is achieved through an iterative conjugate gradient procedure suggested by Powell (1977).

3.3 Locating Additive Trees Through Iterative Projection

The IP algorithm for constructing additive tree structures by Hubert and Arabie (1995b), an adaptation of Dykstra's (1983) general IP algorithm for solving least-squares minimization problems with inequality constraints, allows for solving (3.2) directly, without deployment of intermediate decompositions, penalty functions or gradient-based optimization techniques.

Recall that an additive tree structure is perfectly determined by the collection of constraints, $\mathcal{C}_1, \dots, \mathcal{C}_W$ as defined by the four-point condition in (3.1), where each \mathcal{C}_w is associated with one of the $W = \binom{N}{4}$ quadruples to be generated, given N objects. For a specific quadruple w of objects O_i, O_j, O_k , and O_l , the four point condition translates into three possible inequality constraints, one of which must be satisfied by the six distances involved, representing the associated constraint set \mathcal{C}_w :

$$\begin{aligned} \delta_{ij} + \delta_{kl} &\leq \delta_{ik} + \delta_{jl} = \delta_{jk} + \delta_{il} \\ \delta_{ik} + \delta_{jl} &\leq \delta_{ij} + \delta_{kl} = \delta_{il} + \delta_{jk} \\ \delta_{il} + \delta_{jk} &\leq \delta_{ij} + \delta_{kl} = \delta_{ik} + \delta_{jl}. \end{aligned} \tag{3.3}$$

A given \mathcal{C}_w expands into four inequalities, for example \mathcal{C}_w defined by (3.3) entails:

$$\left. \begin{aligned} \delta_{ij} + \delta_{kl} &\leq \delta_{ik} + \delta_{jl} &\Leftrightarrow \delta_{ij} + \delta_{kl} - (\delta_{ik} + \delta_{jl}) &\leq 0 \\ \delta_{ij} + \delta_{kl} &\leq \delta_{jk} + \delta_{il} &\Leftrightarrow \delta_{ij} + \delta_{kl} - (\delta_{jk} + \delta_{il}) &\leq 0 \\ \delta_{ik} + \delta_{jl} &\leq \delta_{jk} + \delta_{il} \\ \delta_{jk} + \delta_{il} &\leq \delta_{ik} + \delta_{jl} \end{aligned} \right\} &\Leftrightarrow \delta_{ik} + \delta_{jl} - (\delta_{jk} + \delta_{il}) = 0.$$

3.3.1 Non-Heuristic Iterative Projection

In case the constraints $\mathcal{C}_1, \dots, \mathcal{C}_W$ on the $\hat{\delta}_{ij} = d_{ij}$ have already been identified (either through previous research or on theoretical grounds), the additive tree structure is estimated by what has been termed non-heuristic IP (with augmentation, as the \mathcal{C}_w represent convex cones). The practical purpose of fitting a known additive tree structure to a given proximity matrix $\mathbf{P} \equiv \mathbf{p}$ requires the estimation of tree distances $\mathbf{D}^* \equiv \mathbf{d}^* \in \mathcal{C} = \bigcap_1^W \mathcal{C}_w$ that minimize the least-squares loss function $(\mathbf{p} - \mathbf{d})'(\mathbf{p} - \mathbf{d})$, with \mathbf{p} and \mathbf{d} denoting vectorizations of matrices \mathbf{P} and \mathbf{D} . After initializing $\mathbf{d}^{(0)} = \mathbf{p}$, the algorithm proceeds by checking for each quadruple of objects O_i, O_j, O_k , and O_l whether the involved distances

	O_i	O_j	O_k	O_l
O_i		d_{ij}	d_{ik}	d_{il}
O_j			d_{jk}	d_{jl}
O_k				d_{kl}
O_l				

conform to the respective constraints in \mathcal{C}_w . If at iteration $t - 1$ a violation is encountered, the vector of distances $\mathbf{d}^{(t-1)}$ is projected onto \mathcal{C}_w , and the particular distances are replaced by their projections yielding $\mathbf{d}^{(t)}$ (see Dykstra, 1983; Han, 1988). For example, if for the first inequality of \mathcal{C}_w , as defined by (3.3), we observe $d_{ij}^{(t-1)} + d_{kl}^{(t-1)} - (d_{ik}^{(t-1)} + d_{jl}^{(t-1)}) > 0$, the projections are given by:

$$\begin{aligned} d_{ij}^{(t)} &= d_{ij}^{(t-1)} - \frac{1}{4} \left(d_{ij}^{(t-1)} + d_{kl}^{(t-1)} - d_{ik}^{(t-1)} - d_{jl}^{(t-1)} \right) \\ d_{kl}^{(t)} &= d_{kl}^{(t-1)} - \frac{1}{4} \left(d_{ij}^{(t-1)} + d_{kl}^{(t-1)} - d_{ik}^{(t-1)} - d_{jl}^{(t-1)} \right) \\ d_{ik}^{(t)} &= d_{ik}^{(t-1)} + \frac{1}{4} \left(d_{ij}^{(t-1)} + d_{kl}^{(t-1)} - d_{ik}^{(t-1)} - d_{jl}^{(t-1)} \right) \\ d_{jl}^{(t)} &= d_{jl}^{(t-1)} + \frac{1}{4} \left(d_{ij}^{(t-1)} + d_{kl}^{(t-1)} - d_{ik}^{(t-1)} - d_{jl}^{(t-1)} \right). \end{aligned}$$

Proceeding in this manner, IP cycles through $\mathcal{C}_1, \dots, \mathcal{C}_W$ until convergence to \mathbf{d}^* . As an aside, to ensure non-negativity of the estimated distances d_{ij} , for each triplet of objects O_i, O_j , and O_k the triangle inequality has to hold:

$$d_{ik} \leq d_{ij} + d_{jk},$$

which can either be effected by the explicit incorporation as additional constraints, or by the translation of the final distance estimates by an additive constant of sufficient size.

The VAF criterion serves as a measure for assessing the quality of an estimated additive tree structure, defined by

$$\text{VAF} = 1 - \frac{\sum_{i < j} (p_{ij} - d_{ij}^*)^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} = 1 - \frac{\text{tr}(\mathbf{P} - \mathbf{D}^*)(\mathbf{P} - \mathbf{D}^*)'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}})(\mathbf{P} - \bar{\mathbf{P}})'},$$

with \bar{p} denoting the mean of the off-diagonal entries in $\mathbf{P} = \{p_{ij}\}$, and d_{ij}^* the fitted distances. It should be noted that the solution vector \mathbf{d}^* minimizing the least-squares criterion also maximizes the squared correlation between \mathbf{p} and a vector $\mathbf{d} \in \mathcal{C}$, thus qualifying VAF as a valid coefficient of determination.

3.3.2 Heuristic Iterative Projection

IP can also be applied as a heuristic method when the specific constraints identifying the additive tree structure representation for a given data set are unknown. Heuristic IP essentially relies on the same algorithm as non-heuristic IP, but the necessary inclusion of a construction step for determining the constraints calls for modifications to serve the purpose of a heuristic search strategy. First and foremost, repetitively cycling through construction and estimation may not necessarily result in the sequence $\mathbf{d}^{(t)}$ converging to $P_{\mathcal{C}}(\mathbf{p}) = \mathbf{d}^*$, the vector of optimal tree distances, as Hubert and Arabie (1995b) report. Rather, augmentation, also required with heuristic IP due to the specific nature of the side-constraints, may cause an indeterminacy problem such that the algorithm eventually oscillates through a fixed collection of vectors associated with different constraint sets. Based on the notion that IP without augmentation always produces a convergent sequence $\mathbf{d}^{(t)}$, but possibly not to the desired optimum \mathbf{d}^* , Hubert and Arabie (1995b) recommend as heuristic IP search strategy a two-step procedure, initially employing IP with augmentation until oscillation occurs, continued by IP without augmentation of a randomly chosen vector from the set of oscillating solutions, which will always converge (“quadruple reduction method”, see Barthélemy & Guénoche, 1991, pp. 66–77). Hubert and Arabie (1995b) advocate a final ‘polish’ for the resulting distances by an application of non-heuristic IP to \mathbf{p} onto the \mathcal{C}_w as they were identified at the terminal stage of convergence. Of course, the VAF criterion can also be used to assess the fit of a heuristic additive tree structure. It is obvious that the additive tree representation obtained through heuristic IP depends on the initial order of the elements in \mathbf{P} because the IP algorithm proceeds by stepwise inspection of all object quadruples O_i , O_j , O_k , and O_l . Thus, to reduce the risk of detecting a purely locally optimal solution, a common nuisance to any heuristic procedure, utilization of heuristic IP with multiple runs on the randomly permuted input proximity matrix \mathbf{P} is strongly recommended to identify a solution with

maximal VAF.

3.4 Locating Additive Trees for Three-Way Data

Algorithms for locating additive tree structures for three-way two-mode data generally follow the rationale that different sources base their judgments on a shared frame of reference. The assumption of a common denominator of perception/judgment is operationalized by the technical requirement of locating a particular collection of trees all restricted to an identical topology as an optimal representation of the S individual proximity matrices. The topological structure of an additive tree is uniquely determined by the equality constraint on the two largest sums of pairs of distances for each quadruple of objects. Imposing the restriction of an identical topology onto a collection of S additive trees translates into the constraint of those two largest sums of each quadruple of objects referring to the same pairs of distances across all sources S . Individual variation is modelled through differential shrinking or stretching of branch lengths. Imposing the same-topology condition can be done in two very different ways.

INDTREES (Carroll et al., 1984), a generalization of the Carroll-Pruzansky (1980) penalty algorithm, imposes the same topology condition through the ultrametric-centroid decomposition of additive trees. Initially, an ultrametric representation of the average of the individual proximity matrices is identified from hierarchical clustering. Subsequently, the individual proximity matrices are fit against the average representation by the conjugate gradient method. For each source s , conformity of the distance estimates to the constraints defined by the ultrametric inequality is enforced, as in the two-way case, through a penalty term securing equality of the two largest distances. The same-topology condition is put into effect through a second penalty term imposing the same two pairs of distances of each object triple O_i, O_j, O_k to be the largest two for all sources. Construction of source-specific additive tree structures is accomplished through subsequent least-squares estimation of individual star components to the residual proximities $p_{ij(s)} - u_{ij(s)}$. INDTREES iterates through the two estimation steps until convergence. As INDTREES does not fit directly an additive tree to a proximity matrix, one should observe the important technical issue arising when we wish to fit multiple additive tree structures: INDTREES generates multiple ultrametric tree representations through successive residualization of the input proximity data, then followed by fitting a single star tree to the remaining residuals. No explanation is offered how to distribute the star component across the previously retrieved multiple ultrametric structures to retrieve the desired multiple additive trees. Also, we need to re-emphasize that INDTREES has never been made available as a fully implemented computer program.

The task of locating additive tree structures to three-way two-mode data through IP for analyzing individual variability is addressed within a deviation-from-the-mean framework: based on the individual proximity matrices aggregated across sources (usually by averaging), denoted by \mathbf{P}_a , a best-fitting reference tree structure \mathbf{D}_a^* is identified through application of heuristic IP, against which the individual data are fit. Concretely, the additive tree representation \mathbf{D}_a^* of \mathbf{P}_a is determined by the associated collection of constraint sets $\mathcal{C}_{1(a)}, \dots, \mathcal{C}_{W(a)}$ as identified by heuristic IP. In the subsequent analysis of individual source variability, $\mathcal{C}_{1(a)}, \dots, \mathcal{C}_{W(a)}$, is treated as the given set of constraints for non-heuristic IP used for fitting the S individual proximity matrices. Thus, the specific additive tree structure \mathbf{D}_a^* serves as frame of reference for the confirmatory fitting of the individual additive trees of sources s , thereby restricting them to the same topology; individual variability is modelled through constrained least-squares estimation of differential branch lengths. Clearly, in comparison with INDTREES, the direct construction of three-way additive trees, without an intermediate ultrametric-centroid decomposition and not involving penalty terms, represents the most distinctive feature of the proposed IP-based routine. The individual loss functions to be minimized are defined as

$$\min_{\mathbf{D}_s} \{L(\mathbf{D}_s)\} = \min_{\mathbf{D}_s} \left\{ \sum_{i < j} (p_{ij(s)} - d_{ij(s)})^2 \right\} = \min_{\mathbf{D}_s} \left\{ \frac{1}{2} \text{tr}(\mathbf{P}_s - \mathbf{D}_s)(\mathbf{P}_s - \mathbf{D}_s)' \right\},$$

with \mathbf{D}_s subject to the sets of linear inequality constraints, $\mathcal{C}_{1(a)}, \dots, \mathcal{C}_{W(a)}$, associated with \mathbf{D}_a^* . The VAF criterion obtained for each individual tree serves as a fit index quantifying how closely the fitted additive tree distances reflect the properties of the reference structure:

$$\text{VAF}_s = 1 - \frac{\sum_{i < j} (p_{ij(s)} - d_{ij(s)}^*)^2}{\sum_{i < j} (p_{ij(s)} - \bar{p}_s)^2} = 1 - \frac{\text{tr}(\mathbf{P}_s - \mathbf{D}_s^*)(\mathbf{P}_s - \mathbf{D}_s^*)'}{\text{tr}(\mathbf{P}_s - \bar{\mathbf{P}}_s)(\mathbf{P}_s - \bar{\mathbf{P}}_s)'}. \quad (3.4)$$

3.5 Applications

As already mentioned, unfortunately, INDTREES by Carroll, Clark, and DeSarbo (Carroll et al., 1984) is not available as a computer program. INDTREES would have been the first, and most natural choice for a comparative evaluation of the performance of our combinatorial algorithm for locating additive tree representations of three-way data. However, one might generally consider such a comparison as not particularly germane, because of the indeterminacy of the INDTREES solutions with regard to the distribution of the single centroid component across the multiple ultrametric tree structures.

Table 3.1: Heuristic additive tree fitting to the aggregated proximity data, \mathbf{P}_a : frequency distributions of the VAF scores, as obtained from 10,000 random starts.

VAF	Frequency
.749	3
.784	4
.790	7
.791	7
.792	1
.793	2
.797	3
.801	14
.802	464
.803	115
.804	16
.805	254
.836	3941
.847	11
.852	5158

3.5.1 Three-Way Additive Tree Structures for Schematic Face Stimuli

Before presenting details and results of the analysis, a brief comment regarding the additive tree representation finally selected seems in order. Initial experimentation with a single additive tree structure (not reported here) suggested that a biadditive tree model provides a more adequate and accurate representation of the structural properties of the data. The inclusion of a second tree structure appears to ‘clean up’ the first tree component, a phenomenon presumably due to the correlation between the two structures, which in turn is an immediate consequence of the cyclic fitting scheme. The biadditive representation lends itself to a more convincing interpretation than could be deduced from the single additive tree structure. So, our preference is with the former because of the methodological (rather than substantive) points to be taken, the choice of a single or biadditive representation seems irrelevant.

Finding and fitting the biadditive tree structure was carried out through a MATLAB routine, `biatreefnd.m` (see Hubert et al., 2006), performing heuristic IP on the vector of proximities in the average matrix derived from the 22 individual data matrices, denoted \mathbf{P}_a . Conveniently, `biatreefnd.m` allows for multiple runs with initial random permutations of the input proximity matrix; we used 10,000 random starts. The frequency distribution of the VAF scores obtained is reported in Table 3.1.

The solution with the largest VAF value (.852) was chosen as reference additive tree structure against which we fit the 22 individual rating matrices \mathbf{P}_s . The tree graphs of the biadditive reference structure are presented in Figure

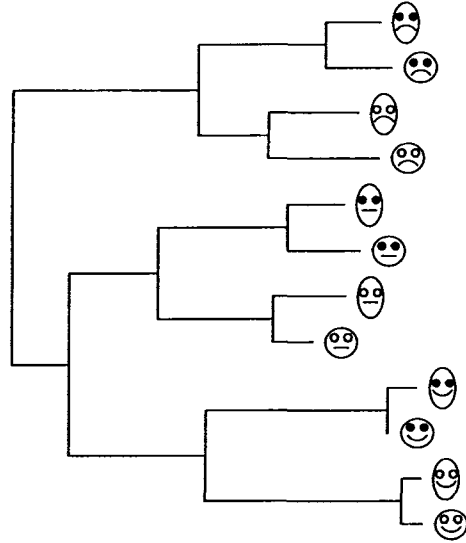
3.1; their interpretation is straightforward (all trees have been rooted utilizing the Sattath-Tversky minimum variance criterion [see Sattath & Tversky, 1977]). Clearly, the first additive tree structure identifies three segments of face stimuli based on the primary criterion emotional appearance as implied by the factor Mouth with its levels ‘frown’, ‘flat’, and ‘smile’. Notice that apparently the three categories are not perceived as equally distinct, rather ‘flat’ and ‘smile’ are merged, while ‘frown’ is still set apart. Within each group a secondary separation into faces with ‘open’ versus ‘solid’ circled eyes can be observed, while the factor Facial Shape differentiates between stimuli on the tertiary level. The second additive tree structure produces a perfectly balanced grouping dominated by the feature Facial Shape, subsequently contrasting stimuli with ‘open’ and ‘solid’ circled eyes, while emotional appearance is employed as tertiary criterion.

For a deeper understanding of the relation between the hierarchy of VAF scores and the corresponding tree representations, graphs of all solutions reported in Table 3.1 were obtained. The worst solution (.749) differs from the best only in terms of the first tree structure with the order of secondary and tertiary criteria, Eyes and Facial Shape, reversed. Solutions with VAF values in-between those two extremes basically represent intermediate stages as resulting from transforming the worst into the best additive tree representation.

Fitting the biadditive tree structure for the 22 sources was carried out through the MATLAB routine `biatreefit.m`, performing non-heuristic IP on each of the individual proximity vectors derived from the \mathbf{P}_s , with constraints defined by those detected when locating the biadditive reference structure through `biatreefind.m`. Table 3.2 presents the sorted VAF scores of the 22 sources, due to (3.4), indicating how closely the various individuals were actually represented by the reference tree structure. Subjects 5, 12, and 17 form the top three group, while the bottom three comprise subjects 1, 14, and 2. For succinctness, we will only consider the two extremes of individuals exceptionally poorly or well-represented, the top and bottom three. Figure 3.2 presents the additive tree graphs; due to limited space only the first additive structure is displayed. As sole representative of the top three group, subject 5 is chosen, whereas the additive tree structures of subjects 1, 14, and 2 are completely documented because they promise to provide deeper diagnostic insight into how sources with data of apparent ill-fit are handled by `biatreefit.m`. Not too surprising, the additive tree structure of subject 5 matches the reference representation closely, which to a lesser degree applies to subjects 2 and 14, though both tree structures display several ties, indicated by vertical bars joining more than two branches at a time. Ties can occur as a result of remedying violations of the four-point condition through averaging the involved distances. For subject 2, the ‘smile’ segment conforms well to the reference structure, while the ‘flat’ and ‘frown’ categories are tied, draping the (arbitrarily) split ‘frown’ category around the ‘flat’ seg-

Figure 3.1: Biadditive reference tree representation (VAF=.8521) of the aggregated proximity matrix, \mathbf{P}_a .

First Structure



Second Structure

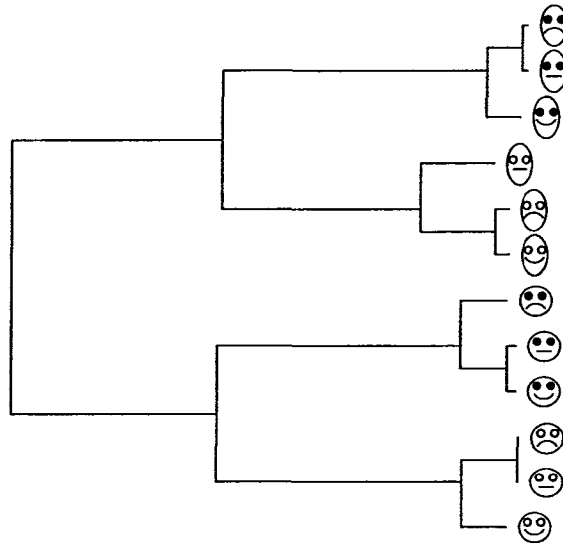


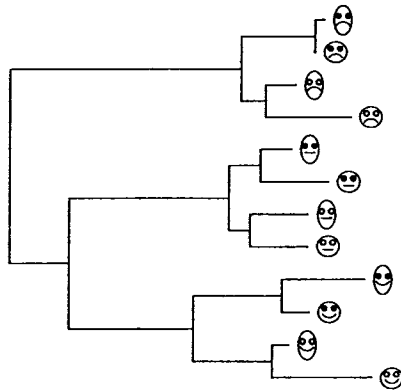
Table 3.2: Ranking of individual fit measures.

Source	VAF
5	.857
12	.849
17	.818
.....	
11	.817
20	.789
16	.787
13	.771
22	.771
19	.756
7	.732
21	.723
9	.722
15	.705
3	.683
10	.669
4	.660
18	.638
8	.629
6	.623
.....	
2	.580
14	.469
1	.104

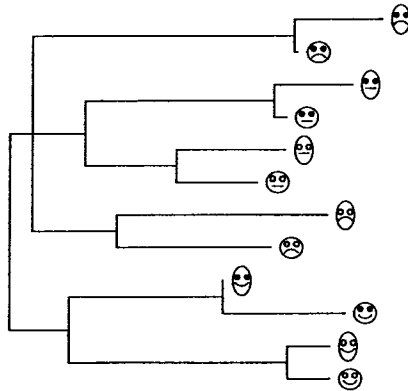
ment. Such inversions do not occur with subject 14. Superficially, in its lay-out the additive tree structure perfectly matches the reference tree. However, closer inspection reveals ties between the 'flat' and 'smile' segment as well as within the 'frown' category, which compromise the representation because obviously for this particular source conformity to the reference structure could only be attained through tied distance estimates. The individual additive tree structure obtained for subject 1 appears extremely distorted, due to numerous ties, and is basically non-interpretable.

Figure 3.2: Individual biadditive tree representations for selected subjects: first structure.

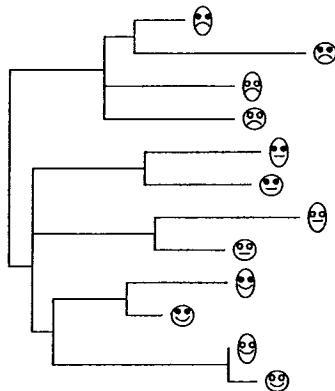
Subject 5 (VAF=.857)



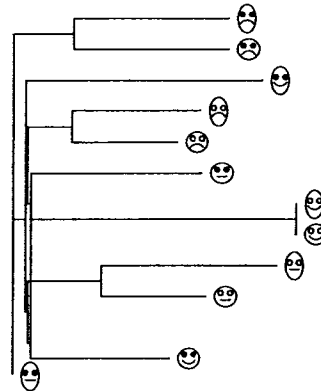
Subject 2 (VAF=.580)



Subject 14 (VAF=.469)



Subject 1 (VAF=.104)



To further explore the deviant perspective of the three bottom subjects as suggested by their tree representations, additional insight into the exact nature

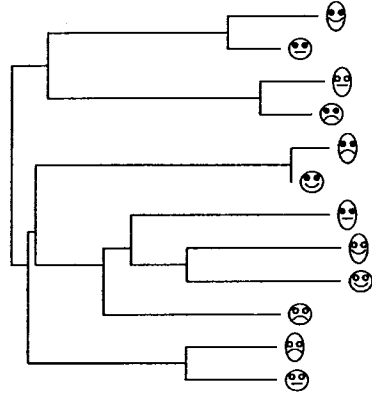
of their judgments seems desirable. Therefore, in treating respondents 1, 14, and 2 as independent sources, their dissimilarity data were re-analyzed, with separate biadditive tree representations to be fit heuristically. The main objective is to determine whether subjects just provided random judgments, or whether they applied idiosyncratic criteria that simply did not match the rest of the sample. Each individual proximity matrix was subjected to 10,000 random starts of `biatreefnd.m`. For each source the solution with the highest VAF score was chosen as its final representation. The respective graphs of the first additive tree structure are presented in Figure 3.3. Looking at the tree graph for subject 1 suggests an immediate explanation: the arrangement of the face stimuli reveals no interpretable pattern. Obviously, subject 1 just made random judgments. The situation is different with subjects 14 and 2. The tree diagrams reveal that they obviously place different priority on the criteria for judging the face stimuli as compared to the reference tree. Both respondents primarily use eye shape to distinguish between the twelve schematic faces, while Facial Shape (subject 14) and emotional appearance (subject 2) serve (not too consistently) as secondary criteria.

3.6 Discussion and Conclusion

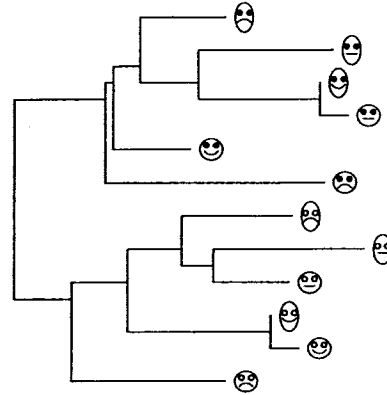
Smith (1998) in a comprehensive evaluation of three heuristic additive tree fitting algorithms, IP, ADDTREE devised by Corter (1982, 1996) based on Sattath and Tversky's (1977) original work, and De Soete's (1983) penalty function approach, observed that heuristic IP consistently outperformed its competitors in locating the best-fitting additive tree. The critical reader may be tempted to suggest further research to explore whether this result can also be validated for IP as a tool for modelling individual differences through additive tree structures as proposed in this paper, in comparison, say, with INDTREES (Carroll et al., 1984). However, INDTREES is not available. Still, we considered refitting the examples given by Carroll and his collaborators (Carroll et al., 1984). Unfortunately, as soon as more than a single additive tree structure is fit to the data, the results are incompatible with those obtained through the combinatorial additive tree algorithm, because INDTREES fits multiple ultrametric components and only a single centroid structure — so, how to actually split the centroid structure to reconstruct multiple additive tree components? Recall that our IP-based algorithm directly fits additive tree structures without a decomposition step. Therefore, we must restrict ourselves to the indirect conclusion that, given incidental evidence and Smith's results, IP additive tree fitting very likely is superior to INDTREES (recall that INDTREES is a generalization of the Carroll-Pruzansky [1980] penalty algorithm that was later on refined by De Soete [1983] into a procedure no longer relying on the ultrametric-centroid

Figure 3.3: Independent biadditive tree representations for the bottom three subjects: first structure.

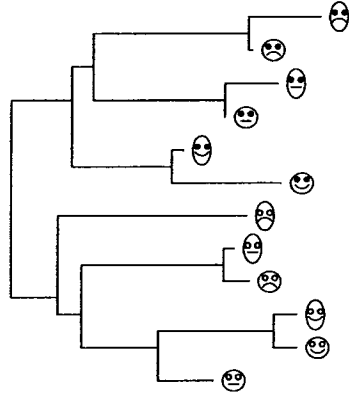
Subject 1 (VAF=.862)



Subject 14 (VAF=.805)



Subject 2 (VAF=.802)



decomposition).

The approach adopted in this paper is guided by a principle common in statistics as well as of immediate intuitive appeal, namely, to analyze individual variability within a deviation-from-the-mean framework: based on the aggregated individual proximity matrices, an additive reference tree is located representing the average structure, against which the individual data are fit directly in a confirmatory manner, without involving an intermediate ultrametric-centroid decomposition nor penalty terms.

Lastly, as an alternative approach rooted in a very different philosophy of how to analyze three-way data, we need to mention consensus tree modelling: initially, tree representations are found and fit to each individual data source

independently, followed by an integrative step aimed at locating a prototypical tree representation that captures a maximum of the individual variability observed among the independent source trees, analogous to a majority voting rule. The crucial question of which way is the ‘best’ may not have a definitive answer, which rather, at least to a certain extent, depends on personal preference as well as the objective of a study and specific features of the data at hand.

Disregarding all these methodological peculiarities, it remains unfortunate that additive tree representations as a data analytic tool appear to be rarely used in Psychology. The reasons for this may be in some part due to the relative inaccessibility of appropriate software to the potential user of a more applied inclination. Only SYSTAT offers an implementation of additive tree construction and fitting through the ADDTREE algorithm that, however, does not allow for representation of individual tree differences. An obvious conclusion to this manuscript is that if one is interested in applying additive tree structures to three-way two-mode proximity data, the MATLAB heuristics `atreefnd.m` (for fitting a single tree structure) and `biatreefnd.m`, as well as their non-heuristic counterparts `atreefit.m` and `biatreefit.m` would be natural choices to consider.

4 Multidimensional Scaling

The development of models for multidimensional scaling (MDS) that explicitly account for individual differences in, for example, the judgment or perception of objects has been a research topic of longstanding interest in quantitative psychology (see the monograph by Arabie, Carroll, & DeSarbo, 1987, or the comprehensive reviews by Carroll & Arabie, 1980, 1998). The most prominent approach to individual differences scaling is through what is usually called the weighted Euclidean model. Based on a common group stimulus space, derived from the aggregated sample data, individual object configurations for each subject ('private spaces') are represented in reference to this common space through differential weighting of its dimensions, thereby shrinking or extending coordinates of objects along the respective axes. A number of methods for the estimation of the weighted Euclidean model for individual differences scaling has been implemented, such as INDSCAL (Carroll & Chang, 1970; Carroll, 1972), SINDSCAL (Pruzansky, 1975), ALSCAL (Takane, Young, & de Leeuw, 1977), MULTISCALE (Ramsay, 1977), and most recently, PROXSCAL (Commandeur & Heiser, 1993; Busing, Commandeur, & Heiser, 1997; Meulman, Heiser, & SPSS, 1999), as part of the CATEGORIES module in SPSS from version 10 onwards. All these methods, as implied by the name of the underlying model, attempt to fit Euclidean distances. The goal of this paper is to introduce a method for individual differences scaling within the city-block metric.

Attneave (1950) raised the issue of an adequate metric for scaling stimuli. He introduced the (phenomenological) distinction between analyzable and integral stimuli, with the latter representing a holistic mix of constituting dimensions that at an observer's perceptual level are assumed extremely difficult to decompose into well-defined, separate contributions; the dimensions are 'unanalyzable'. Colors varying in lightness and saturation may serve as the standard example. Analyzable stimuli, in contrast, are postulated to be composed of perceptually separable dimensions. Examples are forms varying in size and orientation. Empirical evidence suggests that for integral stimuli, distances in psychological space are best approximated by the Euclidean metric, while the city-block metric allows for superior accuracy in mapping the distance structure of analyzable stimuli (for reviews, see Arabie, 1991; Garner, 1974; Shepard, 1991; for general theoretical results on the city-block metric the reader may consult Fichet (1987), Le Calvé (1987); for a survey of applications of the city-block metric in statistics, see Dodge, 1997).

In light of these findings, it is surprising that, to our knowledge, only two attempts for incorporating the city-block metric into individual differences scaling have been published in the literature (Okada & Imaizumi, 1980; Heiser, 1989a). In both instances the city-block model is fit using gradient-, or subgradient-based optimization. To avoid the well-documented difficulty of local minima when fitting city-block models (see De Leeuw & Heiser, 1977; Hubert & Arabie, 1986, 1988) the method developed here utilizes combinatorial optimization that instead of trying to identify optimal object coordinates by solving the stationary equations associated with a specific loss function, attempts to solve the optimization problem through finding an arrangement, or permutation of objects along a continuum that is optimal in the sense that the order-induced object locations immediately yield coordinates minimizing the given loss function. The combinatorial rationale for city-block scaling, comprising unidimensional and multidimensional scale construction, can also accommodate individual differences scaling within a city-block metric.

The subsequent section summarizes the key features of the weighted Euclidean as well as the city-block model for individual differences scaling. A review of the theoretical underpinnings of the combinatorial approach to scale construction, and its extension to individual differences scaling in particular, will be presented in the next two sections; a third section gives an illustrative application of combinatorial individual differences scaling to schematic face stimuli within a city-block metric. We conclude with a discussion of our findings in the last section.

4.1 The Weighted Euclidean and the City-Block Model for Individual Differences Scaling

For a given set of objects $\mathcal{O} = \{O_1, \dots, O_N\}$ the Euclidean distance between objects O_i and O_j , with $1 \leq i, j \leq N$, in M -dimensional (psychological) space is defined by

$$d_{ij}(\mathbf{X}) = \left(\sum_m (x_{im} - x_{jm})^2 \right)^{\frac{1}{2}},$$

where x_{im} and x_{jm} denote the coordinates of O_i and O_j on the m^{th} dimension, $m = 1, \dots, M$, and \mathbf{X} the $N \times M$ coordinate matrix for the entire set \mathcal{O} . Through differential weighting of the axes of the psychological space, the weighted Euclidean model allows for constructing individualized psychological spaces for different data sources (i.e., individuals, situations, scenarios, experimental conditions, etc.). The weighted Euclidean distance between objects O_i and O_j within the representational space for source s , with $s = 1, \dots, S$, can be

defined as:

$$\begin{aligned} d_{ij(s)}(\mathbf{X}, \mathbf{W}_s) &= \left(\sum_m (w_{mm(s)}x_{im} - w_{mm(s)}x_{jm})^2 \right)^{\frac{1}{2}} \\ &= \left(\sum_m w_{mm(s)}^2 (x_{im} - x_{jm})^2 \right)^{\frac{1}{2}} \end{aligned}$$

where \mathbf{W}_s denotes an $M \times M$ diagonal matrix of nonnegative weights $w_{mm(s)}$ for every dimension m for source s . Thus, in matrix notation the individualized spatial coordinates of the elements in \mathcal{O} for source s , are given by

$$\mathbf{X}_s = \mathbf{X} \mathbf{W}_s.$$

In other words, the weights $w_{mm(s)}$ account for differential salience, or importance of the m^{th} dimension for source s through shrinking or extending dimension m of the psychological space. Typically, object coordinates and dimension weights are estimated such that the least squares loss function is minimized.

The city-block model for individual differences scaling retains the concept of differential weighting of dimensions as a means for constructing the individualized psychological space for source s , but the inter-object distances are defined within the city-block metric:

$$\begin{aligned} d_{ij(s)}(\mathbf{X}, \mathbf{W}_s) &= \sum_m |w_{mm(s)}x_{im} - w_{mm(s)}x_{jm}| \\ &= \sum_m w_{mm(s)} |x_{im} - x_{jm}|. \end{aligned}$$

Heiser (1989a), for example, minimizes the least squares criterion through a subgradient approach, employing an iterative majorization algorithm, combined with alternating least squares estimation, and a pairwise interchange heuristic of the distance estimates as a precaution against local minima, as suggested by Hubert and Arabie (1988). As an exemplary application, Heiser fits his model to the “Cola” data analyzed by Schiffman, Reynolds, and Young (1981), using, among other programs, INDSCAL, SINDSCAL, ALSCAL, and MULTISCALE. Heiser reports satisfactory results that come close to those obtained through MULTISCALE by Schiffman, Reynolds, and Young (1981).

4.2 Combinatorial Scaling

Basically, any proximity-based scale construction, uni- or multidimensional, can be represented as a combinatorial optimization problem as it amounts to identifying a specific object ordering along one or several dimensions, among all possible object arrangements, which after translation into coordinates, gives inter-object distances approximating the observed proximities in a least-squares opti-

mal sense. Hubert, Arabie, and Meulman (Hubert & Arabie, 1986, 1988; Hubert et al., 2002, 2006) initially developed a combinatorial optimization framework for unidimensional scaling that with some minor algorithmic adjustments, readily generalizes to the task of representing a set of objects in M -dimensional space, as well as to multidimensional individual differences scaling within the city-block metric.

4.2.1 Combinatorial Unidimensional Scaling

Let $\mathbf{P} = \{p_{ij}\}$ denote an $N \times N$ square-symmetric two-way one-mode proximity matrix. Constructing a unidimensional scale of the N objects, based on their proximities, requires estimation of coordinates x_1, \dots, x_N representing the object locations such that the induced $N(N-1)/2$ interpoint distances, $d_{ij} = |x_j - x_i|$, approximate the proximities in \mathbf{P} optimally, minimizing the least-squares criterion:

$$L(\mathbf{x}) = \sum_{i < j} (p_{ij} - |x_j - x_i|)^2 = \frac{1}{2} \text{tr}(\mathbf{P} - \mathbf{D})(\mathbf{P} - \mathbf{D})', \quad (4.1)$$

with $\mathbf{D} = \{|x_j - x_i|\}$. Without loss of generality, the sum of the coordinates, $\sum_i x_i$, is restricted to zero, which does not affect the value of the loss function (i.e., any set of values x_1, \dots, x_N can be replaced by $x_1 - \bar{x}, \dots, x_N - \bar{x}$, with $\bar{x} = (1/N) \sum_i x_i$).

The standard iterative gradient-based algorithm to minimizing (4.1), as proposed, for example, by Guttman (1968), upon convergence yields a set of coordinates x_1, \dots, x_N , satisfying the necessary condition (based on the stationary equations of $L(\mathbf{x})$)

$$x_i = \frac{1}{N} \sum_j p_{ij} \text{sign}(x_j - x_i), \quad (4.2)$$

with

$$\text{sign}(x_j - x_i) = \begin{cases} 1 & \text{if } x_j - x_i < 0 \\ 0 & \text{if } x_j - x_i = 0 \\ -1 & \text{if } x_j - x_i > 0 \end{cases}$$

(see De Leeuw & Heiser, 1977; Hubert & Arabie, 1986, 1988). It holds for any proximity matrix \mathbf{P} that the coordinates obtained from the gradient-based algorithm possess the property

$$x_1 = t_1 \leq \dots \leq x_N = t_N,$$

with t_i defined by

$$t_i := \frac{1}{N} (u_i - v_i), \quad (4.3)$$

where u_i denotes the sum of entries in the i^{th} row of matrix \mathbf{P} from the extreme

left up to the main diagonal:

$$u_i := \sum_{j=1}^{i-1} p_{ij} \quad \text{for } i \geq 2;$$

v_i indicates the sum from the main diagonal to the extreme right of \mathbf{P} :

$$v_i := \sum_{j=i+1}^N p_{ij} \quad \text{for } i < N.$$

Note that $u_1 = v_N = 0$. Thus, the gradient procedure permutes the input proximity matrix such that for consecutive rows, the differences between the sums of row entries to the left and right of the main diagonal are monotonically increasing, with the immediate choice for the wanted coordinates, $x_i = t_i$ (for a detailed derivation, see Appendix I).

Obviously, the size of the t_i depends directly on a given row/column permutation $\rho(\cdot) \equiv \rho \in \Omega$ of \mathbf{P} (with Ω indicating the set of feasible permutations of the first N integers). Note that for square-symmetric matrices no canonical order of their rows and columns exists, hence any simultaneous rearrangement of rows and columns is legitimate. In general, a matrix displaying the pattern $t_1 \leq \dots \leq t_N$ is said to be in monotonic form. Unfortunately, the monotonic form of a matrix is not unique, and as a consequence neither are the coordinates provided by the iterative gradient algorithm. In fact, for any given proximity matrix \mathbf{P} , a multitude of gradient-based solutions for the coordinates x_1, \dots, x_N exists, all fulfilling the necessary condition in (4.2). Also, satisfaction of (4.2) alone does not guarantee a global minimum of the least-squares loss function in (4.1), as will become obvious after some algebraic manipulation, relying on the restriction $\sum_i x_i = 0$, and the additional constraint $x_1 \leq \dots \leq x_N$, leading to

$$L(\mathbf{x}) = \sum_{i < j} p_{ij}^2 + N \left(\sum_i (x_i - t_i)^2 - \sum_i t_i^2 \right) \quad (4.4)$$

(for details, see Appendix II). For a gradient-based solution satisfying (4.2), $\sum_i (x_i - t_i)^2$ in (4.4) equals zero because $x_i = t_i$. But, minimizing $L(\mathbf{x})$, in addition, depends on the size of term $\sum_i t_i^2$ that the gradient algorithm neglects. More succinctly, the gradient-based approach does not control for the particular monotonic form used for \mathbf{P} , also maximizing $\sum_i t_i^2$. Therefore, the resulting coordinates usually represent only a feasible solution and not necessarily an optimal one. As Hubert, Arabie, and Hesson-McInnis (1992) show, these findings generalize to settings where a multidimensional representation (i.e., $M > 1$) of a set of objects within the city-block metric is sought.

Under the assumption that the optimal order of objects O_1, \dots, O_N along the continuum is known, Defays (1978) demonstrates that the minimization of (4.1) can be re-phrased as a least-squares problem with a closed form solution

for the spacings between objects, from which their actual coordinates can be deduced immediately (for details, see Appendix III). In addition, Defays (1978) verifies that, given the specific object order maximizing $\sum_i t_i^2$, the loss function of (4.1), indeed, is minimized solely by letting $x_i = t_i$, for $1 \leq i \leq N$, where the t_i are retrieved using (4.3). To summarize, if we manage to find the monotonic form of \mathbf{P} that gives the optimal object order, we can readily provide the desired globally optimal coordinates from $x_i = t_i$. Formally, if ρ denotes a permutation of \mathbf{P} yielding a particular object order, the optimization problem of finding ρ^* and $x_1^* \leq \dots \leq x_N^*$, defining a global minimum of the loss function in (4.1), can be formalized as

$$\sum_{i < j} (p_{\rho^*(i)\rho^*(j)} - |x_j^* - x_i^*|)^2 =$$

$$\min \left\{ \sum_{i < j} (p_{\rho(i)\rho(j)} - |x_j - x_i|)^2 \mid \rho \in \Omega; x_1 \leq \dots \leq x_N; \sum_i x_i = 0 \right\}.$$

As already indicated, the identification of ρ^* represents an NP-hard combinatorial optimization problem, solvable for small data sets by enumerative methods, but calling for heuristic solution strategies for larger N . The combinatorial optimization strategy for minimizing $L(\mathbf{x})$ focuses on maximizing $\sum_i t_i^2$ through QA — formally:

$$\Gamma(\rho) = \sum_{i,j} p_{\rho(i)\rho(j)} |x_j - x_i| = 2N \sum_i x_i t_{\rho(i)},$$

with $|x_j - x_i|$ denoting fixed interobject distances based on a set of equally-spaced, arbitrary coordinates, $x_1 < \dots < x_i < x_j < \dots < x_N$, satisfying constraints $\sum_i x_i = 0$ and $t_1 \leq \dots \leq t_N$; in other words, the target matrix $\mathbf{B} = \{|x_j - x_i|\}$, is in monotonic form. Hence, maximizing $\Gamma(\rho)$ through finding an optimal permutation will render $\sum_i x_i t_{\rho(i)}$ a maximum, thereby also maximizing $\sum_i t_{\rho(i)}^2$ because $x_i = t_{\rho(i)}$.

The fit of an identified permutation ρ , and associated coordinates is assessed through the VAF criterion:

$$\text{VAF} = 1 - \frac{\sum_{i < j} (p_{ij} - |x_j - x_i|)^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} = 1 - \frac{\text{tr}(\mathbf{P} - \mathbf{D})(\mathbf{P} - \mathbf{D})'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}})(\mathbf{P} - \bar{\mathbf{P}})'}$$

with \bar{p} denoting the mean of the off-diagonal entries in $\mathbf{P} = \{p_{ij}\}$.

Minimizing the least-squares loss function $L(\mathbf{x})$, as given in (4.1) or (4.4), however, corresponds to fitting a regression model through the origin (in familiar generic notation: $y_i = \beta x_i + \epsilon_i$), which — even though justifiable on theoretical grounds in certain instances — in general, has serious disadvantages. First, the residuals usually do not sum to zero; second, the sum of the squared residuals, SSE , might exceed the total sum of squares, $SSTO$, hence the coefficient of determination, $R^2 = 1 - SSE/SSTO$, might turn out to be negative, thus

becoming essentially meaningless. Lastly, using uncorrected SS as a remedy will avoid a negative R^2 , but R^2 will no longer be bounded between zero and one, creating another interpretation problem. Yet, by including an additive constant β_0 in the model, $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$, we ensure that the obtained VAF fit score is equivalent to the (bounded) R^2 measure in regression:

$$\text{VAF} = 1 - \frac{\sum_{i < j} (p_{ij} - (|x_j - x_i| + c))^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} = 1 - \frac{\text{tr}(\mathbf{P} - (\mathbf{D} + \mathbf{C}))(\mathbf{P} - (\mathbf{D} + \mathbf{C}))'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}})(\mathbf{P} - \bar{\mathbf{P}})'},$$

with $\mathbf{C} = \{c\}$ denoting a square matrix containing the additive constant, and main diagonal entries equal to zero. The inclusion of an additive constant changes the least-squares loss function in (4.1) to

$$L(\mathbf{x}, c) = \sum_{i < j} (p_{ij} - (|x_j - x_i| + c))^2 = \frac{1}{2} \text{tr}(\mathbf{P} - (\mathbf{D} + \mathbf{C}))(\mathbf{P} - (\mathbf{D} + \mathbf{C}))'.$$

“Two interpretations exist for the role of the additive constant c . We could consider $|x_j - x_i|$ to be fitted to the translated proximities $p_{ij} + c$, or alternatively, $|x_j - x_i| - c$ to be fitted to the original proximities p_{ij} , where the constant c becomes part of the actual model” to be fitted to the untransformed proximities p_{ij} (Hubert et al., 2006, p. 20). Once c is obtained, constructing a unidimensional scale can be regarded as approximating a transformed set of proximities $\hat{d}_{ij} = p_{ij} + c$ by $d_{ij} = |x_j - x_i|$; in other words, we are fitting a metric model. If the p_{ij} are subjected to an optimal monotone transformation, the model represents nonmetric combinatorial unidimensional scaling. As a final comment, fitting a least-squares intercept model implies the conjecture that the proximities have interval (and not ratio) scale properties.

When including an additive constant in the model, we can no longer rely on setting $x_i = t_{\rho(i)}$, but fitting a unidimensional scale now requires estimating the coordinates and the additive constant c . More succinctly, the inclusion of an additive constant conceptually transforms the construction of a combinatorial unidimensional scale into a two-fold (constrained) least-squares minimization problem, separating into the distinct operations of finding an optimal permutation ρ^* through QA, followed by numerical estimation of the object coordinates and c . Both estimations are carried out through IP. In the context of combinatorial scale construction, we need to solve

$$\min_{\mathbf{d}} \{ (\mathbf{p} - \mathbf{d})'(\mathbf{p} - \mathbf{d}) \},$$

subject to $W = \binom{N}{3}$ inequality constraints \mathcal{C}_w , defined by the object ordering

identified through QA:

$$\begin{aligned}
O_i &\prec O_j \prec O_k \\
\Rightarrow d_{ik} &\leq d_{ij} + d_{jk} \\
\Rightarrow |x_k - x_i| &\leq |x_j - x_i| + |x_k - x_j| \\
0 &\leq |x_j - x_i| + |x_k - x_j| - |x_k - x_i| \quad \forall i < j < k.
\end{aligned}$$

Recall that \mathbf{p} and \mathbf{d} indicate vectorizations of matrices \mathbf{P} and $\mathbf{D} = \{|x_j - x_i|\}$; the actual coordinates, x_i, x_j , are reconstructed from the distance estimates, d_{ij} (for details of the alternating iterative estimation of coordinates and additive constant, see Hubert et al., 1997).

