



**PennState**

**The Methodology Center**  
advancing methods, improving health

---

# %LCA\_Covariates\_3Step SAS Macro Users' Guide (Version 1.0)

---

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

Copyright © 2020 The Pennsylvania State University  
ALL RIGHTS RESERVED

The development of the SAS %LCA\_Covariates\_3Step macro was supported by National Institute on Drug Abuse Grants P50 DA10075 and P50 DA039838.

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

Dziak, J. J., Bray, B. C., & Wagner, A. T. (2020). *LCA\_Covariates\_3Step SAS macro users' guide* (Version 1.0). University Park, PA: The Methodology Center, Penn State. Retrieved from <http://methodology.psu.edu>

## Contents

<b>1</b>	<b>About the %LCA_Covariates_3Step Macro .....</b>	<b>2</b>
<b>2</b>	<b>System Requirements.....</b>	<b>3</b>
<b>3</b>	<b>The BCH Approach to Covariates .....</b>	<b>4</b>
<b>4</b>	<b>Using the %LCA_Covariates_3Step Macro .....</b>	<b>7</b>
4.1	<i>Argument Definitions .....</i>	<i>7</i>
4.2	<i>Preparation .....</i>	<i>8</i>
4.3	<i>Estimation of the Latent Class Model in PROC LCA .....</i>	<i>8</i>
4.4	<i>Macro Syntax and Input.....</i>	<i>9</i>
4.5	<i>Output .....</i>	<i>9</i>
<b>5</b>	<b>Demonstration of the %LCA_Covariates_3Step Macro .....</b>	<b>11</b>
5.1	<i>Example without a Grouping Variable .....</i>	<i>11</i>
5.2	<i>Example with a Grouping Variable.....</i>	<i>15</i>
<b>6</b>	<b>References .....</b>	<b>17</b>

# 1 About the %LCA\_Covariates\_3Step Macro

The SAS %LCA\_Covariates\_3Step macro estimates the association between covariates and latent class membership using the approach of Bolck, Croon, and Hagenaars (2004), as adapted by Vermunt (2010) and Vermunt and Magidson (2015). It is a “three-step” (noninclusive) approach, which can be more flexible and robust than the “one-step” (inclusive) approach implemented via the COVARIATES statement that is native to PROC LCA (Lanza et al., 2015). It is therefore offered as an alternative to the COVARIATES statement in PROC LCA. The %LCA\_Covariates\_3Step macro is designed to work with SAS Version 9.1 or higher and PROC LCA.

**NOTE: This macro does not estimate covariates for LCA with a distal outcome.** Two of our macros employ the approach developed by Bolck, Croon, and Hagenaar: the %LCA\_Covariates\_3Step macro for incorporating covariates in a latent class model and the %LCA\_Distal\_BCH macro for estimating the association between a latent class variable and a distal outcome. These two macros are not designed to work together. A combined SAS macro supporting both covariates and distal outcomes in a single model is not currently available.

The %LCA\_Covariates\_3Step macro

- uses simple, minimal syntax;
- estimates regression coefficients and odds ratios for predicting class membership from covariates, as an alternative to in the functionality that is native to PROC LCA;
- provides confidence intervals for regression coefficients; and
- can accommodate separate regressions in 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 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>.

## 2 System Requirements

The %LCA\_Covariates\_3Step 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 Covariates

PROC LCA allows covariates to be included and tested as predictors of latent class membership, using an “inclusive” (one-step) approach in which the meaning of the classes (as determined by the class-specific response probabilities to each categorical indicator item) is estimated at the same time as the coefficients. However, since this functionality was developed, some experts have described the advantages of a non-inclusive method in which the LCA model is first fit without covariates, in order to try to determine more stable and interpretable meanings for the classes without depending on the covariates, and then the covariates are added to the model later. A naïve way to do this would be to treat class membership as a known categorical variable and fit a multinomial logistic regression afterwards with the covariates. However, such an approach would cause bias and inaccurate standard errors because it does not take class-membership uncertainty into account. The BCH method largely overcomes this difficulty by constructing special weights and using them to adjust for class membership uncertainty.

