



CAUSAL MODELING

**A Supplementary Module
for SYSTAT and SYGRAPH**

PC/MS-DOS

Version 1.0
July 1989

**Written by James H. Steiger
University of British Columbia
Department of Psychology**



SYSTAT, Inc.
1800 Sherman Avenue
Evanston, Illinois 60201
Tel. 312 864.5670

Copyright Notice

EzPATH, Version 1.0, July 1, 1989.

Copyright 1989, SYSTAT, Inc., Evanston, Illinois, and James H. Steiger. All rights reserved worldwide. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the express written permission of SYSTAT, Inc.

Limited Warranty

SYSTAT, Inc. and James Steiger warrant for a period of ninety (90) days from the date of delivery that, under normal use and without unauthorized modification, the program substantially conforms to the accompanying specifications and any SYSTAT, Inc. authorized advertising material; that, under normal use the magnetic media upon which this program is recorded will not be defective; and that the user documentation is substantially complete and contains the information SYSTAT, Inc. and James Steiger deem necessary to use the program. If, during the ninety (90) day period, a demonstrable defect in the program magnetic tape media or documentation should appear, you may return the software to SYSTAT, Inc. for repair or replacement, at SYSTAT, Inc.'s option. If SYSTAT, Inc. cannot repair the defect or replace the software with a functionally equivalent software within sixty (60) days of SYSTAT, Inc.'s receipt of the defective software, then you shall be entitled to a full refund of the license fee.

SYSTAT, Inc. and James Steiger cannot and do not warrant that the functions contained in the program will meet your requirements or that the operation of the program will be uninterrupted or error free.

SYSTAT, Inc. and James Steiger disclaim any and all liability for special, incidental, or consequential damages (including loss of profit) arising out of or with respect to the use, operation, or support of this supplement, even if SYSTAT, Inc. has been apprised of the possibility of such damages.

References

LISREL is a trademark of Scientific Software, Inc.

SYSTAT, SYGRAPH, and EzPATH are trademarks of SYSTAT, Inc.

Citation

The proper citation for this manual is:

Steiger, James H. (1989). *EzPATH: A Supplementary Module for SYSTAT and SYGRAPH*. Evanston, IL: SYSTAT, Inc.

AUTHOR'S PREFACE

EzPATH represents my attempt to make "causal" modeling accessible and convenient, both for the beginning student and advanced researcher. My hope is that, by providing a tool which removes much of the mystique and tedium from the modeling process, I will encourage people to think more clearly about basic issues of scientific logic while engaging in that process.

Good science is founded on logic and clear communication. Unfortunately, both these essentials tend to be placed on the shelf when social scientists first import a multivariate methodology. There is considerable evidence, from the history of factor analysis and multidimensional scaling as well as causal modeling, that until a certain "convenience level" is reached in implementing a new multivariate analytic technique, users tend to be (1) overly impressed by technical aspects (which they do not fully understand) and (2) blinded, at least temporarily, to very basic problems with the techniques. The result is that the tail wags the dog, methodologically speaking, and papers demonstrably bereft of any important scientific content are hustled into print because of their "methodological sophistication." Thirty years ago, an individual could earn a Ph.D. simply by being able to generate a successful factor analysis. Today individuals well versed in the vagaries of covariance structure analysis programs enjoy high priest status in many social science departments.

EzPATH, the PATH1 language and EzPATH diagramming rules, are engineered to facilitate clear communication and to promote more careful checking of causal modeling papers prior to publication. In my teaching of causal modeling, I have encountered a number of published papers which have obvious flaws which would have been detected if the models in the papers had been checked against the reported data. In some cases there were errors or ambiguities in the path diagrams. In others the models were not identified, or obvious variations on the published model fit the data as well as or better than the published model did. Still others had errors in the published correlation or covariance matrix. Clearly, no reviewer asked the key questions (1) "Exactly what model was fit to the data?", (2) "Does the model make sense?", (3) "Do other equally sensible models fit the data equally well?", and (4)

"Are the results correct?" One reason the questions were not asked was that asking them would have involved too great an expenditure of time. EzPATH makes it possible to answer these questions relatively quickly.

I want to emphasize that this User's Guide is not a textbook on causal modeling, any more than the SYSTAT manual is a textbook on statistics. For this reason, I concentrate on presentation of a reasonably large number of examples from previously published sources. These examples are presented rather uncritically, although in some cases I cannot resist a comment where I feel it is particularly appropriate.

I have strong personal views about what is right and what is wrong with current causal modeling practice. I hope to present these views soon, with the elaborate justification they require, in a textbook. I have resisted the opportunity to present my views in detail here, because they may be controversial, and I want EzPATH to be evaluated on its own merits as a tool for social science research.

The program does include a number of innovative features. First, and most obvious, are the engineering features of the human interface. Second, and perhaps more important, I have introduced a noncentrality-based procedure for evaluation of model fit, which includes confidence interval estimation methods. This approach attempts to answer the key questions, "How good is model fit in the population?" and "How precisely have we determined the quality of model fit?" I introduced this approach in 1980, but it has been largely ignored by my colleagues until now. I suspect that the approach will become popular very quickly, as it offers obvious merits.

The emphasis in EzPATH is on simplicity and functionality. The program has its imperfections, and is still evolving. There are features of competitive programs which EzPATH doesn't have – and vice-versa.

The fact is, the vast majority of the causal models ever published could have been analyzed with EzPATH 1.0. In many cases, EzPATH would have shortened the model setup time by an

order of magnitude, saving hours of frustration and fatigue. Moreover, some of the most significant (and currently unique) statistical features of the program can add a new perspective to previously published analyses.

I began writing EzPATH primarily for my students. It has enabled me to take a different approach to teaching causal modeling, one which frees more time for substantive issues. I hope that you find the program useful and convenient. Please don't hesitate to write to me with suggestions or criticisms. I'll try my best to respond to them.

James H. Steiger
Department of Psychology
University of British Columbia
Vancouver, B.C., Canada V6T 1Y7
May, 1989

ACKNOWLEDGEMENT

A number of individuals and organizations have contributed, directly or indirectly, to the development of EzPATH. I will acknowledge only a few of them here.

Over the years, the Natural Sciences and Engineering Research Council of Canada (NSERC) has provided generous support for my research in theory and application of correlational methods. Most of my research papers cited in this manual received NSERC support.

Leland Wilkinson contributed an enormous amount of energy, enthusiasm, and useful advice throughout the latter stages of program development. Without his help, and the support of the entire SYSTAT organization (especially Mike Pechnyo, who, several times, gave very useful programming advice), the program might never have been completed.

Many colleagues have played significant roles (some perhaps without realizing it) in shaping my views on statistics, either as teachers, colleagues, or research collaborators, or simply through the quality of their work. Among them, two in particular stand out.

Peter Schönemann, my friend and mentor at Purdue, combines brilliance, honesty, patience and perspective. Peter was a truly devoted teacher, and I will always be grateful for the many hours he spent helping me over the difficult points of multivariate analysis.

Michael Browne, my friend and research collaborator, is the epitome of the phrase "a gentleman and a scholar." He combines intelligence and creativity with an extraordinary attention to detail and devotion to his field.

Both these men have inspired me. It has, indeed, been a privilege to know them and to work with them.

Many other colleagues have offered their support and friendship. I'm afraid to try to list them all. Many of them are cited in this manual. They know who they are, and that I appreciate them.

TABLE OF CONTENTS

1. Introduction	1
2. EzPATH Installation.....	4
3. A Brief Tutorial Exercise	5
Key glossary terms.....	5
Entering a model.....	12
Error recovery	14
Saving output.....	16
Revising a model	18
Interrupting iteration without losing data	21
Restarting iteration after an interruption	22
4. Some More Examples	23
Stability of alienation.....	23
Personality and drug abuse.....	28
5. EzPATH Diagrams and the PATH1 Language	30
EzPATH diagrams.....	30
PATH1 language rules.....	36
6. EzPATH Commands.....	39
ACCURACY	40
AUTOFIX	41
CMODEL.....	43
ESTIMATE	44
FACTORMODEL	45
ITERATIONS.....	47
METHOD	48
MODEL	49
NUMBER.....	50
REPORT.....	51
RMODEL.....	52
SAVE.....	53
SE.....	54

Ross Matsueda and John Steiger offered some very intelligent and helpful comments on the program and documentation during its development. John Steiger also provided computer facilities for some phases of program code development.

My students, who used the program this year while studying causal modeling, offered many useful suggestions for improvement.

I am lucky to have a wonderful family. My mother Mary Steiger, a brilliant and inspirational teacher of mathematics, encouraged my interest in statistics when she noticed me studying baseball records at the age of 6. My late father Ernest, and my brothers John and Paul have always been there when I needed them. My daughters Andree and Roberta are a constant source of pride and inspiration.

Most important is my wife Juanita. She is so special in so many ways. Without her I would have given up about 400 bugs ago.

STANDARDIZE.....	55	9. Illustrative Problems and Examples	96
7. Technical and Theoretical Aspects of EzPATH.....	56	A confirmatory factor analysis model.....	96
Models and methods.....	56	Path models for home environment and mathematics	
The LISREL model.....	57	achievement.....	99
The McArdle–McDonald RAM model	60	Principal components analysis	103
The Bentler–Weeks model.....	63	Testing pattern hypotheses on correlations.....	107
The EzPATH model	66	Testing for circumplex structure.....	108
Statistical estimation	67	Testing for stability of a correlation matrix over time... 113	
Model identification.....	70	Pattern hypothesis tests on a covariance matrix	116
Unconstrained minimization technique.....	74	Test of equality of dependent variances	116
Goodness of fit indices.....	75	Test for compound symmetry	117
General theoretical orientation	76	Test theory models for sets of congeneric tests.....	121
Noncentrality based parameter estimates and		Multitrait–multimethod factor models.....	129
confidence intervals	77	Longitudinal common factor models	133
The Steiger–Lind RMS index.....	80	Effect of peer influences on ambition.....	136
The population gamma index.....	81	10. Hints on Using EzPATH	140
Jöreskog–Sörbom (1984) GFI.....	86	Generating a covariance matrix for your own model.	140
Jöreskog–Sörbom (1984) Adjusted GFI	87	Avoiding system crashes and data loss.....	141
Akaike (1973) information criterion	87	Plotting normalized residuals.....	142
Schwarz's (1978) Bayesian Criterion.....	87	When your output doesn't agree with that other program's 142	
Browne–Cudeck (1989) single sample cross–validation		Watch out for the lonesome latent variable!.....	143
index	87	Entering a covariance or correlation matrix.....	143
Historical background.....	88	Program limitations	143
8.EzPATH Technical Output.....	90	References.....	145
Memory usage table.....	90	Appendix – SAVE File Format	151
Iteration history.....	90	Index.....	152
Results of iteration	90		
Chi–Square test statistic	90		
Fit indices.....	91		
Trace criterion	92		
Standard errors.....	92		
Optional output	93		
Input matrix	93		
Reproduced matrix.....	93		
Standardized residuals	93		
Normalized residuals.....	94		
Results for the fitted model.....	94		

1. Introduction

EzPATH is a user–friendly, interactive program for performing analysis of covariance structures, including techniques commonly referred to as "causal modeling," "confirmatory factor analysis," "structural modeling," and "path analysis."

EzPATH is designed to enable the researcher who is not necessarily an expert in advanced multivariate methods to perform structural modeling procedures quickly, easily, and accurately, while also providing power, efficiency, and time–saving features which will appeal to the experienced causal modeler.

Major new features in EzPATH include:

Introduction of the PATH1 computer language for conveying path diagrams to and from the computer. The PATH1 language allows path diagrams to be entered into a line file in a format which closely matches the diagram itself. The language mimics the diagram so closely that entering a model correctly is almost trivial. Moreover, the PATH1 language also serves as a medium for presenting output from the statistical estimation process. Hence, one can take the results from one analysis, modify them, and use them as input for the next analysis.

Full compatibility with the SYSTAT system, including use of the integrated FEDIT facility for quick revision of input and for inserting comments into output. EzPATH reads variable names directly from the SYSTAT file. There is no need to type in lists of variable names. You can switch back and forth between EzPATH and other SYSTAT modules.

Simplified, *interactive* operation. Models which take hours to set up on other programs can be entered in minutes with EzPATH. The program allows models to be entered interactively with syntax checking. Many model errors are flagged immediately on entry, and recovery from errors is extremely fast and easy.

Advanced statistical features. Other programs offer goodness of fit indices which are sample statistics with no statistical rationale. EzPATH provides a statistical rationale for assessing goodness of fit. It defines and calculates a number of new coefficients, as well as statistically-based variants of some old ones. In particular, EzPATH provides a confidence interval for the population equivalent of the GFI indices calculated by LISREL VI.

Advanced program logic. The program does the work, not you! For example:

There is no need to learn a complicated system of variable and matrix types. Extensive experience with matrix algebra is not required to use the program – EzPATH does most of the work. It creates an internal algebraic model, analyses the model, and reports results in the PATH1 language.

You can use manifest variable names up to 8 characters long in EzPATH. You can use your own names. There is no restriction to use of names like "F1", "X1", etc. Latent variable names can be up to 20 characters long.

There is no need to re-number coefficients after modifying a model. Suppose you have typed a path model into EzPATH with 12 unknowns, labelled 1–12. You analyze the model, then decide to modify it to eliminate the path with unknown number 7. This path can simply be erased.

A number of automatic model-generation commands greatly reduce the amount of typing necessary to create EzPATH input.

For example, factor analysis model command files can be generated automatically. EzPATH has a command called FACTORMODEL which allows you to construct, automatically, the PATH1 commands corresponding to a common factor model. If you have factor analyzed

some data previously with the FACTOR module and have the factor pattern in a SYSTAT data file, EzPATH will automatically incorporate the numerical results as starting values. This feature provides the user with substantial benefits: (i) Confirmatory factor models can be constructed, tested, and modified very rapidly with almost no typing; (ii) The user can get a head start on typing the "measurement model" for many path models.

There is no need to classify or order manifest variables into exogenous and endogenous types. Some programs require you to classify manifest variables into X and Y types. Others require explicit lists of variable names. EzPATH does not require manifest or latent variables to be ordered or classified.

2. EzPATH Installation

BACKUP YOUR DISKETTES!

EZPATH is not copy protected. Before performing any tests with the program, please backup your disks! This is a very simple process. Take the EZPATH diskettes, place write-protect tabs on them, and follow the instructions below.

If you have a hard disk system, place the EzPATH disk in drive A: and type

DISKCOPY A: A:

then follow the prompts.

If you have a floppy-based system, place the DOS disk in drive A: and type

DISKCOPY A: A:

When prompted to insert the source disk in drive A:, insert the EzPATH disk to be copied in drive A:, then press any key to start the diskcopy process.

INSTALL EZPATH IN YOUR SYSTAT DIRECTORY

Place the distribution diskette in either drive A: or drive B:
Type

A: [or **B:** if the diskette is in drive B:]

followed by a carriage return. Then type

INSTALL

START THE PROGRAM

Select the SYSTAT directory, and type

EZPATH

3. A Brief Tutorial Exercise

(Note: Throughout the exercise, we assume that the user is already familiar with SYSTAT, and, in particular, the FEDIT file editing facility. Users unfamiliar with FEDIT should study carefully the documentation and tutorial on pages 11–20 of the SYSTAT installation guide. Remember that EzPATH, like all SYSTAT modules, includes an on-line help facility. Simply type HELP if you need it.) EzPATH is designed to allow the fast entry of structural models directly from a path diagram. This can be especially useful when one is attempting to replicate a published structural model, because often authors use *only* the path diagram to convey their models. Path diagrams occur in the scientific literature in a number of notational variations. Within this manual, we will adopt strict conventions and adhere to them, although we will also discuss some notational variations that are in wide use.

EzPATH introduces a new computer language, PATH1, which is especially designed to facilitate communication of path diagrams both to and from a computer. Because it relies on standard ASCII characters, and is line oriented, PATH1 allows structural equation models to be recorded unambiguously in a way which maintains efficiency without sacrificing portability.

Consider the structural diagram in Figure 1. This diagram represents a common factor model, with 2 common factors defined on 6 observed (i.e., "manifest") variables. We'll use this model to introduce some terminology which we will be using consistently throughout the manual. Here are the key terms.

A. ARROW

A line with an arrowhead on one end. Within EzPATH, we simulate an arrow with the PATH1 programming language by two dashes with an arrowhead on one end. The dashes may, depending on circumstances, have material between them.

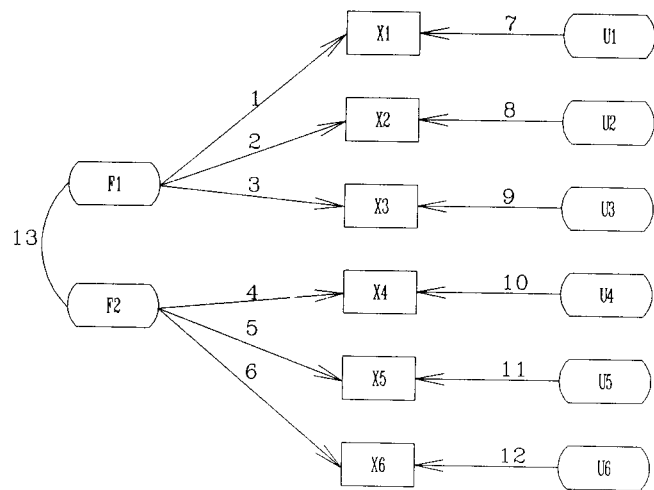


Figure 1. A Common Factor Model.

Examples.

-->
 -1->
 -2 { 0 . 65 } -->

Arrows are used in structural diagrams to indicate a linear relationship between two variables. The "flow of causality" (in the loose sense employed in the causal modeling literature) is to the variable pointed to by the arrowhead.

B. ARROWHEAD

The ">" character

C. ARROWLINE

The straight line part of an arrow.

D. DIRECTED RELATIONSHIP

A relationship between two variables connected by an arrow.

The variable pointed to by the arrow is the dependent variable in the linear relationship, and appears on the left side of the "=" in the linear equation. The variable pointed from is the independent variable in the linear relationship, and appears on the right side of the "=" sign. In EzPATH, it is not necessary to construct any linear equations, nor is it necessary to know how to construct them to use the program.

In Figure 1, for example, there is a directed relationship between latent variable F1 and manifest variable X1.

E. WIRE

In Figure 1 there is a line connecting variables F1 and F2. This line has no arrowhead on either end. We will refer to such a line, for simplicity, as a "wire" throughout this manual. A standard convention in many path diagrams is to present such lines with arrowheads at both ends (Jack McArdle calls these two-headed arrows "slings"). Obviously, lines with no arrowheads can serve the same function as two-headed arrows, with less visual clutter. Moreover, it turns out that using wires instead of "slings" is a physically more efficient way of communicating a key aspect of any path diagram, i.e., which variables are exogenous. We will employ wires rather than "slings" throughout this manual.

F. UNDIRECTED (VARIANCE-COVARIANCE) RELATIONSHIP

"Wires" are used to indicate variances and covariances. In a proper path diagram, only exogenous variables will be "wired together," i.e., have wires connecting them. Some authors refer to variances and covariances indicated in this fashion as "undirected" relationships. The wire between latent variables F1 and F2 in Figure 1 represents an undirected relationship between them (i.e., their covariance).

G. MANIFEST VARIABLE

Manifest variables are variables which have been measured directly, and for which observed data are available. Names of these variables are recorded in the SYSTAT *.SYS file being analyzed. Manifest variables will be represented in all path diagrams in this book by a variable name enclosed within a box. In the PATH1 language, we will represent such variables with a variable name enclosed within square brackets.

Examples.

[X1]
[ANXIETY]

H. LATENT VARIABLE

Latent variables have not been observed directly. They are represented in EzPATH diagrams by the variable name enclosed within an oval or circle. For example, in Figure 1 the common factor F1 is a latent variable.

In the PATH1 language, latent variables are represented with a variable name enclosed within parentheses (i.e., rounded brackets). The PATH1 language statement

(U1) --> [X1]

shows an arrow directed from latent variable U1 to manifest variable X1.

I. EXOGENOUS VARIABLE

Any variable with no arrow pointing to it.

J. ENDOGENOUS VARIABLE

Any variable with at least one arrow pointing to it.

K. STRUCTURAL COEFFICIENT

One way of viewing a path diagram is that it is a pictorial representation of a set of linear equations. In particular, a linear equation $Y = aX$ is represented in the form

[X] -a-> [Y]

The weighting coefficients in the linear equations are written in our path diagrams (and, indeed, in most variations of such diagrams) in the middle of the arrow connecting the independent and dependent variables. Such coefficients are frequently referred to as "structural coefficients."

L. FREE PARAMETER

When we test structural models, structural coefficients are sometimes fixed at a particular numerical value (often 1 or 0). More often, however such coefficients are free to vary, and are estimated by the model-fitting procedure. We refer to such coefficients as "free parameters."

M. PARAMETER NUMBER

For purposes of identification, free parameters in our path diagrams, and within EzPATH, are assigned parameter numbers. These numbers may be thought of as labels. If two structural coefficients are assigned the same parameter number, then they will be assigned the same numerical value during iteration of the solution and in the final fitted model. In this situation, some authors refer to the parameters as "constrained."

N. FIXED VALUE

A structural coefficient which is fixed at a particular numerical value, most commonly zero or one. In our diagrams, coefficients which are fixed will have no parameter number (implicitly a parameter number of zero). Fixed values can be distinguished from parameter numbers, because the latter will always be integers, while the former will always contain decimal points. In the PATH1 language, fixed values are given in braces, with no parameter number preceding the braces. The example below shows a path with a fixed coefficient of .5.

(F2) - {0.5} -> [X5]

O. START VALUE

The value assigned to a free parameter at the beginning of iteration. The default starting value is .50. If you wish to assign other values, they are given in braces immediately following the parameter number. For example, the statement below shows a path from variable F to variable X3, with the coefficient number 2 being assigned a starting value of .45.

(F) - 2 { 0.45 } -> [X3]

Having introduced our basic terminology we are ready to proceed with a numerical example. Figure 2 presents a path diagram of a common factor model in which a single common factor and 4 unique factors reproduce 4 observed variables. This simple diagram illustrates a number of features of our path diagrams which we will attempt to hold constant throughout this manual.

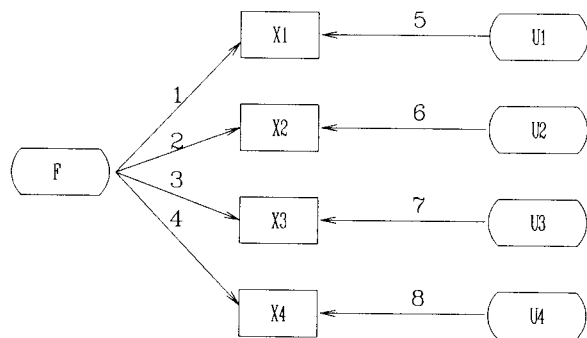


Figure 2. A Single Common Factor Model.

The common factor F is an "exogenous latent" variable which loads on 4 "endogenous manifest" variables -- X1, X2, X3, X4. Note that the manifest variables are represented by the variable names in a rectangular box, while the latent variables are in ovals.

The general convention in structural diagrams is to represent all variances and covariances among *exogenous* variables, unless the variance is 1 or the covariance 0. We will follow that convention for latent exogenous variables in our diagrams.

Hence, the latent exogenous variables F, U1, U2, U3, and U4 can be assumed to be uncorrelated with variances of 1. In our diagrams, we will use a somewhat unusual convention for *manifest exogenous variables*. We will require that either *all* such variables have their variances and covariances represented explicitly, or *none* of them. In the latter case, we assume that all variances and covariances for the manifest exogenous variables are *distinct* free parameters, i.e., have different free parameter numbers not duplicated anywhere else in the diagram.

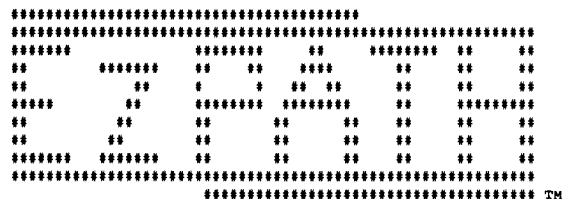
In the initial diagram, all structural parameters are free, and all are assigned different parameter numbers. Hence, in this case, we are fitting an unrestricted common factor analysis model.

In EzPATH, fitting such a model involves 3 main steps: (a) power up the program and import a data file, which must be either a correlation, covariance or SSCP matrix, with the USE command; (b) enter the model with the MODEL command by transcribing the path diagram into the PATH1 language; (c) set desired computational options and ESTIMATE the model coefficients.

To invoke EzPATH, type

EZPATH

You should soon see the following screen display.



Version 1.0
Copyright, 1989
James H. Steiger and SYSTAT, Inc.
You are in EzPATH module.

>
Next, import the data file. Data for this example are in a file called DEMO1.SYS, which was on the distribution disk. Simply enter the command

USE DEMO1

What appears next is the familiar SYSTAT display of variable names. This indicates that the data file has been found and read successfully.

You are now ready to enter the model in the PATH1 language. There are actually several ways you can do this. We will try a method called "interactive entry" first. Begin by issuing the MODEL command by typing the word MODEL followed by a carriage return. When the command cursor reappears, you may enter your model, line by line. In this mode of entry, each line you enter is checked immediately for correct syntax. Any obvious syntax error is flagged immediately, and a diagnostic message is issued. Most errors can be recovered from quickly and easily.

Once you enter the MODEL command, EzPATH will interpret all subsequent commands as PATH1 statements, until you issue any SYSTAT or EzPATH command. Many people trying EzPATH for the first time will accidentally end a model prematurely after forgetting to include at least one path. If this happens don't worry. You can recover from such an error in seconds, and none of your work will be lost. We'll demonstrate how to recover from an error of this type later in the tutorial.

After the MODEL command has been issued, models are entered into the computer one relationship at a time. Any line beginning with a "*" is treated as a comment, and is added to the command log, but is not parsed. Blank lines are treated similarly.

Blank spaces have no effect in PATH1 statements. Consequently, you may use blanks liberally in formatting your input for easy readability.

We now enter the model directly from the path diagram. Consider the arrow in Figure 2 between factor F and manifest variable X1. This arrow represents a single relationship, in this case a "directed" relation. We enter it with the following statement:

(F) -1-> [X1]

(You should type the above statement *exactly as it appears*, with the letters in upper case, and then hit the carriage return.)

Notice that, in the PATH1 language, each arrow and wire in a path diagram produces a single model statement. Arrows are represented with "-->", while wires are represented with "--". If a wire or arrow has a structural coefficient which is a free parameter associated with it, a coefficient number must appear between the two dashes. In the first model statement we entered, the structural coefficient for the arrow was assigned parameter number 1.

