Software to Compute Sampling Weights for Multilevel Analysis

Kim Chantala
Dan Blanchette
C. M. Suchindran

Carolina Population Center, UNC at Chapel Hill

Last Update: April 28, 2011

Research in estimating multilevel models (MLM) from complex survey data is quite recent. Not only has this research resulted in several popular software packages incorporating sampling weights into estimating MLM, but it has emphasized an important point often overlooked by both analysts and providers of the survey data: the sampling weights used for multilevel analysis need to be constructed differently than the sampling weights used for single-level analysis. Both the distributors of data and the developers of MLM software packages often leave the user responsible for proper scaling of MLM sampling weights. In addition, the method of scaling can be different for the various MLM software packages. To address this problem, we created Stata and SAS programs for constructing sampling weights for estimating two-level models that can be used with several popular multilevel software packages (i.e. gllamm (Stata), LISREL, MLwiN, Mplus). These programs can be downloaded from our website: http://www.cpc.unc.edu/research/tools/data_analysis/ml_sampling_weights

Introduction

Although commonly accepted methods of computing sampling weights for estimating single-level or population-average models have been developed, no consensus on the best method to construct sampling weights for multilevel analysis has been reached. Because multilevel weights need to be constructed differently than sampling weights used for single-level models, users need to make sure the weights are scaled as recommended by the particular software package being used for MLM analysis. The most common methods for constructing multilevel weights referenced by developers of popular MLM software packages are Pfefferman (1998) and Asparouhov (2004). We have written Stata and SAS programs that use the weight components from each stage of sampling to construct two-level sampling weights using these methods. These programs can be downloaded from our website.

In this manuscript, we first describe the weight components that the user must provide to our programs to create the two-level weights. Next we provide information about several popular multilevel software packages that allow sampling weights to be used in estimation. We conclude with instructions to install and run these programs.

The Weight Components

The weight components provide information about how many members within a sampling unit are represented by each member selected in the sample. A set of weight components is typically computed for each stage of sampling and then used in calculating the weights for fitting both population-average and multilevel models. For example, a two-stage sampling design might first select a sample of schools within the United States, and then select a sample of students within each school. The two weight components computed for this design are:

- Primary Sampling Unit (PSU) or level 2 weight component. Each school in the sample will have a weight equal to the number of schools in the sampling frame represented by that school. It is computed as the inverse of the probability that school j was selected from the sampling frame:

  $$\text{psu\_wt}_j = 1/\{\text{Pr(school j selected}\}.$$

- Final Sampling Unit (FSU) or level 1 weight component. Each student selected from school j will have a sampling weight that is equal to the number of students within school j represented by that student. It is computed as the inverse of the probability that student i was selected given that school j was selected:

  $$\text{fsu\_wt}_{i|j} = 1/\{\text{Pr(student i selected|school j selected)}.$$

Non-response and post-stratification adjustments are often made to the weight components to ensure that they give the best estimate possible. Analysts wanting to construct sampling weights for estimating multilevel models need to have the weight components from each stage of sampling available. If only weights designed for estimating population-average models are available for each stage of sampling, the analyst can estimate the level 1 weight component for a two-stage design as follows. Since the commonly distributed sampling weight used in estimating population-average or single-level models is computed as:

$$W_{i,j} = \text{psu\_wt}_j \times \text{fsu\_wt}_{i|j}$$

analysts can estimate $\text{fsu\_wt}_{i|j}$ by dividing $W_{i,j}$ by psu_wt_j.

Multilevel Software for Analyzing Data from Complex Surveys

In table 1 we have listed popular software packages and provided information on references for methods of constructing multilevel sampling weights referenced by these packages. Most of these software packages (gllamm (Stata), LISREL, MLwiN) reference methods developed by Pfefferman (1998) that use weight components from both stages of sampling to construct level-one and level-two sampling weights. On the other hand, MPLUS uses a method developed by Asparouhov (2004) that combines the weight components from both stages of sampling to create one weight. For most sampling plans, this combined two-level weight will differ from the commonly distributed sampling weight designed for estimating population-average or marginal models.

While PROC MIXED (SAS version 8.2 and later) does allow users to specify a single weight, the weight is *NOT* expected to be a sampling weight, but a weight designed to be inversely proportional to the variability of the observations. Hence, users must be quite cautious in using PROC MIXED when analyzing data collected with a complex sampling plan. SAS also provides a separate package, PROC NLMIXED, for estimating nonlinear multilevel models. Although there is no weight statement available with PROC NLMIXED, special weighting procedures have been implemented through a SAS macro to adjust for the sampling design (Grilli and Pratesi, 2004). We do not provide any programs for creating the multilevel weights needed for PROC MIXED or NLMIXED.