The BCH method of estimating covariates is a “three-step” method. This means that (1) the parameters of the LCA model are estimated without the covariates, 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 estimate multinomial logistic regression coefficients for predicting membership in 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 account for 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 three-step regression coefficient estimates for predictors of class membership, 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 extremely large and uninterpretable numbers for the regression coefficients); 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  $\mathbf{Y} = Y_1, \dots, Y_m$ , without including the covariates in the model. This will provide posterior probabilities of class membership,  $\omega_{ic} = P(C = c | Y = y_i)$ , for each individual  $i = 1, \dots, N$  in the dataset and each class  $c = 1, \dots, n_c$ .

**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  $w_{ic} = 1$  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  $w_{ic} = \omega_{ic}$  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  $\mathbf{D}$ . The entry in row  $a$  and column  $b$  of  $\mathbf{D}$  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  $\gamma_a$  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 regression coefficients for predicting latent class membership from the covariates, by doing a multinomial logistic regression on 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).

**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\_Covariates\_3Step Macro

### 4.1 Argument Definitions

Table 1. Argument Definitions for the %LCA\_Covariates\_3Step Macro.

Argument	Required	Description
input_data	Yes	Input data set. The ID variable and covariates must all be included among the variables.
covariates	Yes	The name of the covariates to be used in the multinomial regression to predict class membership.
id	Yes	The name of the subject identification variable used in the datasets specified in input_data and post.
postprobs	Yes	Name of the data set generated by PROC LCA as the OUTPOST output when analyzing the input_data dataset. The data set contains estimates of the posterior probabilities. The ID variable, along with the covariates, must be included in the ID statement of PROC LCA. <b>The model itself must not use the COVARIATES statement in PROC LCA.</b>
groups	No	The name of the variable, if any, by which analyses should be done separately by category. If specified, it must also be used in the GROUPS statement in LCA.
adjustment	No	The method, if any, of adjusting the class membership weights for the possibility of misclassification. This may be "BCH" (default, recommended) or "unadjusted," without quotes.
assignment	No	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.
ref_class	No	The reference class for the multinomial regression; this class's regression parameters are not estimated because of identifiability constraints. All other classes are compared to this one. The default is 1 as in PROC_LCA.
automatically_add_intercept	No	Whether to automatically include an intercept column (1) or not (0). The default is 1.
sampling_weight	No	Name of the variable specifying survey weight. If used, it assumes that WEIGHT has also been used in the previous call to PROC LCA.



## 4.2 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_Covariates_3Step_v10.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\";
```

## 4.3 Estimation of the Latent Class Model in PROC LCA

Use PROC LCA to generate the output needed for use by the %LCA\_Covariates\_3Step 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, Wagner, & Collins, 2015).

Once model selection is complete, generate a file containing the posterior probabilities to be used in the macro by estimating the latent class model with the covariates. This file can be generated using the OUTPOST option in PROC LCA. (See section 5.3 of the *PROC LCA & PROC LTA Users' Guide* for more information.)

**NOTE: The model should not use the COVARIATES statement.**

The PROC LCA syntax will be similar to the following:

```
PROC LCA DATA = my_data OUTPOST = my_post; /* the input data set and
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.

