PennState The Methodology Center
advancing methods, improving health

LCA_Distal_BCH SAS Macro Users' Guide (Version 1.1)

John J. Dziak
Bethany C. Bray
Aaron T. Wagner
Penn State

Please send questions and comments to *MChelpdesk@psu.edu*.

The development of the SAS %LCA_Distal_BCH macro was supported by National Institute on Drug Abuse Grants P50 DA10075 and P50 DA039838. The authors would like to thank Amanda Applegate for helpful comments.

Thank you for citing this users' guide when you use this macro. Suggested citation:

Dziak, J. J., Bray, B. C., & Wagner, A. T. (2017). *LCA_Distal_BCH SAS macro users' guide* (Version 1.1). University Park, PA: The Methodology Center, Penn State. Retrieved from http://methodology.psu.edu

Contents

1 About the %LCA_Distal_BCH Macro

The SAS %LCA_Distal_BCH macro estimates the association between a latent class variable and a distal outcome using the approach of Bolck, Croon, and Hagenaar (2004), as adapted by Vermunt (2010) and Vermunt and Magidson (2015). The %LCA_Distal_BCH macro is designed to work with SAS Version 9.1 or higher and PROC LCA.

The %LCA_Distal_BCH macro

- uses simple, minimal syntax;
- estimates class-specific response probabilities and standard errors for binary and categorical distal outcomes;
- estimates class-specific means and standard errors for continuous and count distal outcomes;
- provides significance tests to compare distal outcome means or proportions between classes
- can accommodate distal outcomes for multiple demographic groups.

This guide assumes the user has a working knowledge of latent class analysis and PROC LCA. The book, *Latent class and latent transition analysis: With applications in the social, behavioral, and health sciences* (Collins & Lanza, 2010), provides a comprehensive introduction to the use of latent class analysis in applied research.

To use this macro, you must have PROC LCA version 1.3.2 or higher installed. PROC LCA and the accompanying users' guide can be downloaded from http://methodology.psu.edu/downloads.

This macro differs from the previous LCA_Distal macro, which has now been renamed LCA_Distal_LTB. The difference is that the LCA_Distal_BCH macro uses the BCH (Bolck, Croon, & Hagenaar, 2004) method, rather than the LTB (Lanza, Tan, & Bray, 2013) method. The LTB method can be biased for the means of continuous outcomes when the response variance differs between latent classes. The BCH method is more robust and can offer better performance (Bakk, Oberki, & Vermunt, 2016; Bakk & Vermunt, 2016; Dziak, Bray, Zhang, Zhang, & Lanza, 2016).

The variable designated as a distal outcome, for purposes of this macro, does not literally have to represent an event that occurs later in time than the original measurements. The macro was originally intended for describing the proportions of later events among classes (e.g., predicting relapse from latent class of withdrawal symptoms), but the same method works for describing other covariates that characterize classes (e.g., comparing psychological profiles or demographic variables among latent classes of drug withdrawal symptoms). However, the current macro works for one outcome at a time and currently does not allow adjusting for other outcomes simultaneously. Instead, it is intended to be run separately for each outcome or covariate of interest.

2 System Requirements

The %LCA_Distal_BCH macro requires

- SAS Version 9.1 or higher (Windows version),
- PROC LCA & PROC LTA Version 1.3.2 or higher (to fit LCA models),

Note: SAS/STAT is sold separately from the base SAS package, but most university licenses include it. If you can run PROC LCA, you can run this macro.

3 The BCH Approach to LCA With a Distal Outcome

Researchers are often interested in the relationship between a latent class variable, *C,* and a distal outcome, *Z*. Often, they wish to compare the class-specific expected value E(*Z*|*C*=c) for each class *c*. This expected value is the same as the mean (average) for count or continuous variables. For binary variables coded as 0 or 1, the expected value is the proportion of 1s in the population, or equivalently, the probability of a 1 rather than a 0 for a single randomly selected population member.

The BCH method is a kind of "three-step" method. This means that (1) the parameters of the LCA model first are estimated without the distal outcome, then (2) the posterior probabilities of class membership based on this model are used to compute a special weighting variable, and finally (3) the weighting variable is used to calculate a weighted average of *Z* for each class. The simplest approach to creating weights is to use either the posterior probabilities themselves as weights ("proportional assignment") or to round the highest probability for each subject to 1 and the others to zero ("modal assignment"), and to apply no further adjustment. However, this treats the posterior probabilities as if they were known quantities measuring degrees of class membership, and does not take into account uncertainty introduced by possible misclassification when estimating the model parameters. Bolck, Croon, and Hagenaars (2004) proposed a more accurate method that accounts for misclassification probabilities. Although they first proposed this method only in the case of categorical outcomes, Vermunt (2010) explained how to adapt it to continuous outcomes as well.

This macro will calculate distal outcome estimates with either modal or proportional assignment, and either with BCH adjustment ("BCH" estimates) or without it ("naïve" or "unadjusted" estimates). It is generally better to use BCH adjustment rather than unadjusted estimates. However, as long as BCH adjustment is being used, it usually does not matter very much whether modal assignment or proportional assignment is used. Occasionally, BCH assignment has been found to give an uninterpretable value (such as a negative probability); in this case, it is better to revert to the unadjusted assignment. The three steps followed by this macro are described further below.

Step 1. Fit the LCA model to define latent class memberships, using only the indicator variables **Y=**$Y_1,\ldots,Y_m$, without including the distal outcome *Z* in the model. This will provide posterior probabilities of class membership, $\omega_{ic} = \mathrm{P}(C = c|Y = y_i)$, for each individual i=1,…,N in the dataset and each class c=1,…,n_c.

Step 2. Construct the weights for use in calculating weighted averages for each class on the distal outcome. The details depend on the options chosen.

- Unadjusted Modal Assignment. For each individual i and each possible class c, define the class weights w_{ic}. Specifically, let $\mathrm{w}_{ic} = 1$ if c is the most likely class (the maximum among $\omega_{i1}, \dots, \omega_{in_c}$) for a given individual, and 0 otherwise. For example, if individual i is estimated to have a 60% chance of belonging to class 2, then individual i will count as 100% of a member of class 2 and 0% of other classes.
- Unadjusted Proportional Assignment. Define the class weights as $\mathrm{w}_{ic} = \omega_{ic}$ for each individual i and each possible class c. For example, if individual i is estimated to have a 60% chance of belonging to class 2, then individual i will count as 60% of a member of class 2 when calculating weighted averages; the remaining 40% of the membership of individual i is divided among the remaining classes.
- BCH-Adjusted Modal Assignment. Calculate the misclassification matrix **D**. The entry in row a and column b of **D** represents the estimated probability that a subject who truly belongs to class a would be labeled as belonging to class b. Specifically, $\mathbf{D}_{ab}$ is calculated as $\sum_{i=1}^{N} \omega_{ia} w_{ib}^{unadj} / N\, \gamma_a$, where N is the number of subjects, w_{ib}^{unadj} is the unadjusted modal weight for individual i in class b, and γ_a is the estimated overall class probability P(C=a). Then calculate the vector of BCH weights using linear algebra as $\mathbf{w}^{BCH} = \mathbf{w}^{unadj}\mathbf{D}^{-1}$, where $\mathbf{w}^{unadj}$ is the $N \times n_c$ matrix of unadjusted modal weights w.
- BCH-Adjusted Proportional Assignment**.** Same as BCH-adjusted modal, but use the proportional weights for w instead of using the modal weights.

