Multivariate Behavioral Research, 1988, 23, 281-290
Computer Software

Aspects of Person-Machine Communication in Structural Modeling of Correlations and Covariances

James H. Steiger
University of British Columbia

Analysis of covariance or correlation structure is characterized, unfortunately, by repetitive suboptimal communication between persons and/or computer programs. After analyzing some aspects of this suboptimality, I suggest some approaches to improving the situation.

With the continuing growth of power/cost ratio in desk-top computers, there are very few worthwhile mainframe statistical programs which will not be available in personal computer versions. For example, consider the area of linear structural relations modeling (hereafter referred to as LSRM), which is the main object of my discussion here. Low cost versions of a number of LSRM programs are already available for the IBM microcomputer family. With increasing competition, further improvements in this field can be expected. One area where improvement should be most noticeable is improvement in the quality and ease of use of user interfacing. Peter Bentler's (1985) EQS has already made some important strides in this area.

However, progress in the field of structural modeling has been uneven. In particular, during the past 3 years, while I was teaching a course in correlational methods and structural modeling, it became increasingly clear to me that there were serious communication problems, at several levels, in the LSRM literature. While not peculiar to LSRM papers, these problems appeared in them with alarming frequency.

In discussing these problems there is no need to refer directly to the papers where they surfaced. I am not claiming to have optimal solutions either. (I have developed a structural modeling program called EzPATH which alleviates some of the problems discussed in this paper, and which expands on some of the solutions I propose. The program will be available later this year, but I will not describe it here.)

Structural modeling, like most scientific activity, involves a number of paths for information flow. These paths involve persons and computer programs. There are many possible taxonomic schemes for describing these information paths. The one I shall use, based on a 2x2 table with Person and Computer as the marginal entries on each side, is perhaps most obvious.

Person-Person Communication

Replicability is as essential an aspect of structural modeling as it is of any other scientific endeavor. Many researchers, attempting to evaluate LSRM research, begin by attempting to replicate the model-fitting results in the published paper. Therefore Person-Person communication in LSRM depends at the most fundamental level on how well a research paper answers basic questions like the following:

1) What were the data?

2) How many subjects were there, and what are the demographic details?

3) What computer program was used (name and version)?

4) What, precisely, were the models used in each analysis? What sequence of models was analyzed? How many models were tested and rejected prior to the final model?

5) What starting values were used in the iterative solution?

6) Which computational options were employed?

7) What difficulties, or anomalies, were encountered during the analytic process?

That these questions are key ones would seem undeniable. Reliable replication is impossible without clear answers to most of them. Replication of computed results, even if it did occur, would be difficult to evaluate without key demographic details. Yet many causal modeling papers appear without answers to at least one of these questions.

For example I am frustrated when I read an interesting causal modeling paper and the covariance matrix is not printed. Most causal modeling papers involve analysis of at most 20 manifest variables, and so it would be reasonable (often easy) to include the covariance matrix in the published article. Since re-analysis is impossible without the covariance matrix, it is difficult to understand why authors fail to include it in the published article. Perhaps they are defensive about the models they have published, or perhaps the editor guards journal space too jealously. My opinion is that the gain in accessibility of a paper to close examination makes publication of the covariance matrix (or matrices) not only worthwhile, but almost imperative.

Some LSRM discussions, (including at least one case where the covariance matrix was presented) have failed to mention the number of subjects on which the covariance matrix was based.

Many LSRM papers give ambiguous or incorrect descriptions of the models analyzed. For example, one paper never mentioned that several manifest variables were excluded from the final version of an analysis. In several papers, there were apparent discrepancies between the path diagram, as drawn, and the analysis, as performed. Reconciliation in such cases often involves hours of trial-and-error experimentation. In one such case, I was never able to reproduce the results presented in the manuscript.

Many papers rely on a path diagram to convey the structural model being tested. Such diagrams convey the ideas behind causal models in a particularly natural way, so this is not surprising. Unfortunately path diagrams, despite their advantages, have not been entirely adequate (in all but the simplest cases) for unambiguous communication of structural models. There are several reasons for this.