#### 4.4 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_Covariates_3Step_v10(
  input_data = data set name,
  postprobs = name of OUTPOST data set created by PROC LCA,
  covariates = variables,
  id = variable,
  groups = variable,
  sampling_weight = survey weighting variable name,
  adjustment = word describing the misclassification adjustment
method (BCH or unadjusted),
  assignment = word describing the class membership weight
assignment option (modal or proportional),
  ref_class = reference class, default = 1,
  automatically_add_intercept = 1 (for automatically including an
intercept; recommended and default) or 0 (for no-intercept model;
not recommended),
) ;

```

#### 4.5 Output

The macro produces both screen output and SAS datasets. The screen output first presents a table of estimates and standard errors for the the beta coefficients (multinomial logistic regression coefficients) in the model. They are provided for the relationship of each covariate to class membership, separately for each class other than the reference class. The reference

class is considered to have coefficients of zero for model identifiability, and it is the baseline to which the other classes are compared. Next, a table of approximate 95% confidence intervals for the beta coefficients is shown. Last, the exponentials of the beta coefficients are shown. The exponentiated beta for the intercept of a class is the estimated odds of being in that class, relative to being in the reference class, when the covariates are set to zero. The exponentiated beta for the effect of a covariate is the estimated odds ratio of being in the class rather than the reference class, associated with a one-unit increase in the covariate.

The output is also put in a dataset in SAS's work directory, which can be saved for later analysis. Specifically, the information shown on screen is also included in the dataset `Covariates_Macro_Answers`.

Intermediate results and calculations which may be of technical interest are stored in the datasets `Class_Weights`, `Class_Weights_Long`, `Covariates_Macro_Convergence`, and `Sandwich_Covariance`.

## 5 Demonstration of the %LCA\_Covariates\_3Step Macro

In this section, we first describe the structure of the data sets and the variables to be analyzed. Then, we illustrate how to calculate the BCH-adjusted three-step estimates of the coefficients predicting latent class using the %LCA\_Covariates\_3Step macro. We also describe the resulting output of the macro.

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

### 5.1 Example without a Grouping Variable

#### 5.1.1 Example Data

Below are the first 10 observations from the simulated SAS data set **simple\_example.sas7bdat**, which is available along with the macro.

id	X1	X2	G	I1	I2	I3	I4	I5
1	2.1	0.9	2	2	1	1	1	1
2	0.5	0.2	3	1	1	1	1	1
3	-0.4	1.1	1	2	2	2	2	1
4	-0.2	0.4	2	2	2	2	1	1
5	1.8	-0.9	2	2	2	2	1	1
6	0.7	0.4	1	2	2	2	1	1
7	-1.8	0.3	2	2	2	2	2	2
8	-0.7	-0.3	3	2	2	2	1	2
9	-0.3	0.3	1	1	1	1	2	1
10	-1.6	-0.3	1	2	1	2	2	2

ID = subject's identification variable,

X1, X2 = covariates,

I1, ..., I5 = 8 items used to measure the latent class variable

G = group membership

### 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 but is saved to the posterior probabilities dataset using the ID statement.

```
PROC LCA DATA=sasf.simple_simulated_example
  OUTPOST=Noninclusive_Post ;
  NCLASS 3;
  ITEMS I1 I2 I3 I4 I5;
  CATEGORIES 2 2 2 2 2;
  SEED 11111;
  NSTARTS 5;
  ID id X1 X2;
  RHO prior=1;
  RUN;
```

The output is described in the *PROC LCA & PROC LTA Users' Guide*. It should include the file Noninclusive\_Post in the WORK directory.

id	X1	X2	I1	I2	I3	I4	I5	Latent Class 1 Posterior Prob	Latent Class 2 Posterior Prob	Latent Class 3 Posterior Prob	BEST
1	2.1	0.9	2	1	1	1	1	0.787367	0.212088	0.000545	1
2	0.5	0.2	1	1	1	1	1	0.997839	0.002152	0.00001	1
3	-0.4	1.1	2	2	2	2	1	0.000285	0.423439	0.576276	3
4	-0.2	0.4	2	2	2	1	1	0.000253	0.994102	0.005645	2
5	1.8	-0.9	2	2	2	1	1	0.000253	0.994102	0.005645	2
6	0.7	0.4	2	2	2	1	1	0.000253	0.994102	0.005645	2
7	-1.8	0.3	2	2	2	2	2	0.000013	0.013224	0.986763	3
8	-0.7	-0.3	2	2	2	1	2	0.000273	0.762372	0.237354	2
9	-0.3	0.3	1	1	1	2	1	0.998317	0.000815	0.000868	1
10	-1.6	-0.3	2	1	2	2	2	0.002108	0.009858	0.988035	3

Now the %LCA\_Covariates\_3Step macro can be run. Include the macro and enter the proper syntax in SAS.