Now let's enter the path from F to X2. There are actually two distinct ways you can do this. Either of the following two lines will work. (DON'T TYPE EITHER OF THESE LINES IN YET!)

-2-> [X2]

or

(F) -2-> [X2]

This illustrates one of the standard PATH1 conventions which makes PATH1 files easy to read, and easy to convert to and from path diagrams. In any relationship there are two variables. If, on

the line following a relationship, the first variable is left out, *it is assumed to be the first variable from the last preceding line which had two variables in it*. This feature can save the user a considerable amount of typing, and can contribute significantly to the readability of the file.

Before continuing to enter our model, let's simulate an error in EzPATH, so we can examine some of the facilities the program provides to make error recovery easy. Pretend you were typing the first alternative version of the second model line, but forgot one of the dashes. Type the following erroneous entry exactly as it appears.

-2> [X2]

In this case, you forgot the second dash in the arrow, and thus attempted to enter a line with incorrect syntax. EzPATH therefore rejects the line with an error message. Because this line caused no serious damage to the emerging picture of your model which EzPATH is constructing, the program informs you that you may re-enter the preceding line. You could re-enter the entire line, but there is, as in any SYSTAT module, an easier way. Here we use the SYSTAT command line editing facility, which allows fast editing of the previous 5 command lines. (Users unfamiliar with this facility should immediately read page 24 in the SYSTAT manual.) Simply hit the **F9** function key, and a copy of the previous line will appear. You can edit this line using standard SYSTAT line editing functions. Hit the **<Ins>** key to enter insert mode, insert the missing dash and hit the carriage return key to re-enter the line. This time EzPATH should accept the line without complaint.

Now try entering the rest of the model on your own. But in the process of doing this, let's simulate another error. Type the following lines carefully, exactly as they appear:

-3-> [X3]
-4-> [X4]
(U1) -5-> [X1]
(U2) -6-> [X2]
(U3) -7-> [X3]

ESTIMATE

Following the last line, you will see a message that the model has been parsed successfully. This message is of course, premature, because there is one path in the diagram which you "forgot" to enter.

EzPATH is not clairvoyant. In this case, because you included the **ESTIMATE** command, EzPATH assumed you wished to terminate the model. In effect, you have proposed a common factor model in which the 4th manifest variable has no unique variance! It might be the case that you wouldn't notice the oversight in your model until after going through the entire estimation process.

When you exit the model, EzPATH reminds you that it can't begin estimation yet because you have not yet indicated the sample size for this data matrix. It will ask you whether you wish to use a default value of 101, or enter a value.

Imagine, for example, that you suddenly realized your error at this point. First, enter

N

in response to the prompt, because you do not wish to use the default value of 101.

We would like to add an additional line to the model. Once you leave model entry mode, you cannot simply enter another line, because EzPATH goes through an extensive set-up procedure to convert your model to a set of set of model equations immediately after leaving model entry. However, error recovery is extremely simple. Type

FEDIT >

to activate FEDIT and edit your previous command lines. Notice that your MODEL statement, and all your *correct* PATH1 model specification statements, are available in the command line log.

At this point you have several options. One option is to use the facility in **FEDIT** which allows you to mark a set of lines as a block (they will appear on your screen highlighted in reverse video). If you do this, the highlighted commands will be executed after you leave **FEDIT**. (If you are unfamiliar with the method for marking blocks of lines in the **FEDIT** editor, consult your SYSTAT documentation. For most MSDOS machines, blocks are marked using the <F9> function key at the beginning and end of the block.) In this case, you can simply highlight the **MODEL** statement and all the **PATH1** lines up to but not including **ESTIMATE**, and exit **FEDIT**. When asked if you wish to save your changes, you can say "N." At this point **EZPATH** will execute all the lines you highlighted. In essence, then, you will have returned to where you were in the model entry process just before you made your error. Then you can enter the last line in the model, i.e.,

(U4)-8-> [X4]

Of course, you could, while in **FEDIT**, have simply added the line there, highlighted all the lines by marking them as a block, and exited. The result would have been the same.

At this point we have correctly entered the factor model, and we are ready to estimate the coefficients of the model. If you wish to save a permanent copy of the output, you would type

OUTPUT DEMO1

As is the case in other SYSTAT modules, this will create a file called "DEMO1.DAT," which will hold the results of your analysis. Before estimating our model, we must inform **EzPATH** of the number of observations on which the covariance or correlation matrix was based, because this information is not present in the **DEMO1.SYS** file. In this case, the number is 130, so type

NUMBER=130

and a carriage return.

Then type

ESTIMATE

Since you have not activated any special estimation options, **EzPATH** will begin iterating a best-fitting solution to the model you have entered. The standard option is to perform a least squares fit of the model first, then use the values from this estimation as starting values for a maximum likelihood estimation. During the computation, **Ezpath** will report the progress of the iteration on the screen by printing the value of the discrepancy function along with the iteration number. (You can alter this behavior with the **REPORT** command.) At the end of each screen of output, the program will pause. You can disable this pausing function with the command

PAGE SCREEN=SCROLL

which is described on page 45 of the SYSTAT manual.

When the maximum likelihood estimation has finished, **EzPATH** will print some summary statistics, and then output the estimated model in the **PATH1** language. What you should see in the last part of the output is the following :

Here are the results for the fitted model.

MODEL

(F)-1{ 0.320}-> [X1]
-2{ 0.674}-> [X2]
-3{ 0.431}-> [X3]
-4{ 0.520}-> [X4]

(U1)-5{ 0.947}-> [X1]
(U2)-6{ 0.739}-> [X2]
(U3)-7{ 0.903}-> [X3]
(U4)-8{ 0.854}-> [X4]

The output will look identical to the input, except that, immediately following the parameter numbers, the estimates of the various coefficients have been inserted within braces. If you had requested standard errors for these estimates (by typing

SE=YES), they also would have been printed inside the braces, and the output would have looked like this:

Here are the results for the fitted model.

```

MODEL
(F)-1{ 0.320 SE= 0.112}->[X1]
      -2{ 0.674 SE= 0.133}->[X2]
      -3{ 0.431 SE= 0.114}->[X3]
      -4{ 0.520 SE= 0.119}->[X4]

(U1)-5{ 0.947 SE= 0.064}->[X1]
(U2)-6{ 0.739 SE= 0.109}->[X2]
(U3)-7{ 0.903 SE= 0.067}->[X3]
(U4)-8{ 0.854 SE= 0.075}->[X4]
  
```

At this point, let's imagine that, after inspecting the model, you decided to modify it and test it again. Suppose, for example, you decided to test a revised model in which the structural coefficient in the path from factor F to manifest variable X3 was constrained to be the same as the coefficient from factor F to variable X2. This new model corresponds to the path diagram in Figure 3.

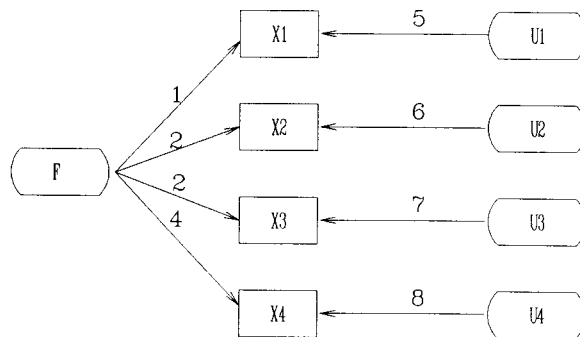


Figure 3. Constrained Single Factor Model.

In order to test this revised model, we need only enter the same model as before, but with one change -- the parameter numbers assigned to the two coefficients forced to be equal would now have to be the same.

[*Note:* It is *not* necessary to re-number the parameter numbers so that they range from 1 to k. You could, for example, number both of the coefficients 2, or both of them 3, or both 75, for that matter.]

FEDIT makes this model modification especially easy, as the experienced SYSTAT user has probably already realized. You *could*, for example, type "**FEDIT** >" and simply edit the command log, highlight the word **MODEL**, the revised model commands, and the **ESTIMATE** command. After hitting **F10** to exit **FEDIT**, the new model would be entered immediately.

However, there is another method which may produce the results faster when the model is complex and iteration takes a fair amount of time. We'll use this method to illustrate another key feature of EZPATH and the PATH1 command language.

Instead of editing the command log, we'll edit the output log, and use it instead. Type "**FE ***", and you will see the results just displayed from the previous model. The way PATH1 syntax works, the first number appearing within braces in a line can serve two purposes. When the line is output, the number is the *result* of estimation. When the line is input, the number is the *start value* for iteration.

This means that you can take the output from one EzPATH analysis, highlight it, change it, and use it as input for the next analysis. The output values from the old analysis will serve as start values for the new analysis.

EzPATH prints the word **MODEL** just prior to the PATH1 output statements. This means that, if you decide to use the output from one analysis as the input to the next, you don't have to bother to type in this command.

Recycling EzPATH output in this fashion can speed up iteration substantially when compared to performance using default starting values. If no start values are specified, EzPATH estimates starting values using a non-iterative method. If any starting values are specified, EzPATH uses default values of .5 for the other free parameters, and 1.0 for exogenous latent variable variances.

To continue with our example, move the cursor just above the first PATH1 output statement. Notice that the command **MODEL** is already there. Move to the path from factor F to variable X3, and change the coefficient to 2. Move the end of the PATH1 statements, and type the word estimate. Now, move back to the **MODEL** statement, and, using the **F9** key, highlight the **MODEL** command, PATH1 statements, and **ESTIMATE** command. After marking the end of the block with the **F9** key, hit the **F10** key to exit. Tell EzPATH to submit the commands, and watch your revised model estimated and tested in seconds.

Of course, you could modify the model still further, by modifying the output log again.

In the above example, if you changed the parameter number in the path from factor F to variable X3 to 2, it would not be necessary to change the values within braces to be equal. When two paths have the same parameter number, EzPATH uses the last starting value input for that parameter as the starting value.

These design features make PATH1 especially suited for interactive structural modeling. The language serves equally well for conveying results to or from the computer. This can offer significant advantages in operation, one of which we just demonstrated. Here is another. Suppose you are analyzing a large model, one which takes a lot of time to estimate, and you receive an urgent phone call which requires you to use your computer immediately to search for some data. You are in the middle of lengthy iteration. With other analysis systems, you would have no choice but to reset the computer and lose all your work. However, with EzPATH that won't happen. Here's why. At any time during iteration, if you hit **<Ctrl>-<Break>** keys simultaneously, iteration will stop, and the results, based on the iterations which have been completed, will be printed. At this point, if you have

been saving your work to an OUTPUT file, you can immediately type **QUIT** and leave EzPATH. Otherwise, you should type **FEDIT ***, type a blank or some other character into your screen buffer file (to activate the filesave switch), then hit **F10** and save the output to a file before **QUIT**ting. Once your phone call is completed, you can re-enter EZPATH, edit the output file, and submit the previous results as your model. These results, used as start values, will resume the iteration approximately where it was.

Here are some tips for optimizing your use of this facility. Whenever analyzing a large model, always use the **OUTPUT** command to attach an output file so that, if you must interrupt the iteration and leave EzPATH, you will be able to save all of your output conveniently. Also, when loss of time could be critical, always set **FORMAT=5** or higher, so that the printed results will have enough accuracy to prevent loss of iterations due to round-off error. (For larger models, we find that 3 digit accuracy is sometimes insufficient. EzPATH performs its internal calculations in IEEE double precision, i.e., 16 digit accuracy, but results are read back in only to the accuracy level they were printed in the PATH1 output statements.)

If you interrupt iteration during the **LS** estimation, and the **DUAL** option is in force, EzPATH will jump into the **ML** estimation phase. To leave this phase, simply interrupt the program a second time.

Remember that, if you interrupt the program, the current parameter estimates remain active, and will be used as start values if you restart estimation with an **ESTIMATE** command. (If you have progressed to the **ML** phase of a **DUAL** estimation procedure, you will restart with a **LS** estimation *unless you first type METHOD=ML.*)

The ability to re-enter the estimation procedure is useful in several ways. For example, if your output is printed with a large number of decimal places, you can always change the format to a lower number and reprint the results. Suppose you have obtained results by maximum likelihood estimation with 5 decimal places, and you wish to reprint them with 2. The sequence of operations would be as follows:

METHOD=ML
FORMAT=2
ESTIMATE

This begins the estimation process with starting values equal to the values just obtained. Since the solution has already converged and is still active, you will see the reformatted results almost immediately.

When analyzing a series of models with EzPATH, you can maintain, in the attached output file, a complete record of your model fitting session. There are several devices at your disposal which will make the documentation process more effective. First, feel free to use the **NOTE** command to add comments to your output. Second, don't hesitate to use **FEDIT** to enter the output file, and add clarifying comments as to why you chose a particular model to analyze. I urge you to do this.

A word of caution – EzPATH makes it possible to analyze many models in a short amount of time. But the efficiency of EzPATH is a two-edged sword. It can, in the wrong hands, become a tool for mindless model-fiddling. In the right hands, it will remove the mystique from covariance structure modeling, and allow the user to concentrate on substantive considerations. Any newcomer to structural modeling would profit from Norman Cliff's (1983) thoughtful advice.

4. Some More Examples¹

A. Stability of Alienation.

A familiar example in the structural modeling literature is the study on stability of alienation by Wheaton, Múthen, Alwin, and Summers (1977). Jöreskog and Sörbom, in the LISREL VI manual (1984, pp. 22–30), analyzed two very similar structural models with the Wheaton data. The covariance matrix for these data, based on a sample size of 932, is in a file called WHEATON.SYS on the distribution disk.

Figure 4 on page 26 is the structural diagram for the first model analyzed by Jöreskog and Sörbom.

This diagram is similar to one in Jöreskog and Sörbom (1984), with an important difference. Their diagram *does not include (explicitly) the wire labelled with parameter number 4 in our diagram*, although their table of results makes it clear that this parameter (i.e., the variance of SES) was in fact estimated.

This distinction is important, because in EzPATH, exogenous latent variable variances are, for convenience, always assumed to be fixed at one unless explicitly declared otherwise.

Remember, any arrow without any numerical index attached is assumed to have a fixed coefficient of 1. Such arrows may be given in the PATH1 language in a simplified form – for example, the arrow from EPSILON1 to ANOMIA67 is denoted as follows:

¹ Throughout this user's guide, we will use examples from published literature to illustrate the use of EzPATH. Inclusion of any particular example should not be taken as an endorsement of that example. In fact, if you explore these examples carefully using EzPATH, you will find that they can be a springboard for numerous interesting scientific and statistical questions. Detailed explication and/or exploration of these questions would, unfortunately, multiply the length of this volume by a factor of about 5.

(EPSILON1) --> [ANOMIA67]

Try setting up this model yourself with EzPATH. If you fit the model correctly, you will obtain a chi-square value of approximately 71.47 with 6 degrees of freedom.

If you run into difficulty, here is one version of the PATH1 statements for the model. These statements are contained in a file called WHEATONA.CMD on the distribution disk.

*** Wheaton Model A.**

```
(SES) -1-> (AL67)
      -2-> [SEINDEX]
      --> [EDUCATN]
      -3-> (AL71)

(SES) -4- (SES)

(AL67) --> [ANOMIA67]
      -5-> [POWLES67]
      -6-> (AL71)

(AL71) --> [ANOMIA71]
      -7-> [POWLES71]

(DELTA1) --> [EDUCATN]
(DELTA2) --> [SEINDEX]

(DELTA1) -8- (DELTA1)
(DELTA2) -9- (DELTA2)

(ZETA1) --> (AL67)
(ZETA2) --> (AL71)
(ZETA1) -10- (ZETA1)
(ZETA2) -11- (ZETA2)

(EPSILON1) --> [ANOMIA67]
(EPSILON2) --> [POWLES67]
(EPSILON3) --> [ANOMIA71]
(EPSILON4) --> [POWLES71]
```

```
(EPSILON1) -12- (EPSILON1)
(EPSILON2) -13- (EPSILON2)
(EPSILON3) -14- (EPSILON3)
(EPSILON4) -15- (EPSILON4)
```

*** END OF MODEL**

Jöreskog and Sörbom (1984) point out that a modified version of the above model, in which EPSILON1 and EPSILON3 are allowed to correlate, fits the data much better than the original version. This model is represented in the path diagram in Figure 5.

Of course, the only difference between Figures 4 and 5 is the wire between EPSILON1 and EPSILON3. Producing this revised model from your original model should take you about 30 seconds with EzPATH. The revised model is in a file WHEATONB.CMD. To analyze either of the Wheaton models using our command files, simply make sure the files are in your current directory, and type

SUBMIT WHEATONA

or

SUBMIT WHEATONB

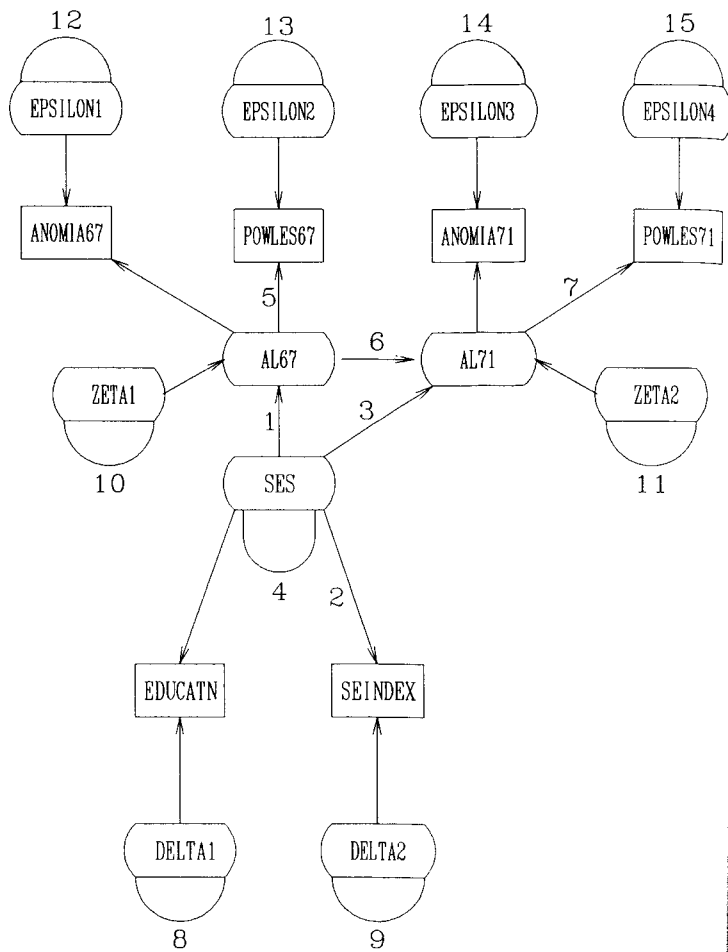


Figure 4. Structural Model A for Stability of Alienation

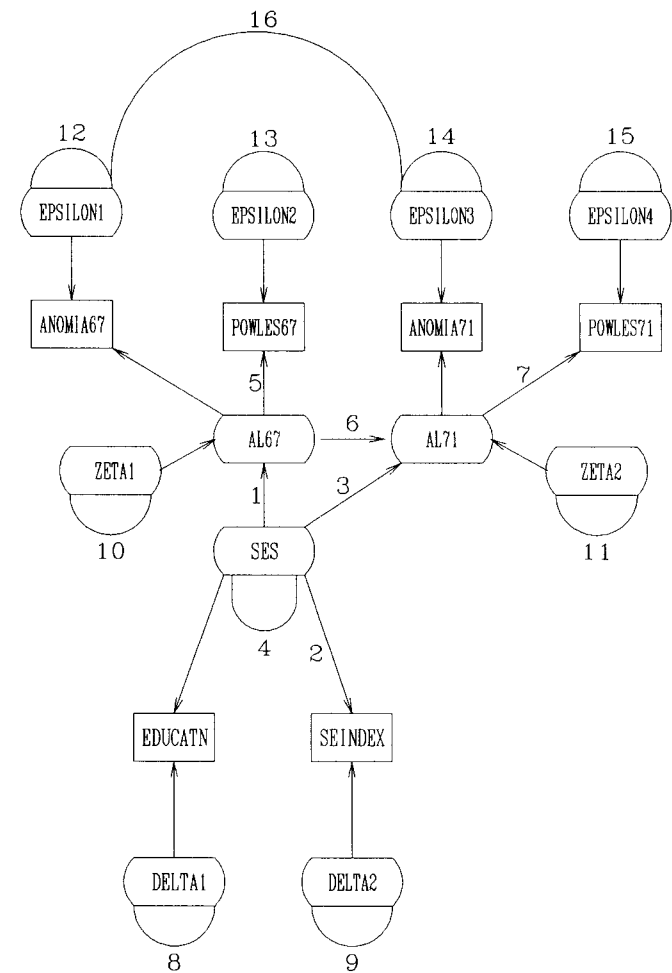


Figure 5. Structural Model B for Stability of Alienation

B. Personality and Substance Abuse.

Huba and Harlow (1987) present a structural model relating personality characteristics to alcohol and marijuana consumption in adolescents. The correlation matrix (to the two-digit level of precision given in their printed article) for their data, based on 257 observations, is given in a file called HH.SYS, on the distribution disk². Their first model corresponds to the path diagram in Figure 6.

EzPATH users are urged to obtain a copy of the Huba and Harlow article, because their Figure 1 presents interpretation problems which are typical of those encountered by readers of the structural modeling literature. In their diagram, a term Ψ_1 is used to refer to the variance of the latent variable "Law Abidance." On the other hand, Ψ_3 stands for the variance of a disturbance, or "error" latent variable "Beer Consumption." In each case, the symbols (Ψ_1 and Ψ_3) are attached to arrows, and it might seem to the casual observer that they mean the same thing, although they don't. The first symbol simply represents a wire, whereas the second actually represents a latent variable, an arrow, and a wire. Our diagram makes clear the distinction. The file HH.CMD, which contains a PATH1 representation of the model in Figure 6, is on the distribution disk.

We urge the user to try to produce the PATH1 statements corresponding the Figure 6 *before* examining the contents of HH.CMD. Issuing the command

SUBMIT HH

will produce output (including standard errors) for the model from our file.

Results obtained with EzPATH will correspond very closely, but not exactly, to those reported in the first column of Table

2 in Huba and Harlow (1987). I assume the discrepancy is due to the fact that we are using the correlations from their article, which they reported to only two digit accuracy.

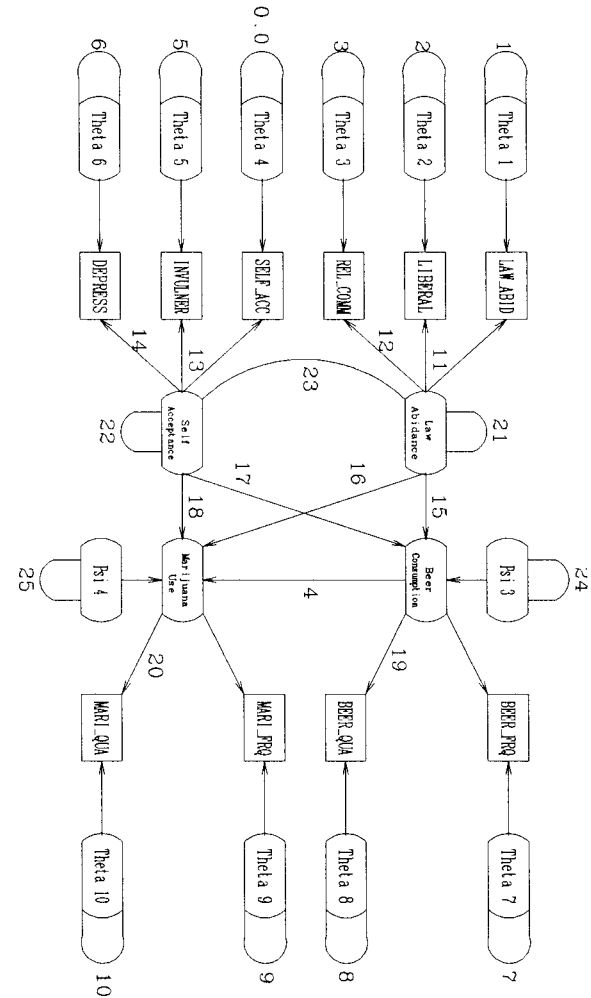


Figure 6. Structural Model for 10 Personality and Drug Use Variables

² In general, it is not strictly correct to analyze a correlation matrix as though it were a covariance matrix. However, many authors have done this in the past, and we shall, in replicating their work, analyze the same data in the same way.

5. EzPATH Diagrams and the PATH1 Language

A. EzPATH Diagrams

I pointed out above that, in their original article, Huba and Harlow (1985) used the same pictorial device (i.e., the Greek letter " Ψ " attached to an arrow) in their path diagram to represent two distinctly different model entities. On the left side of their diagram the letter Ψ stands for a wire, while on the right side it represents considerably more. Although an experienced structural modeler would be able to sort out this inconsistency/ambiguity rather quickly, the newcomer might find their diagram confusing.

The Huba–Harlow example dramatizes a problem which is common in the structural modeling literature – i.e., *confusion created by an absence of clear, and widely accepted, standards in structural modeling diagrams.* (See Steiger, 1988, for a discussion of communication problems in structural modeling.) The Huba and Harlow paper is by no means unusual in this regard. In fact, in many ways it is a model of clarity relative to other papers in the field, and the problem in their diagram is relatively minor.

Ideally, structural diagrams should be completely unambiguous. The fact that they often are not is disturbing, given the negative consequences which ensue. It is difficult to imagine, let alone estimate, the number of hours which have been wasted by researchers attempting to reproduce structural models from inaccurate, misleading, or ambiguous structural diagrams.

We wish to establish rules for path diagrams which will guarantee that the diagram will represent accurately any model which fully accounts for all variances and covariances of all variables, both manifest and latent. Our rules are based on the following considerations.

Path diagrams consist of variables connected by wires and arrows, representing, respectively, undirected and directed relationships. These variables must be endogenous or exogenous. They must also be manifest or latent. Hence any variable can be classified into 4 categories: (a) manifest endogenous, (b) manifest exogenous, (c) latent endogenous, and (d) latent exogenous.

If random variables are related by linear equations, then variables which are endogenous (i.e., appear on the left side of equations) have variances and covariances which are determinate functions of the variables they regress on. For example if X and Y are orthogonal and $W = aX + bY$, then the $\sigma_w^2 = a^2\sigma_x^2 + b^2\sigma_y^2$.

