Appendix C

The MDLV MATLAB toolbox

The data formats for fitting LCM, LMM, MLCM, and MLMM with the MDLV MATLAB toolbox are described in Appendix C1. The inputs and outputs of the main estimation functions for each model are annotated in Appendix C2. In addition, scripts for simulating data according to each of the four types of models are provided in Appendix C3. Appendix C3 also gives detailed descriptions of several utility functions used in the toolbox. All of these functions are organized in the “MDLV_MATLAB” folder and are available by request.

Appendix C1: Data Format

- *LCM*.
  The data required for LCM consist of a rectangular matrix (S) with rows being unique response patterns in the data and a vector of frequencies for the response patterns (yv). For data matrices with rows corresponding to response vector of a single subject, the utility function ‘aggregate.m’ can be used to form the matrix of unique response patterns and the vector of corresponding frequencies.

- *LMM*.
  The data set for LMM is a three-dimensional matrix: individuals by items by time points ($N \times J \times T$). The ‘2Dto3D.m’ function can be used to transform a rectangular matrix of individuals by items and time points to a three-dimensional data matrix needed when fitting LMM.

- *MLCM*.
  The data set for MLCM is a three-dimensional matrix of ($G \times Ng \times J$), where G is the number of groups, Ng is the number of individuals in group g , and J is the number of items.

- *MLMM*.
  The data set for MLMM is a four-dimensional matrix of ($G \times Ng \times J \times T$), where G is the number of groups, Ng is the number of individuals in group g , J denotes the number of the items, and T is the number of occasions.

Appendix C2: Inputs and outputs of main estimation functions

The main estimation functions for each of the four models for discrete latent variables are LCM.m, LMM.m, MLCM.m, and MLMM.m. The input and output variables for each model are listed below with corresponding descriptions.

LCM.m

```
function [lk,it,n,J,pi1,Rho,np,AIC,BIC] = LCM(S,yv,M,start);
```

Input:

Parameters		Dimension
`S`	Matrix of unique response patterns for all items	#unique vector X J
`Yv`	Vector of corresponding frequencies of the unique response patterns in matrix *S*	#unique vector X 1
`M`	Number of latent classes	Scalar
`Start`	Specify the type of starting values: 0 = deterministic (estimated from observed data), 1 = random, and 2 = load user specified values from the file 'start.mat'.	Depending on number of classes and number of items

Output:

Parameters		Dimension
`lk`	Log-likelihood value	Scalar
`it`	Number of iterations	Scalar
`n`	Number of subjects	Scalar
`J`	Number of items	Scalar
`pi1`	Estimated latent class probabilities	$M \times 1$
`Rho`	Estimated conditional response probability	$M \times J$
`np`	Number of parameters	Scalar
`AIC`	Akaike information criterion	Scalar
`BIC`	Bayesian information criterion	Scalar

LMM.m

```
function [lk,iteration,n,T,J,M,pi1,TranMatrix,rho,np,AIC,BIC] =
LMM(X,pi1,tau,rho,n,T,J,M)
```

Input:

Parameters		Dimension
`X`	Data Matrix	$N \times J \times T$
`pi1`	Starting latent class probabilities	$M \times 1$
`tau`	Starting latent transition matrix	$M \times M$
`rho`	Starting conditional response probabilities	$M \times J$
`n`	Number of subjects	Scalar
`T`	Number of occasions	Scalar
`J`	Number of items	Scalar
`M`	Number of latent classes	Scalar

Output:

Parameters		Dimension
`lk`	Log-likelihood value	Scalar
`iteration`	Number of iterations	Scalar
`n`	Number of subjects	Scalar
`T`	Number of time points	Scalar
`J`	Number of items	Scalar
`M`	Number of latent classes	Scalar
`pi1`	Estimated latent classes probabilities	$M \times 1$
`TranMatrix`	Estimated latent transition matrix	$M \times M$
`rho`	Estimated conditional response probabilities	$M \times J$
`np`	Number of parameters	Scalar
`AIC`	Akaike information criterion	Scalar
`BIC`	Bayesian information criterion	Scalar

MLCM.m

```
function
[lk,iteration,G,n_g,J,L,M,est_PH_g,est_PXgivenH,est_CondResProd,Est_h,n
p,AIC,BIC]=...
    MLCM(Data3D,G,n_g,L,M,J,est_PH_g,est_PXgivenH,est_CondResProd)
```

Input:

Parameters		Dimension
`Data3D`	Data Matrix	$G \times N_g \times J$
`G`	Number of groups	Scalar
`n_g`	Number of subjects per group	Scalar
`L`	Numbers of latent clusters	Scalar
`M`	Number of latent classes	Scalar
`J`	Number of items	Scalar
`est_PH_g`	Starting latent class probabilities	$M \times 1$
`est_PXgivenH`	Starting latent transition matrix	$M \times M$
`est_CondResProd`	Starting conditional response probabilities	$M \times J$