4.2.2 Combinatorial Multidimensional Scaling

The generalization of the combinatorial paradigm to MDS in M -dimensional space rests on fitting multiple unidimensional structures, incorporating the city-block metric, to successive residualizations of the initial proximity matrix \mathbf{P} (unlike Euclidean distances, city-block distances are additive across dimensions). The multidimensional setting is aided by the introduction of an additive constant to accommodate computationally for the successive fitting of possibly negative residuals. The loss function is given by:

$$\begin{aligned}
L(\mathbf{x}_1, \dots, \mathbf{x}_M, c_1, \dots, c_M) &= \sum_{i < j} (p_{ij} - \sum_m (|x_{jm} - x_{im}| + c_m))^2 \\
&= \frac{1}{2} \text{tr}(\mathbf{P} - \sum_m (\mathbf{D}_m + \mathbf{C}_m)) (\mathbf{P} - \sum_m (\mathbf{D}_m + \mathbf{C}_m))'.
\end{aligned}$$

The VAF criterion is defined by

$$\begin{aligned}
\text{VAF} &= 1 - \frac{\sum_{i < j} (p_{ij} - \sum_m (|x_{jm} - x_{im}| + c_m))^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} \\
&= 1 - \frac{\text{tr}(\mathbf{P} - \sum_m (\mathbf{D}_m + \mathbf{C}_m)) (\mathbf{P} - \sum_m (\mathbf{D}_m + \mathbf{C}_m))'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}}) (\mathbf{P} - \bar{\mathbf{P}})'}.
\end{aligned}$$

For the most common case of a two-dimensional configuration, the least-squares loss function becomes

$$\begin{aligned}
L(\mathbf{x}_1, \mathbf{x}_2, c_1, c_2) &= \sum_{i < j} (p_{ij} - (|x_{j1} - x_{i1}| + c_1) - (|x_{j2} - x_{i2}| + c_2))^2 \\
&= \frac{1}{2} \text{tr}(\mathbf{P} - (\mathbf{D}_1 + \mathbf{C}_1) - (\mathbf{D}_2 + \mathbf{C}_2)) (\mathbf{P} - (\mathbf{D}_1 + \mathbf{C}_1) - (\mathbf{D}_2 + \mathbf{C}_2))';
\end{aligned}$$

notice the legitimate interpretation $c = c_1 + c_2$. Initially, a single unidimensional structure, $\{|x_{j1} - x_{i1}| + c_1\}$, is fit to \mathbf{P} , followed by a second structure, $\{|x_{j2} - x_{i2}| + c_2\}$, fit to the residual matrix $\{p_{ij} - (|x_{j1} - x_{i1}| + c_1)\}$. In attempting to improve the fit of the two structures obtained, the residuals

$\{p_{ij} - \sum_m (|x_{jm} - x_{im}| + c_m)\}$ are added back to $\{|x_{j1} - x_{i1}| + c_1\}$, succeeded by re-constructing a first unidimensional structure, with potentially better fit, producing revised residuals $\{p_{ij} - (|x_{j1} - x_{i1}| + c_1)\}$ that in turn are restored to the previously estimated second structure $\{|x_{j2} - x_{i2}| + c_2\}$, followed by re-fitting the second structure, and so on. The process continues by repetitively fitting the residuals from the second structure by the first, and the residuals from the first by the second, until the sequence converges. The procedure of fitting \mathbf{P} through consecutive unidimensional structures generalizes to higher dimensional configurations (i.e., $M > 2$).

4.2.3 Combinatorial Individual Differences Scaling

In addressing the task of combinatorial individual differences scaling, we choose to aggregate the individual proximity matrices \mathbf{P}_s into a single proximity matrix \mathbf{P}_a . Based on the latter, an M -dimensional city-block scaling is constructed representing the common, or group space, as it is referred to in the context of the weighted Euclidean model. The coordinates of objects in the private space, \mathbf{X}_s , are estimated through IP, with restrictions defined by the ordering of objects along the dimensions of the group space. In other words, the object orderings obtained for \mathbf{P}_a are used as confirmatory frame of reference against which the S individual proximity matrices \mathbf{P}_s are fit, with the sole restriction that the order of objects from the group space be preserved. The spacings between objects, however, are free to vary such that the loss function attains a minimum, provided the order constraints imposed by the group space representation are observed. The VAF criterion obtained for each source serves as a fit index quantifying how closely a private space reflects the properties of the common space.

4.3 Application: Scaling of Schematic Faces

Before presenting details and results of the analysis, a brief comment regarding the dimensionality of the final model seems in order: Of course, the factorial design for generating the face stimuli implies a focus on a three-dimensional representation. However, as the scope of this paper is methodological in nature rather than substantive, we decided for the benefit of clarity to confine our presentation to two-dimensional solutions because for the points to be taken here it is irrelevant whether a three- or two-dimensional model is preferred.

Combinatorial individual differences scaling was carried out using the two MATLAB routines `biscalqa.m` and `bimonscalqa.m`. The first, `biscalqa.m`, performs bidimensional scaling of an input proximity matrix within the city-block metric, including the estimation of an additive constant using iterative QA. Thus, the `biscalqa.m` model can be considered equivalent to what Borg and Groenen (1997, p. 161) refer to as metric multidimensional scaling, with the disparities obtained through a linear transformation of the input proximities.

The MATLAB routine `bimonscalqa.m` basically relies on the same algorithm as `biscalqa.m`, but in addition, allows for an optimal monotonic transformation of the input proximity matrix. As `biscalqa.m` and `bimonscalqa.m` are both stand-alone routines, they conveniently allow for multiple runs with initial random permutations of the objects in the input proximity matrix to reduce the risk of ending up with a merely locally optimal solution; 10,000 initial random permutations of the input proximity matrix P_a were used for both `biscalqa.m` and `bimonscalqa.m`. In case of `biscalqa.m` all 10,000 runs converged to the same solution with $VAF = .649$, while for `bimonscalqa.m`, 5220 runs yielded a $VAF = .795$ versus 4780 runs resulting in a $VAF = .796$ (it should be noted that the strict absence of any variability in the VAF values strongly suggests that both MATLAB routines had detected a global optimum). For each of `biscalqa.m` and `bimonscalqa.m`, the solution with the largest VAF value was selected as the two-dimensional group space. The identified object orderings along the two dimensions then served as constraints for the confirmatory fitting of the 22 individual rating matrices P_s .

To gain an intuition for the performance of the two combinatorial routines, it was decided to obtain some ad-hoc benchmarks through readily available commercial software. Of course, none of the two programs chosen performs combinatorial optimization; they rely on gradient or subgradient optimization. However, in both instances the least-squares loss function is used, which can be considered as the common denominator with the combinatorial routines. These supplementary analyses are not intended to imply a comprehensive software evaluation. As an analogue to `biscalqa.m`, the SYSTAT scaling model was used, with settings “R-metric: 1”, “split loss: by matrix”, and “loss function: minimizing Kruskal’s stress with linear transformation”. Thus, SYSTAT performs gradient-based city-block individual differences (interval) scaling. Unfortunately, SYSTAT does not permit multiple random starts. It is mainly for this reason that the seemingly natural choice for a `bimonscalqa.m`-benchmark, the SYSTAT scaling routine with settings “city-block metric” and “minimizing Kruskal’s stress with monotone transformation” of the proximities, was not realized. Instead, we chose SPSS PROXSCAL as it allows for multiple random starts, although it does not support MDS within a city-block metric. However, features like an iterative majorization algorithm (essentially performing subgradient optimization), combined with alternating least squares estimation, make SPSS PROXSCAL a technically up-to-date and refined MDS program (for details see Busing, Commandeur, & Heiser, 1997; Borg & Groenen, 1997; Meulman et al., 1999). SPSS PROXSCAL was run with settings specified to: “source: multiple matrix”, “model: weighted Euclidean”, “transformation: ordinal”, “apply transformation within each source separately”, “no restrictions”, “options: multiple random starts=10000”. The solution of the 10,000 random starts, with the smallest normalized raw stress (NRS) value of .081, equivalent to a “dispersion accounted for” (DAF) of .919 (i.e., $NRS = 1 - DAF$) was chosen.

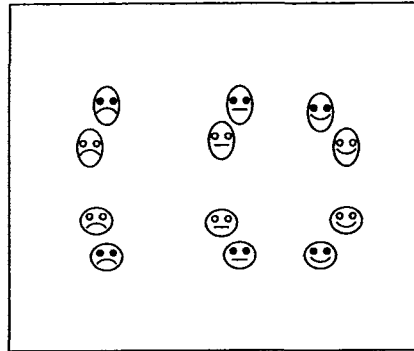
The fit of the SYSTAT solution, in terms of Kruskal's STRESS 1 (S-1), equals 0.318 (which corresponds to a proportion of variance explained (RSQ) of 0.202). The fit measures, however, should rather be seen as relative magnitudes as they are not directly comparable across algorithms; instead, the focus should be on the resulting geometrical representations. The latter are given in Figure 4.1; their interpretation is straightforward. The exact position of each face stimulus is at its left eye. Clearly, the first dimension represents factor Mouth, ordering face stimuli along the continuum negative vs. positive expression of emotion, i.e., from 'frowning' to 'flat' to 'smiling'. The second dimension distinguishes between 'oval' and 'circle' shaped faces, thereby capturing the factor Facial Shape. The most obvious distinction between the two combinatorial and the gradient-based SPSS PROXSCAL and SYSTAT routines is the reversed order of solid and open circled eyes. Both, `bimonscalqa.m` and `biscalqa.m`, place the solid eyed stimuli at the extremes of dimension 2, while faces with open circled eyes occupy the midrange. In contrast, SPSS PROXSCAL and SYSTAT display an interchanging pattern of solid and open circled eyes. It should be noted that the reversed order of solid and circled eyes, as produced by the two combinatorial algorithms, presumably represents an artifact (and disappears when a three-dimensional solution is chosen).

Table 4.1 presents the fit measures for the 22 sources, indicating how closely the various individuals were actually represented by the group spaces (for ease of legibility the original fit scores were multiplied by 1000, thereby eliminating decimal points and leading zeros). It seems most instructive to focus on the extremes, and to consider only individuals who are exceptionally poorly or well-represented. To save space, only the top and bottom three subjects as discovered by the `bimonscalqa.m` routine are selected for further inspection. It was decided to choose `bimonscalqa.m` as a point of reference because, overall it provides a higher fit than `biscalqa.m` due to the inclusion of an optimal monotonic transformation of the dissimilarities. Therefore, in the table below, individuals are sorted according to their VAF scores as obtained through `bimonscalqa.m` (i.e., values in the second column from the left), with the top and bottom three subjects marked by \bullet , and by \circ , respectively. It should be noted that for subject 1 no VAF value is available because the solution is degenerate, i.e., all stimuli lump together at the origin indicating maximal misfit. To facilitate tracking the `bimonscalqa.m`-top-bottom-three pattern across the remaining columns, top and bottom three subjects, in terms of the fit score for the other methods, are also highlighted. Thus, it becomes immediately obvious whether or not a 'bimonscalqa.m-top-or-bottom-three' subject maintains its rank when a different method is used.

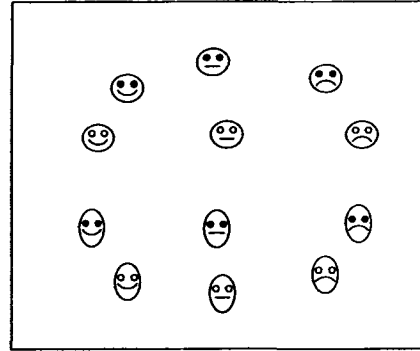
Comparing the ranks of the individual fit measures across the four algorithms shows that `bimonscalqa.m` and SPSS PROXSCAL, apart from minor rank in-

Figure 4.1: Group Spaces

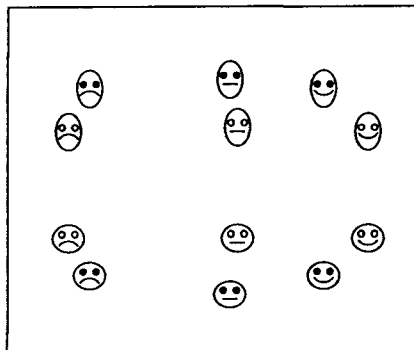
bimonscalqa.m (VAF=.796)



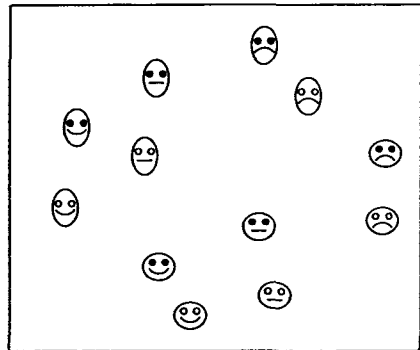
SPSS PROXSCAL (NRS=.081, DAF=.919)



biscalqa.m (VAF=.649)



SYSTAT (S1=.318, RSQ=.202)



terchanges, produce almost identical results: Subjects 12, 20, and 3 form the top three group, while the bottom three are represented by subjects 1, 14, and 2. The top and bottom ranks as obtained through `biscalqa.m` are very close to those of `bimonscalqa.m` and SPSS PROXSCAL, with the single exception of subject 13 now holding rank 3. The ranking of sources yielded by SYSTAT deviates from those of the other three algorithms. In particular, subjects 12 and 20 no longer appear among the top three which instead is composed of subjects 13, 3, and 11; subject 4 drops into the bottom three category.

Figures 4.2–4.5 present the graphs of the private spaces, as obtained through the four methods. As sole representative of the top three group, subject 12 is chosen (see Figure 4.2), whereas the private spaces of subjects 1, 14, and 2 (see Figures 4.3, 4.4, and 4.5, respectively) are completely documented because

Table 4.1: Ranking of individual fit measures.

Ranked Sources	Combinatorial MDS		SPSS		SYSTAT	
	bimonscalqa.m	biscalqa.m	PROXSCAL			
	VAF	VAF	DAF	S-1	RSQ	
12	945 •	767 •	940 •	302	289	
20	903 •	722 •	947 •	313	238	
3	824 •	522 •	935 •	297 •	299 •	•
13	817	666 •	935	285 •	359 •	•
5	812	631	931	305	263	
22	799	567	925	324	167	
16	789	573	930	307	262	
11	782	562	929	298 •	295 •	•
17	776	604	928	309	246	
15	771	576	927	300	289	
10	764	516	921	313	229	
19	756	486	922	310	240	
8	749	512	920	316	210	
18	743	502	917	323	177	
7	731	465	910	317	209	
9	696	467	911	332	129	
21	684	448	921	311	234	
6	617	358	907	334	116	
4	608	386	907	346 ◦	53 ◦	◦
2	580 ◦	301 ◦	881 ◦	352 ◦	18 ◦	◦
14	571 ◦	257 ◦	905 ◦	340	87	
1	— ⁽¹⁾ ◦	18 ◦	877 ◦	349 ◦	35 ◦	◦

(1) Recall that the bimonscalqa.m solution for subject 1 was degenerate.

they provide deeper insight into how the various algorithms handle sources with data of apparent ill-fit. Not surprisingly, the results of all four algorithms for subject 12 (see Figure 4.2) basically coincide with the group space. However, the two combinatorial routines seem to emphasize the second dimension less, distinguishing ‘oval’ and ‘circle’ shaped faces. It should be recalled that due to degeneracy, no private space for subject 1 (see Figure 4.3) is available for the `bimonscalqa.m` algorithm. The private space plot for subject 1 obtained through `biscalqa.m` is clear. The configuration seems to represent random scatter, whereas the two plots by SPSS PROXSCAL and SYSTAT do not provide any hint at the presumably problematic nature of subject 1. Similar findings can be stated for subjects 14 (see Figure 4.4) and 2 (see Figure 4.5). The graphs yielded by `bimonscalqa.m` display all stimuli collapsing into a dense region about the origin; it seems fair to conclude that for both sources, the obtained solutions border on degeneracy. The `biscalqa.m` plots are less dramatic, though on closer inspection the overlap among some stimuli reveals a presumable lack of consistency in judgment or discrimination. The SPSS PROXSCAL as well as the SYSTAT plots, however, look unsuspicious.

4.4 Discussion and Conclusion

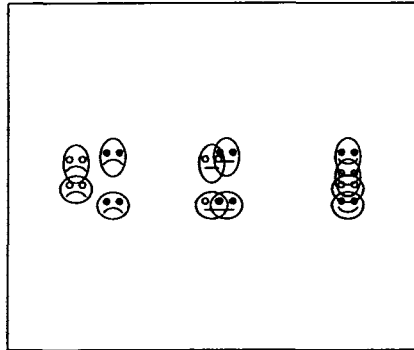
4.4.1 Discussion

To further explore the observed discrepancies between the combinatorial and gradient-based algorithms, additional insight into the exact nature of the judgments given by the three bottom subjects seems desirable. Therefore, their dissimilarity data were re-analyzed within a non-confirmatory setting (i.e., respondents 1, 14, and 2 were treated as singular sources with separate MDS models to be fit). The main objective is to determine whether subjects just provided random judgments, or whether they applied idiosyncratic criteria that simply did not match the rest of the sample. It should be recalled that for all four algorithms, the most salient criteria associated with the two dimensions of the group spaces are emotional expression and facial shape.

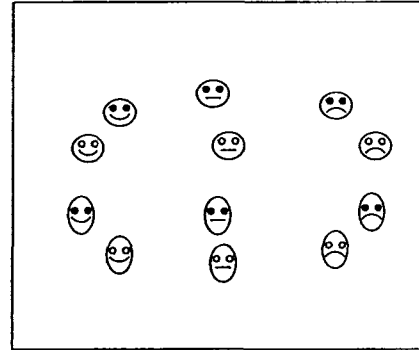
Figure 4.6 presents the four exploratory spaces obtained for subject 1. The `bimonscalqa.m` solution again is degenerate. Looking at the other three plots suggests an immediate explanation: the scatter of the face stimuli reveals no interpretable pattern. Obviously, subject 1 just made random judgments. The

Figure 4.2: Private spaces for subject 12.

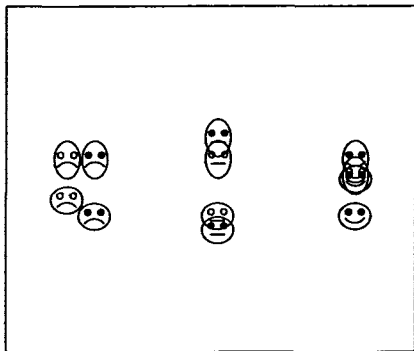
bimonscalqa.m (VAF=.945)



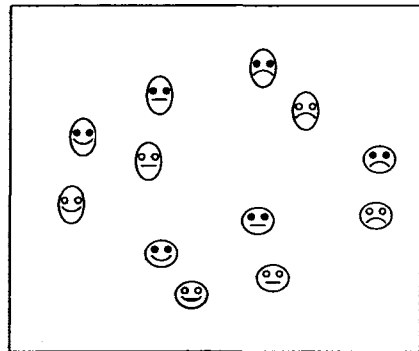
SPSS PROXSCAL (NRS=.060, DAF=.940)



biscalqa.m (VAF=.767)



SYSTAT (S1=.302, RSQ=.289)

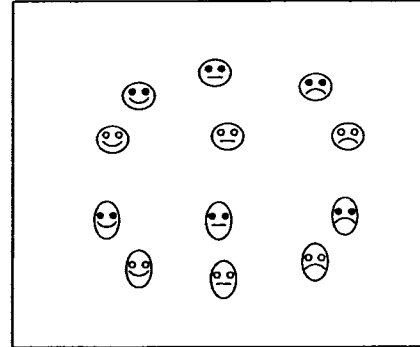


situation is slightly different with subjects 14 (see Figure 4.7) and 2 (see Figure 4.8). The findings from all four algorithms reveal that both respondents obviously place different priority on the criteria for judging the face stimuli as compared to the group spaces. Subject 14 uses Facial Shape and Eye Shape to distinguish between the twelve schematic faces, while subject 2 (not too consistently) utilizes emotional expression and Eye Shape.

Figure 4.3: Private spaces for subject 1.

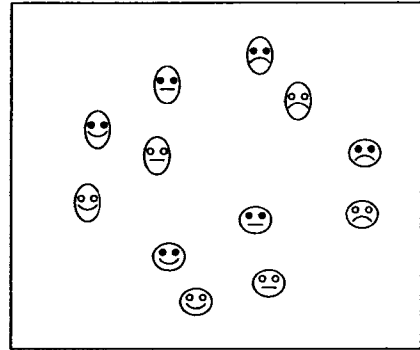
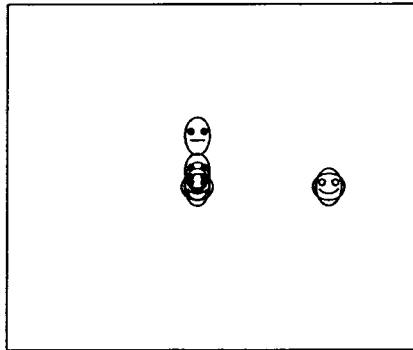
bimonscalqa.m (n.a.)

SPSS PROXSCAL (NRS=.123, DAF=.877)



biscalqa.m (VAF=.018)

SYSTAT (S1=.349, RSQ=.035)



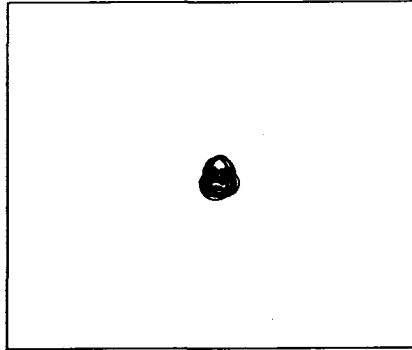
4.4.2 Concluding Comments

An obvious conclusion seems that if one is interested in applying the city-block metric to individual differences scaling (metric or non-metric), the combinatorial MATLAB routines `biscalqa.m` and `bimonscalqa.m` would be natural alternatives to consider. In particular, as widely available commercial software either does not support the city-block metric, or uses a deficient optimization routine prone to local minima. In contrast, the combinatorial search strategy, as implemented in `biscalqa.m` and `bimonscalqa.m`, elegantly avoids the notorious difficulties of gradient-based fitting of city-block models by relying on a mathematically different type of optimization scheme.

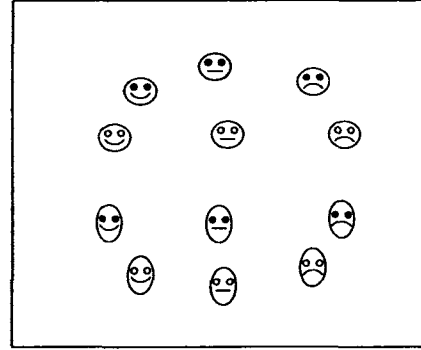
The indication that `bimonscalqa.m`, specifically, might have a slight diagnostic advantage in detecting idiosyncratic data sources that otherwise would

Figure 4.4: Private spaces for subject 14.

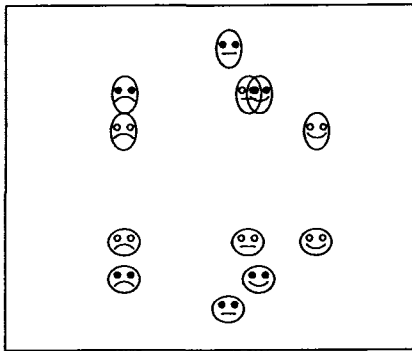
bimonscalqa.m (VAF=.571)



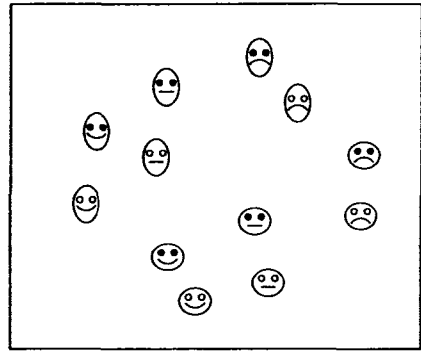
SPSS PROXSCAL (NRS=.095, DAF=.905)



biscalqa.m (VAF=.257)



SYSTAT (S1=.340, RSQ=.087)

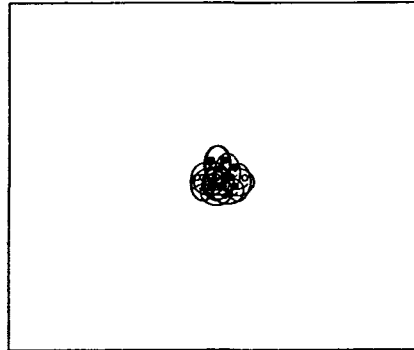


have gone unnoticed, through producing a degenerate, or quasi degenerate solution (as the cases of subjects 1, 14, and 2 imply), can so far only be based on incidental evidence, and deserves further systematic investigation. Besides, in playing devil's advocate, one could simply reverse this suggestion to its opposite: `bimonscalqa.m` is simply more susceptible to degenerate solutions. Upon closer inspection, however, such an argument seems hardly backed by available evidence. As the additional exploratory MDS analyses of the critical data sources demonstrated, these subjects, indeed, applied criteria in judging the schematic faces that strongly disagreed with the communal perspective as captured by the group space. Thus, the flag raised with these 'heretics' by `bimonscalqa.m` appears justified.

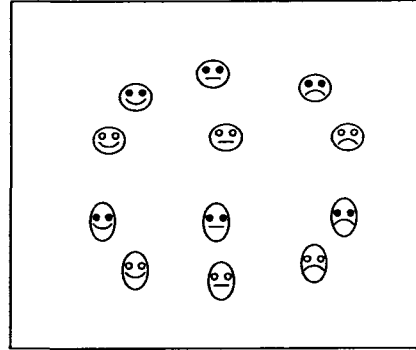
A possible (post-hoc) explanation for this phenomenon could be the specific

Figure 4.5: Private spaces for subject 2.

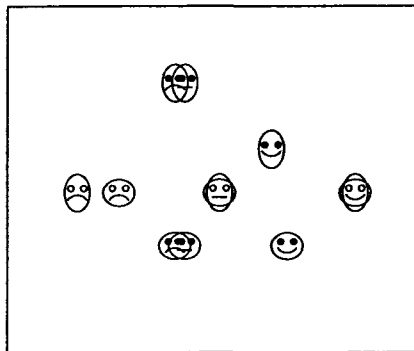
bimonscalqa.m (VAF=.580)



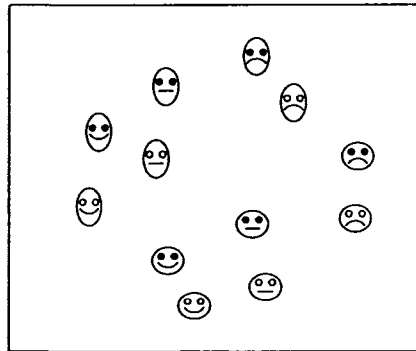
SPSS PROXSCAL (NRS=.119, DAF=.881)



biscalqa.m (VAF=.301)



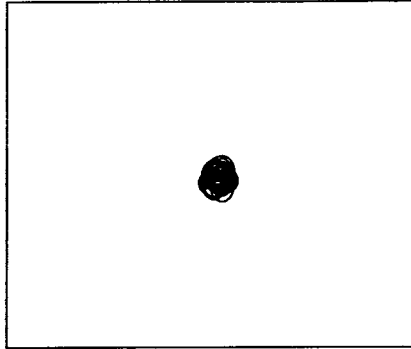
SYSTAT (S1=.352, RSQ=.018)



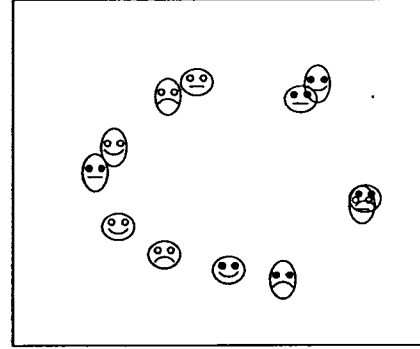
way of the combinatorial approach for generating a very general private space for each source, where the actual coordinates along axes are unique to that source, subject only to the object order constraints of the group space. This strategy offers an individual differences model generalization over the more restrictive weighted Euclidean model where the latter allows only differential axes scaling (through stretching or shrinking) in constructing private spaces. Thus, within the weighted Euclidean model, the relative object spacings on each dimension are completely preserved, thereby confining the mapping of individual differences to linear transformations of the group configuration. Moreover, as `bimonscalqa.m`, applies also an optimal monotone transformation to the input proximities, the degrees of freedom for shifting around the coordinates are even further increased (again, as long as the order restrictions are not violated),

Figure 4.6: Exploratory spaces for subject 1.

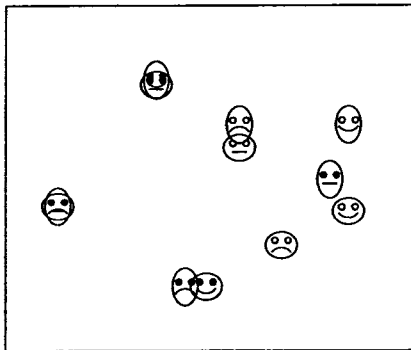
bimonscalqa.m (VAF=.930)



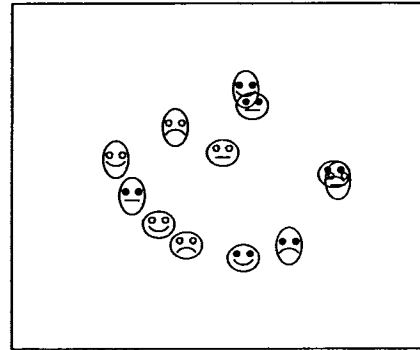
SPSS PROXSCAL (NRS=.055, DAF=.945)



biscalqa.m (VAF=.615)



SYSTAT (S1=.242, RSQ=.543)

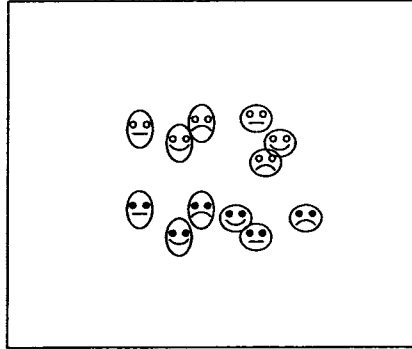


which in turn for an idiosyncratic data source, may cause points to be moved until most or all lump into a single spot, i.e., a degenerate result.

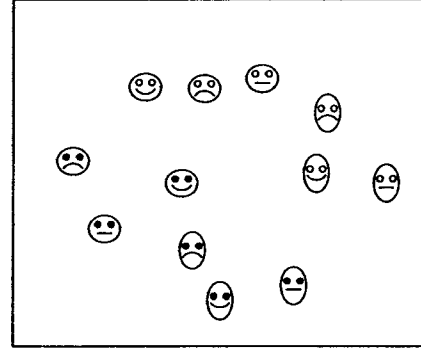
The critical reader may be tempted to suggest further research to explore the method of city-block combinatorial individual differences scaling, in comparison with dimension weighting models (for example, Heiser, 1989a). Such an evaluation is not particularly germane, however, as the two approaches represent different philosophies of how to scale individual differences. The approach adopted in this paper is guided by a principle common in statistics as well as of immediate intuitive appeal, namely, to analyze individual variability within a deviation-from-the-mean framework: based on the aggregated individual proximity matrices, a reference space is identified representing the average structure, against which the individual data are fit. Dimension weighting models, in con-

Figure 4.7: Exploratory spaces for subject 14.

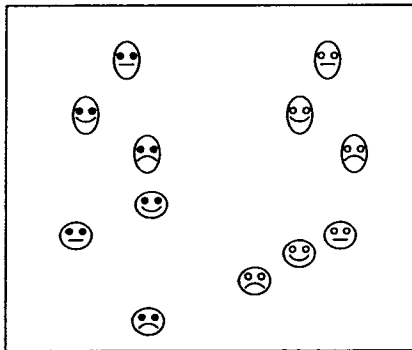
bimonscalqa.m (VAF=.890)



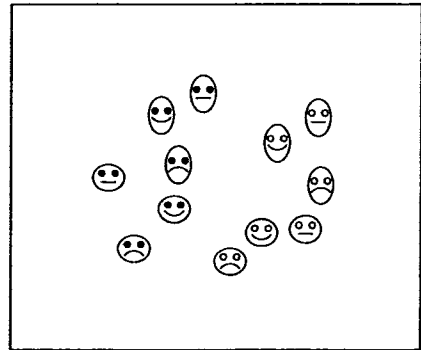
SPSS PROXSCAL (NRS=.049, DAF=.951)



biscalqa.m (VAF=.664)



SYSTAT (S1=.256, RSQ=.523)

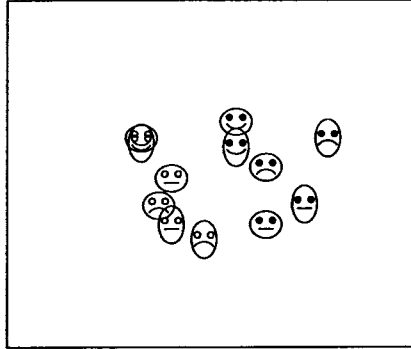


trast, choose a different route: instead of using the data twice (i.e., once for constructing the reference structure, then for fitting the individual data that beforehand have already been used to establish the average representation), an 'all-in-one' approach is done, simultaneously constructing the reference or group representation and modelling individual variation through estimating dimension weights.

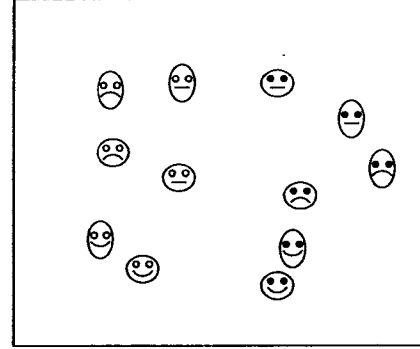
Independent of methodological preferences, both approaches rest on the assumption of the existence of a common space, albeit an abstraction, most faithfully representing the communalities of structural properties among the data at individual source level. Tucker and Messick (1963) intended their points of view (POV) model as a compromise for preserving the concept of individual differences scaling when the common space assumption simply is no longer ten-

Figure 4.8: Exploratory spaces for subject 2.

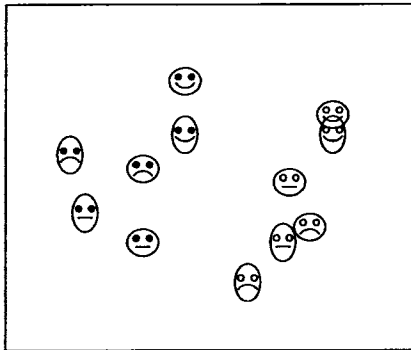
bimonscalqa.m (VAF=.852)



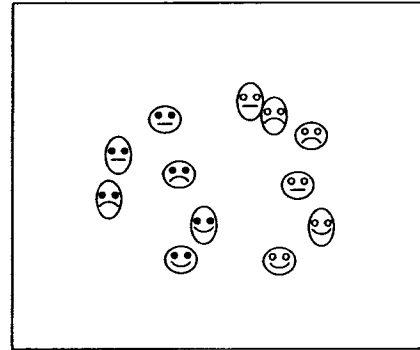
SPSS PROXSCAL (NRS=.047, DAF=.953)



biscalqa.m (VAF=.626)



SYSTAT (S1=.251, RSQ=.546)



able because the involved data sets are too heterogeneous. Briefly, POV tries to identify homogeneous groupings of sources in treating the vectorized individual proximity matrices as variables that are subjected to a principal component analysis. The resulting linear combinations of the individual proximity data – the different points of view – are analyzed by separate MDS. The procedure introduced by Tucker and Messick (1963) has been refined by Meulman and Verboon (1993) through combining the estimation of source weights and coordinates for the points of view representations into an iterative alternating least squares algorithm utilizing majorization. An alternative approach to detect homogeneous subgroups of sources in the sample was proposed by Winsberg and De Soete (1993) by augmenting the weighted Euclidean model by a latent class component such that for each class of subjects, different sets of dimension

weights can be estimated. Most recently, Brusco and Cradit (2005) have presented a *non-metric method* for identifying clusters of sources with equivalently ordered proximity matrices. Specifically, for each pair of sources a concordance measure is computed by counting the agreements and disagreements of dissimilarity orderings among object triples, based on which a partition of sources into homogeneous subgroups is constructed using a branch-and-bound algorithm (see Brusco & Stahl, 2005a). Subsequently, the individual proximity matrices of sources within each cluster are pooled and subjected to independent MDS. As the Brusco-Cradit ConPar (CONcordance PARTitioning) procedure focuses on the ordinal structure in the data, it promises a particularly viable complement to the combinatorial individual differences scaling approach as developed in this paper.

5 Unfolding

Unfolding has become a collective term for a broad range of methods for analyzing any rectangular two-way two-mode proximity matrix, \mathbf{Q} , including distance-based metric and nonmetric approaches devised within a multidimensional scaling (MDS) framework. So far and unfortunately, distance-based unfolding models, cannot necessarily be regarded as an overwhelming success due to the unresolved issue of degeneracy. However, an efficient distance-based unfolding technique would be a highly desirable data analytic technique, because the inter-set dissimilarities of row and column objects, q_{rc} , could be represented in a joint geometrical space, with an immediate and mathematically legitimate distance interpretation. Alternative methods such as correspondence analysis generate row and column object representations in separate spaces. To obtain a joint representation, the two configurations are then rescaled and superimposed. Thus, notwithstanding their looks and wide-spread misunderstanding, joint correspondence analysis plots do not represent the relationships between objects from different sets as distances, and should not be interpreted as such.

Recently, to reduce the risk of a degenerate solution, Busing, Groenen, and Heiser (2005; see also Busing, 2006) have introduced a penalty function algorithm for fitting distance-based unfolding models (implemented as PREFSCAL in SPSS CATEGORIES since version 14). PREFSCAL can also perform three-way unfolding, based on the weighted Euclidean model. Hubert and his collaborators reformulate unfolding as a genuine combinatorial data analytic problem, and develop an unfolding algorithm using discrete optimization, without relying on (sub)gradients nor penalty functions for minimizing the loss function (see Hubert et al., 2006). We will demonstrate that combinatorial unfolding provides an alternative computational strategy for avoiding degenerate solutions. Remarkably, Hubert et al. (2006) do not mention at all the potential of combinatorial unfolding to successfully remedy degeneracy. In addition, we propose an extension of the combinatorial unfolding algorithm that also accommodates three-way data.

The next section provides a review of the theory and methods of unfolding, including the combinatorial approach. The second section covers degeneracy and reports the results from comparing the performance of PREFSCAL and combinatorial unfolding. In using the identical data sets employed by Busing et al. (2005) for illustrating the effectiveness of their approach we show that combinatorial unfolding obtains non-degenerate representations and, in fact, almost

identical solutions as presented by Busing et al. (2005). Three-way unfolding models are presented in the third section. The exemplary application to the contraceptive data from Weller and Romney (1990) illustrates combinatorial three-way unfolding. We conclude with a discussion and suggestions for further study.

5.1 Theory and Methods of Unfolding

The typical data set of an unfolding problem can be characterized as a rectangular two-way two-mode proximity matrix, $\mathbf{Q} = \{q_{rc}\}$, of $N_R \times N_C$, with $r = 1, \dots, N_R$ and $c = 1, \dots, N_C$, where rows and columns represent two distinct sets of entities, $\mathcal{O}_R = \{O_{1R}, \dots, O_{N_R R}\}$ and $\mathcal{O}_C = \{O_{1C}, \dots, O_{N_C C}\}$, containing N_R and N_C elements, respectively. In M -dimensional (representational) space, let x_{rm} denote the coordinates of objects in the row set \mathcal{O}_R , with $m = 1, \dots, M$, while y_{cm} stands for the coordinates of objects in the column set \mathcal{O}_C ; \mathbf{X} and \mathbf{Y} represent coordinate matrices for objects in \mathcal{O}_R and \mathcal{O}_C , respectively, of size $N_R \times M$ and $N_C \times M$.

In its original conception by Coombs (1950, 1964), the unfolding model was intended as a geometric representation for ordinal preference and choice data, collected into a rectangular data matrix, with rows representing subjects and columns stimuli. Unfolding attempts to jointly arrange the elements of both sets along a continuum so the coordinates of locations induce interpoint distances reflecting the ordinal preference information as closely as possible. For an individual's object preference representation, Coombs coined the term "I-scale": the location of his/her (imaginary) ideal object coincides with his/her point on the line; manifest stimuli are positioned at increasing distances from the subject point according to the provided preference ranking. The computational challenge posed by the unfolding model is the construction of the so-called "J-scale", a simultaneous mapping of all I-scales onto the line minimizing the discrepancies between the collective subjects-stimuli arrangements and the original rank order data. Hays and Bennett (Hays & Bennett, 1960; Bennett & Hays, 1961) prepared the theoretical grounds for generalizing the unfolding model to a multidimensional setting, the multidimensional unfolding model.

To date, a plethora of models has been devised to solve the task of multidimensional unfolding including probabilistic models (see De Soete & Carroll, 1992; MacKay & Zinnes, 1995; Marden, 1995; Marley, 1992; Van Blokland-Vogeleang, 1989, 1993; Zinnes & Griggs, 1974), external unfolding (Carroll, 1972, 1980; Davison, 1983, Ch. 7; Meulman, Heiser, & Carroll, 1986), models within the dual/optimal scaling framework, in particular homogeneity analysis/multiple correspondence analysis (see Gifi, 1990; Greenacre, 1984; Heiser, 1981, Ch. 5; Nishisato, 1980), and adaptations of existing techniques for two-way one-mode data to suit the rectangular two-way two-mode format of, for example, preference judgments, namely scalar product (or projection) models

based on principal component analysis, and distance-based approaches rooted in the tradition of multidimensional scaling, with the distinction between metric and nonmetric models (see Borg & Groenen, 1997, Chs. 14–15, 2005, Chs. 14–16; Carroll & Arabie, 1980, 1998; Cox & Cox, 2001, Ch. 8; De Leeuw & Heiser, 1982; Krzanowski, 1988, Chs. 4.3–4.4). The ideal point model of unfolding can be considered the prototypical distance model, while Tucker’s vector model of unfolding (Tucker, 1960) probably represents the most famous instance of scalar product/projection models. Our main concern here are MDS-related unfolding models, but we consider scalar product unfolding strategies when it benefits our presentation.

5.1.1 Scalar Product versus Distance Models of Unfolding

Scalar Product Models. The basic notion of a scalar product model for multidimensional unfolding involves approximating the observed dissimilarities in \mathbf{Q} through scalar products of the respective coordinate vectors. Sometimes, these models are referred to as projection models “because the scalar products of a set of points with a fixed point are proportional to the projections of these points onto a vector from the origin of the coordinate system to the fixed point” (Carroll & Arabie, 1980, p. 621). If \mathbf{Q} , for example, contains subject-object preferences, with rows representing subjects and columns objects, then the q_{rc} are usually reconstructed by projecting the object vectors, \mathbf{y}_c , onto the subject vector, \mathbf{x}_r , normalized to unit length:

$$\begin{aligned} q_{rc} &= \sum_m x_{rm} y_{cm} \\ &= \mathbf{x}'_r \mathbf{y}_c \\ \Rightarrow \mathbf{Q} &= \mathbf{X} \mathbf{Y}'. \end{aligned}$$

The actual coordinate vectors, collected into matrices \mathbf{X} and \mathbf{Y} , are obtained through a singular value decomposition (SVD) of the original input data matrix \mathbf{Q} :

$$\begin{aligned} \mathbf{Q} &= \mathbf{U} \mathbf{\Lambda} \mathbf{V}' \\ \Rightarrow q_{rc} &= \sum_m u_{rm} v_{cm} \lambda_m, \end{aligned}$$

where M equals the rank of \mathbf{Q} . We obtain $\mathbf{X} = \mathbf{U}$, with unit-length restrictions imposed on (subject) vectors, $\mathbf{x}_1, \dots, \mathbf{x}_{N_R}$, such that $\text{diag}(\mathbf{X} \mathbf{X}') = \text{diag}(\mathbf{I})$; $\mathbf{Y} = \mathbf{V} \mathbf{\Lambda}$.

Distance Models. Distance models of unfolding aim at translating the dissimilarities, q_{rc} , directly into distances in a geometrical representation of the underlying configuration of objects in psychological space. A general definition

of distances between row and column points r and c of \mathbf{Q} is given by

$$d_{rc}(\mathbf{X}, \mathbf{Y}) = \left(\sum_m |x_{rm} - y_{cm}|^\lambda \right)^{\frac{1}{\lambda}};$$

the choice of $\lambda = 1$ yields the city-block, and $\lambda = 2$ the Euclidean distance, with the latter as the traditional and most often used option in unfolding analysis. The set coordinates are retrieved from the distances best fitting the observed proximities, q_{rc} , operationalized through minimizing a given loss function. Common practice suggests to transform the dissimilarities, q_{rc} , into pseudo-distances, denoted \hat{d}_{rc} . If the dissimilarities are presumed to possess interval scale properties, “the function relating the data to distances is generally assumed to be inhomogeneously linear — that is, linear with an additive constant as well as a slope coefficient” (Carroll & Arabie, 1998, p. 183) — in other words, the observed proximities are subjected to a linear transformation

$$f(q_{rc}) = \hat{d}_{rc} = a + bq_{rc};$$

typically, with $b = 1$; the corresponding distance model is called metric. “In the case of ordinal data, the functional relationship is generally assumed to be monotonic” (Carroll & Arabie, 1998, p. 183), and the respective unfolding models are referred to as nonmetric, where the transformation $f(\cdot)$ has only to satisfy the monotonicity condition

$$q_{rc} \leq q_{r'c'} \Rightarrow f(q_{rc}) = \hat{d}_{rc} \leq f(q_{r'c'}) = \hat{d}_{r'c'}.$$

A clarification of a technical detail concerning the difference between metric distance and scalar product unfolding seems in order here. Like Torgerson’s (1958) classic metric MDS model, most metric unfolding models require for computational convenience the squared dissimilarities to be transformed through double centering into a scalar product matrix, $\mathbf{B} = \{b_{rc}\}$, with

$$f(q_{rc}^2) = b_{rc} = -\frac{1}{2}(q_{rc}^2 - q_{r\cdot}^2 - q_{\cdot c}^2 + q_{\cdot\cdot}^2),$$

where

$$\begin{aligned} q_{\cdot c}^2 &= \frac{1}{N_R} \sum_r q_{rc}^2 \\ q_{r\cdot}^2 &= \frac{1}{N_C} \sum_c q_{rc}^2 \\ q_{\cdot\cdot}^2 &= \frac{1}{N_R N_C} \sum_c q_{rc}^2. \end{aligned}$$

Subsequently, \mathbf{B} is subjected to an SVD to retrieve the coordinates:

$$\begin{aligned}\mathbf{B} &= \mathbf{X}\mathbf{Y}' \\ \Rightarrow \mathbf{B} &= \mathbf{U}\mathbf{\Lambda}\mathbf{V}' \\ \Rightarrow \mathbf{X} &= \mathbf{U}\mathbf{\Lambda}^{\frac{1}{2}} \\ \Rightarrow \mathbf{Y} &= \mathbf{V}\mathbf{\Lambda}^{\frac{1}{2}}\end{aligned}$$

We need to re-emphasize that only employing a linear transformation of the proximities, $\hat{d}_{rc} = a + bq_{rc}$, qualifies a model as metric. The transformation of the q_{rc} to scalar products, is motivated solely by computational ease, and is not a characteristic feature of the metric approach. Other computational procedures, without the scalar product transformation and through gradient-based optimization have been suggested to fit the metric model (see Lingoes & Roskam, 1973). Although, the distinction between scalar product and metric distance unfolding models may be subtle, from a technical point of view it is not: scalar product models do not incorporate a transformation of the input proximities to, well, scalar products (although, preprocessing of the data is required, e.g., transforming the subject vectors to unit length). In addition, the geometric configuration derived from scalar product models does not lend itself to an immediate interpretation in terms of distances between subjects and objects (because SVD generates row and column object representations in separate spaces). Rather, any inspection of between sets relations should be solely restricted to the ordinal information extracted from the projection of the object onto the subject vectors. In contrast, metric distance models, while often employing the same mathematical device of an SVD, reconstruct between set proximities as genuine (Euclidean) distances that may be directly interpreted as such.