One is the absence of clear standards for construction of such diagrams. For example, some authors (apparently) consider disturbance terms to be uncorrelated and always present by default. Others consider them absent by default. Often the implicit rules under which the authors are operating are never stated -- they can only be deduced by trying several models, and comparing them to the results presented in the papers. This ambiguity, when coupled with the inconvenience and poor design of some existing computer programs, can result in substantial waste of time and effort.

For example, consider the path diagram segment shown in Figure 1. Such segments appear in numerous published diagrams, and are actually ambiguous. There are at least two possible interpretations, which can actually lead to two different results with some data sets. One interpretation is that the manifest variable y can be written as

[1] $$Y = fX + D, \text{ where } \mathrm{Var}(D) = e$$

An alternative interpretation is that

[2] $$Y = fX + kD, \text{ where } \mathrm{Var}(D) = 1, \text{ and } e=k^2$$

To the casual observer it might appear that the two interpretations are the same. Both versions include a segment (appearing after the "+" sign) involving a latent variable, and the entire segment has a variance of e.

However, some experienced LSRM practitioners know that there is an important difference. Most structural modeling programs find their solution by performing unconstrained optimization with an iterative routine. The

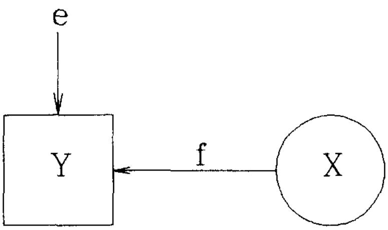

Figure 1. Typical Segment from a Path Diagram.

solution found by the iterative routine for the unknowns e (in Equation 1) and k (in Equation 2) may assign negative values to these unknowns.

The model expressed in Equation 2 forces the structural modeling program to yield a non-negative value for e, even if k is negative. The causal modeling program might yield a negative value for e if Equation 1 is fitted. Figures 2 and 3 show how a more explicit diagrammatic representation can distinguish between the two representations.

In this representation, all latent variables are represented in circles, and manifest (observed) variables are in rectangles. All exogenous variables have variances of 1, unless stated otherwise. All structural coefficients are 1, unless stated otherwise. "Undirected" (variance-covariance) relations are indicated by lines with no arrowheads (rather than a line segment with arrowheads at each end), to keep the display less cluttered. Figure 2 corresponds to Equation 1, while Figure 3 corresponds to Equation 2.

Another recurring problem, which would exist even if coherent standards for path diagrams were adopted, is that the diagrams are required to convey too much information in too small a space. Capabilities of graphic artists are frequently overtaxed as a consequence, and it is not at all unusual

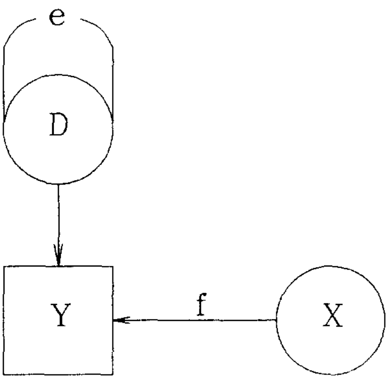

Figure 2. Path Diagram of Equation 1.

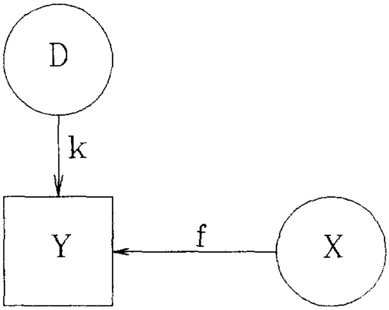

Figure 3. Path Diagram of Equation 2.

to see a path diagram where at least one arrow or arrowhead was omitted or mis-drawn. There are numerous examples of this phenomenon, and the culprits include some of our most distinguished practitioners.

Occasionally papers fail to mention which computer program *(and/or* version number) produced the results. (The fact that LISREL VI might produce different results from LISREL IV apparently does not occur to some authors.)

In numerous papers many models were tested before the final one was arrived at. It is not always easy to trace the model development process in such papers.

Report of starting values is rare. This can be a serious omission if the model is large, because good starting values become increasingly important as the number of free parameters increases.

