LCA Bootstrap Stata function users' guide (Version 1.0)

Liying Huang
John J. Dziak
Aaron T. Wagner
Stephanie T. Lanza
Penn State

Please send questions and comments to MChelpdesk@psu.edu.
The development of the LCA bootstrap Stata function was supported by National Institute on Drug Abuse Grant P50 DA039838. Shu Xu made significant contributions to the SAS version of this software. The authors would like to thank Amanda Applegate for her helpful comments.

Thank you for citing this users' guide when you use this macro. The suggested citation is Huang, L., Dziak, J. J., Wagner, A. T., & Lanza, S. T. (2016). *LCA bootstrap Stata function users' guide* (Version 1.0). University Park: The Methodology Center, Penn State. Retrieved from http://methodology.psu.edu

Contents

1 About the LCA Bootstrap Stata function

The LCA Bootstrap Stata function can assist users in choosing the number of classes for latent class analysis (LCA) models. It works in conjunction with Stata version 11.0 or higher and the LCA Stata plugin, version 1.2.1 or higher. This macro can perform the bootstrap likelihood ratio test to compare the fit of a latent class analysis (LCA) model with k classes ($k \geq 1$) to the fit of one with $k + 1$ classes.

The parametric bootstrap likelihood ratio test for LCA is described in McLachlan and Peel (2000); Nylund, Asparouhov, and Muthén (2007); and Collins, Fidler, Wugalter, and Long (1993). It is used to choose the number of classes in an LCA. More specifically, it tests the null hypothesis that a k-class LCA model is adequate to describe the population a particular sample came from, versus the alternative hypothesis that a $(k + 1)$-class LCA model is required. To do this, the null and alternative models are first fit to a given empirical dataset, and then the difference between them (in terms of a likelihood ratio test statistic) is recorded. Many random samples are then generated from a population in which the k-class null hypothesis is true, and then analyzed under both the k-class and the $(k+1)$-class models, to get an estimate of what the likelihood ratio test statistic distribution would be if the null hypothesis were true. If the observed likelihood ratio test statistic is bigger than most (say, 95%) of the simulated likelihood ratio test statistics, then the hypothesis is rejected.

The bootstrap p-value as used here is $(s + 1)/(B + 1)$, where B is the total number of bootstrap samples generated, and s is the number of bootstrap datasets having a likelihood ratio test statistic larger than that of the observed sample.

One limitation of the bootstrap test is that it can take hours to perform, since many LCA datasets are being simulated and analyzed. Also, the current version of this function is only for classic LCA models without covariates or special sampling features. Specifically, it is assumed that the models being compared do not have polytomous items as indicators (those with more than 2 possible responses) and do not have `covariates, groups, weights, clusters`, or special parameter restrictions (`restrict` option), even though these features are allowed in the LCA Stata plugin (See *LCA Stata Plugin User's Guide*; Lanza, Dziak, Huang, Wagner & Collins, 2015).

Note: The current version of the LCA Bootstrap Stata function handles only LCA models based on dichotomous items (e.g., yes/no). Items with more than 2 possible responses are not currently supported in this function.

2 Using the LCA Bootstrap Stata function

2.1 Managing files and preparing data

Three steps are required to set up the function before use.

1. Set up the LCA Stata plugin as described in the *LCA Stata Plugin Users' Guide.*
2. Unzip the folder downloaded from methodology.psu.edu and place all the files in the same folder where you installed the LCA Stata plugin.
3. Run an example.
   a. Open relevant ".do" file
   b. In the **4th line of code,** modify the path "D:\project\Stata_lca\LcaBootstrap-64bit\" to match the folder path where you placed the files. (This line has a comment, "/*CHANGE THIS PATH TO MATCH THE FILE LOCATION ON YOUR MACHINE*/") **NOTE: If there is one or more spaces in your directory path, you will need to put the path in double quotation marks, per Stata convention.**
   c. Save the changes.

The function is ready to use.

2.2 Syntax and input

Table 1. Option definitions for the LCA Bootstrap Stata function

Argument	Required	Description
`null_gammalist`	Y	Name of the list created from the gamma matrix output by the LCA Stata plugin run on the null (*k*-class) model
`null_rholist`	Y	Name of the list created from the rho matrix output by the LCA Stata plugin run on the null (*k*-class) model
`null_loglikelihood`	Y	Name of the list created from the log likelihood matrix output by the LCA Stata plugin run on the null (*k*-class) model
`alt_loglikelihood`	Y	Name of the list created from the log likelihood matrix output by the LCA Stata plugin run on the alternative ((*k*+1)-class) model
`simulate_samplesize`	Y	Original sample size used for the LCA, counting only those cases included in the analysis. This will be used as the sample size for the generated bootstrap dataset.
`num_bootstrap`	N	Number of bootstrap replications, which should be at least 99. A value of 999 is preferable but calculations may take longer. **(default = 99)**
`null_starts`	N	Number of starting values to fit under the null hypothesis in each bootstrap replication to help find the global maximum likelihood under this hypothesis. Must be at least 2, but 20 or more is recommended **(default = 20)**
`alt_nstarts`	N	Number of starting values to fit under the alternative hypothesis in each bootstrap replication to help find the global maximum likelihood under this hypothesis. Must be less than *null_starts* as the alternative model is more complex. Must be at least 2, but 20 or more is recommended **(default = 20)**
`cores`	N	Number of processor cores to use if more than one is available. **(default = 1)**

2.3 Output

If the calculations for the test are successful, the bootstrap *p*-value is displayed on the screen.