Output:

Parameters		Dimension
`lk`	Log-likelihood value	Scalar
`iteration`	Number of iterations	Scalar
`G`	Number of groups	Scalar
`n_g`	Number of subjects per group	Scalar
`J`	Number of items	Scalar
`L`	Number of latent clusters	Scalar
`M`	Number of latent classes	Scalar
`est_PH_g`	Estimated latent cluster probabilities	$L \times 1$
`est_PXgivenH`	Estimated conditional latent class probabilities	$M \times L$
`est_CondResProd`	Estimated conditional response probabilities	$M \times J$
`Est_h`	Estimated latent cluster membership	$G \times 1$
`np`	Number of parameters	Scalar

AIC	Akaike information criterion	Scalar
BIC	Bayesian information criterion	Scalar

MLMM.m

```
function
[lk,iteration,G,Ng,J,T,L,M,PH_g,PXgivenH,TauGivenH,CondResProd,Est_h,np
,AIC,BIC]=...
    MLMM(Data4D,Data2D,G,Ng,L,M,J,T,PH_g,PXgivenH,TauGivenH,CondResProd)
```

Input:

Parameters		Dimension
Data4D	Data Matrix	$G \times Ng \times J \times T$
Data2D	Data Matrix	$(G \times Ng \times T) \times J$
G	Number of groups	Scalar
Ng	Number of subjects per group	Scalar
L	Number of latent clusters	Scalar
M	Number of latent classes	Scalar
J	Number of items	Scalar
T	Number of time points	Scalar
PH_g	Starting latent class probabilities	$M \times 1$
PXgivenH	Starting latent transition matrix	$M \times M \times L$
CondResProd	Starting conditional response probabilities	$M \times J$

Output:

Parameters		Dimension
lk	Log-likelihood value	Scalar
iteration	Number of iteration	Scalar
G	Number of groups	Scalar
n_g	Number of subjects per group	Scalar
J	Number of items	Scalar
T	Number of time points	Scalar
L	Number of latent clusters	Scalar
M	Number of latent classes	Scalar
PH_g	Estimated latent cluster probabilities	$L \times 1$
PXgivenH	Estimated conditional latent class probabilities	$M \times L$
TauGivenH	Estimated conditional latent transition matrix	L of $M \times M$
CondResProd	Estimated conditional response probabilities	$M \times J$
Est_h	Estimated latent cluster membership	$G \times 1$
np	Number of parameters	Scalar
AIC	Akaike information criterion	Scalar
BIC	Bayesian information criterion	Scalar

Appendix C3: Data Simulation Scripts and Utility Functions

The MDLV MATLAB toolbox provides functions for simulating data from the specified model using given parameters and supporting utility functions.

Data Simulation

- Simulate_LCM.m: A script that can be used to simulate data based on a LCM with the specified number of latent classes, latent class probabilities, and conditional response probabilities. The output is an individuals by items ($N \times J$) data matrix.
- Simulate_LMM.m: A script that can be used to simulate data based on a LMM. The function requires: (1) the number of latent classes, (2) the number of time points, (3) a vector of latent class probabilities, (4) a transition matrix, and (5) the conditional response probabilities. The output is a ($N \times J \times T$) data matrix.
- Simulate_MLCM.m: A script that can be used to simulate data based on a MLCM. The number of individuals per group is fixed to be the same (N_g).

  Groups are first assigned to one of the latent clusters, and individuals of each group are then assigned to latent classes based on the latent class probabilities given their group's latent cluster membership. The responses of each individual are simulated based on his/her latent class membership. This function generates a three-dimensional data matrix of $G \times N_g \times J$.
- Simulate_MLMM.m: A script that can be used to simulate data based on a MLMM. In addition to specifications as required in 'Simulate_MLCM.m', this function also needs the conditional transition matrix for each latent cluster. The latent class memberships at second and subsequent time points are generated from the transitional probabilities. All other steps for simulating the data follow the same procedure as explained in 'Simulate_MLCM.m'. The output of this function is a four-dimensional ($G \times N_g \times J \times T$) data matrix.

Utility Functions

- aggregate.m: Transforms an individual by item matrix into a matrix containing unique response patterns in the rows, and a vector of corresponding frequencies for the unique patterns.
- rsum.m: Sum over each row of a vector or matrix.
- csum.m: Sum over each column of a matrix.
- diagv.m: Computes $N = \text{diag}(v) \times M$ that avoids the *diag* operator.
- Sample_discrete.m: Sample from a non-uniform discrete distribution. Syntax of sample discrete ([0.3 0.7], 1, 5) generates a row vector of 5 random integers from {1,2}, where the probability of being 1 is 0.3 and the probability of being 2 is 0.7.
- StartingValue_RandomV1.m: Generate random starting values for model parameters in a MLCM and a MLMM.