```
%LCA_Covariates_3Step(postprobs = Noninclusive_Post,  
  id = id,  
  Covariates = X1 X2);
```

The `postprobs` 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 (do not include the covariates in this line). The `covariates` argument specifies the covariates to be used (do not include the intercept in this line).

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:

## Beta Estimates from LCA\_Covariates\_3Step Macro

CLASS	NAME	BETA	STD_ERR	Z	P
2	_Intercept_	0.07461	0.10817	0.68975	0.24518
2	X1	-0.02756	0.11145	-0.24727	0.40235
2	X2	-0.02721	0.13717	-0.19837	0.42138
3	_Intercept_	-0.66611	0.13581	-4.90477	0.00000
3	X1	-0.19033	0.14549	-1.30823	0.09540
3	X2	-0.09323	0.16637	-0.56038	0.28761

## Approximate 95% Confidence Intervals for Beta Coefficients

CLASS	NAME	BETA	LOWER_CI_BETA	UPPER_CI_BETA
2	_Intercept_	0.07461	-0.13740	0.28661
2	X1	-0.02756	-0.24599	0.19088
2	X2	-0.02721	-0.29606	0.24164
3	_Intercept_	-0.66611	-0.93229	-0.39992
3	X1	-0.19033	-0.47549	0.09482
3	X2	-0.09323	-0.41930	0.23285

## Exponentiated Betas (Odds and Odds Ratios) and Approximate 95% Confidence Intervals

CLASS	NAME	EXP_BETA	LOWER_CI_EXP_BETA	UPPER_CI_EXP_BETA
2	_Intercept_	1.07746	0.87162	1.33191
2	X1	0.97282	0.78193	1.21031
2	X2	0.97316	0.74374	1.27334
3	_Intercept_	0.51370	0.39365	0.67037
3	X1	0.82668	0.62158	1.09947
3	X2	0.91099	0.65750	1.26219

The output is also put in a dataset in SAS's WORK directory, which can be saved for later analysis. Specifically, the information shown on screen is also included in the dataset `Covariates_Macro_Answers`.

## 5.2 Example with a Grouping Variable

Now we suppose that the observed categorical variable `G` will be used as a grouping variable, and that we want to estimate the betas separately within each group. The `GROUPS` statement now needs to be included in the `PROC LCA` call.

```
PROC LCA DATA=sasf.simple_simulated_example
  OUTPOST=Noninclusive_Post ;
  NCLASS 3;
  ITEMS I1 I2 I3 I4 I5;
  CATEGORIES 2 2 2 2 2;
  SEED 11111;
  NSTARTS 5;
  GROUPS G;
  ID id X1 X2;
  RHO prior=1;
RUN;
```

The `groups =` argument also needs to be included in `%LCA_Covariates_3Step`.

```
%LCA_Covariates_3Step(postprobs = Noninclusive_Post,
  id = id,
  groups = G,
  Covariates = X1 X2);
```

The `Noninclusive_Post` that is generated in the WORK directory should include a column "G" between the columns `I5` and `Latent Class 1 Posterior Prob`.

Now, the output tables will include separate results for each group. For example:

Beta Estimates from LCA\_Covariates\_3Step Macro

Group	CLASS	NAME	BETA	STD_ERR	Z	P
1	2	_Intercept_	-0.01522	0.19419	-0.07839	0.46876
1	2	X1	-0.03415	0.19046	-0.17927	0.42886
1	2	X2	-0.07735	0.24877	-0.31093	0.37793
1	3	_Intercept_	-0.69622	0.23609	-2.94897	0.00159
1	3	X1	-0.09844	0.23340	-0.42179	0.33659



<b>Group</b>	<b>CLASS</b>	<b>NAME</b>	<b>BETA</b>	<b>STD_ERR</b>	<b>Z</b>	<b>P</b>
1	3	X2	0.24907	0.30891	0.80628	0.21004
2	2	_Intercept_	0.29927	0.18069	1.65631	0.04883
2	2	X1	0.02578	0.18357	0.14046	0.44415
2	2	X2	-0.04257	0.23297	-0.18272	0.42751
2	3	_Intercept_	-0.68444	0.23541	-2.90744	0.00182
2	3	X1	-0.19625	0.23933	-0.81998	0.20611
2	3	X2	-0.19309	0.28496	-0.67759	0.24902
3	2	_Intercept_	-0.31554	0.19663	-1.60476	0.05427
3	2	X1	-0.00457	0.23042	-0.01982	0.49209
3	2	X2	0.17145	0.23526	0.72879	0.23306
3	3	_Intercept_	-0.37983	0.20198	-1.88052	0.03002
3	3	X1	-0.22690	0.22014	-1.03068	0.15135
3	3	X2	-0.26519	0.24216	-1.09513	0.13673

Group-specific results for each group are also saved in SAS datasets.

## 6 References

- Bolck, A., Croon, M., & Hagnaars, 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.
- 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](http://methodology.psu.edu)
- 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). LG-Syntax™ user's guide: Manual for latent GOLD® 5.0 syntax module November 20, 2015.