Analyzing Complex Survey Data: Some key issues to be aware of

Richard Williams, University of Notre Dame, https://www3.nd.edu/~rwilliam/

Last revised September 10, 2024

Be sure to read the Stata Manual's *Introduction to Survey Commands* first. It explains how and why the survey design and the survey data collection need to be taken into account when doing your analysis. Pay particular attention to the introduction and skim the rest. There are just a few additional points I want to illustrate here.

Most of the analysis we have done so far assumes that cases were selected via simple random sampling – the equivalent of drawing names out of a hat. In reality, sampling schemes are often much more complicated than that. Survey data (I am quoting a lot from the Stata Manual here) are characterized by

- *sampling weights*, aka probability weights or pweights: "In sample surveys, observations are selected through a random process, but different observations may have different probabilities of selection," e.g. Black individuals may be oversampled
- *cluster sampling*: "Individuals are not sampled independently in most survey designs. Collections of individuals (for example, counties, city blocks, or households) are typically sampled as a group known as a cluster."
- *stratification*: "In surveys, different groups of clusters are often sampled separately. These groups are called strata. For example, the 254 counties of a state might be divided into two strata, say, urban counties and rural counties. Then 10 counties might be sampled from the urban stratum, and 15 from the rural stratum"

Failure to take the sampling scheme into account can lead to inaccurate point estimates and/or flawed estimates of the standard errors.

The svyset command and the svy: prefix. Your data need to be `svyset` first. The `svyset` command tells Stata everything it needs to know about the data set's sampling weights, clustering, and stratification. You only need to `svyset` your data once. Hopefully, the provider of your data has told you what you need for the `svyset` command or has even `svyset` the data for you. If not, you are going to have to do some reading or get some help to figure out how to do it yourself.

If the data are already `svyset`, then typing `svyset` by itself will show what the settings are.

```
. webuse nhanes2f, clear
. svyset

      pweight: finalwgt
          VCE: linearized
  Single unit: missing
     Strata 1: stratid
         SU 1: psuid
        FPC 1: <zero>
```

```
. sum finalwgt stratid psuid

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
    finalwgt |     10337    11320.85    7304.457       2000      79634
     stratid |     10337    16.65986    9.499389          1         32
       psuid |     10337    1.482151    .4997055          1          2
```

The pweight variable is finalwgt. The summary statistics show you that each person in the sample represents anywhere from 2,000 to 79,634 people in the population. Put another way, if the pweight for a person is 10,000, that means that the respondent had one chance in 10,000 of being selected for the sample; or, if you prefer, that person represents 10,000 people in the population.

Once the data are `svyset`, you need to remember to use the `svy:` prefix with your commands. For example, instead of typing

```
. reg weight height age i.female i.black

      Source |       SS       df       MS              Number of obs =   10337
-------------+------------------------------           F(  4, 10332) =  881.52
       Model |  620082.606     4  155020.652           Prob > F      =  0.0000
    Residual |  1816944.64 10332  175.856044           R-squared     =  0.2544
-------------+------------------------------           Adj R-squared =  0.2542
       Total |  2437027.25 10336    235.7805           Root MSE      =  13.261

------------------------------------------------------------------------------
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      height |   .7485279     .01966    38.07   0.000     .7099905    .7870652
         age |   .1237255   .0078948    15.67   0.000     .1082501    .1392009
    1.female |  -1.540187   .3721392    -4.14   0.000    -2.269652   -.8107221
     1.black |   3.679295   .4256284     8.64   0.000     2.844981    4.513609
       _cons |  -59.05337   3.563342   -16.57   0.000    -66.03822   -52.06853
------------------------------------------------------------------------------
```

You should type

```
. svy: reg weight height age i.female i.black
(running regress on estimation sample)

Survey: Linear regression

Number of strata   =        31                  Number of obs      =      10337
Number of PSUs     =        62                  Population size    =  117023659
                                                Design df          =         31
                                                F(   4,     28)    =     880.32
                                                Prob > F           =     0.0000
                                                R-squared          =     0.2887

------------------------------------------------------------------------------
             |             Linearized
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      height |   .7409099   .0274549    26.99   0.000     .6849151    .7969046
         age |   .1520647   .0117481    12.94   0.000     .1281044    .1760251
    1.female |  -2.924562   .6054516    -4.83   0.000    -4.159389   -1.689736
     1.black |   4.033541   .7436242     5.42   0.000     2.516909    5.550172
       _cons |  -58.19333   4.909523   -11.85   0.000    -68.20637   -48.18029
------------------------------------------------------------------------------
```

Notice some differences in the output. You are told that this sample represents a population of 117 million people. You don't get an ANOVA table anymore. The coefficients are somewhat different, reflecting the fact that cases are not being weighted equally anymore. The changes in coefficients and also the fact that clustering and stratification are taken into account affect the significance tests and confidence intervals.

