Select_sites: SAS Macro for Selecting Districts and Schools for Impact Studies in Education

Documentation

February 7, 2023

Azim Shivji
The University of Chicago
Harris School of Public Policy
ashivji@uchicago.edu

Daniel Litwok
Abt Associates
dan_litwok@abtassoc.com

Robert B. Olsen
George Washington Institute of Public Policy
The George Washington University
robolsen@gwu.egu

ACKNOWLEDGEMENTS

The research reported here is supported by the Institute of Education Sciences, US. Department of Education, through Grant R305D190020 to Westat.

SUGGESTED CITATIONS

Shivji, A., Litwok, D., & Olsen, R. (2023). Select_sites: SAS Macro for Selecting Districts and Schools for Impact Studies in Education. https://github.com/select-sites/exval.

LICENSE

Overview

This guide includes documentation for a SAS macro, “select_sites,” developed to support impact evaluations in selecting a sample of districts and schools that represent a study’s target population. Users should run the macro on a school-level data file that nests schools within districts for the entire target population for their study. The macro accommodates stratification and implements one of several different sampling approaches to select districts and schools for an impact study.[1]

Stratification. Stratification ensures adequate representation of different types of districts and/or schools in the study sample. Best practice is to stratify using district- and/or school-level factors that a researcher might hypothesize would moderate impacts (Tipton & Olsen, 2022). The macro gives users the choice of whether to stratify the population at the district level, the school level, or both.

The “select_sites” macro provides two options for stratification at each level. The first option is to (1) construct strata using other software, (2) include the stratification variable in the data to be used by “select_sites”, and (3) identify the stratification variable when running the macro. Under this option, users may wish to first construct strata with a separate SAS macro called “cluster_k.” This macro uses cluster analysis and allows the users to choose from various cluster analyses algorithms.[2] The second option is for the user to ask the “select_sites” macro to construct strata. If the user provides a list of stratifying variables to the macro, “select_sites” will divide schools into strata based on all possible combinations of the stratifying variables.

The macro proceeds by organizing districts within their strata. Then it organizes schools by their districts, and within those districts, by their school strata. The macro forms final sampling strata by crossing any district-level strata with school-level strata. The user supplies the total number of schools that the sample should include, and the macro apportions the total across the strata to calculate the target number of schools from each stratum.

Sampling Approaches. Separately at the district and school levels, the macro allows users to choose from one of three sampling approaches: random selection with equal probabilities of selection, random selection with probabilities of selection proportional to size, and balanced selection (Litwok et al., 2022). Random selection rank-orders districts or schools (within strata, if the user specifies stratification) either completely randomly or with probability proportional to a size variable specified by the macro user. Balanced selection, based on the balanced sampling method described in Tipton

[1] Some users might have districts or schools they would like to include with probability equal to one (also known as “certainty districts” or “certainty schools”). Certainty districts and schools are not supported by select_sites. Users with certainty districts/schools should exclude those districts/schools from both the sampling frame fed into this macro and the requested school sample size.
[2] For the cluster_k macro and its documentation, see https://github.com/select-sites/exval.

(2013), rank-orders districts or schools based on their multivariate distance from the stratum mean or median using a distance metric specified by the macro user.[3]

The macro outputs three datasets for a user to conveniently track recruitment of districts and schools:

1. A rank-ordered list of districts within strata;
2. A rank-ordered list of schools within districts and school strata; and
3. A summary of the macro's sampling procedures.

If necessary, the rank-ordered lists in (1) and (2) can help users identify replacement districts or schools for recruitment. These lists can be output as SAS datasets or as Excel spreadsheets.

"Select_sites" draws from the SAS code used in simulations to test the performance of stratified random and balanced sampling for impact studies (Litwok et al., 2022). This guide includes an overview that explains the goal of the macro, syntax for using the macro, programmatic details and notes, descriptions of macro parameters and user options, output, and a worked example.

The worked example uses the same data as Litwok et al. (2022), based on the Common Core of Data. The data are available for download in our Open Science Framework repository at https://osf.io/fehjc/ or our Github repository at https://github.com/select-sites/exval. This document's appendix includes a codebook for the data.

[3] See the documentation for "cluster_k", available at https://github.com/select-sites/exval, for a deeper discussion of distance metrics.

Select_sites Macro

Introduction

This macro produces rank-ordered lists of districts and schools for users to recruit to participate in an impact study.

Users should call the macro using a school-level dataset containing the target population of schools nested within districts. Users specify the target sample size of schools as well as any caps the macro should apply to school recruitment (e.g., a maximum number of schools per district). If users would like to include stratification, they can specify one or more district- and/or school-level variables to be used in constructing sampling strata. By default, the macro allocates the sample proportionally by stratum, but the macro also supports custom user-submitted sampling targets by stratum. Users also indicate their preferred method for sampling at both the district and school levels (random equal, random unequal, or balanced).

The macro outputs three datasets for a user to conveniently track recruitment of districts and schools: a rank-ordered list of districts, a rank-ordered list of schools, and a summary of the sampling.

Syntax

```
%select_sites(
      input_data =,
      sample_size_school =,
      district_id =,
      school_id =,
      seed =,
      max_schools_per_district =,
      max_type =,

      strata_d =,
      method_d =,
      rand_size_d =,
      bal_distance_d =,

      strata_s =,
      method_s =,
      rand_size_s =,
      bal_distance_s =,

      targets_data =,
      out_district =,
      out_school =,
      out_summary =
);
```

Mandatory Parameters

The following parameters must be specified in each macro call. They are the minimum parameters required for the macro to run:

- input_data
- sample_size_school
- district_id
- school_id
- method_d
- method_s

Additionally, the following parameters are *conditionally* mandatory—whether they are required depends on the user's specifications in other parameters.

