
Online Appendix to accompany S.K. Sterba and J. Pek
Individual Influence on Model Selection. *Psychological Methods*.

Part I. *Mplus* 6.1 syntax to export casewise loglikelihoods.....p. 2-4

Part II. SAS 9.2 syntax that uses casewise loglikelihoods to calculate three model
selection influence diagnostics: $\Delta\text{ind}_{\text{CHI}}$, $\Delta\text{ind}_{\text{BIC}}$, $\Delta\text{ind}_{\text{AIC}}$p. 5-8

Part I. Mplus 6.1 syntax to export casewise loglikelihoods

In *Mplus* 6.1 (Muthén & Muthén, 1998-2011), input syntax needs to have a few key additions in order to export casewise likelihood values for a given fitted model. These additions are mentioned below, under their respective section of the *Mplus* input code. Also, please note that a few special instructions are mentioned in blue for researchers using multilevel models.

Input syntax additions:

DATA: No modifications needed in this section; read in data as per usual.

VARIABLE: One helpful (but optional) modification for this section is to include the syntax:

```
IDVARIABLE IS id;
```

Here, *id* is the variable name corresponding with the case identifier in your dataset. In your dataset the case identifier may be called something different, such as *subject*, and if so this name should be mentioned instead. Recall that a case is the highest-level unit in an analysis. The command `IDVARIABLE IS` ensures that the case identifier is listed in the exported dataset alongside the casewise loglikelihood values, as both will be needed for constructing index plots of case identifier values against influence diagnostic values. Note that researchers fitting multilevel models in *Mplus* (using the `TYPE=TWOLEVEL` analysis command) should not include this `IDVARIABLE` statement. Their case identifier will be the variable used on the '`CLUSTER =` ' statement and this variable will automatically be listed in the exported dataset.

ANALYSIS: Sometimes full information maximum likelihood is the default estimator, but to be on the safe side here we request it explicitly with the following addition to the analysis statement.

```
ESTIMATOR IS ML;
```

MODEL: No modifications to this section are needed.

SAVEDATA: This section is required in order to export casewise loglikelihood values into an ASCII file. The name for the exported file is your choice (here called *modela.dat*) and it will be saved in the same location as your input file. The `SAVE` command mentions `LOGLIKELIHOOD` in order to request that the casewise loglikelihoods for the hypothesized (fitted) model be included in the exported file.

```
FILE IS modela.dat;  
SAVE= LOGLIKELIHOOD;
```

In sum, here is an example of a one factor CFA input file that includes in red the aforementioned necessary additions in order to export casewise loglikelihoods.

```
DATA: FILE IS mydata.dat;  
VARIABLE: NAMES ARE id y1-y6;  
USEVARIABLES ARE y1-y6;  
IDVARIABLE IS id;  
ANALYSIS: ESTIMATOR IS ML;  
MODEL: f1 BY y1-y6*; y1-y6*; f1@1;  
SAVEDATA:  
FILE IS modela.dat;  
SAVE= LOGLIKELIHOOD;
```

Relevant output:

At the very end of the regular Mplus output file, Mplus will list the order of variables in your exported dataset. Each name in the list refers to a column in the exported dataset, reading from left to right. It is crucial to inspect this list since the exported dataset will not have variable names included at the tops of the columns. Often, this list will begin with all the variables used in the analysis, followed by your subject identifier (as requested on the IDVARIABLE input statement, or for [multilevel modeling applications, as indicated on the CLUSTER statement](#)), followed by the casewise loglikelihood values for the fitted model. For example, for the running appendix example--the one factor CFA model mentioned above--the list would look like:

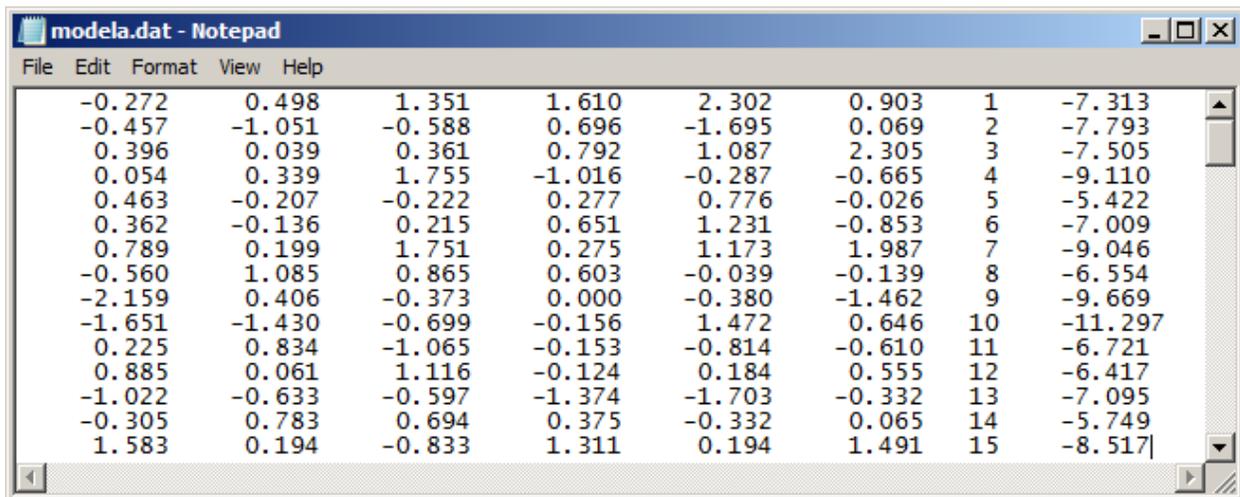
```
SAVEDATA INFORMATION

Order and format of variables
Y1          F10.3
Y2          F10.3
Y3          F10.3
Y4          F10.3
Y5          F10.3
Y6          F10.3
ID          I7
OUTLOGL     F10.3

Save file
modela.dat
```

where Y1-Y6 are all the variables used in this analysis, and ID was the requested case identifier and OUTLOGL refers to the casewise loglikelihood values for the fitted model. Mplus reminds us of the name we called our exported datafile at the end of the output (here, *modela.dat*).

An excerpt of the corresponding ASCII file *modela.dat* for the running appendix example would look like:



A researcher would want to output casewise loglikelihood values for both competing models fitted in Mplus. For instance, a researcher might call the file with casewise loglikelihood values for Model A (e.g., a one factor CFA), *modela.dat* and might call the file with casewise loglikelihood values for Model B (e.g., a two factor CFA), *modelb.dat*. Once ASCII files containing casewise loglikelihood values for both fitted models have been obtained, the researcher can move on to run the SAS code in the next section to calculate and plot model selection case influence diagnostics.

Part II. SAS 9.2 syntax that uses casewise loglikelihoods to calculate three model selection influence diagnostics: $\Delta\text{ind}_{\text{CHI}}$, $\Delta\text{ind}_{\text{BIC}}$, $\Delta\text{ind}_{\text{AIC}}$

After fitting the two competing models in *Mplus* in Part I, a researcher should have two ASCII files containing the casewise loglikelihood values from each fitted model. In our running illustration, the corresponding files were labeled *modela.dat* and *modelb.dat*.

1. Read in data. The researcher will need to read their two ASCII datasets into SAS. Example code for reading in the two ASCII datasets from our running illustration is shown below. The researcher will need to modify the **red portions** for their particular application. Variables must be listed on the INPUT line in the exact order that they were listed at the end of the *Mplus* output file, but they can be given different names.

- The SAS code for calculating model selection influence diagnostics assumes that these ASCII datasets are read in as temporary SAS datasets called *modela* and *modelb*.
- The SAS code also assumes the casewise loglikelihood contributions are read in for model A as *LL_A* and for Model B as *LL_B*.
- If models are nested, Model A should correspond with the more restricted model, and Model B should correspond with the less restricted model (consistent with the manuscript notation). If models are not nested, $\Delta\text{ind}_{\text{CHI}}$ diagnostics and plots should not be interpreted, but $\Delta\text{ind}_{\text{BIC}}$ and $\Delta\text{ind}_{\text{AIC}}$ diagnostics and plots can still be interpreted.

```
DATA modela;
INFILE "C:\modela.dat";
INPUT y1-y6 id LL_A; run;

DATA modelb;
INFILE "C:\modelb.dat";
INPUT y1-y6 id LL_B; run;
```

If the researcher's fitted models were single-level models, he/she can immediately proceed to running the code below that calculates the diagnostics. If the researcher's fitted models were multilevel models, however, he/she will need to reshape the exported dataset from Long format (one record per level 1 observation) to Wide format (1 record per case, a.k.a. cluster) before proceeding. When multilevel models are fitted, *Mplus* will provide the exported dataset in long format, where the loglikelihood contribution for the case (a.k.a. cluster) is repeated for each level 1 record. Computations for influence diagnostics require a wide format dataset with as many rows as cases (a.k.a. clusters) and one loglikelihood contribution per case. SAS code for converting from Long to Wide format is widely available, but for convenience an example script is provided in a supplementary section at the end of this document.