Hence, one way of guaranteeing that a diagram can account for variances and covariances among all its variables is to require: (1) representation of all variances and covariances among exogenous variables, (2) no variances or covariances to be directly represented in the diagram for endogenous variables, and (3) all variables in the diagram be involved in at least one relationship.

There is a significant practical problem with many path diagrams – lack of space. In many cases, there are so many exogenous variables that there is simply not enough room to represent, adequately, the variances and covariances among them. Diagrams which try often end up looking like piles of spaghetti. For a beautiful example of a spaghetti diagram, see page 147 of the excellent text by James, Mulaik, and Brett (1982).

One way of compensating for this problem is to include rules for "default" variances and covariances which allow a considerable number of them to be represented implicitly in the diagram.

These considerations lead to the following rules:

1. Manifest variables are always represented in boxes (squares or rectangles) while latent variables are always in ovals or circles.
2. Directed relationships are always represented explicitly.
3. Undirected relationships need not be represented explicitly. (See rule 10 below regarding implicit representation of undirected relationships.)
4. Directed relationships are represented by arrows between two variables.
5. Undirected relationships, when represented explicitly are shown by a wire from a variable to itself, or from one variable to another.
6. Endogenous variables may never have wires connected to them.
7. Free parameter numbers for a wire or arrow are always represented with integers placed on or slightly above the middle of the wire or arrowline.
8. Fixed values for a wire or arrow are always represented with a floating point number **containing a decimal point**. The number is generally placed on or slightly above the middle of the wire or arrowline.
9. Different statistical populations are represented by a line of demarcation and the words Group 1 (for the first population or group), Group 2, etc., in each diagram section.
10. All exogenous variables must have their variances represented either explicitly or implicitly. If variances and covariances are not represented explicitly, then the following rules hold:

a) For latent variables, variances not explicitly represented in the diagram are assumed to be 1.0, and covariances not explicitly represented are assumed to be 0.

b) For manifest variables, variances and covariances not explicitly represented are assumed to be free parameters each having a different parameter number. These numbers are not equal to any number appearing explicitly in the diagram.

By adopting a consistent standard for path diagrams, we can facilitate clear communication of path models, regardless of what system is used to analyze them. Besides standing on their own as a coherent standard for path diagrams, the above rules for EzPATH diagrams are designed to match the PATH1 language, and allow quick translation from the diagram to the language, and vice-versa.

Within this manual we will adhere to these simplifying conventions. However, the typical EzPATH user will attempt to use the program to reproduce results from published papers employing a wide variety of standards for their path diagrams. In some cases this will create no problems, and the user will be able to translate directly to and from the published path diagram to a PATH1 representation of the model. However, experience has taught me that it is often useful to translate published diagrams into an EzPATH diagram, i.e., one which obeys rules 1–10 above, *before* coding the diagram in the PATH1 language. Frequently the translation process will draw attention to errors or ambiguities in the published diagram.

Here are some examples of the kinds of things you will see in published causal modeling papers.

Figure 7 shows a portion (this is *not* a complete diagram and it does not conform to EzPATH diagramming rules) of a path diagram which is quite typical of what is found in the

literature. Some of the diagram is clear and routine, but what do we make of the symbols D1 and D2? Variable L1 is a latent exogenous variable. It has arrows pointing away from it and no arrows pointing to it. Since, by rule 9 for EzPATH diagrams, all exogenous variables must have their variances explained, the most reasonable assumption is that D1 stands for the variance of latent variable L1. Hence, we modify the diagram to make D1 a parameter attached to a wire from L1 to itself.

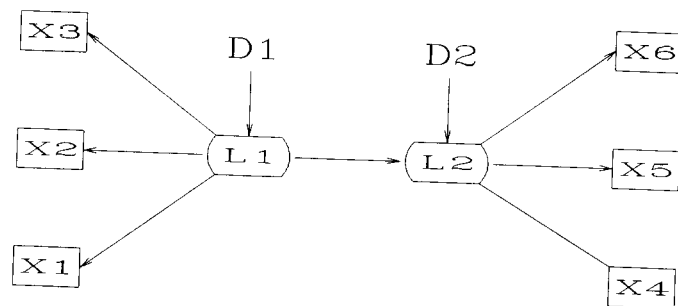


Figure 7. A Section From an Ambiguous Path Diagram

But what are we to make of D2? In the diagram it looks just like D1, but on closer inspection we find it must mean something different. D2 is connected to L2, and L2 is an *endogenous* latent variable. Consequently, the most reasonable interpretation is that D2 represents an "error variance" for latent variable L2. We represent it with an "error latent variable" E2 with variance D2.

The revised path diagram, accurately reflecting the author's model, is shown in Figure 8.

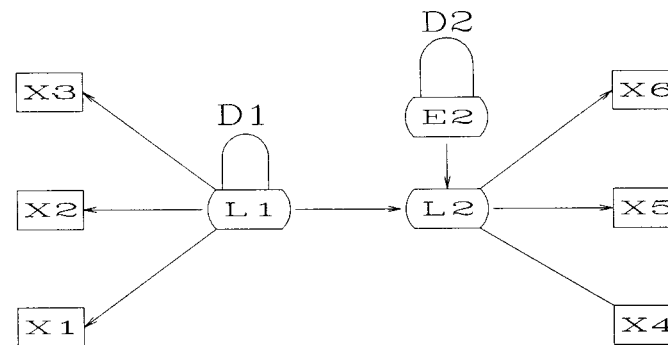


Figure 8. Revised Version of the Diagram in Figure 7

Some path diagrams don't represent the error variance attached to endogenous latent variables at all -- they leave this to the reader to figure out for him/her self. Whenever an endogenous latent variable has no error term, you should suspect that an error latent variable has been left out, especially if your degrees of freedom don't agree with those of the published paper.

In some cases you will have to be creative, tenacious, and lucky to figure out what the author intended. Even the most accomplished and generally careful authors will leave out paths, forget to mention that some values were fixed rather than free parameters, or simply misrepresent the model actually tested. Some times the only way to figure out what the author actually did is to try several models with EzPATH, until you find coefficients which agree with the published values. Sometimes even this approach will not work, because on occasion correlation or covariance matrices are printed incorrectly.

Needless to say, if authors were to adopt EzPATH diagramming rules and/or report their models in the PATH1 language, these problems would be reduced.

B. The PATH1 Language.

PATH1 was designed to allow quick conversion of EzPATH diagrams to a form where they can be read by a computer. Here are the rules for the PATH1 language.

1. Each arrow and wire is represented on a separate line.
2. Blanks never count. They are stripped from the line before parsing.
3. Manifest variable names are represented as a name enclosed within brackets. The name enclosed in brackets must follow the rules for a SYSTAT variable name. The name thus can be at most 8 alphanumeric characters in length. Characters must be upper case. The underscore character "_" is also allowed.

Examples. **[MATH1]**
 [HS_MATH]
 [ANXIETY]

4. Latent variable names are represented by a variable name in parentheses. The name can be up to 20 characters in length. Upper and lower case characters are allowed, and are distinguished. Underscores are allowed, but dashes are not allowed.

Examples. **(Verbal_Intelligence)**
 (Explosive_Strength)

5. Directed relationships are represented in the following form in a PATH1 input line:

VNAME1-<#1>{<#2>->VNAME2

where **VNAME1** and **VNAME2** are valid manifest or latent

variables adhering to rules 3 and 4 above,

<#1> is an integer representing the coefficient number, and **<#2>** is a real value representing the start value.

If **VNAME1** is omitted, then the first variable in the directed relationship is assumed to be **VNAME1** in the last preceding line having two variable names in it.

<#1> is required if the path has a coefficient which is a free parameter. If the coefficient is a fixed value, **<#1>** is omitted. Otherwise, **<#1>** is the integer value for the parameter number. It must be between 1 and 999 in value.

If the coefficient for the arrow is a free parameter, then **<#2>** is the starting value used during iteration. If the coefficient is fixed, then **<#2>** represents the fixed value. If both **<#1>** and **<#2>** are omitted, then the path is assumed to have a fixed coefficient with a value of 1.

Examples. **(IQ_10)-1->[WECHSLER]**
 (X)-->[Y]

6. Undirected relationships are presented in the following form

VNAME1-<#1>{<#2>-}VNAME2

where **VNAME1**, **VNAME2**, **<#1>**, and **<#2>** are the same as in the preceding section.

If **VNAME1** is omitted, then the first variable in the undirected relationship is assumed to be **VNAME1** in the last preceding line having two variable names in it.

Examples. **(L1)-1-(L2)**

7. Different statistical populations are denoted in the PATH1 language by a **GROUP** statement of the form

GROUP <#>

where <#> is the number of the population. All PATH1 files begin (implicitly) with a **GROUP 1** statement. All statements are assumed to refer to the group referred to in the last **GROUP** statement. The current version of EzPATH does not implement the **GROUP** statement, as multiple group tests are not supported.

8. Blank lines, and any lines beginning with a *, are treated as comment lines, and are not analyzed as PATH1 statements.

As the examples in the tutorial have demonstrated, each element in an EzPATH diagram has an obvious corresponding element in PATH1. However there is one situation where the EzPATH diagram may represent information implicitly which must be expressed explicitly in the PATH1 representation. This is when the model has manifest exogenous variables. As noted in rule 10b for EzPATH diagrams, the diagram allows you the option of not representing all the variances and covariances among exogenous manifest variables explicitly. However, for reasons of computational efficiency, the user must follow one of two courses of action. If the AUTOFIX switch is ON, then no variances and covariances among exogenous manifest variables are to be expressed explicitly in the diagram. If the AUTOFIX switch is OFF, then *all* variances and covariances among exogenous manifest variables must be expressed explicitly in the PATH1 representation. This restriction in the program (rather than the PATH1 language itself) presents few problems in practice.

6. EzPATH Commands

EzPATH Commands are:

ACCURACY
AUTOFIX
CMODEL
ESTIMATE
FACTORMODEL
ITERATIONS
METHOD
MODEL
NUMBER
REPORT
RMODEL
SAVE
SE

These commands are described on the following pages. As with other SYSTAT commands, these may be abbreviated by the first two letters.

EzPATH also supports the full range of SYSTAT commands, including

CHARSET	OPTIONS
FEDIT	OUTPUT
FORMAT	PAGE
FPATH	QUIT
HELP	SUBMIT
NAMES	SWITCHTO
NOTE	USE

These latter commands are described on pages 33–48 of the SYSTAT manual. If you are new to SYSTAT, it would be an excellent idea to study these pages before proceeding.

ACCURACY (Select level of accuracy for the iterative procedure.)

The **ACCURACY** command is used to control the approximate level of accuracy (in terms of number of digits) of a solution attempted during the iterative estimation procedure.

Syntax for this command is:

ACC = <#>

Values between 1 and 9 are acceptable. Default is 3.

AUTOFIX (Handle all variance–covariance relationships for manifest exogenous variables automatically.)

Syntax for this command is:

AUTOFIX = YES | NO

Manifest exogenous variables which are unconstrained in their relations to other manifest variables are referred to in the LISREL model as "x–variables." By the normal EzPATH diagram rules the variances and covariances for such variables must be represented explicitly.

However, a characteristic of such variables is that, because their variance covariance relationships are essentially unconstrained by the model, the estimated variances and covariances under the model turn out to be precisely the same as the observed sample variances and covariances. Since we know in advance the values these estimates will take on, there is no purpose estimating them iteratively. Indeed, including them in the iteration process wastes computing time.

Since the manifest exogenous variables are unconstrained by the model, their variances and covariances do not contribute to the loss function during estimation. Consequently, if these variables (and their variances and covariances) are treated as fixed instead of random, the results of the analysis will be exactly the same.

When **AUTOFIX** is in the **YES** mode, all manifest exogenous variables are automatically treated as fixed variables, and their variances and covariances are treated as fixed parameters equal to the observed sample variances and covariances. Model degrees of freedom are automatically corrected.

If **AUTOFIX** is in the **YES** mode, no manifest exogenous variables may have their variances and covariances specified in a **PATH1** statement. If, on the other hand, **AUTOFIX** is in the **NO** mode, then *all variances and covariances for all manifest exogenous variables must be specified with **PATH1** statements.*

EzPATH will refuse to accept a model which disobeys the restrictions in the preceding paragraph.

Besides being useful in accelerating the processing of causal models with exogenous manifest variables, the **AUTOFIX** command also allows convenient processing of models with fixed exogenous variables.

CMODEL (Automatically create PATH1 commands for a complete covariance matrix)

Syntax for this command is

CMODEL <file name>

CMODEL creates a file called <file name>.CMD containing PATH1 file statements for the entire covariance matrix. This facility can be a timesaver when direct tests on the structure of the covariance matrix are to be performed.

Examples of such tests facilitated by the **CMODEL** command are given in Section 9.E.

ESTIMATE (Begin estimation of a selected model.)

When the estimate command is given, EzPATH begins the statistical estimation procedure. Iteration continues until the conditions for termination are met, maximum number of iterations is reached, an error condition occurs, or the user terminates the model with a user interrupt.

Following the completion of the estimation procedure, technical information is printed, and the results of estimation are output in the PATH1 language.

It is possible to interrupt estimation at any point by pressing the <Ctrl> and <Break> keys simultaneously. If you do this, the results of estimation up to the point when the interrupt occurred are still stored in memory, until you **USE** another data matrix or enter another **MODEL** command. You may resume the estimation by typing **ESTIMATE** again.

If you are using the **METHOD = DUAL** option, and have progressed to the maximum likelihood estimation before interrupting iteration, you should type **METHOD = ML** before resuming execution. Otherwise, the program will begin a least squares estimation rather than a maximum likelihood estimation.

FACTORMODEL (Automatically create PATH1 statements for a complete common factor model.)

Syntax for the command is:

```
FACTORMODEL <file name> [/ |COV|, |NOSTART|]
```

<file name> is a .SYS file. **COV** and **NOSTART** are optional switches whose functions are discussed below.

FACTORMODEL is one of the most powerful and useful commands in EzPATH. It allows the SYSTAT system to construct a complete factor model specification, in the PATH1 language, automatically.

One of the most effective uses of EzPATH is in the construction of measurement models via the type of exploratory–confirmatory factor analysis described by Karl Jöreskog (1987) in his 1978 Presidential Address to the Psychometric Society. This method begins with a fully specified common factor model, which is then pruned of insignificant paths.

Typing a full factor model for a reasonably large number of variables is tedious, to say the least. The purpose of the **FACTORMODEL** command is to relieve the user from this tedium.

FACTORMODEL operates on two .SYS files simultaneously. One file is a correlation or variance–covariance matrix file (which must be actively in use at the time). The second is a factor loading matrix file, which is specified in the **FACTORMODEL** command line.

Typical operation of this command is as follows. First you perform a factor analysis on the correlation or covariance matrix, using the SYSTAT **FACTOR** module. The factor loading matrix (usually a rotated version, which must not be created with the **SORT** option) is **SAVED** to a *.SYS file. Then you **SWITCHTO EzPATH** (the covariance matrix will remain in use) and type

```
FACTORMODEL <file name> [/COV, NOSTART]
```

where <file name> is the name of the .SYS file containing the factor loadings. EzPATH will create a .CMD file with the same name as <file name>. This file will contain the full PATH1 specification for a common factor model with starting values equal to the numerical values in the loadings file. (If you wish to suppress these starting values, use the /NOSTART switch.)

If you use the /COV option, EzPATH will add statements for factor covariances. Factor variances are left unspecified under this option, which means that (1) they are implicitly defined to be 1.0, and (2) factor covariances are also factor intercorrelations.

You can use the **FACTORMODEL** command with a factor loading matrix which you have typed in directly using the EDIT module. However, you must make sure that none of the manifest variable names appears out of order in your factor loading file.

ITERATIONS (Stop estimation after a certain number of iterations.)

Syntax for this command is:

ITERATIONS = <#>

Default is 500.

METHOD (Choose the method of estimation.)

Syntax for this command is

METHOD = LS | ML | DUAL

LS causes ordinary least squares estimates to be produced. **ML** produces maximum likelihood estimates. **DUAL** causes least squares estimates to be produced first, and then uses these estimates starting values in a maximum likelihood estimation. **DUAL** is the default.

MODEL (Input and parse the structural model.)

This command indicates to EzPATH that a model is to be entered. All the following statements will be interpreted as PATH1 model statements until a valid EzPATH or SYSTAT command is encountered. When such a command is given, model entry is terminated immediately, and the command is executed.

Syntax for this command is

MODEL | <variable list> |

Where **<variable list>** is an optional list of variables to be modeled from the variables in the current file. **This list must be given if EzPATH is used to model a subset of the variables in your file.**

NUMBER (Enter the number of observations on which the covariance or correlation matrix was based.)

Syntax for this command is

NUMBER = <#>

REPORT (Control the amount and timing of information printed during iterative estimation.)

The **REPORT** command controls information printed during maximum likelihood and/or least squares iteration. Syntax for this command is

REPORT = <#> [/DETAIL]

Normally, a report is given every <#> iterations. The iteration number and the current value of the function being minimized are printed on the screen. For example

REPORT = 10

will cause the function value to be printed each 10 iterations.

The **/DETAIL** switch causes additional technical information (i.e., the value of each parameter and the gradient of the loss function) to be printed.

RMODEL (Automatically create PATH1 commands for a complete correlation matrix.)

Syntax for this command is

```
RMODEL <file name> [/NOSTART]
```

RMODEL creates a file called <file name>.CMD containing PATH1 file statements for the entire correlation matrix. The model begins with a set of statements creating "alias" latent variables having unit variance. Then, statements creating all possible correlations among these latent variables are added. You can quickly modify this file to create a PATH1 file corresponding to any pattern hypothesis on correlations. This facility can be a time saver when direct tests on the structure of the correlation matrix are to be performed.

As a default, the program inserts values of the sample correlation matrix as starting values. Using the **/NOSTART** switch will cause these values to be left out.

Examples of such tests facilitated by the **RMODEL** command are given in Section 9.D.

SAVE (Place technical information from the latest EzPATH iterative run in files for later processing.)

Syntax for this command is

```
SAVE <file name>
```

If PRINT=LONG has been activated, and the SAVE command is in effect, then:

1. The normalized residuals for the entire covariance matrix are saved in a file called <file name>.RES.
2. The estimated variance/covariance matrix is saved in a COV file called <file name>.HAT.

SE (Select whether or not standard errors are to be computed.)

Syntax for this command is

SE = YES | NO

Default is **NO**.

If maximum likelihood estimates are obtained, and the model Hessian is positive definite, then standard errors for the estimates will be calculated if you have indicated SE = YES.

STANDARDIZE (Prints coefficients for latent variables standardized to have unit variance).

Syntax for this command is

STANDARDIZE = YES | NO

Default is **NO**

If the standardization option is invoked, EzPATH prints, in addition to the ordinary coefficients, a "standardized solution" containing values for latent variables standardized to have variances of 1. Manifest variables are left in the original metric.

Standard errors are not printed for a standardized solution.

7. Technical and Theoretical Aspects of EzPATH

In this section we present some of the more important technical and theoretical details on the models, methods, and techniques underlying EzPATH. It is not *absolutely* necessary to understand any of this material to use the program. However, we recommend that all prospective users of the program at least browse through this section.

Two aspects of EzPATH are of particular theoretical interest. First, the structural model equation system employed by EzPATH is novel. Second, EzPATH provides theory for, and a practical implementation of, asymptotic maximum likelihood statistical estimation and confidence intervals for measures of model fit.

A. Models and Methods

Structural equation models have achieved increasing popularity in the social sciences. Much of the credit for this popularity can be attributed to the flexibility and power of the methods themselves. Equally important has been the availability of computer software for performing the modeling process.

An enormous amount of material has been written on structural models in the last 5 years alone. There are now numerous textbooks and monographs for the beginner. Users in need of such an elementary account are referred to books by Long (1983a,b), James, Mulaik and Brett (1982), Kenny (1979), Everitt (1984), among others. All of these books have significant virtues, and all are relatively brief. The reader with a serious interest in the subject should probably at least browse through all of these books.

For a very interesting debate on the value of structural models in the social sciences, I strongly recommend the Summer 1987 issue of the *Journal of Educational Statistics*,

which contained a critique of path analysis by D.A. Freedman, and responses to that critique by a number of writers.

I will not attempt to generate another such textbook here. (However, a subsequent chapter will deal with some important topics which are seldom discussed in the existing textbooks.) Rather, I will provide a relatively terse discussion of important technical aspects of EzPATH.

We begin with a discussion of statistical models. The model behind EzPATH is best understood in historical context, and so we begin with a review of several important models for the analysis of covariance structures.

In his 1986 review of developments in structural modeling, Bentler described 3 general approaches to covariance structure representations. The first and most familiar involved integration of the psychometric factor analytic (FA) tradition with the econometric simultaneous equations model (SEM). This approach, originated by a number of authors including Keesling, Wiley, and Jöreskog was described by Bentler with the neutral acronym FASEM. The well-known LISREL model is of course the best known example of this approach.

The LISREL model can be written in three interlocking equations. Perhaps the key equation is the structural equation model, which relates latent variables.

$$\eta = B\eta + \Gamma\xi + \zeta \quad (1)$$

The endogenous, or "dependent" latent variables are collected in the vector η , while the exogenous, or "independent" latent variables are in ξ . B and Γ are coefficient matrices, while ζ is a random vector of residuals, sometimes called "errors in equations" or "disturbance terms." The elements of B and Γ would be path coefficients for directed relationships among latent variables. It is assumed in general that ζ and ξ are uncorrelated, and that $I - B$ is of full rank.

Because usually η and ξ are not observed without error, there are also factor model (or "measurement model") equations to account for measurement of these latent variables through manifest variables. The measurement models for the two sets of latent variables are

$$y = \Lambda_y \eta + \epsilon \quad (2)$$

and

$$x = \Lambda_x \xi + \delta \quad (3)$$

With the assumptions that (1) ζ is uncorrelated with ξ , (2) ϵ is uncorrelated with η , (3) δ is uncorrelated with ξ , (4) ζ , ϵ , and δ are mutually uncorrelated, (5) B has zeroes in its diagonal, and (6) $I - B$ is of full rank, we find that the population covariance matrix Σ can be written as

$$\Sigma = \begin{pmatrix} \Sigma_{yy} & \Sigma_{yx} \\ \Sigma_{xy} & \Sigma_{xx} \end{pmatrix} \quad (4)$$

where

$$\Sigma_{yy} = \Lambda_y (I - B)^{-1} (\Gamma \Phi \Gamma' + \Psi) (I - B')^{-1} \Lambda_y' + \theta_\epsilon \quad (5)$$

$$\Sigma_{xx} = \Lambda \Phi \Lambda' + \theta_\delta \quad (6)$$

$$\Sigma_{yx} = \Lambda_x \Phi \Gamma' (I - B')^{-1} \Lambda_y' \quad (7)$$

and Φ , Ψ , θ_ϵ , and θ_δ are the covariance matrices for ξ , ζ , ϵ , and δ respectively.

This model reduces to a number of well-known special cases. For example, if there are no y-variables, then the model reduces to the common factor model, as can be seen from Equation (7).

An important aspect of the LISREL approach is that, in using it, variables must be arranged according to type. Manifest and latent, "exogenous" and "endogenous" variables are

used in different places in different equations. Moreover, LISREL's typology for manifest variables is somewhat different from that used by other models. Specifically, in LISREL a manifest variable is designated as x or y on the basis of the type (exogenous or endogenous) of latent variable it loads on.

It is, of course, possible to translate models from a path diagram representation of a model to a LISREL model. However, this is not always easy. In some well known cases special strategies must be used to "trick" the LISREL model into analyzing a path diagram representation. For example, LISREL does not allow direct representation of a path in which an arrow goes from a manifest exogenous variable to a latent endogenous variable. Consequently a "dummy" latent variable (identical to the manifest variable) must be created in such cases.

In his review, Bentler (1986) referred to the models of McArdle (1978) and Bentler and Weeks (1979) as "generic" approaches, in that their emphasis was on the distinction between independent (exogenous) and dependent (endogenous) variables, rather than manifest and latent variables.

McArdle (1978) proposed an approach which was considerably simpler than the LISREL model. This approach, in essence, did not require any partitioning of variables into types. Only two matrices needed to be defined, one representing directed relationships among variables, the other undirected relationships. McArdle's approach, which he called the RAM model, could easily be tested as a special case of McDonald's COSAN model.

McArdle's specification was innovative, and offered substantial benefits. It allowed path models to be grasped and fully specified in their simplest form -- as linear equations among manifest and latent variables. Instead of 18 model matrices, and a plethora of different variable types, one only needed 3 matrices!

Ironically, it took some time for McArdle's work to gain widespread acceptance, and it was some time before a detailed algebraic treatment (McArdle and McDonald, 1984) surfaced.

We begin with a brief description of the McDonald's COSAN model. Let Σ be a population variance-covariance matrix for a set of variables. The COSAN model (McDonald, 1978) holds if Σ may be expressed as

$$\Sigma = F_1 F_2 \dots F_k P F_k' \dots F_2' F_1' \quad (8)$$

where P is symmetric, and any of the elements of any F matrix or P may be constrained under the model to be a function of the others, or to be specified numerical values. As a powerful additional option, any square F matrix may be specified to be the *inverse* of a patterned matrix. This "*patterned inverse*" option is critical for applications to path analysis. A COSAN model with k F matrices is referred to as "a COSAN model of order k ."

Obvious special cases are: Orthogonal and oblique common factor models, confirmatory factor models, and patterned covariance matrices.

McDonald's COSAN model is a powerful and original approach which offers many benefits to the prospective tester of covariance structure models. Testing and estimation for the model were implemented in a computer program called, aptly enough, COSAN (See Fraser and McDonald, 1988 for details on the latest version of this program, which has been available since 1978).

In 1978, J.J. McArdle proposed some simple rules for translating any path diagram directly to a structural model. In collaboration with McDonald, he proposed an approach which yielded a model directly testable with the COSAN computer program.

McArdle's approach is based on the following covariance structure model, which he has termed the RAM model:

Let v be a $(p+n) \times 1$ random vector of p manifest variables and n latent variables in the path model, possibly partitioned into manifest and latent variables subsets in ${}_p m_1$ and ${}_n l_1$, respectively, in which case $v = [m' : l']'$. (This partitioning is somewhat convenient, but not necessary.) For simplicity assume all variables have zero means. Let F be a matrix of multiple regression weights for predicting each variable in v from the $p+n-1$ other variables in v . (F will have all diagonal elements equal to zero.) In general, some elements of F may be constrained by hypothesis to be equal to each other, or to specified numerical values (often zero). Let r be a vector of residuals. The path model may then be written

$$v = Fv + r \quad (9)$$

In path models, all endogenous variables are perfectly predicted through the paths leading to them. Consequently, elements of r corresponding to endogenous variables in v will be null. The matrix F contains the regression coefficients normally placed along the arrows in a path diagram. f_{ij} is the path coefficient from v_j to v_i . If a variable v_i is exogenous, i.e., has no arrow pointing to it, then row i of F will be null, and $r_i = v_i$. Hence, the non-null elements of the variance covariance matrix of r will be the coefficients in the "undirected" relationships in the path diagram. Define $P = E(rr')$. Furthermore, let $W = E(vv')$, $\Sigma = E(mm')$.