There are a few other key differences between the analysis of survey and non-survey data that you need to be aware of.

Subsample analyses. [Note: I don't really understand why subsample analysis works this way. Some very smart people do understand so I accept what follows on blind faith.] One thing to be careful of is subsample analyses, e.g. analyzing men only. With non-svy data, you usually just create an extract first which has only your desired cases; or you include an `if` qualifier with your command, e.g. something like

```
reg y x1 x2 x3 if female==0
```

With svy data, however, that kind of approach can, under certain conditions, seriously bias your results, i.e. the standard error calculations can be wrong if all the data are not available. Instead, you should use the `subpop` option to specify your sample. As UCLA explains in its Stata FAQs, "When the subpopulation option(s) is used, only the cases defined by the subpopulation are used in the calculation of the estimate, but all cases are used in the calculation of the standard errors." UCLA further adds that "Using `if` in the `subpop` option does not remove cases from the analysis. The cases excluded from the subpopulation by the `if` are still used in the calculation of the standard errors, as they should be." So, do something like this:

```
. svy, subpop(if female==0): reg weight height age i.black
(running regress on estimation sample)

Survey: Linear regression

Number of strata   =        31                  Number of obs     =       10,337
Number of PSUs     =        62                  Population size   =  117,023,659
                                                Subpop. no. obs   =        4,909
                                                Subpop. size      =   56,122,035
                                                Design df         =           31
                                                F(   3,     29)   =       241.42
                                                Prob > F          =       0.0000
                                                R-squared         =       0.1977

------------------------------------------------------------------------------
             |             Linearized
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      height |   .8606775    .030968    27.79   0.000     .7975178    .9238373
         age |    .108198   .0161444     6.70   0.000     .0752713    .1411248
     1.black |  -.1866542   .7693002    -0.24   0.810    -1.755652    1.382344
       _cons |  -77.00092   5.700856   -13.51   0.000    -88.62789   -65.37395
------------------------------------------------------------------------------
```

Some commands do not work with svy: Some commands, like summarize, do not work with the svy: prefix. Sometimes you can find an alternative command that does, e.g. `svy: mean`.

```
. webuse nhanes2f, clear
. svy: summarize diabetes black weight height
summarize is not supported by svy with vce(linearized); see help svy estimation for a
list of Stata estimation commands that are supported by svy
r(322);

. svy: mean diabetes black weight height
(running mean on estimation sample)

Survey: Mean estimation

Number of strata =      31       Number of obs    =      10335
Number of PSUs   =      62       Population size  =  116997257
                                  Design df        =         31

--------------------------------------------------------------
             |             Linearized
             |       Mean   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
    diabetes |   .0342853   .0018197      .0305739    .0379966
       black |   .0956367   .0127804      .0695709    .1217026
      weight |   71.91131   .1670327      71.57065    72.25198
      height |   168.4647   .1471856      168.1645    168.7649
--------------------------------------------------------------
```

Also check out the options these commands have. For example if you wanted the means by gender you could use the `over` option, e.g.

```
. svy: mean weight, over(female)
(running mean on estimation sample)

Survey: Mean estimation

Number of strata =      31      Number of obs    =       10,337
Number of PSUs   =      62      Population size  =  117,023,659
                                 Design df        =           31

            0: female = 0
            1: female = 1

--------------------------------------------------------------
             |             Linearized
        Over |       Mean   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
weight       |
           0 |   78.63267   .2091726      78.20606    79.05928
           1 |   65.71242   .2688519      65.16409    66.26075
--------------------------------------------------------------
```

Another example: `svy:` won't work in combination with the `sw`: prefix, because stepwise methods are not considered appropriate with svy data. (For a discussion of why, see https://www.stata.com/support/faqs/statistics/stepwise-regression-with-svy-commands/).

```
. sw, pe(.05): svy: logit diabetes black weight height
svy is not supported by stepwise
r(199);
```

The failure of a user-written command to support svy may just reflect the fact that the writer did not bother to add support or support was not available at the time the command was written. In other cases it may be that svy support would be inappropriate (e.g. the user-written `firthlogit` command does not currently support svy for this reason). The user-written `gologit2` program requires that you use the user-written `gsvy` prefix rather than svy if you want to do survey data analysis using `gologit2`'s autofit option.