- rand_size_d*…if* `method_d=random unequal`
- bal_distance_d*…if* `method_d=balanced`
- rand_size_s*…if* `method_s=random unequal`
- bal_distance_s*…if* `method_s=balanced`

Optional Parameters

The user may choose to specify the following parameters or let the macro choose default values for them:

- seed
- max_schools_per_district
- max_type
- strata_d
- strata_s
- targets_data
- out_district
- out_school
- out_summary

Macro parameters and user options

This section describes each of the macro parameters and user options in detail. For each parameter, the guide indicates whether it is mandatory or optional to specify, a description of the parameter, acceptable values, default values, and corresponding notes.

input_data

- *Mandatory/Optional to Specify:* **Mandatory**

- *Description:* The name of the dataset containing the sampling frame.

- *Acceptable Values:* Any valid one- or two-level dataset name (note that two-level names including the libref are compatible).
    - Examples:
        - `input_data = mydata`
        - `input_data = mylib.mydata`
    - The dataset should be at the school level, with one observation per school. Schools should be nested within districts; each district may have one or more schools—and consequently one or more observations in the dataset.

- *Default Value:* None

sample_size_school

- *Mandatory/Optional to Specify:* **Mandatory**

- *Description:* Specifies the desired sample size in terms of the number of schools

- *Acceptable Values:* Positive integers

- *Default Value:* None

district_id

- *Mandatory/Optional to Specify:* **Mandatory**

- *Description:* The name of a variable in your `input_data` that uniquely identifies districts

- *Acceptable Values:* Any valid variable name in `input_data` that uniquely identifies districts

- *Default Value:* None

school_id

- *Mandatory/Optional to Specify:* **Mandatory**

- *Description:* The name of a variable in your `input_data` that uniquely identifies schools

- *Acceptable Values:* Any valid variable name in `input_data` that uniquely identifies schools

- *Default Value:* None

seed

- *Mandatory/Optional to Specify:* **Optional** because there is a default value

- *Description:* A positive integer specifying the seed used to generate random variables. These random variates are used not only for the two random sampling methods supported by the macro, but also for the balanced sampling method, to break ties in the rank ordering of districts or schools.

- *Acceptable Values:* Positive integers

- *Default Value:* Automatically generated in SAS by running: `call streaminit('pcg', 0)`

max_schools_per_district

- *Mandatory/Optional to Specify:* **Optional**

- *Description:* Specifies the maximum number of schools that may be sampled in each district. For instance, if the user specifies `max_schools_per_district=5`, then the macro will sample no more than five schools in each district. (That does not guarantee, however, that exactly five schools will be sampled in each selected district. Rather, between one to five schools will be sampled in each selected district.) The details of how this maximum is applied to districts depend on the related `max_type` parameter.

- *Acceptable Values:* Positive integers

- *Default Value:* The default is no maximum number of schools per district. If a user does not specify a number for this parameter, the macro will not impose a limit on the number of schools that may be sampled in each district.

max_type

- *Mandatory/Optional to Specify:* **Optional** because there is a default value

- *Description:* Specifies how the macro will apply the `max_schools_per_district` parameter, with respect to the school strata in each district. See the table of acceptable values for details.

- *Acceptable Values:*

Value[1]	**Description**

`overall`	The maximum is applied at the overall district level. The macro will sample no more than the maximum number for each district. In some cases, this will be straightforward. If the district only has schools in one school stratum, then the macro simply imposes the maximum value on the schools in that stratum. However, in cases where a district has schools in multiple school strata (and has more schools in total than the maximum value specified by the user), the macro needs to allocate the maximum. In these cases, the macro will distribute the maximum proportionally, according to the relative frequencies of school strata in the district. Consider an example where a user specifies a maximum value of 5 schools. Let's say that there are three school strata, and there is a district with 10 total schools—6 in school stratum A, 4 in school stratum B, and none in school stratum C. The macro would distribute the maximum of 5 schools proportionally, targeting 3 schools in school stratum A, 2 in B, and none in C.
`stratum`	The maximum is applied separately to each school stratum in the district. If a district has schools in three school strata, and if the maximum value is five, the macro will sample no more than five schools per school stratum in the district—and no more than fifteen schools total in the district.

[1] Value is case-insensitive.

- *Default Value:* `overall`

strata_d

- *Mandatory/Optional to Specify:* **Optional**

- *Description:* The name(s) of the variable(s) in your `input_data` to be used in constructing the sampling strata for districts. If you list a single variable in this parameter, the macro will treat each unique value of that variable as a separate district stratum. If you list multiple variables, the macro will treat each unique combination of the variables' values as a separate district stratum. If your sampling design has no stratification at the district level, you should leave this parameter blank.

- *Acceptable Values:* A space-delimited list of variable names (or just a single variable name). Abbreviated variable lists (such as name prefix lists or numbered range lists) are not supported. You may get either errors or incorrect results if you use abbreviated variable lists.
  For instance, the following space-delimited variable list is acceptable:
    - `strata_d = var1 var2 var3`

  Abbreviated variable lists such as the following are NOT acceptable:
    - `strata_d = var:`
    - `strata_d = var1-var3`

- *Default Value:* No district stratification

- *Notes:* The companion “cluster_k” macro produces a cluster variable (named in the `output_var_cluster` parameter of “cluster_k”) that can be entered in `strata_d` to identify the sampling strata.

  Where relevant, the output datasets created by this macro will include your original district stratification variables, as specified in this parameter. Additionally, these output datasets will include a numeric variable created by the macro, called `_x_stratum_d`, which uniquely identifies district strata. This variable’s purpose is primarily for the convenience of the macro processing, but you may also find it useful, especially if you specified multiple variables to form the district strata. And note that if you did not specify any district stratification, then the `_x_stratum_d` variable will still be created, but it will only have one value: 1.

method_d

- *Mandatory/Optional to Specify:* **Mandatory**

- *Description:* Specifies the sampling method by which the macro will order districts within district strata. The table of acceptable values below describes the three supported methods.