We wish to examine the implications of Equation (9) for the structure of Σ , the variance-covariance matrix of the manifest variables. Regardless of whether the manifest and latent variables were partitioned in v , it is easy to construct a "filter matrix" J which carries v into m . If the variables in v are partitioned into manifest and latent variables, we have $J = [I : 0]$,

$$m = Jv, \quad (10)$$

and we may write

$$\Sigma = E(mm') = JE(vv')J' = JWJ'. \quad (11)$$

Since (assuming $I-F$ is nonsingular) Equation (9) may be rewritten in the form

$$v = (I-F)^{-1}r, \quad (12)$$

we obtain

$$W = (I-F)^{-1}P(I-F)^{-1} = (F-I)^{-1}P(F-I)^{-1}. \quad (13)$$

Equations (11) and (13) imply

$$\Sigma = J(F-I)^{-1}P(F-I)^{-1}J' \quad (14)$$

Equation (14) shows that any path model may be written as a COSAN model of order 2, where $F_1 = J = [I : 0]$, and $F_2 = (F-I)^{-1}$.

McArdle's formulation may thus be characterized as follows:

For convenience order the manifest variables in the vector m , and the latent variables in the vector l . The path model is then tested as a COSAN model of order 2, in which

$$F_1 = [I_p : p_0 0_n]$$

F_2 is the inverse of a square matrix B of "of directed relationships." B is constructed from the path diagram as follows. Set all diagonal entries of B to -1 . Examine the path diagram for arrows. For each arrow pointing from v_j to v_i , record its path coefficient in position b_{ij} in matrix B .

P contains coefficients for "undirected" paths between variable v_i and v_j recorded in positions P_{ij} .

Obviously, EzPATH could have been written around the elegant and straightforward RAM model. The approach would require simply creating a list of manifest and latent variables, ordering them, and filling the matrices B and P with coefficients obtained by parsing PATH1 model statements.

However, the RAM model is somewhat wasteful in terms of the size of some of its matrices. Bentler and Weeks (1979) produced an alternative model which is somewhat more efficient in the size of its matrices. Specifically, the F_2 and P matrices are quite large in the RAM model, and have a large number of zero elements. Bentler and Weeks showed how, in situations where there are no manifest exogenous variables (i.e., all manifest variables have at least one arrow pointing to them), the McArdle-McDonald approach may be modified to reduce the size of the model matrices.

Partition $v' = [m_n \ l_n \ l_x]$, where the subscripts x and n refer to "exogenous" and "endogenous," respectively.

Then we may write $v = Fv + r$ in a partitioned form as

$$\begin{array}{cccccc} m_n & & F_1 & F_2 & F_3 & & m_n & & 0 \\ l_n & = & F_4 & F_5 & F_6 & & l_n & + & 0 \\ l_x & & 0 & 0 & 0 & & l_x & & l_x \end{array} \quad (15)$$

Now define n as a vector containing all the endogenous, or "dependent" variables. $n' = [m_n \ l_n]$.

We may then write

$$n = B_0 n + \Gamma l_x \quad (16)$$

where

$$B_0 = \begin{array}{ccc} F_1 & F_2 & \\ F_4 & F_5 & \end{array} \quad \Gamma = \begin{array}{c} F_3 \\ F_6 \end{array} \quad (17)$$

We may now proceed with an algebraic development similar to the RAM–COSAN equations.

$$\underline{n} = B_0 \underline{n} + \Gamma \underline{l}_x \quad (18)$$

$$(I - B_0) \underline{n} = \Gamma \underline{l}_x \quad (19)$$

$$\underline{n} = (I - B_0)^{-1} \Gamma \underline{l}_x \quad (20)$$

$$\underline{m}_n = [I : 0] (I - B_0)^{-1} \Gamma \underline{l}_x = J (I - B_0)^{-1} \Gamma \underline{l}_x \quad (21)$$

whence, letting $G = [I : 0]$, $F_2 = (I - B_0)^{-1}$, $F_3 = \Gamma$, and $P = E(\underline{l}_x \underline{l}_x')$

$$\Sigma = GF_2 F_3 P F_3' F_2' G' \quad (22)$$

where G is a filter matrix similar to F_1 in the McArdle–McDonald specification, $F_2 = B_2^{-1}$, where B_2 is a matrix containing path coefficients for directed relationships among endogenous variables only, and having -1 as each diagonal element, F_3 contains path coefficients from exogenous variables to endogenous variables only, and P contains coefficients for undirected relationships, i.e., the variance–covariance parameters for the latent exogenous variables.

This clever algebraic refinement allowed some of the virtues of the McArdle approach to be retained, while expressing the essential relationships in smaller matrices. (Notice how several of the null submatrices in Equation 15 are eliminated.) However, this model also had some minor drawbacks. It required partitioning variables into exogenous and endogenous types, and it did not allow direct expression of manifest exogenous variables.

An alternative model allows us to treat manifest exogenous variables explicitly. If we add a vector of manifest variables to each of the two variable lists in the Bentler–Weeks (1979) model, and modify the regression coefficient matrices accordingly, we arrive at the model used in EzPATH. In this model, which is similar to one given by Bentler and Weeks (1980), we partition variables into two groups.

Let \underline{m}_x be a vector of manifest exogenous variables. Let $\underline{s}_1' = [m_n', m_x', l_n']$, and $\underline{s}_2' = [l_x', m_x']$.

Then we may write

$$\underline{s}_1 = B \underline{s}_1 + \Gamma \underline{s}_2 \quad (23)$$

where

$$B = \begin{bmatrix} F_1 & 0 & F_2 \\ 0 & 0 & 0 \\ F_4 & 0 & F_5 \end{bmatrix} \quad (24)$$

and

$$\Gamma = \begin{bmatrix} F_3 & F_7 \\ 0 & I \\ F_6 & F_8 \end{bmatrix} \quad (25)$$

Assuming a nonsingular $I - B$, we may rewrite Equation 23 as

$$\underline{s}_1 = (I - B)^{-1} \Gamma \underline{s}_2 \quad (26)$$

Let G be a filter matrix which extracts the manifest variables from \underline{s}_1 , and let $P = E(\underline{s}_2 \underline{s}_2')$ be the covariance matrix for \underline{s}_2 .

Then

$$\underline{m} = G \underline{s}_1 = G(I - B)^{-1} \Gamma \underline{s}_2 \quad (27)$$

and we arrive at the following model for covariance structure:

$$\Sigma = G(B - I)^{-1} \Gamma P \Gamma'(B' - I)^{-1} G' \quad (28)$$

The model of Equation (28) allows direct correspondence between all permissible PATH1 statements and the algebraic model. There is no need to concoct dummy latent variables. All possible types of relationships among manifest and latent variables are accounted for. After a model is complete, all variables can immediately be assigned to one of the 4 vectors m_n , m_x , l_n , or l_x . All coefficients for (arrows) are then assigned to the matrices F_1 through F_8 . The column index for a variable (in any of these 8 matrices) represents the variable from which the arrow points, the row index the variable to which the arrow points. Coefficients for wires are represented in a similar manner in the matrix P .

The model of Equation 28 sacrifices some of the simplicity of the RAM model, because variables must be assigned to 4 types before the location of model coefficients can be determined. However, in our typology and with the EzPATH diagramming rules the typing of each variable into one of 4 categories can be determined by looking *only at that variable in the path diagram*. Because two headed arrows are eliminated, a variable is endogenous if and only if it has an arrowhead directed toward it. A variable is latent if and only if it appears in an oval or circle. (If it is not already obvious, let me point out that with two headed arrows one must look away from the variable of interest to determine if the variable is endogenous, because an arrowhead attached to the variable and pointing to it might be two-headed! Not only is the EzPATH system less cluttered, but it is also visually more efficient.)

Before we close this section, two points should be emphasized. First, it is not clear which of the above models is, in any overall sense, "superior" to the others. I chose the EzPATH model of Equation 28 primarily because it offered, to me, a good trade-off between certain conceptual and computational advantages. However in the course of programming EzPATH, I found, for reasons too complex to describe here, that there were also definite advantages, both conceptual and computational, in each of the other model formulations.

Second, it is possible to express some of the models as special cases of the others. For example, the LISREL model can be written easily as a RAM model and tested with COSAN. To see why, suppose that the manifest and latent variables were ordered in the v of Equation 9 so that $v' = [y' \ x' \ \eta' \ \xi']$. Then it follows immediately that we may write $v = F^*v + r^*$, where

$$F^* = \begin{bmatrix} 0 & 0 & \Lambda_y & 0 \\ 0 & 0 & 0 & \Lambda_x \\ 0 & 0 & B & \Gamma \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (29)$$

and $r^* = [\epsilon \ \delta \ \zeta \ \xi]$.

If we define P^* as the covariance matrix of r^* , then clearly we can test any LISREL model as a COSAN model of the form

$$\Sigma = G(F^* - I)^{-1}P(F^* - I)^{-1}G' \quad (30)$$

where G is a matrix which filters x and y from v .

B. Statistical Estimation

In the preceding section, I outlined the statistical model for Σ which EzPATH attempts to fit to the sample data. If the model fits perfectly in the population, then Equation (28) holds. This is, of course, extremely unlikely to happen.

It is, in fact, virtually certain that Equation (28) does *not* hold exactly for your statistical population, and that in fact an additional error term E_{pop} should be added to the right side of the equation. The size of the elements of this error matrix would reflect how badly a particular model fit in the population. You could find out what E_{pop} was if you somehow knew Σ . (You would simply input Σ to EzPATH and fit your model to it.) If you did, you would be faced with a difficult problem of exactly how to quantify the information in E .

There is an additional complication. In practice, you do not know Σ . You only have S , an estimate of it from sample data. It is this estimate, usually the ordinary sample covariance matrix based on N independent observations, which we attempt to fit with EzPATH.

Consequently, in practice we attempt to fit S rather than Σ with the model of Equation 28, and we have as a result of this model fitting procedure a *sample* matrix of residuals E_{samp} . In general the object of the estimation process is to make the elements of E_{samp} as "small as possible" in some sense. This notion of "smallness" is quantified in a "discrepancy function."

Define θ as the current vector of free parameter values. Let C_θ be our best attempt at reproducing S by minimizing a particular discrepancy function with the free parameter values in θ . The corresponding values in the population are Σ_θ and Σ .

$$C_\theta = G(B_\theta - I)^{-1} \Gamma_\theta P_\theta \Gamma_\theta' (B_\theta' - I)^{-1} G' \quad (31)$$

where B_θ , Γ_θ , and P_θ are obtained by inserting "best values" for the free parameters. The discrepancy function $F(S, C_\theta)$ is a measure on S and C_θ .

In general, if a model is identified (see Section 8.C below), minimization of a discrepancy function satisfying the following three restrictions will lead to consistent estimates for the elements of θ :

$$\begin{aligned} F(S, C_\theta) &\geq 0 \\ F(S, C_\theta) &= 0 \text{ if and only if } S = C_\theta \\ F(S, C_\theta) &\text{ is continuous in } S \text{ and } C_\theta \end{aligned}$$

One simple measure of how badly C_θ fits S is to examine the sum of squared elements of E_{samp} . The function is known as the Ordinary Least Squares discrepancy function, and may be written

$$F_{\text{OLS}}(S, C_\theta) = 1/2 \text{Tr}(S - C_\theta)^2 \quad (32)$$

The OLS discrepancy function has a number of difficulties, summarized nicely by Everitt (1984). In particular, it is not *scale free* -- different scalings of the manifest variables can produce different discrepancy function values. Moreover, when calculated on sample discrepancies, simple sums of squares may be inappropriate, because the elements of S are not independent random variables, and because they usually have different sampling variances.

The Generalized Least Squares discrepancy function compensates for these problems by, in effect, standardizing each element of E_{samp} by an estimate of its variability. The resulting discrepancy function is

$$F_{\text{GLS}} = 1/2 \text{Tr}[(S - C_\theta)S^{-1}]^2 \quad (33)$$

A more complex function is the Maximum Likelihood discrepancy function. This function may be written

$$F_{\text{ML}}(S, C_\theta) = \ln |C_\theta| - \ln |S| + \text{Tr}(SC_\theta^{-1}) - p \quad (34)$$

where $\text{Tr}()$ denotes the trace operator, $|S|$ the determinant, and p is the number of manifest variables.

If S is has a Wishart distribution, the model is identified, and θ has t free parameters, then under fairly general conditions $(N-1)F_{\text{ML}}(S, C_\theta)$ has an asymptotic chi-square distribution with $p(p+1)/2 - t$ degrees of freedom.

This chi-square statistic, often described as a "goodness of fit" statistic (but perhaps more accurately called a "badness of fit" statistic) allows us to test statistically whether a particular model fits Σ *perfectly* in the population (i.e., whether $\Sigma = \Sigma_\theta$). There is a long tradition of performing such a test, although it is becoming increasingly clear that the procedure is seldom appropriate.

Browne (1974, 1984) discussed the properties of an alternative discrepancy function

$$F(S, C_\theta | C_\theta^{-1}) = 1/2 \text{Tr} [(S - C_\theta) C_\theta^{-1}]^2 \quad (35)$$

and showed that, under typical assumptions for maximum likelihood estimation, the two statistics, $(N-1) F(S, C_\theta | C_\theta^{-1})$ and $(N-1) F_{ML}(S, C_\theta)$, will converge stochastically, and will both be distributed as chi-square variates. Moreover, as $n \rightarrow \infty$, the probability that the two discrepancy functions F_{ML} and $F(S, C_\theta | C_\theta^{-1})$ will be minimized by different θ vectors converges to zero.

In practice, then, one may essentially always obtain maximum likelihood estimates by minimizing $F(S, C_\theta | C_\theta^{-1})$ rather than F_{ML} .

C. Model Identification

For practical purposes it is usually not enough to have a particular model which, when expressed in the framework of Equation (28), reproduces Σ . For a model to be of much conceptual or practical value, its parameters must be identified. That is, there must exist *only one* parameter vector θ for which $\Sigma = \Sigma_\theta$.

Perhaps the simplest example of a covariance structure model which is not identified is a common factor analysis model with two manifest variables and one common factor. In this case (assuming the common factor has a variance of 1) the covariance structure model becomes

$$\Sigma_\theta = ff' + U^2$$

In this case the parameter vector θ has 4 elements, the two elements in f and the two diagonal elements of U^2 . Suppose Σ is

$$\Sigma = \begin{bmatrix} 1 & .5 \\ .5 & 1 \end{bmatrix}$$

We can see that if $f' = [.7071 \ .7071]$, and the diagonal elements of U^2 are both .5, then $\Sigma = \Sigma_\theta$, and the model fits perfectly. In this case $\theta' = [.7071 \ .7071 \ .5000 \ .5000]$. But there are other values of θ which will reproduce Σ equally well. In fact there are infinitely many such values. For example, let $\theta' = [.9000 \ .5555 \ .1900 \ .6914]$

If we restrict U^2 to be positive definite, clearly any two values for the first two elements of θ which have a product of .5, and are both less than one in absolute value will produce a discrepancy function value of zero. The diagonal elements of U^2 are then obtained by subtracting the square of the corresponding element of f from 1.0.

Note that this is *not* a problem of the well known "rotational indeterminacy" in factor analysis. (With only one factor, there is no rotation.) Rather it is an example of a lesser known phenomenon, namely, that the elements of U^2 may not be identified in the common factor model. If U^2 is not identified, then there may exist common factor patterns which reproduce Σ equally well, but which are not obtainable from each other by rotation.

Even in the relatively comfortable confines of the common factor model, the phenomena of model identification are not well understood. Some of the most significant textbooks on factor analysis have failed to ever mention the problem. Moreover, several authoritative figures in the history of psychometrics have produced "results" on model identification in factor analysis which they have later had to retract or correct.

In general, necessary and sufficient conditions for identification are not available. However, it is often possible to determine that a model is *not* identified by showing that a necessary condition is violated.

There are some results available on when U^2 in the factor model is definitely *not* identified. One of the best-known was given by Anderson and Rubin (1956). They showed that if, in unrestricted factor analysis, *under any orthogonal or*

oblique rotation, there existed a factor pattern with only 2 non-zero elements in any column, then U^2 is not identified. Clearly then, if such a situation exists (see Everitt 1984, pp. 45–49 for an example), additional constraints will have to be imposed to yield an identified solution. (For another example of this situation, examine carefully the correlation matrix discussed on page 409 of the SYSTAT manual.)

The Anderson–Rubin result has an important implication which is often overlooked in discussions of the identification issue. Namely, *it may not be possible to prove identification in the population* without knowing Σ ! In other words the same model may be identified for one Σ , but not for another. You cannot prove identification merely by counting equations and unknowns.

For some (relatively simple) models, it may be possible to prove identification by deriving unique equations, showing each parameter as a function of the elements of Σ . Unfortunately this approach is often impractical, and so checking for identification usually involves two stages.

First very obvious sources of lack of identification should be removed. The most obvious source of underidentification in path models occurs when the measurement scale of an exogenous latent variable is left indeterminate. Consider the oblique common factor model, which can be written

$$\Sigma = F\Psi F' + U^2$$

The variances of the common factors are found on the diagonal of Ψ . The factor loading coefficient for manifest variable i on factor j is found in element F_{ij} . It is easy to show that unless restrictions are imposed on this model, the variance for factor j and the loadings on this factor are jointly indeterminate. To see why, suppose we were to multiply all the factor variances by 2. If we were to multiply all the columns of F by $.7071$, we would have exactly the same Σ . More generally, if we were to scale the diagonal of Ψ with a diagonal scaling matrix D , we could compensate by scaling

the columns of F with D^{-1} . In other words, for positive definite D ,

$$\begin{aligned}\Sigma &= F\Psi F' + U^2 \\ &= (FD^{-1})(D\Psi D)(D^{-1}F) + U^2 \\ &= F^*\Psi^*F'^* + U^2.\end{aligned}$$

so that for any F and Ψ there are infinitely many F^* and Ψ^* which reproduce Σ equally well.

There are several ways of eliminating this problem in practice. One way (the one I recommend) is to fix the variances of the exogenous latent variables at 1. (This fix may not be sufficient in all cases.) Another approach is to apply some constraint to the factor loading coefficients themselves. This approach is popular in structural models where the main interest is in the relations between latent variables. In this case, identification is often obtained by fixing one of the coefficients on a particular variable to 1.

Once obvious sources of identification have been eliminated, it is productive to examine whether either of the following necessary (and easily tested) conditions for identification is violated.

1. The number of degrees of freedom for the model must be nonnegative. That is $p(p+1)/2 \geq t$, where p is the order of Σ , and t is the number of free parameters in the model.
2. The Hessian (the matrix of second derivatives of the discrepancy function with respect to the parameters) must be positive definite.

Violation of either of these conditions can indicate an identification problem, and EzPATH warns the user if they are violated.

D. Unconstrained Minimization Technique

EzPATH produces its estimates for the elements of θ by minimizing a discrepancy function under choice of θ . The problem of finding the θ which minimizes $F(S, C_\theta)$ is certainly a difficult one. It is, in fact, non-trivial even when θ has only one element!

"Unconstrained minimization," i.e., the minimization of a function of θ , is a major area in the field of numerical analysis. The interested reader is urged to read an especially clear treatment of this area is given by Dennis and Schnabel (1983). Our description here is brief and non-technical.

The discrepancy function is minimized by an iterative process. We start with initial estimates (we refer to them as "starting values") for the elements of θ . On each iteration, the current value of the function is calculated, and the program estimates, using derivatives, which direction of change for θ will produce a further decrease in the discrepancy function. θ is changed in that direction by an initial amount (called the "step length"), and the function is recalculated. It may be that, according to certain criteria, the initial step went either too far, or not far enough. In that case the step length may go through several adjustments during an iteration.

Once a new θ is found which has reduced the discrepancy function by a reasonable amount, the whole cycle is repeated until at least one of several *stopping criteria* are met:

1. The discrepancy function is extremely close to zero.
2. An iteration fails to change any of the elements of θ more than some very small percentage.
3. The *relative* gradient of the discrepancy function (the relative rate of change of the discrepancy function divided by the relative change in θ) is effectively a null vector. This means that either a global or local minimum has been found. [Some programs base stopping criteria on the absolute gradient. This can lead to premature termination of the algorithm in cases where the discrepancy function at the

minimum is small. The absolute gradient criterion has a number of other shortcomings as well. See Dennis & Schnabel (1983, p. 159) for details.]

To prevent overflow problems to rounding error in internal calculations, our algorithm reverts to an absolute gradient criterion in cases where the discrepancy function is less than 10^{-9} .

In general, with path models we find that the more elements there are in θ , the more difficult it is to find the actual θ which minimizes the discrepancy function. Thus, all other things being equal, we would expect to need more iterations to converge to a solution for a large problem than for a small one. Unfortunately, the larger the problem, the longer each iteration tends to take. Consequently, good initial estimates can be very important for large problems. Indeed, without good initial estimates, even the best "state of the art" non-linear optimization routine will fail to find solutions for some problems.

EzPATH uses the algorithms given in the appendix of Dennis and Schnabel (1983), and also described in Schnabel, Koontz, and Weiss (1985), to produce its minimization. In particular, the factored secant, BFGS update method, with linesearch algorithm A.6.3.1 was used, and suggestions on pages 281–282 of Dennis and Schnabel (1983) were implemented to reduce storage requirements. The line search algorithm is a backtracking line search using safeguarded quadratic interpolation for the first backtrack and safeguarded cubic interpolation for any subsequent backtracks at each iteration.

E. Fit Indices

Besides the chi-square value and its probability level, EzPATH prints a number of indices of fit which can be used to interpret how well a model fits the data. The indices printed here are all single model indices, i.e., values which can be computed from a single model tested on one data set. Several of these indices are new and are my own

creation, and so I will describe the rationale behind them in a fair amount of detail. Full technical details are given in Steiger (1989).

1. General Theoretical Orientation

When we attempt to assess how well a model fits a particular data set, we must realize at the outset that the classic hypothesis-testing approach is inappropriate. Consider common factor analysis. When maximum likelihood estimation became a practical reality, the chi-square "goodness of fit" statistic was originally employed in a sequential testing strategy. According to this strategy, one first picked a small number of factors, and tested the null hypothesis that this factor model fit the population Σ perfectly. If this hypothesis was rejected, the model was assumed to be too simple (i.e., to have too few common factors) to fit the data. The number of common factors was increased by one, and the preceding procedure repeated. The sequence continued until the hypothesis test failed to reject the hypothesis of perfect fit.

Steiger and Lind (1980) pointed out that this logic was essentially flawed, because, for any population Σ (other than one constructed as a numerical example directly from the common factor model!) the a priori probability is essentially 1 that the common factor model will not fit perfectly so long as degrees of freedom for the chi-square statistic were positive.

In essence, then, population fit for a covariance structure model with positive degrees of freedom is never really perfect. Testing whether it is perfect makes little sense. It is what statisticians sometimes call an "accept-support" hypothesis test, because accepting the null hypothesis supports what is generally the experimenter's point of view, i.e., that the model does fit.

Accept-support hypothesis tests are subject to a host of problems. In particular, of course, the traditional priorities between Type I and Type II error are reversed. If the proponent of a model simply performs the chi-square test with low enough power, the model can be supported. As a natural consequence of this, hypothesis testing approaches to the assessment of model fit *should* make some attempt at power evaluation. Steiger and Lind (1980) demonstrated that performance of statistical tests in common factor analysis could be predicted from a noncentral chi-square approximation. A number of papers dealing with the theory and practice of power evaluation in covariance structure analysis have been published (Matsueda & Bielby, 1986; Satorra and Saris, 1985; Steiger, Shapiro, & Browne, 1985). Unfortunately, power estimation in the analysis of a multivariate model is a difficult, somewhat arbitrary procedure, and such power estimates have not, in general, been reported in published studies.

The main reason for evaluating power is to gain some understanding of precision of estimation in a particular situation. An alternative (and actually more direct) approach to the evaluation of precision is to *construct a confidence interval on the population noncentrality parameter* (or some particularly useful function of it). This approach, first suggested in the context of covariance structure analysis by Steiger and Lind (1980) offers two worthwhile pieces of information at the same time. It allows one, for a particular model and data set to express (1) how bad fit is in the population, and (2) how precisely the *population* badness of fit has been determined from the *sample* data.

2. Noncentrality based parameter estimates and confidence intervals

Let S be the sample covariance matrix based on N observations, and for notational convenience, define $n = N - 1$. C_{θ} is the attempt to reproduce S with a

particular model and a particular parameter vector θ , and C_{ml} the value of C_θ obtained when θ is the vector of maximum likelihood estimates obtained by minimizing the discrepancy function of Equation 34. Equations 34 and 35 give two alternative chi-square statistics for testing structural models. Suppose one has obtained maximum likelihood estimates. Then $nF(S, C_\theta | C_\theta^{-1})$ has an approximate asymptotic noncentral chi-square distribution with $p(p+1)/2 - t$ degrees of freedom, where t is the number of free parameters in the model, and p is the order of S . The noncentrality parameter is nF^* , where

$$F^*(\Sigma, \Sigma_{ML} | \Sigma_{ML}^{-1}) = 1/2 \text{Tr}[(\Sigma - \Sigma_{ML})\Sigma_{ML}^{-1}]^2 \quad (36)$$

F^* is the value of the statistic in equation (35) obtained if S is replaced by the population covariance matrix Σ , and maximum likelihood estimation is performed on Σ instead of S , yielding Σ_{ML} . Hence, for the quadratic form statistic, the noncentrality parameter is in effect the "population badness of fit statistic."

Interestingly, if one divides by n , one obtains a measure of population badness of fit which depends only on the model, Σ , and the method of estimation.

If one has a single observation from a noncentral chi-square distribution, it is very easy to obtain an unbiased estimate of the noncentrality parameter. By well known theory, if noncentral chi-square variate X has noncentrality parameter Λ and degrees of freedom v , the expected value of X is given by

$$E(X) = v + \Lambda,$$

whence it immediately follows that an unbiased estimate of Λ is simply $X - v$. Consequently a "large sample unbiased" estimate of F^* in Equation 35 is $(X - v)/n$. Since F^* can never be negative, the simple unbiased estimator is generally modified in practice by converting negative values to zero. The estimate

$$F^+ = \text{Max} \{[(X-v)/n], 0\} \quad (37)$$

is the result.