Also, user-written post-estimation commands may or may not work after using the `svy:` prefix (and if they do work you should try to check to see if they work correctly. For example, even Stata's old `adjust` command does not work correctly with `svy` data.)

Post-estimation commands that do work with svy: Several post-estimation commands (e.g. `margins`) work with svy. For a complete list, from within Stata type `help svy postestimation`. For some commands that are more unique to svy, see `help svy_estat`. For example, the `estat sd` command can be handy:

```
. webuse nhanes2f, clear
. svy: mean female black height weight diabetes
(running mean on estimation sample)

Survey: Mean estimation

Number of strata =      31       Number of obs    =      10335
Number of PSUs   =      62       Population size  =  116997257
                                  Design df        =          31

--------------------------------------------------------------
             |             Linearized
             |       Mean   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
      female |   .5203132   .0057615      .5085627    .5320638
       black |   .0956367   .0127804      .0695709    .1217026
      height |   168.4647   .1471856      168.1645    168.7649
      weight |   71.91131   .1670327      71.57065    72.25198
    diabetes |   .0342853   .0018197      .0305739    .0379966
--------------------------------------------------------------

. estat sd

-------------------------------------
             |       Mean   Std. Dev.
-------------+-----------------------
      female |   .5203132    .4996114
       black |   .0956367    .2941067
      height |   168.4647    9.702569
      weight |   71.91131    15.43409
    diabetes |   .0342853    .1819697
-------------------------------------
```

Linear regression: Some of the statistics and tests you are used to using are inappropriate. There are numerous things you are used to doing with linear regression that will not work with svyset data. This is at least partly because, with survey data, assumptions that

cases are independent of each other are violated. In other cases, it may be because Stata hasn't figured out how to adapt the test or procedure to svyset data.

Example 1: Wald tests work but incremental F tests do not (the `ftest` command should be downloaded from SSC)

```
. svy: reg weight height age
(running regress on estimation sample)

Survey: Linear regression

Number of strata   =        31                  Number of obs      =      10337
Number of PSUs     =        62                  Population size    =  117023659
                                                Design df          =         31
                                                F(   2,     30)    =    1803.49
                                                Prob > F           =     0.0000
                                                R-squared          =     0.2785

------------------------------------------------------------------------------
             |             Linearized
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      height |   .8486177   .0148894    56.99   0.000     .8182505    .8789849
         age |   .1593711   .0118398    13.46   0.000     .1352236    .1835186
       _cons |  -77.78301   2.514129   -30.94   0.000    -82.91061   -72.65541
------------------------------------------------------------------------------

. est store m1
. svy: reg weight height age i.female i.black
(running regress on estimation sample)

Survey: Linear regression

Number of strata   =        31                  Number of obs      =      10337
Number of PSUs     =        62                  Population size    =  117023659
                                                Design df          =         31
                                                F(   4,     28)    =     880.32
                                                Prob > F           =     0.0000
                                                R-squared          =     0.2887

------------------------------------------------------------------------------
             |             Linearized
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      height |   .7409099   .0274549    26.99   0.000     .6849151    .7969046
         age |   .1520647   .0117481    12.94   0.000     .1281044    .1760251
    1.female |  -2.924562   .6054516    -4.83   0.000    -4.159389   -1.689736
     1.black |   4.033541   .7436242     5.42   0.000     2.516909    5.550172
       _cons |  -58.19333   4.909523   -11.85   0.000    -68.20637   -48.18029
------------------------------------------------------------------------------

. est store m2
. test 1.female 1.black

Adjusted Wald test

 ( 1)  1.female = 0
 ( 2)  1.black = 0

       F(  2,    30) =   20.86
            Prob > F =    0.0000
```

```
. ftest m1 m2
Linearized vce not allowed
r(198);
```