The definition of a distance model accommodates conceptualizing unfolding as (metric or nonmetric) MDS of the between sets proximities q_{rc} by the analogy of constructing a square-symmetric supermatrix $\mathbf{P}^{(\mathbf{Q})}$, with the off-diagonal blocks containing the between-sets proximities of \mathbf{Q} , and within-sets proximities in the submatrices along the main diagonal of $\mathbf{P}^{(\mathbf{Q})}$ missing. Ross and Cliff (1964), Schönemann (1970), Gold (1973), and Greenacre and Browne (1986), develop solutions for the metric unfolding of \mathbf{Q} ; Kruskal and Carroll extend nonmetric MDS to nonmetric unfolding by adapting STRESS-1, the loss function typically minimized in nonmetric MDS (see Kruskal, 1964a, 1964b, 1977; Kruskal & Carroll, 1969; Kruskal & Wish, 1978), for fitting an unfolding model:

$$\sigma_1(\hat{\mathbf{D}}, \mathbf{X}, \mathbf{Y}) = \left(\frac{\sum_r \sum_c w_{rc} (\hat{d}_{rc} - d_{rc})^2}{\sum_r \sum_c w_{rc} d_{rc}^2} \right)^{\frac{1}{2}},$$

with d_{rc} denoting Euclidean distances, the matrix $\hat{\mathbf{D}} = \{\hat{d}_{rc}\}$ indicating a mono-

tone transformation of the original input proximities q_{rc} , and fixed weights w_{rc} of value 0 if q_{rc} is missing, and 1 otherwise.

In the case of unfolding, however, STRESS-1 was soon discovered to be extremely vulnerable to degenerate solutions. An unfolding solution is called degenerate if it satisfies a given optimality criterion perfectly, while at the same time resulting in a substantively trivial, non-informative spatial representation (we cover degeneracy in detail in the second section). As a remedy, Kruskal and Carroll (1969) propose to use a modified loss function, called STRESS-2:

$$\sigma_2(\hat{\mathbf{D}}, \mathbf{X}, \mathbf{Y}) = \left(\frac{\sum_r \sum_c w_{rc} (\hat{d}_{rc} - d_{rc})^2}{\sum_r \sum_c w_{rc} (d_{rc} - \bar{d})^2} \right)^{\frac{1}{2}},$$

with

$$\bar{d} = \frac{1}{N_R N_C} \sum_r \sum_c d_{rc}.$$

Unfortunately, STRESS-2 is also prone to degeneracies (for detailed discussions of various degeneracy scenarios for unfolding, see Borg & Bergermaier, 1982; Borg & Groenen, 1997, 2005; Busing et al., 2005; De Leeuw, 1983; Heiser, 1981, 1989b; Kruskal & Carroll, 1969).

5.1.2 Combinatorial Unfolding

Combinatorial unfolding is based on the generalization of combinatorial unidimensional scaling developed by Hubert, Arabie, and Meulman (Hubert & Arabie, 1986, 1988; Hubert et al., 2002, 2006 — see also Brusco, 2002b) that is presented in detail in Chapter 4. Recall the conceptualization of unfolding as an MDS problem: the rectangular proximity matrix, \mathbf{Q} , is embedded in a 2×2 square-symmetric supermatrix $\mathbf{P}^{(\mathbf{Q})} = \{p_{ij}^{(\mathbf{Q})}\}$, resulting in a matrix of size $N \times N$, with $N = N_R + N_C$ (thus, merging the previous separate indices $1, \dots, N_R$ and $1, \dots, N_C$ into one set, with the first N_R row objects subscripted as $1, \dots, N_R$ and the remaining N_C column objects assigned indices $N_R + 1, \dots, N_R + N_C = N$). The off-diagonal blocks of $\mathbf{P}^{(\mathbf{Q})}$ contain the between-sets proximities of \mathbf{Q} , whereas the missing within-sets proximities between row and column objects in the main diagonal blocks are set to zero:

$$\mathbf{P}^{(\mathbf{Q})} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{Q} \\ \hline \mathbf{Q}' & \mathbf{0} \end{array} \right].$$

Combinatorial unfolding is performed by fitting a single or multiple (city-block) unidimensional structure(s) to $\mathbf{P}^{(\mathbf{Q})}$ yielding coordinates of the joint optimal arrangement of row and column objects of \mathbf{Q} , along the dimension(s) of the geometric space.

The least-squares loss function for the unidimensional combinatorial unfold-

ing problem is given by

$$L(\mathbf{x}, c) = \sum_{i < j} w_{ij} (p_{ij}^{(\mathbf{Q})} - (|x_j - x_i| + c))^2,$$

where $w_{ij} = 0$ if both i and j are row or column objects, and equal to 1 otherwise. Of course, as in the MDS case, given a particular choice of transformation of the $p_{ij}^{(\mathbf{Q})}$, metric or nonmetric combinatorial unfolding can be performed. Multidimensional combinatorial unfolding is performed through fitting multiple unidimensional (city-block) structures through successive residualizations of $\mathbf{P}^{(\mathbf{Q})}$.

The search for the optimal permutation ρ^* of $\mathbf{P}^{(\mathbf{Q})}$ is carried out through QA. In the context of fitting a single unidimensional structure, we let as before, $\rho(\cdot) \equiv \rho$ denote a feasible permutation of the first N integers. A brief comment for clarifying a technical detail of the QA-induced row and column reordering of $\mathbf{P}^{(\mathbf{Q})}$ might be helpful: were we to apply the QA exchange/interchange algorithm to $\mathbf{P}^{(\mathbf{Q})}$ in its primary block structure, the reordering process would render $\mathbf{P}^{(\mathbf{Q})}$ in its initial partitioning, with only the partial nonzero rows and columns within \mathbf{Q} and \mathbf{Q}' permuted because the entries in the zero blocks on the main diagonal of $\mathbf{P}^{(\mathbf{Q})}$ will not be picked up by the reordering mechanism as they cannot contribute to any increase of the QA-criterion. As a consequence, the initial order of rows and columns of $\mathbf{P}^{(\mathbf{Q})}$ will be completely preserved, keeping the two sets strictly apart. Greater flexibility, however, could be gained were the QA search allowing for mixing the orders of row and column sets. Subtracting the grand mean, $\bar{q} = [1/(N_R N_C)] \sum_r \sum_c q_{rc}$, from the entries in \mathbf{Q} and \mathbf{Q}' of $\mathbf{P}^{(\mathbf{Q})}$ will turn the zero blocks into meaningful entities, thereby inducing the desired mixing of row and column objects. Whether mean-deviation is the best strategy awaits further research; to date the effect of certain transformations combined with specific patterning of $\mathbf{P}^{(\mathbf{Q})}$ on the reordering process has not been studied in detail (see Hubert & Arabie, 1995).

Additive constant and least squares optimal coordinates are estimated through Dykstra's IP algorithm. Due to the complex structure of $\mathbf{P}^{(\mathbf{Q})}$, however, the linear constraints deduced from a specific permutation ρ of row and column objects are more involved than in the MDS case and might be best explained through a brief schematic representation of how these constraints are generated. Let O_{rR} , $O_{r'R}$ and O_{cC} , $O_{c'C}$ denote two arbitrary row and column objects; d_{rc} , $d_{rc'}$, $d_{r'c}$, and $d_{r'c'}$ in the 2×2 matrix below represent distances to be fit to the four proximity values observed for O_{rR} , $O_{r'R}$, O_{cC} , and $O_{c'C}$:

	O_{cC}	$O_{c'C}$	
O_{rR}	d_{rc}	$d_{rc'}$	
$O_{r'R}$	$d_{r'c}$	$d_{r'c'}$	

Detailed inspection of all possible permutations of O_{rR} , $O_{r'R}$, O_{cC} and $O_{c'C}$

reveals that the following set represents an exhaustive collection of all possible order constraints the entities d_{rc} , $d_{rc'}$, $d_{r'c}$, and $d_{r'c'}$ have to satisfy (see Hubert et al., 2006):

1. $O_{rR} \prec O_{r'R} \prec O_{cC} \prec O_{c'C}$ implies $d_{rc} + d_{r'c'} = d_{rc'} + d_{r'c}$;
2. $O_{rR} \prec O_{cC} \prec O_{r'R} \prec O_{c'C}$ implies $d_{rc} + d_{r'c} + d_{r'c'} = d_{rc'}$;
3. $O_{rR} \prec O_{cC} \prec O_{c'C} \prec O_{r'R}$ implies $d_{rc} + d_{r'c} = d_{rc'} + d_{r'c'}$;
4. $O_{rR} \prec O_{r'R} \prec O_{cC}$ implies $d_{r'c} \leq d_{rc}$;
5. $O_{rR} \prec O_{cC} \prec O_{c'C}$ implies $d_{rc} \leq d_{rc'}$.

5.2 Degeneracy

Recall the definition of a degenerate unfolding solution as satisfying a given optimality criterion perfectly, while at the same time resulting in a substantively trivial, non-informative spatial representation. We also already mentioned that Kruskal's (see Kruskal, 1964a, 1964b, 1977; Kruskal & Carroll, 1969; Kruskal & Wish, 1978) STRESS-1 loss function proved extremely susceptible to degeneracy when applied to an unfolding problem:

$$\sigma_1(\hat{\mathbf{D}}, \mathbf{X}, \mathbf{Y}) = \left(\frac{\sum_r \sum_c w_{rc} (\hat{d}_{rc} - d_{rc})^2}{\sum_r \sum_c w_{rc} d_{rc}^2} \right)^{\frac{1}{2}}.$$

For example, when identical within set coordinates are chosen, $\sigma_1(\hat{\mathbf{D}}, \mathbf{X}, \mathbf{Y})$ attains zero, with objects of both sets lumped together into two separate points. In technical terms, the possible inclusion of an intercept in the monotone transformation $f(\cdot)$ allows for all \hat{d}_{rc} to take on equal values, which in turn will lead to identical within set coordinates: let $f \mid f(q_{rc}) = \hat{d}_{rc} = a + bq_{rc}$ denote an admissible monotone transformation, with the legitimate choice $b = 0$ for the slope, yielding $\hat{d}_{rc} = a = \bar{q}$, defined by

$$\bar{q} = \frac{1}{N_R N_C} \sum_r \sum_c q_{rc}.$$

5.2.1 PREFSCAL: Avoiding Degeneracy by Penalizing the Loss Function

PREFSCAL by Busing et al. (2005; Busing, 2006) uses a majorization algorithm for minimizing the least-squares loss function augmented by a penalty term to maintain a sufficient level of variability among the transformed proximities, or pseudo-distances, which will prevent the distance estimates and associated object locations from collapsing into a single/few positions. Specifically, PREFSCAL (Busing, 2003, 2006; Heiser, 2004; Busing et al., 2005) employs a loss function modelled after Kruskal's STRESS, but explicitly penalizing on

decreasing variability of the transformed proximities \hat{d}_{rc} to avoid degenerate solutions:

$$\sigma_p^2(\hat{\mathbf{D}}, \mathbf{X}, \mathbf{Y}) = \left(\frac{\sum_r \sum_c w_{rc} (\hat{d}_{rc} - d_{rc})^2}{\sum_r \sum_c w_{rc} \hat{d}_{rc}^2} \right)^\lambda \left(1 + \frac{\omega}{v^2(\hat{\mathbf{d}})} \right),$$

with λ and ω representing penalty parameters, and $v(\cdot)$ the Pearson coefficient of variation, defined by

$$v(\hat{\mathbf{d}}) = \frac{\text{standard deviation}(\hat{\mathbf{d}})}{\text{mean}(\hat{\mathbf{d}})} = \left(\frac{\frac{1}{N_R N_C} \sum_r \sum_c \hat{d}_{rc}^2}{\left(\frac{1}{N_R N_C} \sum_r \sum_c \hat{d}_{rc} \right)^2} - 1 \right)^{\frac{1}{2}}.$$

Thus, a large coefficient of variation ensures the penalty term approaching one, while it grows to infinity the less variability occurs among the \hat{d}_{rc} . PREFSCAL uses a subgradient approach for minimizing the loss function through an iterative majorization algorithm, combined with alternating least squares estimation: starting with an arbitrary choice of d_{rc} , a monotone transformation of the given proximities is sought, $f(q_{rc}) = \hat{d}_{rc}$, best approximating the d_{rc} , followed by updating d_{rc} such that they optimally fit the previously identified \hat{d}_{rc} , and so on, until the estimates converge (see Busing, 2003; Heiser, 1981; Borg & Groenen, 1997, 2005). Monte Carlo results presented by Busing et al. (2005) recommend penalized STRESS as a promising candidate for effectively avoiding degeneracies in unfolding.

5.2.2 Evaluation: PREFSCAL versus Combinatorial Unfolding

For a comparative assessment of performance, combinatorial unfolding models are fit to the same data sets that Busing et al. (2005) use to demonstrate the effectiveness of the PREFSCAL algorithm: the brewery data (Borg & Bergmaier, 1982) and the breakfast data (Green & Rao, 1972). Both have been studied extensively (see also Borg & Groenen, 1997, 2005) as template examples of data sets considered to be particularly susceptible to degenerate solutions. In addition, Busing et al. (2005) conduct a Monte Carlo simulation for studying various choices of the penalty parameters λ and ω in their effect on avoiding degeneracy for error-free and error-perturbed data. The design of the study is inspired by previous work of Coombs and Kao (1960) and Kruskal and Carroll (1969), and employs synthesized random proximity matrices of a 30×15 rectangular format, generated from the coordinates of 30 row and 15 column objects in two dimensions drawn from a standard normal distribution. As combinatorial unfolding does not incorporate any penalty parameters, we simply use the original random data from Coombs and Kao (1960).

Metric and non-metric combinatorial unfolding of all data sets are performed by two MATLAB routines, `biscaltmac.m` and `bimonscaltmac.m`, re-

Table 5.1: Summary of unfolding models fit to data matrices generated from the original Coombs and Kao (1960) random coordinates of 30 row and 15 column objects.

Perturbation	Distances	Unfolding Model	Graph
Unperturbed	City-Block	Metric	Figure 5.1
		Nonmetric	
	Euclidean	Metric	
		Nonmetric	
25% Error-Perturbed	City-Block	Metric	Figure 5.2
		Nonmetric	
	Euclidean	Metric	
		Nonmetric	

spectively (including calls of subroutines for QA and IP); `biscaltmac.m` and `bimonscaltmac.m` conveniently allow multiple random permutations to initialize the input proximity matrix. For each data matrix, 1000 random starts are used, yielding a frequency distribution of the associated VAF scores, from which the one with highest value is selected as the (at least) locally optimal unfolding solution, subsequently represented as a geometric configuration displaying the structural relationship between row and column objects.

Synthetic Data (Coombs & Kao, 1960)

Based on the first two column vectors from the original Coombs-Kao 45×3 standard-normal random coordinate matrix, two rectangular 30×15 proximity matrices are computed, one containing Euclidean, the other city-block distances between the 30 row and 15 column objects. Following Busing et al. (2005), two variants for both matrices are computed, one unperturbed, the other with 25% random perturbation according to

$$d_{ij} \text{ (perturbed)} = (d_{ij}) \exp\{(.25)e_{ij}\},$$

with $e_{ij} \sim \mathcal{N}(0, 1)$, resulting in a total of four proximity matrices. The chosen procedure for generating random error perturbations is equivalent to adding log-normal error, thereby ensuring that no negative perturbed distances occur. Table 5.1 summarizes the input distance matrices and combinatorial unfolding algorithms used.

The graphical representations of the unfolding solutions with largest VAF scores (ranging from .94–.98 for unperturbed and .75–.84 for error-perturbed matrices) are shown in Figure 5.1 (unperturbed) and Figure 5.2 (error-perturbed); the original configuration based on the first two vectors of the Coombs-Kao (1960) standard normal coordinates is presented at the top of both graphs. In-

specting Figures 5.1 and 5.2, we observe for all four data matrices exceptionally good recoveries of the original structure. Most remarkable, the choice of Euclidean or city-block distance seems to have only a negligible impact on recovery. Note that for ease of legibility only the column objects of the data matrices are labelled from 1 to 15.

Brewery Data (Borg & Bergermaier, 1982)

Borg and Bergermaier (1982) collected a rectangular 26×9 proximity matrix, containing ratings of nine breweries on 26 brand image attributes. The original data represent similarities and, hence, were first transformed into dissimilarities. The graphs of the metric and non-metric unfolding solutions with highest VAF score (.86 and .90, respectively) are given in Figure 5.3. To avoid clutter, only the nine columns representing the breweries are labelled from 1 to 9. Apart from breweries 9 and 5 switching positions, the structural representation obtained through combinatorial unfolding is almost identical to the PREFSCAL configuration presented by Busing et al. (2005, p. 86, bottom panel; with a VAF value of .91).

Breakfast Data (Green & Rao, 1972)

The data were collected from 42 subjects, Wharton School MBA students and their wives, who, among other tasks had to rank 15 breakfast food items according to overall preference and preference within the context of certain menus and serving occasions. Following Busing et al. (2005), we restrict our analysis to the first 42×15 data matrix consisting of overall preference rankings. The graphs of the spaces constructed through `biscaltmac.m` and `bimonscaltmac.m` (VAF scores equal to .57 and .68, respectively) are presented in Figure 5.4; only the breakfast items are labelled. The combinatorial unfolding representations are almost indistinguishable from the one obtained through PREFSCAL by Busing et al. (2005, p. 90, bottom panel; with VAF = .81 — notice that Busing and his collaborators modelled the data as ‘row-conditional’, employing separate transformations of the proximities in each of the N_R row entries). Clearly, the first dimension represents a ‘toast’ factor, discriminating between toast/hard rolls and muffin-/pastry-type breakfast items. The second dimension can be interpreted as a ‘butter’ factor, arranging toast or muffins garnished with butter or margarine at the bottom as opposed to sugary, marmalade- or jelly-related items at the top.

5.3 Extensions of the Unfolding Model

Various attempts have been made to stretch the bounds of the unfolding paradigm. For greater flexibility in incorporating individual differences, consider Carroll's (metric) weighted Euclidean unfolding model (see Carroll, 1972, p. 118):

$$d_{rc}(\mathbf{X}, \mathbf{Y}, \mathbf{W}) = \left(\sum_m w_{rm} (x_{rm} - y_{cm})^2 \right)^{\frac{1}{2}};$$

\mathbf{W} denotes an $N_R \times M$ matrix of weights, w_{rm} , that account for differential 'salience' or 'importance' of the m^{th} dimension for object r (typically, a subject within a subjects-rate-objects context), through shrinking or extending dimension m of the geometric representation.

In his model for individual differences scaling, Heiser (1989a) retains the concept of differential weighting of dimensions as a means for constructing the individualized psychological spaces, but the distances between objects are defined within a city-block metric. A straightforward extension of Heiser's model to a weighted city-block unfolding model would be given by

$$d_{rc}(\mathbf{X}, \mathbf{Y}, \mathbf{W}) = \sum_m w_{rm} |x_{rm} - y_{cm}|.$$

One should note that the weighted Euclidean unfolding model of Carroll basically alters the original idea of a joint display of all subjects and objects in a single graph. Rather, similar to the individual differences scaling paradigm in MDS, separate private spaces for each subject would have to be constructed displaying the objects only, with axes of the geometric space adjusted to the respective weights as applied by a particular individual. An extension of the weighted unfolding model, called the IDIOSCAL model (see Carroll, 1972, 1980; Carroll & Arabie, 1998; Carroll & Wish, 1974), allows for differential axes rotation of the representational space (as Euclidean distances are rotation invariant).

The second generalization of unfolding models is targeted at analyzing truly three-mode three-way data, with the third way representing different data sources, such as situations, scenarios, experimental conditions, or time points. These data can be characterized as a cube, with multiples of the rectangular data matrix \mathbf{Q} stacked as slices along the third dimension. As before, by introducing the source index $s = 1, \dots, S$, we derive as notation for the entire data cube $\mathbf{Q}^{(S)} = \{q_{rcs}\}$, with the layers of individual sources denoted by $\mathbf{Q}_s = \{q_{rc(s)}\}$. Scalar product and distance models have been developed to analyze $\mathbf{Q}^{(S)}$. The former, like *CANDECOMP* (Carroll & Chang, 1970), *PARAFAC* (see Harshman, 1970; Harshman & Lundy, 1984, 1994), or the Tucker models (Kiers, 1991; Kroonenberg, 1983, 1988, 1994; Kroonenberg & De Leeuw, 1980; Ten Berge, De Leeuw, & Kroonenberg, 1987; Tucker, 1966, 1972), are basically generalizations of Tucker's (1960) original vector model for two-mode preference data. The simplest generalization of the two-way vector (scalar product) model to a three-way

data cube $\mathbf{Q}^{(S)}$, is given by the following decomposition

$$\begin{aligned}
q_{rc(s)} &= \sum_m x_{rm(s)} y_{cm(s)} \\
&= \mathbf{x}'_{rs} \mathbf{y}_{cs} \\
\Rightarrow \mathbf{Q}_s &= \mathbf{X}_s \mathbf{Y}'_s \\
&= \mathbf{U} \mathbf{\Lambda}_s \mathbf{V}' \\
\Rightarrow q_{rc(s)} &= \sum_m u_{rm} v_{cm} \lambda_{ms} \\
\Rightarrow \mathbf{Q}^{(S)} &= \sum_s \mathbf{U} \mathbf{\Lambda}_s \mathbf{V}',
\end{aligned}$$

the CANDECOMP-PARAFAC model, independently discovered by Carroll and Chang (1970), and Harshman (1970; Harshman & Lundy, 1984, 1994). For a comprehensive and state of the art presentation of multi-way analysis as a generalization of principal component analysis see Smilde, Bro, and Geladi (2004).

Three-way distance models, to our knowledge, were first presented by Carroll (1980; DeSarbo & Carroll, 1981, 1985). For example, a straightforward extension of the weighted Euclidean model, including an additive constant for source s , was proposed by DeSarbo and Carroll (DeSarbo & Carroll, 1981, 1985):

$$d_{rc(s)}^2 + a_s = \sum_m w_{ms} (x_{rm} - y_{cm})^2 + a_s.$$

Stimulus points y_{cm} and subject ideal points x_{rm} are supposed to be fixed across sources; the latter affect the weighting of dimensions and the location of points differentially, as expressed through the w_{ms} and a_s .

All these models are metric and defined in terms of squared Euclidean distances and represent straightforward extensions of the weighted Euclidean model in (5.3), also used for three-way nonmetric unfolding in PREFSCAL: following the familiar logic of individual differences scaling, the private spaces representing the individual sources are derived from the group space through differential re-scaling of its coordinate axes. As an instructive PREFSCAL application to three-way data, Heiser and Busing (2004) report a re-analysis of the famous breakfast data, originally published by Green and Rao (1972).

Briefly, among other tasks, 42 subjects ranked 15 breakfast food items according to six preference (i.e., consumption) scenarios. The data form a three-way three-mode data cube, with subjects, breakfast items, and preference scenarios representing the three modes. Following the familiar logic of individual differences scaling, a common or group space was constructed based on the data aggregated across all six consumption scenarios (=sources). In a second step, the individual scenarios were fitted against the representation of the common space, allowing only for shrinking or stretching the dimensions of the common space to accommodate for unique features of the individual structures observed with each consumption scenario. The goal of the three-way unfolding analy-

sis was to demonstrate how priming respondents with different consumption scenarios causes a shift in the preference structure of the breakfast items.

5.3.1 Three-way Combinatorial Unfolding

Combinatorial unfolding for three-way data rests on analyzing individual variability within a deviation-from-the-mean framework: based on the individual proximity matrices aggregated across sources, \mathbf{Q}_a , a best-fitting M -dimensional ‘average’ unfolding representation is generated, to serve as frame of reference, against which the individual data matrices \mathbf{Q}_s are fit in a confirmatory manner. The object orderings obtained for $\mathbf{P}_a^{(\mathbf{Q})}$ serve as frame of reference for constructing individual (‘private’) spaces. The coordinates of objects in the private space are estimated using IP, with constraints defined by the identified object orderings of the reference representation. For the individual configurations, the spacings between objects are free to vary such that the loss function attains a minimum, provided the order of objects from the reference structure be preserved. The VAF criterion computed for each source serves as a fit index quantifying how closely the data reflect the properties of the reference space (for an M -dimensional solution):

$$\text{VAF}_s = 1 - \frac{\sum_{i < j} w_{ij(s)} (p_{ij(s)}^{(\mathbf{Q})} - \sum_m (|x_{jm(s)} - x_{im(s)}| + c_{m(s)}))^2}{\sum_{i < j} w_{ij(s)} (p_{ij(s)}^{(\mathbf{Q})})^2};$$

recall that the grand mean of $\mathbf{P}^{(\mathbf{Q})}$ equals zero.

5.3.2 Application: The Contraceptive Data from Weller and Romney (1990)

Matrix *TOTAL* was subjected to bidimensional metric and nonmetric unfolding, employing two stand-alone MATLAB routines `biscaltmac.m` and `bimonscaltmac.m`, conveniently allowing for multiple runs with initial random permutations of the input proximity matrix. For each unfolding model, 1000 random starts were used; the metric and nonmetric solutions with the largest VAF values (`biscaltmac.m`: .927, `bimonscaltmac.m`: .980) were chosen as reference structures, against which the *FEMALE* and *MALE* matrices were fit by confirmatory unfolding. Both input matrices were normalized by the sum of squares of the total data matrix prior to the confirmatory analysis to put all three matrices on the same scale, thereby adjusting for each of the gender split matrices representing only half of the total number of respondents. The graphs of the *TOTAL* reference as well as the individual (confirmatory) *FEMALE* and *MALE* spaces are presented in Figures 5.5 and 5.6, respectively. The metric and nonmetric unfolding representations are essentially identical: Clearly, the first dimension of the *TOTAL* reference spaces orders the fifteen contraceptive methods along a left-to-right continuum, with opposing poles of ‘behavioral’

versus 'surgical' methods. In borrowing terminology from Hubert et al. (2001a, p. 110), the segments of this continuum, starting from the left, can be characterized as "lottery" (Rhy, Wit), "liquid" (Spe, Foa, Dou), "latex" (Con, Dia), "alternative (or no) sexual behavior" (Ora, Abs), "medical/female related" (Pil, Iud), "medical/out-patient surgical" (Tub, Vas), and "medical/surgical" (Hys, Abo). The location of the criteria points S, A, C, and E span a scale ranging from (S)afety (in the sense of non-invasive)/immediate (A)vailability to (E)ffectiveness (through invasive surgical measures), with (C)onvenience approximately defining the midpoint. The second dimension appears to introduce a distinction among behavioral methods by contrasting the "liquid" (Spe, Foa, Dou) with the "latex" (Con, Dia) and "alternative (or no) sexual behavior" (Ora, Abs) segments. The position of the "lottery" methods (Rhy, Wit) implies that the ordinate captures a (secondary) evaluation, integrating effectiveness and convenience, of the behavioral contraceptive measures that do not demand medical consultation ("home-grown"; see Hubert et al., 2001a, p. 110). In comparing the (confirmatory) unfolding representations for females and males, the differential spread of points along the axis suggests that both sexes differ markedly in their perception of the contraceptive methods. In particular, the segment of behavioral methods receives a finer grained evaluation by females than males.

To further explore the differences between females and males, their data matrices were also analyzed as independent sources through (exploratory) two-way metric and nonmetric unfolding with 1000 random starts each. The solutions with highest VAF scores were chosen (biscaltmac.m: .892 and .878, bimonscaltmac.m .957 and .947 for *FEMALE* and *MALE*, respectively). The graphs obtained (see Figures 5.7 and 5.8) confirm the distinction into behavioral versus medical/surgical methods, with the latter considered to be most effective. The previous conclusion of the two sexes profoundly disagreeing in their evaluations of the behavioral contraceptive methods is further substantiated. Specifically, whereas females consider Ora and Abs as most convenient and effective, men choose Iud.

As an alternative to the previous independent (exploratory) analysis of the diverging perspectives of women and men, a confirmatory fitting of the *MALE* data to *FEMALE* was conducted utilizing the latter as reference decomposition (with VAF scores for males equal to .672 and .850). Given the observations from the exploratory analysis, the comparatively low VAF scores for males come

as no surprise. The clutter of points representing the behavioral methods particularly indicates mediocre fit, and can be regarded as an immediate consequence of the specific way confirmatory spaces are constructed in combinatorial scaling: Object spacings are allowed to vary freely as long as the imposed order from the reference space is not violated, which in the case of disagreement as with males and females, will result in points being shifted closer and closer together. Figures 5.9 and 5.10 present the graphs of the female reference and male confirmatory spaces. The configurations are in line with the results of the previous analyses: females display greater sophistication in their judgments of the behavioral methods as opposed to males who tend to perceive most of them as almost interchangeable; both groups agree in their evaluation of the (invasive) surgical measures. The nonmetric confirmatory plot for males provides an illustration of the possible need to limit the iterative fitting of an optimal monotone transformation (see Hubert et al., 1998a, p. 568).

5.4 Discussion and Conclusion

The findings of the analyses of the brewery and breakfast data raise the immediate question of why two approaches mathematically so very different like PREFSCAL and combinatorial unfolding, can yield almost identical results. Although, in general we cannot often easily draw a direct formal comparison between a continuous and a discrete optimization algorithm, we claim that PREFSCAL and combinatorial unfolding essentially attempt to optimize the same objective. Recall that any degenerate unfolding solution is first and foremost characterized by equal within-sets distance estimates for the elements in \mathcal{O}_R and \mathcal{O}_C , implying that row and column objects in the extreme collapse into two strictly distinct locations — or more to the point: no mixing occurs between the elements of those two sets. Both algorithms attempt to remedy exactly that particular feature of degenerate solutions. By inducing a thorough mixing of row and column objects, the estimation process is prevented from degenerating into a configuration strictly separating the two sets. In PREFSCAL this goal is accomplished in an indirect way by penalizing the loss function for decreasing variability among the transformed proximities; in combinatorial unfolding this is done directly by mean-deviating the entries in $\mathbf{P}^{(\mathbf{Q})}$, enforcing a mixing of row and column objects during the QA maximization of $\Gamma(\rho)$.

The proposed combinatorial unfolding algorithm essentially fits multiple uni-dimensional city-block structures to the input proximity matrix \mathbf{Q} and one might immediately object that the city-block metric was merely chosen for the computational convenience of its additivity property. Indeed, so far, a combinatorial approach to mathematically accommodating a Euclidean model directly does

not exist (and may never be implementable as the nonlinear nature of the Euclidean metric poses severe computational obstacles). However, we would like to point out that the task of fitting a Euclidean structure, say with $M = 2$, requires

$$q_{rc} \approx d_{rc} = [(x_{r1} - y_{c1})^2 + (x_{r2} - y_{c2})^2]^{\frac{1}{2}},$$

which can be rewritten as

$$q_{rc}^2 \approx d_{rc}^2 = (x_{r1} - y_{c1})^2 + (x_{r2} - y_{c2})^2 = |x_{r1} - y_{c1}|^2 + |x_{r2} - y_{c2}|^2,$$

implying that we can approximate the squared entries in \mathbf{Q} by the sum of two squared unidimensional unfolding structures. Thus, the combinatorial framework can also indirectly account for a Euclidean model. The striking similarity between the unfolding models fit to the Euclidean and city-block distance matrices from the Coombs and Kao random coordinates may serve as anecdotal evidence in support of our conjecture.

Nonmetric models have proven to be extremely vulnerable to degenerate solutions because the monotonicity condition inherently facilitates a fitting process with all coordinates ultimately converging. A certain amount of shrinkage in the coordinates was detected with the nonmetric model for the brewery data and the confirmatory fitting of the male contraceptive data, but not to a degree that the resulting solution meets the degeneracy condition. Still, depending on the specific properties of a particular data set, we should consider the possible need to limit the iterative fitting of an optimal monotone transformation: “If carried through to convergence, a perfect representation may be obtained, but only at the expense of losing almost all [structure] contained in the original proximity matrix” (Hubert, Arabie, & Meulman, 1998a, p. 568) – in other words, risking a degenerate solution.

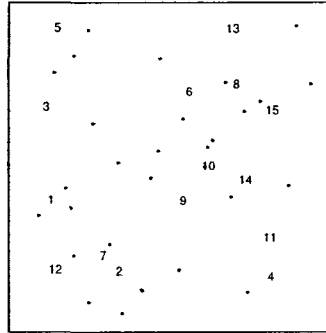
From a more practical data analytic point of view, we might also observe that the Euclidean metric does not appear to be equally well-suited for all types of stimuli (see Arabie, 1991). Empirical evidence suggests that the Euclidean metric tends to underperform in approximating the distance structure of analyzable stimuli in psychological space in comparison to the city-block metric. Following Attneave’s (1950) phenomenological distinction, analyzable stimuli are postulated to be composed of perceptually separable dimensions; for example, forms varying in size and orientation. In contrast, integral stimuli, supposedly better accommodated by the Euclidean metric, represent a holistic mix of constituting dimensions that at an observer’s perceptual level are assumed to be extremely difficult to decompose into well-defined, separate contributions; colors varying in lightness and saturation may serve as the standard example (see Arabie, 1991; Garner, 1974; Shepard, 1991). Within this context, a city-block unfolding model attains a unique status because widely available commercial software currently does not offer any viable implementation of city-block unfolding (SYSTAT offers the possibility to carry out two-way unfolding within the city-block metric, but

uses a deficient algorithm extremely susceptible to degeneracy).

As an obvious conclusion, if one is interested in using unfolding models (metric or non-metric), then the combinatorial MATLAB routines `biscaltmac.m` and `bimonscaltmac.m` would be natural choices to consider. Particularly, if one wishes to avoid tedious and time consuming experimentation to determine the settings of PREFSCAL penalty parameters optimal for a specific data set. In fact, we observed many instances, where the default choices produce degenerate solutions. The combinatorial search strategy for solving metric and nonmetric unfolding elegantly avoids the difficulties of gradient-based approaches by utilizing a mathematically different type of optimization scheme.

Figure 5.1: Metric and non-metric combinatorial unfolding analysis of Euclidean and city-block distance matrices computed from the Coombs-Kao (1960) standard-normal random coordinates — unperturbed.

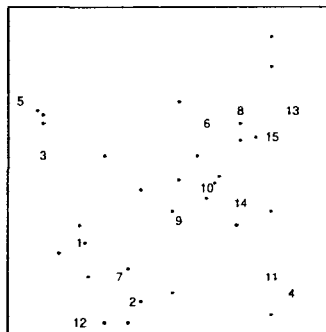
Original Random Coordinates



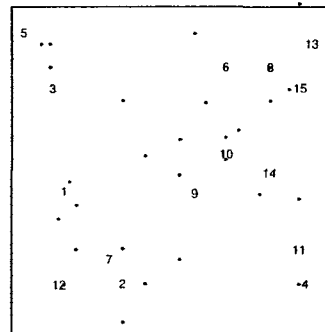
Euclidean

City-Block

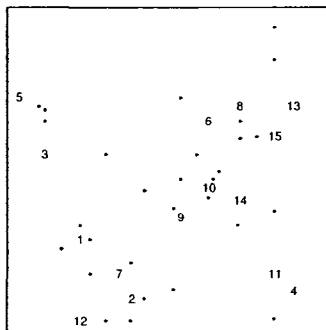
biscaltmac.m (VAF = .94)



biscaltmac.m (VAF = .97)



bimonscaltmac.m (VAF = .95)



bimonscaltmac.m (VAF = .98)

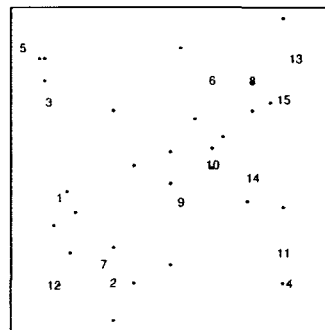
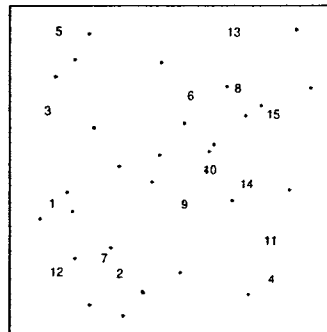


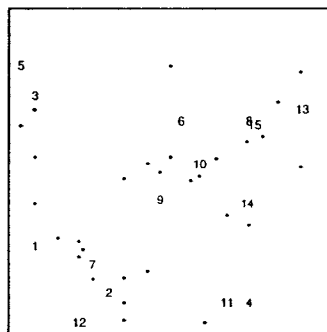
Figure 5.2: Metric and non-metric combinatorial unfolding analysis of Euclidean and city-block distance matrices computed from the Coombs-Kao (1960) standard-normal random coordinates — 25% error-perturbed.

Original Random Coordinates



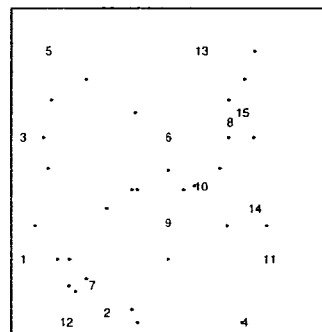
Euclidean

biscaltmac.m (VAF = .75)

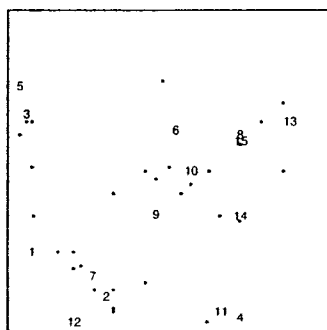


City-Block

biscaltmac.m (VAF = .79)



bimonscaltmac.m (VAF = .81)



bimonscaltmac.m (VAF = .84)

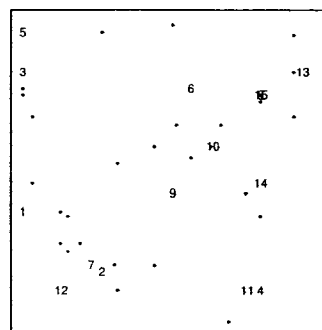


Figure 5.3: Metric and non-metric combinatorial unfolding analysis of the Brewery Data (Borg & Bergermaier, 1982).

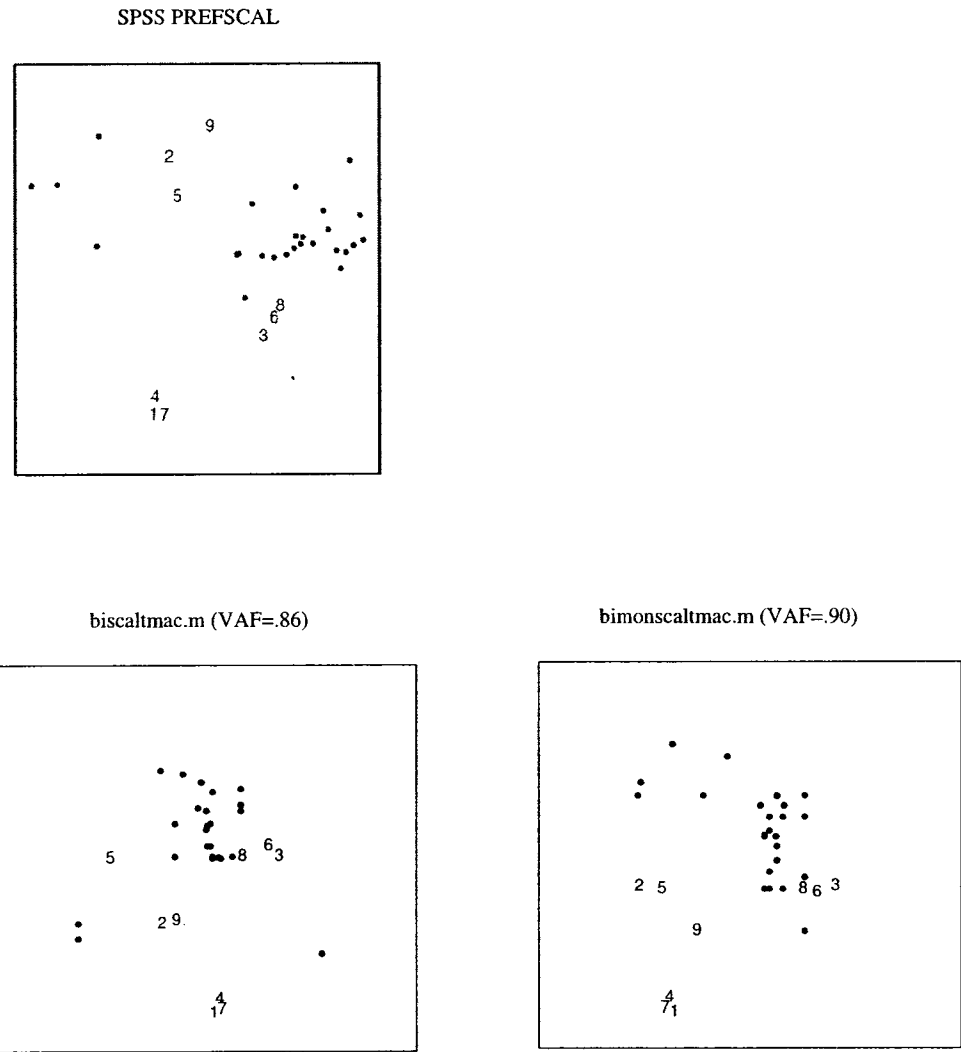
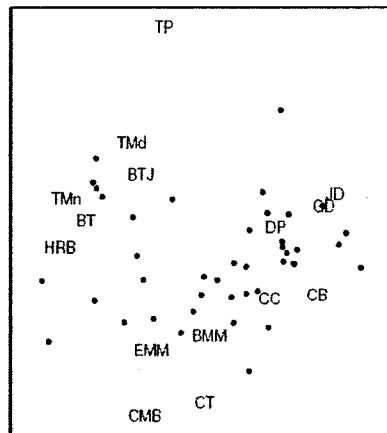
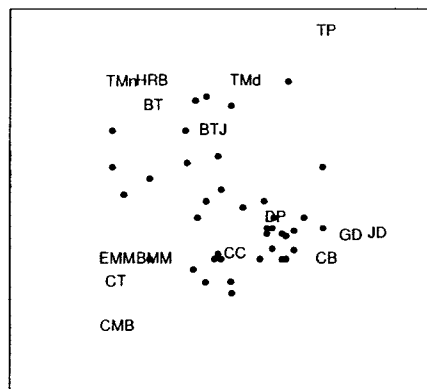


Figure 5.4: Metric and non-metric combinatorial unfolding analysis of the Breakfast Data (Green & Rao, 1972): Toast Pop-up (TP), Buttered Toast (BT), English Muffin and Margarine (EMM), Jelly Donut (JD), Cinnamon Toast (CT), Blueberry Muffin and Margarine (BMM), Hard Rolls and Butter (HRB), Toast and Marmalade (TMd), Buttered Toast and Jelly (BTj), Toast and Margarine (TMn), Cinnamon Bun (CB), Danish Pastry (DP), Glazed Donut (GD), Coffee Cake (CC), Corn Muffin and Butter (CMB).

SPSS PREFSCAL



biscaltmac.m (VAF=.57)



bimonscaltmac.m (VAF=.68)

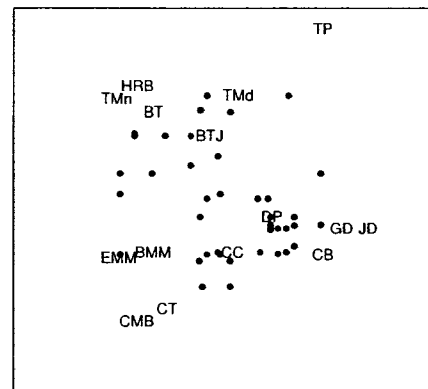
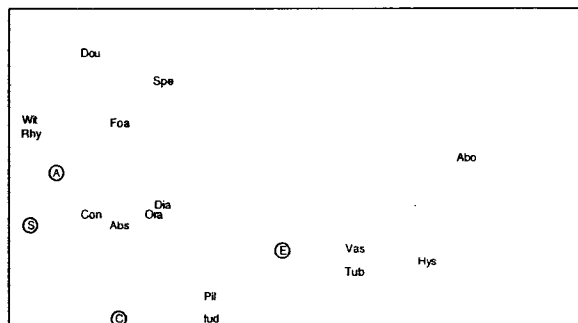
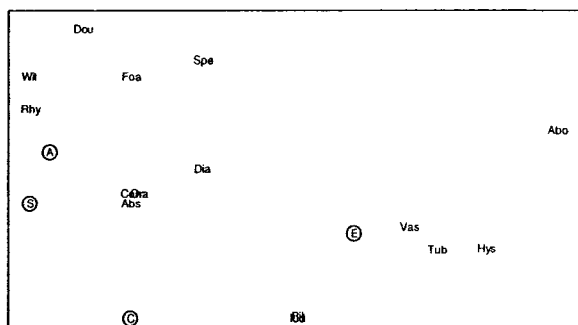


Figure 5.5: Metric Unfolding: reference space based on matrix *TOTAL*, and corresponding confirmatory spaces of matrices *FEMALE* and *MALE*.

biscaltmac.m TOTAL SAMPLE (VAF=.927)



biscaltmac.m FEMALES vs. TOTAL SAMPLE (VAF=.866)



biscaltmac.m MALES vs. TOTAL SAMPLE (VAF=.794)

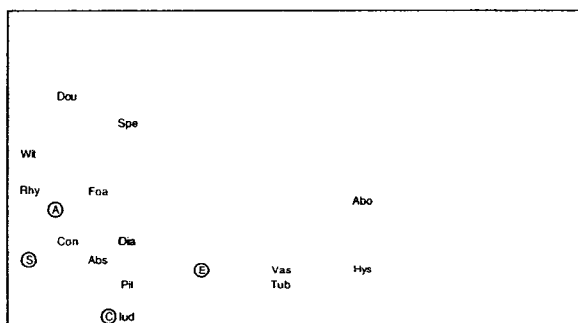
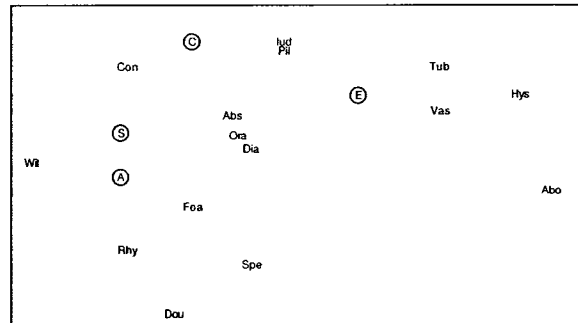
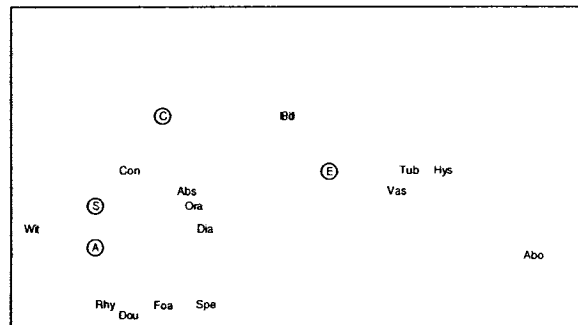


Figure 5.6: Nonmetric Unfolding: reference space based on matrix *TOTAL*, and corresponding confirmatory spaces of matrices *FEMALE* and *MALE*.

bimonscaltmac.m TOTAL SAMPLE (VAF=.980)



bimonscaltmac.m FEMALES vs. TOTAL SAMPLE (VAF=.939)



bimonscaltmac.m MALES vs. TOTAL SAMPLE (VAF=.935)

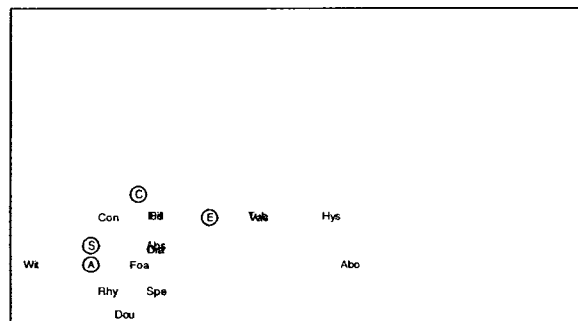


Figure 5.7: Metric Unfolding: exploratory spaces of matrices *FEMALE* and *MALE*.