- *Acceptable Values:*

Value[1]	**Description**	**Associated Parameter**
`random equal`	The macro will order districts randomly within district strata, and each district within a district stratum will have the same probability of selection into the sample (and the same probability of appearing at any given place in the ordered list).	N/A
`random unequal`	The macro will order districts randomly within district strata, and	`rand_size_d`

	each district's probability of selection into the sample will be proportional to its size, as specified in the associated `rand_size_d` parameter. Larger districts will have a greater probability of appearing higher in the ordered list (and, thus, a greater probability of selection into the sample). This sampling method mimics probability proportional to size (PPS) sampling. But whereas traditional PPS sampling selects a fixed number of units, our adapted approach generates an ordered list. See Appendix B of Litwok et al. (2022) for a description of this approach and a comparison with traditional PPS sampling.	
`balanced`	The macro will order districts in ascending order, by their multivariate distance from the stratum centroid, using the distance measure specified in the associated `bal_distance_d` parameter. Districts that are more "typical" (i.e., districts with the shortest distance from the stratum centroid) will be higher on the ordered list.[2] Users are expected to already have a variable with their preferred measure in their dataset. The companion "cluster_k" macro can be used to generate a distance variable (named in the `output_var_distance` parameter of "cluster_k") to be entered as the `bal_distance_d` parameter.	`bal_distance_d`

[1] Value is case-insensitive.

[2] Ties between districts in the same stratum sharing the same distance value will be broken randomly. Therefore, the `seed` parameter is relevant not only for the two random sampling methods, but also for balanced sampling.

- *Default Value:* None

rand_size_d

- *Mandatory/Optional to Specify:* **Mandatory** *only if* the user specifies `method_d=random unequal`

- *Description:* The name of a variable in your `input_data` that contains the size measure for the macro's variant of PPS sampling. This size variable determines the district selection probabilities when the user specifies the `random unequal` method for district sampling.

- *Acceptable Values:* Any valid variable name in `input_data`. Additionally, there are two important restrictions on the values contained in the variable, itself:
  1) All values must be greater than zero.
  2) The algorithm the macro uses to implement PPS sampling as a rank-ordered list requires that the size values all be integers. If the macro detects non-integers in the `rand_size_d` variable, it will issue a warning and apply a ceiling function to the variable, as described in the *Notes* below.

- *Default Value:* None

- *Notes:* As noted above, the macro will apply a ceiling function to non-integer values in the `rand_size_d` variable. That does not necessarily mean that you cannot use a variable with non-integer values, but you may want to rescale your variable first, before calling the macro. For instance, if your desired size variable ranges from >0 to 1 (recall that values of precisely zero are prohibited), the ceiling function in the macro will convert all values to 1, and there will not be any variation in the selection probabilities. In such a case, you may want to rescale your variable (by 100, or 1,000, etc.) before calling the macro. Although, note that you may run into resource issues in SAS if you choose a scale factor too large (or if your original size variable has extremely large values) because the macro creates an intermediate dataset with the number of observations equal to the overall sum of the random size measure.

bal_distance_d

- *Mandatory/Optional to Specify:* **Mandatory** *only if* the user specifies `method_d=balanced`

- *Description:* The name of a variable in your `input_data` that describes the multivariate distance of districts from their stratum centroid. This parameter is designed for use with the `balanced` method for district sampling.

- *Acceptable Values:* Any valid variable name in `input_data`.

- *Default Value:* None

Notes: The macro will use this distance metric to rank-order districts for balanced sampling. This macro does not calculate the distance metric: users must supply it. The companion "cluster_k" macro can be used to generate a variety of multivariate distance measures, including the squared Euclidean distance recommended by Tipton (2013). For a complete description of implementing this approach, see the detailed example "Implementing balanced sampling as in Tipton (2013)" in the "Examples" section.

Note that users are not limited to a distance measure; you are technically free to choose any variable, including variables completely unrelated to balanced sampling. For instance, if you wish to order districts by district enrollment, you can specify your district enrollment variable in the `bal_distance_d` parameter. The macro would then order districts within district strata by ascending enrollment. If you wish to order districts in descending order by enrollment, you could first create a new variable, multiplying the enrollment by -1, and then feed that new variable into the `bal_distance_d` parameter.

strata_s

- *Mandatory/Optional to Specify:* **Optional**

- *Description:* The name(s) of the variable(s) in your `input_data` that form the sampling strata for schools. The macro will organize schools first by their districts; then within those districts, the macro will organize schools by the school strata specified in this parameter. If you list a single variable in this parameter, the macro will treat each unique value of that variable as a separate school stratum. If you list multiple variables, the macro will treat each unique combination of the variables' values as a separate school stratum. If your sampling design has no stratification at the school level, you should leave this parameter blank.

- *Acceptable Values:* A space-delimited list of variable names (or just a single variable name). Abbreviated variable lists (such as name prefix lists or numbered range lists) are not supported. You may get either errors or incorrect results if you use abbreviated variable lists.
  For instance, the following space-delimited variable list is acceptable:
  - `strata_s = var1 var2 var3`

  Abbreviated variable lists such as the following are NOT acceptable:
  - `strata_s = var:`
  - `strata_s = var1-var3`

- *Default Value:* No school stratification

- *Notes:* The companion "clutser_k" macro produces a cluster variable (named in the `output_var_cluster` parameter of "clutser_k") that can be entered in `strata_s` to form the sampling strata.

Where relevant, the output datasets created by this macro will include your original school stratification variables, as specified in this parameter. Additionally, these output datasets will include a numeric variable created by the macro, called _x_stratum_s, which uniquely identifies school strata. This variable's purpose is primarily for the convenience of the macro processing, but you may also find it useful, especially if you specified multiple variables to form the school strata. And note that if you did not specify any school stratification, then the _x_stratum_s variable will still be created, but it will only have one value: 1.

method_s

- *Mandatory/Optional to Specify:* **Mandatory**

- *Description:* Specifies the sampling method by which the macro will order schools within districts and within school strata. The table of acceptable values below describes the three supported methods.