Step 3. Estimate the expected value of the distal outcome within each latent class by taking a weighted average of the observed values for all participants, weighted by each participant's value of $\mathbf{w}^{unadj}$ or $\mathbf{w}^{BCH}$, as requested by the user. Standard errors are calculated using Taylor linearization ("sandwich" covariance estimation).

Standard errors and tests. In principle, there are two ways of doing tests, or obtaining standard errors or confidence intervals, for non-normal distal outcomes. One is to treat them as simply averages and ignore the fact that they are not normally distributed. This is convenient and asymptotically valid, although not the most statistically efficient. The other is to assume a non-normal distribution (here we use Bernoulli for binary and Poisson for count) and construct the confidence intervals or tests for the underlying parameter (the logit probability or log mean) of this distribution. This macro mostly imitates the behavior of the LatentGOLD software, in that standard errors are provided using the simpler method, and tests are performed using the more complicated method. For the binary case, non-symmetric confidence intervals are additionally provided using the more complicated method (calculating standard errors and confidence interval limits for the logit, and then back-transforming the ends of this confidence interval to describe the observed mean).

Pairwise and omnibus tests. The macro provides Wald tests and p-values for comparing the expected values of the distal outcome between each pair of latent classes, testing the null hypothesis that the expected values are equal. The p-values are not adjusted for multiple comparisons, but a user who wishes to apply a Bonferroni correction could simply divide the alpha level used for comparison (e.g., .05) by the number of pairs being compared: specifically, by $n_c(n_c - 1)/2$. In addition to these tests, an omnibus test simultaneously comparing all of the expected values is also performed. For categorical outcomes in the current version of the macro, only an omnibus test, rather than pairwise tests, is performed.

Sampling weights. If complex survey sample weights are used in the LCA (the `weight` option in PROC LCA) then these must be specified in this macro also (using the `sampling_weight=` optional argument). Sampling weights are implemented by multiplying each w_{ic} by the corresponding sampling weight s_i. This is done before postmultiplying by $\mathbf{D}^{-1}$ in the BCH method. Note that although survey weights can be accommodated, the current version of the macro does not account for clustering when calculating standard errors.

Grouping variable. The calculations of the macro can accommodate an observed grouping variable (usually gender or other demographic categories) as in the `groups` command in PROC LCA. The macro assumes measurement invariance across groups and performs calculations separately for each group. Separate output is also provided for each group.

4 Using the %LCA_Distal_BCH Macro

Table 1. Argument Definitions for the %LCA_Distal Macro.

Argument	Required	Description
`input_data`	Y	Input data set. The distal outcome must be included as one of the variables.
`param`	Y	Name of the data set generated by PROC LCA as the OUTPARAM output. The data set contains estimates of the beta parameters.
`post`	Y	Name of the data set generated by PROC LCA as the OUTPOST output. The data set contains estimates of the posterior probabilities.
`id`	Y	Subject identification variable.
`distal`	Y	Distal outcome variable.
`metric`	Y	Metric assumed for the within-class distribution of the distal outcome variable. This may be the word “binary,” “categorical,” “count,” or “numerical,” without quotes.
`group`	N	Variable for multiple groups. If no group argument is supplied, the macro assumes there is only one group.
`Alpha`	N	Significance level. Default = 0.05.
`sampling_weight`	N	Name of the variable specifying survey weight. This option only works in the binary outcome case. It assumes that WEIGHT has also been used in the previous call to PROC LCA.
`adjustment_method`	N	The method, if any, of adjusting the class membership weights for the possibilitiy of misclassification. This may be “BCH” (default, recommended) or “unadjusted,” without quotes.
`assignment`	N	The method of generating class membership weights based on the posterior probabilities, before doing the BCH adjustment if any. This may be “modal” (default) or “proportional,” without quotes.

4.1 Preparation

A SAS macro is a special block of SAS commands. The block is first defined and then called when needed. Four steps need to be completed before you run the macro.

1. If you haven't already done so, download and save the macro to a designated path (e.g., `S:\myfolder\`).
2. Direct SAS to read the macro code from the path, using a SAS %INCLUDE statement such as

```
%INCLUDE “S:\myfolder\LCA_Distal_BCH_v110.sas”;
```

3. Direct SAS to the input data file. We assume the data set is a permanent file saved to a designated directory. If so, we recommend using a “libname” statement. The statement should give the libname command, name the library, and then identify the path to the data. For example,

```
libname sasf “s:\myfolder\”;
```

4. Ensure that the distal outcome is coded as follows:
   - Binary: 0, 1
   - Continuous: original coding or standardized variable
   - Count: original coding (0, 1, 2, …)
   - Categorical: 1, 2, …

Note: Missingness in the distal outcome variable should be imputed (e.g., multiple imputation; Schafer, 1997). Otherwise, cases with missing values in the distal outcome variable must be removed from the analysis.

4.2 Estimation of the Latent Class Model in PROC LCA

Use PROC LCA to generate the output needed for use by the %LCA_Distal macro. First, you must select the LCA model. This process is described in Chapter 5 of the *PROC LCA & PROC LTA Users' Guide* (Lanza, Dziak, Huang, Xu, & Collins, 2011).

Once model selection is complete, generate a file containing the parameter estimates to be used in the macro by estimating the latent class model with the distal outcome included as a covariate. This file can be generated using the OUTPARAM option in PROC LCA. (See section 5.3 of the *PROC LCA & PROC LTA Users' Guide* for more information.) The PROC LCA syntax will be similar to the following:

```
PROC LCA DATA = my_data OUTPARAM = my_param OUTPOST = my_post; /* the
    input data set, the file to be generated containing the parameter
    estimates, the file to be generated containing the posterior
    probabilities */
    NCLASS 5; /* the number of latent classes */
    ITEMS item001 item002 item003 item004 item005 item006 item007
    item008; /* indicator variables used to measure the latent class
    variable */
    CATEGORIES 2 2 2 2 2 2 2 2; /* number of response categories for
    each indicator variable (in this case, all dichotomous) */
    ID SubjectID /*the unique integer representing each case */
    SEED 54327; /* an arbitrary number to be used as a seed for
    generating reproducible random starting values */
RUN;
```

The `covariates` statement should not be used. The `group` or `weight` statement may be used, if demographic groups or survey weights are required in the model. Other arguments available in PROC LCA, such as `rho prior`, `maxiter`, and `criterion` may be necessary for estimation of the latent class model. Refer to the *PROC LCA& PROC LTA Users' Guide* for more information. The %create_group macro, which is sometimes used in the Methodology Center's LCA_Distal_LTB macro, is not needed for the LCA_Distal_BCH macro.

4.3 Macro Syntax and Input

Call the macro using a percent sign, its name, and user-defined arguments in parentheses. The macro parameters are shown below.

```
%LCA_Distal (
     input_data = data set name,
     param = name of OUTPARAM data set created by PROC LCA,
     post = name of OUTPOST data set created by PROC LCA,
     id = variable,
     distal = variable,
     group = variable,
     metric = word describing the outcome metric (binary,
     continuous, count or categorical),
     sampling_weight = survey weighting variable name (optional and
     used in binary case only),
     adjustment_method = word describing the misclassification
     adjustment method (BCH or unadjusted),
     assignment = word describing the class membership weight
```

```
        assignment option (modal or proportional)
    ) ;
```

4.4 Output

The macro produces both screen output and SAS datasets. The screen output first presents a table of estimates and standard errors for the expected value of the distal outcome within each class. In addition, for binary distal outcomes, a table of log odds estimates and asymmetric confidence intervals is provided. The macro then provides a table of Wald chi-squared tests for testing the equality of expected values between classes. These include both pairwise and omnibus tests, except for categorical distal outcomes, for which only omnibus tests are provided.

