

Using SAS® to Write a LISREL® Program

Francis J. Kelley

University Computing & Networking Services, The University of Georgia, Athens, GA 30602

For many years, LISREL has been a standard for structural equation modeling in many academic fields. While other software is available (EQS® and SAS' PROC CALIS, for example), LISREL has remained quite popular. At The University of Georgia, LISREL is available both as a "standalone" product (from Scientific Software) and as a part of SPSS®¹.

This macro was originally developed when a graduate student asked if there were a way to write a correlation matrix in a form he could use in a LISREL program. Of course, SAS could easily do such a thing, but it seemed "nicer" to have it do this with as little required coding as possible. A Macro would be ideal; it could be used to write the correlation (or covariance) matrix and several other data (or label) vectors as needed. This would serve as a stub around which the rest of the program could be written by any researcher familiar with LISREL. Several problems arose, the program was originally intended to be used on the VM/CMS™ system at UGA via the LISREL interface provided by SPSS. In this version, LISREL itself cannot read an external file, any data to be read must be done by SPSS or be included in the Lisrel code. The data could not be wider than 80 columns and ideally should be less than that. Finally, although LISREL will read a square matrix containing the correlations or covariances, a lower-triangular matrix is more commonly used. The smaller amount of data would make the resulting file look less cluttered and be somewhat easier to read, especially if the data wrapped to subsequent lines. Some other factors intervened, UGA also has a "standalone" LISREL 7 on the MVS/TSO™ system and it seemed reasonable to want the macro to be available there as well. The problem with this idea was the slight variation between the way an MVS LISREL 7 command looked and the way the same command looked in SPSS' LISREL 7 (CMS). LISREL commands given within SPSS are prefaced with a "/". This could be viewed as, essentially, being an attempt to have LISREL commands conform to the general structure of SPSS (sub)commands. Although not required, vectors of means and standard deviations may be passed to the LISREL program; these are included in the output from Proc CORR, and easily made available. The names for the variables in the LISREL program (specified in the LISREL Label line) are taken from the matrix created by SAS. Finally, I

included the ability to omit certain variables that were in the matrix passed.

The SAS2LIS Macro
Syntax:

```
%sas2lis(type=corr,dsn=,format=10.5,list=,
ofile=PRINT,title=,std=yes,version=spss)
```

The SAS2LIS Macro has only one required parameter, the name of the SAS file containing the correlation or covariance matrix. This is specified in the macro call with the DSN= keyword. The file itself is written by Proc CORR when the OUTP= option is specified, and contains the Pearson correlations (see Appendix 3 for an example.) If the COV option is also specified, the file named by OUTP= will additionally contain the covariance matrix. A special variable in the file called _TYPE_ identifies which is being written: N, STD, MEAN, COV and CORR are the number of valid observations, standard deviation, mean, covariances and correlations for each variable. By default, the TYPE= keyword in the macro uses CORR but COV may be specified. Other macro options are: LIST= to specify a list of variables to be omitted, OFILE= to specify the output destination. The default is Print, it writes the result to your output listing and is useful for debugging; specifying a path and file here will cause output to be written to the file named. By default the vectors of means and standard deviations are written in the LISREL stub, but changing the current STD=YES to anything else (even blank) will prevent that. The default, version written is for the SPSS implementation, but changing VERSION=SPSS to anything else (again including blank) will cause the macro to write the program stub in the standard LISREL 7 form. The keyword FORMAT=10.5 attempts to set a general default instead of the usual "BEST" format; do not change this unless you need or want more (less) decimal precision. The TITLE= keyword allows you to pass a title to be used by the LISREL program. This title may not begin with the letters "DA" it will conflict with the LISREL DATA (data parameters) line; LISREL is not case-sensitive.

The SAS2LIS Macro itself (shown in Appendix 1), will for the first 100 or so lines perform error checking and set parameters. For example, when "VERSION=SPSS" the macro variable "prefix" is given the value "/"; if anything else is used here the

"prefix" will have no value. The "LIST" is parsed into the individual variables (via the %SCAN function); the individual values are saved as macro variables. The unwanted variables are both "dropped" (since they are variable names) and "deleted" since they are values of the variable "_NAME_". In a LISREL program, "NI" is the "number of input variables." The count of numeric variables left, after removing the unwanted variables, will give this. In a LISREL program, "NO" is the "number of output variables," in this case it is the largest value on the observation corresponding to SAS' "_TYPE_=N". Specifying type=corr (the default) assigns the LISREL Matrix type of KM. If type=cov, the LISREL matrix type assigned is CM. See Appendix 5 for an example of a LISREL stub program.