- *Acceptable Values:*

Value[1]	**Description**	**Associated Parameter**
random equal	The macro will order schools randomly within districts and within school strata, and each school within a district and a school stratum will have the same probability of selection into the sample (and the same probability of appearing at any given place in the ordered list).	N/A
random unequal	The macro will order schools randomly within districts and within school strata, and each school's probability of selection into the sample will be proportional to its size, as specified in the associated rand_size_s parameter. Larger schools will have a greater probability of appearing higher in the ordered list (and, thus, a greater probability of selection into the sample). This sampling method mimics probability proportional to size (PPS) sampling. But whereas traditional PPS sampling selects a fixed number of units, our adapted approach generates an ordered list. See Appendix B of Litwok et al. (2022) for	rand_size_s

	a description of this approach and a comparison with traditional PPS sampling.	
balanced	The macro will order schools in ascending order, by their multivariate distance from the stratum centroid, using the measure specified in the bal_distance_s parameter. Schools that are more "typical" (i.e., schools with the shortest distance from the stratum centroid) will be higher on the ordered list.[2] Users are expected to already have a variable with the distance measure in their dataset. The companion "clutser_k" macro can be used to generate a distance variable (named in the output_var_distance parameter of "cluster_k") to be entered as the bal_distance_s parameter.	bal_distance_s

[1] Value is case-insensitive.

[2] Ties between schools in the same district and same school stratum sharing the same distance value will be broken randomly. Therefore, the seed parameter is relevant not only for the two random sampling methods, but also for balanced sampling.

- *Default Value:* None

rand_size_s

- *Mandatory/Optional to Specify:* **Mandatory** *only if* the user specifies method_s=random unequal

- *Description:* The name of a variable in your input_data that contains the size measure for the macro's variant of PPS sampling. This size variable determines the school selection probabilities when the user specifies the random unequal method for school sampling.

- *Acceptable Values:* Any valid variable name in input_data. Additionally, there are two important restrictions on the values contained in the variable, itself:
  1) All values must be greater than zero.
  2) The algorithm the macro uses to implement PPS sampling as a rank-ordered list requires that the size values all be integers. If the macro detects non-integers in the rand_size_s variable, it will issue a warning and apply a ceiling function to the variable.

- *Default Value:* None

- *Notes:* As noted above, the macro will apply a ceiling function to non-integer values in the rand_size_s variable. That does not necessarily mean that you cannot use a variable with non-integer values, but you may want to rescale your variable first, before calling the macro. For instance, if your desired size variable ranges from >0 to 1 (recall that values of precisely zero are prohibited), the ceiling function in the macro will convert all values to 1, and there will not be any variation in the selection probabilities. In such a case, you may want to rescale your variable (by 100, or 1,000, etc.) before calling the macro. Although, note that you may run into resource issues in SAS if you choose a scale factor too large (or if your original size variable has extremely large values)—because the macro creates an intermediate dataset with the number of observations equal to the overall sum of the random size measure.

bal_distance_s

- *Mandatory/Optional to Specify:* **Mandatory** *only if* the user specifies method_s=balanced

- *Description:* The name of a variable in your input_data that describes the multivariate distance of schools from their stratum centroid. This parameter is designed for use with the balanced method for school sampling.

- *Acceptable Values:* Any valid variable name in input_data.

- *Default Value:* None

  Notes: The macro will use this parameter to rank-order schools for balanced sampling. This macro does not calculate the distance metric: users must supply it. The companion “cluster_k” macro can be used to generate a variety of multivariate distance measures, including the squared Euclidean distance recommended by Tipton (2013). For a complete description of implementing this approach, see the detailed example “Implementing balanced sampling as in Tipton (2013)” in the “Examples” section.

  Note that users are not limited to a distance measure; you are technically free to choose any variable, including variables completely unrelated to balanced sampling. For instance, if you wish to order schools by school enrollment, you can specify your school enrollment variable in the bal_distance_s parameter. The macro would then order schools within districts and within school strata by ascending enrollment. If you wish to order schools in descending order by enrollment, you could first create a new variable, multiplying the enrollment by -1, and then feed that new variable into the bal_distance_s parameter.

targets_data

- *Mandatory/Optional to Specify:* **Optional**

- *Description:* The name of a dataset containing sampling targets by stratum. By default, the macro will determine its own sampling targets, proportionally allocating the user's specified sample size over the district and school strata. If you would prefer an alternative allocation, you may specify custom sampling targets in a dataset and feed that dataset to this parameter. Note that this only applies to stratified sample designs. If there is no stratification, at either the district or school level, then there are no targets to specify beyond the overall school sample size in the `sample_size_school` parameter.

- *Acceptable Values:* Any valid one- or two-level dataset name (note that two-level names including the libref are compatible). Additionally, the dataset must have these specifications:
    - All of the variables specified in the `strata_d` and `strata_s` parameters must be present in this dataset and must have the same variable attributes (most importantly, character/numeric type) as their versions in the `input_data`.
    - There must be only one observation in the dataset per unique combination of the stratification variables (those specified in the `strata_d` and `strata_s` parameters).
    - You must create a numeric variable called `target` with your desired sampling target. And the sum of the target variable, over all the observations in this dataset, must be equal to the specified `sample_size_school`.

- *Default Value:* None

- *Example:*

```
* create the targets dataset;
** we will generate the targets from the
base_school dataset available at https://osf.io/fehjc/;
** we will set a sampling target of 5 schools for
each unique combination of district stratum (strat_d) and
school stratum (strat_s) (and with a total of 18 such unique
combinations, our total sample size will be 90
schools);
proc sql;
   create table targets as
        select     distinct strat_d, strat_s,
                   5 as target
        from       base_school;
quit;

* run the selection macro;
```

```
%select_sites(
   input_data = base_school,
   sample_size_school = 90,
   district_id = leaid,
   school_id = ncessch,
   seed = 1234,
   max_schools_per_district = 5,
   max_type = overall,

   strata_d = strat_d,
   method_d = random unequal,
   rand_size_d = nschools_elig,

   strata_s = strat_s,
   method_s = random equal,

   targets_data = targets
);
```

out_district

- *Mandatory/Optional to Specify:* **Optional** because there is a default value

- *Description:* The name of the output dataset that will be produced by the macro, showing the ordered recruitment list of districts. This list will include all districts in the sampling frame. They will be organized by district strata and ordered within those strata according to the specified sampling method. A variable in the dataset will identify which districts were selected for the initial sample, as well as how many schools were selected in each of those districts. The remaining districts (those not identified as part of the initial sample) may be used as prospective replacements if a sampled district declines to participate.