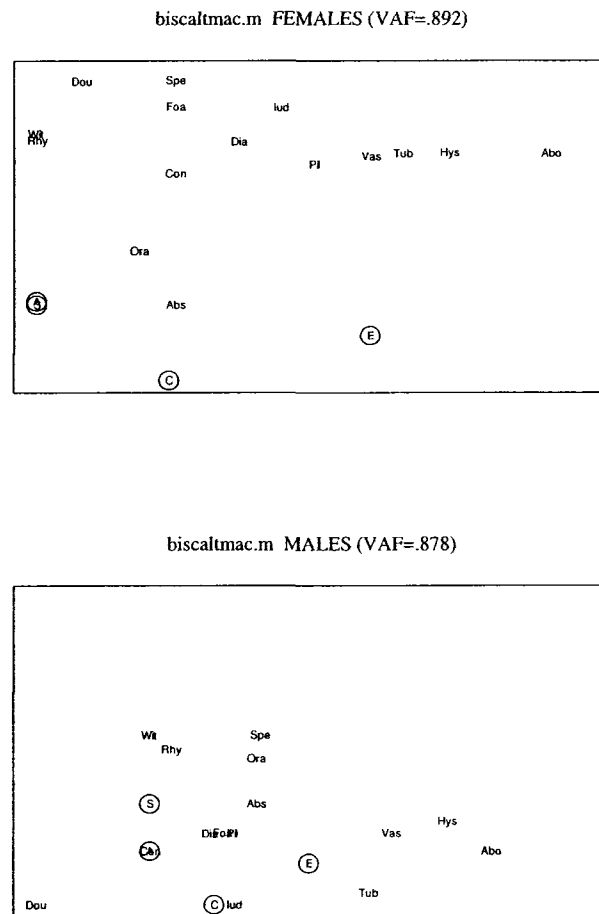


Figure 5.8: Nonmetric Unfolding: exploratory spaces of matrices *FEMALE* and *MALE*.

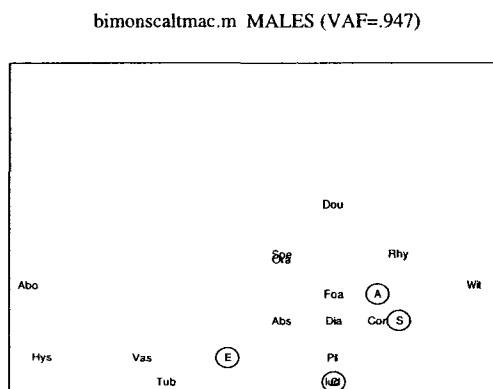
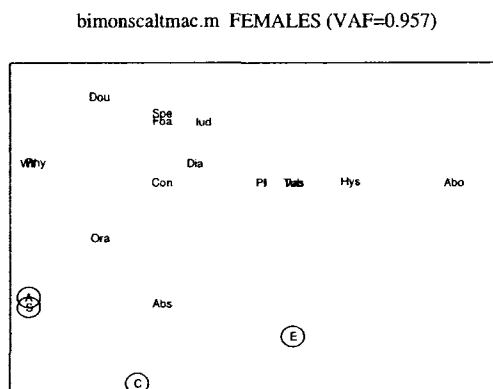


Figure 5.9: Metric Unfolding: reference space based on matrix *FEMALE*, and corresponding confirmatory space of matrix *MALE*.

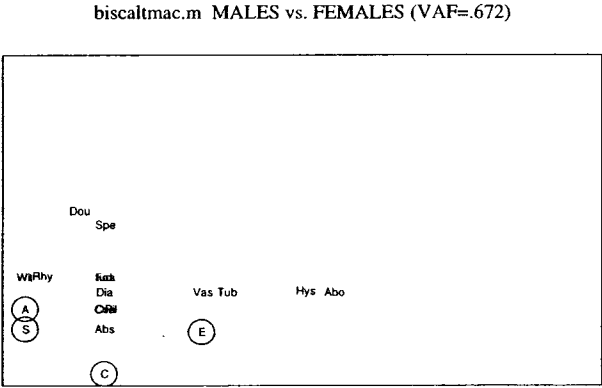
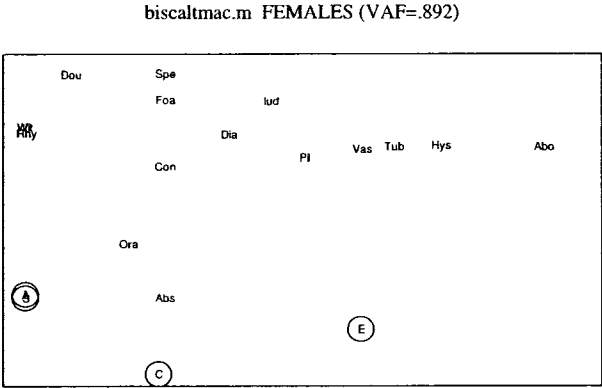
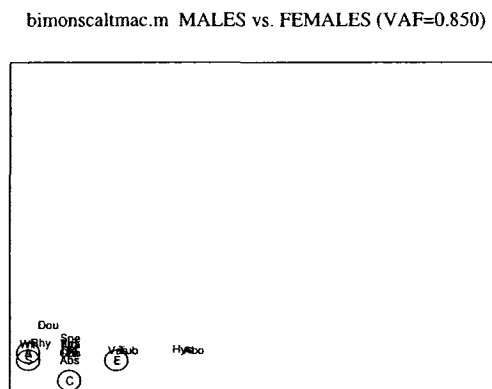
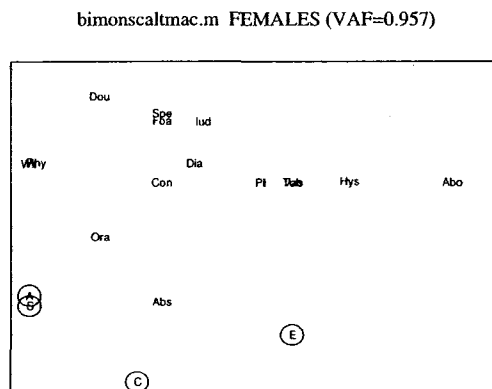


Figure 5.10: Nonmetric Unfolding: reference space based on matrix *FEMALE*, and corresponding confirmatory space of matrix *MALE*.



6 Order-Constrained Matrix Decomposition

Spectral decomposition of a square-symmetric matrix into a sum of identically-sized matrices is immediately deducible from eigenvalue factorization. The rank reduction theorem in linear algebra states that the original matrix can be approximated by the sum of a small number of components, referred to as the low-rank approximation. In applied statistics, eigenvalue decomposition, and its generalization to rectangular matrices, singular value decomposition, (EVD, SVD, respectively) represent the major computational devices for multivariate techniques such as principal component and factor analysis (see Johnson & Wichern, 2007), correspondence analysis (see Greenacre, 1984), or in a wider context, the construction of biplots (see Gower & Hand, 1996) and optimal scaling (see Gifi, 1990, Nishisato, 1980).

Spectral decomposition of a matrix ensures additivity through metric constraints on the components. Alternative non-metric additive matrix decompositions have been developed. For example, Hubert and Arabie (1994; see also Hubert et al., 1998b, 2006) propose decomposing a square-symmetric proximity matrix into a sum of matrices of identical size, with components only constrained to display a specific order pattern: either the Robinson form for proximity matrices containing similarities, or the anti-Robinson form for dissimilarity matrices. A similarity (dissimilarity) matrix is said to be in Robinson (anti-Robinson) form — in honoring the person who first introduced the concept (Robinson, 1951) — if for each row or column the cell entries never increase (decrease) when moving away from the main diagonal. Similar to the generalization of spectral decomposition to an SVD-based representation of rectangular matrices, order-constrained (anti-)Robinson decomposition for square-symmetric matrices extends to rectangular two-way two-mode proximity matrices, the order-constrained (anti-)Q decomposition, such that the given matrix is represented exhaustively by a sum of equally-sized matrices restricted to having (anti-)Q form (Hubert & Arabie, 1995a; see also Hubert et al., 2006). The Q form of a rectangular similarity matrix defines a specific order pattern for its rows and columns, characterized by entries nondecreasing up to a maximum, and from there never increasing; conversely, a rectangular dissimilarity matrix is said to be in anti-Q form if the row and column entries are nonincreasing to a minimum and nondecreasing thereafter (see Kendall, 1971a, 1971b). Equivalent to metric decomposition, the ultimate objective of order-constrained matrix decomposition strategies is to identify an ‘order-constrained low-rank’ reconstruction of

the original proximity matrix, providing a parsimonious representation of its total variability by the sum of a small number of order-constrained matrix components.

Due to their specific patterning, the extracted components — either in (anti-)Robinson or in (anti-)Q form — lend themselves immediately to displays of the relationships between their row and column objects through a continuous unidimensional scale, or a discrete non-spatial configuration in the form of an ultrametric tree diagram. Under substantive considerations, the two structures address distinct perspectives on how objects are mentally represented. Formally, both models involve the same number of estimates and are fit through least-squares to the exact same data base. Thus, a direct comparison of their differential fit is legitimate to decide which model provides the most faithful representation of the relationships among row and column objects of each component matrix. In other words, we can determine whether the amount of variability associated with each component is better approximated by a discrete or a continuous structure.

Within this context, order-constrained matrix decomposition attains the status of a combinatorial data analytic meta-technique, when multiple discrete (non-spatial) or continuous (spatial) structures are fit to a given proximity matrix to increase the information extracted about the relationships among row and column objects vis-à-vis a representation featuring only a single structure (see, for example, Carroll, 1976; Carroll & Pruzansky, 1980; Carroll, Clark, & DeSarbo, 1984; Hubert & Arabie, 1995b; Hubert et al., 2006). In such instances, we are frequently confronted with inconclusive evidence as to which representation to favor, whether the proximity matrix is more appropriately represented by multiple continuous or multiple discrete structures. To illustrate, assume that for a given proximity matrix two (continuous) unidimensional scales provide superior overall fit as compared to a representation by two (discrete) ultrametric tree structures. At the same time, the first ultrametric tree structure attains substantially higher fit than the first unidimensional scale. We can only directly compare those two first structures. The second structures refer to different data bases — so, which model should we ultimately choose? We propose to resolve this ambiguity through order-constrained matrix decomposition that readily allows to determine which structure provides a superior representation of the extracted matrix components and the associated amount of total variability in the data — a feature not available when fitting multiple structures to a proximity matrix without order-constrained decomposition.

We develop a generalization of order-constrained matrix decomposition to three-way data for studying individual differences that provides the analyst with an instrument to explore complex hypotheses concerning the appropriateness of continuous or discrete stimuli representations from an interindividual as well as an intraindividual perspective. We advocate an approach guided by the principle of analyzing individual variability within a deviation-from-the-mean frame-

work. First, the individual proximity matrices are aggregated across sources into a single average matrix that is then fit by a small number of order-constrained components (say, two or three). The identified decomposition serves as a frame of reference against which the individual proximity matrices are decomposed in a confirmatory manner, informing us how well the various individuals are represented by the reference decomposition. In addition, for each individual source, the obtained (confirmatory) order-constrained components themselves can be fit by unidimensional and ultrametric structures. Thus, we obtain (1) an interindividual assessment to what extent the various sources conform to the structural reference representation, and (2) an intraindividual comparison whether the distinct (confirmatory) order-constrained matrix components are better represented in terms of a (continuous) unidimensional or (discrete) ultrametric model, with possible implications about the underlying individual cognitive representations of the stimuli.

The second section introduces order-constrained anti-Robinson matrix decomposition and its generalization to accommodate three-way two-mode data; as an illustrative application, the Face Data are analyzed in terms of structural individual differences in judgments of the schematic face stimuli. The extension of order-constrained anti-Q matrix decomposition to three-way three-mode data is developed in the third section, including the exemplary analysis of the Weller-Romney (1990) contraceptive data. We conclude with a discussion and summary.

As we are concerned with dissimilarity matrices, in the following, without loss of generality, we consider solely the anti-Robinson and anti-Q decomposition of proximity matrices, and their representation through sums of matrices of the according pattern. To simplify notation, and the subsequent presentation, we denote matrices having anti-Robinson as well as anti-Q form by \mathbf{Z} ; the interpretation of $\mathbf{Z} = \{z_{ij}\}$, with $1 \leq i, j \leq N$, as a (square-symmetric) two-way one-mode matrix, or $\mathbf{Z} = \{z_{rc}\}$, with $r = 1, \dots, N_R$ and $c = 1, \dots, N_C$, as a (rectangular) two-way, two-mode matrix will be apparent from the respective context.

6.1 Order-Constrained Anti-Robinson Matrix Decomposition

6.1.1 Definitions and Formal Concepts

A square-symmetric matrix $\mathbf{Z} = \{z_{ij}\}$ is said to have anti-Robinson (AR) form if

$$z_{ij} \leq z_{i(j+1)} \quad \text{for } 1 \leq i < j \leq N-1, \quad (6.1)$$

$$\begin{aligned} z_{ij} &\leq z_{i(j-1)} \quad \text{for } 2 \leq j < i \leq N \\ \Leftrightarrow z_{ij} &\leq z_{(i-1)j} \quad \text{for } 2 \leq i < j \leq N. \end{aligned}$$

In words, moving away from the main diagonal of \mathbf{Z} , within each row or column, the entries never decrease.

A decomposition of \mathbf{P} is sought as the sum of matrices $\mathbf{Z}_1 + \dots + \mathbf{Z}_k + \dots + \mathbf{Z}_K$ such that the least-squares loss function is minimized, under the constraint of all \mathbf{Z}_k having AR form (see Hubert & Arabie, 1994):

$$\begin{aligned} \min_{\mathbf{Z}_1, \dots, \mathbf{Z}_K} \{L(\mathbf{Z}_1, \dots, \mathbf{Z}_K)\} &= \min_{\mathbf{Z}_1, \dots, \mathbf{Z}_K} \left\{ \sum_i \sum_j \left(p_{ij} - \sum_k z_{(k)ij} \right)^2 \right\} \\ &= \min_{\mathbf{Z}_1, \dots, \mathbf{Z}_K} \left\{ \text{tr} \left(\mathbf{P} - \sum_k \mathbf{Z}_k \right) \left(\mathbf{P} - \sum_k \mathbf{Z}_k \right)' \right\} \\ &= \min_{\mathbf{Z}_1, \dots, \mathbf{Z}_K} \left\{ \left[\mathbf{p} - (\mathbf{I}, \dots, \mathbf{I}) \begin{pmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_K \end{pmatrix} \right] \left[\mathbf{p} - (\mathbf{I}, \dots, \mathbf{I}) \begin{pmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_K \end{pmatrix} \right]' \right\}, \end{aligned}$$

subject to

$$\mathbf{G}_k \mathbf{z}_k \geq \mathbf{0} \quad \text{for } 1 \leq k \leq K;$$

where tr denotes the trace function and K is at most equal to N ; \mathbf{p} and \mathbf{z}_k stand for vectorizations of \mathbf{P} and \mathbf{Z}_k of size $V \times 1$, with $V = N(N-1)$, such that the main diagonal entries have been dropped; \mathbf{I} indicates a $V \times V$ identity matrix; \mathbf{G}_k is a design matrix of $W \times V$, with $W = (N-1)(N-2)$, representing the AR inequality constraints as defined in (6.1), given N objects. A brief comment concerning the determination of W might be helpful: the AR condition imposes for each row and column of \mathbf{Z}_k an at least monotonically increasing pattern of consecutive cell entries in the off-diagonal sections. As \mathbf{Z}_k is symmetric and the main diagonal entries are zero, it suffices to inspect only its upper (or lower) triangular part containing $(1/2)N(N-1)$ elements. The monotonicity condition reduces the valid, non-redundant constraints to the number of pairs of adjacent rows (columns) such that in each of the $N-1$ rows (columns) we are

losing one element (by excluding comparisons with the zero entries along the main diagonal), resulting in $(1/2)N(N-1) - (N-1)$ constraints imposed on consecutive row (column) elements in the upper (lower) triangular matrix of \mathbf{Z}_k , which can be rewritten as $(1/2)(N-1)(N-2)$ and reduces to $W = (N-1)(N-2)$ when multiplied by 2 in taking into account rows and columns together.

The task of identifying an additive decomposition of \mathbf{P} is addressed by fitting $\sum_k \mathbf{Z}_k$ through successive residualizations of \mathbf{P} : \mathbf{Z}_1 is fit to \mathbf{P} , \mathbf{Z}_2 to the residual matrix $\mathbf{P} - \mathbf{Z}_1$, \mathbf{Z}_3 to the residual matrix $(\mathbf{P} - \mathbf{Z}_1) - \mathbf{Z}_2$, and so on. However, the given proximity data matrix \mathbf{P} initially does not have AR form (otherwise, a decomposition attempt were futile), posing a substantial technical impediment to the identification of a least-squares optimal AR matrix \mathbf{Z}_1 . Therefore, the rows and columns of \mathbf{P} need first to be re-arranged into a form that matches the desired AR pattern as closely as possible through identifying a suitable permutation $\rho_1(\cdot) \equiv \rho_1$, where $\rho(\cdot)$ denotes a permutation of the first N integers. Similarly, the estimation of each subsequent AR component, $\mathbf{Z}_2, \dots, \mathbf{Z}_K$, requires the preceding search for a permutation ρ_2, \dots, ρ_K , producing a rearrangement of rows and columns of the associated residual matrices, optimally approximating the AR form. Hence, conceptually, generating an optimal AR decomposition of \mathbf{P} into $\sum_k \mathbf{Z}_k$ represents a two-fold least-squares minimization problem, separating into the distinct operations of finding a collection of permutations, ρ_1, \dots, ρ_K , for attempting to reorder the matrices into AR form, followed by an estimation step to actually numerically identify the desired AR components, $\mathbf{Z}_1, \dots, \mathbf{Z}_K$, conforming to the order constraints as established through the respective permutations.

6.1.2 Optimal AR Decomposition: Algorithmic and Computational Details

The search for an optimal collection of permutations, ρ_1, \dots, ρ_K , represents an NP-hard combinatorial optimization problem, addressable through a QA heuristic (see Hubert & Arabie, 1994; Hubert et al., 2006), where the target matrix, \mathbf{B} , is defined as $b_{ij} := |j - i|$, with $1 \leq i < j \leq N$, which provides a perfect and equally spaced AR structure. Estimation of the matrix components in perfect AR form, $\mathbf{Z}_1, \dots, \mathbf{Z}_K$, is carried out through IP.

Recall that, given the actual numerical entries in \mathbf{P} , ρ^* indicates the best possible reordering of the N rows and columns of \mathbf{P} into a form matching an AR pattern. Our goal is to estimate an $N \times N$ least-squares approximation $\mathbf{Z} = \{z_{ij}\}$ to \mathbf{P}_{ρ^*} , which has perfect AR form. In other words, the arrangement of objects in \mathbf{Z} is determined by ρ^* , while the numerical estimates z_{ij} have to satisfy (6.1). Given N objects, (6.1) expands into the collection of $W = (N-1)(N-2)$ sets of order constraints, $\mathcal{C}_1, \dots, \mathcal{C}_W$, each representing a single inequality of the form $z_{i(j+1)} - z_{ij} \geq 0$ or $z_{(i-1)j} - z_{ij} \geq 0$. Formally, we need to find a perfect AR matrix $\mathbf{Z}^* \equiv \mathbf{z}^* \in \mathcal{C} = \bigcap_{w=1}^W \mathcal{C}_w$ minimizing the least-squares loss function

$(\mathbf{p}_{\rho^*} - \mathbf{z})'(\mathbf{p}_{\rho^*} - \mathbf{z})$, with \mathbf{p}_{ρ^*} and \mathbf{z} denoting vectorizations of matrices \mathbf{P}_{ρ^*} and \mathbf{Q} . After initializing $\mathbf{z}^{(0)} := \mathbf{z}_{\rho^*}$, the IP algorithm proceeds by checking for each adjacent pair of row or column objects whether the involved proximities conform to the respective constraints in \mathcal{C}_w . If at iteration $t-1$ a violation is encountered, the vector $\mathbf{z}^{(t-1)}$ is projected onto \mathcal{C}_w , and the particular proximities are replaced by their projections (see Dykstra, 1983); for example, if we observe $z_{i(j+1)}^{(t-1)} - z_{ij}^{(t-1)} < 0$ the projections are given by:

$$\begin{aligned} z_{ij}^{(t)} &= z_{ij}^{(t-1)} - \frac{1}{2} \left(z_{i(j+1)}^{(t-1)} - z_{ij}^{(t-1)} \right) = \frac{1}{2} \left(z_{ij}^{(t-1)} + z_{i(j+1)}^{(t-1)} \right) \\ z_{i(j+1)}^{(t)} &= z_{i(j+1)}^{(t-1)} + \frac{1}{2} \left(z_{i(j+1)}^{(t-1)} - z_{ij}^{(t-1)} \right) = \frac{1}{2} \left(z_{ij}^{(t-1)} + z_{i(j+1)}^{(t-1)} \right). \end{aligned}$$

Proceeding in this manner, IP cycles through $\mathcal{C}_1, \dots, \mathcal{C}_W$ until convergence to $\mathbf{z}^* \equiv \mathbf{Z}^*$.

The actual identification of the collection $\mathbf{Z}_1, \dots, \mathbf{Z}_K$ through successive residualizations of \mathbf{P} relies on engineering a complex interplay of the two optimization routines, QA and Dykstra's IP. The algorithm is initialized by subjecting \mathbf{P} to a QA-based search for a permutation ρ_1 providing a rearrangement of the rows and columns of \mathbf{P} , matching the desired AR form as closely as possible (denoted by \mathbf{P}_{ρ_1}). Based on the order of row and column objects, as retrieved from ρ_1 , the least-squares optimal AR matrix \mathbf{Z}_1 is fit to \mathbf{P}_{ρ_1} through IP, with constraints $\mathcal{C}_{(1)1}, \dots, \mathcal{C}_{(1)W}$ defined by (6.1). By updating the target matrix, \mathbf{B} , through \mathbf{Z}_1 , we initiate a second QA search for a possibly superior permutation ρ_1 of \mathbf{P} . The resulting \mathbf{P}_{ρ_1} will potentially even be closer to optimal AR patterning. Subsequently, \mathbf{Z}_1 is refit through IP. The algorithm cycles through QA and IP until convergence (i.e., updating \mathbf{B} by \mathbf{Z}_1 does not result in any changes of ρ_1). The residual matrix $\{p_{\rho_1(i)\rho_1(j)} - z_{(1)ij}\}$ is submitted to the search for the second AR component, \mathbf{Z}_2 . The algorithm switches back and forth between QA and IP, until ρ_2 and \mathbf{Z}_2 have been identified, yielding the residual matrix $\{\{p_{\rho_1(i)\rho_1(j)} - z_{(1)ij}\}_{\rho_2(i)\rho_2(j)} - z_{(2)ij}\}$ to be forwarded to a third QA-IP cycle bearing ρ_3 and \mathbf{Z}_3 . The algorithm continues until a complete decomposition of \mathbf{P} has been attained, usually with $K \ll N$ as Hubert and Arabie (1994) report (see also Hubert & Köhn, 2006). Evidently, the ultimate solution depends on the initial arrangement of rows and columns of \mathbf{P} (i.e., the particular order, in which \mathbf{P} is passed on to the first QA search cycle); therefore, to reduce the risk of detecting a purely locally optimal solution, a common nuisance to any heuristic procedure, utilization of the algorithm with multiple starts on the randomly permuted input matrix \mathbf{P} is strongly recommended.

6.1.3 Low AR Rank Approximation to a Proximity Matrix

Given a square-symmetric proximity matrix, \mathbf{P} , of full rank N , the rank-reduction theorem in linear algebra suggests that \mathbf{P} can be approximated by a low-rank $L \ll N$ sum of matrices of identical size as \mathbf{P} . In an analogous manner, we can state our ultimate analytic objective so as to identify a low AR rank decomposition of \mathbf{P} such that

$$\mathbf{P} \approx \mathbf{Z}_1 + \cdots + \mathbf{Z}_k + \cdots + \mathbf{Z}_L \quad \text{with } 1 \leq k \leq L < K \ll N.$$

When only L AR components are retrieved, the algorithm capitalizes on repetitively refitting the residuals of the different \mathbf{Z}_k . Explicitly, assume that the extraction of L AR components has left us with a residual matrix $\{p_{ij} - \sum_{k=1}^L z_{(k)ij}\}$. In attempting to additionally improve the fit of the decomposition obtained, the residuals $\{p_{ij} - \sum_{k=1}^L z_{(k)ij}\}$ are added back to \mathbf{Z}_1 , succeeded by another run through the QA-IP fitting cycle, very likely detecting a more effective permutation ρ_1 , producing a revised \mathbf{Z}_1 to recalculate residuals $\{p_{ij} - z_{(1)ij}\}$ that in turn are restored to the previously estimated \mathbf{Z}_2 component, to be subjected then to a new search for updating \mathbf{Z}_2 , and so on. As an immediate consequence of repetitively refitting the residuals through subsequent \mathbf{Z}_k , unlike the spectral decomposition of a matrix, the components $\mathbf{Z}_1, \dots, \mathbf{Z}_L$ are correlated.

6.1.4 Fit Measure

The quality of a specific lower AR rank approximation to \mathbf{P} is assessed through the VAF criterion defined by

$$\text{VAF} = 1 - \frac{\sum_{i < j} (p_{ij} - \sum_{k=1}^L z_{(k)ij})^2}{\sum_{i < j} (p_{ij} - \bar{p})^2} \quad (6.3)$$

$$(6.4)$$

$$= 1 - \frac{\text{tr}(\mathbf{P} - \sum_{k=1}^L \mathbf{Z}_k)(\mathbf{P} - \sum_{k=1}^L \mathbf{Z}_k)'}{\text{tr}(\mathbf{P} - \bar{\mathbf{P}})(\mathbf{P} - \bar{\mathbf{P}})'}, \quad (6.5)$$

with \bar{p} denoting the mean of the off-diagonal entries in $\mathbf{P} = \{p_{ij}\}$, and $z_{(k)ij}$ the fitted values of the k^{th} AR component.

6.1.5 The Representation of AR Matrix Components Through Secondary Structures

Each of the AR components $\mathbf{Z}_1, \dots, \mathbf{Z}_K$ lends itself immediately to a more restricted structural representation, either in the form of a unidimensional scaling (see Brusco, 2002b; Hubert & Arabie, 1986, 1988; Hubert, Arabie, & Meulman, 2002, 2006) of the set of row and column objects of the AR matrices,

$\mathcal{O} = \{O_1, \dots, O_N\}$, along a single axis, or through a discrete ultrametric tree diagram (see Barthélemy & Guénoche, 1991; Hubert & Arabie, 1995b; Hubert, Arabie, & Meulman, 2006).

For a specific AR component matrix $\mathbf{Z}_k = \{z_{(k)ij}\}$, the unidimensional scale representation can be constructed by estimating object coordinates $x_{(k)j}$, $x_{(k)i}$ on the line minimizing the least-squares loss function (including an additive constant c_k), as was presented in detail in Chapter 4:

$$\begin{aligned} L(\mathbf{x}_k, c_k) &= \sum_{i < j} \left(z_{(k)ij} - (|x_{(k)j} - x_{(k)i}| + c_k) \right)^2 \\ &= \frac{1}{2} \text{tr}(\mathbf{Z}_k - (\mathbf{D}_k + \mathbf{C}_k))(\mathbf{Z}_k - (\mathbf{D}_k + \mathbf{C}_k))', \end{aligned} \quad (6.3)$$

subject to constraints on the coordinate values implied by the triangle (in)equality:

$$\begin{aligned} O_{(k)i} &< O_{(k)j} < O_{(k)l} \\ \Rightarrow d_{(k)il} &= d_{(k)ij} + d_{(k)jl} \\ \Rightarrow |x_{(k)l} - x_{(k)i}| &= |x_{(k)j} - x_{(k)i}| + |x_{(k)l} - x_{(k)j}| \quad \forall i < j < l; \end{aligned}$$

where $\mathbf{D}_k = \{d_{(k)ij}\} = \{|x_{(k)i} - x_{(k)j}|\}$ denotes the matrix of (city-block) distances among objects $O_i, O_j \in \mathcal{O}$ defined as the absolute difference between object coordinates on the line.

An ultrametric tree structure can be characterized as a weighted acyclic connected graph with a natural root, defined as the node equidistant to all leaves or terminal nodes. The terminal nodes of an ultrametric tree represent a set of N objects, $\mathcal{O} = \{O_1, \dots, O_N\}$. As a necessary and sufficient condition for a unique ultrametric tree representation, the weights along the paths connecting objects O_i, O_j , typically with a distance interpretation and collected into an $N \times N$ matrix $\mathbf{U} = \{u_{ij}\}$, have to satisfy the three-point condition or ultrametric inequality: for any distinct object triple O_i, O_j , and O_l , the largest two path length distances among u_{ij} , u_{il} , and u_{jl} must be equal. Fitting an ultrametric structure $\mathbf{U} = \{u_{ij}\}$ to \mathbf{Z}_k also rests on minimizing a least-squares loss function:

$$L(\mathbf{U}_k) = \sum_{i < j} (z_{(k)ij} - u_{(k)ij})^2 = \frac{1}{2} \text{tr}(\mathbf{Z}_k - \mathbf{U}_k)(\mathbf{Z}_k - \mathbf{U}_k)', \quad (6.4)$$

subject to the constraints as defined by the ultrametric inequality:

$$u_{(k)ij} \leq \max \{u_{(k)il}, u_{(k)jl}\} \quad \text{for } 1 \leq i, j, l \leq N$$

(see also the detailed description given in Chapter 3).

The choice of the unidimensional scale and ultrametric tree representation is justified by a remarkable result in combinatorial data analysis (see Hubert & Arabie, 1995a; Hubert et al., 2006, Ch. 11), which links both models directly

to the AR form of a square-symmetric data matrix. Specifically, the matrices of estimated interobject (scale) distances and (ultrametric tree) path lengths themselves can be permuted into perfect AR form. More to the point, optimal unidimensional scale and ultrametric tree structures induce the AR form on the matrices of fitted values. In other words, the AR form is a necessary condition for identifying these two more restricted models, as defined by the triangle or ultrametric (in)equality constraints. We should re-emphasize that, for a given AR component, \mathbf{Z}_k , both models involve the same number of least-squares estimates. Thus, their VAF scores — obtainable through normalizing the respective loss function by $\sum_{i < j} (z_{(k)ij} - \bar{z}_k)^2$ — are on an equal scale and directly comparable as to which structure provides a superior representation of \mathbf{Z}_k . A brief technical note will be helpful to elaborate this claim. Recall that fitting a unidimensional scale includes estimating an additive constant, c to ensure that the obtained VAF fit criterion is equivalent to the (bounded) R^2 measure in regression. Hence, one might object that fitting a unidimensional scale involves an additional estimate in comparison with the ultrametric tree model. However, fitting an ultrametric structure through least-squares is translation invariant, thus not necessitating the explicit inclusion of an additive constant, c . More succinctly, models with or without intercept are identical due to the special structural side constraints imposed by the ultrametric inequality:

$$u_{ij} \leq \max \{u_{il}, u_{jl}\} \quad \Leftrightarrow \quad u_{ij} + c \leq \max \{u_{il} + c, u_{jl} + c\}.$$

However, note that translation invariance does not hold for the unidimensional structure $d_{ij} = |x_j - x_i|$ to be fit to the p_{ij} as the triangle (in)equality constraints will be violated: let $d_{il} + c = |x_l - x_i| + c$ (and $d_{ij} + c, d_{jl} + c$ correspondingly), then

$$\begin{aligned} d_{il} + c &\neq d_{ij} + c + d_{jl} + c \\ |x_l - x_i| + c &\neq |x_j - x_i| + |x_l - x_j| + 2c. \end{aligned}$$

As a final comment, observe that fitting secondary structures to the \mathbf{Z}_k by a least-squares intercept model implies the conjecture that the proximities have interval (and not ratio) scale properties.

6.1.6 Extending AR Matrix Decomposition to Three-Way Data

Our concern here will be three-way two-mode data, so the slices consist of square-symmetric two-way one-mode proximity matrices. Modelling three-way data and individual variability follow the deviation-from-the-mean principle. First, the individual proximity matrices, \mathbf{P}_s , are aggregated across sources into matrix \mathbf{P}_a , which is subsequently decomposed into a collection of L AR components through successive residualization. Second, the obtained AR matrices

then serve as frame of reference, against which the individual data matrices, \mathbf{P}_s , are fit in a confirmatory manner. Recall that the chosen reference AR structure, $\mathbf{Z}_1, \dots, \mathbf{Z}_L$, is well-defined through the associated object permutations, ρ_1, \dots, ρ_L . Thus, for a confirmatory fitting of the individual proximity matrices, we do not need to identify object orderings: they are already given to us. Hence, we can skip the QA step and continue directly with IP to numerically estimate the cell entries of the individual confirmatory AR components. Notice that the IP constraints are deducable from the known object permutations associated with each of the AR matrices of the reference structure. In technical terms, assume that for \mathbf{P}_a the low AR rank approximation $\mathbf{Z}_1, \dots, \mathbf{Z}_L$ was identified. Each component is characterized by a specific object ordering as expressed by the associated permutations ρ_1, \dots, ρ_L . The componentwise object arrangements embodied in ρ_k serve as a blueprint for (re)constructing the set of corresponding IP-constraints, $\mathcal{C}_{(k)1}, \dots, \mathcal{C}_{(k)W}$. They determine the specific AR form of the \mathbf{Z}_k , and provide a frame of reference for the confirmatory decomposition of the S individual proximity matrices into a collection of L matrices having AR form. As for each of the L components the AR-optimal object order is fixed, the confirmatory decomposition of the individual sources precludes the QA step, and proceeds directly with fitting the L source-specific AR components $\mathbf{Z}_{k(s)}$ through IP to the given individual data matrices, \mathbf{P}_s , employing the L sets of constraints $\mathcal{C}_{(k)1}, \dots, \mathcal{C}_{(k)W}$.

For each source s , we can compute a general fit measure, VAF_s , indicating how closely the observed proximities, \mathbf{P}_s , conform to the imposed AR reference structure, $\mathbf{Z}_1, \dots, \mathbf{Z}_L$. Specifically, $\sum_{k=1}^L \mathbf{Z}_{k(s)}$ represents the confirmatory AR decomposition of \mathbf{P}_s . Thus, we can compute VAF_s as the normalized sum of squared deviations between the observed proximities and the fitted confirmatory AR decomposition:

$$\begin{aligned} \text{VAF}_s &= 1 - \frac{\sum_{i < j} (p_{ij(s)} - \sum_{k=1}^L z_{(k)ij(s)})^2}{\sum_{i < j} (p_{ij(s)} - \bar{p}_s)^2} \\ &= 1 - \frac{\text{tr}(\mathbf{P}_s - \sum_{k=1}^L \mathbf{Z}_{k(s)}) (\mathbf{P}_s - \sum_{k=1}^L \mathbf{Z}_{k(s)})'}{\text{tr}(\mathbf{P}_s - \bar{\mathbf{P}}_s) (\mathbf{P}_s - \bar{\mathbf{P}}_s)'} \end{aligned} \quad (6.5)$$

In a subsequent analysis step, to each of the L confirmatory AR components, estimated for each source, a unidimensional scale and an ultrametric tree structure are fit. Hence, in addition to the general VAF measure, VAF_s , for each source, we obtain a collection of specific fit scores informing us whether the confirmatory AR components, identified for source s , are better represented by a continuous spatial or a discrete non-spatial structure. To summarize, the VAF_s scores, enable us to quantify the extent at which different sources reflect the properties of the reference structure. Moreover, the source-specific VAF measures for the continuous and discrete models, fit to each AR component, al-

Table 6.1: Frequency distributions of the VAF scores as obtained from 10,000 random starts for $K = \{1, 2, 3, 4\}$.

K	VAF	freq	perc	max	VAF increase $K \rightarrow K + 1$.
1	.54456550	4888	48.9		
	.54456551	5112	51.1	•	—
2	.93474819	6987	69.9		
	.95931572	3013	30.1	•	.41475021
3	.99892695	1817	18.2		
	.99963474	5160	51.6		
	.99981285	804	8.0		
	.99999319	2219	22.2	•	.04067747
4	.99999764	5100	51.0		
	.99999928	1904	19.0		
	.99999950	2231	22.3		
	.99999970	161	1.6		
	.99999983	604	6.0	•	.00000664

low for an in-depth inter- and intra-subject analysis, be that by comparing the fit scores between sources or within a given source s , to determine whether the continuous or discrete structure provides superior representation, with possible implications about the underlying individual cognitive processes.

6.1.7 Application: Judgments of Schematic Face Stimuli

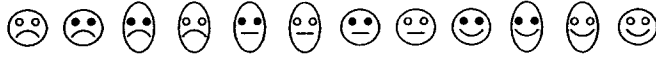
Matrix \mathbf{P}_a was subjected to a QA-IP-based search for an optimal AR decomposition with $K = \{1, 2, 3, 4\}$, employing a stand-alone MATLAB routine `multiarobfnd.m`, conveniently allowing for multiple runs with initial random permutations of the input proximity matrix. For each decomposition into K components, 10,000 random starts were used. The frequency distributions of the VAF scores obtained for each K are reported in Table 6.1; the number of decimal places used may seem excessive, but is done here to make the distinct locally optimal solutions apparent.

Figure 6.1 presents the orders of facial stimuli as discovered for the \mathbf{Z}_k belonging to the AR decomposition of \mathbf{P}_a into K components with maximum VAF.

Both, the distribution results in Table 6.1 as well as the displays of the stimulus orderings in Figure 6.1 clearly suggest that the aggregated proximity matrix \mathbf{P}_a is of AR rank $L = 2$: first, the VAF increment for $K > 3$ is minimal; second, while the order of stimuli associated with the first two components reveals an obvious pattern — \mathbf{Z}_1 : emotional appearance as captured by factor Mouth; \mathbf{Z}_2 : ‘circle’ shaped versus ‘solid’ eyes — the stimulus arrangements along the third

Figure 6.1: The orders of schematic face stimuli obtained for the various Z_k components of the AR decompositions into K components with maximum VAF score.

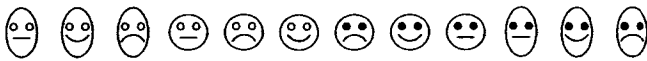
$K=1$, VAF = .54456551



$K=2$, VAF = .95931572

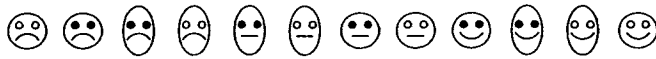


First AR Component

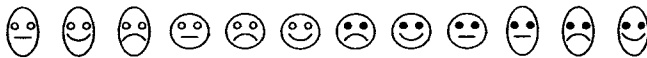


Second AR Component

$K=3$, VAF = .99999319



First AR Component

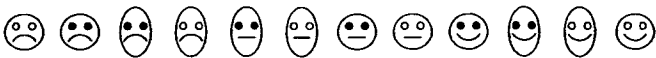


Second AR Component



Third AR Component

$K=4$, VAF = .99999983



First AR Component



Second AR Component



Third AR Component



Fourth AR Component

and fourth AR components, Z_3 and Z_4 , lack such pattern. Lastly, across all four decompositions the first two AR components maintain a stable order of the stimuli (apart from a minimal inversion occurring on the second AR component concerning the two right-most faces), which does not hold for the third and fourth AR components. Thus, the biadditive solution with the largest VAF score (.9593) was chosen as reference AR decomposition, against which the 22 individual rating matrices were fit.

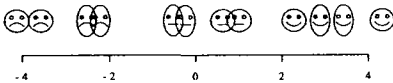
The graphs of the unidimensional scale and ultrametric tree representation of the biadditive AR reference components Z_1 and Z_2 are shown in Figure 6.2; their interpretation is straightforward: the unidimensional scale constructed for Z_1 orders the face stimuli along the continuum negative to positive expression of emotion, grouping them into three consecutive categories formed by the levels ‘frowning’, ‘flat’, and ‘smiling’ of factor Mouth (the exact position of each face stimulus is at its left eye). The corresponding ultrametric tree structure of Z_1 produces an almost perfectly balanced threefold segmentation of the face stimuli, also based on the primary criterion emotional appearance as expressed by the levels of factor Mouth. Notice that the three categories are not perceived as equally distinct; rather, ‘flat’ and ‘smile’ are merged, while ‘frown’ is still set apart. Within each group a secondary classification into faces according to Facial Shape can be observed, whereas ‘open’ and ‘solid’ circled eyes differentiate between stimuli at the tertiary level (note that this pattern is slightly violated by the ‘smile’ segment). The unidimensional scale representing Z_2 separates faces with ‘circle’ from ‘solid’ eyes, with ‘oval’ faced stimuli placed at the extreme scale poles – an arrangement of stimuli accurately reflected by the ultrametric tree structure. The latter also implies that Facial Shape and emotional appearance serve as subsequent descriptive features to group the stimuli (the tied triplet of ‘circle’ shaped faces having ‘solid’ eyes indicates a compromised fit such as the imposition of the ultrametric structure could only be accomplished by equating the respective stimulus distances). One might be tempted to conclude that the unidimensional scaling of the second AR component is degenerate because the twelve stimuli are lumped into four locations only. Upon closer inspection, however, this appears as an absolutely legitimate representation: the scale appropriately discriminates between ‘solid’ and ‘circle’ shaped eyes, while incorporating a secondary distinction based on facial shape. Thus, the retrieved scale accurately reflects a subset of constituting facial features ordering the stimuli in perfect accordance.

Fitting the biadditive AR reference structure for the 22 sources was conducted through the MATLAB routine `biarobfit.m`, performing confirmatory biadditive AR decomposition on each of the individual proximity matrices P_s , with IP constraints derived from the object orderings detected when locating the biadditive reference structure. Table 6.2 presents the results for the 22 sources,

Figure 6.2: Biadditive reference AR decomposition of \mathbf{P}_a (VAF=.959): unidimensional scale and ultrametric tree representations.

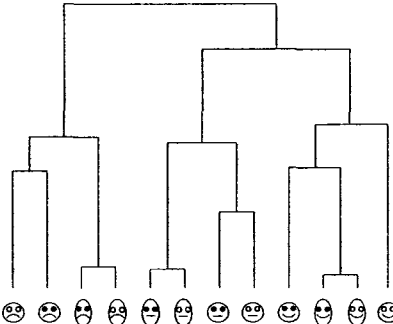
Unidimensional Scale Representation

First AR Component (VAF = .8968)

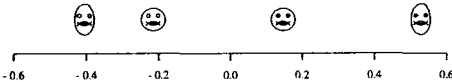


Ultrametric Tree Representation

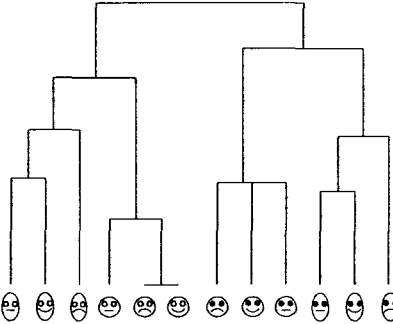
First AR Component (VAF = .7722)



Second AR Component (VAF = .2420)



Second AR Component (VAF = .7337)



sorted according to their VAF scores as defined by (6.10), indicating how closely the various individuals actually match the biadditive reference AR structure. A variety of additional diagnostic measures is provided for assessing the differential contribution of the AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$. The VAF1 and VAF2 scores indicate how well the two AR components actually fit the corresponding residualizations of \mathbf{P}_s ; they are defined similar to the overall or individual VAF measures in (6.3) and (6.10), however, with minor modifications to adjust for the respective residualization; for example:

$$\text{VAF1} = 1 - \frac{\sum_{i < j} [(p_{ij(s)} - z_{2ij(s)}) - \overline{z_{1ij(s)}}]^2}{\sum_{i < j} [(p_{ij(s)} - z_{2ij(s)}) - \overline{(p_{ij(s)} - z_{2ij(s)})}]^2},$$

where $\overline{(p_{ij(s)} - z_{2ij(s)})}$ denotes the mean of the residual matrix $\mathbf{P}_s - \mathbf{Z}_{2(s)}$. Measures $\text{VAF}(\mathbf{Z}_1)$ and $\text{VAF}(\mathbf{Z}_2)$ are obtained by refitting the final estimates of the AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$ as sole predictors to the individual proximity matrices \mathbf{P}_s to assess their marginal variance contribution. The coefficients of partial determination $R_{\mathbf{P}\mathbf{Z}_1, \mathbf{Z}_2}^2$ and $R_{\mathbf{P}\mathbf{Z}_2, \mathbf{Z}_1}^2$ quantify the marginal contribution of $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$ in reducing the total variation of \mathbf{P}_s when $\mathbf{Z}_{2(s)}$ or $\mathbf{Z}_{1(s)}$, respectively, has already been included in the model; for example:

$$R_{\mathbf{P}\mathbf{Z}_1, \mathbf{Z}_2}^2 = \frac{SSE(\mathbf{Z}_{2(s)}^*) - SSE(\mathbf{Z}_{1(s)}, \mathbf{Z}_{2(s)})}{SSE(\mathbf{Z}_{2(s)})},$$

with SSE denoting the sum of squared errors. Thus, $SSE(\mathbf{Z}_{2(s)})$ indicates the sum of squared errors when $\mathbf{Z}_{2(s)}$ serves as sole predictor, while $SSE(\mathbf{Z}_{1(s)}, \mathbf{Z}_{2(s)})$ represents the sums of squares due to error resulting when both AR components are in the model. Lastly, COV stands for the covariance of the two AR components defined by

$$\text{COV}(\mathbf{Z}_{1(s)}, \mathbf{Z}_{2(s)}) = \sum_{i,j} (z_{1ij(s)} - \bar{z}_{1(s)})(z_{2ij(s)} - \bar{z}_{2(s)}).$$

In terms of the overall VAF score, subjects 11, 19, and 21 form the top three group, while the bottom three comprise subjects 1, 4, and 6, with the top and bottom three subjects marked by \bullet , and by \circ , respectively (for ease of legibility, all original fit scores were multiplied by 1000, thereby eliminating decimal points and leading zeros). Table 6.3 reports the VAF scores obtained for the 22 sources when fitting unidimensional scales and ultrametric tree structures to the individual AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$; the VAF scores correspond to normalizations of the least-squares loss criteria in (6.6) and (6.9) by $\sum_{i < j} (z_{(k)ij(s)} - \bar{z}_{k(s)})^2$. For all sources, except subject 1, the unidimensional scaling of $\mathbf{Z}_{1(s)}$ results in a much better fit than the corresponding ultrametric tree structure; however, for the second AR component, the ultrametric tree representation attains superior fit.