The stub written by SAS2LIS is only the start of the program that would be needed. The macro could be enhanced to add other arrays, but that requires greater familiarity with LISREL and the macro would go beyond the original goal of using the data provided in the OUTP dataset (from Proc CORR) as the basis of a LISREL program. Nevertheless, it could be done. Another potential enhancement might be to allow the number of variable values written to a line (currently fixed at 6) to vary depending upon the FORMAT specification. Additionally, Proc CORR could be executed within the macro itself, with only minor modifications, to the existing code. The author is informed (but has not verified) that this code will work with LISREL 8 as well as LISREL 7.

The SAS2LIS macro should work with all versions of SAS software. It has been tested in mainframe and directory-based environments. The code below was run in OS/2, and should run without modifications on Windows™ or UNIX hosts. In an MVS environment one could specify a Dataset name with the OFILE keyword, though it would be best to modify the macro to allow the use of a "fileref" as created by either a "DD" or "Filename" statement. In a CMS environment one would specify files using the usual filename, filetype, and filemode as he/she would with other files on that system.

Footnotes

1. In early 1996 SPSS, Inc. announced that it would no longer be distributing LISREL software.

References

- SAS Institute Inc. (1990), *SAS Language: Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc.
- SAS Institute Inc. (1990), *SAS Procedures Guide, Version 6, Third Edition*, Cary, NC: SAS Institute Inc.
- SAS Institute Inc. (1990), *SAS Guide to Macro Processing, Version 6, Second Edition*, Cary, NC: SAS Institute Inc.
- Karl G. Jöreskog and Dag Sörbom. (1989), *LISREL 7 User's Reference Guide*, Mooresville, IN: Scientific Software.
- Karl G. Jöreskog and Dag Sörbom. (1989), *LISREL 7: A Guide to the Program and Applications*, Chicago, IL: SPSS Inc.
- Karl G. Jöreskog and Dag Sörbom. (1989), *SPSS LISREL 7 and PRELIS: User's Guide and Reference*, Chicago, IL: SPSS Inc.

Acknowledgement

All writers can benefit from the work of a good editor, and I am indebted to Jeannie McElhannon for all her efforts on this.

Author Contact Information

Francis Joseph Kelley
UCNS Client Services
Computer Services Annex
Athens, GA 30602-1911
jkelly@uga.cc.uga.edu
706-542-5359

Trademarks

SAS is a registered trademark of SAS Institute, Inc. in the USA and other countries.
LISREL and PRELIS are registered trademarks of Scientific Software, Inc.
SPSS is a registered trademark of SPSS, Inc. in the USA and other countries.
EQS is a registered trademark of Multivariate Software in the USA and other countries.
MVS, VM, OS/2, CMS and TSO are registered trademarks or trademarks of International Business Machines in the USA and other countries.
Windows is a trademark of Microsoft Corp.
© Indicates USA registration.

Appendix 1:

sas2lis.mac

```
%macro sas2lis(type=corr,dsn=,format=10.5,list=,ofile=PRINT,title=,std=yes,version=spss);
```

* SAS2LIS Macro

* This Macro will take an output Correlation or Covariance matrix (created as an output dataset by PROC CORR) and write it out in a form that may serve as a stub around which a LISREL program may be written. It is assumed the version of Lisrel which will be used is part of the SPSS package, though the VERSION= option allows you to change that.

* Arguments:

* The following KEYWORD parameters may be specified, defaults are indicated by *

Keyword	Meaning
dsn=	_REQUIRED_ this is the output dataset from PROC CORR. Several correlation datasets may be created, specifying OUTP=dsn will write the Pearson correlations to the SAS file "dsn". If the COV Option is specified, the file named in OUTP= will also contain the variance-covariance matrix. A special variable named "_TYPE_" is created in these datasets..., correlation matrices have a "_TYPE_=CORR" and covariance matrices have a "_TYPE_=COV".
type=	either cov or corr (*) with corr as the default.
ofile=	The output file...the default is print (*) any other value will cause the output to be written to the file "ofile". PRINT files will display in the LISTING file.
list=	The variables to be DROPPed/DELETED from the output. The variables in this list should be separated by spaces, they will not be passed to the Lisrel stub. By default _all_ variables are used.
title=	The TITLE is specified on the first line of the LISREL program. Titles in LISREL may _not_ begin with the letters "DA" because there is a conflict with the "DA line". The macro checks for this and will, with a warning, terminate if found.
format=	The numeric format used to print the data. The default is 10.5 (*)...do not change this unless you need decimal prescision.
std=	Default is YES (*). This writes the Mean (ME) and Standard Deviation (SD) vectors in the stub. Anything else (or nothing) specified will omit these.
version=	The type of LISREL to be coded for. The default is spss (*), anything else (or nothing) will cause the Macro to write a non-SPSS-dependant stub. Use the default if you use Lisrel delivered by Scientific Software, Inc. (ie "standalone").
	10/18/94 F.J. Kelley UCNS/UGA
	ported to OS/2 5/20/96 - F.J. Kelley

```
%if &dsn = %str() %then %do;
```

```
  %put *****;
  %put
  %put ERROR
  %put You _must_ name the SAS file which contains the values written by PROC CORR.
  %put *****;
  %goto getout;
```

```
%end;
```

```
%let type = %upcase(&type) ;
```

```
%if &type = %str(CORR) %then %let matype=KM;
```

```
%else %if &type = %str(COV) %then %let matype =CM ;
```

```
%else %do;
```

```

%put *****
%put
%put ERROR
%put The "TYPE" specified must be either CORR or COV. The Default is CORR. You must specify
%put this so SAS writes the correct matrix, and writes the correct LISREL specification for that
%put matrix - LISREL identifies Correlation matrices as "KM" and Covariance matrices as "CM".
%put The TYPE you specified was: &type. Please correct this and resubmit the program
%put *****
%goto getout ;
%end;
%if &ofile = %str() %then %do;
%put *****
%put
%put ERROR
%put If you use the OFILE= option in the macro call, you must use a valid Filename.
%put The default output file is PRINT which directs output to your Listing file
%put *****
%goto getout;
%end;
%else %if %upcase(&ofile) ^= %str(PRINT) %then
%let ofile = "&ofile" ;
%let list = %upcase(&list);
%let subscr = 0 ;
%do %until(&st = );
%let subscr = %eval(&subscr + 1) ;
%let st = %scan(&list,&subscr);
%let nam&subscr = "&st";
%end;
%let subscr = %eval(&subscr - 1) ;
%let std = %upcase(&std);
%let version = %upcase(&version) ;
%if &version = %str(SPSS) %then %let prefix = %str( /);
%else %do ;
%let prefix = %str() ;
%let version = %str(Scientific Software);
%end;

%put *-----*
%put Parameters Used:
%put dsn:      &dsn
%put type:     &type
%put list:     &list
%put ofile:    &ofile
%put title:    &title
%put format:   &format
%put std:      &std
%put version:  &version
%put *-----*

```

```

%if %str(&title) ^= %str() %then %do ;
  %if %upcase(%substr(&title,1,2)) = %str(DA) %then %do;
    %put *****;
    %put NOTE:
    %put This is not a SAS problem. The title you have chosen for the LISREL program -
    %put "title=" &title begins with the letters "DA". Your LISREL program will interpret this
    %put as a "DA line" and will not execute properly. Please change your title.
    %put *****;
    %goto getout ;
  %end;
%end;

data _setup_ ; set &dsn ;

%if &subscr > 0 %then %do;
  drop &list ;
  if _NAME_ in (
%do i = 1 %to &subscr ;
  &&nam&i
%end;
  ) then delete ;
%end;

data _setup_ _descr_ ; set _setup_ ;
array stuff(*) _NUMERIC_ ;
if _TYPE_ = 'N' then do;
  maxn = 0 ;
  d = dim(stuff) ;
  do i=1 to d ;
    if stuff(i) > maxn then maxn = stuff(i) ;
  end;
  call symput('_NO_',trim(left(put(maxn,6.))));
  call symput('_NI_',trim(left(put(d,5.))));
end;
else if _TYPE_ = "&type" then output _setup_ ;
else if _TYPE_ in ("STD" "MEAN") then output _descr_ ;

data _NULL_ ;
set _setup_ _descr_ end=last ;
file &ofile ;
retain names1-names&_NI_ ;
retain incr 2 ;
array stuff(*) _NUMERIC_ ;
array names(&_NI_) $8 ;
format default=&format ;
if (_N_ = 1) then do;
%if &version = %str(SPSS) %then %do;
  put @1 "LISREL " ;
%end;
  put @1 "&prefix.&title " /
    @1 "&prefix.DA NI=&_NI_ NO=&_NO_ MA=&matype " /
    @1 "&prefix.&matype SY " /
    @1 "&prefix.* " ;
end;
if (_N_ <= &_NI_) then do ;
  names(_N_) = _NAME_ ;

```

```

do j = 1 to _N_ ;
  if mod(j,6) = 1 then do ;
    if j = 1 then put @1 "&prefix" @ ;
    else put / "&prefix" @ ;
  end;
  put stuff(j) +incr @;
end;
put ;
end;
%if ( &std = %str(YES) ) %then %do ;
  else if ( _N_ > &_NI_ ) then do;
    if _TYPE_ = 'STD' then put @1 "&prefix.SD" /
      "&prefix.*";
    else if _TYPE_ = 'MEAN' then put @1 "&prefix.ME" /
      "&prefix.*";
    do j = 1 to &_NI_ ;
      if mod(j,6) = 1 then do ;
        if j = 1 then put @1 "&prefix" @ ;
        else put / "&prefix" @ ;
      end;
      put stuff(j) +incr @;
    end;
    put ;
  end;
%end;
if last then do;
  incr = 4 ;
  put @1 "&prefix.LA " ;
  do i = 1 to &_NI_ ;
    if mod(i,6) = 1 then do ;
      if i = 1 then put @1 "&prefix " @ ;
      else put / "&prefix " @ ;
    end;
    put names(i) $8. + incr @ ;
  end;
  put ;
end;

proc datasets library=WORK nolist ;
  delete _setup_ _descr_ ;
%getout: run;
%mend ;