It is also possible, by a variety of methods, to obtain, from a single observation from a chi-square distribution with known v , a maximum likelihood estimate of the noncentrality parameter K , and confidence intervals for K as well. (See, e.g., Saxena and Alam, 1982; Spruill, 1986, Steiger, 1989.)

Since we can obtain a maximum likelihood estimate and confidence interval for nF^* , we can obtain a confidence interval and maximum likelihood estimate for F^* by dividing by n .

EzPATH obtains the maximum likelihood estimate and confidence interval for nF^* by iterative methods. (See Steiger, 1989 for technical details.) The "point estimate" of the "population noncentrality index" printed by EzPATH is the result of the maximum likelihood estimation procedure. If the test statistic were distributed exactly as a noncentral chi-square, then this procedure would yield a maximum likelihood estimate. The term "asymptotic MLE" is more appropriate here, since the noncentral chi-square distribution is only an asymptotic result. We will refer to such estimators as AMLE's throughout the remainder of the text.

We mentioned above a simple unbiased estimate. The AMLE procedure always yields a larger value than the unbiased estimate, so we might view the AMLE approach as, in practice, a slightly more conservative alternative.

The population noncentrality index is a quadratic form, and as such is a weighted sum of squares of the residuals. Suppose we were to place the non-redundant elements of Σ in a vector σ . Let the

Maximum Likelihood estimates of Σ for a particular structural model be placed in a vector σ^* , and the residuals in a vector e . Then

$$\sigma = \sigma^* + e.$$

Furthermore (see Browne, 1974 for details), F^* can be written in the form

$$F^* = e'W_e$$

where nW is the inverse variance-covariance matrix of the elements of S . F^* can thus be viewed as a weighted sum of squared residuals.

3. The Population RMS Index

The Population Noncentrality Index F^* (PNI) offers some significant virtues as a measure of badness of fit (see, e.g., Steiger & Lind, 1980; McDonald, 1989). First, it is a weighted sum of discrepancies. Second, unlike the Akaike information criterion, for example, it is relatively unaffected by sample size.

However, there are two obvious problems with using the population noncentrality index as an index of population badness of fit.

The PNI is not in the metric of the original parameters, because the quadratic form squares the residuals.

The PNI fails to compensate for model complexity. In general, for a given Σ , the more complex the model the better it fits. A method for assessing population fit which fails to compensate for this will inevitably lead to choosing the most complex models, even when much simpler models fit the data nearly as well. The PNI fails to compensate for the size or complexity of a model. Hence it has limited utility as a device for comparing models.

The RMS index, first proposed by Steiger and Lind (1980), takes a relatively simplistic (but not altogether unreasonable) approach to solving these problems. Since model complexity is reflected directly in the number of free parameters, and inversely in the number of degrees of freedom, we divide the PNI by degrees of freedom, and then take the square root to return the index to the same metric as the original parameters.

Hence

$$R^* = (F^*/v)^{1/2} \quad (38)$$

The RMS index R^* can be thought of as a root mean square standardized residual. Values below .10 indicate a good fit, values below .05 a very good fit. Point estimates below .01 indicate an outstanding fit, and are seldom obtained.

In practice, we obtain point and interval estimates of the population RMS index as follows. First, we obtain point and interval estimate of the PNI. [The point estimate we use is the asymptotic MLE. We could just as easily use the unbiased estimate. It is not yet clear which would be superior, but the differences between the two approaches are likely to be trivial.] Since all these are non-negative, and R^* is a monotonic transform of the PNI, we obtain point estimates and a confidence interval for R^* by inserting the corresponding values for F^* in Equation 38.

4. Population Gamma Index Γ_1

Tanaka and Huba (1985, 1988) have provided a general framework for conceptualizing certain fit indexes in covariance structure analysis. In their first paper, Tanaka and Huba (1985, equation 19) gave a general form for the *sample* fit index for covariance structure models under arbitrary generalized least squares estimation.

In the Tanaka–Huba treatment, we assume that a covariance structure model has been fit by minimizing an arbitrary GLS discrepancy function of the form

$$F(S, C_\theta) = 1/2 \text{Tr} [(S - C_\theta)V]^2 \quad (39)$$

or, equivalently (see Browne, 1974)

$$F(s, c^*) = [s - c^*]'W [s - c^*] \quad (40)$$

where s is the vector containing $p^* = p(p+1)/2$ non-duplicated elements of the sample covariance matrix S , and p is again the number of observed variables. c^* is the $p^* \times 1$ vector of non-duplicated elements of the reproduced covariance matrix C_θ .

V in Equation 39 and W in equation 40 are arbitrary matrices. Appropriate choice of V or W can yield GLS or ML estimators. For example, minimization of Equation 39 with $V = S^{-1}$ if S has a Wishart distribution yields the well-known GLS estimators (Browne, 1974). Setting $V = C_\theta^{-1}$ yields maximum (Wishart) likelihood estimators under the same conditions. (Browne gives formulas for obtaining W from V .)

The Tanaka–Huba fit index can be written as

$$\gamma = 1 - [e'We / s'Ws] \quad (41)$$

where e is the vector of residuals given by

$$e = s - c^*$$

In their more recent paper, Tanaka and Huba (1988) demonstrate a deceptively simple, but important result which holds for models which are invariant under a constant scaling function (ICSF). A covariance structure model is ICSF if multiplication of an y covariance matrix which fits the model by a positive

scalar yields another covariance matrix which also satisfies the model exactly (though possibly with different free parameter values).

If a model which is ICSF has been estimated by minimizing a discrepancy function of the form given in Equations 39 and 40, then

$$e'Wc^* = 0, \quad (42)$$

i.e., e and c^* are orthogonal "in the metric of W ," and, consequently,

$$s'Ws = c^*Wc^* + e'We. \quad (43)$$

If Equation 43 holds, then γ may be written

$$\gamma = c^*Wc^* / s'Ws = 1 - [e'We / s'Ws]. \quad (44)$$

In this form, we see that γ defines a weighted coefficient of determination.

Under Maximum Wishart Likelihood estimation, $V = C_\theta^{-1}$. One immediately obtains

$$F^* = e'We = 1/2 \text{Tr} [(S - C_\theta)C_\theta^{-1}]^2 \quad (45)$$

$$s'Ws = 1/2 \text{Tr} [SC_\theta^{-1}]^2, \quad (46)$$

whence

$$\gamma_{ML} = 1 - \{\text{Tr} [SC_\theta^{-1}]^2 / \text{Tr} [SC_\theta^{-1}]^2\} \quad (47)$$

which is equivalent to the Jöreskog and Sörbom (1984) GFI index.

Moreover, if the model is ICSF, then, under maximum Wishart likelihood estimation, we have the simplifying result (Browne, 1974, Proposition 8)

$$\text{Tr}(\text{SC}_\theta^{-1}) = p. \quad (48)$$

Substituting in Equation (45), we find

$$F^* = e'We = 1/2 [\text{Tr}[\text{SC}_\theta^{-1}]^2 - p], \quad (49)$$

and so

$$\gamma_{\text{ML}} = p / 2 s'Ws = p / \text{Tr}[\text{SC}_\theta^{-1}]^2. \quad (50)$$

Tanaka and Huba (1985, 1988) based their derivation of γ on sample quantities. However, in principle we are interested in a sample index only as a vehicle for estimating the corresponding *population* index. The corresponding population quantities are obtained by substituting Σ for S , and Σ_θ for C_θ in Equations 49 and 50.

We have

$$\Gamma_1 = p / \text{Tr}[\Sigma\Sigma_\theta^{-1}]^2 \quad (51)$$

Γ_1 can be thought of as a *weighted population coefficient of determination for the multivariate model*. (It may also be thought of as the population equivalent of the Jöreskog–Sörbom GFI index.)

An accurate point estimate for Γ_1 will provide useful information about the extent to which a model reproduces the information in Σ . A confidence interval, however, provides even more useful information, because it conveys not only the size of Γ_1 , but also the precision of our estimate.

Let F^* as defined in Equation 36 be the Population Noncentrality Index. From Equations 49 and 50, it is easy to see that

$$\Gamma_1 = p / [2F^* + p] \quad (51)$$

Equation 51 demonstrates that, under maximum likelihood estimation with ICSF models, Γ_1 can be expressed solely as a function of the Population Noncentrality Index and p , the number of manifest variables. Any consistent estimate of F^* will yield a consistent estimate of Γ_1 when substituted in Equation 51.

Equation 51 implies that an asymptotic maximum likelihood estimate (AMLE) for the population noncentrality index F^* may be converted readily to an AMLE for Γ_1 , simply by substituting the AMLE for F^* in Equation 51. Similarly, substitution of the endpoints of the confidence interval for F^* in Equation 51 will generate a confidence interval for Γ_1 .

It is perhaps worth adding a conjecture that the sample index γ_{ML} of Equation 49 can be a rather biased estimate of the corresponding population index Γ_1 at moderate sample sizes. To see why, suppose that, for a moderately large n , the asymptotic distributional result holds exactly. Then, recalling the results in Equations 36 and 37 and the accompanying discussion, we may show that

$$E \{ \text{Tr}(\text{SC}_\theta^{-1})^2 \} = \text{Tr} \{ \Sigma\Sigma_\theta^{-1} \}^2 + v/n$$

Since, for moderate n and large v , the denominator of Equation 49 is positively biased, one *suspects* (Monte Carlo evidence would be necessary to confirm this) that this estimate (i.e., the GFI calculated by LISREL VI), is a biased estimate of the population quantity (Γ_1) which we are really interested in, and that, in general, the GFI index will tend to underestimate Γ_1 .

5. Adjusted Population Gamma Index Γ_2

Γ_1 , like F^* , fails to compensate for the effect of model complexity. Consider a sequence of *nested* models, where the models with more degrees of freedom are special cases of those with fewer degrees of freedom.

(See Steiger, Shapiro, and Browne, 1985, for a discussion of the statistical properties of chi-square tests with nested models.) For a nested sequence of models, the more complex models (i.e., those with more free parameters and fewer degrees of freedom) will always have Γ_1 coefficients as low or lower than those which are less complex.

Goodness of fit, as measured by Γ_1 , improves more or less inevitably as more parameters are added. The adjusted population gamma index Γ_2 attempts to compensate for this tendency.

Just as Γ_1 is computed by subtracting a ratio of sums of squares from 1, Γ_2 is obtained by subtracting a corresponding ratio of mean squares from 1. Let σ^* be a $p^* [= p(p+1)/2] \times 1$ vector of non-duplicated elements of the population reproduced covariance matrix Σ_θ for a model with v degrees of freedom, and σ a vector of corresponding elements of Σ . Let $e = \sigma - \sigma^*$. Then Γ_2 is

$$\begin{aligned}\Gamma_2 &= 1 - [e'We/v]/[\sigma'W\sigma/p^*] \\ &= 1 - [p^*/v][1 - \Gamma_1].\end{aligned}\quad (52)$$

Consistent estimates and confidence intervals for Γ_1 may thus be converted into corresponding quantities for Γ_2 by applying Equation 52.

6. Jöreskog-Sörbom (1984) GFI.

This sample based index of fit is computed as in Equation 47.

7. Jöreskog-Sörbom (1984) Adjusted GFI.

See Equation 52 above.

8. Rescaled Akaike (1973) Information Criterion.

In a number of situations the user must decide among a number of competing *nested* models of differing dimensionality. (The most typical example is the choice of the number of factors in common factor analysis.) Akaike (1973, 1983) proposed a criterion for selecting the dimension of a model. Steiger and Lind (1980) presented an extensive Monte Carlo study of the performance of the Akaike criterion. Here we rescale the criterion (without affecting the decisions it indicates) so that it remained more stable across differing sample sizes. The rescaled Akaike criterion is as follows.

Let $F_{ML,k}$ be the maximum likelihood discrepancy function and q_k be the number of free parameters for the model M_k . Let N be the sample size

Select the model M_k for which

$$A_k = F_{ML,k} + 2q_k/(N-1)$$

is a minimum.

9. Schwarz's (1978) Bayesian Criterion

This is similar in use to Akaike's index, except one selects the model M_k for which

$$S_k = F_{ML,k} + \ln(N) q_k / (N-1)$$

is a minimum.

10. Browne-Cudeck (1989) Single Sample Cross-Validation Index.

Browne and Cudeck (1989) recently proposed a single sample cross-validation index as a clever follow-up to their earlier (Cudeck & Browne, 1983) paper on cross-validation. Cudeck and Browne had proposed a cross-validation index which, for model k in a set of competing models M_k is of the form $F_{ML}(S_v, C_k^*)$. In this

case, F is the maximum likelihood discrepancy function, S_v is the covariance matrix calculated on a cross-validation sample, and C_k^* the reproduced covariance matrix obtained by fitting model M_k to the original calibration sample. In general, better models will have smaller cross-validation indices.

The drawback of the original procedure is that it requires two samples, i.e., the calibration sample for fitting the models, and the cross-validation sample. The new measure estimates the original cross-validation index from a single sample.

The measure is

$$c_k = F_{ML,k} + 2q_k/(N-p-2)$$

where q_k is the number of free parameters in model k , p is the number of manifest variables, and N is the sample size.

12. Historical Background.

My ideas on fitting of covariance structure models have a documentable history dating back to 1980. I have long delayed publication of some of this material, because I felt some key theoretical aspects were ambiguous or missing. However, I feel that it is perhaps worth mentioning that I first presented (in print and in a public forum) a technique of noncentrality parameter interval estimation as a means of assessing badness of fit in covariance structural models almost a decade ago.

In 1980, I presented a paper (Steiger and Lind, 1980) at the Psychometric Society Meetings which proposed several ideas concerning goodness of fit evaluation in covariance structure analysis.

The major ideas were

1. That the power of the likelihood ratio test could be closely approximated by a noncentral chi-square distribution,

2. That the noncentrality parameter of that distribution might serve, with some modification, as an excellent measure of "badness of fit,"

3. A *confidence interval* approach to the assessment of the population badness of fit offered significant advantages over a point estimation approach,

4. A proper measure of badness of fit must attempt to compensate for model complexity.

At that time these ideas were novel.

The paper also presented an extensive Monte Carlo analysis of the performance of the likelihood ratio test, Akaike's information criterion, and Schwarz's Bayesian criterion for selecting the number of common factors in factor analysis. The paper was presented May 30, 1980, and a written handout (available from me by request) was distributed to those in attendance at Iowa City. Two consecutive afternoon symposia (one on Covariance Structures, one on Factor Analysis) were held in the room where I gave my talk. The identity of some of those present may be of some historical interest, and can be ascertained by reading pages 504-505 of the 1980 Psychometrika, where a listing of symposia chairmen and paper presenters at the two sessions is given.

8. EzPATH Technical Output

When any model is analyzed, EzPATH prints out technical output concerning the estimation process. When the DUAL option is invoked, EzPATH prints only the iteration history and summary for the OLS estimation, then proceeds to perform Maximum Likelihood estimation, for which it gives complete output.

A. Memory Usage Table.

EzPATH uses a dynamic memory allocation scheme which requires that a model be parsed before memory to analyze it is allocated. The program uses two main storage regions, one for integers and one for floating point variables. Prior to estimating any model, the program prints the amount of each storage region required for that particular model. If memory requirements are exceeded, the model is not analyzed.

B. Iteration History.

During iteration a report of the values of the discrepancy function, and possibly gradient and coefficient values as well is printed. This can be useful in evaluating the progress of iteration.

C. Results of Iteration

The final Discrepancy Function Value, Largest Absolute Gradient, and number of iterations is printed. The largest absolute gradient is useful diagnostic information. It should be close to zero (i.e., on the order of 10^{-3} or less) if a true minimum has been found.

D. Chi-Square Test Statistic

The sample size, degrees of freedom, and chi-square value are printed. The Probability Level of the chi-square is the probability of obtaining a value greater than or equal to the obtained value.

It is important to keep in mind that, strictly speaking, the chi-square statistic may not be valid if (1) the manifest variables do not have a multivariate normal distribution; (2) the sample correlation matrix is used instead of the sample covariance matrix, or (3) the sample size is small.

E. Fit Indices

As part of its technical output, EzPATH prints the Asymptotic MLE's and confidence intervals for F^* , R^* , Γ_1 , and Γ_2 . They are referred to in the output as "Population Noncentrality Index," "Steiger-Lind RMS Index," "Population Gamma Index," and "Adjusted Population Gamma" respectively. It also prints a number of other indices. The theory behind these indices is given in detail in section VIII E.

The relatively nontechnical user will probably find the confidence intervals on R^* and Γ_2 to be the most useful. R^* can be thought of as a root mean square standardized residual, *adjusted for model complexity*. Values of this index below .10 indicate a reasonably good fit, while values below .05 generally indicate an excellent fit.

Γ_2 can be thought of as a coefficient of model determination, adjusted for model complexity. Values above .90 indicate good fit, values above .95 an excellent fit.

Excessively wide confidence intervals for the various fit indices indicate inadequate statistical power in the particular testing situation. In such a situation, a high probability level for the chi-square statistic might not necessarily mean that the model fits well. (Remember, the chi-square test of fit is an accept-support test. Low power tends to produce results supporting a model!)

Very narrow confidence intervals indicate high precision of estimate. In general, if the confidence interval for R^* or Γ_2 is very narrow and the indices are not near the endpoints of their range, the chi-square probability is no longer very relevant, because statistical power is high.

For example, suppose the 90% confidence interval for Γ_2 ranged from .91 to .92. This would mean that, with very high precision, we had determined that model fit was quite good, though not outstanding, and almost certainly not perfect. In this case, the traditional chi-square test of fit would almost certainly reject with an extremely low p-level!

Asymptotic MLE's and confidence intervals for fit indices are obtained by an iterative procedure. If the procedure fails to converge, an error message will be generated.

F. Gamma Coefficient Confidence Interval Trace Criterion

As discussed in the chapter on theory, the point estimates and confidence intervals for the Γ_1 and Γ_2 coefficients, to be strictly valid, require that proper maximum likelihood estimates be obtained, *and that the model be invariant under a constant scaling factor (ICSF)*. If these assumptions are satisfied, then the following equation holds:

$$\text{Trace Criterion} = \text{Tr}(\text{SC}_\theta^{-1}) - p = 0$$

If the trace criterion is not acceptably close to zero, it indicates that at least one of the necessary conditions for strict validity of the gamma coefficient confidence interval and point estimates has been violated.

G. Standard Errors. (Optional)

When maximum likelihood estimates are obtained, the user has the option of requesting estimated standard errors for each of the free parameters estimated by the program. These standard errors are printed immediately after the value of the free parameter, with the prefix "SE=".

These standard errors can be used to assess whether a parameter is significantly different from zero. A simple, and rather crude approach, is to calculate an asymptotic z score (some writers call this a "T-score") value for a parameter by dividing by the square root of its estimated standard error. In evaluating the significance of a coefficient, one must remember the Bonferonni inequality. In a model with many parameters, individual coefficients should be tested at a fairly stringent probability level. The crude criterion adopted by many practitioners is to declare a coefficient "significant" if its absolute value is more than two standard errors. This is probably not conservative enough, especially when larger models are tested.

Standard errors in EzPATH are computed using the formulas in Browne (1982), pages 105–107. The standard errors are obtained by inverting the estimated Hessian. If a model is not identified, this matrix will be singular (or possibly quite ill-conditioned, due to rounding error). In that case, standard errors are not printed. If the Hessian matrix is not positive definite or round-off error occurs, a message to that effect will be given, and output will be printed without the standard errors.

H. Optional Output (Produced by the PRINT=LONG command)

1. Input Matrix (S)

The input covariance or correlation matrix will be formatted and printed.

2. Reproduced Matrix (C_θ)

The "best" attempt to reproduce the input matrix using the model will be printed. This is a useful feature if you wish to create a population matrix corresponding to a particular model. (See page xxx).

3. Standardized Residuals

Residuals are useful sometimes for locating the "problem areas" in fitting a model. Large residuals for a particular variable or pair of variables will often indicate areas of a model which need further attention.

Raw residuals depend, of course, on the scaling of the covariance matrix. Consequently, EzPATH prints standardized residuals, i.e., residuals divided by the standard deviations of the variables. Each entry in the standardized residual matrix is $(s_{ij} - c\theta_{ij})/(s_i s_j)$.

EzPATH also computes the root mean square standardized residual, and the mean absolute standardized residual as indices of fit.

4. Normalized Residuals

Standardized residuals compensate for scale differences, but fail to take into account that different elements of a covariance matrix have different sampling variabilities. The normalized residuals are computed by dividing each element of the matrix of residuals by the square root of its estimated asymptotic variance. We use the same formula as LISREL VI for computing these indices. (Note: the formula is given incorrectly in a number of sources, including the LISREL VI manual. The formula used for the estimated variance is $(c_{\theta ii} + c_{\theta ij}^2) / (n-1)$).

Normalized residuals can be SAVEd to a file for plotting or further analysis. A normal probability plot of the normalized residuals can sometimes provide evidence of violations of statistical assumptions underlying maximum likelihood estimation.

I. Results for the Fitted Model

The final step in EzPATH output is to print the results of the estimation process in the PATH1 language. Immediately before printing the results, EzPATH prints the word MODEL on a separate line. This saves you the trouble of adding the command if you decide to modify the model and recycle the output.

Model coefficient values resulting from the estimation process are printed in braces immediately after the coefficient number. If standard errors have been requested, they are also printed within the braces following the identifier "SE =".

If you recycle EzPATH output during a model modification, you do not need to erase the standard error information

from within braces. The program ignores this information when reading a set of PATH1 statements as input.

9. Illustrative Problems and Examples

A. A Confirmatory Factor Analysis Model

Everitt (1984 p. 45–52) discusses, in considerable detail, a confirmatory factor analysis of a data set in Child (1970). The EzPATH diagram for this model is given in Figure 9.

Data for this example are in the file CHILD.SYS on the distribution diskette. Below is the PATH1 specification for this model along with some additional commands. These statements are contained in the file CHILD.CMD on the distribution diskette. To produce the output for the CHILD example, simply type

SUBMIT CHILD

Output for the analysis will be placed in a file called CHILD.DAT.

This PATH1 command structure, shown on the next page, is particularly instructive, because it demonstrates how appropriate use of spacing and comments can add greatly to the readability of a PATH1 file.

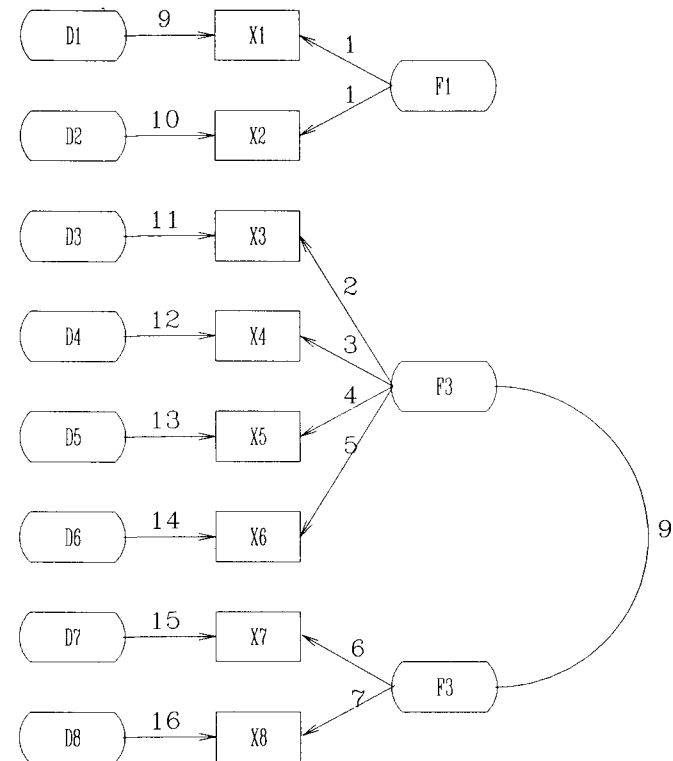


Figure 9. Path Diagram for a Confirmatory Factor Model from Everitt (1984).

```

PAGE SCREEN=SCROLL
USE CHILD
METHOD=ML
NUMBER=93
OUTPUT CHILD
MODEL
* Child Example from Everitt, p. 45-50
*****
* This is a straightforward confirmatory common factor analysis *
* with 3 common factors. The common factors are called F1, F2, F3. *
* N is not given in Everitt's book, but, by reverse computation, *
* we have decided it must be 93. *
*****
* Here are the loadings of the common factors on the observed variables *
* For identification, F1 is constrained to load equally on X1 and X2, *
* a fact which Everitt discusses in the book. *
*****
(F1)-1->[X1]
      -1->[X2]

(F2)-2->[X3]
      -3->[X4]
      -4->[X5]
      -5->[X6]

(F3)-6->[X7]
      -7->[X8]

*****
* Here is the correlation between factors 2 and 3 *
*****
(F2)-8-(F3)

*****
* And finally, the unique factors and their loadings. *
*****
(D1)-9->[X1]
(D2)-10->[X2]
(D3)-11->[X3]
(D4)-12->[X4]
(D5)-13->[X5]
(D6)-14->[X6]
(D7)-15->[X7]
(D8)-16->[X8]

ESTIMATE

```

B. Path Models for Home Environment and Mathematics Achievement

Jöreskog and Sörbom (1982) discuss several structural models which they fit to data from a study of home environment and school achievement by Keeves (1972).

Keeves studied 215 sixth grade boys over a one year period, and measured initial mathematics achievement (Y_1), a structural dimension in the home (X_1), an attitudinal dimension of the home environment (X_2), a process dimension of the home environment (X_3), and final mathematics achievement (Y_2). The correlation matrix for the Keeves data is in a file called KEEVES.SYS on the distribution diskette.

According to Jöreskog and Sörbom,

The three home environment dimensions were derived as the principal components of three sets of items which sampled each of the three home domains. Thus, the structural dimension was based on home interview items which determined the level of father's education, father's occupation, mother's occupation before marriage, religious affiliation, and number of children in the family. The attitudinal dimension was based on parents' attitudes toward the child's present education, future education, and occupation, and the parents' aspirations for themselves. The process dimension was based on relations between home and school, use of books and libraries, parents' help with formal school work, and arrangements for doing home assignments.

Cooley and Lohnes (1976) analyzed Keeves' data with a path model on the observed variables. This model, essentially a standardized multiple regression, is shown in Figure 10.