It seems most instructive to focus on the extremes, and consider for further

Table 6.2: Ranking of Individual VAF scores plus various fit measures quantifying the separate contributions of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$.

Ranked Sources	VAF		VAF1	VAF2	VAF(\mathbf{Z}_1)	VAF(\mathbf{Z}_2)	$R^2_{\mathbf{P}\mathbf{Z}_1:\mathbf{Z}_2}$	$R^2_{\mathbf{P}\mathbf{Z}_2:\mathbf{Z}_1}$	COV
11	963	•	941	957	365	549	992	957	-1378
19	951	•	926	939	385	500	995	939	-834
21	948	•	940	935	484	411	991	935	-1730
12	943		950	654	877	33	996	654	-365
5	932		931	864	648	237	990	864	-1224
13	914		907	846	602	242	983	846	-1088
20	885		894	590	777	99	981	590	-806
9	885		886	861	467	315	980	861	-1887
14	883		830	857	303	451	984	857	-785
8	874		862	784	559	283	983	784	-706
17	862		874	606	706	121	968	606	-1026
16	860		769	847	262	551	969	847	-1108
2	849		779	792	395	453	980	792	-622
22	841		771	858	212	533	973	858	-1793
10	838		785	759	424	350	975	759	-444
15	836		820	736	481	238	974	736	-893
18	813		732	767	328	473	969	767	-598
7	803		787	797	313	374	972	797	-988
3	793		701	759	267	433	964	759	-395
6	748	◦	724	611	421	242	963	611	-505
4	747	◦	693	643	354	248	952	643	-782
1	262	◦	135	229	70	186	965	229	-53

Table 6.3: Individual VAF scores for the unidimensional scale (US) and ultrametric tree (UT) representations of the two AR components $Z_{1(s)}$ and $Z_{2(s)}$

Ranked Sources	VAF(US 1)	VAF(UT 1)	VAF(US 2)	VAF(UT 2)
11	914	619	222	832
19	870	719	221	684
21	914	671	196	716
12	900	776	208	583
5	918	861	227	774
13	941	754	240	687
20	940	878	246	552
9	914	697	265	769
14	855	693	264	650
8	927	664	251	730
17	962	723	222	818
16	883	796	233	745
2	852	737	241	602
22	882	737	194	745
10	893	800	284	782
15	892	723	203	679
18	861	734	261	715
7	899	599	222	811
3	885	728	237	729
6	896	684	148	641
4	878	763	170	839
1	768	818	204	797

inspection only individuals who are exceptionally poorly or well-represented: the top and bottom three subjects. For succinctness, from the top three group only the best fitting source, subject 11, is chosen, whereas the results for the bottom group, subjects 1, 4, and 6, are completely documented because they promise to provide deeper diagnostic insight into how sources with data of apparent ill-fit are handled by the AR decomposition. Figures 6.3 and 6.4 present the unidimensional scales and ultrametric tree graphs fit to the two individual AR components $Z_{1(s)}$ and $Z_{2(s)}$ (for the ultrametric dendrograms the face stimuli have been arranged to match their order associated with the respective AR components; recall that for a fixed ultrametric structure there exist $2^{(N-1)}$ different ways of positioning the terminal object nodes of the tree diagram).

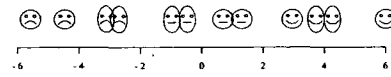
Not too surprising, the unidimensional scale as well as the non-spatial tree representations of Z_1 and Z_2 for Subject 11 are almost identical to those of the reference structure, which by and large also applies to the unidimensional scales constructed for subjects 6 and 4. However, the findings for the latter two sources are put into perspective by their ultrametric dendrograms: wildly dispersed branches, accompanied by several misallocations of stimuli, particularly for the representations of the second AR component, indicate an at best mediocre fit of the data from respondents 6 and 4 to the imposed AR structures. Such ambiguities do not pertain to subject 1, who is an obvious misfit: first, notice the equidistant alignment of stimuli along the unidimensional scale representing the first AR component, signaling a presumable lack of differentiation in the judgments of this subject. Moreover, inspecting the remaining three graphs corroborates the notion of the problematic nature of the data contributed by source 1 as the face stimuli appear to be arranged more or less in random order.

To further substantiate the tentative conclusions from the confirmatory analysis, the individual data matrices of respondents 11, 6, 4 and 1 were reanalyzed through independent biadditive AR decompositions. Based on the logic that the confirmatory fitting could lead to a compromised structural representation of the individual data by imposing constraints inappropriate for a particular source, one might consider the obtained confirmatory configurations – in a transferred sense – as a lower bound representation. In other words, we would expect a well-fitting subject to profit from an independent analysis with the prospect of an only slightly better representation. For sources with poor fit, however, the independent analysis will either yield a substantial increase in fit because their idiosyncratic perspective, in disagreement with the majority of the sample, will no longer be distorted by an inadequate confirmatory reference frame, or these subjects will qualify as having provided just inconsistent, or random judgments not amenable to any improved data representation. Each of the four individual

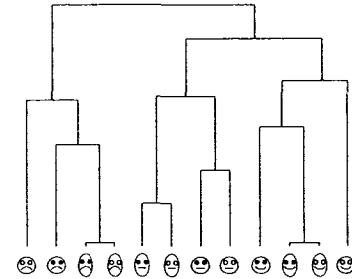
Figure 6.3: Confirmatory biadditive AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the first AR component $Z_{1(s)}$.

Subject 11 (VAF = .9628)

Unidimensional Scale Representation (VAF = .9144)

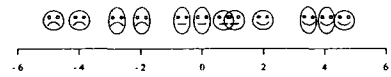


Ultrametric Tree Representation (VAF = .6188)

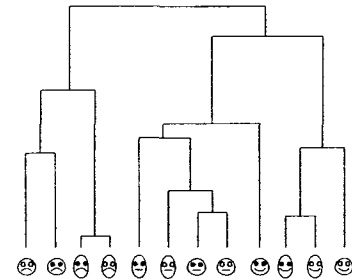


Subject 6 (VAF = .7477)

Unidimensional Scale Representation (VAF = .8962)

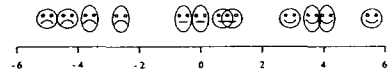


Ultrametric Tree Representation (VAF = .6844)

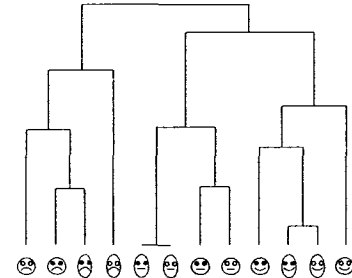


Subject 4 (VAF = .7465)

Unidimensional Scale Representation (VAF = .8784)

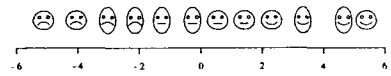


Ultrametric Tree Representation (VAF = .7630)



Subject 1 (VAF = .2621)

Unidimensional Scale Representation (VAF = .7676)



Ultrametric Tree Representation (VAF = .8180)

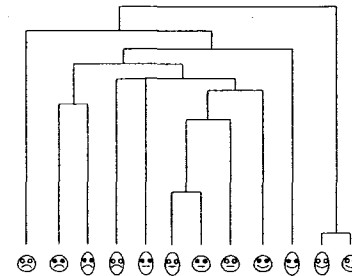
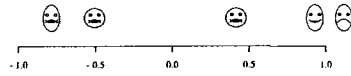


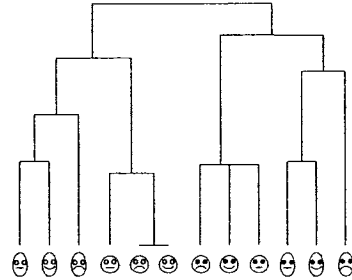
Figure 6.4: Confirmatory biadditive reference AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the second AR component $Z_{2(s)}$.

Subject 11 (VAF = .9628)

Unidimensional Scale Representation (VAF = .2219)

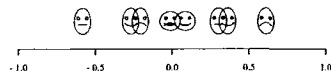


Ultrametric Tree Representation (VAF = .8321)

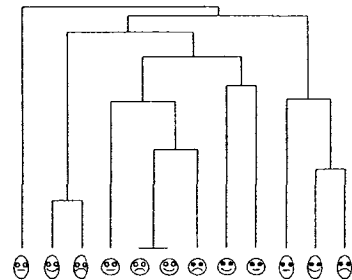


Subject 6 (VAF = .7477)

Unidimensional Scale Representation (VAF = .1481)

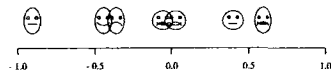


Ultrametric Tree Representation (VAF = .6408)

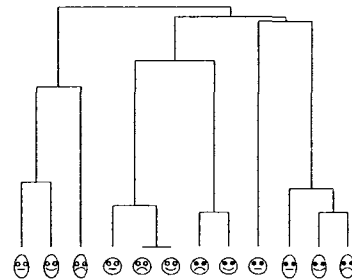


Subject 4 (VAF = .7465)

Unidimensional Scale Representation (VAF = .1701)

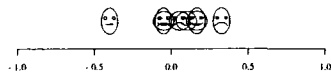


Ultrametric Tree Representation (VAF = .8385)



Subject 1 (VAF = .2621)

Unidimensional Scale Representation (VAF = .2042)



Ultrametric Tree Representation (VAF = .7966)

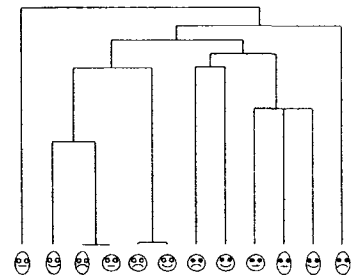


Table 6.4: Independent biadditive AR decomposition of selected sources: Individual VAF scores plus various fit measures quantifying the separate contributions of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$.

Ranked Sources	VAF	VAF1	VAF2	VAF(\mathbf{Q}_1)	VAF(\mathbf{Q}_2)	$R^2_{\mathbf{P}\mathbf{Q}_1\mathbf{Q}_2}$	$R^2_{\mathbf{P}\mathbf{Q}_2\mathbf{Q}_1}$	COV
11	981	980	966	624	332	996	966	-1494
6	967	968	955	568	283	995	955	-1237
4	924	932	912	541	349	986	912	-2485
1	970	964	913	754	483	999	913	-99

Table 6.5: Individual VAF scores for the unidimensional scale (US) and ultrametric tree (UT) representations of the two AR components $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$ from the independent analysis of selected sources.

Ranked Sources	VAF(US 1)	VAF(UT 1)	VAF(US 2)	VAF(UT 2)
11	941	693	237	632
6	919	574	260	587
4	937	699	278	721
1	822	858	197	650

proximity matrices was subjected to 10,000 random starts of the appropriate MATLAB routine (`biarobfnd.m`); the solutions with the highest VAF score were chosen as final representations. The key diagnostics for subjects 11, 6, 4, and 1 are reported in Table 6.4. For all sources the overall VAF scores imply that their data are exhaustively approximated by an AR decomposition of rank $l = 2$. The VAF scores for the unidimensional scale and ultrametric tree representations fit subsequently to the two AR components are presented in Table 6.5. As before, the unidimensional scale representations of the $\mathbf{Z}_{1(s)}$ attain much higher VAF values across all four sources than the ultrametric tree structures, while the latter provide superior representations of the second AR components $\mathbf{Z}_{2(s)}$.

For sources 11, 6, 4 and 1, graphs of the unidimensional scales and ultrametric tree structures for $\mathbf{Z}_{1(s)}$ and $\mathbf{Z}_{2(s)}$ are given in Figures 6.5 and 6.6, respectively. The scale and dendrogram displays obtained for subject 11 afford an immediate interpretation: the arrangement of facial stimuli associated with the first AR component is dominated by the factor Eyes, with factors Facial Shape and Mouth as secondary and tertiary perceptual criteria; the ordering of faces identified with the second AR component is determined by the feature hierarchy Mouth, Facial Shape and Eyes. So, the independent analysis reveals that subject 11, in comparison with the reference structure, utilizes the identi-

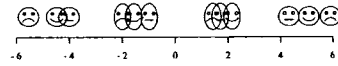
cal feature hierarchies, but in reversed order: whereas the order of stimuli along the first AR reference component can be characterized in terms of the sequential utilization of criteria Mouth, Facial Shape and Eyes, subject 11 employs the exact same hierarchy on the second independent AR component, and vice versa. For subject 6, the results are far less consistent. The independent analysis identifies Mouth as the primary factor, followed by Facial Shape and Eyes, which is in agreement with the first AR component of the reference structure. But this order of criteria only applies to faces with 'flat' or 'frowning' mouth lines; for smiling faces we observe a switch to Eyes first, then Facial Shape — a finding also evidenced by the dendrogram. Its branching pattern indicates additional inconsistencies in the grouping of stimuli as exemplified by the misclassifications of the two stimuli with 'oval' face, 'solid' eyes and 'flat' mouth, or 'circle' shaped face, 'solid' eyes and 'frowning' mouth line, respectively. The displays for the second AR component confirm these conclusions because both the unidimensional scale as well as the dendrogram are hardly interpretable and, rather resemble a non-systematic, ad hoc mix of factors Facial Shape and Eyes. Similar findings can be stated for subject 4: the scale as well as the tree representation of the first AR component appear to be determined by an inconsistent combination of factors Eyes and Facial Shape, while the arrangement of stimuli in the representations of the second AR component, seems to be governed by Mouth, but also in an inconsistent manner. In summarizing, remarkably and contrary to our expectations, respondents 6 and 4, do not emerge as subjects with a deviant, nevertheless well interpretable pattern of perceptual criteria, but as sources with profound inconsistencies in their judgments. The fuzzy results for source 1 allow for only an obvious explanation: the arrangement of the face stimuli reveals no discernable relationship, implying that subject 1 just contributed random responses.

Under substantive considerations, the results suggest that the set of schematic face stimuli is best represented by a combination of continuous spatial and discrete non-spatial structures: the first AR component, associated with factor Mouth and interpretable as emotional appearance, receives superior representation through a unidimensional scale, whereas the second AR component, reflecting a mix of Facial Shape and Eyes, is better represented by a discrete non-spatial structure — from hindsight not too surprising, given the specific manner the stimuli were generated. The most remarkable finding concerns the independent analysis of subjects 6 and 4, who, although attaining numerically satisfactory fit scores, do not gain from it in terms of a more meaningful stimulus representation. Instead, and contrary to the initial hypothesis attributing their mediocre confirmatory fit to the biadditive AR reference structure as too

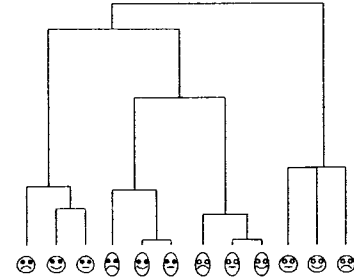
Figure 6.5: Independent biadditive AR decomposition for selected sources: uni-dimensional scale and ultrametric tree representations for the first AR component $\mathbf{Z}_{1(s)}$.

Subject 11 (VAF = .9809)

Unidimensional Scale Representation (VAF = .9410)

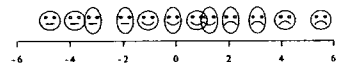


Ultrametric Tree Representation (VAF = .6931)

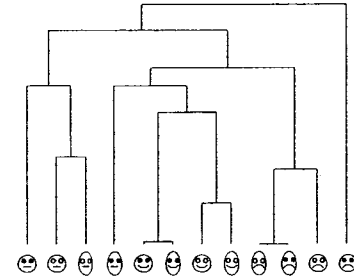


Subject 6 (VAF = .9667)

Unidimensional Scale Representation (VAF = .9189)

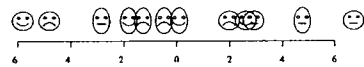


Ultrametric Tree Representation (VAF = .5736)

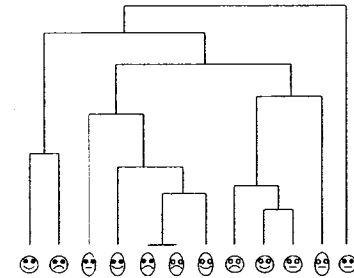


Subject 4 (VAF = .9243)

Unidimensional Scale Representation (VAF = .9368)

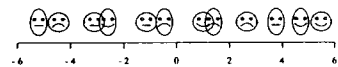


Ultrametric Tree Representation (VAF = .6987)



Subject 1 (VAF = .9702)

Unidimensional Scale Representation (VAF = .8217)



Ultrametric Tree Representation (VAF = .8584)

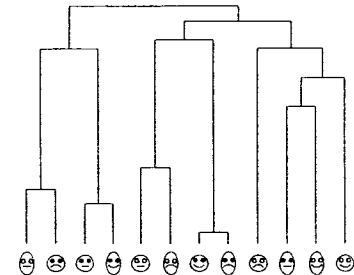
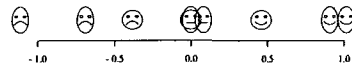


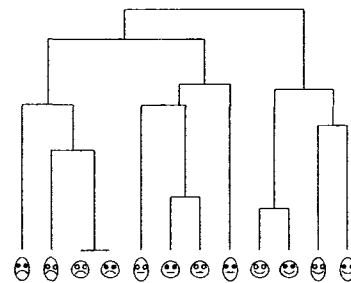
Figure 6.6: Independent biadditive AR decomposition for selected sources: unidimensional scale and ultrametric tree representations for the second AR component $Z_{2(s)}$.

Subject 11 (VAF = .9809)

Unidimensional Scale Representation (VAF = .2368)

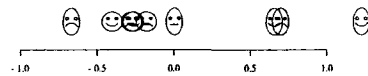


Ultrametric Tree Representation (VAF = .6318)

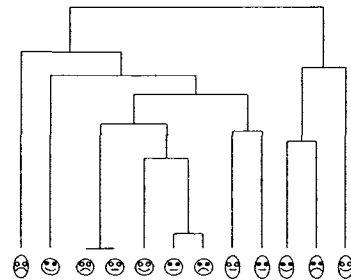


Subject 6 (VAF = .9667)

Unidimensional Scale Representation (VAF = .2600)

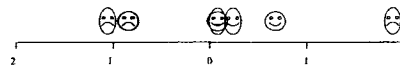


Ultrametric Tree Representation (VAF = .5871)

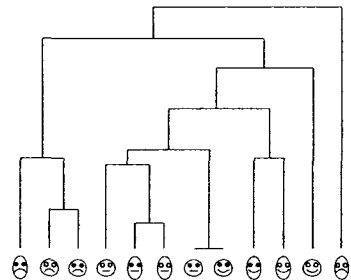


Subject 4 (VAF = .9243)

Unidimensional Scale Representation (VAF = .2782)



Ultrametric Tree Representation (VAF = .7141)

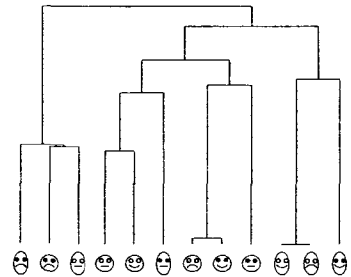


Subject 1 (VAF = .9702)

Unidimensional Scale Representation (VAF = .1972)



Ultrametric Tree Representation (VAF = .6504)



restrictive for their distinctive perception, the outcome of the independent analysis rather confirms the notion that these subjects simply entertained a weakly elaborated, inconsistent mix of criteria.

6.2 Order-Constrained (Anti-)Q Matrix Decomposition

6.2.1 Definitions and Formal Concepts

Recall that the rows and columns of an $N_R \times N_C$ rectangular two-way two-mode proximity matrix, $\mathbf{Q} = \{q_{rc}\}$, represent two distinct sets of entities, $\mathcal{O}_R = \{O_{1R}, \dots, O_{N_R R}\}$ and $\mathcal{O}_C = \{O_{1C}, \dots, O_{N_C C}\}$, with N_R and N_C elements, respectively, and $r = 1, \dots, N_R$ and $c = 1, \dots, N_C$; the within-sets proximities are not available. Generally, the anti-Q form of a rectangular matrix $\mathbf{Z} = \{z_{rc}\}$ is characterized by a pattern of row and column entries nonincreasing to a minimum, the inflection points, h_r and h_c , respectively, and nondecreasing thereafter. Formally, for each row r “there exists a (not necessarily unique) integer h_r , $1 \leq h_r \leq N_C$, such that”

$$\begin{aligned} z_{rh} &\geq z_{r(h+1)} && \text{for } 1 \leq h \leq h_r - 1, \\ z_{rh} &\leq z_{r(h+1)} && \text{for } h_r \leq h \leq N_C - 1, \end{aligned}$$

and for each column c “there exists a (not necessarily unique) integer h_c , $1 \leq h_c \leq N_R$, such that”

$$\begin{aligned} z_{hc} &\geq z_{(h+1)c} && \text{for } 1 \leq h \leq h_c - 1, \\ z_{hc} &\leq z_{(h+1)c} && \text{for } h_c \leq h \leq N_R - 1 \end{aligned}$$

(see Hubert & Arabie, 1995a, p. 576–577).

We seek to decompose \mathbf{Q} into the sum of matrices $\mathbf{Z}_1 + \dots + \mathbf{Z}_k + \dots + \mathbf{Z}_K$, with K at most equal to $\min\{N_R, N_C\}$, such that the least-squares loss function,

$$\begin{aligned} L(\mathbf{Z}_1, \dots, \mathbf{Z}_K) &= \sum_r \sum_c (q_{rc} - \sum_k z_{(k)rc})^2 \\ &= \text{tr} \left(\mathbf{Q} - \sum_k \mathbf{Z}_k \right) \left(\mathbf{Q} - \sum_k \mathbf{Z}_k \right)', \end{aligned}$$

is minimized, given the constraint of all \mathbf{Z}_k having anti-Q form.

The sum of anti-Q matrices is obtained through successive residualizations of \mathbf{Q} , and proceeds in the exact same manner as order-constrained AR decomposition. However, the specific patterning of anti-Q matrices requires the identification of a collection of pairs of suitable row and column permutations $\{\varphi_1, \psi_1\}, \dots, \{\varphi_K, \psi_K\}$, with $\varphi(\cdot) \equiv \varphi$ and $\psi(\cdot) \equiv \psi$ denoting permutations of the first N_R and N_C integers, respectively, rearranging \mathbf{Q} and the subsequent

residual matrices into a form that matches the desired anti-Q pattern as closely as possible. Thus, like order-constrained AR decomposition, finding a least-squares optimal anti-Q decomposition of \mathbf{Q} into $\sum_k \mathbf{Z}_k$ involves the search for a collection of permutations $\{\varphi_1, \psi_1\}, \dots, \{\varphi_K, \psi_K\}$, for rearranging the matrices into approximate anti-Q form, followed by estimating the anti-Q components, $\mathbf{Z}_1, \dots, \mathbf{Z}_K$, conforming to the order constraints as established through the respective row and column permutations.

6.2.2 Optimal Anti-Q Decomposition: Algorithmic and Computational Details

The search for $\{\varphi_1, \psi_1\}, \dots, \{\varphi_K, \psi_K\}$ presents a major technical challenge to order-constrained anti-Q matrix decomposition. Distinct from a square-symmetric $N \times N$ proximity matrix $\mathbf{P} = \{p_{ij}\}$, with $p_{ij} = p_{ji}$, $p_{ii} = 0$ and $1 \leq i, j \leq N$, where a single permutation ρ simultaneously rearranges the rows and columns of \mathbf{P} (for example, into AR form), the rectangular format of \mathbf{Q} requires separate permutations of rows and columns, φ and ψ . However, a (presumably) iterative search procedure interchanging row and column permutations, even though transforming \mathbf{Q} into anti-Q form would place the inflection points h_r and h_c at ambiguous and not necessarily optimal positions. Instead, the specific nonincrease-nondecrease patterning about the minima of row and column entries of an anti-Q form, requires the incorporation of inflection point locations, h_r and h_c , explicitly into the search for optimal permutations φ and ψ , posing a further complication.

Hubert and Arabie (1995a) devised an ingenious solution to this problem by embedding the two-way two-mode proximity matrix \mathbf{Q} into a 2×2 square-symmetric supermatrix $\mathbf{P}^{(\mathbf{Q})} = \{p_{ij}^{(\mathbf{Q})}\}$, resulting in a matrix of size $N \times N$, with $N = N_R + N_C$ (thus, merging the previous separate indices $1, \dots, N_R$ and $1, \dots, N_C$ into one set, with the first N_R row objects subscripted as $1, \dots, N_R$ and the remaining N_C column objects assigned indices $N_R + 1, \dots, N_R + N_C = N$). The off-diagonal blocks of $\mathbf{P}^{(\mathbf{Q})}$ contain the between-sets proximities of \mathbf{Q} , whereas the missing within-sets proximities (i.e., between row and column objects) in the submatrices blocks along the main diagonal are set to zero:

$$\mathbf{P}^{(\mathbf{Q})} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{Q} \\ \hline \mathbf{Q}' & \mathbf{0} \end{array} \right].$$

Given the specific block structure of $\mathbf{P}^{(\mathbf{Q})}$, the task of identifying separate row and column permutations, φ and ψ , of \mathbf{Q} can be condensed into the search for a single permutation, $\rho = \varphi \cup \psi$, of $\mathbf{P}^{(\mathbf{Q})}$ in “an attempt to find a row/column reordering as close as possible to an anti-Robinson form in its nonmissing entries. In turn, any reorderings of $\mathbf{P}^{(\mathbf{Q})}$ must also induce separate row and column reorderings for the optimal $N_R \times N_C$ proximity matrix \mathbf{Q} ” (Hubert et al., 2001a,

p. 69–70). For a given permutation of the first N integers, $\rho(\cdot) \equiv \rho$, the AR form of $\mathbf{P}^{(\mathbf{Q})}$ is defined by

$$\begin{aligned} p_{\rho(i)\rho(j)}^{(\mathbf{Q})} &\leq p_{\rho(i)\rho(j+1)}^{(\mathbf{Q})} && \text{for } 1 \leq i < j \leq N-1, \\ p_{\rho(i)\rho(j)}^{(\mathbf{Q})} &\leq p_{\rho(i)\rho(j-1)}^{(\mathbf{Q})} && \text{for } 2 \leq j < i \leq N. \end{aligned}$$

“If $\mathbf{P}^{(\mathbf{Q})}$ can be placed in a perfect anti-Robinson form for the objects in $\mathcal{O} = \mathcal{O}_R \cup \mathcal{O}_C$ the reordered matrix \mathbf{Q} would also display a perfect order pattern for its entries within each row and within each column. Explicitly, the entries within each row (or column) would be nonincreasing to a minimum and nondecreasing thereafter. Such a pattern can be called an anti-Q form” (Hubert et al., 2001a, p. 69–70 — notation modified for consistency).

In addition, embedding \mathbf{Q} into $\mathbf{P}^{(\mathbf{Q})}$ and subsequently permuting the latter into a particular row/column arrangement ρ such that the magnitude pattern of nonzero cell entries best approximates the AR form, denoted by $\mathbf{P}_\rho^{(\mathbf{Q})} = \{p_{\rho(i)\rho(j)}^{(\mathbf{Q})}\}$, as a byproduct yields optimal positions for the row and column inflection points of \mathbf{Q} , located about the main diagonal of $\mathbf{P}_\rho^{(\mathbf{Q})}$. When reordering $\mathbf{P}^{(\mathbf{Q})}$ in search for ρ , we keep track of the positions of the entries q_{rc} . Thus, after ρ has been identified, we can locate and extract them from $\mathbf{P}_\rho^{(\mathbf{Q})}$ to construct the reordered matrix, $\mathbf{Q}_{\varphi,\psi} = \{q_{\varphi(r)\psi(c)}\}$, satisfying the anti-Q form as faithfully as possible. The search for the optimal permutation ρ^* of $\mathbf{P}^{(\mathbf{Q})}$ is carried out through QA.

A brief comment for clarifying a technical detail of the QA-induced row and column reordering of $\mathbf{P}^{(\mathbf{Q})}$ might be helpful: were we to apply the QA exchange/interchange algorithm to $\mathbf{P}^{(\mathbf{Q})}$ in its primary block structure, the reordering process would render $\mathbf{P}^{(\mathbf{Q})}$ in its initial partitioning, with only the partial nonzero rows and columns within \mathbf{Q} and \mathbf{Q}' permuted because the entries in the zero blocks on the main diagonal of $\mathbf{P}^{(\mathbf{Q})}$ will not be picked up by the reordering mechanism as they cannot contribute to any increase of the QA-criterion. As a consequence, the initial order of rows and columns of $\mathbf{P}^{(\mathbf{Q})}$ will be completely preserved, keeping the two sets strictly apart, and so amounting to performing (simultaneously) separate permutations for the row and column sets of \mathbf{Q} . However, as already mentioned, such a procedure is not particularly germane as the resulting locations of the inflection points might not be optimal. Specifically, although the patterning of $\mathbf{Q}_{\varphi,\psi}$ (or $\mathbf{Q}'_{\psi,\varphi}$), retrieved in such a manner, will match an anti-Q form as closely as possible, it may not represent the best one for any given \mathbf{Q} because the AR form imposed on $\mathbf{P}^{(\mathbf{Q})}$ confines the inflection points h_r and h_c to locations within the first column and last row of the permuted matrix $\mathbf{Q}_{\varphi,\psi}$, respectively. Greater flexibility for choosing the position of the inflection points, however, could be gained were the QA search allowing for mixing the orders of row and column sets. Subtracting the grand mean, $\bar{m} = [1/(N_R N_C)] \sum_r \sum_c q_{rc}$, from the entries in \mathbf{Q} and \mathbf{Q}' of $\mathbf{P}^{(\mathbf{Q})}$ will turn the zero blocks into meaningful entities, thereby inducing the

desired mixing of row and column objects, leading to a variable placement of the inflection points at any most suitable row or column positions of \mathbf{Q} . Whether mean-deviation is the best strategy awaits further research; to date the effect of certain transformations combined with specific patterning of $\mathbf{P}^{(\mathbf{Q})}$ on the reordering process has not been studied in detail (see Hubert & Arabie, 1995a, 1995b).

The second optimization task of estimating a least-squares approximation \mathbf{Z} (in perfect anti-Q form) to $\mathbf{Q}_{\varphi^*, \psi^*}$ is carried out through IP. For computational convenience, the IP estimation step is conducted directly on $\mathbf{P}_{\rho^*}^{(\mathbf{Q})}$, not on $\mathbf{Q}_{\varphi^*, \psi^*}$.

Recall that, given the actual numerical entries in $\mathbf{P}^{(\mathbf{Q})}$, ρ^* indicates the best possible reordering of the N rows and columns of $\mathbf{P}^{(\mathbf{Q})}$ into a form matching an AR pattern. Our goal is to estimate an $N \times N$ least-squares approximation $\mathbf{Z}^\circ = \{z_{ij}^\circ\}$ to $\mathbf{P}_{\rho^*}^{(\mathbf{Q})}$, which has perfect AR form. Subsequently, the desired perfect anti-Q least-squares approximation \mathbf{Z}^* to $\mathbf{Q}_{\varphi^*, \psi^*}$ is extracted from \mathbf{Z}° . In other words, by using $\mathbf{P}^{(\mathbf{Q})}$ as a vehicle for the successive residualization of \mathbf{Q} and the actual identification of the sum $\mathbf{Z}_1 + \dots + \mathbf{Z}_K$, the decomposition of \mathbf{Q} relies on the exact same algorithm as order-constrained AR decomposition of a square-symmetric proximity matrix \mathbf{P} .

6.2.3 Low-Anti-Q-Rank Approximation to a Proximity Matrix

The ultimately desired low-anti-Q-rank approximation to \mathbf{Q} can be formalized as

$$\mathbf{Q} \approx \mathbf{Z}_1 + \dots + \mathbf{Z}_k + \dots + \mathbf{Z}_L \quad \text{with } 1 \leq k \leq L \ll K \leq \min\{R, C\}.$$

When only L anti-Q components are retrieved, the algorithm capitalizes on repetitively refitting the residuals of the different \mathbf{Z}_k , yielding correlated anti-Q components $\mathbf{Z}_1, \dots, \mathbf{Z}_L$.

6.2.4 Fit Measure

The quality of a specific low-anti-Q-rank approximation to \mathbf{Q} is assessed through the VAF criterion defined by

$$\text{VAF} = 1 - \frac{\sum_r \sum_c (q_{rc} - \sum_{k=1}^L z_{(k)rc})^2}{\sum_r \sum_c (q_{rc} - \bar{q})^2} = 1 - \frac{\text{tr}(\mathbf{Q} - \sum_{k=1}^L \mathbf{Z}_k)(\mathbf{Q} - \sum_{k=1}^L \mathbf{Z}_k)'}{\text{tr}(\mathbf{Q} - \bar{\mathbf{Q}})(\mathbf{Q} - \bar{\mathbf{Q}})'},$$

with \bar{q} denoting the mean of the entries in \mathbf{Q} , and $z_{(k)rc}$ the fitted values of the k^{th} anti-Q component.

6.2.5 The Representation of Anti-Q Matrix Components Through Secondary Structures

The relationship among the elements in \mathcal{O}_R and \mathcal{O}_C of each anti-Q component \mathbf{Z}_k can be represented either in (continuous) form through unidimensional unfolding (see Hubert & Arabie, 1995a; Hubert et al., 2001a, 2006), or as a (discrete) two-mode ultrametric tree diagram (see Barthélemy & Guénoche, 1991; De Soete, DeSarbo, Furnas, & Carroll, 1984; Hubert & Arabie, 1995a; Hubert et al., 2006).

For a specific anti-Q component matrix $\mathbf{Z}_k = \{z_{(k)rc}\}$, the unidimensional unfolding representation can be constructed by estimating object coordinates $x_{(k)r}$ and $y_{(k)c}$ on the line for objects in the row set and column sets, \mathcal{O}_R and \mathcal{O}_C , respectively, minimizing the least-squares loss function (by including an additive constant c_k):

$$L(\mathbf{x}_k, \mathbf{y}_k, c_k) = \sum_r \sum_c (z_{(k)rc} - |x_{(k)r} - y_{(k)c}| - c_k)^2, \quad (6.6)$$

subject to order constraints on the coordinate estimates for row and column objects imposed by ρ_k^* associated with \mathbf{Z}_k :

1. $O_{rR} \prec O_{r'R} \prec O_{cC} \prec O_{c'C}$ implies $d_{rc} + d_{r'c'} = d_{rc'} + d_{r'c}$;
2. $O_{rR} \prec O_{cC} \prec O_{r'R} \prec O_{c'C}$ implies $d_{rc} + d_{r'c} + d_{r'c'} = d_{rc'}$;
3. $O_{rR} \prec O_{cC} \prec O_{c'C} \prec O_{r'R}$ implies $d_{rc} + d_{r'c} = d_{rc'} + d_{r'c'}$;
4. $O_{rR} \prec O_{r'R} \prec O_{cC}$ implies $d_{r'c} \leq d_{rc}$;
5. $O_{rR} \prec O_{cC} \prec O_{c'C}$ implies $d_{rc} \leq d_{rc'}$,

with the interobject distances defined as $d_{rc} := |x_r - y_c|$ (see Hubert et al., 2006, p. 50, and the detailed technical description in Chapter 5).

In the special case of a two-mode proximity matrix \mathbf{Q} , a joint (or two-mode) ultrametric tree representation $\mathbf{U} = \{u_{rc}\}$ of the two sets of objects, \mathcal{O}_R and \mathcal{O}_C , is sought. Specifically, for a given anti-Q component $\mathbf{Z}_k = \{z_{(k)rc}\}$ we want to minimize

$$L(\mathbf{U}_k) = \sum_r \sum_c (z_{(k)rc} - u_{(k)rc})^2, \quad (6.7)$$

subject to two-mode ultrametric constraints such that, for all distinct quadruples of objects $(O_{rR}, O_{r'R}, O_{cC}, O_{c'C})$, with path length distances u_{rc} , $u_{rc'}$, $u_{r'c}$ and $u_{r'c'}$, the two largest must be equal (see Furnass, 1980; Hubert et al., 2006).

For a detailed technical description of how to construct a unidimensional unfolding or a two-mode ultrametric tree representation, the reader may want to refer to the cited references. Like in the AR case, the continuous and discrete models fit to the anti-Q components are on an equal scale and immediately comparable as to which structure provides a superior representation of \mathbf{Z}_k . Lastly,

recall that fitting the unfolding model also includes estimating an additive constant, c ; yet, the two-mode ultrametric structure is translation invariant and does not necessitate the explicit inclusion of an additive constant, c :

$$u_{rc} \leq \max \{u_{rc'}, u_{r'c}, u_{r'c'}\} \Leftrightarrow u_{rc} + c \leq \max \{u_{rc'} + c, u_{r'c} + c, u_{r'c'} + c\}.$$

The unfolding model, however, is not invariant to translation: the second order constraint, $d_{rc} + d_{r'c} + d_{r'c'} = d_{rc'}$, will be violated if all distances are incremented by c :

$$d_{rc} + d_{r'c} + d_{r'c'} + 3c \neq d_{rc'} + c.$$

6.2.6 Extending Anti-Q Matrix Decomposition to Three-Way Data

The slices of the three-way data cube consist of rectangular two-way two-mode proximity matrices, so the entire data set is characterized as three-way three-mode. The three-way extension of order-constrained anti-Q decomposition to the analysis of individual differences follows exactly the procedure outlined for three-way AR matrix decomposition within a deviation-from-the-mean framework. First, the individual proximity matrices, \mathbf{Q}_s , are aggregated across sources into matrix \mathbf{Q}_a , which is subsequently decomposed into a collection of L anti-Q components. Second, the obtained anti-Q matrices then serve as frame of reference, against which the individual data matrices, \mathbf{Q}_s , are fit in a confirmatory manner.

For each source s , we can compute a general fit measure, VAF_s , indicating how closely the observed proximities, \mathbf{Q}_s , conform to the imposed anti-Q reference structure, $\mathbf{Z}_1, \dots, \mathbf{Z}_L$:

$$\begin{aligned} \text{VAF}_s &= 1 - \frac{\sum_r \sum_c (q_{rc(s)} - \sum_{k=1}^L q_{(k)rc(s)})^2}{\sum_r \sum_c (q_{rc(s)} - \bar{q}_s)^2} \\ &= 1 - \frac{\text{tr}(\mathbf{Q}_s - \sum_{k=1}^L \mathbf{Z}_{k(s)})(\mathbf{Q}_s - \sum_{k=1}^L \mathbf{Z}_{k(s)})'}{\text{tr}(\mathbf{Q}_s - \bar{\mathbf{Q}}_s)(\mathbf{P}_s - \bar{\mathbf{Q}}_s)'} \end{aligned} \quad (6.8)$$

Subsequently, the L confirmatory anti-Q components, estimated for each source, are fit by a unidimensional unfolding and two-mode ultrametric tree structure. The resulting collections of specific VAF scores inform us whether the confirmatory anti-Q components, identified for source s , are better represented by a continuous spatial or a discrete non-spatial structure and allow for a complex inter- as well as intraindividual analysis.

Table 6.6: Frequency distributions of the VAF scores as obtained from 10,000 random starts on *TOTAL* for $K = \{1, 2\}$.

K	VAF	Frequency	Percent	Maximum	VAF Increase $K \rightarrow K + 1$.
1	.58522531	1187	11.9		
	.62487195	723	7.2		
	.89237372	8090	80.9	•	—
2	.99168640	1214	12.1		
	.99507188	1758	17.6		
	.99682924	884	8.8		
	.99866220	501	5.0		
	.99869583	1566	15.7		
	.99869657	1504	15.0		
	.99926377	1790	17.9		
	.99980678	215	2.2		
	.99991288	83	.8		
	.99998761	37	.4		
	.99998967	448	4.5	•	.10761595

6.2.7 Application: The Contraceptive Data from Weller and Romney (1990)

Matrix *TOTAL* was subjected to a QA-IP-based search for an optimal and exhaustive anti-Q decomposition, employing a stand-alone MATLAB routine `multiarobfndtm.m`, conveniently allowing for multiple runs with initial random permutations of the input proximity matrix. For each decomposition into K components, 10,000 random starts were used. The frequency distributions of the VAF scores obtained for different K are reported in Table 6.6; the number of decimal places used may seem excessive, but is done here to make the distinct locally optimal solutions apparent.

The distribution results in Table 6.6 suggest that matrix *TOTAL* can be exhaustively decomposed into $K = 2$ anti-Q components. The VAF increment, however, for $K > 1$ is relatively small. In addition, subsequent analyses, not reported here, revealed that the order of evaluative criteria and contraceptive measures associated with the second anti-Q component, Q_2 , was not interpretable at all. Obviously, component two represents an instance of overfitting, and *TOTAL* can be regarded as having anti-Q rank $L = 1$. Therefore, the single anti-Q component solution with the highest VAF score (.8924) was chosen as reference structure, against which the *FEMALE* and *MALE* matrices were fit by confirmatory anti-Q decomposition (utilizing the MATLAB routine `arobfittm.m`), with IP constraints derived from the object ordering detected when locating the uni-componential reference structure. The VAF scores as defined by (6.8) — for *FEMALE* = .8802 and *MALE* = .7341 — indicate a satisfactory fit for females, whereas the male data match the reference anti-Q

Table 6.7: VAF scores for the unidimensional unfolding (UU) and two-mode ultrametric tree (UT) representations of the fitted anti-Q components of *TOTAL*, *FEMALE* and *MALE*.

Source	VAF(anti-Q)	VAF(UU)	VAF(UT)
<i>TOTAL</i>	.8924	.8916	.6967
<i>FEMALE</i>	.8802	.8930	.7459
<i>MALE</i>	.7341	.8613	.6838

structure only mediocely. Table 6.7 reports the VAF scores obtained when fitting unidimensional unfolding and two-mode ultrametric tree structures to the anti-Q components estimated for *TOTAL*, *FEMALE* and *MALE*; the VAF scores correspond to normalizations of the least-squares loss criteria in (6.6) and (6.7) by $\sum_r \sum_c (q_{rc} - \bar{q})^2$, or $\sum_r \sum_c (q_{(s)rc} - \bar{q}_{(s)})^2$ for the female and male sources; for all three anti-Q components the unidimensional unfolding provides superior fit in comparison with the two-mode ultrametric tree structure.

Figure 6.7 presents the unidimensional unfolding and two-mode ultrametric tree graphs obtained for the anti-Q component of *TOTAL* as well as for its confirmatory counterparts of *FEMALE* and *MALE*; the two-mode ultrametric dendrogram nodes have been re-arranged to match the joint order of criteria and contraceptive measures associated with the reference anti-Q component as closely as possible (recall that for a fixed ultrametric structure, there exist $2^{(N-1)}$ different ways of positioning the terminal N object nodes of the tree diagram). The unfolding of the reference component **Q** of *TOTAL* orders the fifteen contraceptive methods along a continuum, with opposing poles of ‘behavioral’ versus ‘surgical’ measures. In borrowing terminology from Hubert, Arabie, and Meulman (2001b, p. 110), the segments of this continuum, starting from the left, can be characterized as “lottery” (Rhy, Wit), ad hoc/ubiquitously available (Dou, Con, Foa), “alternative (or no) sexual behavior” (Abs, Ora), “medical/female related” (Dia, Spe, Pil, Iud), “medical/out-patient surgical” (Vas, Tub), and “medical/surgical” (Hys, Abo). The location of the criteria points S, A, C, and E span a scale ranging from (S)afety (in the sense of non-invasive)/immediate (A)vailability to (E)ffectiveness (through invasive surgical measures), with (C)onvenience approximately defining the midpoint. The corresponding two-mode ultrametric tree diagram categorizes the contraceptive measures into three groups: behavioral, medical/surgical and Abo, forming a group by itself as it can, indeed, barely be considered a contraceptive method. As one might expect perception to relate anything in tablet form or any foreign object placed inside the body with pharmaceutical preparations, the medical character of Pil and Iud is emphasized by placing them into the same group with the surgical methods. The two-mode tree diagram, however, displays inferior fit as it contains several ties, indicated by horizontal bars joining more than two

Table 6.8: VAF scores for the unidimensional unfolding (UU) and two-mode ultrametric tree (UT) representations of the fitted anti-Q components from the independent analysis of *FEMALE* and *MALE*.

Source	VAF(anti-Q)	VAF(UU)	VAF(UT)
<i>FEMALE</i>	.8947	.8823	.7603
<i>MALE</i>	.8141	.8644	.7898

branches at a time. Ties can occur as a result of remedying violations of the ultrametric condition through averaging the involved distances — in other words, conformity to the given anti-Q component could only be attained through tied distance estimates (see Pil and Iud; Dou, Con with the S-Wit-Rhy-A cluster).

In comparing the (confirmatory) unfolding representations for females and males, the differential spread of points along the axis suggests that both sexes differ markedly in their perception of the contraceptive methods. In particular, the segment of behavioral methods (Wit/Rhy to Spe, Pil, Iud) receives a finer grained evaluation by females than males. The two-mode ultrametric tree diagrams provide additional evidence for such an interpretation: the frequent ties for the behavioral measures observed with the male diagram indicate a lack of differentiation in their judgments. Underscoring their different perception from that of women, men do not include Pil and Iud with the surgical methods, while at the same time Hys is moved into its own cluster, receiving a singular position much like Abo.