Two SAS datasets, Distal_Estimates and Distal_Tests, are also created. These contain similar information to what is shown on screen. For binary outcomes, a dataset called Distal_Log_Odds is also created. Although these datasets contain the same information that is shown on screen, they can be useful if you want to copy and save the results of many analyses into a larger compilation (e.g., in a simulation loop).

5 Demonstrations of the %LCA_Distal_BCH Macro

In this section, we first describe the structure of the data sets and the variables to be analyzed. Then, we illustrate how to estimate the distribution of the distal outcome within each latent class using the %LCA_Distal_BCH macro and describe the output of the macro. Section 5.1 describes use of the macro with a binary distal outcome. Continuous, count, and categorical outcomes are discussed in sections 5.2, 5.3, and 5.4, respectively.

For demonstrations of the macro with multiple groups, see chapter 6.

5.1 Estimating a Binary Distal Outcome

Before attempting to complete the following example, please download the file *%LCA_Distal Examples* from the %LCA_Distal macros download page at http://methodology.psu.edu. Also, verify that you are running PROC LCA v.1.3.2 or higher.

5.1.1 Example Data

Below are the first 10 observations from the SAS data set **simdata_binary.sas7bdat**, which is contained in the *%LCA_Distal Examples* file.

ID	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	Z
1	2	2	1	2	2	2	2	2	1
2	1	1	2	2	2	2	2	2	0
3	2	1	2	1	1	1	1	1	0
4	2	2	2	2	2	2	2	2	1
5	2	2	2	2	2	2	2	2	1
6	1	1	1	2	2	2	2	2	1
7	2	2	1	2	2	2	2	2	1
8	2	2	2	2	2	2	2	2	1
9	2	2	2	2	2	2	2	2	1
10	2	2	2	2	1	2	2	2	1

ID= subject's identification variable,

Item001,…, Item008= 8 items used to measure the latent class variable

Z= the distal outcome (Note: binary distal outcome should be coded using 0s and 1s.)

5.1.2 Example Syntax

Include a "libname" statement prior to running the macro to direct SAS to the data file.

```
libname sasf "S:\myfolder\";
```

Note: We suppose that the SAS data set exists in the folder S:\myfolder\. This path represents any user-specified folder.

Once the LCA model has been identified, estimate the LCA model using PROC LCA. Notice that Z is not included as a covariate in this step.

```
PROC LCA DATA = SimData_Binary OUTPARAM = Binary_param OUTPOST =
Binary_post;
  ID id;
  NCLASS 5;
  ITEMS item001-item008;
  CATEGORIES 2 2 2 2 2 2 2 2;
  SEED 12345;
  RHO PRIOR = 1;
  NSTARTS 20;
  MAXITER 5000;
  CRITERION 0.000001;
RUN;
```

The output is described in the *PROC LCA & PROC LTA Users' Guide.* It should include the files Binary_param and Binary_post in the WORK directory.

VIEWTABLE: Work.Binary_param

	Parameter Type	Group Number	Variable Name	Response Category	Estimate - Latent Class 1	Estimate - Latent Class 2	Estimate - Latent Class 3	Estimate - Latent Class 4	Estimate - Latent
1	GAMMA	1		.	0.103589	0.280180	0.158494	0.327222	0.130514
2	RHO	1	Item001	1	0.932447	0.906181	0.130193	0.058604	0.977193
3	RHO	1	Item002	1	0.872605	0.880844	0.136937	0.070370	0.827092
4	RHO	1	Item003	1	0.853765	0.849715	0.134888	0.081468	0.960024
5	RHO	1	Item004	1	0.947957	0.882878	0.048055	0.080657	0.875625
6	RHO	1	Item005	1	0.933083	0.089037	0.711402	0.081087	0.680476
7	RHO	1	Item006	1	0.859787	0.002642	0.798495	0.015504	0.966975
8	RHO	1	Item007	1	0.919828	0.063420	0.093312	0.121607	0.001692
9	RHO	1	Item008	1	0.771090	0.154943	0.092573	0.118761	0.084783
10	RHO	1	Item001	2	0.067553	0.093819	0.869807	0.941396	0.022807
11	RHO	1	Item002	2	0.127395	0.119156	0.863063	0.929630	0.172908
12	RHO	1	Item003	2	0.146235	0.150285	0.865112	0.918532	0.039976
13	RHO	1	Item004	2	0.052043	0.117122	0.951945	0.919343	0.124375
14	RHO	1	Item005	2	0.066917	0.910963	0.288598	0.918913	0.319524
15	RHO	1	Item006	2	0.140213	0.997358	0.201505	0.984496	0.033025
16	RHO	1	Item007	2	0.080172	0.936580	0.906688	0.878393	0.998308
17	RHO	1	Item008	2	0.228910	0.845057	0.907427	0.881239	0.915217

Binary_param

VIEWTABLE: Work.Binary_post

	id	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	Latent Class 1 Posterior Prob	Latent Class 2 Posterior Prob	Latent Class 3 Posterior Prob	Latent Class 4 Posterior Prob	Latent Class 5 Posterior Prob	BEST
1	1	2	2	1	2	2	2	2	2	0.000000	0.014031	0.045757	0.940174	0.000037	4
2	2	1	1	2	2	2	2	2	2	0.000029	0.755819	0.029729	0.213072	0.001350	2
3	3	2	1	2	1	1	1	1	1	0.991977	0.000015	0.007931	0.000060	0.000017	1
4	4	2	2	2	2	2	2	2	2	0.000000	0.000228	0.026933	0.972839	0.000000	4
5	5	2	2	2	2	2	2	2	2	0.000000	0.000228	0.026933	0.972839	0.000000	4
6	6	1	1	1	2	2	2	2	2	0.000040	0.987036	0.001071	0.004365	0.007489	2
7	7	2	2	1	2	2	2	2	2	0.000000	0.014031	0.045757	0.940174	0.000037	4
8	8	2	2	2	2	2	2	2	2	0.000000	0.000228	0.026933	0.972839	0.000000	4
9	9	2	2	2	2	2	2	2	2	0.000000	0.000228	0.026933	0.972839	0.000000	4
10	10	2	2	2	2	1	2	2	2	0.000001	0.000146	0.436037	0.563814	0.000002	4
11	11	2	2	2	2	2	2	2	2	0.000000	0.000228	0.026933	0.972839	0.000000	4
12	12	2	2	1	2	2	2	2	2	0.000000	0.014031	0.045757	0.940174	0.000037	4
13	13	1	1	1	1	1	1	2	2	0.018478	0.000575	0.000157	0.000000	0.980790	5
14	14	2	2	2	2	2	2	1	2	0.000001	0.000112	0.020163	0.979724	0.000000	4
15	15	1	1	1	2	2	2	2	2	0.000040	0.987036	0.001071	0.004365	0.007489	2
16	16	1	2	2	2	2	2	2	2	0.000001	0.032931	0.060351	0.906625	0.000091	4

Binary_post

Now the distal outcomes macro can be run. Include the macro and enter the proper syntax in SAS.

```
%LCA_Distal_BCH(input_data = SimData_Binary,
                param = Binary_param,
                        post = Binary_post,
                        id = id,
                        distal = z,
                        metric = binary );
```

The `input_data` argument identifies the data file. The `param` argument directs the macro to the parameters in the outparam file generated by PROC LCA. The `post` argument directs the macro to the posterior probabilities in the outpost file generated by PROC LCA. The `id` variable identifies the column in the dataset that uniquely identifies subjects. The `distal` argument identifies the distal outcome variable in the data set. The `metric` argument indicates that the distal outcome is binary.