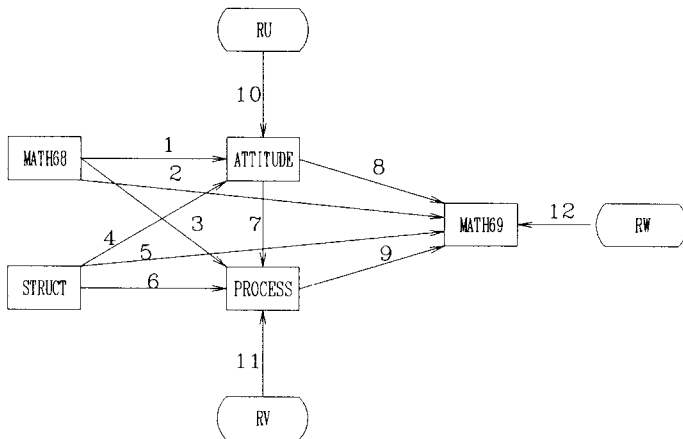


Figure 10. Path Diagram for Cooley and Lohnes Model

PATH1 statements corresponding to this model can be found in a file KEEVES1.CMD.

Jöreskog and Sörbom fit two structural models of their own to the Keeves data. The EzPATH diagram for the first (Figure 3A in their paper) is in Figure 11.

This model, according to Jöreskog and Sörbom,

takes measurement errors in the home variables into account and treats them merely as fallible indicators of an aggregate construct variable "home".

PATH1 statements for this model are in a file called KEEVES3A.CMD. on the distribution diskette.

An alternative model is represented in the EzPATH diagram in Figure 12. This model incorporates errors of measurement in the measures of mathematics achievement. It assumes that reliabilities for the two measures are known to be .90. Consequently the path coefficients representing true score and error variance coefficients are shown as fixed in the diagram. PATH1 statements for this model are in a file called KEEVES3B.CMD on the distribution diskette.

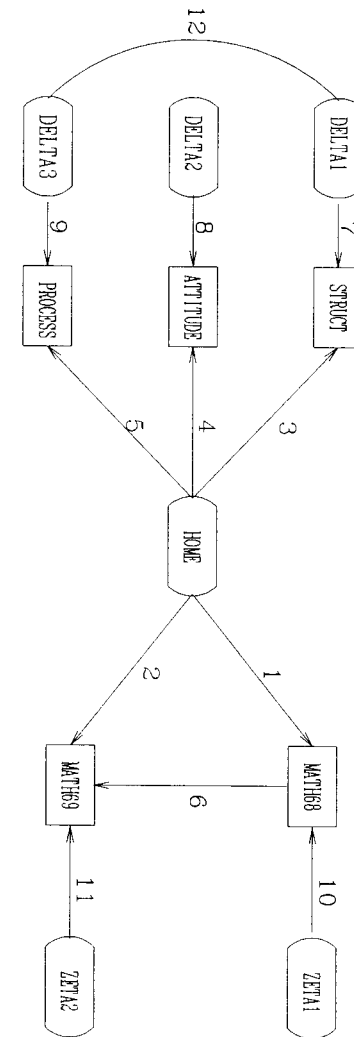


Figure 11. EzPATH Diagram Corresponding to Figure 3A, Jöreskog and Sörbom (1982)

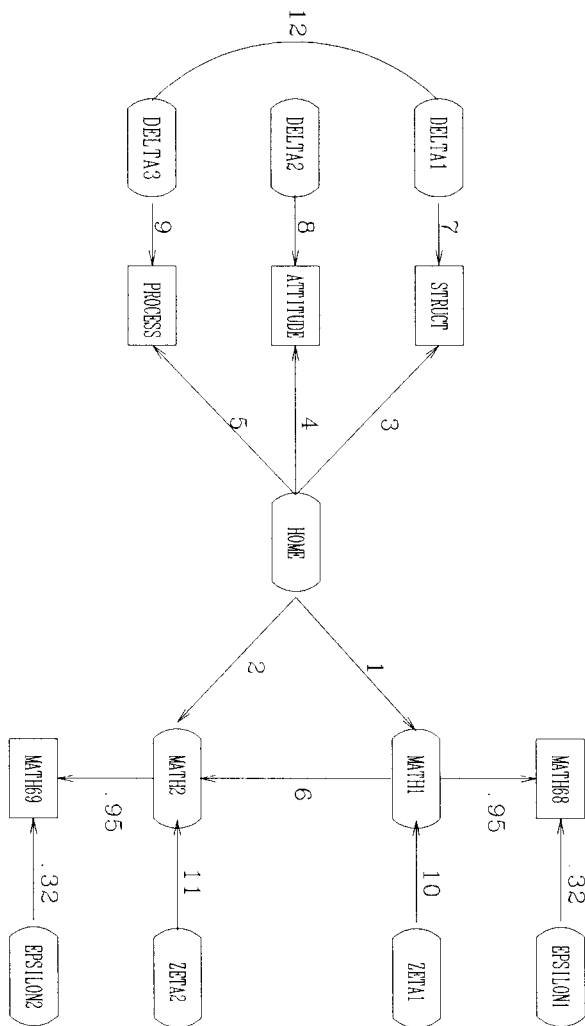


Figure 12. EzPATH Diagram Corresponding to Figure 3B, Jöreskog and Sörbom (1982)

C. Principal Components Analysis

The models we have discussed so far are, for the most part, extensions of the common factor model. Indeed, in the LISREL model formulation, the typical model is conceived as a set of structural relations between two distinct sets of common factors.

The common factor model has a number of conceptual problems connected with it. One of the most interesting, and oft-debated problems is that of factor indeterminacy. Factor indeterminacy refers to the fact that more than one different set of latent variables will fit exactly the same common and unique factor loadings for a given set of observed variables. Suppose, then, you believed you had "identified" a common factor on the basis of a factor analysis. It might turn out that two different versions of this factor (in the population), so different that they correlate zero, might fit all your model parameters equally well.

This raises some interesting logical problems. What sense does it make to say you have found "a" common factor, when different versions of the same factor exist, and these versions produce scores that correlate no more than two columns of random numbers?

Factor indeterminacy occurs, basically, because in common factor analysis we seek to determine more latent variables than we have observed variables. This cannot, in general be done.

Factor indeterminacy is not a sampling problem. It occurs in the population as well as the sample.

There are many conflicting views on how one should conceptualize factor indeterminacy. The issue is almost as old as factor analysis itself and was discussed heatedly in the 1920's and 1930's, although the advent of digital computers, and the ensuing wave of enthusiasm for factor analysis, swept it temporarily out of the collective consciousness of the psychometric community. Literature

reviews by Steiger and Schönemann (1978) and Steiger (1979) contain numerous references for the interested reader. McDonald and Mulaik (1979) provide an alternative account.

Partly because of the indeterminacy problem, and partly for reasons of computational efficiency, many data analysts prefer component analysis over factor analysis. Component models express their "latent" variables (i.e., components) as explicit linear functions of the observed variables, and consequently are not plagued by factor indeterminacy problems.

Schönemann and Steiger (1976) presented a detailed theoretical analysis of the relationship between component and factor analysis. They described a data analytic system called regression component analysis which shares many of the properties of common factor analysis, but is fully determinate. Schönemann and Steiger demonstrated how all factor models have an equivalent, falsifiable, regression component model. Hence the common factor model could be re-expressed in terms of components, a point of view echoed from a different perspective by Bartholomew (1984). They concluded that, in the vast majority of cases, little if anything would be lost by substituting a model based on components for one based on common factors.

Extensive evidence supports this point of view. Velicer & Jackson (1989) have recently provided a succinct summary of this evidence. They demonstrate that most arguments for the superiority of factor analysis over component analysis are based on obvious fallacies or questionable premises.

Although the FACTOR module provides a superior facility for performing principal components analysis, it is possible to extract principal components using EzPATH.

The key is to recall that common factors with no unique variance are components. The first sample principal component is that linear combination of the observed variables which has maximum variance. However, the first

standardized principal component can also be defined as that unit variance variable whose covariances with the observed variables, when placed in a vector f , have the property that $\text{Tr}(S - ff')^2$ is a minimum.

Consequently, if we write a single common factor model, but leave out the unique variances, and estimate using the METHOD = LS option, we will obtain loadings for the first principal component.

To see this, first enter the CORR module, USE the CRIME file, compute a PEARSON correlation matrix, and SAVE the result to a file called CRIMCORR.SYS.

Then enter EzPATH, and type the following commands.

```
USE CRIMCORR
NUMBER=50
METHOD=LS
OUTPUT=PC
MODEL
(F1) -1-> [MURDER]
      -2-> [RAPE]
      -3-> [ROBBERY]
      -4-> [ASSAULT]
      -5-> [BURGLARY]
      -6-> [LARCENY]
      -7-> [AUTOTHFT]
      -8-> [REGION]
ESTIMATE
```

You should obtain the following output.

```
MODEL
(F1) -1 { 0.578 }-> [MURDER]
      -2 { 0.869 }-> [RAPE]
      -3 { 0.739 }-> [ROBBERY]
      -4 { 0.790 }-> [ASSAULT]
      -5 { 0.875 }-> [BURGLARY]
      -6 { 0.749 }-> [LARCENY]
      -7 { 0.673 }-> [AUTOTHFT]
```

-8{ 0.409}->[REGION]

You may quickly verify for yourself that the resulting model coefficients are indeed the same values produced by the FACTOR module.

Once loadings for the first principal component have been determined, you may go on to extract a second. Simply edit the output from the first run by removing the coefficient numbers. This fixes the loadings for the first component as *fixed values* equal to the loadings you obtained. Then add to your model loadings for a second component. Your input will look like this

MODEL

**(F1)-{ 0.578}->[MURDER]
- { 0.869}->[RAPE]
- { 0.739}->[ROBBERY]
- { 0.790}->[ASSAULT]
- { 0.875}->[BURGLARY]
- { 0.749}->[LARCENY]
- { 0.673}->[AUTOTHFT]
- { 0.409}->[REGION]**

**(F2)-9->[MURDER]
-10->[RAPE]
-11->[ROBBERY]
-12->[ASSAULT]
-13->[BURGLARY]
-14->[LARCENY]
-15->[AUTOTHFT]
-16->[REGION]**

At this point you might be asking, "Why can't I extract the two components simultaneously?" Well, in a sense you can, except that the two components you obtain will, in general, be an arbitrary rotation of the first two principal components, rather than the first and second principal components themselves.

By extracting the components sequentially, we guarantee that each, in turn, accounts for maximum variance.

D. Testing Pattern Hypotheses on Correlations

EzPATH can be used to test pattern hypotheses about the structure of correlation matrices.

A pattern hypothesis on the elements of a correlation matrix is any hypothesis that the correlations are equal to each other, or to specified numerical values.

A number of interesting tests about correlational structure can be phrased as pattern hypotheses on the population correlation matrix. Steiger (1980) reviewed the literature on these tests.

Steiger (1979) produced a computer program called MULTICORR which tests pattern hypotheses on correlations using the sample correlation matrix as input. Pattern hypotheses tests on the population correlations can also be performed using the sample covariance matrix as input, using the following device. For each manifest variable, an "alias" latent variable is created which loads only on its manifest variable, with a free parameter coefficient. The variances of the latent variables are left unspecified, which means they will be fixed at one.

Then desired constraints on the correlation matrix are expressed as covariance constraints on the alias latent variables.

This technique is formally equivalent to testing a hypothesis of the form

$$\Sigma = \text{DPD}$$

where D is a diagonal matrix of scaling constants, and P is a correlation matrix. (See, e.g., Jöreskog, 1978, p. 475.)

This particular kind of covariance structure model has an advantage, in that it is *scale free*. Any changes in scale in the observed variables will not affect estimates of the elements of P (although such changes will affect the estimates of the elements of D). Consequently, the sample correlation matrix may be analyzed instead of the sample covariance matrix, and the chi-square statistic will still be correct. Point estimates for the elements of P will also be correct. Generally, the elements of D are not of interest in such cases.

The following examples should make the general approach clear.

1. Testing for Circumplex Structure

A perfect circumplex correlation matrix (Guttman, 1954) has equal correlations on sub-diagonal strips. For example, a 6x6 correlation matrix would be of the form

1						
ρ_1	1					
ρ_2	ρ_1	1				
ρ_3	ρ_2	ρ_1	1			
ρ_2	ρ_3	ρ_2	ρ_1	1		
ρ_1	ρ_2	ρ_3	ρ_2	ρ_1	1	

A data set used frequently to demonstrate a Guttman circumplex is a correlation matrix among 6 different kinds of abilities for 710 Chicago school children. This matrix, from Guttman (1954), is in a file called GUTTMAN.SYS on the distribution diskette.

Below is the set of commands for testing a hypothesis of circumplex structure. The commands are in a file called CIRCLE.CMD on the distribution diskette.

The first 6 model statements create 6 unit variance "alias" latent variables corresponding to the 6 manifest variables. These are, in effect, the manifest variables rescaled to have unit variance. Hence hypotheses

about the correlations among the manifest variables can be tested as hypotheses about the covariances of these alias latent variables. The remaining model statements establish the constraints described above.

```
use guttman
nu=710
model
```

```
(F1)-1->[X1]
(F2)-2->[X2]
(F3)-3->[X3]
(F4)-4->[X4]
(F5)-5->[X5]
(F6)-6->[X6]
```

```
(F1)-7-(F2)
      -8-(F3)
      -9-(F4)
      -8-(F5)
      -7-(F6)
```

```
(F2)-7-(F3)
      -8-(F4)
      -9-(F5)
      -8-(F6)
```

```
(F3)-7-(F4)
      -8-(F5)
      -9-(F6)
```

```
(F4)-7-(F5)
      -8-(F6)
```

```
(F5)-7-(F6)
```

ESTIMATE

By SUBMITting the file CIRCLE to EzPATH, you can generate the following output from the maximum likelihood estimation.

Sample Size (N)	=	710
Degrees of Freedom	=	12
Chi-Square	=	27.0515
Probability Level	=	.0076

Fit indices:

Point Estimates:

Population Noncentrality Parameter	=	.0225
Steiger-Lind Adjusted RMS Index	=	.0433
Population Gamma Index	=	.9926
Adjusted Population Gamma Index	=	.9870

Confidence Intervals:

Population Noncentrality Index	=	.0053,	.0485
Steiger-Lind Adjusted RMS Index	=	.0210,	.0636
Population Gamma Index	=	.9841,	.9982
Adjusted Population Gamma Index	=	.9722,	.9969

Calculations Completed.

Here are the results for the fitted model.

MODEL

```
(F1)-1{ 0.988}->[X1]
(F2)-2{ 0.991}->[X2]
(F3)-3{ 0.997}->[X3]
(F4)-4{ 1.009}->[X4]
(F5)-5{ 1.016}->[X5]
(F6)-6{ 1.000}->[X6]
```

```
(F1)-7{ 0.396}-(F2)
      -8{ 0.295}-(F3)
      -9{ 0.250}-(F4)
      -8{ 0.295}-(F5)
      -7{ 0.396}-(F6)
```

```
(F2)-7{ 0.396}-(F3)
      -8{ 0.295}-(F4)
      -9{ 0.250}-(F5)
      -8{ 0.295}-(F6)
```

```
(F3)-7{ 0.396}-(F4)
      -8{ 0.295}-(F5)
      -9{ 0.250}-(F6)
```

```
(F4)-7{ 0.396}-(F5)
      -8{ 0.295}-(F6)
```

```
(F5)-7{ 0.396}-(F6)
```

In your output in this case the covariances among the alias latent variables will be the maximum likelihood estimates of the population correlations under the null hypothesis.

The sample size is very large in this example. Hence, we would expect precision of estimate to be very high. At the same time, we would have to keep in mind that the "reject-support" approach of the chi-square test would be of very limited usefulness in this situation. We recognize that a model with this many constraints will almost certainly not fit perfectly in the population, and we have very high power to detect imperfect fit.

The chi-square statistic yields, in this case, a value of 27.05 with 12 degrees of freedom. The probability level is .0076, indicating that the null hypothesis of perfect fit must be rejected. Jöreskog (1978), analyzing these data, remarked that they "do not fit a circumplex well."

This conclusion seems, in the 20-20 vision of hindsight, unjustified, and incorrect. The correct conclusion is that it is highly probable that they do not fit a circumplex *perfectly*. Goodness of fit indices indicate how well these data actually do fit a circumplex. The 90% confidence interval for the Steiger-Lind (1980) RMS index is between .021 and .063. (These numbers are in fact contained in the Steiger-Lind 1980 handout.)

The corresponding confidence interval for the *adjusted* population gamma coefficient is between .97 and .99.

A reasonable conclusion would seem to be that Guttman's data fit a circumplex *very* well, a fact which we have verified statistically.

It is possible to generate the PATH1 commands for a correlational pattern hypothesis quickly and efficiently using the **RMODEL** command. This command creates a file having the appropriate alias latent variables, and

statements corresponding to all possible correlations among them.

Try the following example. Type

```
USE GUTTMAN
RMODEL DEMOCORR
FEDIT "DEMOCORR.CMD"
```

You will see the following:

*** Note: Highest parameter number created by EzPATH is 21**

```
(F1)-1->[X1]
(F2)-2->[X2]
(F3)-3->[X3]
(F4)-4->[X4]
(F5)-5->[X5]
(F6)-6->[X6]

(F2)-7{ 0.446}-(F1)
(F3)-8{ 0.321}-(F1)
      -9{ 0.388}-(F2)
(F4)-10{ 0.213}-(F1)
      -11{ 0.313}-(F2)
      -12{ 0.396}-(F3)
(F5)-13{ 0.234}-(F1)
      -14{ 0.208}-(F2)
      -15{ 0.325}-(F3)
      -16{ 0.352}-(F4)
(F6)-17{ 0.442}-(F1)
      -18{ 0.330}-(F2)
      -19{ 0.328}-(F3)
      -20{ 0.247}-(F4)
      -21{ 0.347}-(F5)
```

It takes only a few seconds to modify this file to reflect circumplex structure, save it, highlight it, and perform the statistical test described above.

2. Testing for Stability of a Correlation Matrix over Time

Suppose you measured a set of variables twice, and wished to test the hypothesis that the correlation coefficient had not changed from time 1 to time 2. For example, suppose 120 individuals are measured twice on verbal, quantitative, and analytical ability. In this case, the covariance matrix would be 6x6.

This hypothesis can be tested easily as a correlational pattern hypothesis (see Steiger, 1980). The key to setting up this problem is to conceptualize the 3 variables measured at 2 times as 6 variables measured on a single group of subjects.

Suppose that the correlation matrix (contained in a file called TWOCORR.SYS on the distribution diskette) for the 6 variables is as follows:

```
VERBAL_1 1.00
QUANT_1 0.65 1.00
ANALY_1 0.54 0.68 1.00
VERBAL_2 0.27 0.30 0.21 1.00
QUANT_2 0.32 0.21 0.27 0.59 1.00
ANALY_2 0.18 0.26 0.22 0.48 0.55 1.00
```

Suppose further that the sample standard deviations for the variables are:

```
VERBAL_1 2.1
QUANT_1 3.0
ANALY_1 2.4
VERBAL_2 1.6
QUANT_2 3.3
ANALY_2 2.2
```

Then the sample covariance matrix for the data (contained in a file called TWOCOV.SYS on the distribution diskette) is:

```

4.4100
9.0000
5.7600
0.8064 2.5600
2.1384 3.1152 10.8900
1.1616 1.6896 3.9930 4.8400

```

The hypothesis for stability of correlations is tested with the following model statements:

MODEL

*** FIRST CREATE ALIAS LATENT VARIABLES**

```

(F1)-1->[VERBAL_1]
(F2)-2->[QUANT_1]
(F3)-3->[ANALY_1]
(F4)-4->[VERBAL_2]
(F5)-5->[QUANT_2]
(F6)-6->[ANALY_2]

```

*** THEN SET UP CORRELATIONS FOR TIME 1**

```

(F1)-7-(F2)
(F1)-8-(F3)
(F2)-9-(F3)

```

*** MAKE THE CORRELATIONS FOR TIME 2 EQUAL**

```

(F4)-7-(F5)
(F4)-8-(F6)
(F5)-9-(F6)

```

*** DON'T FORGET TO INCLUDE THE CROSS-CORRELATIONS!**

```

(F1)-10-(F4)
-11-(F5)
-12-(F6)
(F2)-13-(F4)
-14-(F5)

```

```

-15-(F6)
(F3)-16-(F4)
-17-(F5)
-18-(F6)

```

*** END OF MODEL**

NUMBER=120

SE=YES

PRINT=LONG

ESTIMATE

The estimated correlations among the alias latent variables are maximum likelihood estimates of the correlations among the manifest variables, under the null hypothesis.

As I mentioned above, it will make no difference in the hypothesis test, or if you test the hypothesis on the sample correlation or covariance matrix. The chi-square statistic will be identical. However, estimates for the scaling factors (coefficients 1 through 6 in the above model) will change.

You can verify these facts for yourself by running TWOCORR.CMD on both the covariance (TWOCOV.SYS) and correlation (TWOCORR.SYS) matrices.

I should add a note of caution here. The tests performed by EzPATH on correlation matrices are valid asymptotic statistics. However, the evidence suggests that (especially at smaller sample sizes) tests especially designed for correlational hypotheses (Steiger 1980a, 1980b; Steiger & Browne, 1984; Steiger & Hakstian, 1982; Wilson & Martin, 1983), which use the Fisher transform, will generally be more accurate and more powerful than tests performed with EzPATH.

E. Pattern Hypothesis Tests on a Covariance Matrix

Besides allowing tests on pattern hypotheses on correlations, EzPATH also allows similar tests on the covariance matrix. This is complicated slightly by the fact that EzPATH currently requires all manifest variables to appear in at least one path.

Consequently, since all manifest variables are exogenous, and some variances or covariances are specified, we cannot use the AUTOFIX command, and must in fact specify variance and covariance relationships for *all* manifest variables. The CMODEL command will simplify this by creating a PATH1 specification corresponding to a completely covariance matrix. You can then apply whatever constraints you wish to this matrix using the FEDIT file editor.

1. Test for Equality of Dependent Variances

As a simple example, consider some data gathered by William E. Coffman, and reported by Lord (1963). These data represent performance on the Stanford Achievement Test for 95 students measured in the seventh and eighth grades. The covariance matrix for the data is in a file called VAR2.SYS on the distribution disk. Suppose you wished to test the hypothesis that the population variance had not changed from seventh to eighth grade. The PATH1 specification for such a hypothesis (contained in the file VAR2.CMD on the distribution diskette) is as follows:

MODEL

```
[X1]-1-[X1]
[X2]-1-[X2]
[X1]-2-[X2]
```

By submitting the command file, you can obtain the following output for the hypothesis test:

```
Sample Size (N)           =          95
Degrees of Freedom       =           1
Chi-Square                =       15.4174
Probability Level        =           .0001
```

Noncentrality Based Fit indices:

Point Estimates:

```
Population Noncentrality Index = .1513
Steiger-Lind Adjusted RMS Index = .3889
Population Gamma Index = .8686
Adjusted Population Gamma Index = .6058
```

Confidence Intervals:

```
Population Noncentrality Index = .0481, .3120
Steiger-Lind Adjusted RMS Index = .2193, .5586
Population Gamma Index = .7622, .9541
Adjusted Population Gamma Index = .2865, .8624
```

Additional Single Sample Indices:

```
Joreskog-Sorbom GFI = .8686
Joreskog-Sorbom AGFI = .6058
Rescaled Akaike Criterion = .2066
Rescaled Schwarz Criterion = .2609
Browne-Cudeck Cross-Validation Index = .2080
```

Calculations Completed.

Here are the results for the fitted model.

```
MODEL
[X1]-1{ 168.100}-[X1]
[X2]-1{ 168.100}-[X2]
[X1]-2{ 144.295}-[X2]
```

The chi-square statistic indicates that the hypothesis of equal variances can be rejected, as the probability level is only about .0001.

2. Test for Compound Symmetry

A test for "compound symmetry" of the covariance matrix is sometimes performed in the context of repeated measures analysis of variance (See, e.g., Winer, 1971, p. 596-598). This hypothesis states that the covariance matrix equal diagonal elements, and equal off-diagonal elements. The file WINER.SYS

contains the pooled covariance matrix analyzed by Winer. Because the matrix was obtained by pooling two samples of size 5, it would have the same distribution (assuming both populations have the same covariance matrix) as a covariance matrix based on a single sample of size 9. Hence, we use **NUMBER=9** below. We test that the covariance matrix has the form

$$\begin{array}{ccc} \sigma^2 & \rho\sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \rho\sigma^2 & \sigma^2 \end{array}$$

Let's use this example to illustrate the use of the **CMODEL** command. Make sure the WINER.SYS file is in your SYSTAT directory.

Then type

```
USE WINER
CMODEL SYMCOV
```

This will create the following PATH1 commands in a file called SYMCOV.CMD.

MODEL

* Note: Highest parameter number created by EzPATH is 6

```
[X1]-1{ 3.100}-[X1]
```

```
[X2]-2{ 1.920}-[X1]
      -3{ 2.800}-[X2]
```

```
[X3]-4{ 1.820}-[X1]
      -5{ 2.000}-[X2]
      -6{ 3.800}-[X3]
```

Type

```
FEDIT "SYMCOV.CMD"
```

and edit the file so that the covariance matrix has equal diagonal elements, and equal off-diagonal elements.

As usual, wires from a variable to itself represent variances, and wires between different variables represent covariances. Variances are thus located at the bottom of each group of commands. Change all the variances first so that they all have coefficient number 1. Then change the covariances so that they all have coefficient number 2. There is no need to change the starting values so that they are the same for equivalent parameters. EzPATH simply uses the last value for each parameter. (I suggest removing the comment line from the file as well. It is no longer relevant.)

The file should look like this when you are done:

MODEL

```
[X1]-1{ 3.100}-[X1]
```

```
[X2]-2{ 1.920}-[X1]
      -1{ 2.800}-[X2]
```

```
[X3]-2{ 1.820}-[X1]
      -2{ 2.000}-[X2]
      -1{ 3.800}-[X3]
```

Hit the <F10> key and save the file. Then type the following commands

```
NUMBER=9
FORMAT=2
OUTPUT SYMCOV
SUBMIT SYMCOV
ESTIMATE
```

You will get the following output:

Sample Size (N) = 9
 Degrees of Freedom = 4
 Chi-Square = .6944
 Probability Level = .9520

Noncentrality Based Fit indices:

Point Estimates:

Population Noncentrality Index = .0000
 Steiger-Lind Adjusted RMS Index = .0000
 Population Gamma Index = 1.0000
 Adjusted Population Gamma Index = 1.0000

Confidence Intervals:

Population Noncentrality Index = .0000, .0000
 Steiger-Lind Adjusted RMS Index = .0000, .0000
 Population Gamma Index = 1.0000, 1.0000
 Adjusted Population Gamma Index = 1.0000, 1.0000

Additional Single Sample Indices:

Joreskog-Sorbom GFI = .9457
 Joreskog-Sorbom AGFI = .9185
 Rescaled Akaike Criterion = .5868
 Rescaled Schwarz Criterion = .6361
 Browne-Cudeck Cross-Validation Index = 1.0868

Calculations Completed.

