Introduction to Mx Scripts

Michael C Neale
Virginia Institute for Psychiatric and Behavioral Genetics
Virginia Commonwealth University

17th International Workshop on Methodology of Twin and Family Studies

General Comments

Mx Script Language

- Case insensitive, except for filenames under Unix
- Comments: anything following a !
- Blank lines ok
- Indenting ok
- Commands: usually identified by first 2 letters, BUT recommended to use full words

Job Structure

- Three types of groups:
  - Data, Calculation, Constraint
- Number of groups indicated by
  - #NGroups 3
  - at the beginning of job
- Jobs can be stacked in one run

Group Structure

- Title
  - Group type: data, calculation,constraint
  - [Read observed data, Select, Labels]
  - Matrices declaration
  - [Specify numbers, parameters, etc.]
  - Algebra section and/or Model statement
  - [Options]
- End

Read Observed Data

- Data NObservations=123 NInputvars=2
- CMatrix/Means/ACov/CTable
  - Rectangular/ VLength/ Ordinal
    - File= filename
  - ! i.e. reads either Summary statistics or raw data
- Labels *varlist*
- Select variables ; [by number/label]
- Select If var=value

Typically Data & Labels appear in .dat file
.dat file may read data from file, e.g., Rectangular file=mydata.rec

Format of data files

- CMatrix/PMatrix/ACov/Ainv
  - Has fortran-style format as first line
  - Usually just * for free-format read
  - May read either lower triangular matrix (default) or full matrix
- Rectangular/Ordinal
  - No format
  - 1 case per line
  - Different variables in different columns
  - . is used for missing values by default

Matrices Declaration

- Begin Matrices; [= Group 1*]
  - C Full 2 3 Free = A1
    - (name type rows colums [free])
    - default element is fixed at 0
  - ... more matrices
- End Matrices;

- * copies all matrices from group 1

Matrix Types

see Mx Manual page 57

TYPE	STRUCTURE	SHAPE	# of FREE ELEMENTS
Zero	null (zeros)	any	0
Unit	unit (ones)	any	0
Iden	identity	square	0
Diag	diagonal	square	r
SDiag	subdiagonal	square	r(r-1)/2
Stand	standardized	square	r(r-1)/2
Symm	symmetric	square	r(r+1)/2
Lower	lower triangular	square	r(r+1)/2
Full	full	any	rxc
Computed	equated to	any	0

Constrained Matrices

to special quantities in previous groups

SYNTAX	MATRIX QUANTITY	DIMENSIONS
%On	Observed covariance matrix	NI x NI
%En	Expected covariance matrix	NI x NI
%Mn	Expected mean vector	1 x NI
%Pn	Expected proportions	NR x NC
%Fn	Function Value	1 x 1

n refers to group number n

Matrix Algebra/ Model

- Begin Algebra;
  - B = A*A';
  - C = B+B;
  - ...
- End Algebra;

- Covariances A*A';
- Means [raw data]
- Thresholds [categorical variables]
- Weight/ Frequency

Matrix Operations

see Mx Manual page 60

SYMBOL	NAME	FUNCTION	EXAMPLE	PRIORITY
~	Inverse	inversion	A~	1
'	Transpose	transposition	A'	1
^	Power	element powering	A^B	2
*	Star	multiplication	A*B	3
.	Dot	dot product	A.B	3
@	Kronecker	Kronecker product	A@B	3
&	Quadratic	quadratic product	A&B	3
%	Eldiv	element division	A%B	3
'+'	Plus	addition	A+B	4
'-'	Minus	subtraction	A-B	4
|	Bar	horizontal adhesion	A|B	4
_	Under	vertical adhesion	A_B	4