Table 1. Differences in multilevel sampling weights used by popular multilevel software packages.

	Allows MLM Sampling Weights	Method for Scaling MLM Sampling Weights	Responsibility for Scaling MLM Sampling Weights	SAS or Stata program available for download
MPLUS 3.1	Yes	Asparouhov (2004)	User	mpml_wt.sas mpml_wt.do
LISREL 8.7	Yes	Pfefferman (1998)	User	pwigls.sas pwigls.do
GLLAMM (Stata 8)	Yes	Pfefferman (1998)	User	pwigls.sas pwigls.do
MLWIN 1.1	Yes	Pfefferman (1998)	User or MLWIN default	pwigls.sas pwigls.do
MIXED (SAS 8.2)	No[1]	Unknown	User	Not available
NLMIXED (SAS 8.2)	No[2]	Grilli. (2004)	User	Not available

[1] PROC MIXED does have a weight statement, but it is not designed to function as a sampling weight.
[2] PROC NLMIXED does not have a weight statement, but Grilli (2004) has used SAS macros that can include a sampling weight during estimation.

Creating Two-Level Weights

The formulas for constructing the two-level sampling weights developed by Pfefferman (1998) and Asparouhov (2004) are described in Appendix A and have been implemented in SAS and Stata programs that can be downloaded from our website. This section shows how the data file needs to be structured for use by these programs, how to run the programs, and describes the variables created by each program.

Data Preparation

Each program requires the same four variables to compute the two-level sampling weights. These variables are described in table 2. In addition to the two variables for the weight components, a variable identifying the primary sampling unit (PSU) and a variable identifying each final sampling unit (FSU) are needed for computing the two-level sampling weights. Each observation in the data should have a unique value for the fsu_id variable. The values of each fsu_wt variable may or may not be unique. All observations belonging to the same PSU should have the same value for both the psu_id and psu_wt variables.

Table 2. Variables need by the SAS and Stata programs to construct two-level sampling weights.

Level	Variable	Meaning
1	fsu_wt	Within-PSU sampling weight component for each final sampled unit (FSU). The value of fsu_wt indicates how many members of the population belonging to the sampled PSU are represented by the observation.
	fsu_id	FSU Identifier. Every observation in the data should have a unique value for fsu_id.
2	psu_wt	Sampling weight component for primary sampling units (PSU). All observations that are sampled from the same PSU should have the same value of psu_wt. The value of psu_wt indicates how many clusters in the general population are represented by the PSU to which the observation belongs.
	psu_id	PSU Identifier. All observations that are sampled from the same PSU should have the same value of psu_id.

An example of how the data should be organized for using the PWIGLS or MPML_WT programs is shown below:

psu_id	psu_wt	fsu_id	fsu_wt
313	160.4	131	52.3
313	160.4	132	69.0
313	160.4	133	75.1
313	160.4	134	27.6
540	211.9	169	87.6
540	211.9	170	62.3
540	211.9	171	62.3
611	58.1	201	38.3
611	58.1	202	39.1
611	58.1	203	49.7
611	58.1	204	28.9
611	58.1	205	33.5

Using the PWIGLS programs

The SAS and Stata PWIGLS programs will compute two-level sampling weights using both PWIGLS Method 1 and PWIGLS Method 2 described in Appendix A. These programs examine each weight component and print warnings if any weights have missing or zero values. The level-1 weight component is then scaled according to the recommended method from Pfefferman, 1998. In general, the level-2 weight component for the primary sampling unit is used as the level-2 multilevel sampling weight. For sampling plans where the sampling of level 1 units is non-informative while the sampling of level 2 units is informative, weights constructed with PWIGLS Method 1 is an appropriate choice. If the sampling is informative at both levels of sampling then weights constructed with PWIGLS Method 2 is a better choice. A sampling process is informative if the probability of selecting a sampling unit is related to the outcome variable even after conditioning on the model covariates. The analyst should refer to

Pfefferman, 1998 for more information on the PWIGLS methods of weight construction and guidelines for determining which method is best suited for the sampling plan used to collect the data being analyzed.