Person-Machine Communication

LSRM researchers must communicate the precise model they wish to analyze to a computer program. Since models are often revised and retested several times in a sequence, the efficiency of such communication can be very important. The "user-friendliness" of the program interface is thus a paramount consideration in a structural modeling program.

The user-friendly interface is a relatively new innovation in structural modeling. LISREL (Jöreskog & Sörbom, 1984), the oldest and best known of the current programs, fits path models via the LISREL model. This model, as described by Long (1983, p. 57) has no fewer than 18 separate matrices. The complexity of the LISREL model no longer seems justifiable or necessary. This complexity would be tolerable, perhaps, if it did not spill over into path diagrams and model descriptions, burdening them with a plethora of cryptic Greek symbols. But the LISREL model is also difficult to apply to some interesting structural models, which the LISREL program needs to be "tricked" into fitting. Learning these tricks seems distinctly unrewarding, since the program is still unwieldy to use, and offers few important advantages over its competitors.

LISREL broke new ground in the structural modeling field, and deserves respect on that basis alone. Unfortunately, some researchers seem to think that LSRM and LISREL are synonymous, and are unable to distinguish between a structural model, a LISREL model, and the LISREL program, all of which are distinct entities. (For example, it is possible to express a LSRM as a "Bentler-Weeks" model and fit it with the COSAN program, rather than Bentler's EQS.)

COSAN (Fraser & McDonald, this issue) offers a much simpler model structure than LISREL, at a fraction of the cost. Model set-up is much easier on COSAN than on LISREL VI. Using it for complex models is, however, still somewhat time-consuming and error-prone, expect in a few special cases. Models still have to be translated into matrices. The matrices are large, and the potential for misplacing a coefficient or two is still significant.

For ease of use, EQS is the leader in currently available programs. It offers a host of options, plus an interface which allows path models to be expressed in a command language in a relatively easy way.

Users who struggled with the early versions of LISREL can only applaud the recent trend toward user friendliness in LSRM programs. However, the trend is not without its dangers. The inveterate "model-fiddler," perhaps deterred by the difficulties of earlier programs, will find fresh temptation once it becomes relatively easy to test an entire sequence of models.

Machine-Person Communication

Communicating Results of an Analysis

Conveying a model to a computer program quickly and efficiently is only one aspect of Person-Machine Communication. Once the model has been analyzed and results produced, the program should present information in a form that may be evaluated easily in terms of the conceptual model under study. Neither LISREL nor COSAN (both of which present results in model matrices) present output which is conveniently linked to a diagrammatic representation. Numbers must be painstakingly extracted from matrices and conveyed to the appropriate position in the path diagram. EQS is much easier to use, as it presents its output in line format rather than in matrices, and these lines correspond in a relatively obvious way to the segments of the path diagram.

Communication of Sequential Model Modifications

Besides having a convenient input language and producing easily interpretable output, an optimal LSRM program should facilitate sequential model testing. This requires careful program design. Indeed, it is possible for a program to have very efficient input and readily interpretable output, but be relatively inefficient for allowing model modification.

For example, one program required parameter numbers to be consecutive integers from 1 to n. Consequently if a parameter was eliminated during a specification search, many numbers might have to be changed.

Machine-Machine Communication

The three computer programs (EQS, LISREL, COSAN) now available and mentioned above are all capable of analyzing, under the standard statistical assumptions, the vast majority of structural models which have been presented in published papers. However, they are not, in an important sense, able to take advantage of this communality. Suppose one researcher wishes to replicate and extend the modeling process using another researcher's data. It is possible, (though, as noted above, often difficult) to replicate a LSRM analysis "starting from scratch" from the published paper, provided adequate details are given. However, since one researcher has already gone to the trouble of constructing computer input corresponding to the models analyzed, this represents a substantial duplication of effort, and consequently a waste of time.

If the originator and would-be replicator of a study possess the same program (and the same version), the replicator could save considerable time by having access to the input data file from which the published results were generated. Unfortunately, such files are seldom available. Moreover, different programs have a distinctly different command language. There are, in fact, incompatibilities among different versions of the same program. So this potential benefit is seldom realized in practice.

The situation, as it now exists, is essentially a "Tower of Babel." While communication at the level of the path diagram is far from ideal, it is seldom possible at all at the level of computer program input.