- *Acceptable Values:* Any valid one- or two-level dataset name (note that two-level names including the libref are compatible). Examples:
  - `out_district = list_d`
  - `out_district = mylib.list_d`

  Default Value: `_out_district`

out_school

- *Mandatory/Optional to Specify:* **Optional** because there is a default value

- *Description:* The name of the output dataset that will be produced by the macro, showing the ordered recruitment list of schools. This list will include all schools in the sampling frame. They will be organized by district (ordered within district strata by the district recruitment order shown in the `out_district` list) and by school strata and ordered within those strata according to the specified sampling method. A variable in the dataset will identify which schools were selected for the initial sample. The remaining schools (those not identified as part of the initial sample) may be used as prospective replacements if a sampled district or school declines to participate.

- *Acceptable Values:* Any valid one- or two-level dataset name (note that two-level names including the libref are compatible). Examples:
  - `out_school = list_s`
  - `out_school = mylib.list_s`

  Default Value: `_out_school`

out_summary

- *Mandatory/Optional to Specify:* **Optional** because there is a default value.

- *Description:* The name of the output dataset that will be produced by the macro, showing a summary of the sampling. The dataset will report the total number of schools sampled by each sampling stratum (each unique combination of the district and school stratification variables), along with the total number of schools that were available for sampling in the sampling frame.

- *Acceptable Values:* Any valid one- or two-level dataset name (note that two-level names including the libref are compatible). Examples:
  - `out_summary = summary`
  - `out_summary = mylib.summary`

- *Default Value:* `_out_summary`

Output

The macro outputs a series of datasets. These datasets include:

(i) The district recruitment list, identifying the initial sample of selected districts as well as the ordered list of prospective replacement districts

(ii) The school recruitment list, identifying the initial sample of selected schools as well as the ordered list of prospective replacement schools

(iii) A summary of the sample results, by stratum

Users may find it convenient to output the recruitment lists as spreadsheets (e.g., as Microsoft Excel documents). The example below includes code for outputting the data as a spreadsheet.

Examples

Creating rank-ordered lists of districts and schools

The example that follows demonstrates how to use the macro for creating rank-ordered lists of districts and schools. The example uses the sample data “base_school” available for download in our Open Science Framework repository at https://osf.io/fehjc/ or our Github repository at https://github.com/select-sites/exval.

```
* PREP
-----------------------;
* run macro definition program;
%include "[your file path to the program]\select_sites.sas" / lrecl=32767;

* load the sample data from the macro GitHub repository and import into SAS;
let indir = [your file path to the data];
proc import datafile=
      "&indir.\base_school.csv"
      out=base_school
      dbms=csv replace;
run;


* SAMPLING
-----------------------;
* version 1: random sampling;
%select_sites(
      input_data = base_school,
      sample_size_school = 200,
      district_id = leaid,
      school_id = ncessch,
      seed = 1234,
      max_schools_per_district = 5,
      max_type = overall,

      strata_d = strat_d,
      method_d = random unequal,
      rand_size_d = nschools_elig,

      strata_s = strat_s,
      method_s = random equal,

      out_district = v1_district,
      out_school = v1_school,
      out_summary = v1_summary
);

* version 2: balanced sampling;
%select_sites(
      input_data = base_school,
      sample_size_school = 200,
      district_id = leaid,
      school_id = ncessch,
      seed = 1234,
      max_schools_per_district = 5,
      max_type = overall,

      strata_d = strat_d,
      method_d = balanced,
      bal_distance_d = distance_d_y,

      strata_s = strat_s,
      method_s = balanced,
      bal_distance_s = distance_s_y,

      out_district = v2_district,
      out_school = v2_school,
      out_summary = v2_summary
);
```

After running the macro, users may find it convenient to output the datasets produced by the macro as spreadsheets to be used in their recruitment effort. The code below outputs the three files as three different worksheets in a Microsoft Excel workbook.

```
%let outdir = "[your file path for output]";
proc export outfile=
      "&outdir.\Sample Selection Output.xlsx"
      data=v1_district
      dbms=excel2007
      label
      replace;
      sheet="District_labeled";
run;

proc export outfile=
      "&outdir.\Sample Selection Output.xlsx"
      data=v1_school
      dbms=excel2007
      label
      replace;
      sheet="School_labeled";
run;

proc export outfile=
      "&outdir.\Sample Selection Output.xlsx"
      data=v1_summary
      dbms=excel2007
      label
      replace;
      sheet="Summary";
```