Stata version of PWIGLS:

The PWIGLS help and ado files can be installed from Stata window by typing:

```
net install pwigls, from (http://www.cpc.unc.edu/restools/data_analysis/ml_sampling_weights/) replace
```

To run the PWIGLS program, the analyst should supply the variables identifying the weight components and identifiers specified in Table 2 as well as names for the four variables that will be computed and added to the data set in memory:

```
pwigls, psu_id(psu_identifier) fsu_id(fsu_identifier)
psu_wt(psu_weight_component) fsu_wt(fsu_weight_component)
psu_m1wt(method_1_level_2_adjust) fsu_m1wt(method_1_level_1_weight)
psu_m2wt(method_2_level_2_adjust) fsu_m2wt(method_2_level_1_weight)
```

The variables specified by psu_m1wt(*varname*) and fsu_m1wt(*varname*) are the adjustment factor and level-1 sampling weights computed with PWIGLS Method 1. The variables specified by psu_m2wt(*varname*) and fsu_m2wt(*varname*) are the adjustment factor and level-1 sampling weights computed with PWIGLS Method 2.

SAS version of PWIGLS:

The SAS PWIGLS program can be downloaded from the link on our webpage. The program can be invoked with the following statements:

```
%include '/the_directory/pwigls.sas';

%pwigls(input_set=input_data_set_name,
      psu_id=psu_identifier,
      psu_wt=psu_weight_component,
      fsu_id=fsu_identifier,
      fsu_wt=fsu_weight_component,
      output_set=output_set_name,
      psu_m1wt = method_1_level_2_adjust,
      fsu_m1wt = method_1_level_1_weight,
      psu_m2wt = method_2_level_2_adjust,
      fsu_m2wt = method_1_level_1_weight,
      replace=replace);
```

The %INCLUDE statement identifies the name of the file containing the PWIGLS macro and its location. The %PWIGLS command invokes the procedure. In addition to specifying the variables identifying the weight components and identifiers specified in Table 2, the analyst must specify the input data set containing these variables (input_set=*input_data_set_name*), the name of the output data set (output_set=*output_data_set_name*) where the computed multilevel weights will be stored, variable names for the PWIGLS Method 1 adjustment factor and level-one sampling weights (psu_m1wt = *method_1_level_2_adjust*, fsu_m1wt = *method_1_level_2_weight*,) and variable names for the PWIGLS Method 2 adjustment factor and level-one sampling weights (psu_m2wt = *method_2_level_2_adjust*, fsu_m2wt = *method_2_level_1_weight*). The option replace=replace will replace the specified output data set if it exists.

Using the MPML_WT programs

The SAS and Stata MPML_WT programs will compute a composite weight designed for estimating two-level models using the MPML Method A of weight construction described in Appendix A. This method is designed for sampling strategies that use an informative sampling plan for selecting units at both level 1 and level 2. The MPML METHOD AI composite weight described in Appendix A is appropriate for data collected with sampling plans that are invariant across the clusters (the level 2 units) and is identical to the sampling weight used in estimating population-average models. Because this weight is commonly distributed with data from population surveys, no software program is provided for computing this weight. See Asparouhov, T. (2004) for more information. If the analyst is uncertain of the method used to collect the data being analyzed, then a prudent choice would be to assume an informative sampling plan used to select units at both levels and use the MPML_WT program to compute a composite two-level weight using MPML Method A.

Stata version of MPML_WT:

The mpml_wt help and ado files can be installed from Stata window by typing:

```
net install mpml_wt, from (http://www.cpc.unc.edu/restools/data_analysis/ml_sampling_weights/) replace
```

To run the mpml_wt program, the analyst will need to specify the names of the variables identifying the weight components and identifiers specified in Table 2, plus an additional variable for the composite multilevel sampling weight computed using MPML Method A described in Appendix A:

```
mpml_wt, psu_id(psu_identifier) fsu_id(fsu_identifier)
psu_wt(psu_weight_component) fsu_wt(fsu_weight_component)
mpml_wta(composite_weight)
```

The composite weight variable will be added to the data set in memory.

SAS version of MPML_WT:

The SAS MPML_WT program can be downloaded from the link on our webpage. The program can be invoked with the following statements:

```
%include '/the_directory/mpml_wt.sas';

%mpml_wt(input_set=input_data_set_name,
        psu_id = psu_identifier,
        fsu_id = fsu_identifier,
        psu_wt = psu_weight_component,
        fsu_wt= fsu_weight_component,
        output_set = output_data_set_name,
        mpml_wta = composite_weight,
        replace=replace);
```

The %INCLUDE statement identifies the name of the file containing the MPML_WT macro and its location. The %MPML_WT statement invokes the procedure. In addition to the four variables listed in Table 2, the analyst must specify the data set containing these variables (input_set=*input_data_set_name*), the output data set (output_set=*output_data_set_name*), and a variable name for the composite weight variable (mpml_wta=*composite_weight)*. The option replace=*replace* will replace the specified output data set if it exists.