Some Suggested Solutions

International Standards for Reporting Structural Models

Journals which publish LSRM papers should attempt to establish minimum standards for structural modeling papers, in much the same way as the APA has established a reporting format for papers it publishes. At the very least, the 7 questions discussed above under "Person-Person Communication" should be answered in every article.

Journal BBS Systems

One reason why academic journals hesitate to accept the responsibility for making raw data (or covariance matrices) available to readers is the expense and inconvenience involved in mailing and/or maintaining the information. In some cases, the covariance matrices are simply too large for printing while in others (such as papers involving some ADF methods) raw data are required.

The microcomputer bulletin board system (BBS) furnishes a superb solution to the above problems. Users simply dial the BBS and download the required information from files. Such a system also furnishes a very convenient device for distributing public domain software referenced in articles printed by the journal.

Cross-Translation Programs

One approach to problems in machine-machine communication would be to construct "cross-translators," computer programs which would, for example convert LISREL input to, say, COSAN input in cases where this is possible. This is a potentially costly solution, because LISREL and COSAN input options may change from version to version. Moreover, owners of any program would have to own several cross-translators to allow them to access (directly) input for these programs.

A High Level Language for Path Models

An alternative, and more appealing solution is a "high level language" for path models. Different computer CPUs have different machine languages, and machine language programs are usually incompatible across machines - that is why high level languages like FORTRAN and PASCAL exist. If a standard ASCII character language for conveying path diagrams existed, and if all LSRM programs could read it, then this language could be used (a) as a reporting medium in journal articles, and (b) as a way of facilitating machine-machine communication.

An ASCII character language offers a number of other advantages, the greatest of which is portability. Computer files with descriptions of path models can be sent via electronic mail, in hard copy form, or in the widest variety of machine-readable formats.

Moreover, a properly designed ASCII language can facilitate sequential modeling, by allowing rapid modification and retesting of models.

Automated Model Trace and Specification Search

As LSRM programs improve the efficiency of the structural modeling process, the temptation to fiddle with models post-hoc will increase. Current software provides only the most rudimentary facilities for the user to document and facilitate the "specification search" process. Future programs, hopefully, will provide a number of tools for this purpose. Some possibilities are listed below.

1. Automated Model Record. Modeling software could be designed to maintain a convenient record of all models tested on a particular data set. This can be accomplished in a number of ways. Files pertaining to a particular data set could be maintained automatically with a common label and sequential numbering system. A brief descriptive label, chi-square statistic, and degrees of freedom could be maintained for each model in a special model summary file.

2. Comparative Testing Facilities. A special model summary file would facilitate comparative testing procedures, like computation of sequential difference tests and goodness of fit statistics.

3. Subsampling and Bootstrapping for Cross-Validation. Automated subsampling procedures, coupled with a bootstrapping routine, would greatly facilitate the assessment of the outcome of a specification search. The notion that the results of a lengthy specification search should be subjected to some form of cross-validation in order to protect against capitalization on chance is a familiar one (see, e.g., Cudeck & Browne, 1983), and seems eminently worthwhile. However, it is seldom implemented in practice. Having built-in cross-validation capability in the LSRM software may accelerate the implementation of this idea.

Conclusions

Information exchange in the LSRM literature has been faulty in a number of areas. Many of the problems can be traced to outmoded traditions in model-fitting software, and lax reporting standards in journal articles. With moderate effort, many of these problems can be overcome, and the field will benefit substantially as a result.

References

Bentler, P.M. (1985). Theory and implementation of EQS, a structural equations program. Los Angeles: BMDP Statistical Software.

Cudeck, R., & Browne, M.W. (1983). Cross-Validation of covariance structures. *Multivariate Behavioral Research*, *18*, 147-167.

Fraser, C., & McDonald, R.P. (1988). COSAN: Covariance structure analysis. *Multivariate Behavioral Research*, *23*, 263-265.

Jöreskog, K.G., & Sörbom, D. (1984). LISREL-VI user's guide. Mooresville, IN: Scientific Software.

Long, J.S. (1983). Covariance structural models: An introduction to LISREL. Newbury Park, CA: Sage.