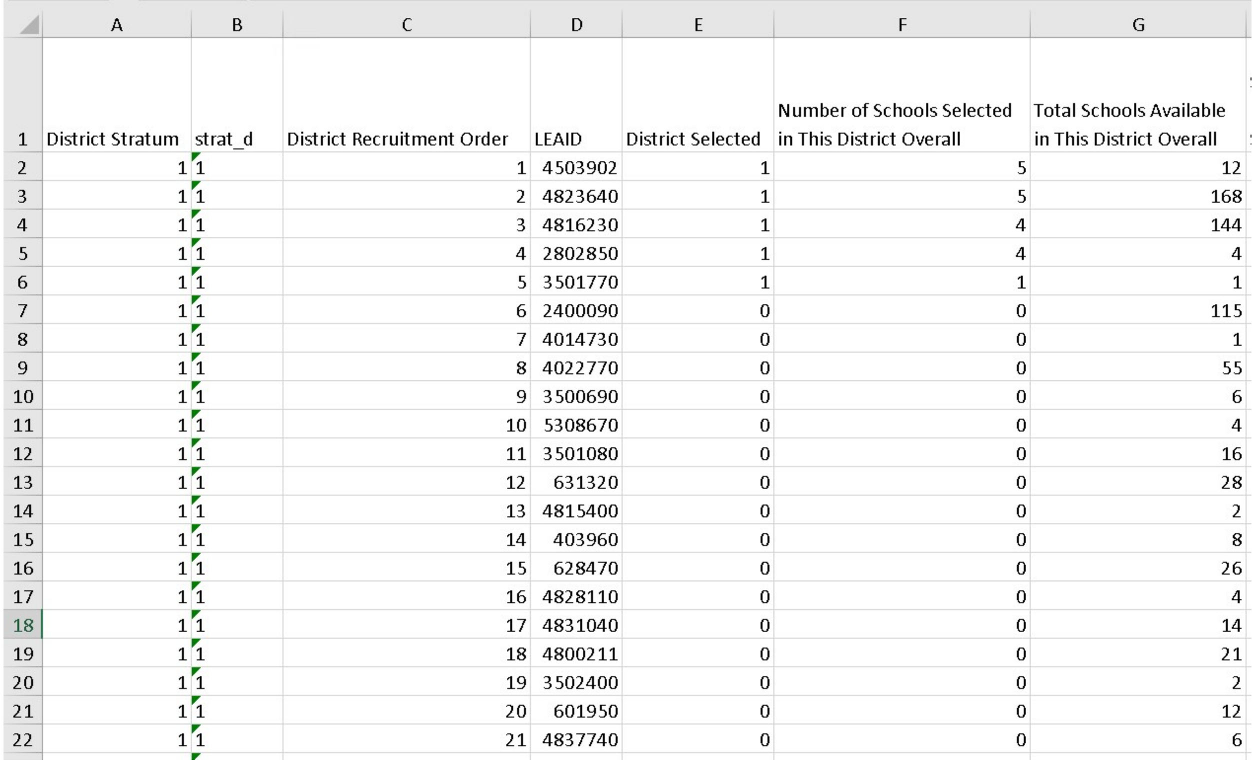
```
run;
```

Figures 1, 2, and 3 show screenshots of the three data files output to Microsoft Excel.

Figure 1 shows the rank-ordered list of districts. The output includes one row for each district in the target population, organized within the district strata entered by the user. All the districts pictured in Figure 1 are in District Stratum 1. Column E is an indicator for whether the macro selected this district to be included in the initial sample (those marked with a 1 are part of the initial sample selected by the macro, and those marked with a 0 are prospective replacements if those districts initially sampled decline to participate). Users may want to filter the spreadsheet by Column E to limit the spreadsheet to those districts the macro selected for the initial sample. Columns F and G report the number of schools the macro selected from this district as well as the total number of schools available in the district, respectively. Not pictured in Figure 1 but also included in the output dataset are pairs of columns akin to Columns F and G for each school stratum.

We envision evaluators would find a spreadsheet like Figure 1 useful for identifying both the districts to recruit initially and prospective replacement districts for those that decline to participate. For instance, if any of the districts in rows 2 through 6 of Figure 1 decline to participate, evaluators would move to the districts in rows 7, 8, 9, etc. as replacements, in order.

Figure 1. Rank-Ordered District List

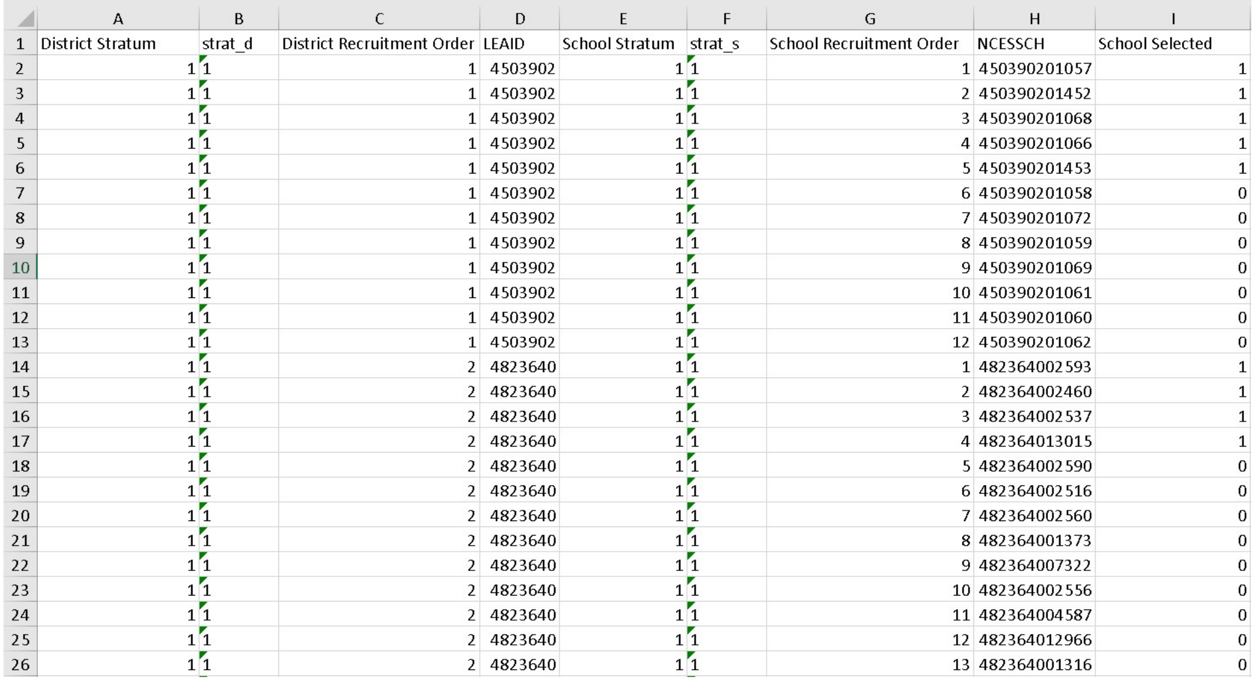


	A	B	C	D	E	F	G
1	District Stratum	strat_d	District Recruitment Order	LEAID	District Selected	Number of Schools Selected in This District Overall	Total Schools Available in This District Overall
2	1	1	1	4503902	1	5	12
3	1	1	2	4823640	1	5	168
4	1	1	3	4816230	1	4	144
5	1	1	4	2802850	1	4	4
6	1	1	5	3501770	1	1	1
7	1	1	6	2400090	0	0	115
8	1	1	7	4014730	0	0	1
9	1	1	8	4022770	0	0	55
10	1	1	9	3500690	0	0	6
11	1	1	10	5308670	0	0	4
12	1	1	11	3501080	0	0	16
13	1	1	12	631320	0	0	28
14	1	1	13	4815400	0	0	2
15	1	1	14	403960	0	0	8
16	1	1	15	628470	0	0	26
17	1	1	16	4828110	0	0	4
18	1	1	17	4831040	0	0	14
19	1	1	18	4800211	0	0	21
20	1	1	19	3502400	0	0	2
21	1	1	20	601950	0	0	12
22	1	1	21	4837740	0	0	6