3 Example application

The following example is based on bootstrap-example2.do, available in the download. This is loosely based on the adolescent delinquent behaviors example shown on pages 11-12 of Collins and Lanza (2010). The analysis presented here is based on `N = 2000` adolescents' responses on five questionnaire items: rowdy, vandal, shoplift, steal, and fight. First a two-class and then a three-class model are fit to the data. The bootstrap likelihood ratio test is used to test whether the two-class model is adequate relative to the three-class alternative.

```
qui doLCA Rowdy Vandal Shoplift Steal Fight, ///
      nstart(10) ///
      nclass(2) ///
      freq(Count) ///
      seed(1000) ///
      categories(2 2 2 2 2)
```

Note: The next block of code is a basic Stata operation and not part of our function, but we include it here as a convenient review for some users.

The LCA Bootstrap Stata function relies on inputs from the LCA Stata plugin, but the LCA Stata plugin generates matrices, and the LCA Bootstrap Stata function (by Stata convention) cannot accept matrices as input for options. This means that two matrices—r(gamma) and r(rho)—must be converted to lists. The following code can be used for this purpose.

```
mat G = r(gamma)
mat R = r(rho)
local null_logliks = r(loglikelihood)

forvalues i=1/`=rowsof(G)' {
      forvalues j=1/`=colsof(G)' {
            local glist `glist' `=G[`i', `j']'
      }
}

forvalues i=1/`=rowsof(R)/2' {
      forvalues j=1/`=colsof(R)' {
            local rholist `rholist' `=R[`i', `j']'
      }
```

Then, the LCA Stata plugin is run with the alternate model which uses *k*+1 classes, in this case 3 classes.

```
qui doLCA Rowdy Vandal Shoplift Steal Fight, ///
      nstart(10) ///
      nclass(3) ///
      freq(Count) ///
      seed(1000) ///
      categories(2 2 2 2 2)
local alt_logliks = r(loglikelihood)

cap drop _all
```

Finally, the LCA Bootstrap Stata function is run to compare the relative fit of the two-class and 3-class models and the p-value is displayed.

```
doLcaBootstrap, ///
      null_gammalist(`glist')    ///
            null_rholist(`rholist')  ///
            null_loglikelihood(`null_logliks') ///
            alt_loglikelihood(`alt_logliks') ///
            simulate_samplesize(2000)  ///
            num_bootstrap(99)  ///
            null_nstarts(10)   ///
            alt_nstarts(10)

dis e(p_value));
```

3.1 Example results

The *p*-value is .01, suggesting that the two-class model is too restrictive.

```
. dis e(p_value)
.01
```

```
.
end of do-file
```

Note that because the bootstrap procedure is partially random, your *p*-value may not be exactly the same, even with the same data. To reduce the random variability of the *p*-value, use a higher value for `num_bootstrap`, such as `499` or `999`.

3.2 Model comparison

Now imagine that sequential analyses were conducted for selecting the number of latent classes of adolescent delinquent behaviors based on bootstrap likelihood ratio tests. The theoretical results are summarized in the table below. Because no difference was detected between a four-class model and a five-class model in terms of likelihood ratio test statistics, we select a four-class model of adolescent delinquent behaviors.

Null model	Vs.	Alternative model	p-value
1-class		2-class	.01
2-class		3-class	.01
3-class		4-class	.01
4-class		**5-class**	**.45**

4 References

Collins, L. M., Fidler, P. L., Wugalter, S. E., & Long, J. L. (1993). Goodness-of-fit testing for latent class models. *Multivariate Behavioral Research, 28,* 375-389.

Collins, L. M., & Lanza, S. T. (2010). *Latent class and latent transition analysis: With applications in the social, behavioral, and health sciences.* New York, NY: Wiley.

Lanza, S. T., Dziak, J. J., Huang, L., Wagner, A. T., & Collins, L. M. (2015). *LCA Stata plugin users' guide (Version 1.2).* University Park: The Methodology Center, Penn State. Retrieved from methodology.psu.edu

McLachlan, G., & Peel, D. (2000). *Finite mixture models.* New York, NY: Wiley.

Nylund, K. L., Asparouhov, T., & Muthén, B. O. (2007). Deciding on the number of classes in latent class analysis and growth mixture modeling: A Monte Carlo simulation study. *Structural Equation Modeling, 14,* 535-569.