2. Specify input values and run code to compute and plot model selection influence diagnostics.

Researchers will need to modify the **red portions** of the below code for their particular application.

- N , k^A , and k^B were defined in the manuscript. To reiterate, N is the number of cases (highest level units in the analysis). k^A is the number of estimated (free) parameters in Model A, and k^B is the number of estimated (free) parameters in Model B. The variable *case* should be set equal to whatever the researcher is calling their case identifier in the SAS datasets. In the running illustration $N=100$, $k^A=18$, $k^B=19$, and the case identifier is called *id*.
- The researcher should indicate the path name to the folder where index plots will be saved, and the desired name of the new document containing plots. In the running illustration, the index plots will be saved to a file on the C drive called '*analyses*' in a document called

'myindexplots.' The document will by default be saved in RTF (rich text format), which allows plots to be easily copied and pasted into Word documents.

```

%let N=100;
%let case=id;
%let k_A=18;
%let k_B=19;

proc sort data=modela; by &case.; run;
proc sort data=modelb; by &case.; run;
data compare; merge modela modelb; by &case.; run;

data compare; set compare;
deltaindchi= -2*(LL_A-LL_B);
deltaindbic= deltaindchi+ ((&k_A.-&k_B.)*log(&N./(&N.-1)));
deltaindaic= deltaindchi;

parsimonyfacbic= (&k_A.-&k_B.)*log(&N./(&N.-1));
run;

proc means data=compare sum; var deltaindchi; run;
proc print data=compare (obs=1); var parsimonyfacbic; run;

options helpbrowser=sas;
ods rtf file="C:\analyses\myindexplots.rtf"; options orientation=portrait;

goptions reset=all hsize=5 vsize=4 ftext='times new roman' htext=1.15 ;
axis1 label=(a=90 h=1.15 f='times new roman' "deltaindCHI") minor=none;
axis2 label=(a=0 h=1.15 f='times new roman' "ID") minor=none value=(h=1.15
f='times new roman');

proc gplot data=compare;
symbol v=dot pointlabel=("#&case.") w=.4 h=.4 c=black;
plot deltaindchi*&case. / vaxis=axis1 haxis=axis2 ; run;

axis1 label=(a=90 h=1.15 f='times new roman' "deltaindBIC") minor=none;

proc gplot data=compare;
symbol v=dot pointlabel=("#&case.") w=.4 h=.4 c=black;
plot deltaindbic*&case. / vaxis=axis1 haxis=axis2 ; run;

axis1 label=(a=90 h=1.15 f='times new roman' "deltaindAIC") minor=none;

proc gplot data=compare;
symbol v=dot pointlabel=("#&case.") w=.4 h=.4 c=black;
plot deltaindaic*&case. / vaxis=axis1 haxis=axis2 ; run;

ods rtf close;

```

Running the above code supplies the following information. In the SAS output window, the sum of Δind_{CHI} across all cases is provided so that the researcher can cross-check this sum against the sample-level $\Delta \chi^2$ to make sure no coding errors were made. They should match (within rounding

error). For example, sums from the running appendix illustration were:

Variable	Sum
deltaindchi	13.9080000

Also in the SAS output window, the "parsimony correction factor" (discussed in the manuscript) for $\Delta\text{ind}_{\text{BIC}}$ is provided. It may be useful when presentation space is limited; a researcher could report a plot of $\Delta\text{ind}_{\text{CHI}}$ (or $\Delta\text{ind}_{\text{AIC}}$) and simply mention the parsimony correction factor for $\Delta\text{ind}_{\text{BIC}}$, rather than reporting index plots for multiple diagnostics. This reporting strategy was discussed in the manuscript. For example, parsimony correction factors from the running appendix illustration were:

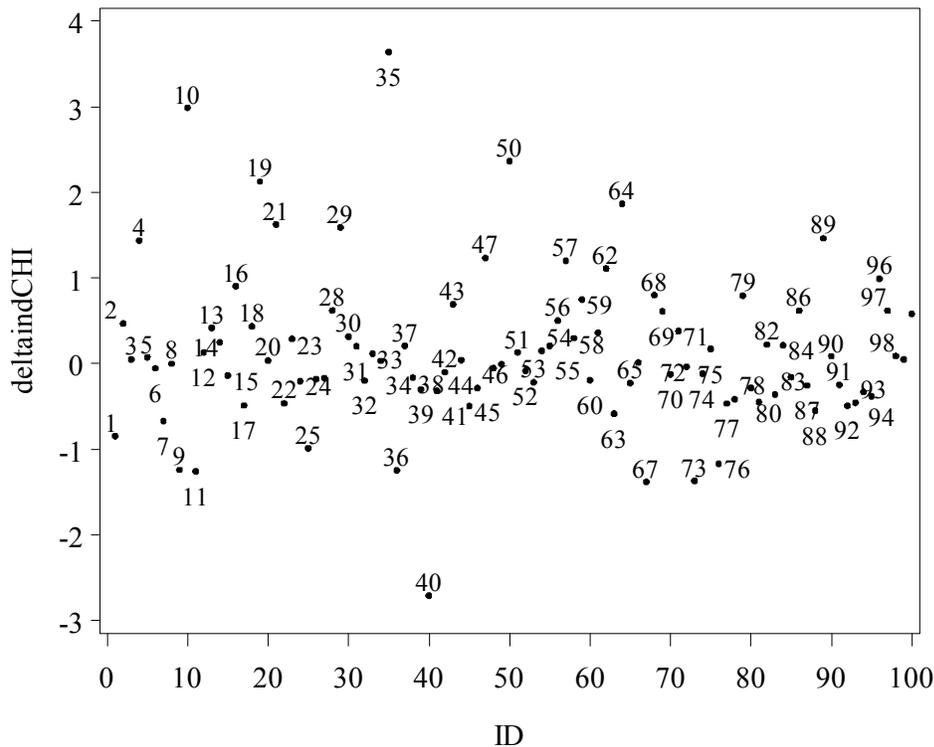
parsimonyfacbic	-0.010050
-----------------	-----------

Finally, the .RTF file containing three index plots:

- case ID vs. $\Delta\text{ind}_{\text{CHI}}$
- case ID vs. $\Delta\text{ind}_{\text{BIC}}$
- case ID vs. $\Delta\text{ind}_{\text{AIC}}$

will be saved to the directory requested by the researcher. As explained in the manuscript, $\Delta\text{ind}_{\text{CHI}}$ and $\Delta\text{ind}_{\text{AIC}}$ are numerically equivalent, but their interpretation and usage differs. Because a researcher may be interested in using one diagnostic instead of the other, we have provided separate plots of their values, with axes labeled for each diagnostic.

Each point on the index plot is labeled with its associated case ID for convenience. Plot formatting is similar to the index plots provided in the manuscript. For example, the case ID vs. $\Delta\text{ind}_{\text{CHI}}$ index plot from the running appendix illustration is:



References:

Muthén, L.K. and Muthén, B.O. (1998-2011). *Mplus User's Guide. Sixth Edition*. Los Angeles, CA: Muthén & Muthén.

Supplementary information on reshaping a dataset from Long to Wide in order to obtain influence diagnostics from multilevel models.

Reasons why multilevel analyses require an extra step of Long-to-Wide dataset conversion before computing case influence diagnostics were discussed earlier. Example code to accomplish this conversion is provided here, but other methods can be found in SAS documentation or online. Here the long-format temporary SAS datasets are called *modela_long* and *modelb_long*, the wide format datasets are called *modela_wide* and *modelb_wide*, and the case identifier is called *clusterid*.

```
proc sort data=modela_long; by clusterid; run;
proc transpose data=modela_long out=modela_wide; var LL_A; by clusterid; run;
data modela_wide; set modela_wide; keep COL1 cluster; rename COL1=LL_A; run;

proc sort data=modelb_long; by clusterid; run;
proc transpose data=modelb_long out=modelb_wide; var LL_B; by clusterid; run;
data modelb_wide; set modelb_wide; keep COL1 cluster; rename COL1=LL_B; run;
```