Figure 2 shows the rank-ordered list of schools. The output includes one row for each school in the target population, organized within district, district stratum, and school stratum. All the schools pictured in Figure 2 are in District Stratum 1 and School Stratum 1; the first 12 schools are in one district; the next 25 schools are in a second district. Column I is an indicator for whether the macro selected this school to be included in the initial sample (those marked with a 1 are part of the initial sample selected by the macro, and those marked with a 0 are prospective replacements if those schools initially sampled decline to participate). Users may want to filter the spreadsheet by Column I to limit the spreadsheet to those schools the macro selected for the initial sample.

We envision evaluators would find a spreadsheet like Figure 2 useful for identifying both the schools to recruit initially and prospective replacement schools for those that decline to participate. For instance, if any of the schools in rows 2 through 6 of Figure 2 decline to participate, evaluators would move to the schools in rows 7, 8, 9, etc. as replacements, in order.

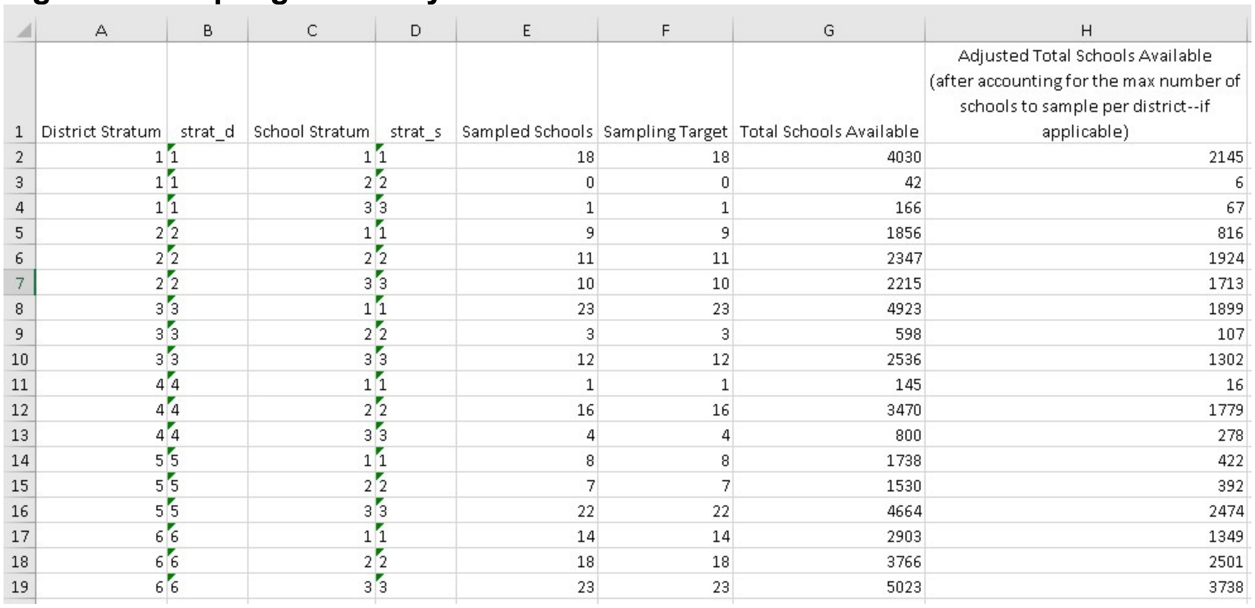
Figure 2. Rank-Ordered School List



	A	B	C	D	E	F	G	H	I
1	District Stratum	strat_d	District Recruitment Order	LEAID	School Stratum	strat_s	School Recruitment Order	NCESSCH	School Selected
2	1	1	1	4503902	1	1	1	450390201057	1
3	1	1	1	4503902	1	1	2	450390201452	1
4	1	1	1	4503902	1	1	3	450390201068	1
5	1	1	1	4503902	1	1	4	450390201066	1
6	1	1	1	4503902	1	1	5	450390201453	1
7	1	1	1	4503902	1	1	6	450390201058	0
8	1	1	1	4503902	1	1	7	450390201072	0
9	1	1	1	4503902	1	1	8	450390201059	0
10	1	1	1	4503902	1	1	9	450390201069	0
11	1	1	1	4503902	1	1	10	450390201061	0
12	1	1	1	4503902	1	1	11	450390201060	0
13	1	1	1	4503902	1	1	12	450390201062	0
14	1	1	2	4823640	1	1	1	482364002593	1
15	1	1	2	4823640	1	1	2	482364002460	1
16	1	1	2	4823640	1	1	3	482364002537	1
17	1	1	2	4823640	1	1	4	482364013015	1
18	1	1	2	4823640	1	1	5	482364002590	0
19	1	1	2	4823640	1	1	6	482364002516	0
20	1	1	2	4823640	1	1	7	482364002560	0
21	1	1	2	4823640	1	1	8	482364001373	0
22	1	1	2	4823640	1	1	9	482364007322	0
23	1	1	2	4823640	1	1	10	482364002556	0
24	1	1	2	4823640	1	1	11	482364004587	0
25	1	1	2	4823640	1	1	12	482364012966	0
26	1	1	2	4823640	1	1	13	482364001316	0

Figure 3 shows the summary table produced by the macro. The table includes one row for each of the 18 sampling strata in this example (defined by 6 district strata crossed with 3 school strata). Columns A through D define the strata. Columns E and F report the number of schools sampled and targeted by the macro, respectively. The number of schools sampled should be equal to the number of schools targeted—if these are not equal the macro will issue an error, and the output in Figure 3 will be helpful for users to assess what has gone wrong. Column G reports the total number of schools available in each of the strata, and Column H reports the total number of schools available in each of the strata after applying the maximum number of schools per district.