Appendix A. Commonly Used Methods of Scaling Weights for Two-Level Models

Because no one way of scaling the weights for multilevel modeling has been widely accepted, the method of scaling the weights can be different for different MLM software packages. In this appendix we describe some of the methods most commonly recommended by developers of the MLM software packages. Special macros written in SAS and Stata can be downloaded from our website so users can easily construct the sampling weights needed for estimating multilevel models. Users should check with the user service of the software they are using to ensure that the methods used in these macros are the best method of scaling the multilevel sampling weights for the analytical techniques implemented by the software developers.

PWIGLS scaling method.

This method of scaling is from the following paper: Pfeffermann, D., Skinner, C., Holmes, D. , Goldstein, H. and Rasbash, J. (1998) Weighting for unequal selection probabilities in multilevel models. Journal of the Royal Statistical Society B 60, 23-56. This method of scaling weights presented in this article has recommended by the developers of gllamm and referenced by the developers of LISREL and MLWIN. The formulas below follow these methods and have been used in creating the pwigls.sas and pwigls.do programs.

PWIGLS Method 1:

If the probability of selecting a sampling unit is related to the outcome variable even after conditioning on the model covariates, the sampling process is informative. Analyses using multilevel weights scaled with Method 1 should satisfy the following assumptions concerning informative sampling:

- the sampling of the level 1 units is non-informative
- the sampling of the level 2 units is informative.

To construct the multilevel sampling weights, the an adjustment factor for each PSU is first computed by summing the square of the within-PSU sampling weight for each unit *i* sampled in PSU *j* and then dividing by the sum of the within-PSU sampling weights for each unit *i* sampled in PSU *j*:

$$psu\_m1wt_j = \frac{\sum_i^{n_j} \left(fsu\_wt_{i|j}\right)^2}{\sum_i^{n_j} fsu\_wt_{i|j}} \qquad (1)$$

The level 1 sampling weight for each unit *i* sampled from PSU *j* is computed by dividing the within-PSU sampling weight for unit *i* by the level 2 adjustment factor for PSU j:

$$fsu\_m1wt_{i|j} = \frac{fsu\_wt_{i|j}}{psu\_m1wt_j} \qquad (2)$$

PWIGLS Method 2:

PWIGLS weighted scaling method 2 is recommended for informative sampling methods (Pfefferman, 1998) used for selecting units at both levels of sampling. First an adjustment factor for each PSU is computed by summing the within-PSU sampling weight for each unit *i* sampled in PSU *j* and then dividing by the number sampled within PSU *j*:

$$psu\_m2wt_j = \frac{\sum_i^{n_j} fsu\_wt_{i|j}}{n_j} \qquad (3)$$

The level 1 sampling weight for each unit *i* sampled within PSU *j* is computed by dividing the within-PSU sampling weight for each unit *i* sampled in PSU j by the level 2 adjustment factor:

$$fsu\_m2wt_{i|j} = \frac{fsu\_wt_{i|j}}{psu\_m2wt_j} \qquad (4)$$

MPML Scaling Method

The method of constructing multilevel sampling weights for estimating 2-level models in Mplus is given in: Asparouhov, T. (2004). Weighting for unequal probability of selection in multilevel modeling. Mplus Web Notes: No. 8. Mplus uses sampling weight components from both levels to compute just one weight used in their analysis.

Method AI:

This method is recommended only for data collected with a sampling mechanism that is invariant across the clusters. The multilevel weight computed with method AI is simply the sampling weight used in population-average or single-level models and is the product of level 1 and level 2 weight components:

$$mpmlwtai = fsu\_wt_{i|j} * psu\_wt_j = \text{FINAL\_WT}_{ij} \qquad (5)$$

Since this weight is commonly distributed with data from population surveys, no software program is needed or provided for computing this weight.

Method A:

Method A weight construction involves dividing the product of the level 1 and level 2 weight components by the average of the level 1 weight components for units sampled from cluster j:

$$mpml\_wta_{i,j} = \frac{fsu\_wt_{i|j} * psu\_wt_j}{\left(\frac{\sum_{i}^{n_j} fsu\_wt_{i|j}}{n_j} \right)} \qquad (6)$$

References

Asparouhov, T. (2004). Weighting for Unequal Probability of Selection in Multilevel Modeling, Mplus Web Notes No. 8 available from http://www.statmodel.com/

Grilli, L., and Pratesi, M. (2004) Weighted estimation in multilevel ordinal and binary models in the presence of informative sampling designs. Survey Methodology, Volume 30, 93-103.

Pfeffermann, D., Skinner, C. J., Holmes D. J,and H. Goldstein, Rasbash, J., (1998). Weighting for Unequal Selection Probabilities in Multilevel Models. JRSS, Series B, 60, 123-40.