```

Appendix 2:

sas2lis.sas

An example program that generates dummy data and runs SAS2LIS

```

data dummy ;
  array vars {*} v1 z5 test v2 z scor1 scor3 test8;
  retain seed 123456789 ;
  do i = 1 to 100 ;
    do j = 1 to dim(vars) ;
      vars[j] = ranuni(seed) * 100 ;
    end ;
    output ;
  end ;

```

```
proc corr data=dummy out=cordat noprint ;
run;
```

```
proc print data=cordat ;
title 'Dataset: CORDAT' ;
run;
```

```
%sas2lis(dsn=cordat,list=i j seed,title=A Test LISREL Pgm, version=,ofile=f:\sesug96\test1.lsr)
```

```
run;
```

Appendix 3:

sas2lis.lst

This is the output file produced by PROC CORR.

Dataset: CORDAT

OBS	_TYPE_	_NAME_	V1	Z5	TEST	V2	Z
1	MEAN		52.904	46.975	48.277	51.386	49.716
2	STD		29.236	30.073	28.650	29.203	30.956
3	N		100.000	100.000	100.000	100.000	100.000
4	CORR	V1	1.000	0.084	-0.044	-0.044	-0.055
5	CORR	Z5	0.084	1.000	0.019	0.048	-0.077
6	CORR	TEST	-0.044	0.019	1.000	0.021	-0.074
7	CORR	V2	-0.044	0.048	0.021	1.000	0.195
8	CORR	Z	-0.055	-0.077	-0.074	0.195	1.000
9	CORR	SCOR1	-0.081	0.009	0.038	-0.001	0.022
10	CORR	SCOR3	0.042	0.067	0.074	-0.042	0.004
11	CORR	TEST8	-0.120	-0.020	-0.090	0.016	0.182
12	CORR	SEED					
13	CORR	I	-0.325	-0.185	-0.064	0.027	-0.013
14	CORR	J					

OBS	SCOR1	SCOR3	TEST8	SEED	I	J
1	51.324	50.590	52.512	123456789	50.500	9
2	29.063	27.223	27.868	0	29.011	0
3	100.000	100.000	100.000	100	100.000	100
4	-0.081	0.042	-0.120	.	-0.325	.
5	0.009	0.067	-0.020	.	-0.185	.
6	0.038	0.074	-0.090	.	-0.064	.
7	-0.001	-0.042	0.016	.	0.027	.
8	0.022	0.004	0.182	.	-0.013	.
9	1.000	-0.158	-0.067	.	0.010	.
10	-0.158	1.000	0.008	.	0.048	.
11	-0.067	0.008	1.000	.	0.036	.
12	.	.	.	.	.	.
13	0.010	0.048	0.036	.	1.000	.
14	.	.	.	.	.	.

Appendix 4:

sas2lis.log (partial)

This is the "diagnostic" written by the SAS2LIS macro:

```

*-----*
Parameters Used:
dsn:          cordat
type:         CORR
list:         I J SEED
ofile:        "f:\sesug96\test1.lsr"
title:        A Test LISREL Pgm
format:       10.5
std:          YES
version:      Scientific Software
*-----*

```

Appendix 5:

test1.lsr

Finally, this is the LISREL stub program written by the SAS2LIS Macro:

A Test LISREL Pgm

DA NI=8 NO=100 MA=KM

KM SY

```

*
  1.00000
  0.08355    1.00000
 -0.04396    0.01907    1.00000
 -0.04443    0.04756    0.02144    1.00000
 -0.05467   -0.07705   -0.07413    0.19527    1.00000
 -0.08093    0.00850    0.03824   -0.00060    0.02240    1.00000
  0.04244    0.06745    0.07355   -0.04183    0.00433   -0.15829
  1.00000
 -0.11994   -0.01966   -0.08965    0.01581    0.18170   -0.06664
  0.00754    1.00000
ME
*
  52.90363   46.97500   48.27671   51.38595   49.71635   51.32448
  50.58984   52.51231
SD
*
  29.23578   30.07276   28.64952   29.20280   30.95594   29.06264
  27.22309   27.86834
LA
V1          Z5          TEST          V2          Z          SCOR1
SCOR3      TEST8

```