In this example there were no survey weights. If there had been, it would be necessary to add a line such as

```
WEIGHT SurveyWeight;
```

to the PROC LCA syntax and a line such as

```
sampling_weight=SurveyWeight,
```

to the macro syntax.

5.1.3 Example Output

Below is the onscreen output. It includes the class-specific distribution estimates for the distal outcome, the estimated class-conditional probabilities, the Wald test statistic on class-conditional probabilities, and the p-value on class-conditional probabilities.

Binary dataset
BCH Estimation of Proportions of z by Latent Class

Estimates using BCH Modal Weighting

CLASS	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.53702	0.073266	0.39342	0.68062
2	0.82074	0.033527	0.75503	0.88645
3	0.77324	0.053807	0.66778	0.87870
4	0.92123	0.024249	0.87371	0.96876
5	0.69274	0.061255	0.57268	0.81281

Binary dataset
BCH Estimation of Proportions of z by Latent Class

Confidence Intervals for Probabilities

CLASS	DISTAL_PROBABILITY	DISTAL_LOG_ODDS	CI95_LOWER_LOG_ODDS	CI95_UPPER_LOG_ODDS	CI95_LOWER_PROBABILITY	CI95_UPPER_PROBABILITY
1	0.53702	0.14836	-0.42922	0.72593	0.39431	0.67391
2	0.82074	1.52137	1.07473	1.96802	0.74550	0.87740
3	0.77324	1.22670	0.62523	1.82818	0.65141	0.86154
4	0.92123	2.45921	1.80423	3.11419	0.85866	0.95747
5	0.69274	0.81298	0.24892	1.37705	0.56191	0.79852

Binary dataset
BCH Estimation of Proportions of z by Latent Class

Wald Chi-Squared Tests

NAME	ESTIMATE	STD_ERROR	WALD_STATISTIC	DF	P_VALUE
Difference in Log Odds: Class 2 versus Class 1	1.37301	0.37516	13.3940	1	0.00025
Difference in Log Odds: Class 3 versus Class 1	1.07835	0.42770	6.3567	1	0.01169
Difference in Log Odds: Class 4 versus Class 1	2.31085	0.44519	26.9436	1	0.00000
Difference in Log Odds: Class 5 versus Class 1	0.66462	0.42545	2.4403	1	0.11825
Difference in Log Odds: Class 3 versus Class 2	-0.29467	0.38342	0.5906	1	0.44218
Difference in Log Odds: Class 4 versus Class 2	0.93784	0.40823	5.2777	1	0.02160
Difference in Log Odds: Class 5 versus Class 2	-0.70839	0.37206	3.6250	1	0.05692
Difference in Log Odds: Class 4 versus Class 3	1.23251	0.49528	6.1927	1	0.01283
Difference in Log Odds: Class 5 versus Class 3	-0.41372	0.42683	0.9395	1	0.33240
Difference in Log Odds: Class 5 versus Class 4	-1.64623	0.43989	14.0056	1	0.00018
Omnibus Test	.	.	31.9581	4	0.00000

5.1.4 Overall Response Proportions

When interpreting the estimated response proportions within each of the latent classes, it may be useful to compare them to the overall estimated response proportion, ignoring latent class. This can be accomplished in the usual way using PROC FREQ (if survey weights are not used) or by using PROC SURVEYFREQ with its `weight` statement (if survey weights are being used). For example, one can use the syntax

```
PROC FREQ DATA= SimData_Binary;
      TABLES z;
RUN;
```

In the artificial dataset provided for this example, exactly 80% of the distal outcomes are yes (1).

The FREQ Procedure

z	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	100	20.00	100	20.00
1	400	80.00	500	100.00

Technical note: PROC LCA (and therefore %LCA_Distal_BCH) ignores participants who omit all of the answers to the indicators (items). If there are many subjects who omit all items, then the subsample being described by %LCA_Distal_BCH may noticeably differ from the whole sample. If so, the user might consider omitting these subjects before running PROC FREQ or PROC SURVEYFREQ, for compatibility with the results found in %LCA_Distal_BCH. However, in most cases this will probably not be necessary, because most participants will answer at least some of the LCA items.

5.2 Estimating a Continuous Distal Outcome

Before attempting to complete the following example, please download the file *%LCA_Distal Examples* from the %LCA_Distal_BCH macro download page.

5.2.1 Example Data

In **simdata_conti.sas7bdat**, the data structure is similar to the data set in section 5.1 of this document. However, instead of binary values for z, the values are continuous.

ID	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	Z
1	2	1	2	2	2	2	2	2	-1.8513098
2	1	1	2	1	2	2	2	2	-0.5950087
3	2	2	2	1	1	1	2	2	1.55437269
4	2	1	2	2	2	2	2	2	0.89742276
5	1	1	1	1	2	2	2	2	-0.3121734
6	1	1	1	1	2	2	2	2	-1.5068341
7	2	2	1	2	2	2	2	2	0.73713821
8	1	1	1	2	1	1	2	2	1.8747736
9	1	1	1	1	1	1	2	1	-0.0463611
10	1	1	1	1	1	1	1	1	-0.1706686

ID= subject's identification variable

Item001,…, Item008= 8 items used to measure the latent class variable

Z= the distal outcome (in this case a CONTINUOUS distal outcome)

5.2.2 Example Syntax

Include a "libname" statement prior to running the macro to direct SAS to the data file.

```
libname sasf "S:\myfolder\";
```

Note: we suppose that the SAS data set exists in the folder `S:\myfolder\`. This path represents any user-specified folder.

Estimate the LCA model using PROC LCA.

```
PROC LCA DATA = SimData_conti OUTPARAM = conti_param OUTPOST =
conti_post ;
     ID id;
     NCLASS 5;
     ITEMS item001-item008;
     CATEGORIES 2 2 2 2 2 2 2 2;
     SEED 12345;
     RHO PRIOR = 1;
     NSTARTS 20;
     MAXITER 5000;
     CRITERION 0.000001;
RUN;
```

Now, include the macro and enter the following syntax in SAS.

```
%LCA_Distal_BCH(input_data = SimData_conti,
                param = conti_param,
                post = conti_post,
                id = id,
                distal = z,
                metric = Continuous );
```

The `input_data` argument identifies the data file. The `param` argument directs the macro to the parameters generated in the OUTPARAM file generated by PROC LCA. The `id` and `distal` arguments identify the subject identification variable and the distal outcome. The `metric` argument indicates that the distal outcome is continuous, and `output_dataset_name` names the macro's output.

In this example there were no survey weights. If there had been, it would be necessary to add a line such as

```
WEIGHT SurveyWeight;
```

to the PROC LCA syntax and a line such as

```
sampling_weight=SurveyWeight,
```

to the macro syntax.

5.2.3 Example Output

The estimated means, along with standard errors and 95% confidence intervals, are shown in the output below.

BCH Estimation of Means of z by Latent Class

Estimates using BCH Modal Weighting

CLASS	DISTAL_MEAN	DISTAL_STD_ERROR_FOR_MEAN	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	-0.00534	0.13035	-0.26083	0.25015
2	0.18505	0.12571	-0.06134	0.43144
3	-0.20812	0.08393	-0.37262	-0.04361
4	0.43142	0.19112	0.05683	0.80601
5	-0.38141	0.06457	-0.50796	-0.25486

Tests of the differences between means are shown in the output below.

