The Structural Representation of Proximity Matrices With MATLAB

Contents

List of Tables

List of Figures

Preface

As the title of this monograph implies, our main goal is to provide and illustrate the use of functions (by way of m-files) within a MATLAB[1] computational environment to effect a variety of structural representations for proximity information assumed available on a set of objects. The structural representations that will be of interest have been discussed and developed primarily in the applied (behavioral science) statistical literature (e.g., in psychometrics and classification), although interest in these topics has now extended much more widely (for example, to bioinformatics and chemometrics). We subdivide the monograph into three main sections depending on the general class of representations being discussed. Part I will develop linear and circular uni- and multi-dimensional scaling using the city-block metric as the major representational device; Part II is concerned with characterizations based on various graph-theoretic tree structures, and specifically with those usually referred to as ultrametrics and additive trees; Part III uses representations defined solely by order properties, and particularly to what are called (strongly) anti-Robinson forms. Irrespective of the part of the monograph being discussed, there generally will be two kinds of proximity information analyzed: one-mode and two-mode. One-mode proximity data are defined between the n objects from a *single* set, and usually given in the form of a square ($n \times n$) symmetric matrix with a zero main diagonal; two-mode proximity data are defined between the objects from two distinct sets containing, say, n_a and n_b objects, respectively, and given in the form of a rectangular ($n_a \times n_b$) matrix. Also, there will generally be the flexibility to allow the fitting (additively) of multiple structures to either the given one- or two-mode proximity information.

It is not the intent of the monograph to present formal demonstrations of the various assertions we might make along the way, such as for the convergence of a particular algorithm or approach. All of this is generally available in the literature (and much of it by the authors of the current monograph), and the references to this source material is given when appropriate. The primary interest here is to present and demonstrate how to actually find and fit these structures computationally with the help of some sixty-five functions (though m-files) we provide that are usable within a MATLAB computational environment. The usage header information for each of these functions is given in Appendix A (listed alphabetically). The m-files themselves can be downloaded individually from

[1]MATLAB is a registered trademark of The MathWorks, Inc.

http://cda.psych.uiuc.edu/srpm_mfiles

Also, there is a "zipped" file called `srpm_mfiles.zip` at this site that includes them all, as well as the few small data sets used throughout the monograph to illustrate the results of invoking the various m-files (or we might say, invoking the various "m-functions"); thus, the reader should be able to reproduce all of the examples given in the monograph (assuming, obviously, access to a MATLAB environment).

The computational approach implemented in the provided m-files for obtaining the sundry representations, are by choice, invariably least-squares, and based on what is called the Dykstra-Kaczmarz (DK) method for solving linear inequality constrained least-squares tasks. The latter iterative strategy is reviewed in Chapter 1 (Section 1.4, in particular). All of the representations of concern (over all three monograph Parts) can be characterized by explicit linear inequalities; thus, once the latter constraints are known (by, for example, the identification of certain object permutations through secondary optimization problems such as quadratic assignment), the actual representing structure can be obtained by using the iterative DK strategy. Also, as we will see particularly in Part II dealing with graph-theoretic tree structures (ultrametrics and additive trees), the DK approach can even be adopted heuristically to first identify the inequality constraints that we might wish to impose in the first place. And once identified in this exploratory fashion, a second application of DK could then do a confirmatory fitting of the now fixed inequality constraints.

As noted above, our purpose in writing this monograph is to provide an applied documentation source for a collection of m-files that would be of interest to applied statisticians and data analysts but also accessible to a notationally sophisticated but otherwise substantively focused user. Such a person would typically be most interested in analyzing a specific data set by adopting one (or some) of the structural representations we discuss. The background we have tried to assume is at the same level required to follow the documentation for good, commercially available optimization subroutines, such as the Numerical Algorithms Group (NAG) Fortran subroutine library, or at the level of one of the standard texts in applied multivariate analysis usually used for a graduate second-year methodology course in the behavioral and social sciences. An excellent example of the latter would be the widely used text now in its fifth edition by Johnson and Wichern (2002). Draft versions of the current monograph have been used as supplementary material for a course relying on the latter text as the primary reference.

The research reported in this monograph has been partially supported by the National Science Foundation through Grant No. SES-981407 (to LH), and by the Netherlands Organization for Scientific Research (NWO) through Grant No. 575-67-053 for the 'PIONEER' project 'Subject Oriented Multivariate Analysis' (to JM).

Lawrence Hubert
Phipps Arabie
Jacqueline Meulman

September, 2004