So, in general you should use Wald tests for hypothesis testing. However, you can also use the `nestreg` command (without factor variables – unless you have Stata 16 or later) since `nestreg` basically just does Wald tests on each model.

```
. nestreg, quietly: svy: reg weight (height age) (female black)

Block  1: height age
Block  2: female black
```

Block	F	Block df	Design df	Pr > F	R2	Change in R2
1	1803.49	2	31	0.0000	0.2785	
2	20.86	2	31	0.0000	0.2887	0.0102

Example 2: Numerous OLS regression post-estimation diagnostic commands do not work

```
. quietly svy: reg weight height age female black
. dfbeta
option dfbeta() not allowed after svy estimation
r(198);

. estat hetttest
invalid subcommand hetttest
r(321);

. estat imtest
invalid subcommand imtest
r(321);

. rvfplot
option resid not allowed
r(198);

. estat lvr2plot
invalid subcommand lvr2plot
r(321);
```

Given that these are diagnostic tests, you may want to do exploratory analyses that ignore the svysetting of the data.

Categorical Data Analysis: Maximum Likelihood – and Statistics based on it – are inappropriate. With survey data, the ML assumptions that cases are independent of each other are violated. As a result, you can't get several statistics you are used to, e.g. Model LR Chi^2, BIC, AIC. You get F statistics and T statistics instead of chi-squares and zs. You have to do Wald tests instead of LR Chi^2 model contrasts. The following will illustrate this.

```
. webuse nhanes2f, clear
. * Constrained model
. svy: logit diabetes female black
(running logit on estimation sample)

Survey: Logistic regression

Number of strata   =        31                  Number of obs      =      10335
Number of PSUs     =        62                  Population size    =  116997257
                                                Design df          =         31
                                                F(   2,     30)    =      15.02
                                                Prob > F           =     0.0000

------------------------------------------------------------------------------
             |             Linearized
    diabetes |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      female |   .2937317   .1198141     2.45   0.020     .0493693    .5380941
       black |    .644294   .1157506     5.57   0.000      .408219    .8803689
       _cons |  -3.582378    .100637   -35.60   0.000    -3.787628   -3.377127
------------------------------------------------------------------------------
```

Already some key differences in the output are obvious. We got an F test instead of a Model LR Chi^2 statistic. For each coefficient we got a T statistic instead of a Z statistic. No Log Likelihood for the model was reported. The Population size line told us how many people are in the population that this sample represents (about 117 million). Now let's see what happens when we try to contrast nested models.

```
. est store m1
. svy: logit diabetes female black weight height
(running logit on estimation sample)

Survey: Logistic regression

Number of strata   =        31                  Number of obs      =      10335
Number of PSUs     =        62                  Population size    =  116997257
                                                Design df          =         31
                                                F(   4,     28)    =      26.97
                                                Prob > F           =     0.0000

------------------------------------------------------------------------------
             |             Linearized
    diabetes |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      female |  -.1591092   .1481466    -1.07   0.291    -.4612563    .1430379
       black |   .5026699   .1270514     3.96   0.000     .2435468     .761793
      weight |   .0290168   .0033496     8.66   0.000     .0221852    .0358484
      height |  -.0574331   .0081596    -7.04   0.000    -.0740747   -.0407915
       _cons |   4.149243   1.295284     3.20   0.003     1.507495    6.790992
------------------------------------------------------------------------------

. est store m2
. lrtest m1 m2, all
lrtest is not appropriate with survey estimation results
r(322);
```

The `lrtest` command does not work because the assumptions behind it are violated with complicated survey designs. (It won't even work if you include the `force` option.) Hence, we don't get a likelihood ratio chi-square contrast. We also don't get BIC or AIC statistics because

the ML assumptions behind those statistics are also violated. Instead, we have to use `test` commands.

```
. * Use Wald test instead
. test weight height

Adjusted Wald test

 ( 1)  [diabetes]weight = 0
 ( 2)  [diabetes]height = 0

       F(  2,    30) =   40.50
            Prob > F =    0.0000
```

Other Comments

1. svy and multiple imputation can usually peacefully coexist. You have to get the prefixes in the right order. If you want to use mi and svy together, the MI manual says to do it this way:

```
mi estimate: svy: estimation_command ...
```

Also, if the data have not been `svyset` before imputation you should use the `mi svyset` command. See `help mi_xxxset`