Figure 3. Sampling Summary



	A	B	C	D	E	F	G	H
1	District Stratum	strat_d	School Stratum	strat_s	Sampled Schools	Sampling Target	Total Schools Available	Adjusted Total Schools Available (after accounting for the max number of schools to sample per district--if applicable)
2	1	1	1	1	18	18	4030	2145
3	1	1	2	2	0	0	42	6
4	1	1	3	3	1	1	166	67
5	2	2	1	1	9	9	1856	816
6	2	2	2	2	11	11	2347	1924
7	2	2	3	3	10	10	2215	1713
8	3	3	1	1	23	23	4923	1899
9	3	3	2	2	3	3	598	107
10	3	3	3	3	12	12	2536	1302
11	4	4	1	1	1	1	145	16
12	4	4	2	2	16	16	3470	1779
13	4	4	3	3	4	4	800	278
14	5	5	1	1	8	8	1738	422
15	5	5	2	2	7	7	1530	392
16	5	5	3	3	22	22	4664	2474
17	6	6	1	1	14	14	2903	1349
18	6	6	2	2	18	18	3766	2501
19	6	6	3	3	23	23	5023	3738

Implementing balanced sampling as in Tipton (2013)

The simulations in Litwok et al. (2022) implement balanced sampling at both the district and school levels. The simulations used “cluster_k” to create distance measures to be used by “select_sites” when implementing balanced sampling at both the district and school levels. We create six clusters of districts using the district-level average of the number of students enrolled in the school, the district-level average of the percentage of students who are FRPL-eligible, the Census region, and expenditures per pupil. We also create three clusters of schools using the number of students enrolled in the school and the percentage of students who are FRPL-eligible. For convenience, the “base_school” data file already includes the resulting district-level and school-level distance measures used by Litwok et al. (2022) as variables on the file (distance_d_y at the district-level and distance_s_y at the school-level).

To implement balanced sampling at the district level, the simulations in Litwok et al. (2022) call “select_sites” with the following parameter values:

```
method_d = balanced,
bal_distance_d = distance_d_y,
```

To implement balanced sampling at the school level, the simulations in Litwok et al. (2022) call “select_sites” with the following parameter values:

```
method_s = balanced,
bal_distance_s = distance_s_y,
```

Programmatic Details and Notes

- SAS Version: These macros were programmed and tested in SAS 9.4.

- Notes:

  - The macro creates various intermediate datasets (temporary working datasets that are used in the processing of the macros and are deleted after they are no longer needed). To reduce the likelihood that these datasets conflict with (and overwrite) existing datasets in the user's work library, the macro uses SAS's "DATAn naming convention."

    With this feature, SAS defines a set of potential dataset names (DATA1, DATA2, etc., all the way to DATA9999) and sequentially searches for unused dataset names among that group, in the work library. When it finds an unused name, it will assign that to the intermediate dataset the macro creates.

    This *significantly reduces the likelihood of* (rather than categorically prevents) conflicts because there is still a possibility (however remote) that the user may have existing datasets in their work library that cover all of the potential names that SAS could assign (from DATA1 to DATA9999). If all 9,999 potential dataset names are in use, SAS will start from the beginning of the list and assign the name DATA1 to the new dataset (and overwrite the existing DATA1 dataset). In that unlikely scenario, the user should beware that their data may be overwritten by the macro.

  - As mentioned above, the macro deletes the intermediate datasets when it is done with them. However, if the macro terminates early because of errors, there may be leftover intermediate datasets that were not yet deleted (they should be easy to identify because they all follow the DATAn naming convention, as in DATA1, DATA2, etc.). The user should feel free to delete those intermediate datasets, if they want to clean up their work library, but there is no need to do so. Whether those datasets remain in the work library will not interfere with the processing of the macro.

References

Litwok, D., Nichols, A., Shivji, A., & Olsen, R. (2022). Selecting districts and schools for impact studies in education: A simulation study of different strategies. *Journal of Research on Educational Effectiveness.* DOI: https://doi.org/10.1080/19345747.2022.2128952.

Tipton, E. (2013). Stratified sampling using cluster analysis: A sample selection strategy for improved generalizations from experiments. *Evaluation Review*, 37(2), 109-139.

Tipton, E., & Olsen, R. B. (2022). *Enhancing the Generalizability of Impact Studies in Education*. (NCEE 2022-003). National Center for Education Evaluation and Regional Assistance, Institute of Education Sciences, U.S. Department of Education. Washington, DC.

Appendix

Along with the macro, the Github repository makes available a sample data set (“base_school”) that can be used to test the macro’s performance. Applying the macro to these data are the basis for many of the examples that appear within this documentation. This appendix includes a codebook for that sample dataset. See Litwok et al. (2022) for detailed discussion of (i) defining the population included in this dataset; (ii) imputing missing values for certain variables; (iii) our approach to Probability Proportional to Size (PPS) sampling; and (iv) strata for schools and districts.

Sample data codebook

Variable Name	Variable Description
leaid	NCES district identifier
strat_d	District stratum
distance_d_y	Distance from district stratum group mean (district-level)
lottery_d_start	Starting number for district lottery (“PPS” sampling)
lottery_d_end	Ending number for district lottery (“PPS” sampling)
ncessch	NCES school identifier
enr_tot_d	Total enrollment (district-level)
region_d	Region (district-level)
nschools_elig	Total number of eligible schools (district-level)
enr_tot_imp	Total enrollment (school-level)
sme_pct_frp_tc_imp	% eligible for free/reduced-price lunch (school-level)
epp_d_imp	Expenditures per pupil (district-level)
other_d	Administrator leadership (district-level)
es_nschools_elig	Number of eligible schools (district-level; standardized)
es_epp_d_imp	Expenditures per pupil (district-level; standardized)
es_enr_tot_imp	Total enrollment (school-level; standardized)
es_sme_pct_frp_tc_imp	% eligible for free/reduced-price lunch (school-level; standardized)
es_other_d	Administrator leadership (district-level; standardized)
nschools	Total number of schools (district-level)
strat_s	School stratum
distance_s_y	Distance from school stratum group mean (school-level)
distance_s_y_overall	Distance from overall mean (school-level)