To further explore the differences between females and males, their data matrices were also analyzed as independent sources through (exploratory) anti-Q decompositions (by the MATLAB routine `biarobfnd.m` with 10,000 random starts for each matrix). For both sources, the VAF scores imply that their data are exhaustively represented by a biadditive anti-Q decomposition (the VAF distributions are not reported due to space limitations), and a low anti-Q rank approximation $\mathcal{L} = 1$ is considered satisfactory: the obtained VAF scores are .8947 and .8141 for *FEMALE* and *MALE*, respectively. Table 6.8 reports the VAF scores for the unidimensional unfolding and ultrametric tree structure fit subsequently to the anti-Q components. As before, the unidimensional unfolding of the $\mathbf{Q}_{(s)}$ attains higher VAF values than the ultrametric tree representation. The graphs (see Figure 6.8) confirm the distinction into behavioral versus medical/surgical methods, with the latter considered to be most effective. The previous conclusion of the two sexes profoundly disagreeing in their evaluations of the behavioral contraceptive methods is further substantiated. The ordering of stimuli observed with the females (Figure 2, first panel) mostly coincides with the solution for *TOTAL*, with the exception of “alternative/no sexual behavior” switching positions with Foa and Con (i.e., moving the for-

mer next to the “lottery” segment). Males (Figure 2, last panel) arrange the behavior-related measures in a slightly different manner, most notable positioning Con in-between Wit and Rhy, followed by Dia, Foa, Pil, Iud, Dou, Abs, Ora and Spe, which, as one might speculate, could represent an order determined by ‘convenience in application’ in the sense of ‘least technical/preparatory effort’. The corresponding tree diagrams are more or less in line with this interpretation; again, the many ties in the male dendrogram in the mid section, comprising behavioral methods requiring ‘preparatory effort’, may be considered as an indication of an uninformed evaluation.

As an alternative to the previous independent (exploratory) analysis of the diverging perspectives of women and men, a confirmatory fitting of the *MALE* data to *FEMALE* was conducted utilizing the latter as reference decomposition (with a VAF for males equal to .6917). The second panel in Figure 2 presents the graphs of the secondary structures fit to the retrieved (confirmatory) anti-Q component for males; both graphs support the results of the previous analyses: females display greater sophistication in their judgments of the behavioral methods as opposed to males who tend to perceive most of them as almost interchangeable; both groups agree in their evaluation of the (invasive) surgical measures.

Lastly, we would like to emphasize that none of the unidimensional unfoldings produced a degenerate solution (i.e., a configuration with perfect fit, but completely uninformative structure) — in considering that unfolding models, in particular, have been plagued by degeneracy for ages, a most remarkable result.

6.3 Discussion and Conclusion

In contrast to (traditional) calculus-based data representation techniques, combinatorial data analysis attempts to construct continuous spatial or discrete non-spatial representations based on optimal permutations of the data, where ‘optimal’ is operationalized within the context of a specific method. For example, Defays (1978) demonstrates that constructing a unidimensional scale of a set of objects from their pairwise proximities can be accomplished solely by permuting the rows and columns of the data matrix such that a certain patterning among cell entries is satisfied. The desired numerical scale values can be immediately deduced from the reordered matrix. Similarly, for hierarchical clustering problems, in their more refined guise as searches for ultrametric or additive tree representations, optimal solutions correspond to a specific patterning of the input proximity matrix, obtainable by means of simultaneous row and column permutations. To summarize, combinatorial techniques for data representations rely on identifying optimal permutations of the input data.

We emphasized that within this context, order-constrained matrix decomposition attains the status of a combinatorial data analytic meta technique: the total variability of a given proximity matrix is approximated by a minimal number of order-constrained additive components, capturing distinct ‘meaningful’ (i.e., order-related) aspects of variation. Each component in turn is associated with a unique optimal order of objects that can be translated into a continuous or discrete representation. As the two models involve identical numbers of parameters and are fit through least-squares to the same data base, they are directly comparable formally in terms of fit as well as interpretability. In other words, we can immediately determine whether a given order-constrained matrix component is more suitably represented by a continuous spatial or a discrete non-spatial structure. Thus, order-constrained matrix decomposition elegantly resolves the ambiguities in assessing differential fit, particularly encountered when a proximity matrix is fit by multiple continuous or discrete structures (as a means to increase the variability accounted for) without initial decomposition. So far, order-constrained matrix decomposition was only available for two-mode data. The extension of the method to accommodate three-way data provides the data analyst with an instrument to explore complex hypotheses concerning the appropriateness of continuous or discrete stimuli representations from an interindividual as well as intraindividual perspective.

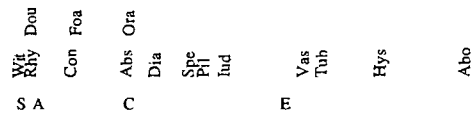
In conclusion, we would like to raise the technical issue — definitely deserving future study — whether an inconsistent, incomprehensible representation observed with a confirmatory order-constrained decomposition (see, for example, the AR representations of the data of subjects 6 and 4) provides sufficient diagnostic evidence to generally discredit the respective data set. More succinctly: is a questionable representation obtained from a confirmatory order-constrained decomposition the incidental effect of an inappropriate structural frame of reference, or does it in general hint at data of poor quality, notwithstanding the context of a confirmatory or independent analysis? We may conjecture that the analysis of individual structural differences through order-constrained decomposition is far less restrictive in its reliance on purely ordinal constraints, and therefore, is not so susceptible to masking actual inconsistencies hidden in the data by the imposition of a more rigid (continuous) reference configuration.

As a final comment, given the extreme importance of the VAF criterion in selecting the order-constrained reference decomposition as well as in assessing source-specific fit, it seems mandatory to conduct a separate large-scale simulation study to systematically investigate the behavior of the VAF measure vis-a-vis different data settings. For example, can different object orders lead to identical VAF scores? To what extent do different stimuli orders effect substantial alterations in VAF values, and so on?

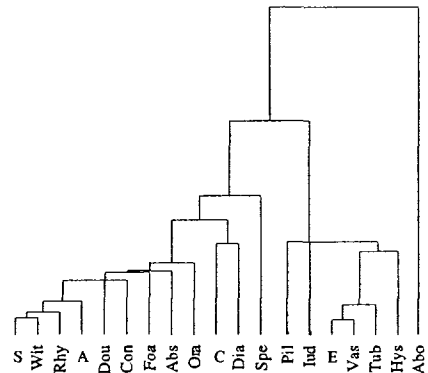
Figure 6.7: Unidimensional unfolding and two-mode ultrametric tree graphs of the reference anti-Q component extracted from *TOTAL*, with confirmatory counterparts for *FEMALE* and *MALE*.

TOTAL (VAF = .8924)

Unidimensional unfolding (VAF = .8916)

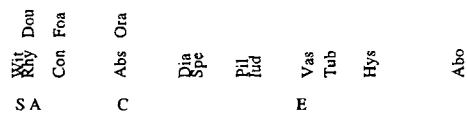


TM ultrametric tree diagram (VAF = .6967)

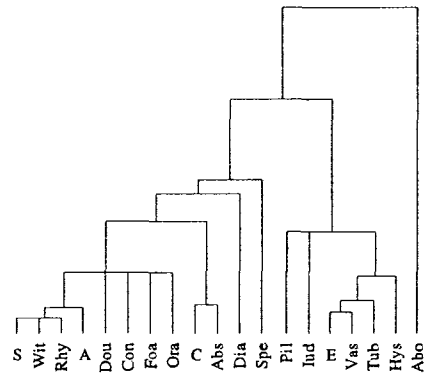


FEMALES vs. TOTAL (VAF = .8802)

Unidimensional unfolding (VAF = .8930)

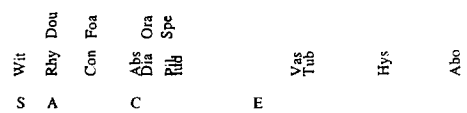


TM ultrametric tree diagram (VAF = .7459)



MALES vs. TOTAL (VAF = .7341)

Unidimensional unfolding (VAF = .8613)



TM ultrametric tree diagram (VAF = .6838)

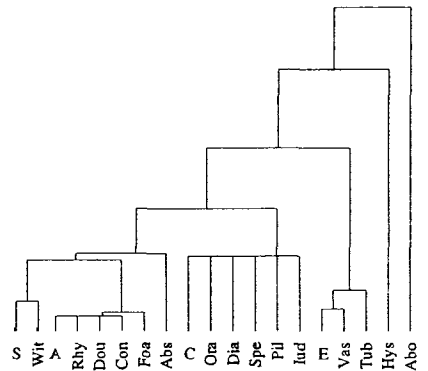
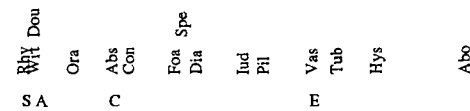


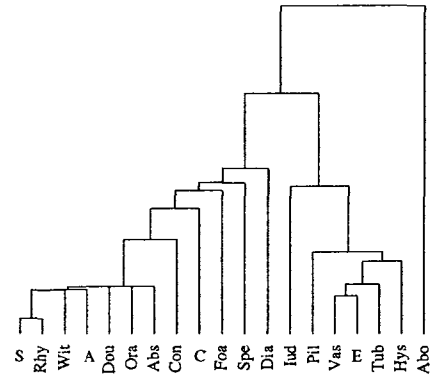
Figure 6.8: Unidimensional unfolding and two-mode ultrametric tree graphs of the anti-Q components extracted through the independent analyses of *FEMALE* and *MALE* and confirmatory anti-Q component from fitting the male data set against the female reference structure.

FEMALES (VAF = .8947)

Unidimensional unfolding (VAF = .8823)

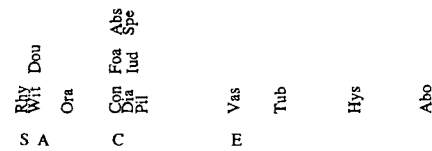


TM ultrametric tree diagram (VAF = .7603)

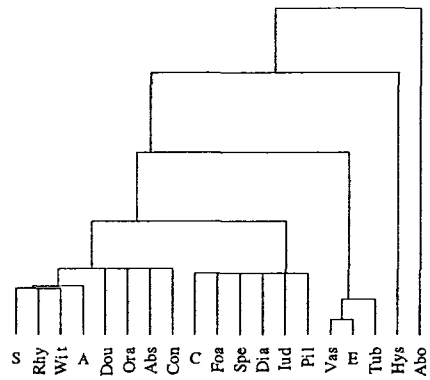


MALES vs. FEMALES (VAF = .6917)

Unidimensional unfolding (VAF = .8482)

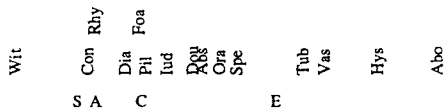


TM ultrametric tree diagram (VAF = .6798)

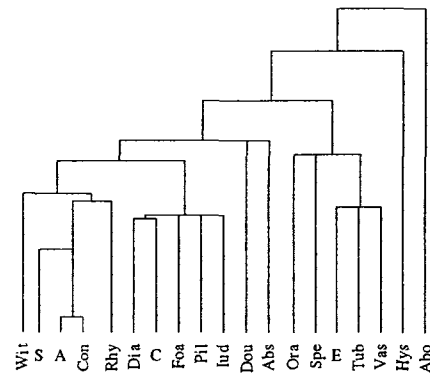


MALES (VAF = .8141)

Unidimensional unfolding (VAF = .8644)



TM ultrametric tree diagram (VAF = .7898)



7 Multiobjective Programming

Our initial interest in multiobjective programming was inspired by the question of whether it might offer an alternative to established procedures for constructing structural representations of three-way proximity data for the study of individual differences. As we will observe later on, the work of Brusco and Stahl provides definite encouragement, even though the number of matrices considered is much smaller than the typical sample size in the social sciences. The survey to follow focuses on multiobjective programming strategies as employed to solve problems in quantitative psychology. The first section summarizes a few theoretical key concepts; practical applications of multiobjective programming to optimizing psychometric test assembly and structural data representations are reviewed in the second section. Evolutionary multiobjective programming will not be discussed, because, to our knowledge, it has not been used in quantitative psychology; detailed presentations and reviews can be found elsewhere (see, Coello Coello, 1999, 2000; Coello Coello & Romero, 2002; Coello Coello, Lamont, & Van Feldhuizen, 2007; Deb, 2001).

7.1 Theory and Concepts

Decisions confronting us with the choice among alternatives serving conflicting goals are ubiquitous (not just) in daily life. Consider the numerous options we are tempted to explore when we wish to buy a new car — we might easily feel like the proverbial donkey caught between multiple haystacks. Flashy and fast cars are loved, but we are also concerned with gas economy. We value luxury amenities, but at the same time, definitely do not want to spend a fortune on a new vehicle. Unfortunately, a car with a 400 hp engine, running a thousand miles on a gallon, offering the highest level of comfort, and available for a price at \$500, does not exist. The antagonistic nature of our decision objectives poses a dilemma we are forced to resolve by compromises: less hp, but higher economy on gallons per miles; trading luxury for a lower price; and so on.

Multiple decision making, a collective term for the study of rational strategies for identifying the best possible compromise among conflicting objectives, first gained wide attention within public planning. How to deal with the extremely complex decisions involved with tasks such as securing water supplies for a large city, determining the optimal location of a power plant, or devising a land-use plan for a community? Soon, multiple decision making developed into a highly

active, interdisciplinary research area, with main contributions from mathematics and operations research, computer science and psychology. Today, the most accepted categorization of the field distinguishes between multiobjective programming and multicriteria decision analysis. The latter is concerned with developing a rational calculus for maximizing the gain of the decision maker vis-a-vis an idiosyncratic preference structure and a decision problem defined by multiple objectives with explicitly given alternatives. Multicriteria decision analysis has strong ties with psychological utility and mathematical game theory. Multiobjective programming, in contrast, represents a discipline in operations research that aims at devising efficient mathematical algorithms for identifying optimal solutions to decision problems characterized by multiple objectives. The set of feasible alternatives is only implicitly defined by constraints in the form of mathematical functions. The decision maker and preference structure are at best of secondary interest.

7.1.1 Single- versus Multiobjective Programming

The single-objective linear program in canonical form is given as:

$$[\text{SOLP}] \quad \min_{x_j \in \mathbb{R}} \left\{ y = \sum_{j=1}^n c_j x_j \right\},$$

with c_j denoting fixed coefficients and x_j real-valued decision variables, subject to a set of m linear inequality constraints

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1n}x_n &\geq b_1, \\ &\vdots \\ a_{i1}x_1 + \cdots + a_{in}x_n &\geq b_i, \\ &\vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n &\geq b_m, \end{aligned}$$

and where $x_j \geq 0 \quad \forall j = 1, \dots, n$. The $m + n$ restrictions define the set of feasible solutions (sometimes referred to as the feasible region), \mathcal{X} , from which the set of values of the decision variables is chosen minimizing the objective function, denoted $\{x_1^*, \dots, x_n^*\}$. In recalling the conversion, $\min y = \max(-y)$, we will express all mathematical programs as minimization problems and regard ‘optimization’ and ‘minimization’ as synonyms. In compact matrix notation, the SOLP is expressed as:

$$\min_{x \in \mathbb{R}^n} \{y = c'x\}$$

subject to

$$\begin{aligned} Ax &\geq b, \\ x &\geq 0. \end{aligned}$$

For succinctness, we will henceforth only use the general functional notation encompassing nonlinear programs as well (characterized by nonlinearity of the objective function and constraints):

$$[\text{SOP}] \quad \min_{\mathbf{x} \in \mathcal{X}} \{y = f(\mathbf{x})\};$$

\mathcal{X} denotes the feasible set, defined as $\mathcal{X} := \{\mathbf{x} \in \mathbb{R}_{\geq}^n : g_i(\mathbf{x}) \geq 0, i = 1, \dots, m\}$; \mathbb{R}_{\geq}^n represents the nonnegative orthant of \mathbb{R}^n , $\{\mathbf{x} \in \mathbb{R}_{\geq}^n : \mathbf{x} \geq \mathbf{0}\}$ — as we will discuss in greater detail later, $\mathbf{x} \geq \mathbf{0}$ denotes $x_j \geq 0 \ \forall j = 1, \dots, n$, with the possible inclusion of $\mathbf{x} = \mathbf{0}$ (i.e., $x_j = 0 \ \forall j = 1, \dots, n$). The functions f and g_i are commonly assumed to be continuously differentiable. The decision vector, \mathbf{x} , minimizing y is indicated by \mathbf{x}^* such that $\min\{y = f(\mathbf{x})\} = y^* = f(\mathbf{x}^*)$.

A multiobjective programming problem consists of multiple objectives, typically written as a p -dimensional vector function:

$$[\text{MOP}] \quad \min_{\mathbf{x} \in \mathcal{X}} \left\{ \mathbf{y} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_p(\mathbf{x}))' \right\}.$$

The conflicting nature of the objectives to be solved simultaneously, not their multiplicity, represents the essential characteristic of multiobjective programs; non-antagonistic objective functions would render the optimization problem trivial, because it can be condensed into a single-objective program (see Miettinen, 1999, p. 5).

7.1.2 The Notion of ‘Optimality’

Multiobjective programs represent vector optimization problems, with a far more complex structure than single-objective programs. We must emphasize, in particular, the conceptually crucial distinction between n -dimensional decision space, \mathbb{R}_{\geq}^n , embedding the feasible region as a subset, $\mathcal{X} \subseteq \mathbb{R}_{\geq}^n$, and p -dimensional objective, or criterion space, \mathbb{R}^p , that contains the image of the feasible set under $\mathbf{f} = (f_1, \dots, f_p)'$, $\mathbf{f}(\mathcal{X}) = \mathcal{Y} \subset \mathbb{R}^p$. We seek to solve the multiobjective optimization problem through identifying a feasible decision vector, $\mathbf{x}^* \in \mathcal{X}$, that will minimize the resulting criterion vector, $\mathbf{y} = \mathbf{f}(\mathbf{x}) \in \mathcal{Y}$.

Yet, the connotation of ‘minimization’, well-defined in case of a single-objective optimization problem, needs further specification within the context of multiobjective programs since a solution vector \mathbf{x}^* yielding a joint optimum for all p objectives generally does not exist. Therefore, the exact interpretation of ‘optimum’ depends on the convention adopted as to how objective function vectors, $\mathbf{y} = \mathbf{f}(\mathbf{x}) \in \mathcal{Y}$, for different alternatives $\mathbf{x} \in \mathcal{X}$ should be compared. Specifically, the evaluation of $\mathbf{y} = (f_1(\mathbf{x}), \dots, f_p(\mathbf{x}))'$, based on one objective function, $f_k(\mathbf{x}), k = 1, \dots, p$, might contradict the comparison with regard to another, $f_{k'}(\mathbf{x}), k' \neq k$. As an example, consider three objective functions, $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}))'$, and three candidate solutions,

$\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$, and $\mathbf{x}^{(3)}$. We must compare the vector quantities $\mathbf{y}^{(1)} = \mathbf{f}(\mathbf{x}^{(1)}) = (f_1(\mathbf{x}^{(1)}), f_2(\mathbf{x}^{(1)}), f_3(\mathbf{x}^{(1)}))'$, $\mathbf{y}^{(2)} = \mathbf{f}(\mathbf{x}^{(2)}) = (f_1(\mathbf{x}^{(2)}), f_2(\mathbf{x}^{(2)}), f_3(\mathbf{x}^{(2)}))'$, and $\mathbf{y}^{(3)} = \mathbf{f}(\mathbf{x}^{(3)}) = (f_1(\mathbf{x}^{(3)}), f_2(\mathbf{x}^{(3)}), f_3(\mathbf{x}^{(3)}))'$ on a component-by-component basis, a rather complex task, as we will demonstrate. First, consider the obvious case, where, say, vector $\mathbf{x}^{(1)}$ provides an unequivocally better solution than $\mathbf{x}^{(2)}$ or $\mathbf{x}^{(3)}$ if and only if $\mathbf{y}^{(1)} \leq \mathbf{y}^{(2)} \wedge \mathbf{y}^{(1)} \leq \mathbf{y}^{(3)}$, with strict inequality holding at least once for objective function f_k such that $y_k^{(1)} < y_k^{(2)} \wedge y_k^{(1)} < y_k^{(3)}$. As an illustration, with arbitrary figures and unspecified decision vectors, $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$, and $\mathbf{x}^{(3)}$, the following table presents a collection of objective function vectors, $\mathbf{y}^{(2)}$ and $\mathbf{y}^{(3)}$, all consistently inferior to $\mathbf{y}^{(1)}$:

k	$\mathbf{y}^{(1)}$	$\mathbf{y}^{(2)}$			$\mathbf{y}^{(3)}$		
1	1	2	1	1	2	2	1
2	2	2	3	2	3	2	3
3	3	3	3	4	3	4	4

More likely, however, we will observe a constellation such as

k	$\mathbf{y}^{(1)}$	$\mathbf{y}^{(2)}$	$\mathbf{y}^{(3)}$
1	2	1	2
2	2	3	3
3	3	3	1

Mathematically, the three \mathbf{y} -vectors are equally acceptable. But, the choice of the ‘best’ solution clearly poses a dilemma, as any improvement in one objective is inevitably contradicted by the decrement of another. Thus, $\mathbf{y}^{(1)}$, $\mathbf{y}^{(2)}$, and $\mathbf{y}^{(3)}$ are incomparable and no final conclusion can be drawn regarding whether $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$, or $\mathbf{x}^{(3)}$ qualifies as ‘the’ optimal solution. More to the point, unlike \mathbb{R} , the multiobjective criterion space, \mathbb{R}^p , with $p \geq 2$, lacks an ordering that would allow for the immediate identification of an unambiguous ‘best’ solution. Thus, to guide the evaluation of competing \mathbf{y} vectors (and associated decision vectors \mathbf{x}) alternative concepts must be employed that invoke a weaker notion than implied by the criterion ‘best’. Instead, we seek to identify a set of ‘most preferred’ solutions (in alluding to the hypothetical preference structure of a fictitious decision maker). As we will discuss in the next section, ‘most preferred’ in multiobjective programming is commonly operationalized through the Pareto relation that induces a (strict) partial order on the set of feasible solutions \mathcal{Y} (or, correspondingly, \mathcal{X}).

7.1.3 Efficiency and Nondominance

The French-Italian economist Pareto was among the first to address the potential dilemma arising from optimization involving conflicting objectives: “We will say that the members of a collectivity enjoy maximum ophelimity [= the capacity to satisfy a need, desire, or want] in a certain position when it is impossible to find a way of moving from that position very slightly in such a manner that the ophelimity enjoyed by each of the individuals of that collectivity increases or decreases. That is to say, any small displacement in departing from that position necessarily has the effect of increasing the ophelimity which certain individuals enjoy, and decreasing that which others enjoy, of being agreeable to some and disagreeable to others” (1896; as quoted by Ehrgott, 2005, pp. 3–4). In honor of Pareto, such a collection of objective function vectors is often termed Pareto-optimal, efficient or nondominated. We will use ‘nondominated’ only in reference to objective function vectors, $\mathbf{y} \in \mathcal{Y} \subset \mathbb{R}^p$ and reserve the attribute ‘efficient’ for a characteristic of the corresponding vectors in decision space, $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n$; ‘Pareto-optimal’ will be used solely when the distinction between \mathcal{X} and \mathcal{Y} is irrelevant.

Notation. To benefit clarity in defining the concepts of efficiency and nondominance, we use Ehrgott’s (2005) notation for vector inequalities:

1. $\mathbf{y}^{(1)} < \mathbf{y}^{(2)}$ stands for $y_k^{(1)} < y_k^{(2)} \quad \forall k = 1, \dots, p$.
2. $\mathbf{y}^{(1)} \leq \mathbf{y}^{(2)}$ indicates $y_k^{(1)} \leq y_k^{(2)}$ for $k = 1, \dots, p$; notice the possible inclusion of $\mathbf{y}^{(1)} = \mathbf{y}^{(2)}$ (i.e., $y_k^{(1)} = y_k^{(2)} \quad \forall k = 1, \dots, p$).
3. $\mathbf{y}^{(1)} \leq \mathbf{y}^{(2)}$ denotes $y_k^{(1)} \leq y_k^{(2)}$, however, $\mathbf{y}^{(1)} \neq \mathbf{y}^{(2)}$; hence, at least for one $k \in \{1, \dots, p\}$, $y_k^{(1)} < y_k^{(2)}$ must be satisfied.

As an arbitrary numerical example with $p = 4$, Figure 1 summarizes the hierarchical relationship between the sets of \mathbf{y} vectors, \mathcal{S} , determined by the different inequality operators, $\mathcal{S}_< \subset \mathcal{S}_\leq \subset \mathcal{S}_\leq$ (in fact, with $p = 4$, for the three columns in \mathcal{S}_\leq , we can generate $\binom{4}{3} = 4$, $\binom{4}{2} = 6$, and $\binom{4}{1} = 4$ different patterns, respectively, all satisfying $\mathbf{y}^{(1)} \leq \mathbf{y}^{(2)}$).

Efficiency. Ehrgott (2005) defines three different levels of efficiency:

1. Weak efficiency: a feasible solution $\hat{\mathbf{x}} \in \mathcal{X}$ is called weakly efficient if there is no $\mathbf{x} \in \mathcal{X}$ such that $\mathbf{f}(\mathbf{x}) < \mathbf{f}(\hat{\mathbf{x}})$ (i.e., $f_k(\mathbf{x}) < f_k(\hat{\mathbf{x}}) \quad \forall k = 1, \dots, p$).
2. Efficiency: a feasible solution $\hat{\mathbf{x}} \in \mathcal{X}$ is called efficient if there is no $\mathbf{x} \in \mathcal{X}$ such that $\mathbf{f}(\mathbf{x}) \leq \mathbf{f}(\hat{\mathbf{x}})$; alternatively: there is no $\mathbf{x} \in \mathcal{X}$ such that $f_k(\mathbf{x}) \leq f_k(\hat{\mathbf{x}})$ for $k = 1, \dots, p$ and no $f_{k'}(\mathbf{x}) < f_{k'}(\hat{\mathbf{x}})$ for at least one $k' \in \{1, \dots, p\}$. In words, no solution is at least as good as $\hat{\mathbf{x}}$ for all p objectives and strictly better for at least one. Observe that the subscript

Figure 7.1: Example: Notation

		\leq				$y^{(2)}$
		$<$	\leq	\leq	\leq	
$y^{(1)}$	①	2	2	2	2	2
	②	②	3	3	3	3
	③	③	③	4	4	4
	④	④	④	④	5	5

$k = 1, \dots, p$ in the first part of the alternative definition, $f_k(\mathbf{x}) \leq f_k(\hat{\mathbf{x}})$, refers to scalars; hence, different from the previous vector context, the operator \leq denotes strict inequality or equality. The second part, $f_{k'}(\mathbf{x}) < f_{k'}(\hat{\mathbf{x}})$, is necessary to impose the additional strict inequality constraint that must hold for at least one of the p objectives, $k' \in \{1, \dots, p\}$.

3. Strict efficiency: a feasible solution $\hat{\mathbf{x}} \in \mathcal{X}$ is called strictly efficient if there is no $\mathbf{x} \in \mathcal{X}$, with $\mathbf{x} \neq \hat{\mathbf{x}}$ such that $\mathbf{f}(\mathbf{x}) \leq \mathbf{f}(\hat{\mathbf{x}})$. Recall that the operator \leq includes the possibility $\mathbf{f}(\mathbf{x}) = \mathbf{f}(\hat{\mathbf{x}})$. The negation, however, further tightens the definition of strict efficiency by even precluding the possibility of candidates, $\mathbf{x} \neq \hat{\mathbf{x}}$, yielding an identical objective function vector, \mathbf{y} , such that $\mathbf{f}(\mathbf{x}) = \mathbf{f}(\hat{\mathbf{x}})$. In other words, a strictly efficient solution is usually unique.

The weakly efficient, efficient and strictly efficient sets are denoted \mathcal{X}_{wE} , \mathcal{X}_E and \mathcal{X}_{sE} , respectively; they are hierarchically nested so $\mathcal{X}_{sE} \subset \mathcal{X}_E \subset \mathcal{X}_{wE} \subset \mathcal{X} \subseteq \mathbb{R}_{\geq}^n$. The definitions of levels of efficiency of $\hat{\mathbf{x}}$ conceptually all rely on the previous identification of $\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}})$ and $\mathbf{y} = \mathbf{f}(\mathbf{x})$. The efficiency of $\hat{\mathbf{x}}$ is determined post hoc, from its image under \mathbf{f} in \mathbb{R}^p .

Nondominance. The corresponding definitions of levels of nondominance of criterion vectors, $\mathbf{y} \in \mathcal{Y}$, are given by:

1. The objective function vector $\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}}) \in \mathcal{Y}$ is called weakly nondominated if the vector $\hat{\mathbf{x}} \in \mathcal{X}$ is weakly efficient.
2. The point $\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}}) \in \mathcal{Y}$ is called nondominated if the vector $\hat{\mathbf{x}} \in \mathcal{X}$ is efficient.

The weakly nondominated and the nondominated sets are denoted \mathcal{Y}_{wN} and \mathcal{Y}_N ; they are also nested: $\mathcal{Y}_N \subset \mathcal{Y}_{wN} \subset \mathcal{Y} \subset \mathbb{R}^p$.

The table below provides illustrations of the nondominance definitions for $p = 4$, based on arbitrary numbers, with unspecified criterion vectors, \mathbf{x} and $\hat{\mathbf{x}}$. Vectors \mathbf{y} in the columns labelled “nonadmissible \mathbf{y} ” represent objective function vectors that would jeopardize the (weakly) nondominated status of

$\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}})$, whereas those listed under “admissible \mathbf{y} ” do not. The column to the right provides examples of objective function values, $\mathbf{y}^\bullet = \mathbf{f}(\mathbf{x}^\bullet)$, dominated by $\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}})$, as an illustration of the concept of dominance (see Deb, 2005) discussed later. As an important detail, observe the implication of the efficiency definition: a nondominated vector, $\hat{\mathbf{y}}$, might result from two different decision vectors, $\hat{\mathbf{x}}^{(1)}$ and $\hat{\mathbf{x}}^{(2)}$, such that $\mathbf{y} = \mathbf{f}(\hat{\mathbf{x}}^{(1)}) = \mathbf{f}(\hat{\mathbf{x}}^{(2)})$; both, $\hat{\mathbf{x}}^{(1)}$ and $\hat{\mathbf{x}}^{(2)}$, would qualify as efficient. Also, there is no such concept as ‘strict nondominance’ (even though this label is used in the table for consistency), because we cannot distinguish in \mathcal{Y} among identical $\hat{\mathbf{y}}$ derived from different $\hat{\mathbf{x}}$.

Status of $\hat{\mathbf{y}}$	non-admissible \mathbf{y}	admissible \mathbf{y}	$\hat{\mathbf{y}}$	dominated \mathbf{y}^\bullet
weakly nondominated (weakly efficient)	1	2 2 2 2	2	2 2 2 3
	2	2 3 3 3	3	3 3 4 4
	3	3 3 4 4	4	4 5 5 5
	4	4 4 4 5	5	6 6 6 6
nondominated (efficient)	1 2 2 2		2	2 2 2 3
	2 2 3 3		3	3 3 4 4
	3 3 3 4		4	4 5 5 5
	4 4 4 4		5	6 6 6 6
“strictly nondominated” (strictly efficient)	1 2 2 2 2		2	2 2 2 3
	2 2 3 3 3		3	3 3 4 4
	3 3 3 4 4		4	4 5 5 5
	4 4 4 4 5		5	6 6 6 6

The fundamental importance of efficiency and nondominance as operationalizations of the preference construct can be summarized as follows: no feasible nonefficient \mathbf{x} , no dominated \mathbf{y} can represent a most preferred solution, because there exists at least one other feasible solution $\hat{\mathbf{x}} \in \mathcal{X}$ such that $\hat{y}_k = f_k(\hat{\mathbf{x}}) \leq y_k = f_k(\mathbf{x})$ for $k = 1, \dots, p$, where strict inequality holds at least once, so that $\hat{\mathbf{x}}$ should clearly be preferred over \mathbf{x} . Hence, the concepts of efficiency and nondominance enable us to determine when exactly a feasible solution qualifies as ‘most preferred’.

Relations and Order. A preference comparison establishes a binary relation among the pairs of solutions and induces a partial order on the sets \mathcal{Y} and \mathcal{X} (in the subsequent discussion, we focus on $\mathbf{y} \in \mathcal{Y}$). In general, a partial order is characterized by certain properties that the constituting binary relation satisfies, namely, incompleteness, reflexivity, antisymmetry, and transitivity. Formally, let $\mathcal{Y} \times \mathcal{Y}$ represent the set of all possible pairs of feasible solutions $\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \dots, \mathbf{y}^{(i)}, \dots, \mathbf{y}^{(j)}, \dots$; any evaluation of elements in $\mathcal{Y} \times \mathcal{Y}$ establishes a binary relation \mathcal{R} between objective function vectors $\mathbf{y}^{(i)}$ and $\mathbf{y}^{(j)}$,

denoted $\mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(j)}$. If \mathcal{R} is derived from a preference comparison, $\mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(j)}$ is read as “ $\mathbf{y}^{(i)}$ is preferred over $\mathbf{y}^{(j)}$ ”, or, alternatively, $\mathbf{y}^{(j)} \mathcal{R} \mathbf{y}^{(i)}$ as “ $\mathbf{y}^{(j)}$ is preferred over $\mathbf{y}^{(i)}$ ”. The completeness axiom (sometimes also referred to as the linearity, totality, or connectedness axiom) requires that a relation \mathcal{R} must exist for each solution pair in the set $\mathcal{Y} \times \mathcal{Y}$ such that either $\mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(j)}$ or $\mathbf{y}^{(j)} \mathcal{R} \mathbf{y}^{(i)}$ holds; the relation is then said to possess ‘comparability’. Apparently, \mathcal{R} based on the efficiency/nondominance concept is incomplete, or ‘partial’, because no \mathcal{R} can be established for elements in \mathcal{Y}_N (or \mathcal{X}_E) due to incomparability. A relation \mathcal{R} is reflexive if $\mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(i)}$ is legitimate; in words, the underlying preference concept allows a solution to be preferred over itself. Note that reflexivity can only be postulated if \mathcal{R} is based on a ‘weak preference’ evaluation that, loosely speaking, allows for indifference towards two alternatives. The negative definition of efficiency and nondominance (i.e., “... there is no \mathbf{x} such that ...”) implies a relation \mathcal{R} interpretable as weak preference. Antisymmetry of \mathcal{R} is defined as $\mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(j)} \wedge \mathbf{y}^{(j)} \mathcal{R} \mathbf{y}^{(i)} \Rightarrow \mathbf{y}^{(i)} = \mathbf{y}^{(j)} \forall \mathbf{y}^{(i)}, \mathbf{y}^{(j)} \in \mathcal{Y}$. Lastly, if $\mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(j)} \wedge \mathbf{y}^{(j)} \mathcal{R} \mathbf{y}^{(k)} \Rightarrow \mathbf{y}^{(i)} \mathcal{R} \mathbf{y}^{(k)}$, the relation \mathcal{R} is said to be transitive.

Deb (2005), for example, gives a positive definition of the Pareto condition in terms of dominance. Consider two feasible decision vectors, $\hat{\mathbf{x}}$ and $\mathbf{x} \in \mathcal{X}$: the criterion vector $\hat{\mathbf{y}}$ dominates \mathbf{y} if $\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}}) \leq \mathbf{y} = \mathbf{f}(\mathbf{x})$ — verbally stated, solution $\hat{\mathbf{y}}$ is not inferior to \mathbf{y} in any objective, and $\hat{\mathbf{y}}$ is strictly better than \mathbf{y} in at least one objective. Different from the preference concept based on efficiency or nondominance, dominance implies ‘strict preference’ not allowing for ties, and consequently, induces a strict partial order on \mathcal{Y} possessing the following properties:

1. Incompleteness
2. Irreflexivity: dominance is not reflexive, because no objective function vector $\hat{\mathbf{y}}$ can dominate itself (recall: an objective function vector $\hat{\mathbf{y}}$ dominates \mathbf{y} if it is strictly better in at least one objective, f_k , such that $\hat{y}_k < y_k$).
3. Asymmetry: dominance implies asymmetry; either $\mathbf{y}^{(i)}$ dominates $\mathbf{y}^{(j)}$, or vice versa.
4. Transitivity: obviously, a relation based on dominance is transitive.

Domination Cones. Dominance can be represented geometrically in the criterion space \mathbb{R}^p (see Yu, 1974, 1985). Specifically, $\mathbb{R}_{\geq}^p := \{\mathbf{y} \in \mathbb{R}^p : \mathbf{y} \geq \mathbf{0}\}$, defines a cone $\mathcal{C} \subset \mathbb{R}^p$ (recall, in general, a set $\mathcal{C} \subset \mathbb{R}^p$ represents a cone if $\alpha \mathbf{y} \in \mathcal{C}$, whenever $\mathbf{y} \in \mathcal{C}$ and $\alpha \in \mathbb{R}_{\geq}$). Given \mathcal{C} , the objective function vector $\hat{\mathbf{y}} \in \mathcal{Y}$ dominates an alternative vector $\mathbf{y} \in \mathcal{Y}$ if and only if $\mathbf{y} - \hat{\mathbf{y}} \in \mathcal{C}$, or equivalently, there exists a direction $\mathbf{d} \in \mathcal{C}$, $\mathbf{d} \neq \mathbf{0}$, such that $\mathbf{y} = \hat{\mathbf{y}} + \mathbf{d}$ (note that the latter expression corresponds to a translation of \mathcal{C} such that $\hat{\mathbf{y}}$ becomes its vertex). Nondominance can be represented in a similar manner. If we define $\mathcal{C} \subset \mathbb{R}^p$ as the nonnegative orthant of \mathbb{R}^p , $\mathbb{R}_{\leq}^p := \{\mathbf{y} \in \mathbb{R}^p : \mathbf{y} \leq \mathbf{0}\}$, then $\hat{\mathbf{y}}$ is nondominated by \mathbf{y} if and only if $\mathbf{y} - \hat{\mathbf{y}} \in \mathcal{C}$.

Steuer (1986, pp. 150–154) presents a detailed description of the geometric representation of efficiency through domination cones in the decision space, \mathbb{R}^n , where $C \subset \mathbb{R}^n$ is defined as the polar cone generated by the gradients of the p objective functions (a polar cone of a cone C is the set of vectors forming angles with all vectors of C of less than or equal to 90°). For more details see Chankong and Haimes (1983, pp. 114–115), Ehrgott (2005, pp. 12–16), Miettinen (1999, pp. 23–25), and Sawaragi, Nakayama, and Tanino (1985, pp. 28–29).

7.1.4 Proper Efficiency and Nondominance

Recall that the definition of nondominated solutions explicitly precludes the possibility of improving one objective function, while retaining the same values for the others. Any improvement is always bound to come at the expense of the deterioration of at least one other criterion. The ratio of the rate of gain in one objective function, f_k , over the corresponding rate of loss in another objective function, $f_{k'}$, is often referred to as trade-off. Kuhn and Tucker (1951) were the first to observe that under certain conditions, nondominated objective function vectors, $\hat{\mathbf{y}}$, infrequently can possess unbounded trade-off ratios; this is an undesirable property, essentially preempting the definition of nondominated objective function vectors.

The narrower concept of properly nondominated solutions eliminates unbounded trade-offs between objective functions. We only present Geoffrion's (1968) definition of proper efficiency (of course, also applying to proper nondominance), as it is conceptually most closely related to the intuitive idea of establishing bounded trade-offs as a tighter selection criterion for admissible, 'proper' solutions. According to Geoffrion (1968), a feasible decision vector $\hat{\mathbf{x}} \in \mathcal{X}$ is a properly efficient solution of the multiobjective optimization problem if $\hat{\mathbf{x}} \in \mathcal{X}_E$, and if there exists $M > 0$ such that for each $k = 1, \dots, p$ and each $\mathbf{x} \in \mathcal{X}$ satisfying $f_k(\mathbf{x}) < f_k(\hat{\mathbf{x}})$ there exists a $k' \neq k$ with $f_{k'}(\mathbf{x}) > f_{k'}(\hat{\mathbf{x}})$ and

$$\frac{f_k(\hat{\mathbf{x}}) - f_k(\mathbf{x})}{f_{k'}(\mathbf{x}) - f_{k'}(\hat{\mathbf{x}})} \leq M.$$

Alternative definitions of proper efficiency have been proposed by Borwein (1977), Benson (1979), Henig (1982), and Kuhn and Tucker (1951) — for more details, the reader is encouraged to consult Ehrgott (2005, Ch. 2.4), Miettinen (1999, Ch. 2.9), or Sawaragi et al. (1985, Ch. 3.1.2).

7.1.5 Properties of Efficient and Nondominated Sets

Chankong and Haimes (1983, Ch. 4), Ehrgott (2005, Ch. 2), Miettinen (1999, Chs. 2.10 and 3), and Sawaragi et al. (1985, Chs. 3 and 4) provide very detailed theoretical presentations and discussions of the structural properties of \mathcal{Y}_N and \mathcal{X}_E , including existence theorems (i.e., conditions for \mathcal{Y}_N or $\mathcal{X}_E \neq \emptyset$) such as first-order conditions for the efficiency of $\hat{\mathbf{x}}$ (Fritz-John, Kuhn-Tucker condi-

tions). We only summarize a few key facts characterizing \mathcal{Y}_N in more practical terms.

As an intuitive result, the nondominated solutions, $\hat{\mathbf{y}} \in \mathcal{Y}_N$, must lie on the boundary of \mathcal{Y} (“southwest” rule in case of a minimization problem, see Cohon, 1978, p. 71). Lower and upper bounds, on the set of nondominated objective function vectors, \mathcal{Y}_N , indicating the range of values nondominated vectors $\hat{\mathbf{y}} \in \mathcal{Y}_N$ can attain, are given by the ideal and nadir point, respectively. The ideal point $\mathbf{y}^{(I)} = (y_1^{(I)}, \dots, y_p^{(I)})'$ of the multiobjective optimization problem is given by

$$y_k^{(I)} := \min_{\mathbf{x} \in \mathcal{X}} f_k(\mathbf{x}) = \min_{\mathbf{y} \in \mathcal{Y}} y_k.$$

Observe the minimization over \mathcal{Y} and not \mathcal{Y}_N : as $\mathcal{Y}_N \subset \mathcal{Y}$, there will always be a feasible nondominated vector, $\hat{\mathbf{y}} \in \mathcal{Y}_N$, with $\hat{y}_k = \min_{\mathbf{y} \in \mathcal{Y}} y_k$ — in other words, the (componentwise) minima of \mathcal{Y}_N and \mathcal{Y} are identical. The ideal point is computed by solving p separate single-objective optimization problems; hence, the ideal point, $\mathbf{y}^{(I)}$, is usually infeasible.

The upper bound on \mathcal{Y}_N — the worst possible, but still nondominated objective function vector — is called the nadir point, $\mathbf{y}^{(N)} = (y_1^{(N)}, \dots, y_p^{(N)})'$, and defined as

$$y_k^{(N)} := \max_{\hat{\mathbf{x}} \in \mathcal{X}_E} f_k(\hat{\mathbf{x}}) = \max_{\hat{\mathbf{y}} \in \mathcal{Y}_N} \hat{y}_k.$$

Notice that the numerical identification of $\mathbf{y}^{(N)}$ requires maximization over \mathcal{Y}_N (recall that $\max \mathcal{Y}_N \neq \max \mathcal{Y}$ as $\mathcal{Y}_N \subset \mathcal{Y}$) — “a very difficult problem. No efficient method to determine $\mathbf{y}^{(N)}$ for a general multiobjective optimization problem is known” (Ehrgott, 2005, p. 34). Pay-off tables provide a common heuristic for obtaining a basic estimate of the nadir point. Essentially, the pay-off method can be regarded as the “inverse” procedure of computing the ideal point: based on the p decision vectors, $\mathbf{x}^{*(1)}, \dots, \mathbf{x}^{*(p)}$, yielding individual minima, $y_k^* = \min_{\mathbf{x} \in \mathcal{X}} f_k(\mathbf{x})$, a square $p \times p$ pay-off table is constructed, with rows and columns representing objective functions, f_1, \dots, f_p , and optimal vectors, $\mathbf{x}^{*(1)}, \dots, \mathbf{x}^{*(p)}$, respectively. The entries along the main diagonal are given by the values y_k^* , whereas the off-diagonal cells contain the objective function values computed for a specific f_k , based on the minima vectors, $\mathbf{x}^{*(k')}$, with $k' \neq k$, the trade-off values. Subsequently, from each row, the maximal value is selected as an estimate of the upper bound of objective f_k , yielding a heuristic approximation to $\mathbf{y}^{(N)}$. Ehrgott (2005), however, cautions that pay-off tables might provide grossly misleading under- or over-estimates, when the multiobjective optimization problem involves more than two objectives.

7.1.6 Solution Methods: Generation of Pareto-Optimal Sets

The small-scale numerical examples presented earlier taught that multiobjective programs, even with just three or four objective functions easily possess

multitudinous efficient or nondominated solutions due to the vast number of possible outcome combinations. Mathematical theory provides powerful tools to determine the Pareto optimality of a candidate solution, \mathbf{x} or \mathbf{y} (for details, see the earlier references, Chankong & Haimes, 1983, Ch. 4; Ehrgott, 2005, Ch. 2; Miettinen, 1999, Chs. 2.10 and 3; Sawaragi et al., 1985, Chs. 3 and 4). From a practical point of view, the biggest challenge to any attempt at solving a multiobjective optimization problem, represents the actual generation of the sets \mathcal{Y}_N and \mathcal{X}_E — recall, mathematically, the problem is considered solved when the Pareto-optimal sets are identified. Any particular choice among these solutions as ‘the’ optimum can only be justified in reference to the (idiosyncratic) preference structure of an external decision maker.

Two categorizations of solution techniques for generating Pareto-optimal sets, \mathcal{X}_E and \mathcal{Y}_N , of multiobjective programs can be found in the literature. Miettinen (1999), for example, characterizes solution methods by the level of involvement of the (manifest) decision maker and the amount of consideration her or his idiosyncratic preference structure receives. Accordingly, ‘a priori’ methods such as goal-programming (not covered here — see Steuer, 1986, Ch. 10) require complete information about all preferences beforehand. ‘A posteriori’ techniques first attempt to generate an exhaustive set of efficient solutions; subsequently, the most suitable solution is selected in close cooperation with the decision maker. ‘Interactive’ strategies alternate between purely computational steps and consultation with the decision maker to achieve a satisfying compromise. Ehrgott (2005) distinguishes between ‘exact’ and ‘approximation’ methods, referring to the degree of accuracy of the solutions obtained. Heuristics and meta-heuristics provide appealing alternatives to exact methods, yet, with a satisfactory level of precision in situations where the exhaustive generation of Pareto-optimal sets would require prohibitive computational investments. We will follow Ehrgott’s (2005) categorization and focus on exact methods, but will briefly introduce some heuristic approaches later within the context of combinatorial multiobjective programming.

Exact Methods. The class of exact solution methods for generating \mathcal{Y}_N and \mathcal{X}_E , divides into scalarization and nonscalarization techniques. Both approaches rely on conversions of the original multiobjective optimization problem either into a (sequence of) single-objective program(s) or into a multiobjective program employing a different optimality concept. Under certain assumptions, these transformed programs will yield optimal solutions to the original multiobjective problem.

Scalarization Methods. Numerous scalarization methods have been devised; their common denominator consists in transforming the multiobjective program into a real-valued scalar function of the original objectives, solvable by traditional single-objective optimization methods. Ideally, the single-objective version of a multiobjective optimization problem should have the following properties:

1. The scalarized problem is not harder to solve than the original problem.
2. An optimal solution of the scalarization is efficient.
3. Every efficient solution of the original multiobjective program is an efficient solution of the scalarized problem for appropriately selected parameters.