BCH Estimation of Means of z by Latent Class

Wald Chi-Squared Tests

NAME	ESTIMATE	STD_ERROR	WALD_STATISTIC	DF	P_VALUE
Difference in Means: Class 2 versus Class 1	0.19039	0.18185	1.0960	1	0.29514
Difference in Means: Class 3 versus Class 1	-0.20278	0.15486	1.7147	1	0.19038
Difference in Means: Class 4 versus Class 1	0.43676	0.23227	3.5358	1	0.06006
Difference in Means: Class 5 versus Class 1	-0.37607	0.14858	6.4068	1	0.01137
Difference in Means: Class 3 versus Class 2	-0.39316	0.15389	6.5276	1	0.01062
Difference in Means: Class 4 versus Class 2	0.24637	0.23358	1.1125	1	0.29154
Difference in Means: Class 5 versus Class 2	-0.56645	0.14133	16.0643	1	0.00006
Difference in Means: Class 4 versus Class 3	0.63954	0.21793	8.6119	1	0.00334
Difference in Means: Class 5 versus Class 3	-0.17329	0.10791	2.5788	1	0.10831
Difference in Means: Class 5 versus Class 4	-0.81283	0.20209	16.1771	1	0.00006
Omnibus Test	.	.	30.6440	4	0.00000

These output tables are also generated as datasets, namely distal_estimates and distal_tests.

5.2.4 Overall Response Means

When interpreting the estimated response means within each of the latent classes, it may be useful to compare them to the overall estimated response mean, ignoring latent class. This can be accomplished using PROC MEANS (if survey weights are not used) or by using PROC SURVEYMEANS with the `weight` statement (if survey weights are being used). For example, one can use the syntax

```
PROC MEANS DATA= SimData_conti;
      VAR z;
RUN;
```

In the artificial dataset provided for this example, the mean of the distal outcome is -0.1786357.

The MEANS Procedure

Analysis Variable : z				
N	Mean	Std Dev	Minimum	Maximum
500	-0.1786357	0.9418012	-3.1881485	2.1672993

Technical note: PROC LCA (and therefore %LCA_Distal_BCH) ignores participants who omit all of the answers to the indicators (items). If there are many subjects who omit all items, then the subsample being described by %LCA_Distal_BCH may noticeably differ from the whole sample. If so, the user might consider omitting these subjects before running PROC MEANS or PROC SURVEYMEANS, for compatibility with the results found in %LCA_Distal_BCH. However, in most cases this will probably not be necessary, because most participants will answer at least some of the LCA items.

5.3 Estimating a Count Distal Outcome

Before attempting to complete the following example, please download the file *%LCA_Distal Examples* from the %LCA_Distal macros download page.

5.3.1 Example Data

In **simdata_count.sas7bdat**, the data structure is similar to the dataset in section 5.1 of this document. However, the item *z* contains count responses with values from 0 to 4.

ID	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	Z
1	2	2	2	2	1	1	2	1	2
2	2	2	2	2	2	2	2	2	0
3	2	1	1	1	2	1	2	2	0
4	2	2	2	2	2	2	2	1	0
5	2	2	1	2	2	2	2	2	0
6	1	1	1	1	2	1	2	2	1
7	2	2	2	2	1	1	2	2	0
8	1	1	1	1	2	2	2	1	0
9	2	2	1	1	2	2	2	2	1
10	2	2	2	2	2	2	2	2	1

ID= subject's identification variable

Item001,…, Item008= 8 items used to measure the latent class variable

Z= the distal outcome (in this case a COUNT distal outcome)

5.3.2 Example Syntax

Include a "libname" statement prior to running the macro to direct SAS to the data file.

```
libname sasf "S:\myfolder\";
```

Note: we suppose that the SAS data set exists in the folder S:\myfolder\. This path represents any user-specified folder.

Estimate the LCA model using PROC LCA.

```
PROC LCA DATA = SimData_Count OUTPARAM = Count_param OUTPOST =
Count_post;
    ID id;
    NCLASS 5;
    ITEMS item001-item008;
    CATEGORIES 2 2 2 2 2 2 2 2;
    SEED 12345;
    RHO PRIOR = 1;
    NSTARTS 20;
    MAXITER 5000;
    CRITERION 0.000001;
RUN;
```

Then, call the macro.

```
%LCA_Distal_BCH(input_data = SimData_Count,
                param = Count_param,
                post = Count_post,
                id=id,
                distal = z,
                metric = Count );
```

The `input_data` argument identifies the data file. The `param` argument directs the macro to the parameters generated in the OUTPARAM file generated by PROC LCA. The `id` and `distal` arguments identify the subject identification variable and the distal outcome. The `metric` argument indicates that the distal outcome is continuous, and `output_dataset_name` names the macro's output.

In this example there were no survey weights. If there had been, it would be necessary to add a line such as

```
WEIGHT SurveyWeight;
```

to the PROC LCA syntax and a line such as

```
sampling_weight=SurveyWeight,
```

to the macro syntax.

5.3.3 Example Output

The first table shows the estimated distal outcome means within each class.

Count dataset
BCH Estimation of Means of z by Latent Class

Estimates using BCH Modal Weighting

CLASS	DISTAL_MEAN	DISTAL_STD_ERROR_FOR_MEAN	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.77210	0.12490	0.52729	1.01690
2	0.40113	0.05298	0.29729	0.50497
3	0.75902	0.08292	0.59649	0.92154
4	0.87883	0.11805	0.64745	1.11021
5	1.24941	0.18330	0.89014	1.60869

The second shows tests of equality of the means between different classes.

Count dataset
BCH Estimation of Means of z by Latent Class

Wald Chi-Squared Tests

NAME	ESTIMATE	STD_ERROR	WALD_STATISTIC	DF	P_VALUE
Difference in Log Means: Class 2 versus Class 1	-0.65483	0.22496	8.4735	1	0.00360
Difference in Log Means: Class 3 versus Class 1	-0.01709	0.19479	0.0077	1	0.93010
Difference in Log Means: Class 4 versus Class 1	0.12948	0.21294	0.3698	1	0.54314
Difference in Log Means: Class 5 versus Class 1	0.48132	0.21819	4.8660	1	0.02739
Difference in Log Means: Class 3 versus Class 2	0.63775	0.17409	13.4203	1	0.00025
Difference in Log Means: Class 4 versus Class 2	0.78431	0.18845	17.3224	1	0.00003
Difference in Log Means: Class 5 versus Class 2	1.13615	0.19739	33.1302	1	0.00000
Difference in Log Means: Class 4 versus Class 3	0.14657	0.18076	0.6575	1	0.41746
Difference in Log Means: Class 5 versus Class 3	0.49840	0.18245	7.4626	1	0.00630
Difference in Log Means: Class 5 versus Class 4	0.35184	0.20934	2.8247	1	0.09282
Omnibus Test	.	.	37.3002	4	0.00000

These output tables are also generated as datasets, namely distal_estimates and distal_tests.

5.3.4 Overall Response Means

When interpreting the estimated response means within each of the latent classes, it may be useful to compare them to the overall estimated response mean, ignoring latent class. This can be accomplished in the usual way using PROC MEANS (if survey weights are not used) or by using PROC SURVEYMEANS with the `weight` statement (if survey weights are being used). This is described further in the corresponding subsection for continuous outcomes; it is not necessary to specify to PROC MEANS or PROC SURVEYMEANS that the response is count rather than continuous.

5.4 Estimating a Categorical Distal Outcome

Before attempting to complete the following example, please download the file *%LCA_Distal Examples* from the %LCA_Distal macros download page.