Here are the results for the fitted model.

MODEL

[X1]-1{ 3.23}-[X1]
 [X2]-2{ 1.91}-[X1]
 -1{ 3.23}-[X2]
 [X3]-2{ 1.91}-[X1]
 -2{ 1.91}-[X2]
 -1{ 3.23}-[X3]

We restrict the output to two significant digits for comparability with the data in Winer (1971). Note that the maximum likelihood estimates under the null hypothesis correspond to the matrix S_0 on page 598 of the Winer text. We find that the chi square statistic, with

4 degrees of freedom, is only .694. This is not significant. In fact, the probability level is suspiciously high, as is sometimes seen with artificial data in ANOVA textbooks! This fact, coupled with the high values for the fit coefficients, suggests that the population matrix deviates only trivially from the hypothesized structure.

F. Test Theory Models for Sets of Congeneric Tests

A variety of interesting test theory models can be tested and estimated using EzPATH. These models are all special cases of the common factor model, and are discussed in Jöreskog (1974) on pages 49–56. The classical test theory model can be expressed as a common factor model. Suppose a group of tests are congeneric, i.e., have the same true scores underlying them. Then if the observed score is equal to the true score plus error, and error is uncorrelated with true scores, we have the path model shown in Figure 13a for two sets of two congeneric tests.

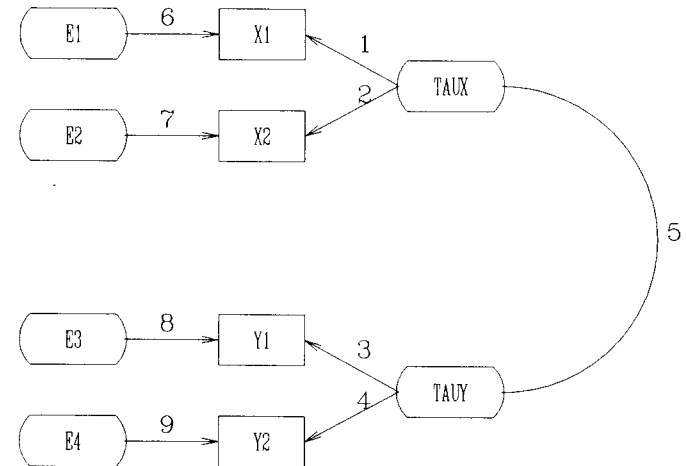


Figure 13a. A Model for Two Sets of Congeneric Tests.

Suppose you wished to assess whether two different tests measure the same trait. In classical test theory, we would say that the tests measure the same trait if their true scores are perfectly correlated, i.e., if the correlation coefficient between the tests, corrected for attenuation, is unity. Jöreskog (1978) refers to such tests as *tau-equivalent*.

Lord (1957) proposed a test for tau-equivalence. His test required the assumption that the elements of each pair of tests have equal variance and equal reliability, i.e., that the tests are *parallel*. The model corresponding to the hypothesis tested by Lord's method, i.e. that the tests are both parallel and tau-equivalent, is illustrated in Figure 13b.

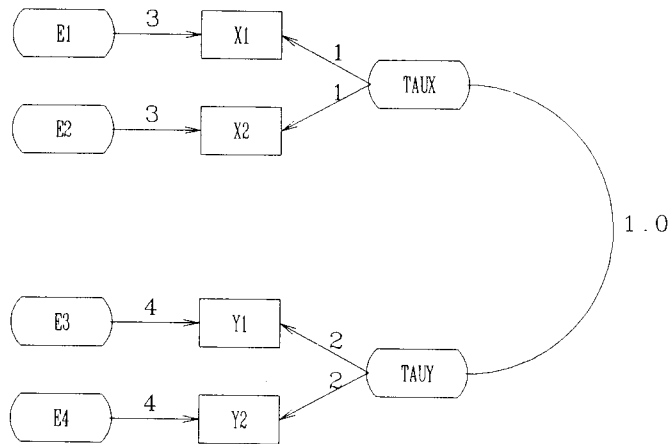


Figure 13b. A Model for Two Sets of Parallel, Tau-equivalent, Congeneric Tests.

It is possible to test the *assumptions* underlying Lord's (1957) test directly, i.e., we can test the hypothesis that the elements of each pair of tests are parallel (i.e., have equal variance and equal reliability) *without testing the hypothesis that the disattenuated correlation coefficient is unity*. This hypothesis is shown in Figure 13c.

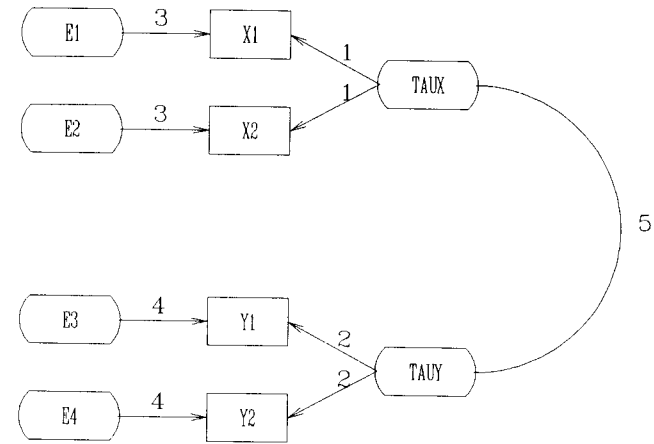


Figure 13c. A Model for Two Sets of Parallel Congeneric Tests.

Models B and C form a nested sequence, since B is a special case of C. If we compute the chi-square statistics for Models B and C, and take their difference, we obtain a chi-square difference statistic with one degree of freedom for testing the hypothesis that the disattenuated correlation is 1.

Lord's test is valid provided its assumptions are met. However, EzPATH allows us to test whether the disattenuated correlation coefficient is unity *without requiring*

the restrictions of Lord's test. Figure 13d illustrates a hypothesis that the tests are congeneric (but not necessarily parallel) and tau-equivalent.

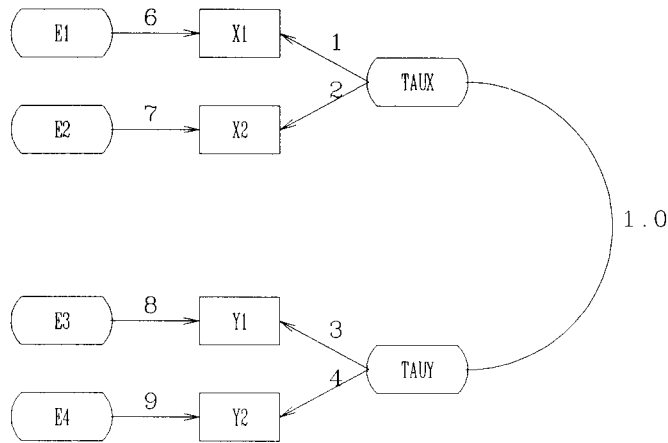


Figure 13d. A Model for Two Sets of Tau-equivalent Congeneric Tests.

Model D is a special case of model A, i.e., the two models form a nested sequence. Consequently, subtracting the chi-square statistics for model A from that obtained for model D will produce a chi-square statistic with one degree of freedom. This provides an alternative test that the disattenuated correlation is one.

Jöreskog tested models A,B,C, and D on some data given in Lord's (1957) paper. The results of the testing are summarized both in Jöreskog (1974) and in Jöreskog (1978).

The covariance matrix for this example is in the file LORD.SYS on the distribution diskette. The PATH1 statements for the 4 models are contained in the file LORD.CMD on the distribution disk. The following output (edited in the interest of brevity) can be obtained by SUBMITting the LORD command file.

```

RESULTS FOR HYPOTHESIS 4, FIGURE A
Sample Size (N) = 649
Degrees of Freedom = 1
Chi-Square = .7030
Probability Level = .4018

Noncentrality Based Fit indices:

Point Estimates:
Population Noncentrality Index = .0000
Steiger-Lind Adjusted RMS Index = .0000
Population Gamma Index = 1.0000
Adjusted Population Gamma Index = 1.0000

Confidence Intervals:
Population Noncentrality Index = .0000, .0095
Steiger-Lind Adjusted RMS Index = .0000, .0974
Population Gamma Index = .9953, 1.0000
Adjusted Population Gamma Index = .9528, 1.0000

Additional Single Sample Indices:
Joreskog-Sorbom GFI = .9995
Joreskog-Sorbom AGFI = .9946
Rescaled Akaike Criterion = .0289
Rescaled Schwarz Criterion = .0910
Browne-Cudeck Cross-Validation Index = .0291

Calculations Completed.

Here are the results for the fitted model.

MODEL
(TAUX)-1{ 7.501 SE= 0.323}->{X1}
-2{ 7.703 SE= 0.320}->{X2}
(TAUY)-3{ 8.509 SE= 0.327}->{Y1}
-4{ 8.675 SE= 0.325}->{Y2}
(TAUX)-5{ 0.899 SE= 0.019}-(TAUY)
(E1)-6{ 5.490 SE= 0.225}->{X1}

```

(E2)-7{ 5.189 SE= 0.234}->[X2]
 (E3)-8{ 4.988 SE= 0.236}->[Y1]
 (E4)-9{ 4.750 SE= 0.247}->[Y2]

Terminating processing for this model.

RESULTS FOR HYPOTHESIS 1, FIGURE B

Sample Size (N) = 649
 Degrees of Freedom = 6
 Chi-Square = 37.3337
 Probability Level = .0000

Noncentrality Based Fit indices:

Point Estimates:

Population Noncentrality Index = .0529
 Steiger-Lind Adjusted RMS Index = .0939
 Population Gamma Index = .9742
 Adjusted Population Gamma Index = .9571

Confidence Intervals:

Population Noncentrality Index = .0264, .0880
 Steiger-Lind Adjusted RMS Index = .0664, .1211
 Population Gamma Index = .9578, .9870
 Adjusted Population Gamma Index = .9297, .9783

Additional Single Sample Indices:

Joreskog-Sorbom GFI = .9705
 Joreskog-Sorbom AGFI = .9509
 Rescaled Akaike Criterion = .0700
 Rescaled Schwarz Criterion = .0976
 Browne-Cudeck Cross-Validation Index = .0701

Calculations Completed.

Here are the results for the fitted model.

MODEL

(TAUX)-1{ 7.186 SE= 0.266}->[X1]
 -1{ 7.186 SE= 0.266}->[X2]
 (TAUY)-2{ 8.442 SE= 0.280}->[Y1]
 -2{ 8.442 SE= 0.280}->[Y2]
 (TAUX)-{1.0}- (TAUY)
 (E1)-3{ 5.890 SE= 0.140}->[X1]

(E2)-3{ 5.890 SE= 0.140}->[X2]
 (E3)-4{ 5.127 SE= 0.136}->[Y1]
 (E4)-4{ 5.127 SE= 0.136}->[Y2]

Terminating processing for this model.

RESULTS FOR HYPOTHESIS 2, FIGURE C

Sample Size (N) = 649
 Degrees of Freedom = 5
 Chi-Square = 1.9335
 Probability Level = .8583

Noncentrality Based Fit indices:

Point Estimates:

Population Noncentrality Index = .0000
 Steiger-Lind Adjusted RMS Index = .0000
 Population Gamma Index = 1.0000
 Adjusted Population Gamma Index = 1.0000

Confidence Intervals:

Population Noncentrality Index = .0000, .0044
 Steiger-Lind Adjusted RMS Index = .0000, .0296
 Population Gamma Index = .9978, 1.0000
 Adjusted Population Gamma Index = .9956, 1.0000

Additional Single Sample Indices:

Joreskog-Sorbom GFI = .9985
 Joreskog-Sorbom AGFI = .9970
 Rescaled Akaike Criterion = .0184
 Rescaled Schwarz Criterion = .0529
 Browne-Cudeck Cross-Validation Index = .0185

Calculations Completed.

Here are the results for the fitted model.

MODEL

(TAUX)-1{ 7.601 SE= 0.268}->[X1]
 -1{ 7.601 SE= 0.268}->[X2]
 (TAUY)-2{ 8.592 SE= 0.279}->[Y1]
 -2{ 8.592 SE= 0.279}->[Y2]
 (TAUX)-5{ 0.899 SE= 0.019}- (TAUY)
 (E1)-3{ 5.344 SE= 0.148}->[X1]
 (E2)-3{ 5.344 SE= 0.148}->[X2]
 (E3)-4{ 4.872 SE= 0.135}->[Y1]
 (E4)-4{ 4.872 SE= 0.135}->[Y2]

Terminating processing for this model.

Sample Size (N)	=	649
Degrees of Freedom	=	2
Chi-Square	=	36.2095
Probability Level	=	.0000

Noncentrality Based Fit indices:

Point Estimates:

Population Noncentrality Index	=	.0573
Steiger-Lind Adjusted RMS Index	=	.1693
Population Gamma Index	=	.9722
Adjusted Population Gamma Index	=	.8608

Confidence Intervals:

Population Noncentrality Index	=	.0304,	.0926
Steiger-Lind Adjusted RMS Index	=	.1232,	.2152
Population Gamma Index	=	.9557,	.9850
Adjusted Population Gamma Index	=	.7786,	.9252

Additional Single Sample Indices:

Joreskog-Sorbom GFI	=	.9714
Joreskog-Sorbom AGFI	=	.8571
Rescaled Akaike Criterion	=	.0806
Rescaled Schwarz Criterion	=	.1358
Browne-Cudeck Cross-Validation Index	=	.0808

Calculations Completed.

Here are the results for the fitted model.

MODEL

(TAUX)-1{ 7.104 SE= 0.322}->{X1}
-2{ 7.269 SE= 0.318}->{X2}

(TAUY)-3{ 8.374 SE= 0.325}->{Y1}
-4{ 8.511 SE= 0.324}->{Y2}

(TAUX)-(1.0)-(TAUY)

(E1)-6{ 5.994 SE= 0.201}->{X1}
(E2)-7{ 5.782 SE= 0.200}->{X2}
(E3)-8{ 5.212 SE= 0.215}->{Y1}
(E4)-9{ 5.038 SE= 0.219}->{Y2}

Both chi-square difference tests yield the same conclusion: the hypothesis that the disattenuated correlation coefficient is 1 can be resoundingly rejected. For example, the chi-square difference statistic (with one degree of freedom) for Model A versus Model D is 35.51. On the other hand, the maximum likelihood estimate for the disattenuated correlation is quite high (.899). Moreover, the standard error for this value is only .019.

G. Multitrait-Multimethod Factor Models

When personality traits or characteristics are measured, variation among people can occur for several reasons. Two obvious contributing factors are variation in the traits themselves, and variations in the way people react to a particular method.

When a trait is measured by only one method, there is a possibility that the variation observed is actually *method variance*, rather than trait variance. For example, if a particular questionnaire does not control for acquiescence response set, variation among people due to a problem in the method is confounded with actual trait variation.

One way around this problem is to try to measure both trait and method variation in the same experiment. The *multitrait-multimethod* correlation matrix contains correlations between t traits or characteristics each measured by the same m methods. Campbell and Fiske (1959) suggested that the multitrait-multimethod correlation matrix should be examined to provide evidence of construct validity.

In their original work, Campbell and Fiske suggested that two kinds of validity, which they termed *convergent validity* and *discriminant validity*, could be evaluated by examining this matrix. There are 4 kinds of correlations in the matrix; (1) same-trait, same-method; (2) same-trait, different-method; (3) different-trait, same-method; (4) different-trait, different-method. Convergent validity is demonstrated if same-trait, different method correlations are large.

Discriminant validity is evidenced if same—trait different—method correlations are substantially higher than different—trait, different—method correlations.

Kenny (1979) analyzed data from Jaccard, Weber, and Lundmark (1975). Their study measured attitude toward cigarette smoking (C) and attitude toward capital punishment (P) with 4 different methods. The methods were: (1) semantic differential, (2) Likert, (3) Thurstone, (4) Guilford. The correlation matrix, based on only 35 observations, is in a file called JACCARD.SYS on the distribution diskette.

Kenny used a classic test theory approach to analyzing the data:

The traits are factors whereas the disturbances or unique factors are allowed to be correlated across measures using the same method. Such a model is identified if there are at least two traits and three methods. Assuming the model fits the data, then convergent validation is assessed by high loadings on the trait factors, discriminant validation by low to moderate correlations between the trait factors, and method variance by highly correlated disturbances.

The EzPATH diagram for the resulting model is shown in Figure 14.

The PATH1 specification for the model is in a file called JACCARD.CMD on the distribution diskette. Running this model (which is in a file called JACCARD.CMD on the distribution diskette) on the data produces the output shown below:

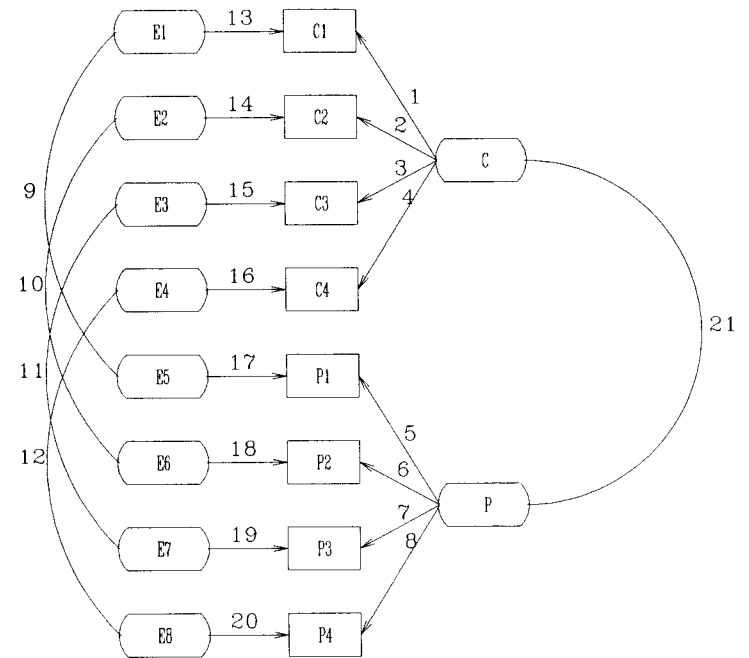


Figure 14. A multitrait—multimethod model.

Sample Size (N)	=	35
Degrees of Freedom	=	15
Chi-Square	=	10.3540
Probability Level	=	.7969

Noncentrality Based Fit indices:

Point Estimates:		
Population Noncentrality Index	=	.0000
Steiger-Lind Adjusted RMS Index	=	.0000
Population Gamma Index	=	1.0000
Adjusted Population Gamma Index	=	1.0000

Confidence Intervals:

Population Noncentrality Index	=	.0000,	.1674
Steiger-Lind Adjusted RMS Index	=	.0000,	.1056
Population Gamma Index	=	.9598,	1.0000
Adjusted Population Gamma Index	=	.9036,	1.0000

Additional Single Sample Indices:

Joreskog-Sorbom GFI	=	.9297
Joreskog-Sorbom AGFI	=	.8313
Rescaled Akaike Criterion	=	1.5398
Rescaled Schwarz Criterion	=	2.5005
Browne-Cudeck Cross-Validation Index	=	1.9845

Calculations Completed.

Here are the results for the fitted model.

MODEL

```
(C)-1{ 0.893 SE= 0.133}->[C1]
-2{ 0.853 SE= 0.137}->[C2]
-3{ 0.914 SE= 0.130}->[C3]
-4{ 0.860 SE= 0.135}->[C4]

(P)-5{ 0.867 SE= 0.132}->[P1]
-6{ 0.957 SE= 0.123}->[P2]
-7{ 0.916 SE= 0.130}->[P3]
-8{ 0.964 SE= 0.128}->[P4]

(E1)-13{ 0.449 SE= 0.075}->[C1]
(E2)-14{ 0.532 SE= 0.078}->[C2]
(E3)-15{ 0.400 SE= 0.077}->[C3]
(E4)-16{ 0.506 SE= 0.076}->[C4]

(E5)-17{ 0.477 SE= 0.064}->[P1]
(E6)-18{ 0.263 SE= 0.059}->[P2]
(E7)-19{ 0.409 SE= 0.059}->[P3]
(E8)-20{ 0.329 SE= 0.057}->[P4]

(E1)-9{ 0.183 SE= 0.201}->[E5]
(E2)-10{ 0.293 SE= 0.229}->[E6]
(E3)-11{ 0.208 SE= 0.218}->[E7]
(E4)-12{ 0.270 SE= 0.206}->[E8]

(P)-21{ 0.248 SE= 0.167}->[C]
```

Kenny remarked that the data fit the model well in this case. With statistically-based fit indices, and the 20-20 vision of hindsight, we can see that the issue is very much in doubt. The sample size is so small that the confidence intervals for the statistically-based fit indices are quite wide. For example, the 90% confidence interval for the Steiger-Lind

RMS index ranges from 0 to .1056. In practice, we would prefer a significantly larger sample size.

H. Longitudinal Factor Analysis Models

Corballis and Traub (1970) presented a longitudinal factor analysis model, which stipulates that factorial structure underlying a set of tests remains constant over two or more administrations of the tests. An example of such a model is given by Everitt (1984, pages 52-55). The data were from a study by Meyer and Bendig (1961), who administered the 5 Thurstone Primary Mental Abilities tests to 49 boys and 61 girls in grades 8 and 11⁴. The tests are Verbal Meaning (V), Space (S), Reasoning (R), Numerical (N), and word fluency (W). The correlation matrix for these data are in a file called MEYER.SYS on the distribution diskette.

The model analyzed by Everitt (1984) stipulates a single common factor underlying the scores on both occasions. The EzPATH diagram for the model is in Figure 15.

The PATH1 translation of the diagram is contained in a file called MEYER.CMD on the distribution diskette.

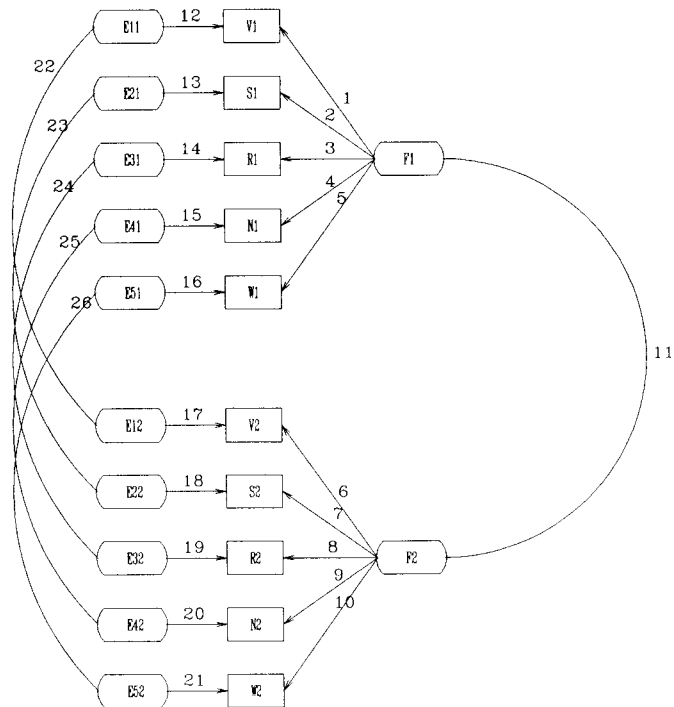


Figure 15. A Longitudinal Factor Analysis Model.

Here is some of the output.

Sample Size (N) = 110
 Degrees of Freedom = 29
 Chi-Square = 51.4562
 Probability Level = .0063

Noncentrality Based Fit indices:

Point Estimates:

Population Noncentrality Index = .1671
 Steiger-Lind Adjusted RMS Index = .0759
 Population Gamma Index = .9677
 Adjusted Population Gamma Index = .9387

Confidence Intervals:

Population Noncentrality Index = .0261, .3704
 Steiger-Lind Adjusted RMS Index = .0300, .1130
 Population Gamma Index = .9310, .9948
 Adjusted Population Gamma Index = .8692, .9902

Additional Single Sample Indices:

Joreskog-Sorbom GFI = .9211
 Joreskog-Sorbom AGFI = .8504
 Rescaled Akaike Criterion = .9491
 Rescaled Schwarz Criterion = 1.5933
 Browne-Cudeck Cross-Validation Index = 1.0027

Calculations Completed.

Here are the results for the fitted model.

MODEL

(F1)-1(0.768 SE= 0.096)->[V1]
 -2(0.406 SE= 0.101)->[S1]
 -3(0.575 SE= 0.097)->[R1]
 -4(0.696 SE= 0.096)->[N1]
 -5(0.397 SE= 0.102)->[W1]

 (F2)-6(0.839 SE= 0.092)->[V2]
 -7(0.314 SE= 0.103)->[S2]
 -8(0.606 SE= 0.095)->[R2]
 -9(0.659 SE= 0.096)->[N2]

 -10(0.292 SE= 0.103)->[W2]

 (F1)-11(0.953 SE= 0.027)-(F2)

 (E11)-12(0.635 SE= 0.077)->[V1]
 (E21)-13(0.917 SE= 0.065)->[S1]
 (E31)-14(0.810 SE= 0.063)->[R1]
 (E41)-15(0.734 SE= 0.067)->[N1]
 (E51)-16(0.918 SE= 0.065)->[W1]

 (E12)-17(0.546 SE= 0.088)->[V2]
 (E22)-18(0.951 SE= 0.066)->[S2]
 (E32)-19(0.785 SE= 0.063)->[R2]
 (E42)-20(0.756 SE= 0.065)->[N2]
 (E52)-21(0.957 SE= 0.066)->[W2]

 (E11)-22(0.557 SE= 0.118)-(E12)
 (E21)-23(0.614 SE= 0.061)-(E22)
 (E31)-24(0.619 SE= 0.066)-(E32)
 (E41)-25(0.573 SE= 0.078)-(E42)

(E51)-26(0.366 SE= 0.085)-(E52)

The chi-square value of 51.46 allows us to reject the null hypothesis of *perfect* fit. On the other hand, the noncentrality-based fit indices indicate that the jury is, in a sense, still out regarding whether the fit of this model is acceptable. Consider, for example, the Steiger-Lind RMS index. The point estimate is .076, but the confidence interval ranges from .03 to .11. Basically, this indicates that the sample size of 110 is insufficient to determine, with adequate precision, the quality of the population fit.