We will concentrate on the weighted-sum and the ε -constraint methods as the most commonly used scalarization strategies. Important variations of scalarization techniques such as the elastic constraint approach, Benson's method, reference point or compromise approaches, including the achievement function technique and goal programming, attempting to solve the problem through minimizing the deviation from an (ideal) reference vector, will not be covered here due to space limitations. Detailed descriptions and further references can be found in Ehrgott (2005), Ehrgott and Wiecek (2005), Miettinen (1999) and Sawaragi et al. (1985).

The weighted-sum method. The original multiobjective program is re-expressed as a weighted sum of the objectives, solvable as a single-objective program:

$$[\text{WS}] \quad \min_{\mathbf{x} \in \mathcal{X}} \left\{ y = \sum_{k=1}^p \lambda_k f_k(\mathbf{x}) \right\},$$

where $\lambda \in \mathbb{R}_{\geq}^p$; usually, the coefficients are normalized such that $\sum_{k=1}^p \lambda_k = 1$ (then, the weighted sum represents a convex combination). Sometimes, the weighted-sum method is presented such that zeros as weighting coefficients are not acceptable. Indeed, at first glance, zero weights might not appear as a sensible choice, essentially indicating the inclusion of a completely insignificant objective function. On the other hand, as Miettinen (1999, p. 84) points out, zero coefficients allow for exploring potential solution changes when particular objective functions are dropped. If the coefficients are all positive or if the solution is unique, then the solution of the weighted-sum method is always Pareto-optimal, without any further assumptions (see Miettinen, 1999, p. 79). However, unless the problem is convex, the weighted-sum method is incapable of detecting all Pareto-optimal solutions — its major disadvantage, as we will discuss in more detail within the context of multiobjective combinatorial programming. If the multiobjective optimization problem is convex, then any Pareto-optimal solution can be found through the weighted-sum method by systematically altering the weighting coefficients consecutively. Observe the option to convexify a non-convex multiobjective program by raising the objective functions to a sufficient power (also known as weighted t^{th} power method). Lastly, we should emphasize the computational convenience of the weighted-sum method, as the scalarized problem is not harder to solve than the original multiobjective program. Par-

ticularly, for problems with a linear structure, parametric programming (see Chvátal, 1983, p. 162–166) offers additional computational economy.

The ε -constraint method. The multiobjective program is scalarized by selecting one of the objective functions for optimization, whereas the other $p - 1$ objectives are converted into constraints by defining an upper bound to each of them — formally

$$[C_\varepsilon] \quad \min_{\mathbf{x} \in \mathcal{X}} \left\{ y_{k'} = f_{k'}(\mathbf{x}) \right\},$$

subject to

$$f_k(\mathbf{x}) \leq \varepsilon_k \quad \forall k \in \{1, \dots, p\}, k \neq k'.$$

The solution of the ε -constraint method is always at least weakly Pareto-optimal. If one can identify ε -constraints such that the solution is unique, then it is Pareto-optimal, regardless of whether the problem is convex or not. Hence, given appropriate choices of ε_k , one can identify an efficient decision vector, $\hat{\mathbf{x}}$, by solving εC (p times) for all $f_{k'}$ objective functions. More formally, the decision vector $\hat{\mathbf{x}}$ is efficient if and only if it is a solution of C_ε for every $k' = 1, \dots, p$, where $\varepsilon_k = f_k(\hat{\mathbf{x}}) \quad \forall k \in \{1, \dots, p\}, k \neq k'$. The major problem remains the determination of the ε -constraints; as Ehrgott (2005, p. 100) concludes: the “ ε_k values are equal to the actual objective values of the efficient solution one would like to find. A confirmation or check of efficiency is obtained rather than the discovery of efficient solutions”. To summarize: theoretically, the set of Pareto-optimal solutions of a multiobjective optimization problem can be found by the ε -constraint method, but only with tremendous computational effort. Chankong and Haimes (1983) present a particularly detailed account of the ε -constraint method.

Lastly, as an interesting aside, we mention a result due to Chankong and Haimes (1983; see also Miettinen, 1999, p. 88) linking the weighted-sum and the ε -constraint method. Suppose $\hat{\mathbf{x}} \in \mathcal{X}$ is an optimal solution of WS, with corresponding weight vector $\hat{\lambda}$. If $\hat{\lambda}_{k'} > 0$, then there exists a vector $\hat{\varepsilon}$ such that $\hat{\mathbf{x}}$ is an optimal solution of objective function $f_{k'}$, while the $k = 1, \dots, p, k \neq k'$ constraints satisfy $\varepsilon_k = f_k(\hat{\mathbf{x}})$. Also, assume \mathcal{X} is convex and f_k are convex functions. If $\hat{\mathbf{x}}$ is an optimal solution of C_ε for some k' , then there exists a weight vector $\hat{\lambda} \in \mathbb{R}_{\geq}^p$ such that $\hat{\mathbf{x}}$ is an optimal solution of WS.

Nonscalarizing Methods. Instead of an explicit scalarizing function, nonscalarizing techniques rely on optimality concepts not based on the Pareto condition. We will only review two methods: the lexicographic and the max-ordering approach.

The Lexicographic Method. The lexicographic approach requires an initial ranking of the objective functions according to their importance. Subsequently, the most important objective function is minimized, subject to the original constraints. If a unique solution can be found, then the entire multiobjective opti-

mization problem is solved. Otherwise, we proceed with minimizing the second most important function, by adding a new constraint to the original restrictions to guarantee the previous most important objective function retains its optimal value. If a unique solution is identified at the second step, then it solves the original problem. Otherwise, the process continues. A lexicographically optimal solution is Pareto-optimal. Note that instead of a preference ranking, systematic permutations of the objective functions can be analyzed to benefit a broader defined solution strategy.

The Max-Ordering Method. The max-ordering approach (sometimes also referred to as “min-max”) only considers the objective function f_k that has the largest value (i.e., within a minimization context the worst value). The solution with the smallest maximum across f_1, \dots, f_p is chosen as the max-ordering optimal solution; it is weakly Pareto-optimal — in formal notation,

$$\min_{\mathbf{x} \in \mathcal{X}} \left\{ \max_{k=1, \dots, p} \{f_k(\mathbf{x})\} \right\}.$$

7.2 Applications of Multiobjective Programming in Quantitative Psychology

Multiobjective programming strategies have been used in quantitative psychology for solving combinatorial optimization problems related to automatized test assembly, and in data analytic tasks arising in the general field of classification involving the identification of structural representations such as object groupings, partitions, or sequences. The specific nature of combinatorial problems affects certain properties of the general multiobjective program encountered so far. We provide a brief discussion of these pitfalls before presenting the actual applications.

7.2.1 Combinatorial Optimization Problems

Distinct from continuous, real-valued optimization problems, combinatorial optimization is characterized by non-smooth functions, through integer (particularly, binary) restrictions placed on at least some of the decision variables. Combinatorial optimization problems are discrete, the set of feasible solutions is finite, and an optimal solution always exists, inviting the misconception that these problems are “easy” and solvable by complete enumeration. The number of feasible solutions, however, grows exponentially with problem size, and nice optimality conditions are not in existence to verify, analytically, guaranteed optimality. Instead, discrete optimization problems require the explicit or implicit search of the entire solution space to locate the global optimum. Even for small-scale problems, the computational effort of an exhaustive enumeration of all feasible solutions is prohibitive. Because of this, most current algorithms for solving combinatorial optimization problems of substantial size are heuristic,

with no guarantee of identifying a global optimum but often producing solutions at least within a close neighborhood of the desired global optimum. Partial enumeration strategies such as dynamic programming and branch-and-bound can often provide guaranteed globally optimal solutions without the need for explicit enumeration of the entire feasible solution set but do face serious limitations on the sizes of problems that can be handled.

7.2.2 Combinatorial Multiobjective Programming

Publications on extensions of the multiobjective programming paradigm for accommodating the discrete nature of combinatorial optimization problems roughly date from within the last two decades (for reviews see Ehrgott, 2005, Ch. 8, 2006; Ehrgott & Gandibleux, 2000, 2002, 2003, 2004; Ulungu & Teghem, 1994a).

Nonsupported Efficient Solutions. Distinct from continuous multiobjective programs, combinatorial counterparts possess efficient solutions that cannot be identified by weighted-sum scalarization (due to the nonconvex nature of combinatorial problems), calling for a distinction between supported and nonsupported efficient solutions. For a formal definition, let $\hat{\mathbf{x}} \in \mathcal{X}_E$; if there exists some $\lambda \in \mathbb{R}_+^p$ such that $\hat{\mathbf{x}} \in \mathcal{X}_E$ is an optimal solution of WS, then $\hat{\mathbf{x}}$ is called a supported efficient solution and any objective function vector, $\hat{\mathbf{y}} = \mathbf{f}(\hat{\mathbf{x}})$, is called supported nondominated (the corresponding vector sets are denoted \mathcal{X}_{SE} and \mathcal{Y}_{SN}). Otherwise, $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ are called nonsupported, with the respective sets indicated by \mathcal{X}_{NE} and \mathcal{Y}_{NN} . Notice that nonsupported Pareto-optimal vectors are mathematically legitimate solutions; they typically outnumber the supported decision or criterion vectors by far, which constitutes their immense practical importance.

Proper Efficiency. As another special feature of multiobjective combinatorial programs, the distinction between efficient and properly efficient solutions disappears due to \mathcal{Y} representing a finite set. Objective function values can assume only finitely many integer values; thus, the trade-off ratio has a guaranteed bound and because the denominator can never be less than one, the required M always exists.

Solution Techniques. The weighted-sum and the ε -constraint method both have serious limitations when applied to multiobjective combinatorial programs. As already mentioned, the former suffers from the incapability of detecting nonsupported solutions — Ehrgott and Gandibleux (2002, p. 377) indignantly comment on the host of publications where “the existence of nonsupported efficient solutions was either not known or ignored”. Yet, the weighted-sum scalarization is not harder to solve than the original multiobjective program, its most advantageous feature. In contrast, ε -scalarizations of multiobjective combinatorial programs almost inevitably turn into *NP*-hard problems due to the large number of constraints (even though, in theory, this approach allows for identifying all Pareto-optimal solutions). A comprehensive presentation of scalariza-

tion techniques for multiobjective combinatorial programs is given by Ehrgott (2006). For biobjective combinatorial programs, Ulungu and Teghem (1994b) propose the two-phase method, a general framework for generating the exact set of efficient solutions. The supported efficient solutions are identified through scalarization; in the second phase, the nonsupported efficient solutions are found by problem-specific methods, using bounds, reduced costs, and so on. Adaptations of single-objective combinatorial techniques such as dynamic programming and branch-and-bound can only handle problems of limited size. In addition, branch-and-bound algorithms require tight and carefully defined bounds that in case of multiobjective combinatorial programs involve the identification of ideal or nadir points for subproblems. Given these obstacles, (meta-)heuristics such as evolutionary algorithms and neighborhood search strategies, including simulated annealing and tabu search, have become popular alternatives to traditional methods for generating \mathcal{X}_E and \mathcal{Y}_N of multiobjective combinatorial programs (for an in-depth review, see Ehrgott & Gandibleux, 2004).

7.2.3 Applications

Test Assembly Problems. The implementation of computerized adaptive tests requires the technical capacity to choose a sequence of items from an item bank that constitutes a flexibly tailored response to the estimated ability level of an individual test-taker. The selection process itself is constrained by considerations of item content as well as formal psychometric criteria such as test information, test length, and so on. Mathematical programming techniques offer a viable approach to the intricate task of selecting an optimal item sequence (see van der Linden, 2005, for a comprehensive presentation). Van der Linden (2005, Ch. 3.3.4) also, in very general terms, discusses the option to employ multiobjective optimization strategies. For details, he refers the reader to Veldkamp (1999), who investigates the performance of six multiobjective programming techniques (including the weighted-sum and ε -constraint method, goal programming, and the lexicographic and max-ordering method) in solving a biobjective automated assembly of a test for measuring two traits, restricted to a length of 25 items. The two objective functions incorporating the two traits are solved through a simulated annealing heuristic.

Identification of Structural Data Representations. Combinatorial optimization methods have been used for identifying structural representations of the relationship between row and column objects of data matrices, either in the form of partitions or sequential arrangements of objects along a continuum. Independent of the specific form of structural representation, two prototypical applications of combinatorial multiobjective programming are reported in the literature: finding a structural representation of multiple data matrices in accord with a single criterion and constructing representations of a single data matrix using multiple competing criteria.

Partitioning Problems. A most concise survey on multiobjective combinatorial optimization for constructing partitions of objects is given by Brusco and Stahl (2005a, Ch. 6). With the exception of Ferligoj and Batagelj (1992) — essentially, introducing a predecessor of later multiobjective programming heuristics — the studies reviewed by Brusco and Stahl (2005a) all employ the weighted-sum method for solving scalarizations of biobjective programs. Brusco, Cradit, and Tashchian (2003), and DeSarbo and Grisaffe (1998) present biobjective approaches to market segmentation for identifying a compromise partition in situations where incongruous cluster solutions arise from different subsets of variables that capture incommensurable aspects of the objects — in technical terms: a single criterion is fit to multiple data matrices. Brusco, Cradit, and Stahl (2002) suggest a biobjective simulated annealing k -means clustering heuristic for identifying a compromise partition; a companion algorithm performing biobjective branch-and-bound k -means clustering is introduced in Brusco and Stahl (2005a).

Brusco and Stahl (2005a), and Brusco and Cradit (2005) propose a multiobjective optimization strategy to address the practical issue of appropriate choice when a partition criterion produces numerous, but significantly different optimal solutions for a single data matrix. For example, minimization of the partition diameter index is prone to yield a plethora of optimal solutions (note that the diameter of a given data cluster is defined as the largest dissimilarity value for any pair of objects within that cluster; the partition diameter represents the maximum cluster diameter across all clusters). Brusco and his collaborators (Brusco & Cradit, 2005; Brusco & Stahl, 2005a) propose biobjective programming to support an educated choice among optimal partitions. Initially, an optimal diameter is identified using either a neighborhood search (Brusco & Cradit, 2005) or a branch-and-bound algorithm (Brusco & Stahl, 2005a). In a subsequent step, a secondary partition criterion, the within-cluster sums index, is minimized subject to the constraint that the partition diameter criterion does not deteriorate by more than a pre-determined limit. The biobjective program is solved through weighted-sum scalarization. In conclusion, we would like to observe that a lexicographic strategy might also be possible, particularly as the inclusion of the diameter constraint implies a ranking among objective criteria.

Object Seriation and Sequencing. Brusco and Stahl (2005a, Ch. 11) provide an excellent review of multiobjective programming applications to object seriation and sequencing problems. Brusco and Stahl (2001) devise bi- and tri-objective dynamic programming as well as quadratic assignment routines for detecting an optimal (simultaneous) permutation of the row and column objects of a single asymmetric proximity matrix, based on three different indices of matrix pattern. Brusco (2002c) adapts multiobjective quadratic assignment and dynamic programming for identifying a single optimal permutation for multiple proximity matrices, representing different data sources (i.e., subjects, time point, experimental conditions, and so on). Analogous to the conflicting objec-

tive functions, f_1, \dots, f_p , the individual permutation scores of each proximity matrix must be simultaneously optimized by a particular arrangement of rows and columns applied across all matrices. The paradigm is further extended by Brusco and Stahl (2005b) so as to incorporate multiple criteria and multiple data matrices. All applications of multiobjective programming to seriation and sequencing problems employ scalarizations through the weighted-sum method; the objective functions are normalized by the individual optima obtained separately for each data matrix (an option also discussed by Miettinen, 1999, p. 18). Brusco and Stahl (2005a) contains a detailed outline of replacing the dynamic programming and quadratic assignment components in the multiobjective programs by branch-and-bound implementations.

7.3 Conclusion and Outlook

As we mentioned at the beginning, our interest in multiobjective programming was initiated by the question of whether it might offer an alternative to established procedures for constructing structural representations of three-way proximity data. The work of Brusco and Stahl provides definite encouragement, even though the number of matrices considered — mostly two, with the exception of Brusco (2002c) fitting four different matrices — is much smaller than the typical sample size in the social sciences. A most notable example of traditional approaches to analyzing three-way data is offered by the INDSCAL implementation of the weighted Euclidean model for scaling individual differences by Carroll and Chang (1970). Individual variability is modelled through shrinking or extending a reference structure constructed from the entire sample. The individual configurations fit the respective data matrices as closely as possible. In a similar vein, Köhn (2006) employed a strategy for modelling three-way data, guided by a principle common in statistics as well as of immediate intuitive appeal, namely, to analyze individual variability within a deviation-from-the-mean framework. The individual proximity matrices are aggregated across sources, followed by generating a best-fitting ‘average’ representation to serve as frame of reference, against which the individual data matrices are fit in a confirmatory manner. Remarkably, the deviation-from-the-mean strategy closely resembles the scalarization of a multiobjective program through the weighted-sum method, with equal weights attached to each objective function. Recall the analogy between the structural representation of an individual data matrix and an objective function. Let $s_k = f(\mathbf{P}_k)$ denote the structural representation of the data matrix of the k^{th} observation, $k = 1, \dots, p$ (for consistency with the previous notation, here and against convention, p indicates the total sample size). Finding representations for the entire sample can be expressed as

a multiobjective program

$$\begin{bmatrix} s_1 \\ \vdots \\ s_p \end{bmatrix} = \begin{bmatrix} f(\mathbf{P}_1) \\ \vdots \\ f(\mathbf{P}_p) \end{bmatrix} \Rightarrow \mathbf{s} = \left(f(\mathbf{P}_1), \dots, f(\mathbf{P}_p) \right)',$$

subject to constraints defined by the specific nature of the desired structural representation. Notice that in multiobjective programming typically the same decision vector, \mathbf{x} , is used for all objective functions. However, in our adaptation, the same function f is applied across varying data matrices. The scalarization yields

$$s^\circ = \sum_{k=1}^p \lambda_k f(\mathbf{P}_k) = \lambda_1 f(\mathbf{P}_1) + \dots + \lambda_p f(\mathbf{P}_p).$$

We claim that s° equals $\bar{s} = f(\bar{\mathbf{P}})$, where $\bar{\mathbf{P}}$ is defined as the average proximity matrix across observations:

$$\bar{\mathbf{P}} = \frac{1}{p} \sum_{k=1}^p \mathbf{P}_k = \sum_{k=1}^p \frac{1}{p} \mathbf{P}_k.$$

Now,

$$\begin{aligned} \bar{s} &= f\left(\sum_{k=1}^p \frac{1}{p} \mathbf{P}_k\right) \\ &= f\left(\frac{1}{p} \mathbf{P}_1 + \dots + \frac{1}{p} \mathbf{P}_p\right) \\ &= \frac{1}{p} f(\mathbf{P}_1) + \dots + \frac{1}{p} f(\mathbf{P}_p) \\ &= s^\circ \quad \text{for } \lambda_k = \frac{1}{p} \quad \forall k, \end{aligned}$$

assuming f to be linear.

Of particular interest is the question whether adapting multiobjective programming to the analysis of three-way data would require the evaluation of a large set of λ weights. Specifically, as data sources represent undivisible entities, assigning fractional weights from a substantive perspective does not seem too reasonable (as opposed to standard applications involving objective functions). On the other hand, fractional λ -values might be justifiable in terms of a differential weighting scheme, indicating the level of idiosyncrasy of individual data sources.

In conclusion, $\bar{\mathbf{P}}$ could also be interpreted as representing an additional, imaginary subject. In agreement with the aforementioned analogy between the structural representation of an individual data matrix and an objective function, we regard $\bar{s} = f(\bar{\mathbf{P}})$ as the highest ranking objective, creating the option to reformulate three-way data analysis as a multiobjective program relying on a

lexicographic optimality concept. Of course, these conjectures await further empirical evaluation.

Appendix I: Guttman's (1968) Gradient-Based Algorithm

Constructing a unidimensional scale of objects O_1, \dots, O_N , based on their proximities collected into an $N \times N$ square-symmetric matrix, $\mathbf{P} = \{p_{ij}\}$, with $p_{ij} = p_{ji}$ and $p_{ii} = 0$, $1 \leq i, j \leq N$, aims at arranging the N objects along a single continuum such that the induced $N(N-1)/2$ interpoint distances between the objects approximate the proximities in \mathbf{P} optimally. The N coordinates, x_1, \dots, x_N have to be chosen such that the least squares criterion is minimized

$$L(\mathbf{x}) = \sum_{i < j} (p_{ij} - |x_j - x_i|)^2.$$

In addition, and without loss of any generality, the sum of the coordinates, $\sum_i x_i$, is restricted to zero, which does not affect the value of the loss function (i.e., any set of values x_1, \dots, x_N can be replaced by $x_1 - \bar{x}, \dots, x_N - \bar{x}$, with $\bar{x} = (1/N) \sum_i x_i$). The standard calculus approach to finding a numerical optimum consists of taking first partial derivatives of $L(\mathbf{x})$ with respect to each of the coordinates x_1, \dots, x_N .

First, we rewrite $L(\mathbf{x})$ as

$$\begin{aligned} L(\mathbf{x}) &= \sum_{i < j} (p_{ij} - |x_j - x_i|)^2 \\ &= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (p_{ij} - |x_j - x_i|)^2 \\ &= \frac{1}{2} \left(\sum_{i=1}^N \sum_{j=1}^N p_{ij}^2 + \sum_{i=1}^N \sum_{j=1}^N (|x_j - x_i|)^2 - 2 \sum_{i=1}^N \sum_{j=1}^N p_{ij} |x_j - x_i| \right). \end{aligned}$$

Notice if $x_j > x_i$, then $x_j - x_i > 0 \Rightarrow |x_j - x_i| = x_j - x_i$; also, if $x_j < x_i$, then $x_j - x_i < 0 \Rightarrow |x_j - x_i| = -(x_j - x_i) = x_i - x_j$. In addition, $(x_j - x_i)^2 =$

$(x_i - x_j)^2$. Therefore,

$$\begin{aligned}
L(\mathbf{x}) &= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N p_{ij}^2 + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (x_j - x_i)^2 - \sum_{i=1}^N \sum_{j=1}^N p_{ij} |x_j - x_i| \\
&= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N p_{ij}^2 + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (x_j^2 + x_i^2 - 2x_j x_i) - \sum_{i=1}^N \sum_{j=1}^N p_{ij} |x_j - x_i| \\
&= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N p_{ij}^2 + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N x_j^2 + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N x_i^2 - \sum_{i=1}^N \sum_{j=1}^N x_j x_i - \sum_{i=1}^N \sum_{j=1}^N p_{ij} |x_j - x_i| \\
&= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N p_{ij}^2 + \frac{1}{2} N \sum_{j=1}^N x_j^2 + \frac{1}{2} N \sum_{i=1}^N x_i^2 - \sum_{i=1}^N x_i \sum_{j=1}^N x_j - \sum_{i=1}^N \sum_{j=1}^N p_{ij} |x_j - x_i| \\
&= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N p_{ij}^2 + N \sum_{i=1}^N x_i^2 - \sum_{i=1}^N \sum_{j=1}^N p_{ij} |x_j - x_i|
\end{aligned}$$

due to

$$\sum_{i=1}^N x_i^2 = \sum_{j=1}^N x_j^2 \text{ and } \sum_{i=1}^N x_i = \sum_{j=1}^N x_j = 0.$$

Taking derivatives and equating to zero yields

$$\frac{\partial L(\mathbf{x})}{\partial x_i} = 2x_i^2 - 2 \sum_{j=1}^N p_{ij} \text{sign}(x_j - x_i) = 0,$$

with

$$\text{sign}(x_j - x_i) = \begin{cases} 1 & \text{if } x_j - x_i < 0 \\ 0 & \text{if } x_j - x_i = 0 \\ -1 & \text{if } x_j - x_i > 0 \end{cases}$$

due to the case distinction already introduced above: if $x_j > x_i$, then $x_j - x_i > 0 \Rightarrow |x_j - x_i| = x_j - x_i$; also, if $x_j < x_i$, then $x_j - x_i < 0 \Rightarrow |x_j - x_i| = -(x_j - x_i) = x_i - x_j$. Solving for x_i gives

$$x_i = \frac{1}{N} \sum_j^N p_{ij} \text{sign}(x_j - x_i),$$

the necessary condition that any optimal solution must satisfy.

Guttman (1968) suggested — as what later should become known as the gradient-based approach to scaling — the iteration

$$x_i^{(t+1)} = \frac{1}{N} \sum_j p_{ij} \text{sign}(x_j^{(t)} - x_i^{(t)})$$

that upon convergence after t iterations would yield an optimal set of coordinates x_1, \dots, x_N . Unfortunately, Guttman's conjecture proved to be untenable as the algorithm regularly displays rather erratic behavior. First, often the iter-

ative process does not converge, but cycles through a repetitive set of solutions indefinitely (see Hubert, Arabie, and Hesson-McInnis, 1992). More important, the solution for the set of coordinates, x_1, \dots, x_N , provided by ‘Guttman’s update algorithm’, as it is also referred to, is not unique. The set of coordinates, x_1, \dots, x_N , identified by the gradient procedure, obviously creates a specific order of the N objects along the continuum, which, if rows and columns of \mathbf{P} are simultaneously re-arranged accordingly, permutes the proximity matrix into ‘monotonic form’. More succinctly, iterative application of Guttman’s update algorithm transforms any given \mathbf{P} into a monotonic matrix. In general, a matrix is said to be in monotonic form if for consecutive rows i and $i+1$ the differences between the sums of row entries to the left, and right of the main diagonal are monotonically increasing. A more precise definition requires notation already introduced in Chapter 4 — recall:

$$u_i := \sum_{j=1}^{i-1} p_{ij} \quad \text{for } i \geq 2$$

and

$$v_i := \sum_{j=i+1}^N p_{ij} \quad \text{for } i < N,$$

with $u_1 = v_N = 0$. Any monotonic matrix satisfies the condition $t_1 \leq \dots \leq t_N$, with t_i defined for each row of \mathbf{P} as $t_i := \frac{1}{N}(u_i - v_i)$, hence, $\frac{1}{N}(u_1 - v_1) \leq \dots \leq \frac{1}{N}(u_N - v_N)$. Recall that the coordinates obtained through the Guttman update algorithm satisfy the necessary condition; in addition they possess the remarkable property

$$x_1 = t_1 \leq \dots \leq x_N = t_N.$$

To re-iterate, Guttman’s update algorithm transforms any given proximity matrix \mathbf{P} into monotonic form with object coordinates $x_i = t_i$ as a ‘byproduct’ of the re-ordering process. A distinctive aspect of unidimensional distances is that they are additive, and thus the triangle inequality becomes an equality for such distances. This property underlies the use of the t_i as estimates of the coordinates. However, the monotonic form of a matrix, generally, is not unique, and as a consequence neither are the obtained coordinates. Thus, for a given proximity matrix a multitude of ‘optimal’ gradient solutions exists in terms of coordinates x_1, \dots, x_N . They all fulfill the necessary condition, which alone, however, does not guarantee a global minimum of $L(\mathbf{x})$.

Defays (1978) found a surprisingly simple, and elegant solution to this problem. Essentially, we are looking for a monotone permutation of \mathbf{P} that will also turn $\sum_i t_i^2$ into a maximum. Assume that we had identified this particular permutation, which is equivalent to knowing the optimal order of objects, O_1, \dots, O_N , along the continuum. Under this condition, as Defays (1978) demonstrates, the minimization of $L(\mathbf{x})$ can be re-expressed as a least squares problem with a

closed form solution for the spacings between objects, from which the actual coordinates can be deduced. Let s_i denote the distance between objects O_i and O_j on the continuum. The least squares solution for s_i is given by

$$\begin{aligned} s_i &= \frac{1}{N} (u_i - v_i - (u_{i+1} - v_{i+1})) \\ &= \frac{1}{N} (t_i - t_{i+1}), \end{aligned}$$

which, as an aside, implies $x_i = t_i$. In addition, Defays (1978) could also verify that $L(\mathbf{x})$, indeed, is minimized by

$$\frac{1}{N} \sum_i^N (u_i - v_i)^2 = N \sum_i^N t_i^2.$$

Appendix II: Re-Expressing the Scaling Loss Function

The least-squares loss function to be minimized in unidimensional scaling of a set of N objects through the appropriate choice of coordinates x_1, \dots, x_N (or distance estimates $d_{ij} = |x_j - x_i|$), with $1 \leq i, j \leq N$ is defined as:

$$\begin{aligned} L(\mathbf{x}) &= \sum_{i < j}^N \left(p_{ij} - |x_j - x_i| \right)^2 \\ &= \sum_{i < j}^N p_{ij}^2 + \sum_{i < j}^N |x_j - x_i|^2 - 2 \sum_{i < j}^N p_{ij} |x_j - x_i| \\ \Rightarrow L(\mathbf{x}) &= \sum_{i < j}^N p_{ij}^2 + A - B. \end{aligned}$$

We prove that:

$$\begin{aligned} L(\mathbf{x}) &= \sum_{i < j}^N p_{ij}^2 + N \left(\sum_i^N x_i^2 - 2 \sum_i^N x_i t_i \right) \\ &= \sum_{i < j}^N p_{ij}^2 + N \left(\sum_i^N x_i^2 - 2 \sum_i^N x_i t_i + \sum_i^N t_i^2 - \sum_i^N t_i^2 \right) \\ &= \sum_{i < j}^N p_{ij}^2 + N \sum_i^N \left((x_i^2 + t_i^2 - 2x_i t_i) - t_i^2 \right) \\ &= \sum_{i < j}^N p_{ij}^2 + N \sum_i^N (x_i - t_i)^2 - N \sum_i^N t_i^2 \\ &= \sum_{i < j}^N p_{ij}^2 + N \left(\sum_i^N (x_i - t_i)^2 - \sum_i^N t_i^2 \right) \end{aligned}$$

As $\sum_{i < j}^N p_{ij}^2$ is obvious, we begin with

$$A = \sum_{i < j}^N |x_j - x_i|^2 = \frac{1}{2} \sum_i^N \sum_j^N |x_j - x_i|^2.$$

For utmost detail, we observe the case distinction required by the absolute value

function:

1. For $j > i$, $x_j - x_i > 0 \Rightarrow |x_j - x_i| = x_j - x_i$
2. For $j < i$, $x_j - x_i < 0 \Rightarrow |x_j - x_i| = -(x_j - x_i) = x_i - x_j$.

Thus, in breaking up the j -index accordingly (and dropping $j = i$ as $|x_j - x_i| = 0$) we write

$$\begin{aligned} A &= \frac{1}{2} \left(\sum_i^N \sum_j^{i-1} |x_j - x_i|^2 + \sum_i^N \sum_{j=i+1}^N |x_j - x_i|^2 \right) \\ &= \frac{1}{2} \left(\sum_i^N \sum_j^{i-1} (x_i - x_j)^2 + \sum_i^N \sum_{j=i+1}^N (x_j - x_i)^2 \right), \end{aligned}$$

and show

$$\begin{aligned} (x_j - x_i)^2 &\stackrel{?}{=} (x_i - x_j)^2 \\ x_j^2 + x_i^2 - 2x_jx_i &= x_i^2 + x_j^2 - 2x_ix_j \\ &= x_j^2 + x_i^2 - 2x_jx_i \\ &= (x_i - x_j)^2 \end{aligned} \quad \square$$

Hence,

$$\begin{aligned} A &= \frac{1}{2} \left(\sum_i^N \sum_j^{i-1} (x_j - x_i)^2 + \sum_i^N \sum_{j=i+1}^N (x_j - x_i)^2 \right) \\ &= \frac{1}{2} \sum_i^N \sum_j^N (x_j - x_i)^2 \\ &= \frac{1}{2} \sum_i^N \sum_j^N (x_j^2 + x_i^2 - 2x_jx_i) \\ &= \frac{1}{2} \left(\sum_i^N \sum_j^N x_j^2 + \sum_i^N \sum_j^N x_i^2 - 2 \sum_i^N \sum_j^N x_ix_j \right) \\ &= \frac{1}{2} \sum_i^N \sum_j^N x_j^2 + \frac{1}{2} \sum_i^N \sum_j^N x_i^2 - \sum_i^N \sum_j^N x_ix_j \\ &= \frac{1}{2} N \sum_j^N x_j^2 + \frac{1}{2} N \sum_i^N x_i^2 - \sum_i^N x_i \sum_j^N x_j \\ &= N \sum_i^N x_i^2 \end{aligned}$$

due to

$$\sum_j^N x_j^2 = \sum_i^N x_i^2, \text{ and } \sum_i^N x_i = \sum_j^N x_j = 0.$$

$$\therefore A = \sum_{i < j}^N |x_j - x_i|^2 = N \sum_i^N x_i^2 \quad \square$$

Given the solution for A, we conclude we need to demonstrate that

$$B = 2 \sum_{i < j}^N p_{ij} |x_j - x_i| = 2N \sum_i^N x_i t_i.$$

Now,

$$B = 2 \sum_{i < j}^N p_{ij} |x_j - x_i| = (2) \frac{1}{2} \sum_i^N \sum_j^N p_{ij} |x_j - x_i|;$$

again, for utmost detail, we proceed by first inspecting the case distinction, which, in following the same steps as for A, gives

$$\begin{aligned} B &= 2 \sum_{i < j}^N p_{ij} |x_j - x_i| \\ &= (2) \frac{1}{2} \sum_i^N \sum_j^N p_{ij} |x_j - x_i| \\ &= \sum_i^N \sum_{j=1}^{i-1} p_{ij} |x_j - x_i| + \sum_i^N \sum_{j=i+1}^N p_{ij} |x_j - x_i| \\ &= \sum_i^N \sum_{j=1}^{i-1} p_{ij} (x_i - x_j) + \sum_i^N \sum_{j=i+1}^N p_{ij} (x_j - x_i) \\ &= \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_i - \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_j + \sum_i^N \sum_{j=i+1}^N p_{ij} x_j - \sum_i^N \sum_{j=i+1}^N p_{ij} x_i \\ &= \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_i - \sum_i^N \sum_{j=i+1}^N p_{ij} x_i - \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_j + \sum_i^N \sum_{j=i+1}^N p_{ij} x_j \\ &= \sum_i^N x_i \left(\sum_{j=1}^{i-1} p_{ij} - \sum_{j=i+1}^N p_{ij} \right) - \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_j + \sum_i^N \sum_{j=i+1}^N p_{ij} x_j \\ &= \sum_i^N x_i (u_i - v_i) - \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_j + \sum_i^N \sum_{j=i+1}^N p_{ij} x_j, \text{ with } u_i = \sum_{j=1}^{i-1} p_{ij}, v_i = \sum_{j=i+1}^N p_{ij} \\ &= \sum_i^N x_i (N) t_i - \sum_i^N \sum_{j=1}^{i-1} p_{ij} x_j + \sum_i^N \sum_{j=i+1}^N p_{ij} x_j, \text{ with } t_i = \frac{1}{N} (u_i - v_i). \end{aligned}$$

More compactly, we write

$$\begin{aligned} B &= N \sum_i^N x_i t_i - a + b \\ &= N \sum_i^N x_i t_i - (a - b). \end{aligned}$$

In other words, we need to prove

$$a - b = -N \sum_i^N x_i t_i$$

to obtain the desired result

$$B = 2N \sum_i^N x_i t_i.$$

Note that one might be tempted to re-express a and b (and subsequently interchange subscripts $i \rightleftharpoons j$) through

$$a = \sum_i^N \sum_j^{i-1} p_{ij} x_j = \frac{1}{2} \sum_i^N \sum_j^N p_{ij} x_j = \frac{1}{2} \sum_j^N x_j \sum_i^N p_{ij}$$

and

$$b = \sum_i^N \sum_{j=i+1}^N p_{ij} x_j = \frac{1}{2} \sum_i^N \sum_j^N p_{ij} x_j = \frac{1}{2} \sum_j^N x_j \sum_i^N p_{ij}.$$

Both transformations are NOT legitimate as for a and b the terms x_j have differing subscript indices:

$$\begin{aligned} a: \quad j < i &\Rightarrow j = \{1, 2, 3, 4\} \quad \text{hence,} \quad 2 \sum_i^N \sum_j^{i-1} p_{ij} x_j \neq \sum_i^N \sum_j^N p_{ij} x_j \\ b: \quad j > i &\Rightarrow j = \{2, 3, 4, 5\} \quad \text{hence,} \quad 2 \sum_i^N \sum_{j=i+1}^N p_{ij} x_j \neq \sum_i^N \sum_j^N p_{ij} x_j. \end{aligned}$$

In all detail (and choosing as example $N = 5$) we observe that

$$a - b = \underbrace{\sum_i^N \sum_j^{i-1} p_{ij} x_j}_{j < i} - \underbrace{\sum_i^N \sum_{j=i+1}^N p_{ij} x_j}_{j > i} \neq 0$$

$$\begin{array}{cccccc} & & & & & \\ & & & & 2 & 3 & 4 & 5 & j \\ 0 & & & & 0 & a & b & c & d \\ a & 0 & & & & 0 & e & f & g \\ b & e & 0 & & & & 0 & n & i \\ c & f & n & 0 & & & & 0 & j \\ d & g & i & j & 0 & & & & 0 \\ \hline j & 1 & 2 & 3 & 4 & & & & \end{array}$$

$$\underbrace{}_{a_j^-} - \underbrace{}_{b_j^+}$$

Moreover, notice that the difference between the sums of p_{ij} (i.e., $a + \dots + j$) associated with a_j^- and b_j^+ equal zero, but the difference between sums of the $p_{ij}x_j$ does not! In other words, transforming a and b such that we can switch from x_j to x_i requires two ‘Cool Hand Luke’ moves; in symbolic notation:

$$\begin{aligned}
(a_j^- - b_j^+) &= (a_j^- + a_j^+ - a_j^+ - b_j^+ + b_j^- - b_j^-) \\
&= (a_j^- + a_j^+ - b_j^- - b_j^+ - a_j^+ + b_j^-) \\
&= (a - b - a_j^+ + b_j^-) \\
&= (a_i^- + a_i^+ - b_i^- - b_i^+ - a_j^+ + b_j^-) \\
&= \underbrace{(a_i^- - b_i^+)}_{=0} + \underbrace{(a_i^+ - b_i^- + b_j^- - a_j^+)}_{=0} \\
&= -N \sum_i x_i t_i + 0
\end{aligned}$$

Therefore, we proceed with

$$\begin{aligned}
a - b &= \sum_i^N \sum_j^{i-1} p_{ij} x_j - \sum_i^N \sum_{j=i+1}^N p_{ij} x_j \\
&= \sum_i^N \sum_j^{i-1} p_{ij} x_j + \sum_i^N \sum_{j=i+1}^N p_{ij} x_j - \sum_i^N \sum_{j=i+1}^N p_{ij} x_j \\
&\quad - \sum_i^N \sum_{j=i+1}^N p_{ij} x_j + \sum_i^N \sum_j^{i-1} p_{ij} x_j - \sum_i^N \sum_j^{i-1} p_{ij} x_j \\
&= \sum_i^N \sum_j^N p_{ij} x_j - \sum_i^N \sum_j^N p_{ij} x_j + \sum_i^N \sum_j^{i-1} p_{ij} x_j - \sum_i^N \sum_{j=i+1}^N p_{ij} x_j
\end{aligned}$$

The previous line confirms the two ‘Cool Hand’ substitutions are correct.

$$\begin{aligned}
i \rightleftharpoons j &= \sum_j^N \sum_i^N p_{ji} x_i - \sum_j^N \sum_i^N p_{ji} x_i + \sum_j^N \sum_i^{j-1} p_{ji} x_i - \sum_j^N \sum_{i=j+1}^N p_{ji} x_i \\
&= \sum_i^N x_i \sum_j^{i-1} p_{ji} + \sum_i^N x_i \sum_{j=i+1}^N p_{ji} - \sum_i^N x_i \sum_j^{i-1} p_{ji} - \sum_i^N x_i \sum_{j=i+1}^N p_{ji} + \\
&\quad \sum_j^N \sum_i^{j-1} p_{ji} x_i - \sum_j^N \sum_{i=j+1}^N p_{ji} x_i \\
&= - \sum_i^N x_i \left(\sum_j^{i-1} p_{ji} - \sum_{j=i+1}^N p_{ji} \right) + \\
&\quad \sum_i^N x_i \sum_j^{i-1} p_{ji} - \sum_i^N x_i \sum_{j=i+1}^N p_{ji} + \sum_j^N \sum_i^{j-1} p_{ji} x_i - \sum_j^N \sum_{i=j+1}^N p_{ji} x_i.
\end{aligned}$$

Next, we will show graphically that

$$0 = \sum_i^N x_i \sum_j^{i-1} p_{ji} - \sum_i^N x_i \sum_{j=i+1}^N p_{ji} + \sum_j^N \sum_i^{j-1} p_{ji} x_i - \sum_j^N \sum_{i=j+1}^N p_{ji} x_i.$$

or in more compact notation:

$$0 = I \quad - II \quad + III \quad - IV.$$

To illustrate, let again $N = 5$:

I	II	III	IV
i	i	i	i
0	1 0 • • • •	1 0 ○ • ○ •	0
2 ○ 0	2 0 ○ ○ ○	2 0 • ○ •	2 • 0
3 • • 0	3 0 • •	3 0 ○ •	3 • ○ 0
4 ○ ○ ○ 0	4 0 ○	4 0 •	4 • ○ • 0
5 • • • • 0	0	0	5 • ○ • ○ 0

The patterns of • and ○, indicating the directions of summation, clearly implying that I and IV as well as II and III cancel out:

$$\begin{aligned}
I & \stackrel{?}{=} IV \\
\sum_i^N x_i \sum_j^{i-1} p_{ji} & \stackrel{?}{=} \sum_j^N \sum_{i=j+1}^N p_{ji} x_i = \sum_{i=j+1}^N x_i \sum_j^N p_{ji} \\
\underbrace{\sum_i^{j-1} x_i \sum_j^{i-1} p_{ji}}_0 + \sum_{i=j+1}^N x_i \sum_j^{i-1} p_{ji} & = \sum_{i=j+1}^N x_i \sum_j^{i-1} p_{ji} + \underbrace{\sum_{i=j+1}^N x_i \sum_{j=i+1}^N p_{ji}}_0, \\
& = \sum_{i=j+1}^N x_i \sum_j^{i-1} p_{ji} + 0,
\end{aligned}$$

due to contradictions $j > i \cup j < i$ and $i > j \cup i < j$. \square

$$\begin{aligned}
II & \stackrel{?}{=} III \\
\sum_i^N x_i \sum_{j=i+1}^N p_{ji} & \stackrel{?}{=} \sum_j^N \sum_i^{j-1} p_{ji} x_i = \sum_i^{j-1} x_i \sum_j^N p_{ji} \\
\sum_i^{j-1} x_i \sum_{j=i+1}^N p_{ji} + \underbrace{\sum_{i=j+1}^N x_i \sum_{j=i+1}^N p_{ji}}_0 & = \underbrace{\sum_i^{j-1} x_i \sum_j^{i-1} p_{ji}}_0 + \sum_i^{j-1} x_i \sum_{j=i+1}^N p_{ji}, \\
\sum_i^{j-1} x_i \sum_{j=i+1}^N p_{ji} + 0 & = 0 + \sum_i^{j-1} x_i \sum_{j=i+1}^N p_{ji},
\end{aligned}$$

due to contradictions $i > j \cup i < j$ and $i < j \cup i > j$. \square

Hence,

$$\begin{aligned}
a - b &= - \sum_i^N x_i \left(\sum_j^{i-1} p_{ji} - \sum_{j=i+1}^N p_{ji} \right) \\
&= - \sum_i^N x_i (u_i - v_i) \\
&= - \sum_i^N x_i (N) t_i \\
&= -N \sum_i^N x_i t_i.
\end{aligned}$$

Substituting gives

$$\begin{aligned}
L(\mathbf{x}) &= \sum_{i < j}^N p_{ij}^2 + A - B \\
&= \sum_{i < j}^N p_{ij}^2 + N \sum_i^N x_i^2 - \left(N \sum_i^N x_i t_i - (a - b) \right) \\
&= \sum_{i < j}^N p_{ij}^2 + N \sum_i^N x_i^2 - \left(N \sum_i^N x_i t_i - \left(-N \sum_i^N x_i t_i \right) \right) \\
&= \sum_{i < j}^N p_{ij}^2 + N \sum_i^N x_i^2 - 2N \sum_i^N x_i t_i \\
&= \sum_{i < j}^N p_{ij}^2 + N \left(\sum_i^N x_i^2 - 2 \sum_i^N x_i t_i \right)
\end{aligned}$$

□

Appendix III: Defays' (1978) Short Note on a Method of Seriation

The following section summarizes the key results of Defays's (1978) presentation, in a more elaborate manner. In particular, detailed mathematical derivations are given, often illustrated by means of a small scale example. The data collected are represented by an $N \times N$ matrix $\mathbf{P} = \{p_{ij}\}$ where p_{ij} is a dissimilarity index between objects O_i and O_j in a set \mathcal{O} . By assumption, \mathbf{P} is symmetric, positive, and defined on $\mathcal{O} \times \mathcal{O}$. Largest indices correspond to the most dissimilar object pairs; $p_{ii} = 0$ for all $O_i \in \mathcal{O}$. A sequencing of objects of \mathcal{O} along a continuum must be constructed such that the interobject distances d_{ij} on this continuum are not too far from the initially observed dissimilarities p_{ij} . In other words, if x_i is the coordinate of O_i on this axis, the problem is to find the coordinates which minimize the measure of loss

$$\begin{aligned} L(\mathbf{x}) &= \sum_{i=1}^{N-1} \sum_{j=i+1}^N (p_{ij} - d_{ij})^2 && \text{with } d_{ij} = |x_j - x_i| \\ &= \sum_{i=1}^{N-1} \sum_{j=i+1}^N (p_{ij} - |x_j - x_i|)^2. \end{aligned}$$

Assume for now, we know the order of the objects along the continuum. Let $s_1 = d_{12}$, $s_2 = d_{23}$, ..., $s_i = d_{ij}$, ..., $s_{N-1} = d_{N-1,N}$. The distances d_{ij} on the continuum are linear functions of the s_i — hence, we write in matrix form

$$\mathbf{d} = \mathbf{A}\mathbf{s},$$

which can be expanded into

$$\begin{bmatrix} d_{12} \\ d_{13} \\ d_{14} \\ \vdots \\ d_{1N} \\ \hline d_{23} \\ d_{24} \\ \vdots \\ d_{2N} \\ \hline \vdots \\ d_{N-2,N-1} \\ d_{N-2,N} \\ d_{N-1,N} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 1 & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 1 & 1 & 1 & 1 & \dots & 1 & 1 \\ \hline 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 1 & 1 & 1 & \dots & 1 & 1 \\ \hline \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 \\ \hline 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} s_1 = d_{12} \\ s_2 = d_{23} \\ s_3 = d_{34} \\ \vdots \\ s_i = d_{i,i+1} \\ \vdots \\ s_{N-2} = d_{N-2,N-1} \\ s_{N-1} = d_{N-1,N} \end{bmatrix}$$

Notice that \mathbf{d} is of $\binom{N}{2} \times 1$, and \mathbf{A} is of $\binom{N}{2} \times (N-1)$, while \mathbf{s} is of $(N-1) \times 1$. Minimizing the least-squares loss function requires to identify a vector \mathbf{s} yielding a minimum for

$$L(\mathbf{x}) = \|\mathbf{p} - \mathbf{d}\|^2 = \|\mathbf{p} - \mathbf{A}\mathbf{s}\|,$$

where \mathbf{p} represents the vectorization of the upper triangle matrix of \mathbf{P} :

$$\mathbf{p}' = [p_{12} \ p_{13} \ p_{14} \ \dots \ p_{ij} \ \dots \ p_{N-1,N}].$$

We can re-express the least-square loss function as

$$\begin{aligned} \|\mathbf{p} - \mathbf{d}\|^2 &= \|\mathbf{p} - \mathbf{A}\mathbf{s}\|^2 \\ &= (\mathbf{p} - \mathbf{A}\mathbf{s})'(\mathbf{p} - \mathbf{A}\mathbf{s}) \\ &= \mathbf{p}'\mathbf{p} - \mathbf{p}'\mathbf{A}\mathbf{s} - \mathbf{s}'\mathbf{A}'\mathbf{p} + \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} \\ &= \mathbf{p}'\mathbf{p} + \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} - 2\mathbf{s}'\mathbf{A}'\mathbf{p} \end{aligned}$$

due to $(\mathbf{p}'\mathbf{A}\mathbf{s})' = \mathbf{s}'\mathbf{A}'\mathbf{p}$. Taking derivatives with respect to \mathbf{s} , and equating to zero yields the familiar set of normal equations to be solved for \mathbf{s} :

$$\begin{aligned} \frac{\partial}{\partial \mathbf{s}} (\mathbf{p}'\mathbf{p} + \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} - 2\mathbf{s}'\mathbf{A}'\mathbf{p}) &= 2\mathbf{A}'\mathbf{A}\mathbf{s} - 2\mathbf{A}'\mathbf{p} = \mathbf{0} \\ \mathbf{A}'\mathbf{A}\mathbf{s} &= \mathbf{A}'\mathbf{p} \\ \mathbf{s} &= (\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\mathbf{p}. \end{aligned}$$

For a given solution $\mathbf{s} = (\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\mathbf{p}$, it can be shown that the term $\mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s}$

minimizes $\| \mathbf{p} - \mathbf{A}\mathbf{s} \|^2$. Recall

$$\begin{aligned} \| \mathbf{p} - \mathbf{d} \|^2 &= \| \mathbf{p} - \mathbf{A}\mathbf{s} \|^2 \\ &= \mathbf{p}'\mathbf{p} + \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} - 2\mathbf{s}'\mathbf{A}'\mathbf{p}. \end{aligned}$$

We can rewrite

$$\begin{aligned} \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} &= \mathbf{s}'\mathbf{A}'\mathbf{A}(\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\mathbf{p} \\ &= \mathbf{s}'\mathbf{I}\mathbf{A}'\mathbf{p} \\ &= \mathbf{s}'\mathbf{A}'\mathbf{p}. \end{aligned}$$

Substituting yields

$$\begin{aligned} \| \mathbf{p} - \mathbf{d} \|^2 &= \| \mathbf{p} - \mathbf{A}\mathbf{s} \|^2 \\ &= \mathbf{p}'\mathbf{p} + \mathbf{s}'\mathbf{A}'\mathbf{p} - 2\mathbf{s}'\mathbf{A}'\mathbf{p} \\ &= \mathbf{p}'\mathbf{p} - \mathbf{s}'\mathbf{A}'\mathbf{p} \\ &= \mathbf{p}'\mathbf{p} - \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} \end{aligned}$$

due to $\mathbf{A}'\mathbf{p} = \mathbf{A}'\mathbf{A}\mathbf{s}$.