5.4.1 Example Data

First, we will examine the structure of the database and the variables to be analyzed. Below are the first 10 observations from the SAS data set **simdata_categ.sas7bdat**, which is contained in the *%LCA_Distal Examples* file available at http://methodology.psu.edu

ID	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	Z
1	2	2	2	2	1	2	2	2	2
2	2	1	2	2	2	2	2	2	3
3	1	1	2	1	1	2	2	1	2
4	2	1	1	1	2	1	2	2	3
5	2	2	2	1	2	2	2	2	1
6	2	2	2	2	1	2	2	2	1
7	1	1	1	1	1	2	2	2	2
8	2	2	2	2	2	2	2	1	3
9	2	2	2	2	2	2	1	2	2
10	2	2	2	1	1	1	2	1	1

ID= subject's identification variable

Item001,…, Item008= 8 items used to measure the latent class variable

Z= the distal outcome (Note: The categorical distal outcome should be coded using 1, 2, 3, …, g, where g = the number of categories.)

5.4.2 Example Syntax

Once the LCA model has been identified, estimate the LCA model using PROC LCA.

```
PROC LCA DATA = SimData_Categ OUTPARAM = Categ_param OUTPOST =
Categ_post;
    ID id;
    NCLASS 5;
    ITEMS item001-item008;
    CATEGORIES 2 2 2 2 2 2 2 2;
    SEED 12345;
    RHO PRIOR = 1;
    NSTARTS 20;
```

```
        MAXITER 5000;
        CRITERION 0.000001;
RUN;
```

The output is described in the *PROC LCA & PROC LTA Users' Guide*.

Then, include and run the macro.

```
%LCA_Distal_BCH(input_data = SimData_Categ,
        param = Categ_param,
        post = Categ_post,
        id = id,
        distal = z,
        metric = categorical );
```

The `input_data` argument identifies the data file. The `param` argument directs the macro to the parameters in the OUTPARAM file generated by PROC LCA. The `id` and `distal` arguments identifies the subject identification and distal outcome variable in the data set. The `metric` argument indicates that the distal outcome is categorical, and `output_dataset_name` names the macro's output.

In this example there were no survey weights. If there had been, it would be necessary to add a line such as

```
WEIGHT SurveyWeight;
```

to the PROC LCA syntax, and a line such as

```
sampling_weight=SurveyWeight,
```

to the macro syntax.

5.4.3 Example Output

The onscreen output contains the estimated proportions of each response category within each latent class.

Categorical dataset
BCH Estimation of Proportions of z by Latent Class

Estimates using BCH Modal Weighting

CLASS	OUTCOME_CATEGORY	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB
1	1	0.30825	0.08871
1	2	0.42488	0.09390
1	3	0.26688	0.08444
2	1	0.30271	0.03687
2	2	0.30822	0.03722
2	3	0.38907	0.03864
3	1	0.36015	0.04536
3	2	0.42180	0.04636
3	3	0.21804	0.04070
4	1	0.36016	0.10002
4	2	0.33650	0.10027
4	3	0.30333	0.09573
5	1	0.37273	0.07092
5	2	0.23640	0.06478
5	3	0.39087	0.07097

Categorical dataset
BCH Estimation of Proportions of z by Latent Class

Wald Chi-Squared Tests

NAME	WALD_STATISTIC	DF	P_VALUE
Omnibus Test	10.0879	8	0.25891

The contents of the output are stored in the distal_estimates and distal_tests datasets, respectively.

5.4.4 Overall Response Proportions

When interpreting the estimated response proportions within each of the latent classes, it may be useful to compare them to the overall estimated response proportion, ignoring latent class. This can be accomplished in the usual way using PROC FREQ (if survey weights are not used) or by using PROC SURVEYFREQS with the `weight` statement (if survey weights are being used). This is described further in the corresponding subsection for binary outcomes; it is not necessary to specify to PROC FREQ or PROC SURVEYFREQ that the response is count rather than continuous.

6 Demonstration of the %LCA_Distal_BCH Macro for Multiple Groups

In this section, we first describe the structure of the data sets and the variables to be analyzed when there are multiple groups. Then, we illustrate how to estimate the distribution of the distal outcome within each latent class using the %LCA_Distal_BCH macro and describe the output of the macro. This section describes use of the macro with a binary distal outcome. The results with other outcomes are very similar. Before attempting to complete the following example, please download the file *%LCA_Distal Examples* from the %LCA_Distal macros download page. Also, verify that you are running PROC LCA v.1.3.2 or higher.

6.1 Example Data

Below are 10 putative observations from the SAS data set **simdata_binary_group.sas7bdat**, which is contained in the *%LCA_Distal Examples* file available at http://methodology.psu.edu.

ID	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	Z	Educ
1	2	2	1	2	2	2	2	2	1	1
2	1	1	2	2	2	2	2	2	0	1
3	2	1	2	1	1	1	1	1	0	1
4	2	2	2	2	2	2	2	2	1	2
5	2	2	2	2	2	2	2	2	1	2
6	1	1	1	2	2	2	2	2	1	2
7	2	2	1	2	2	2	2	2	1	3
8	2	2	2	2	2	2	2	2	1	3
9	2	2	2	2	2	2	2	2	1	3
10	2	2	2	2	1	2	2	2	1	3

ID= subject's identification variable,

Item001,…, Item008= 8 items used to measure the latent class variable,

Z= the distal outcome (Note: binary distal outcome should be coded using 0s and 1s.)

Educ=the variable for multiple groups.

6.2 Example Syntax

Include a “libname” statement prior to running the macro to direct SAS to the data file.

```
libname sasf "S:\myfolder\";
```

Note: we suppose that the SAS data set exists in the folder `S:\myfolder\`. This path represents any user-specified folder.

Once the LCA model has been identified, estimate the LCA model including the distal outcome *Z* as a covariate and *Educ* as the grouping variable using PROC LCA.

```
PROC LCA DATA = simdata_Binary_group OUTPARAM = Binary_param OUTPOST
= Binary_post ;
    ID id;
    NCLASS 5;
    ITEMS item001-item008;
    CATEGORIES 2 2 2 2 2 2 2 2;
    SEED 12345;
    RHO PRIOR = 1;
    NSTARTS 20;
    GROUP educ;
    MAXITER 5000;
    CRITERION 0.000001;
RUN;
```

The output is described in the *PROC LCA & PROC LTA Users’ Guide*.

The output will also include the files Binary_param and Binary_post in the WORK directory.