I. Effect of Peer Influences on Ambition

Duncan, Haller, and Portes (1968) analyzed the effect of peer influences on ambition. The correlation matrix from their study, based on 329 subjects, is contained in the file DHP.SYS on the distribution diskette. Their data have been analyzed in a number of publications, and I will not discuss the substantive content of the example here. Jöreskog and Sörbom (1984) present the results from several path models on these data. Here we analyze the model the results for which are given in Table III.12 of Jöreskog and Sörbom. The EzPATH diagram is shown below in Figure 16. PATH1 statements corresponding to the diagram are in a file called DHPA.CMD on the distribution diskette.

If you study these commands, you will notice that the **AUTOFIX** command is used with this model. The command is used because of the existence in the model of a number of manifest exogenous variables (located on the left hand side of the diagram). We have taken the convenient option here and not specified *any* of the variance-covariance relationships for these variables. If you give the **AUTOFIX=YES** command, EzPATH will handle these relationships automatically. If you fail to include the command, EzPATH will give an error message and abort the analysis, unless you add paths for all of the variance-covariance relationships among these variables.

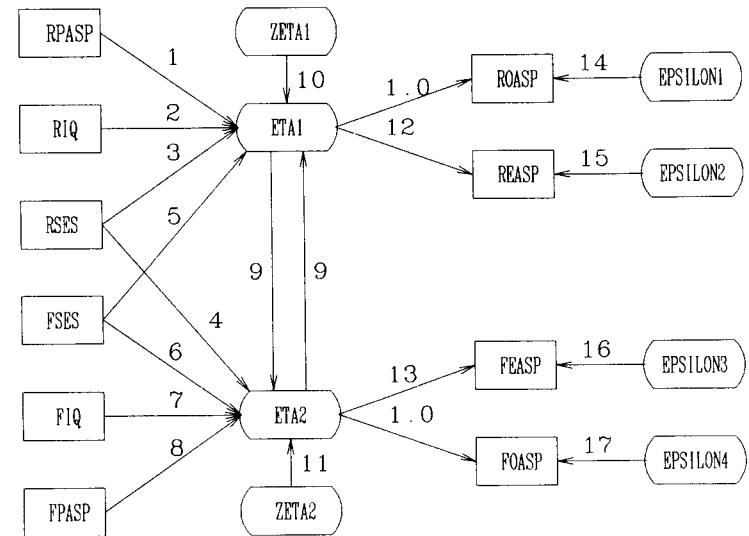


Figure 16. A Model for the Effects of Peer Influence on Aspiration.

The output below gives both the unscaled and standardized maximum likelihood solutions.

```

Sample Size (N)                =          329
Degrees of Freedom              =           17
Chi-Square                     =       26.8987
Probability Level               =         .0596
.

Noncentrality Based Fit indices:

Point Estimates:
Population Noncentrality Index  =         .0295
Steiger-Lind Adjusted RMS Index =         .0417
Population Gamma Index         =         .9941
Adjusted Population Gamma Index =         .9810

Confidence Intervals:
Population Noncentrality Index  =         .0000,         .0826
Steiger-Lind Adjusted RMS Index =         .0000,         .0697
Population Gamma Index         =         .9838,         1.0000

```

Adjusted Population Gamma Index = .9474, 1.0000

Additional Single Sample Indices:

Joreskog-Sorbom GFI = .9843
Joreskog-Sorbom AGFI = .9492
Rescaled Akaike Criterion = .3137
Rescaled Schwarz Criterion = .7535
Browne-Cudeck Cross-Validation Index = .3218

Calculations Completed.

Here are the results for the fitted model.

MODEL

* Duncan Haller Portes -- Original Model in Lisrel IV Manual

[RPASP]-1{ 0.164 SE= 0.039}->(ETA1)
[RIQ]-2{ 0.254 SE= 0.042}->(ETA1)
[RSES]-3{ 0.221 SE= 0.042}->(ETA1)
-4{ 0.068 SE= 0.039}->(ETA2)
[FSES]-5{ 0.077 SE= 0.041}->(ETA1)
-6{ 0.218 SE= 0.039}->(ETA2)
[FIQ]-7{ 0.331 SE= 0.041}->(ETA2)
[FPASP]-8{ 0.152 SE= 0.036}->(ETA2)

(ETA1)-9{ 0.180 SE= 0.039}->(ETA2)
(ETA2)-9{ 0.180 SE= 0.039}->(ETA1)

(ZETA1)-10{ 0.530 SE= 0.044}->(ETA1)
(ZETA2)-11{ 0.479 SE= 0.041}->(ETA2)

(ETA1)-12{ 1.061 SE= 0.089}->[REASP]

-->[ROASP]

(ETA2)-->[FOASP]

-13{ 1.074 SE= 0.081}->[FEASP]

(EPSILON1)-14{ 0.581 SE= 0.044}->[REASP]
(EPSILON2)-15{ 0.642 SE= 0.040}->[ROASP]
(EPSILON3)-16{ 0.560 SE= 0.041}->[FEASP]
(EPSILON4)-17{ 0.635 SE= 0.036}->[FOASP]

Here are the results for the Standardized Solution.

* Duncan Haller Portes -- Original Model in Lisrel IV Manual

[RPASP]-1{ 0.214}->(ETA1)
[RIQ]-2{ 0.331}->(ETA1)
[RSES]-3{ 0.288}->(ETA1)
-4{ 0.089}->(ETA2)
[FSES]-5{ 0.101}->(ETA1)
-6{ 0.283}->(ETA2)
[FIQ]-7{ 0.429}->(ETA2)
[FPASP]-8{ 0.197}->(ETA2)

(ETA1)-9{ 0.179}->(ETA2)
(ETA2)-9{ 0.181}->(ETA1)

(ZETA1)-10{ 0.692}->(ETA1)
(ZETA2)-11{ 0.621}->(ETA2)

(ETA1)-12{ 0.813}->[REASP]

-{ 0.767}->[ROASP]
(ETA2)-{ 0.771}->[FOASP]
-13{ 0.828}->[FEASP]

(EPSILON1)-14{ 0.581}->[REASP]
(EPSILON2)-15{ 0.642}->[ROASP]
(EPSILON3)-16{ 0.560}->[FEASP]
(EPSILON4)-17{ 0.635}->[FOASP]

10. Hints on Using EzPATH.

A. Generating a Covariance Matrix for Your Own Model.

EzPATH allows you to **SAVE** an estimated variance covariance matrix in a SYSTAT file. This facility makes it possible to generate the covariance matrix corresponding to a model you have specified. Having the ability to do this can be quite useful. For example, if you are teaching a causal modeling course, you could create a model with your own desired coefficients as "starting values," then generate the covariance matrix corresponding to the model, and see if your students can deduce, from substantive and/or statistical considerations, what the "true" model was.

Another, less obvious use for this technique occurs when you want to try an alternative model to one in a published source, but the published source doesn't include the covariance matrix. Sometimes, in such cases, you can approximate the covariance matrix using the published model coefficients.

To use this facility, follow these steps:

1. Use the SYSTAT data editor to create a "dummy" covariance matrix. The matrix need contain no data. Just power up the editor, enter the correct manifest variable names, and then save the file.
2. Create you model file as a .CMD file, with your desired coefficients as starting values.
3. Power up EzPATH. Be sure you have set **ITERATIONS=0, METHOD=LS, PRINT=LONG**. Then issue a **SAVE** <File Name> command to save the output matrix to a file called <File Name>.HAT.

4. **ESTIMATE** your model. The estimated variance covariance matrix printed by the program will be saved as a SYSTAT binary file. You can then analyse it like any other covariance matrix, except you will know the model which fits it perfectly!

B. Avoiding System Crashes and Data Loss

Everyone, at one time or another, gets to see the infamous SYSTAT "Input Output Error" message, following which you are unceremoniously beeped back to the operating system. As irritating as this development may be, it actually occurs to protect you from potential major data loss. The problem is, when certain input-output errors occur, DOS can go crazy and start trashing your hard disk. SYSTAT software engineers decided that most users would rather risk losing a little work than megabytes worth.

First, remember the following. EzPATH uses lots of files, and generates lots of output. If you run out of space on the diskette while a write operation is taking place, you will probably crash the system. To minimize the chances of this happening, try not to run the program from a floppy disk, unless it is 1.44M or 1.2M. You're just asking for trouble unless you start with plenty of free space on your directory.

Second, use the **OUTPUT** command liberally. **SAVE** your command log periodically as well. Simply type **FEDIT >**, and enter a single blank at the end of the command log. Then hit **F10** and save the file (I routinely use the filename "SESSION" for my command log). Do this periodically! These two precautionary measures will mean that you will practically never lose any important work.

If the system does crash, (which will seldom happen), check for the screen buffer and command files, SYSTAT\$.LOG and SCREEN\$.BUF. They may be there, with much of your session intact.

Most system crashes with EzPATH will be avoidable input-output problems. Occasionally, it is possible an internal math

error will occur, usually due to arithmetic overflow or underflow. There is extensive error trapping in EzPATH to try to prevent this from happening. I believe you will encounter far fewer "MATH ERROR" system crashes from EzPATH than from most complex programs of its type. Let me know if you have a math error crash and can replicate it. If you send me your input, I'll try to correct the problem.

C. Plotting Normalized Residuals.

If you issue a **SAVE** command in conjunction with **PRINT=LONG**, EzPATH saves normalized residuals into a file with the extension .RES. The variable RESIDUAL in this file will be the normalized residual values. You can do a PLOT with GRAPH or SYGRAPH modules to examine these residuals.

D. When Your Output Doesn't Agree with That Other Program's.

There are several ways to estimate the same causal model. Sometimes this can make it seem like two different programs have obtained different results for the same model. For example, LISREL routinely estimates variances for exogenous latent variables in such matrices as "THETA DELTA." If you use the default approach in EzPATH, there will be no "THETA DELTA." You will not be estimating exogenous latent variable variances, because they will by default be fixed at one. Rather, you will be estimating path coefficients *from* the latent variables to some other variable. *These* coefficients are usually fixed at one by LISREL. The advantage of the default EzPATH approach is that, in effect, latent variable variances cannot "go negative" during iteration. Another advantage is that it requires fewer explicit paths.

Of course, by adding explicit pathways for exogenous latent variable variances, and setting the path coefficients to one, you can mimic the LISREL output.

The preceding discussion covers what is generally the reason for "different" results from LISREL and EzPATH. Generally it is possible to get virtually identical output from covariance structure programs for analyses which they can handle. However, given the vagaries of non-linear estimation routines, the programs will in general converge to different solutions. Usually the one with the lower loss function is the correct solution in that case. Try using the results from one program as the starting values for the other. If one program has the correct results, and the other does not obtain the same discrepancy function value, then the problem almost certainly is a faulty model set up on one or both programs.

E. Watch Out for the Lonesome Latent Variable!

One of the nastiest problems ever encountered by a student of mine using EzPATH kept her puzzled for more than an hour. It turned out that, embedded in her output, was a path to a latent variable which she had *spelled incorrectly*. This created a unique latent variable which was basically sitting there by itself. EzPATH had no way of knowing that she didn't intend for the variable to exist. The result was that a key path was missing, and her results were bizarre. Chances are sometime we'll get a warning message for such things built into the EzPATH parser.

F. Entering A Covariance or Correlation Matrix.

When you use EzPATH to try to replicate a published analysis, you will, of course, want to enter the covariance or correlation matrix. An example of how this is done, using the DATA module, is on page 291 of the SYSTAT manual.

G. Program Limitations

EzPATH Version 1.0 is designed to handle problems with 40 or fewer manifest variables, and 150 or fewer unknowns. Moreover, the total number of relationships cannot exceed 190. In addition, there are limitations on the amount of array

workspace available. If any of these limitations is exceeded, the program will issue an error message.

Keep in mind that some parameterizations of a model require more paths than others. The classic case is the parameterization of a unique variance. EzPATH allows you to express this with a single path with a regression coefficient, (the latent variable's variance will be fixed at one by default), or two paths, one involving a regression coefficient of one, the other representing a latent variable variance.

In other words, this

$$\begin{matrix} (\mathbf{F}) \rightarrow [\mathbf{X}] \\ (\mathbf{F}) - 1 - (\mathbf{F}) \end{matrix}$$

is equivalent to this

$$(\mathbf{F}) - 1 \rightarrow [\mathbf{X}] ,$$

although the value of coefficient 1 in the first version will be the square of the value in the latter.

If you try to directly represent results from LISREL, you will find yourself using the less efficient parameterization. This less efficient parameterization might cause you to exceed a program limitation, in which case you could revert to the standard EzPATH format.

References

- Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In B. N. Petrov and F. Csaki (Eds.), *Second International Symposium on Information Theory*. Budapest: Akademiai Kiado.
- Akaike, H. (1983). Information measures and model selection. *Bulletin of the International Statistical Institute: Proceedings of the 44th Session, Volume 1*. Pages 277–290.
- Anderson, S.B. & Maier, M.H. (1963). 34,000 pupils and how they grew. *Journal of Teacher Education*, 14, 212–216.
- Anderson, T.W., & Rubin, H. (1956). Statistical inference in factor analysis. *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*. Berkeley: The University of California Press.
- Bartholomew, D.J. (1984). The foundations of factor analysis. *Biometrika*, 71, 221–232.
- Bentler, P.M., & Weeks, D.G. (1979). Interrelations among models for the analysis of moment structures. *Multivariate Behavioral Research*, 14, 169–185.
- Bentler, P.M., & Weeks, D.G. (1980). Linear structural equations with latent variables. *Psychometrika*, 45, 289–308.
- Bentler, P.M. (1986). Structural modeling and Psychometrika: A historical perspective on growth and achievements. *Psychometrika*, 51, 35–51.
- Browne, M.W. (1984). Asymptotically distribution free methods for the analysis of covariance structures. *British Journal of Mathematical and Statistical Psychology*, 37, 62–83.
- Browne, M.W., & Cudeck, R. (1989). Single sample cross-validation indices for covariance structures. *Multivariate Behavioral Research*, in press.

- Campbell D.T. & Fiske, D.W. (1959). Convergent and discriminant validation by the multitrait–multimethod matrix. *Psychological Bulletin*, 56, 81–105
- Cliff, N. (1983). Some cautions concerning the application of causal modeling methods. *Multivariate Behavioral Research*, 18, 115–126.
- Cooley, W.W., & Lohnes, P.R. (1976). *Evaluation research in education*. New York: Wiley.
- Corballis, M.C., & Traub, R.E. (1970). Longitudinal factor analysis. *Psychometrika*, 35, 79–98.
- Cudeck, R., & Browne, M.W. (1983). Cross–validation of covariance structures. *Multivariate Behavioral Research*, 18, 147–167.
- Dennis, J.E., & Schnabel, R.B. (1983). *Numerical methods for unconstrained optimization and nonlinear equations*. Englewood Cliffs, NJ: Prentice Hall.
- Duncan, O.D., Haller, A.O., & Portes, A. (1968). Peer influence on aspiration: a reinterpretation. *American Journal of Sociology*, 74, 119–137.
- Everitt, B.S. (1984) *An introduction to latent variable models*. London: Chapman and Hall.
- Fraser, C. & McDonald R.P. (1988). COSAN: Covariance structure analysis. *Multivariate Behavioral Research*, 23, 263–265.
- Glass, G.V., & Stanley, J. (1970). *Statistical methods in education and Psychology*. Englewood Cliffs, NJ: Prentice–Hall.
- Hilton, T.L. (1969). *Growth study annotated bibliography*. Princeton, NJ: Educational Testing Service Progress Report 69–11.
- Huba, G.J., & Harlow, L.L. (1987). Robust structural equation models: implications for developmental psychology. *Child Development*, 58, 147–166.
- Jaccard, J., Weber, J., & Lundmark, J. (1975). A multitrait–multimethod factor analysis of four attitude assessment procedures. *Journal of Experimental Social Psychology*, 11, 149–154.
- James, L.R., Mulaik, S.A., & Brett, J.M. (1982). *Causal analysis. Assumptions, models, and data*. Beverly Hills, CA: Sage Publications.
- Jöreskog, K.G. (1973). A general model for estimating a linear structural equation system. In A.S. Goldberger and O.D. Duncan (Eds.) *Structural Equation Models in the Social Sciences*. New York: Seminar Press.
- Jöreskog, K.G. (1974). Analyzing psychological data by structural analysis of covariance matrices. In David H. Krantz, Richard C. Atkinson, R. Duncan Luce, and Patrick Suppes (Eds.) *Contemporary Developments in Mathematical Psychology*, Vol. II. New York: W.H. Freeman and Company.
- Jöreskog, K.G. (1978). Structural analysis of covariance and correlation matrices. *Psychometrika*, 43, 443–477.
- Jöreskog, K.G., & Sörbom, D. (1984). *Lisrel VI. Analysis of linear structural relationships by maximum likelihood, instrumental variables, and least squares methods*. Mooresville, Indiana: Scientific Software.
- Kelly, E.L., & Fiske D.W. *The prediction of performance in clinical psychology*. Ann Arbor: University of Michigan Press.
- Kenny, David A. (1979). *Correlation and causality*. New York: Wiley.
- Lord, F.M. 1957). A significance test for the hypothesis that two variables measure the same trait except for errors of measurement. *Psychometrika*, 22, 207–220.

- Matsueda, R.L., & Bielby, W.T. (1986). Statistical power in covariance structure models. In N.B. Tuma (Ed.) *Sociological methodology*. Washington, D.C.: American Sociological Association.
- McArdle, J.J. (1978). A structural view of structural models. Paper presented at the *Winter Workshop on Latent Structure Models Applied to Developmental Data*, University of Denver, December, 1978.
- McArdle, J.J., & McDonald, R.P. (1984). Some algebraic properties of the Reticular Action Model for moment structures. *British Journal of Mathematical and Statistical Psychology*, 37, 234–251/
- McDonald, R.P., & Mulaik, S.A. (1979). Determinacy of common factors: A nontechnical review. *Psychological Bulletin*, 86, 297–306.
- McDonald, R.P. (1980). A simple comprehensive model for the analysis of covariance structures. *British Journal of Mathematical and Statistical Psychology*, 31, 59–72.
- McDonald, R.P. (1989). An index of goodness-of-fit based on noncentrality. *Journal of Classification*, in press.
- Saxena, K.M.L., & Alam, K. (1982). Estimation of the noncentrality parameter of a chi squared distribution. *Annals of Statistics*, 10, 1012–1016.
- Schnabel, R.B., Koontz, J.E., and Weiss, B.E. (1985). A modular system of algorithms for unconstrained minimization. *ACM Transactions on Mathematical Software*. 11, 419–440.
- Schönemann, P.H., & Steiger, J.H. (1976). Regression component analysis. *British Journal of Mathematical and Statistical Psychology*, 29, 175–189.
- Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6, 461–464.
- Spruill, M.C. (1986). Computation of the maximum likelihood estimate of a noncentrality parameter. *Journal of Multivariate Analysis*, 18, 216–224.
- Steiger, J.H. (1979). Factor indeterminacy in the 1930's and in the 1970's; some interesting parallels. *Psychometrika*, 44, 157–167.
- Steiger, J.H. (1980a). Tests for comparing elements of a correlation matrix. *Psychological Bulletin*, 87, 245–251.
- Steiger, J.H. (1980b). Testing pattern hypotheses on correlation matrices: Alternative statistics and some empirical results. *Multivariate Behavioral Research*, 15, 335–352.
- Steiger, J.H. (1988). Aspects of person-machine communication in structural modeling of correlations and covariances. *Multivariate Behavioral Research*, 23, 281–290.
- Steiger, J.H. (1989). Noncentrality interval estimation and the evaluation of statistical models. Manuscript in preparation.
- Steiger, J.H., & Browne, M.W. (1984). The comparison of interdependent correlations between optimal linear composites. *Psychometrika*, 49, 11–24.
- Steiger, J.H., & Hakstian, A.R. (1982). The asymptotic distribution of elements of a correlation matrix: Theory and application. *British Journal of Mathematical and Statistical Psychology*, 35, 208–215.
- Steiger, J.H., & Lind, J.C. (1980). Statistically-based tests for the number of common factors. Paper presented at the annual Spring Meeting of the Psychometric Society in Iowa City. May 30, 1980.
- Steiger, J.H., & Schönemann, P.H. (1978). A history of factor indeterminacy. In S. Shye, (Ed.), *Theory Construction and Data Analysis in the Social Sciences*. San Francisco: Jossey-Bass.

- Steiger, J.H., Shapiro, A., & Browne, M.W. (1985). On the multivariate asymptotic distribution of sequential chi-square statistics. *Psychometrika*, 50, 253–264.
- Tanaka, J.S., & Huba, G.J. A fit index for covariance structure models under arbitrary GLS estimation. *British Journal of Mathematical and Statistical Psychology*, 38, 197–201.
- Tanaka, J.S., & Huba, G.J. A general coefficient of determination for covariance structure models under arbitrary GLS estimation. Unpublished Manuscript. August, 1988.
- Velicer, W.F., & Jackson, D.N. (1989). Component analysis vs. factor analysis: some issues in selecting an appropriate procedure. *Multivariate Behavioral Research*, in press.
- Wheaton, B., Múthen, B., Alwin, D., & Summers G. (1977). Assessing reliability and stability in panel models. In D.R. Heise (Ed.) *Sociological Methodology*. New York: Wiley.
- Wilson, G.A., & Martin, S.A. (1983). An empirical comparison of two methods of testing the significance of a correlation matrix. *Educational and Psychological Measurement*, 43, 11–14.
- Winer, B.J. (1971). *Statistical principles in experimental design* (Second Ed.). New York: McGraw Hill.

Appendix – SAVE File Format

The normalized residuals for the entire covariance matrix are saved in a file called <file name>.RES.

This file is formatted as follows, There are three variable names in the file: ROW, COLUMN, and RESIDUAL. ROW refers to the row number for the residual, COLUMN refers to the column number, and RESIDUAL is the residual value.

The estimated variance/covariance matrix is saved in a SYSTAT COV file called <file name>.HAT.

This file contains the estimated variance/covariance matrix, saved as a rectangular COV file, with variable names identical to those in the SYS file being analyzed.

INDEX

Algorithms
 Unconstrained minimization 75

Alienation
 Stability of
 Example 23

Arrow
 Defined 5

Arrowhead
 Defined 6

Arrowline
 Defined 6

Bentler–Weeks model 63

Circumplex structure
 Example 108

Common Factor Model
 Confirmatory
 Example 96
 Longitudinal
 Example 133
 Multitrait–Multimethod
 Example 129
 Single
 Constrained 18
 Example 10

Congeneric Tests
 Example 121

COSAN
 Computer program 61
 Model 60

Directed Relationship
 Defined 7

Endogenous Variable
 Defined 8

Equality of dependent variances
 Testing 116

Error recovery
 During model entry 14

ESTIMATE command 16

Exogenous Variable

 Defined 8

EzPATH
 Commands 39
 Model 66

EzPATH Commands
 ACCURACY 40
 AUTOFIX 41
 CMODEL 43
 ESTIMATE 44
 FACTORMODEL 45
 ITERATIONS 47
 METHOD 48
 MODEL 49
 NUMBER 50
 REPORT 51
 RMODEL 52
 SAVE 53
 SE 54
 STANDARDIZE 55

EzPATH diagrams 30
 Rules 32

EzPATH technical output 90

Factor indeterminacy 104

Figures
 1. A Common Factor Model 6
 2. A Single Common Factor Model 10
 3. Constrained Single Factor Model 18
 4. Structural Model A for Stability of Alienation 26
 5. Structural Model B for Stability of Alienation 27
 6. Structural Model for 10 Personality and Drug Use Variables
 29
 7. A Section From an Ambiguous Path Diagram 34
 8. Revised Version of the Diagram in Figure 7 35
 9. Path Diagram for a Confirmatory Factor Model from Everitt
 (1984) 97
 10. Path Diagram for Cooley and Lohnes Model 100
 11. EzPATH Diagram Corresponding to Figure 3A, Jöreskog
 and Sörbom (1982) 101
 12. EzPATH Diagram Corresponding to
 Figure 3B, Jöreskog and Sörbom (1982) 102
 13a. A Model for Two Sets of Congeneric Tests 122

- 13b. A Model for Two Sets of Parallel, Tau–equivalent, Congeneric Tests 122
- 13c. A Model for Two Sets of Parallel Congeneric Tests 123
- 13d. A Model for Two Sets of Tau–equivalent Congeneric Tests 124
- 14. A multitrait–multimethod model 131
- 15. A Longitudinal Factor Analysis Model 134
- 16. A Model for the Effects of Peer Influence on Aspiration 137
- Fixed Value
 - Defined 9
- Free Parameter
 - Defined 9
- Goodness of fit indices 75
 - Adjusted population gamma index 86
 - Akaike information criterion 87
 - Browne–Cudeck (1989) Single Sample Cross–Validation Index 87
 - Confidence intervals 77
 - General theory 76
 - Jöreskog–Sörbom (1984) Adjusted GFI 87
 - Jöreskog–Sörbom (1984) GFI 86
 - Noncentrality–based 77
 - Population gamma index 81
 - Schwarz’s (1978) Bayesian criterion 87
 - Steiger–Lind RMS index 80
- Home environment and mathematics achievement
 - Example 99
- Identification of a model 70
- Installation 4
- Interactive model entry mode 12
- Interrupting execution without losing data 21
- Latent Variable
 - Defined 8
- Limitations
 - Program 143
- LISREL
 - Model 57
- Manifest Variable
 - Defined 7
- Minimization

- Unconstrained 74
- MULTICORR
 - Computer program 107
- Normalized Residuals 94
 - Plotting 142
- OUTPUT command 16
- PAGE command 17
- Parameter Number
 - Defined 9
- PATH1 language
 - Rules 36
- Pattern hypothesis 116
 - Correlations
 - Example 107
- Peer influence on aspiration, effect of
 - Example 136
- Principal components analysis
 - Example 103
- RAM model 60
- References 144
- Restarting execution
 - After an interruption 22
- Revising a model 18
- Stability of a correlation matrix over time
 - Testing 113
- Standardized Residuals 93
- Start Value
 - Defined 10
- Stopping criteria during iteration 74
- Structural Coefficient
 - Defined 8
- System crashes
 - Avoiding 141
- Technical aspects of Ezpath
 - Models and methods 56
- Tutorial 5
- Undirected Relationship
 - Defined 7
- Wire
 - Defined 7