Appendix to the paper entitled

"A multidimensional finite mixture SEM for non-ignorable missing responses to test items"

Silvia Bacci[1] and Francesco Bartolucci[2]

Department of Economics

University of Perugia, Via A. Pascoli 20, 06123 Perugia (IT)

Here we provide the R syntax of the proposed class of mixture SEM models. The main function we use is named est_multi_poly3 and it is based on the package MultiLCIRT; for a detailed description of this package see Bartolucci et al. (2014). The package MultiLCIRT must be installed from CRAN if it has not been previously installed. The following illustration is based on one of the datasets used in the simulation study described in the paper.

We first load the package and the other useful functions and we load the data, here called data_www.RData:

```
require(MultiLCIRT)
source("est_multi_poly3.R")
load("data_www.RData")
S[S>1] = NA
SR = cbind(S,!is.na(S))
m = 20 # number of items
```

Note that above commands also create a new matrix SR, which merges matrix S with a new one having size $n \times m$, whose cells contain value 1 if the corresponding item response is observed and value 0 in case of missing value.

[1]email: silvia.bacci@stat.unipg.it; phone: (+39)075.585.52.26.

[2]email: bart@stat.unipg.it; phone: (+39)075.585.52.27.

Then, we need to define the multidimensional structure of data. For this aim we use a type of matrix with a number of rows equal to the number of dimensions and elements in each row equal to the indices of the items measuring the dimension corresponding to that row. In particular we have:

```
multi = cbind(matrix(1:m,2,m/2,byrow=TRUE),matrix(1:m,2,m/2,byrow=TRUE)+m)
multi2 = (m+1):(2*m)
```

where matrix `multi` defines two latent abilities U_d $(d = 1, 2)$, each composed by 10 items and influencing both the item responses and the presence of missing values. Matrix `multi2` refers to the presence of an additional latent trait for the missing responses. It is here defined so as to assume only one latent trait for the tendency to answer, but we outline that, defining matrix `multi2` in a suitable way, we can also model more complicated structures (e.g., a different latent trait V_d for each ability U_d, $d = 1, \dots, s$).

The estimation of the models is performed through function `est_multi_poly3`, which requires the following main inputs: matrix `SR`, number of latent classes (`k1` and `k2`), matrix of covariates `X`, type of link function (`link`=1 or 2 in case of binary data), information on the discriminant indices (`disc`=0 for a Rasch-type parametrization and `disc`=1 for a 2PL model), and matrices describing the multidimensional structure of abilities U_d (`multi`) and that of the tendency to answer V (`multi2`). Useful options are `tol` to specify the tolerance level of the estimation algorithm, `fort` to speed up the estimation process, and `disp` to display the log-likelihood evolution step-by-step. The bidimensional LC Rasch and 2PL models are estimated as follows

```
out_rasch = est_multi_poly3(SR,k1=3,k2=3,X=X,link=1,disc=0,multi=multi,multi2=multi2,
> tol=10^-9,fort=F, disp=T)   #Rasch
out_2PL = est_multi_poly3(SR,k1=3,k2=3,X=X,link=1,disc=1,multi=multi,multi2=multi2,
> tol=10^-9,fort=F, disp=T)   #2PL
```

Values for the model selection (i.e., estimated maximum log-likelihood, number of free parameters, and BIC) are then obtained in the usual way, that is, by the commands `out_2PL$lk`, `out_2PL$np`, and `out_2PL$bic` for the 2PL model, where `out_2PL` is substituted by `out_rasch` in case of the Rasch model. Finally, the remaining outputs of interest (similarly to those shown in Tables 5, 7, and 8) are obtained in a similar way. These outputs are `Th` and `th2`, which refer to the support points of latent traits U_d $(d = 1, 2)$ and V, respectively, and `piv1` and `piv2` are the corresponding average weights. Moreover, `De1` and `De2` are the regression coefficients $\hat{\phi}_{jh_1}$ and $\hat{\psi}_{jh_2}$, respectively, whereas the item parameters are contained in `Bec` (difficulty parameters $\hat{\beta}_j$ and $\hat{\delta}_j$), `gac`, and `gac2` (discriminant parameters $\hat{\alpha}_j$, $\hat{\gamma}_{1j}$, and $\hat{\gamma}_{2j}$).

All the above commands are reported in the file `example_www.R` with the addiction of certain commands for standardizing the support points for the latent distributions and for the non-parametric bootstrap method used to compute the standard errors for the parameter estimates.

To conclude, the proposed class of models may be also estimated under the MAR assumption. In this case, the estimation process is performed through function `est_multi_poly` included in the `MultiLCIRT` package, as follows:

```
out_2PL_mar = est_multi_poly(S, k=3, X=X, link=1, multi=matrix(1:m,2,m/2,byrow=TRUE), disc=1)
```

This function has a behavior similar to that of `est_multi_poly3`, being `S` the item response matrix, `k` the number of latent classes, and having inputs `link`, `disc`, and `multi` the same meaning as in `est_multi_poly3`. Note that matrix `multi` differs from the MNAR case, because of the absence of missing indicators. Finally, outputs are contained in matrix `Th` (support points) and in vectors `piv` (latent class weights), `De` (regression coefficients), `Bec` (difficulty parameters), and `gac` (discriminant parameters).

References

Bartolucci, F., Bacci, S. & Gnaldi, M. (2014). MultiLCIRT: An R package for multidimensional latent class item response models, *Computational Statistics and Data Analysis*, *71*, 971–985.