Defays (1978) gives general expressions for $(\mathbf{A}'\mathbf{A})^{-1}$, and $\mathbf{A}'\mathbf{p}$:

$$(\mathbf{A}'\mathbf{A})^{-1} = \frac{1}{N} \begin{bmatrix} 2 & -1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \\ \vdots & & & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & -1 & 2 \end{bmatrix}$$

For the general representation of $\mathbf{A}'\mathbf{p}$, Defays (1978) introduces as notation:

$$\begin{aligned} p_{i\cdot} &:= \sum_{j>i} p_{ij} \quad \forall i \\ p_{\cdot j} &:= \sum_{i<j} p_{ij} \quad \forall j. \end{aligned}$$

More explicitly we can write

$$\begin{aligned} p_{i\cdot} &= \sum_{j=i+1}^N p_{ij} \quad \forall i = \{1, 2, \dots, N\} \\ p_{\cdot j} &= \sum_{i=1}^j p_{ij} \quad \forall j = \{1, 2, \dots, N\}. \end{aligned}$$

Now, by interchanging i and j we can rewrite the last expression as

$$\begin{aligned} p_{\cdot j} &= \sum_{i=1}^j p_{ij} \quad \forall j = \{1, 2, \dots, N\} \\ p_{\cdot i} &= \sum_{j=1}^i p_{ji} \quad \forall i = \{1, 2, \dots, N\} \\ p_{\cdot i} &= \sum_{j=1}^i p_{ij} \quad \forall i = \{1, 2, \dots, N\} \quad \text{due to } d_{ji} = d_{ij}. \end{aligned}$$

In words, $p_{i\cdot}$ is the sum of the entries within the i^{th} row of matrix \mathbf{P} from the main diagonal to the extreme right, whereas $p_{\cdot i}$ denotes the sum from the extreme left up to the main diagonal of \mathbf{P} . Thus, Defays' (1978) notation is equivalent to that introduced in Chapter 4:

$$\begin{aligned} p_{i\cdot} &\Leftrightarrow v_i := \sum_{j=i+1}^N p_{ij} \quad \text{for } i < N \\ p_{\cdot j} = p_{\cdot i} &\Leftrightarrow u_i := \sum_{j=1}^{i-1} p_{ij} \quad \text{for } i \geq 2. \end{aligned}$$

Defays (1978) arrives at the following compact expression for $\mathbf{A}'\mathbf{p}$:

$$\mathbf{A}'\mathbf{p} = \begin{bmatrix} (p_{1\cdot} - p_{\cdot 1}) \\ (p_{1\cdot} - p_{\cdot 1}) + (p_{2\cdot} - p_{\cdot 2}) \\ (p_{1\cdot} - p_{\cdot 1}) + (p_{2\cdot} - p_{\cdot 2}) + (p_{3\cdot} - p_{\cdot 3}) \\ \vdots \\ (p_{1\cdot} - p_{\cdot 1}) + (p_{2\cdot} - p_{\cdot 2}) + (p_{3\cdot} - p_{\cdot 3}) + \dots + (p_{(N-2)\cdot} - p_{\cdot (N-2)}) \\ (p_{1\cdot} - p_{\cdot 1}) + (p_{2\cdot} - p_{\cdot 2}) + (p_{3\cdot} - p_{\cdot 3}) + \dots + (p_{(N-2)\cdot} - p_{\cdot (N-2)}) + (p_{(N-1)\cdot} - p_{\cdot (N-1)}) \end{bmatrix}$$

In case of $i = 1$, for instance, $(p_{i\cdot} - p_{\cdot i})$ expands into

$$(p_{1\cdot} - p_{\cdot 1}) = (p_{12} + p_{13} + p_{14} + \dots + p_{1N}) - 0,$$

because p_{01} does not exist, and has been set to zero. For $i = 3$, as another example, $(p_{3\cdot} - p_{\cdot 3})$ yields

$$(p_{34} + p_{35} + p_{36} + \dots + p_{3N}) - (p_{13} + p_{23}).$$

In addition, notice that the last row of the vector $\mathbf{A}'\mathbf{p}$ reduces to $p_{\cdot N}$. Also, observe that $p_{\cdot 1} = p_{\cdot N} = 0$. Using terms $p_{i\cdot}$ and $p_{\cdot i}$, Defays (1978) also derives general formulae for

$$s_i = \frac{1}{N} [p_{i\cdot} - p_{\cdot i} - (p_{(i+1)\cdot} - p_{\cdot (i+1)})] \quad \text{for } i = 1, \dots, N-1,$$

and

$$\mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} = \frac{1}{N} \sum_{i=1}^N (p_{i\cdot} - p_{\cdot i})^2.$$

Observe that $|p_{i\cdot} - p_{\cdot i}| = |u_i - v_i|$ (i.e., $p_{i\cdot} - p_{\cdot i}$ and $u_i - v_i$ simply have reversed signs). With regard to the above expression of $\mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s}$, we would like to point out that generally

$$\begin{aligned} (a - b)^2 &= (b - a)^2 \\ &= (-a + b)^2 \\ &= [(-1)(a - b)]^2 \\ &= 1(a - b)^2 \\ &= (a - b)^2. \end{aligned}$$

Therefore, $(p_{i\cdot} - p_{\cdot i})^2 = (p_{\cdot i} - p_{i\cdot})^2$. Hence, by interchanging the order of $p_{i\cdot}$ and $p_{\cdot i}$ in $(p_{i\cdot} - p_{\cdot i})^2$ to $(p_{\cdot i} - p_{i\cdot})^2$ we have derived

$$N^2 t_{\rho(i)}^2 = (u_{\rho(i)} - v_{\rho(i)})^2,$$

as it is given, for example in Hubert et al. (2002), with

$$\begin{aligned} u_{\rho(i)} &= \sum_{j=1}^{i-1} p_{\rho(i)\rho(j)} \quad \text{for } i \geq 2 \\ v_{\rho(i)} &= \sum_{j=i+1}^N p_{\rho(i)\rho(j)} \quad \text{for } i < N. \end{aligned}$$

A small Scale Example With $N = 4$

Let us assume the following dissimilarities have been collected on $N = 4$ objects

$$\mathbf{P} = \begin{bmatrix} 0.0 & .2 & .1 & .4 \\ .2 & 0.0 & .3 & .2 \\ .1 & .3 & 0.0 & .5 \\ .4 & .2 & .5 & 0.0 \end{bmatrix}.$$

By vectorizing the upper triangle matrix of \mathbf{P} we obtain

$$\mathbf{p}' = [.2 \ .1 \ .4 \ .3 \ .2 \ .5].$$

Matrix \mathbf{A} has the form

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix};$$

thus, $\mathbf{A}'\mathbf{A}$ equals

$$\mathbf{A}'\mathbf{A} = \left[\begin{array}{ccc|ccc} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{array} \right] \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix},$$

and

$$(\mathbf{A}'\mathbf{A})^{-1} = \begin{bmatrix} .5 & -.25 & 0.0 \\ -.25 & .5 & -.25 \\ 0.0 & -.25 & .5 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}.$$

For $\mathbf{A}'\mathbf{p}$ we obtain

$$\mathbf{A}'\mathbf{p} = \left[\begin{array}{ccc|ccc} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{array} \right] \begin{bmatrix} .2 \\ .1 \\ .4 \\ .3 \\ .2 \\ .5 \end{bmatrix} = \begin{bmatrix} .2 + .1 + .4 \\ .1 + .4 + .3 + .2 \\ .4 + .2 + .5 \end{bmatrix} = \begin{bmatrix} .7 \\ 1.0 \\ 1.1 \end{bmatrix}.$$

In employing the general formula for $\mathbf{A}'\mathbf{p}$ in terms of $(p_{i.} - p_{.i})$, we can re-write the *RHS* vector of the previous equation. The $p_{i.}$ and $p_{.i}$ are given by

$p_{i\cdot}$	$p_{\cdot i}$
$p_{1\cdot} = .2 + .1 + .4 = .7$	$p_{\cdot 1} = 0.0$
$p_{2\cdot} = .3 + .2 = .5$	$p_{\cdot 2} = .2$
$p_{3\cdot} = .5$	$p_{\cdot 3} = .1 + .3 = .4$
$p_{4\cdot} = 0.0$	$p_{\cdot 4} = .4 + .2 + .5 = 1.1,$

yielding

$$\begin{bmatrix} .2 + .1 + .4 \\ .1 + .4 + .3 + .2 \\ .4 + .2 + .5 \end{bmatrix} = \begin{bmatrix} .2 + .1 + .4 - 0 \\ (.2 + .1 + .4 - 0) + (.3 + .2 - .2) \\ (.2 + .1 + .4 - 0) + (.3 + .2 - .2) + [.5 - (.1 + .3)] \end{bmatrix}.$$

Finally, as an estimate of $\mathbf{s} = (\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\mathbf{p}$, we obtain

$$\hat{\mathbf{s}} = \begin{bmatrix} .5 & -.25 & 0.0 \\ -.25 & .5 & -.25 \\ 0.0 & -.25 & .5 \end{bmatrix} \begin{bmatrix} .7 \\ 1.0 \\ 1.1 \end{bmatrix} = \begin{bmatrix} .1 \\ .05 \\ .3 \end{bmatrix},$$

resulting in the following estimate, $\mathbf{d} = \mathbf{A}\mathbf{s}$

$$\mathbf{d}' = (\mathbf{A}\mathbf{s})' = [.1 \ .15 \ .45 \ .05 \ .35 \ .3],$$

yielding a value of $L(\mathbf{x}) = \|\mathbf{p} - \mathbf{d}\|^2 = .14$; and recalling that $\|\mathbf{p} - \mathbf{A}\mathbf{s}\|^2$ is minimized by $\mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s}$ we compute

$$\mathbf{p}'\mathbf{p} - \mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} = .59 - .45 = .14 = L(\mathbf{x}).$$

Lastly, we also want to verify

$$s_i = \frac{1}{N}[p_{i\cdot} - p_{\cdot i} - (p_{(i+1)\cdot} - p_{(i+1)\cdot})] \quad \text{for } i = 1, \dots, N-1,$$

and

$$\mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} = \frac{1}{N} \sum_{i=1}^N (p_{i\cdot} - p_{\cdot i})^2.$$

Compute $(p_{i\cdot} - p_{\cdot i})^2$ as

$$\begin{array}{l} \frac{(p_{i\cdot} - p_{\cdot i})^2}{(.7 - 0.0)^2 = .49} \\ (.5 - .2)^2 = .09 \\ (.5 - .4)^2 = .01 \\ (0.0 - 1.1)^2 = 1.21, \end{array}$$

which, indeed, yield

$$\mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{bmatrix} .1 \\ .05 \\ .3 \end{bmatrix},$$

and

$$\mathbf{s}'\mathbf{A}'\mathbf{A}\mathbf{s} = \left(\frac{1}{4}\right)1.8 = .45.$$

References

- [1] Aarts, E., & Korst, J. (1989). *Simulated annealing and Boltzmann machines: a stochastic approach to combinatorial optimization and neural computing*. New York: Wiley.
- [2] Arabie, P. (1991). Was Euclid an unnecessarily sophisticated psychologist? *Psychometrika*, 56, 567–587.
- [3] Arabie, P., & Hubert, L. (1992). Combinatorial data analysis. *Annual Review of Psychology*, 43, 169–203.
- [4] Arabie, P., & Hubert, L. (1996). An overview of combinatorial data analysis. In P. Arabie, L. J. Hubert, & G. de Soete (Eds.), *Clustering and classification* (pp. 5–63). River Edge, NJ: World Scientific.
- [5] Arabie, P., Carroll, J. D., & DeSarbo, W. (1987). *Three-way scaling and clustering*. Newbury Park, CA: Sage Publications.
- [6] Attneave, F. (1950). Dimensions of similarity. *American Journal of Psychology*, 63, 516–556.
- [7] Barthélemy, J. P., & Guénoche, A. (1991). *Tree and proximity representations*. Chichester: Wiley.
- [8] Bennett, J. F., & Hays, W. L. (1960). Multidimensional unfolding: Determining the dimensionality of ranked preference data. *Psychometrika*, 25, 36–48.
- [9] Benson, H. (1979). An improved definition of proper efficiency for vector maximization problems. *Journal of Mathematical Analysis and Applications*, 71, 232–241.
- [10] Borg, I., & Groenen, P. (1997). *Modern multidimensional scaling. Theory and applications*. New York: Springer.
- [11] Borg, I., & Groenen, P. (2005). *Modern multidimensional scaling. Theory and applications (2nd edition)*. New York: Springer.

- [12] Borg, I., & Bergermaier, R. (1982). Degenerationsprobleme im Unfolding und ihre Lösung [Problems of degeneracy in unfolding and their solution]. *Zeitschrift für Socialpsychologie*, 13, 287–299.
- [13] Borwein, J. (1977). Proper efficient points for maximization with respect to cones. *SIAM Journal on Control and Optimization*, 15, 57–63.
- [14] Boyle, J. P., & Dykstra, R. L. (1985). A method for finding projections onto the intersection of convex sets in Hilbert spaces. In R. L. Dykstra, R. Robertson, & F. T. Wright (Eds.), *Advances in order restricted inference (Lecture Notes in Statistics, Vol 37)* (pp. 28–47). Berlin: Springer.
- [15] Brusco, M. J. (2002a). A branch-and-bound algorithm for fitting anti-Robinson structures to symmetric dissimilarities matrices. *Psychometrika*, 67, 459–471.
- [16] Brusco, M. J. (2002b). Integer Programming methods for seriation and unidimensional scaling of proximity matrices. A review and some extensions. *Journal of Classification*, 19, 45–67.
- [17] Brusco, M. J. (2002c). Identifying a reordering of rows and columns for multiple proximity matrices using multiobjective programming. *Journal of Mathematical Psychology*, 46, 731–745.
- [18] Brusco, M. J., & Cradit, J. D. (2005). ConPar: a method for identifying groups of concordant subject proximity matrices for subsequent multi-dimensional scaling analyses. *Journal of Mathematical Psychology*, 49, 142–154.
- [19] Brusco, M. J., & Stahl, S. (2000). Using quadratic assignment methods to generate initial permutations for least-squares unidimensional scaling of symmetric proximity matrices. *Journal of Classification*, 17, 197–223.
- [20] Brusco, M. J., & Stahl, S. (2001). An interactive multiobjective programming approach to combinatorial data analysis. *Psychometrika*, 66, 5–24.
- [21] Brusco, M. J. & Stahl, S. (2005a). *Branch-and-bound applications in combinatorial data analysis*. New York: Springer.
- [22] Brusco, M. J., & Stahl, S. (2005b). Bicriterion seriation methods for skew-symmetric matrices. *British Journal of Mathematical and Statistical Psychology*, 58, 333–343.
- [23] Brusco, M. J., Cradit, J. D., & Stahl, S. (2002). A simulated annealing heuristic for a bicriterion partitioning problem in market segmentation. *Journal of Marketing Research*, 39, 99–109.

- [24] Brusco, M. J., Cradit, J. D., & Tashchian, A. (2003). Multicriterion clusterwise regression for joint segmentation settings: an application to customer value. *Journal of Marketing Research*, 40, 225–234.
- [25] Busing, F. M. T. A. (2003). *PREFSCAL - A short user's guide for version 1.0*. Unpublished manuscript, Department of Data Theory, University of Leiden, Leiden, The Netherlands.
- [26] Busing, F. M. T. A. (2006). Avoiding degeneracy in metric unfolding by penalizing the intercept. *British Journal of Mathematical and Statistical Psychology*, 59, 419–427.
- [27] Busing, F. M. T. A., Commandeur, J. J. F., & Heiser, W. J. (1997). PROXSCAL: A multidimensional scaling program for individual differences scaling with constraints. In W. Bandilla & F. Faulbaum (Eds.), *Softstat '97. Advances in statistical software*, Vol. 6 (pp. 67–74). Stuttgart, Germany: Lucius & Lucius.
- [28] Busing, F. M. T. A., Groenen, P. J. F., & Heiser, W. J. (2005). Avoiding degeneracy in multidimensional unfolding by penalizing on the coefficient of variation. *Psychometrika*, 70, 71–98.
- [29] Carroll, J. D. (1972). Individual differences and multidimensional scaling. In R. N. Shepard, A. K. Romney, & S. B. Nerlove (Eds.), *Multidimensional scaling: Theory and applications in the behavioral sciences. Vol. I: Theory* (pp. 105–155). New York: Seminar Press.
- [30] Carroll, J. D. (1976). Spatial, non-spatial and hybrid models for scaling. *Psychometrika*, 41, 439–463.
- [31] Carroll, J. D. (1980). Models and methods for multidimensional analysis of preferential choice (or other dominance) data. In E. D. Lantermann & H. Feger (Eds.), *Similarity and choice* (pp. 234–289). Bern: Huber.
- [32] Carroll, J. D., & Arabie, P. (1980). Multidimensional scaling. *Annual Review of Psychology*, 31, 607–649.
- [33] Carroll, J. D., & Arabie, P. (1998). Multidimensional scaling. In M. Birnbaum (Ed.), *Measurement, judgment and decision making. Handbook of perception and cognition* (2nd ed.) (pp. 179–250). San Diego: Academic Press.
- [34] Carroll, J. D., & Chang, J. J. (1970). Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition. *Psychometrika*, 35, 283–320.

- [35] Carroll, J. D., & Pruzansky, S. (1980). Discrete and hybrid scaling models. In: E. Lantermann & H. Feger (Eds.), *Similarity and choice* (pp. 108–139). Bern: Huber.
- [36] Carroll, J. D., Clark, L. A., & DeSarbo, W. S. (1984). The representation of three-way proximities data by single and multiple tree structure models. *Journal of Classification*, 1, 25–74.
- [37] Carroll, J. D., & Wish, M. (1974). Models and methods for three-way multidimensional scaling. In D. H. Krantz, R. C. Atkinson, R. D. Luce, & P. Suppes (Eds.), *Contemporary developments in mathematical psychology. Vol. II* (pp. 57–105). San Francisco: Freeman.
- [38] Chankong, V., & Haimes, Y. Y. (1983). *Multiobjective decision making theory and methodology*. New York: Elsevier Science.
- [39] Chvátal, V. (1983). *Linear programming*. New York: Freeman.
- [40] Coello Coello, C. A. (1999). A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems*, 1, 208–214.
- [41] Coello Coello, C. A. (2000). An updated survey of GA-based multi-objective optimization techniques. *ACM Computing Surveys*, 32, 109–143.
- [42] Coello Coello, C. A., & Romero, C. E. M. (2002). Evolutionary algorithms and multiple objective optimization. In M. Ehrgott & X. Gandibleux (Eds.), *Multiple criteria optimization: state of the art annotated bibliographic surveys (International Series in Operations Research and Management Science, Vol 52)* (pp. 277–331). Boston: Kluwer.
- [43] Coello Coello, C. A., Lamont, G., & Van Veldhuizen, D. (2007). *Evolutionary algorithms for solving multi-objective problems (2nd Ed.)*. New York: Springer.
- [44] Cohon, J. L. (1978). *Multiobjective programming and planning*. New York: Academic Press.
- [45] Commandeur, J. J. F., & Heiser, W. J. (1993). *Mathematical derivations in the proximity scaling (PROXSCAL) of symmetric data matrices. Tech. Rep. No. RR-93-03*. Leiden, The Netherlands: Department of Data Theory, Leiden University.
- [46] Coombs, C. H. (1950). Psychological scaling without a unit of measurement. *Psychological Review*, 57, 148–158.

- [47] Coombs, C. H. (1964). *A theory of data*. New York: Wiley.
- [48] Coombs, C. H., & Kao, R. C. (1960). On a connection between factor analysis and multidimensional unfolding. *Psychometrika*, 25, 219–231.
- [49] Coppi, R., & Bolasco, S. (Eds.) (1989). *Multiway data analysis*. Amsterdam: North Holland.
- [50] Corter, J. E. (1982). ADDTREE/P: A PASCAL program for fitting additive trees based on Sattath and Tversky's ADDTREE algorithm. *Behavior Research Methods and Instrumentation*, 14, 353–354.
- [51] Corter, J. E. (1996). *Tree Models of Similarity and Association*. Thousand Oaks, CA: Sage.
- [52] Cox, T. F., & Cox, M. A. A. (2001). *Multidimensional scaling* (2nd Ed.). Boca Raton: Chapman and Hall.
- [53] Davison, M. L. (1983). *Multidimensional scaling*. New York: Wiley.
- [54] Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Chichester: Wiley.
- [55] Deb, K. (2005). Multi-objective optimization. In E. K. Burke & G. Kendall (Eds.), *Search methodologies* (pp. 273–316). New York: Springer.
- [56] Defays, D. (1978). A short note on a method of seriation. *British Journal of Mathematical and Statistical Psychology*, 3, 49–53.
- [57] De Leeuw, J. (1983). *On degenerate nonmetric unfolding solutions* (Tech. Rep.). Department of Data Theory, FSW/RUL (Leiden University, Holland).
- [58] De Leeuw, J., & Heiser, W. (1977). Convergence of correction-matrix algorithms for multidimensional scaling. In J. C. Lingoes, E. E. Roskam, & I. Borg (Eds.), *Geometric representations of relational data* (pp. 735–752). Ann Arbor, MI: Mathesis Press.
- [59] De Leeuw, J., & Heiser, W. J. (1982). Theory of multidimensional scaling. In P. R. Krishnaiah & L. N. Kanal (Eds.), *Handbook of statistics*, Vol. 2 (pp. 285–316). Amsterdam: North Holland.
- [60] DeSarbo, W. S., & Carroll, J. D. (1981). Three-way metric unfolding. *Proceedings of the 3rd ORSA/TIMS special interest conference on market measurement*, 157–183.

- [61] DeSarbo, W. S., & Carroll, J. D. (1985). Three-way metric unfolding via alternating weighted least squares. *Psychometrika*, 50, 275–300.
- [62] DeSarbo, W. S., & Grisaffe, D. (1998). Combinatorial optimization approaches to constrained market segmentation: an application to industrial market segmentation. *Marketing Letters*, 9, 115–134.
- [63] De Soete, G. (1983). A least-squares algorithm for fitting additive trees to proximity data. *Psychometrika*, 48, 621–626.
- [64] De Soete, G., & Carroll, J. D. (1989). Ultrametric tree representations of three-way three-mode data. In R. Coppi & S. Belasco (Eds.), *Analysis of multiway data matrices* (pp. 415–426). Amsterdam: North Holland.
- [65] De Soete, G., & Carroll, J. D. (1992). Probabilistic multidimensional models of pairwise choice data. In F. G. Ashby (Ed.), *Multidimensional models of perception and cognition* (pp. 61–88). Hillsdale, NJ: Erlbaum.
- [66] De Soete, G., & Carroll, J. D. (1996). Tree and other network models for representing proximity data. In P. Arabie, L. J. Hubert, & G. De Soete (Eds.), *Clustering and classification* (pp. 157–197). River Edge, NJ: World Scientific.
- [67] De Soete, G., DeSarbo, W. S., Furnas, G. W., & Carroll, J. D. (1984). The estimation of ultrametric and path length trees from rectangular proximity data. *Psychometrika*, 49, 289–310.
- [68] Deutsch, F. (2001). *Best approximation in inner product spaces*. New York: Springer.
- [69] Dodge, Y. (Ed.) (1997). *L₁-Statistical procedures and related topics*. Hayward, CA: Institute of Mathematical Statistics.
- [70] Dykstra, R. L. (1983). An algorithm for restricted least squares regression. *Journal of the American Statistical Association*, 78, 837–842.
- [71] Ehrgott, M. (2005). *Multicriteria optimization (2nd Ed.)*. Berlin: Springer.
- [72] Ehrgott, M. (2006). A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operations Research*, 147, 343–360.
- [73] Ehrgott, M., & Gandibleux (2000). A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spektrum*, 22, 425–460.

- [74] Ehrgott, M., & Gandibleux (2002). Multiobjective combinatorial optimization. In M. Ehrgott & X. Gandibleux (Eds.), *Multiple criteria optimization: state of the art annotated bibliographic surveys (International Series in Operations Research and Management Science, Vol 52)* (pp. 369–444). Boston: Kluwer.
- [75] Ehrgott, M., & Gandibleux (2003). Multiple objective combinatorial optimization — a tutorial. In T. Tanino, T. Tanaka, & M. Inuiguchi (Eds.), *Multi-objective programming and goal-programming* (pp. 3–18). Berlin: Springer.
- [76] Ehrgott, M., & Gandibleux (2004). Approximative solution methods for multiobjective combinatorial optimization. *TOP*, 12, 1–88.
- [77] Ehrgott, M., & Wiecek, M. M. (2005). Multiobjective programming. In J. Figueira, S. Greco, & M. Ehrgott (Eds.), *Multiple criteria decision analysis: state of the art surveys (International Series in Operations Research and Management Science, Vol 78)* (pp. 667–722). New York: Springer.
- [78] Ferligoj, A., & Batagelj, V. (1992). Direct multicriteria clustering algorithms. *Journal of Classification*, 9, 43–61.
- [79] Fichet, B. (1987). The role played by L_1 in data analysis. In Y. Dodge (Ed.), *Statistical data analysis based on the L_1 norm and related methods* (pp. 185–193). Amsterdam: North Holland.
- [80] Furnas, G. W. (1980). Objects and their features: the metric representation of two class data. Unpublished doctoral dissertation, Stanford University.
- [81] Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: a guide to the theory of NP-completeness*. San Francisco: Freeman.
- [82] Garner, W. R. (1974). *The processing of information and structure*. New York: Wiley.
- [83] Geoffrion, A. M. (1968). Proper efficiency and the theory of vector maximization. *Journal of Mathematical Analysis and Applications*, 22, 618–630.
- [84] Gifi, A. (1990). *Nonlinear multivariate analysis*. Chichester: Wiley.
- [85] Glover, F., & Laguna, M. (1993). Tabu search. In C. Reeves (Ed.), *Modern heuristic techniques for combinatorial problems* (pp. 70–141). Oxford: Blackwell.

- [86] Gold, E. M. (1973). Metric unfolding: data requirement for unique solution and clarification of Schönemann's algorithm. *Psychometrika*, 38, 555–569.
- [87] Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. New York: Addison-Wesley.
- [88] Gower, J. C., & Hand, D. J. (1996). *Biplots*. London: Chapman & Hall.
- [89] Green, P. E., & Rao, V. (1972). *Applied multidimensional scaling*. Hinsdale, IL: Dryden Press.
- [90] Greenacre, M. J. (1984). *Theory and application of correspondence analysis*. New York: Academic Press.
- [91] Greenacre, M. J., & Browne, M. W. (1986). An efficient alternating least-squares algorithm to perform multidimensional unfolding. *Psychometrika*, 51, 241–250.
- [92] Guttman, L. (1968). A general nonmetric technique for finding the smallest coordinate space for a configuration of points. *Psychometrika*, 33, 469–506.
- [93] Han, S. P. (1988). A successive projection method. *Mathematical Programming*, 40, 1–14.
- [94] Hansen, P., & Mladenovic, N. (2003). Variable neighborhood search. In F. W. Glover & G. A. Kochenberger (Eds.), *Handbook of Metaheuristics* (pp. 145–184), Norwell, MA: Kluwer.
- [95] Hansen, P., & Mladenović, N. (2005). Variable neighborhood search. In E. K. Burke & G. Kendall (Eds.), *Search methodologies* (pp. 211–238). New York: Springer.
- [96] Harshman, R. A. (1970). Foundations of the PARAFAC procedure: Models and conditions for an “explanatory” multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16, 1–84.
- [97] Harshman, R. A., & Lundy, M. E. (1984). The PARAFAC model for three-way factor analysis and multidimensional scaling. In H. G. Law, C. W. Snyder, J. A. Hattie, & R. P. McDonald (Eds.), *Research methods for multimode data analysis* (pp. 122–215). New York: Praeger.
- [98] Harshman, R. A., & Lundy, M. E. (1994). PARAFAC: Parallel factor analysis. *Computational Statistics & Data Analysis*, 18, 39–72.

- [99] Hays, W. L., & Bennett, J. F. (1961). Multidimensional unfolding: Determining configurations from complete rank order preference data. *Psychometrika*, 26, 221–238.
- [100] Heiser, W. J. (1981). *Unfolding analysis of proximity data*. Unpublished doctoral dissertation, University of Leiden, Leiden, The Netherlands.
- [101] Heiser, W. (1989a). The city-block model for three-way multidimensional scaling. In R. Coppi & S. Bolasco (Eds.), *Multiway data analysis* (pp. 395–404). Amsterdam: North Holland.
- [102] Heiser, W. (1989b). Order invariant unfolding analysis under smoothness restrictions. In G. De Soete, H. Feger, & K. C. Klauer (Eds.), *New developments in psychological choice modeling* (pp. 3–33). Amsterdam: North Holland.
- [103] Heiser, W. J. (2004). Geometric representation of association between categories. *Psychometrika*, 69, 513–545.
- [104] Heiser, W. J., & Busing, F. M. T. A. (2004). Multidimensional scaling and unfolding of symmetric and asymmetric proximity relations. In D. Kaplan (Ed.), *The Sage handbook of quantitative methodology for the social sciences* (pp. 25–48). Thousand Oaks, CA: Sage Publications.
- [105] Henig, M. I. (1982). Proper efficiency with respect to cones. *Journal of Optimization Theory and Applications*, 36, 387–407.
- [106] Hubert, L. J. (1987). *Assignment methods in combinatorial data analysis*. New York: Marcel Dekker.
- [107] Hubert, L. J., & Arabie, P. (1986). Unidimensional scaling and combinatorial optimization. In J. de Leeuw, J. Meulman, W. Heiser, & F. Critchley (Eds.), *Multidimensional data analysis* (pp. 181–196). Leiden, The Netherlands: DSWO Press.
- [108] Hubert, L. J., & Arabie, P. (1988). Relying on necessary conditions for optimization: Unidimensional scaling and some extensions. In H. H. Bock (Ed.), *Classification and related methods of data analysis* (pp. 463–472). Amsterdam: Elsevier.
- [109] Hubert, L., & Arabie, P. (1994). The analysis of proximity matrices through sums of matrices having (anti-)Robinson forms. *British Journal of Mathematical and Statistical Psychology*, 47, 1–40.
- [110] Hubert, L. J., & Arabie, P. (1995a). The approximation of two-mode proximity matrices by sums of order-constrained matrices. *Psychometrika*, 60, 573–605.

- [111] Hubert, L. J., & Arabie, P. (1995b). Iterative projection strategies for the least-squares fitting of tree structures to proximity data. *British Journal of Mathematical and Statistical Psychology*, 48, 281–317.
- [112] Hubert, L., & Köhn, H.-F. (2006). Lower (Anti-)Robinson Rank Representations for Symmetric Proximity Matrices. To appear in *Festschrift for Edwin Diday*, series “Studies in Classification, Data Analysis and Knowledge Optimization”, Berlin: Springer.
- [113] Hubert, L. J., Arabie, P., & Hesson-McInnis, M. (1992). Multidimensional scaling in the city-block metric: A combinatorial approach. *Journal of Classification*, 9, 211–236.
- [114] Hubert, L. J., Arabie, P., & Meulman, J. (1997). Linear and circular unidimensional scaling for symmetric proximity matrices. *British Journal of Mathematical and Statistical Psychology*, 50, 253–284.
- [115] Hubert, L. J., Arabie, P., & Meulman, J. (1998a). The representation of symmetric proximity data: dimensions and classifications. *The Computer Journal*, 41, 566–577.
- [116] Hubert, L. J., Arabie, P., & Meulman, J. (1998b). Graph-theoretic representations for proximity matrices through strongly-anti-Robinson or circular strongly-anti-Robinson matrices. *Psychometrika*, 63, 341–358.
- [117] Hubert, L. J., Arabie, P., & Meulman, J. (2001a). *Combinatorial data analysis: optimization by dynamic programming*. Philadelphia: SIAM.
- [118] Hubert, L. J., Arabie, P., & Meulman, J. (2001b). Modelling dissimilarity: Generalizing ultrametric and additive tree representations. *British Journal of Mathematical and Statistical Psychology*, 54, 103–123.
- [119] Hubert, L. J., Arabie, P., & Meulman, J. (2002). Linear unidimensional scaling in the L_2 -norm: basic optimization methods using MATLAB. *Journal of Classification*, 19, 303–328.
- [120] Hubert, L. J., Arabie, P., & Meulman, J. (2006). *The structural representation of proximity matrices with MATLAB*. Philadelphia: SIAM.
- [121] Johnson, R. A., & Wichern, D. W. (2007). *Applied Multivariate Statistical Analysis* (6th ed.). Upper Saddle River, NJ: Pearson Prentice Hall.
- [122] Kendall, D. G. (1971a). Abundance matrices and seriation in archaeology. *Zeitschrift für Wahrscheinlichkeitstheorie*, 17, 104–112.

- [123] Kendall, D. G. (1971b). A mathematical approach to seriation. *Philosophical Transactions of the Royal Society of London, Series A*, 269, 125-135.
- [124] Kiers, H. A. L. (1991). Hierarchical relations among three-way methods. *Psychometrika*, 56, 449-470.
- [125] Köhn, H.-F. (2006). Combinatorial individual differences scaling within the city-block metric. *Computational Statistics & Data Analysis*, 51, 931-946.
- [126] Krivánek, M. (1986). On the computational complexity of clustering. In E. Diday, Y. Escouffier, L. Lebart, J. P. Pagés, Y. Schektman, & R. Tomassone (Eds.), *Data analysis and information, IV* (pp. 89-96). Amsterdam: North Holland.
- [127] Kroonenberg, P. M. (1983). *Three-mode principal component analysis*. Leiden, The Netherlands: DSWO Press.
- [128] Kroonenberg, P. M. (1988). Three-mode analysis. In S. Kotz & N. L. Johnson (Eds.), *Encyclopedia of statistical sciences. Vol. 9* (pp. 231-236). New York: Wiley.
- [129] Kroonenberg, P. M. (1994). The TUCKALS line. A suite of programs for three-way data analysis. *Computational Statistics & Data Analysis*, 18, 73-96.
- [130] Kroonenberg, P. M., & De Leeuw, J. (1980). Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45, 69-97.
- [131] Kruskal, J. B. (1964a). Multidimensional scaling by optimizing goodness-of-fit to a nonmetric hypothesis. *Psychometrika*, 29, 1-27.
- [132] Kruskal, J. B. (1964b). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115-129.
- [133] Kruskal, J. B., & Carroll, J. D. (1969). Geometrical models and badness-of-fit functions. In P. R. Krishnaiah (Ed.), *Multivariate analysis. Vol. 2* (pp. 639-671). New York: Academic Press.
- [134] Kruskal, J. B., & Wish, M. (1978). *Multidimensional scaling*. Beverly Hills, CA: Sage Publications.
- [135] Krzanowski, W. J. (1988). *Principles of multivariate analysis*. Oxford: Clarendon Press.

- [136] Kuhn, H., & Tucker, A. (1951). Nonlinear Programming. In J. Neyman (Ed.), *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability* (pp. 481–492). Berkeley: University of California Press.
- [137] Miettinen, K. (1999). *Nonlinear multiobjective optimization (International Series in Operations Research and Management Science, Vol 12)*. Dordrecht: Kluwer.
- [138] Law, H. G., Snyder, C. W., Hattie, J. A., & McDonald, R. P. (Eds.) (1984). *Research methods for multimode data analysis*. New York: Praeger.
- [139] Le Calvé, G. (1987). L_1 embeddings for a data structure (I, D). In Y. Dodge (Ed.), *Statistical data analysis based on the L_1 norm and related methods* (pp. 195–202). Amsterdam: North Holland.
- [140] Lingoes, J., & Roskam, E. E. (1973). A mathematical and empirical study of two multidimensional scaling algorithms. *Psychometrika Monograph Supplement*, 38.
- [141] MacKay, D. B., & Zinnes, J. L. (1995). Probabilistic multidimensional unfolding - an anisotropic model for preference ratio judgments. *Journal of Mathematical Psychology*, 39, 99–111.
- [142] Marden, J. I. (1995). *Analyzing and modelling rank data*. London: Chapman and Hall.
- [143] Marley, A. A. J. (1992). Developing and characterizing Thurstone and Luce models for identification and preference. In F. G. Ashby (Ed.), *Multidimensional models of perception and cognition* (pp. 299–333). Hillsdale, NJ: Erlbaum.
- [144] Meulman, J., & Verboon, P. (1993). Points of view analysis revisited: fitting multidimensional structures to optimal distance components with cluster restrictions on the variables. *Psychometrika*, 58, 7–35.
- [145] Meulman, J., Heiser, W. J., & Carroll, J. D. (1986). *PREFMAP-3 users's guide*. Unpublished paper, Bell Telephone Laboratories, Murray Hill, NJ.
- [146] Meulman, J., Heiser, W. J., & SPSS (1999). *Categories*. Chicago: SPSS.
- [147] Nishisato, S. (1980). *Analysis of categorical data: dual scaling and its applications*. Toronto: University of Toronto Press.

- [148] Okada, A., & Imaizumi, T. (1980). Nonmetric method for extended INDSCAL model. *Behaviormetrika*, 7, 13–22.
- [149] Powell, M. J. D. (1977). Restart procedures for the conjugate gradient method. *Mathematical Programming*, 12, 241–254.
- [150] Pruzansky, S. (1975). *How to use SINSDCAL: A computer program for individual differences in multidimensional scaling*. Murray Hill, NJ: AT & T Bell Laboratories.
- [151] Ramsay, J. O. (1977). Maximum likelihood estimation in multidimensional scaling. *Psychometrika*, 42, 241–266.
- [152] Rendl, F. (2002). The quadratic assignment problem. In Z. Drezner & H. W. Hamacher (Eds.), *Facility location* (pp. 439–457). Berlin: Springer.
- [153] Robinson, W. S. (1951). A method for chronologically ordering archaeological deposits. *American Antiquity*, 19, 293–301.
- [154] Ross, J., & Cliff, N. (1964). A generalization of the interpoint distance model. *Psychometrika*, 29, 167–176.
- [155] Sattath, S., & Tversky, A. (1977). Additive similarity trees. *Psychometrika*, 42, 319–345.
- [156] Sawaragi, Y., Nakayama, H., & Tanino, T. (1985). *Theory of multiobjective optimization*. Orlando: Academic Press.
- [157] Schiffman, S. S., Reynolds, M. L., & Young, F. W. (1981). *Introduction to multidimensional scaling*. New York: Academic Press.
- [158] Schönemann, P. H. (1970). On metric multidimensional unfolding. *Psychometrika*, 35, 349–366.
- [159] Shepard, R. N. (1991). Integrality versus separability of stimulus dimensions: From an early convergence of evidence to a proposed theoretical basis. In J. R. Pomerantz & G. R. Lockhead (Eds.) *The perception of structure: Essays in honor of Wendell R. Garner* (pp. 53–71). Washington, DC: American Psychological Association.
- [160] Shepard, R. N., & Arabie, P. (1979). Additive clustering: representation of similarities as combinations of discrete overlapping properties. *Psychological Review*, 30, 87–123.

- [161] Smilde, A., Bro, R., & Geladi, P. (2004). *Multi-way analysis: applications in the chemical sciences*. Chichester: Wiley.
- [162] Smith, T. J. (1998). A comparison of three additive tree algorithms that rely on a least-squares loss criterion. *The British Journal of mathematical and Statistical Psychology*, 51, 269–288.
- [163] Steuer, R. E. (1986). *Multiple criteria optimization: theory, computation and application*. New York: Wiley.
- [164] Takane, Y., Young, F. W., & De Leeuw, J. (1977). Nonmetric individual differences multidimensional scaling: An alternating least-squares method with optimal scaling features. *Psychometrika*, 42, 7–67.
- [165] Ten Berge, J. M. F., De Leeuw, J., & Kroonenberg, P. M. (1987). Some new results on principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 52, 183–191.
- [166] Tucker, L. R. (1960). Intra-individual and inter-individual multidimensionality. In H. Gulliksen & S. Messick (Eds.), *Psychological scaling: Theory and applications* (pp. 155–167). New York: Wiley.
- [167] Tucker, L. R. (1964). The extension of factor analysis to three-dimensional matrices. In N. Frederiksen & H. Gulliksen (Eds.), *Contributions to mathematical psychology* (pp. 109–127). New York: Holt, Rinehart, and Winston.
- [168] Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31, 279–311.
- [169] Tucker, L. R. (1972). Relations between multidimensional scaling and three-mode factor analysis. *Psychometrika*, 37, 3–27.
- [170] Tucker, L. R., & Messick, S. (1963). An individual differences model for multidimensional scaling. *Psychometrika*, 28, 333–367.
- [171] Tversky, A. (1977). Features of similarity. *Psychological Review*, 84, 327–352.
- [172] Tversky, A., & Krantz, D. (1969). Similarity of schematic faces: a test of interdimensional additivity. *Perception and Psychophysics*, 5, 124–128.
- [173] Ulungu, E. L., & Teghem, J. (1994a). Multi-objective combinatorial optimization problems: a survey. *Journal of Multi-Criteria Decision Analysis*, 3, 83–104.

- [174] Ulungu, E. L., & Teghem, J. (1994b). The two-phase method: an efficient procedure to solve biobjective combinatorial optimization problems. *Foundations of Computing and Decision Sciences*, 20, 149–165.
- [175] Van Blokland-Vogelzang, A. W. (1989). Unfolding and consensus ranking: A prestige ladder for technical occupations. In G. De Soete, H. Feger, & K. C. Klauer (Eds.), *New developments in psychological choice modelling* (pp. 237–258). Amsterdam: North-Holland.
- [176] Van Blokland-Vogelzang, A. W. (1993). A nonparametric distance model for unidimensional unfolding. In M. A. Fligner & J. S. Verducci (Eds.), *Probability models and statistical analyses for ranking data* (pp. 241–276). New York: Springer.
- [177] van der Linden, W. J. (2005). *Linear models for optimal test design*. Springer: New York.
- [178] Veldkamp, B. P. (1999). Multiple objective test assembly problems. *Journal of Educational Measurement*, 36, 253–266.
- [179] Weller, S. C., & Romney, A. K. (1990). *Metric scaling: Correspondence analysis*. Newbury Park, CA: Sage Publications.
- [180] Winsberg, S., & De Soete, G. (1993). A latent class approach to fitting the weighted Euclidean model, CLASCAL. *Psychometrika*, 58, 315–330.
- [181] Yu, P. (1974). Cone convexity, cone extreme points and nondominated solutions in decision problems with multiobjectives. *Journal of Optimization Theory and Applications*, 14, 319–377.
- [182] Yu, P. (1985). *Multiple criteria decision making: concepts, techniques and extensions*. New York: Plenum.
- [183] Zinnes, J. L., & Griggs, R. A. (1974). Probabilistic multidimensional unfolding analysis. *Psychometrika*, 39, 327–350.

Vita

Education

University of Illinois at Urbana-Champaign, Illinois

Ph.D. Quantitative Psychology, Mai 2007

Thesis: *The Structural Representation of Three-Way Proximity Data*

Advisor: Professor Lawrence J. Hubert

M.S. Statistics 2000

University of Hamburg, Germany

M.S. (Diplom) Psychology 1988

Thesis: *On the Neglect of Learned Base Rates in Judgments Under Uncertainty*

Advisors: Professor Kurt Pawlik, Professor Frank Rösler

Department of Aviation and Space Psychology, DLR German Aerospace Center, Hamburg, Germany

Student Trainee, 1986–88 (parallel to my studies at the University of Hamburg)

University of Cologne, Germany

B.A. (Vordiplom) Psychology 1984

Professional Nonacademic Positions Held

RJ Reynolds Tobacco, Cologne, Germany

Executive Marketing Research Northern Europe, 1998

BBDO Germany, Düsseldorf, Germany

Director Marketing Research, 1997–98

Manager Marketing Research, 1991–96

Marketing Researcher, 1990

Trainee Marketing Research, 1989