	Parameter Type	Group Number	Variable Name	Response Category	Estimate - Latent Class 1	Estimate - Latent Class 2	Estimate - Latent Class 3	Estimate - Latent Class 4	Estimate - Latent Class 5
1	GAMMA	1		.	0.103589	0.280180	0.158494	0.327222	0.130514
2	GAMMA	2		.	0.103589	0.280180	0.158494	0.327222	0.130514
3	GAMMA	3		.	0.103589	0.280180	0.158494	0.327222	0.130514
4	RHO	1	Item001	1	0.932447	0.906181	0.130193	0.058604	0.977193
5	RHO	1	Item002	1	0.872605	0.880844	0.136937	0.070370	0.827092
6	RHO	1	Item003	1	0.853765	0.849715	0.134888	0.081468	0.960024
7	RHO	1	Item004	1	0.947957	0.882878	0.048055	0.080657	0.875625
8	RHO	1	Item005	1	0.933083	0.089037	0.711402	0.081087	0.680476
9	RHO	1	Item006	1	0.859787	0.002642	0.798495	0.015504	0.966975
10	RHO	1	Item007	1	0.919828	0.063420	0.093312	0.121607	0.001692
11	RHO	1	Item008	1	0.771090	0.154943	0.092573	0.118761	0.084783
12	RHO	1	Item001	2	0.067553	0.093819	0.869807	0.941396	0.022807
13	RHO	1	Item002	2	0.127395	0.119156	0.863063	0.929630	0.172908
14	RHO	1	Item003	2	0.146235	0.150285	0.865112	0.918532	0.039976
15	RHO	1	Item004	2	0.052043	0.117122	0.951945	0.919343	0.124375
16	RHO	1	Item005	2	0.066917	0.910963	0.288598	0.918913	0.319524
17	RHO	1	Item006	2	0.140213	0.997358	0.201505	0.984496	0.033025
18	RHO	1	Item007	2	0.080172	0.936580	0.906688	0.878393	0.998308
19	RHO	1	Item008	2	0.228910	0.845057	0.907427	0.881239	0.915217
20	RHO	2	Item001	1	0.932447	0.906181	0.130193	0.058604	0.977193
21	RHO	2	Item002	1	0.872605	0.880844	0.136937	0.070370	0.827092
22	RHO	2	Item003	1	0.853765	0.849715	0.134888	0.081468	0.960024
23	RHO	2	Item004	1	0.947957	0.882878	0.048055	0.080657	0.875625
24	RHO	2	Item005	1	0.933083	0.089037	0.711402	0.081087	0.680476
25	RHO	2	Item006	1	0.859787	0.002642	0.798495	0.015504	0.966975
26	RHO	2	Item007	1	0.919828	0.063420	0.093312	0.121607	0.001692
27	RHO	2	Item008	1	0.771090	0.154943	0.092573	0.118761	0.084783
28	RHO	2	Item001	2	0.067553	0.093819	0.869807	0.941396	0.022807
29	RHO	2	Item002	2	0.127395	0.119156	0.863063	0.929630	0.172908
30	RHO	2	Item003	2	0.146235	0.150285	0.865112	0.918532	0.039976
31	RHO	2	Item004	2	0.052043	0.117122	0.951945	0.919343	0.124375
32	RHO	2	Item005	2	0.066917	0.910963	0.288598	0.918913	0.319524
33	RHO	2	Item006	2	0.140213	0.997358	0.201505	0.984496	0.033025
34	RHO	2	Item007	2	0.080172	0.936580	0.906688	0.878393	0.998308

Binary_param

	id	Item001	Item002	Item003	Item004	Item005	Item006	Item007	Item008	educ	Latent Class 1 Posterior Prob	Latent Class 2 Posterior Prob	Latent Class 3 Posterior Prob	Latent Class 4 Posterior Prob	Latent Class 5 Posterior Prob	BEST	z
1	1	2	2	1	2	2	2	2	2	1	0.000000	0.014031	0.045757	0.940174	0.000037	4	1
2	2	1	1	2	2	2	2	2	2	1	0.000029	0.755819	0.029729	0.213072	0.001350	2	0
3	3	2	1	2	1	1	1	1	1	1	0.991977	0.000015	0.007931	0.000060	0.000017	1	0
4	4	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
5	5	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
6	6	1	1	1	2	2	2	2	2	1	0.000040	0.987036	0.001071	0.004365	0.007489	2	1
7	7	2	2	1	2	2	2	2	2	1	0.000000	0.014031	0.045757	0.940174	0.000037	4	1
8	8	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
9	9	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
10	10	2	2	2	2	1	2	2	2	1	0.000001	0.000146	0.436037	0.563814	0.000002	4	1
11	11	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
12	12	2	2	1	2	2	2	2	2	1	0.000000	0.014031	0.045757	0.940174	0.000037	4	1
13	13	1	1	1	1	1	1	2	2	1	0.018478	0.000575	0.000157	0.000000	0.980790	5	1
14	14	2	2	2	2	2	2	1	2	1	0.000001	0.000112	0.020163	0.979724	0.000000	4	1
15	15	1	1	1	2	2	2	2	2	1	0.000040	0.987036	0.001071	0.004365	0.007489	2	1
16	16	1	2	2	2	2	2	2	2	1	0.000001	0.032931	0.060351	0.906625	0.000091	4	1
17	17	2	2	2	2	1	1	2	2	1	0.000002	0.000000	0.994852	0.005112	0.000033	3	0
18	18	1	1	2	1	2	2	2	1	1	0.001723	0.994893	0.000146	0.002399	0.000839	2	1
19	19	1	1	1	1	1	1	1	2	1	0.991966	0.000182	0.000076	0.000000	0.007776	1	1
20	20	2	2	2	2	1	1	2	2	1	0.000002	0.000000	0.994852	0.005112	0.000033	3	1
21	21	1	1	2	2	1	2	2	1	1	0.054957	0.537354	0.296598	0.100524	0.010567	2	0
22	22	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
23	23	2	2	2	2	2	2	2	2	1	0.000000	0.000228	0.026933	0.972839	0.000000	4	1
24	24	1	1	2	1	2	2	2	2	1	0.000094	0.994721	0.000262	0.003264	0.001660	2	1
25	25	1	1	1	1	2	2	2	2	1	0.000097	0.992810	0.000007	0.000051	0.007035	2	0
26	26	1	2	1	1	2	2	2	2	1	0.000103	0.983844	0.000333	0.004945	0.010774	2	1
27	27	1	1	1	1	1	1	1	1	1	0.999772	0.000010	0.000002	0.000000	0.000216	1	0
28	28	1	1	1	1	2	2	2	2	1	0.000097	0.992810	0.000007	0.000051	0.007035	2	1
29	29	2	2	2	2	2	2	1	2	1	0.000001	0.000112	0.020163	0.979724	0.000000	4	0
30	30	1	1	1	1	2	1	2	2	1	0.002833	0.012567	0.000137	0.000004	0.984460	5	1
31	31	1	1	1	1	2	2	1	2	1	0.016224	0.983487	0.000011	0.000103	0.000174	2	1
32	32	1	1	1	1	1	1	1	1	1	0.999772	0.000010	0.000002	0.000000	0.000216	1	0
33	33	1	1	1	2	2	1	2	2	1	0.001077	0.011545	0.018735	0.000304	0.968339	5	1
34	34	2	2	1	1	2	2	2	2	1	0.000041	0.554246	0.012104	0.432241	0.001368	2	1
35	35	2	2	2	2	1	1	2	2	1	0.000002	0.000000	0.994852	0.005112	0.000033	3	0
36	36	1	2	1	1	1	1	2	2	1	0.012919	0.000372	0.004754	0.000010	0.981945	5	1
37	37	1	1	1	2	2	2	2	2	1	0.000040	0.987036	0.001071	0.004365	0.007489	2	1
38	38	1	1	1	1	2	2	2	2	1	0.000097	0.992810	0.000007	0.000051	0.007035	2	1
39	39	2	2	2	2	1	2	2	2	1	0.000001	0.000146	0.436037	0.563814	0.000002	4	1

Binary_post

Now, include and run the macro:

```
%LCA_Distal_BCH(input_data = simdata_Binary_group,
     param = Binary_param,
      post = Binary_post,
      id = id,
      group = educ,
      distal = z,
      metric = Binary );
```

The `input_data` argument identifies the data file. The `param` argument directs the macro to the parameters in the OUTPARAM file generated by PROC LCA. The `id` and `distal` argument identify the subject identification variable and distal outcome variable in the data set. The `group` argument identifies the variable for multiple groups. The `metric` argument indicates that the distal outcome is binary.

In this example there were no survey weights. If there had been, it would be necessary to add a line such as

```
WEIGHT SurveyWeight;
```

to the PROC LCA syntax, and a line such as

```
sampling_weight=SurveyWeight,
```

to the macro syntax.