Matrix Functions

see Mx Manual page 65 for a full list

KEYWORD	FUNCTION	RESTRICTIONS	DIMENSIONS
\tr()	trace	r=c	1 x 1
\det()	determinant	r=c	1 x 1
\sum()	sum	none	1 x 1
\prod()	product	none	1 x 1
\max()	maximum	none	1 x 1
\min()	minimum	none	1 x 1
\abs()	absolute value	none	r x c
\exp()	exponent	none	r x c
\ln()	natural logarithm	none	r x c
\sqrt()	square root	none	r x c
\stnd()	standardize	r=c	r x c
\mean()	mean of columns	none	1 x c
\cov()	covariance of col	none	c x c
\pdfnor()	mv normal density	r=c+2	1 x 1
\mnor()	mv normal integral	r=c+3	1 x 1

Set Matrix Elements to Particular Values or Parameters

- Values
  - Matrix <name> <number list>
  - Start/Value <name> <value> <element list>
- Parameters
  - Fix/Free <value> <element list>
  - Equate <name> GRC name GRC
  - Specify <name> <integer list>
  - Bound low high <parameter list/element list>
- Label Matrices
  - Label Row/Column <name> <label list>

Options

- Statistical Output
  - Suppressing output: No_Output
  - Appearance: NDecimals=n
  - Residuals: RSiduals
  - Adjusting Degrees of Freedom: DFreedom=n
  - Power Calculations: Power=alpha,df
  - Confidence Intervals:
    - Interval {@value} <matrix element list>

Options

- Optimization options
  - Randomizing Starting Values: THard=n
  - Automatic Restart: THard=-n
  - Jiggling Parameter Starting Values: Jiggle
  - Confidence Intervals on Fit Statistics
  - Comparative Fit Indices: Null
  - Likelihood-Ratio Statistics of Submodels: Issat/ Sat
  - Check Identification of Model: Check
  - Optimization Parameters

Fitting Submodels

- Multiple Fit
  - Option Multiple: Matrix/ Value/ Start/ Equate/ Fix/ Free/ Commands
  - Drop {@value} <parlist> <element list>
  - Binary Save/Get <filename>
  - Writing Matrices to Files: MXn = <filename>
  - Writing Individual Likelihood Statistics to Files: MX%P = <filename>

Additional Options for Multiple Fit

- Issat
  - sets current model to be the saturated one
- Issub
  - sets current model to be a submodel
- Option sat=fit,df or sub=fit,df
- Automatically computes fit statistics differences between models

Conclusion

- Mx language flexible
- Can be used as matrix algebra calculator
- Can be used to fit models with 'standard' methods
- Can be used to specify other models via
- user-defined fit functions

Mx script to estimate covariances & means

```
#ngroups 1                                  !        mydatfile.dat e.g.
                                            !         Data NI=2
#define nvar=2                              !         Labels SBP-T1 SBP-T2
                                            !         Rectangular File=ukbp.rec
G1: Estimate means & Covariances by ML
 #include mydatfile.dat

 Begin Matrices;                            !        ukmzbp.rec e.g.
  C Symm nvar nvar  Free                    !          120.5 142.3
  M Full    1 nvar  Free                    !          102.6 110.7
 End Matrices;                              !           98.3 116.9
  Start 120 M 1 1 to M 1 nvar
  Matrix C 10 0 10                            !       starting values for C
  Covariance C;
  Means M;

 Option NDecimals=2
End
```

Mx script to estimate covariances & means

Cholesky Decomposition

```
#ngroups 1                                  !        mydatfile.dat e.g.
                                            !         Data NI=2
#define nvar=2                              !         Labels SBP-T1 SBP-T2
                                            !         Rectangular File=ukbp.rec
G1: Estimate means & Covariances by ML
 #include mydatfile.dat

 Begin Matrices;                            !        ukmzbp.rec e.g.
  L lower nvar nvar Free                    !          120.5 142.3
  M Full    1 nvar  Free                    !          102.6 110.7
 End Matrices;                              !           98.3 116.9

  Start 120 M 1 1 to M 1 nvar
  Start .5 L 1 1 to L nvar nvar             !        starting values for L
  Covariance L*L';
  Means M;

 End
```