6.2.1 Example Output

Below is the onscreen output for the first group on the `educ` variable. Similar output follows for the second and third groups.

Binary grouped dataset
BCH Estimation of Proportions of z by Latent Class
For Group 1 of 3 on Variable educ
Estimates using BCH Modal Weighting

CLASS	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.53702	0.073266	0.39342	0.68062
2	0.82074	0.033527	0.75503	0.88645
3	0.77324	0.053807	0.66778	0.87870
4	0.92123	0.024249	0.87371	0.96876
5	0.69274	0.061255	0.57268	0.81281

Binary grouped dataset
BCH Estimation of Proportions of z by Latent Class
For Group 1 of 3 on Variable educ
Confidence Intervals for Probabilities

CLASS	DISTAL_PROBABILITY	DISTAL_LOG_ODDS	CI95_LOWER_LOG_ODDS	CI95_UPPER_LOG_ODDS	CI95_LOWER_PROBABILITY	CI95_UPPER_PROBABILITY
1	0.53702	0.14836	-0.42922	0.72593	0.39431	0.67391
2	0.82074	1.52137	1.07473	1.96802	0.74550	0.87740
3	0.77324	1.22670	0.62523	1.82818	0.65141	0.86154
4	0.92123	2.45921	1.80423	3.11419	0.85866	0.95747
5	0.69274	0.81298	0.24892	1.37705	0.56191	0.79852

Binary grouped dataset
BCH Estimation of Proportions of z by Latent Class
For Group 1 of 3 on Variable educ
Wald Chi-Squared Tests

NAME	ESTIMATE	STD_ERROR	WALD_STATISTIC	DF	P_VALUE
Difference in Log Odds: Class 2 versus Class 1	1.37301	0.37516	13.3940	1	0.00025
Difference in Log Odds: Class 3 versus Class 1	1.07835	0.42770	6.3567	1	0.01169
Difference in Log Odds: Class 4 versus Class 1	2.31085	0.44519	26.9436	1	0.00000
Difference in Log Odds: Class 5 versus Class 1	0.66462	0.42545	2.4403	1	0.11825
Difference in Log Odds: Class 3 versus Class 2	-0.29467	0.38342	0.5906	1	0.44218
Difference in Log Odds: Class 4 versus Class 2	0.93784	0.40823	5.2777	1	0.02160
Difference in Log Odds: Class 5 versus Class 2	-0.70839	0.37206	3.6250	1	0.05692
Difference in Log Odds: Class 4 versus Class 3	1.23251	0.49528	6.1927	1	0.01283
Difference in Log Odds: Class 5 versus Class 3	-0.41372	0.42683	0.9395	1	0.33240
Difference in Log Odds: Class 5 versus Class 4	-1.64623	0.43989	14.0056	1	0.00018
Omnibus Test	.	.	31.9581	4	0.00000

7 Demonstration of Assignment and Adjustment Options

Shown here are four different approaches to distal outcome analysis for the binary example. All give roughly similar answers in this example. Simulation studies suggest that the BCH answers may be more accurate than the unadjusted answers (see Chapter 3).

```
TITLE "Modal unadjusted";
%LCA_Distal_BCH(input_data = SimData_Binary,
            param = Binary_param,
            post = Binary_post,
            distal = z,
            id = id,
            metric = binary ,
            adjustment_method = unadjusted,
            assignment = modal);
```

Estimates using Unadjusted Modal Weighting

CLASS	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.54167	0.071990	0.40057	0.68277
2	0.81944	0.032086	0.75656	0.88233
3	0.78082	0.048467	0.68583	0.87582
4	0.91071	0.022022	0.86755	0.95388
5	0.68657	0.056730	0.57538	0.79776

```
TITLE "Proportional unadjusted";
%LCA_Distal_BCH(input_data = SimData_Binary,
            param = Binary_param,
            post = Binary_post,
            distal = z,
            id = id,
            metric = binary ,
            adjustment_method = unadjusted,
            assignment = proportional);
```

Estimates using Unadjusted Proportional Weighting

CLASS	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.56928	0.066427	0.43908	0.69947
2	0.81968	0.031650	0.75764	0.88171
3	0.78363	0.042472	0.70039	0.86688
4	0.91375	0.020210	0.87414	0.95336
5	0.67557	0.055833	0.56614	0.78500

```
TITLE "Modal BCH";
%LCA_Distal_BCH(input_data = SimData_Binary,
            param = Binary_param,
            post = Binary_post,
            distal = z,
            id = id,
            metric = binary ,
            adjustment_method = BCH,
            assignment = modal);
```

Estimates using BCH Modal Weighting

CLASS	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.53702	0.073266	0.39342	0.68062
2	0.82074	0.033527	0.75503	0.88645
3	0.77324	0.053807	0.66778	0.87870
4	0.92123	0.024249	0.87371	0.96876
5	0.69274	0.061255	0.57268	0.81281

```
TITLE "Proportional BCH";
%LCA_Distal_BCH(input_data = SimData_Binary,
                  param = Binary_param,
                  post = Binary_post,
                  distal = z,
                  id = id,
                  metric = binary,
                  adjustment_method = BCH,
                  assignment = proportional);
```

Proportional BCH
BCH Estimation of Proportions of z by Latent Class

Estimates using BCH Proportional Weighting

CLASS	DISTAL_PROB	DISTAL_STD_ERROR_FOR_PROB	DISTAL_CI95_LOWER	DISTAL_CI95_UPPER
1	0.55589	0.072162	0.41445	0.69733
2	0.82212	0.033208	0.75704	0.88721
3	0.76039	0.052412	0.65766	0.86311
4	0.92807	0.022392	0.88418	0.97196
5	0.67326	0.061736	0.55226	0.79426

References

Bakk, Z., Oberski, D. L., & Vermunt, J. K. (2014). Relating latent class assignments to external variables: Standard errors for corrected inference. *Political Analysis, 22*, 520-540. doi:10.1093/pan/mpu003

Bakk, Z., & Vermunt, J. K. (2016). Robustness of stepwise latent class modeling with continuous distal outcomes. *Structural Equation Modeling, 23,* 20-31*.* doi:10.1080/10705511.2014.955104

Bolck, A., Croon, M., & Hagenaars, J. (2004). Estimating latent structure models with categorical variables: One-step versus three-step estimators. *Political Analysis*, *12*(1), 3–27.

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.

Dziak, J. J., Bray, B. C., Zhang, J. - T., Zhang, M., & Lanza, S. T. (2016). Comparing the performance of improved classify-analyze approaches in latent profile analysis. Methodology: *European Journal of Research Methods for the Behavioral and Social Sciences, 12,* 107-116. http://doi.org/10.1027/1614-2241/a000114

Lanza, S. T., Dziak, J. J., Huang, L., Wagner, A. T., & Collins, L. M. (2015). *Proc LCA & Proc LTA users' guide* (Version 1.3.2). University Park: The Methodology Center, Penn State. Available from methodology.psu.edu

Lanza, S. T., Tan, X., & Bray, B. C. (2013). Latent class analysis with distal outcomes: A flexible model-based approach. *Structural Equation Modeling: A Multidisciplinary Journal, 20*, 1-20.

Schafer, J. L. (1997) *Analysis of incomplete multivariate data*. Boca Raton, FL: Chapman and Hall/CRC.

Vermunt, J. K. (2010). Latent class modeling with covariates: Two improved three-step approaches. *Political Analysis, 18,* 450–469.

Vermunt, J. K., & Magidson, J. (2015). *Upgrade manual for Latent GOLD 5.1*. Belmont, MA: Statistical Innovations, Inc.