2. If you don't need significance tests or confidence intervals (e.g. you just need descriptive statistics or point estimates), using other commands with weights might also do (probably pweights or aweights, but check what types of weights the command supports; some commands only support one or the other and sometimes neither):

```
webuse nhanes2f, clear
quietly svy: mean weight height age
estat sd
sum weight height age [aw = finalwgt]
```

```
. estat sd

-------------------------------------
             |       Mean   Std. Dev.
-------------+-----------------------
      weight |   71.90869    15.43333
      height |   168.4625    9.702933
         age |   42.23732    15.50095
-------------------------------------

. sum weight height age [aw = finalwgt]

    Variable |     Obs      Weight        Mean   Std. Dev.       Min        Max
-------------+-----------------------------------------------------------------
      weight |  10,337   117023659    71.90869   15.43333      30.84     175.88
      height |  10,337   117023659    168.4625   9.702933      135.5        200
         age |  10,337   117023659    42.23732   15.50095         20         74
```

If something like `sum` or `tabstat` formatted things more conveniently you might consider using them with weights. See `help weight` if you want more information on weighting. But I would suggest comparing the results with what you got with `svy:` to make sure things are the same.

3. Here is a listing of major help topics you may want to read up on if using svy data:

help svy	Overview of the svy prefix command
help svyset	Manages the survey analysis settings of a dataset. You use svyset to designate variables that contain information about the survey design, such as the sampling units and weights. You must svyset your data before using any svy command;
help svy estimation	Lists the estimation commands that work with the svy: prefix, e.g. `logit`, `regress`, `ologit`.
help svy postestimation	Lists the postestimation commands that are available after svy, e.g. `margins, predict, test`. In many cases the help is generic (i.e. is the same for both svy and non-svy commands) but in a few cases the postestimation command is specifically tailored for svy.
help svy_estat	Postestimation statistics for survey data. Includes things like `estat sd` and `estat size`
help svy: tabulate oneway	Produces one-way tabulations for complex survey data. Sort of like running frequencies (but not quite)
help svy: tabulate twoway	Produces two-way tabulations with tests of independence for complex survey data.
help mi_xxxset	If the data have not been svyset before imputation you should use the `mi svyset` command. More generally this help shows you how to declare mi data to be svy, st, ts, xt, etc. In general I would recommend doing the settings before you mi set your data, e.g. use `svyset` before you start the multiple imputation process.

Conclusion. svy data are not that hard to work with. But, you do have to understand some of the important differences that do exist. For more, see Stata's SVY Manual. Other good references (as of September 10, 2024) include

https://stats.idre.ucla.edu/stata/faq/ (see the lower part of the page for survey commands)

https://www.stata.com/support/faqs/statistics/#survey

Appendix: But, Should You Weight?

Having said all that, there are controversies over whether you should always weight and how you should do it. The following is paraphrased from a post I made at https://www.statalist.org/forums/forum/general-stata-discussion/general/1420211-is-the-use-of-sampling-weights-in-regression-always-best-or-are-there-tradeoffs-that-need-to-be-considered

I suspect most people use weighting, especially for getting estimates of descriptive statistics like means. But for things like logit and OLS, not everyone thinks it should be universally done. See

http://www.annualreviews.org/doi/abs/10.1146/annurev-statistics-011516-012958 [Are Survey Weights Needed? by Bollen et al, 2016]

"At a time when most surveys have unequal probabilities of selection either by design or by other practical constraints, the question of whether to weight variables during the analysis takes on added importance. If weighting data were a cost-free option, then always weighting would be a reasonable strategy. But unnecessarily weighting means lower efficiency and lower statistical power. Tests that determine whether weights are required do exist, but they are rarely applied for several reasons. One is the lack of awareness among researchers. Another is the influence of tradition in different fields—some always weight and others never do. An additional reason is that some of these tests are not readily available in software packages. Furthermore, even when these tests are easy to implement, there is little guidance on which of the many tests to choose."

https://projecteuclid.org/euclid.ss/1190905511 [Struggles with Survey Weighting and Regression Modeling. By Gelman, 2007]

"Survey weighting is a mess. It is not always clear how to use weights in estimating anything more complicated than a simple mean or ratios, and standard errors are tricky even with simple weighted means. (Software packages such as Stata and SUDAAN perform analysis of weighted survey data, but it is not always clear which, if any, of the available procedures are appropriate for complex adjustment schemes. In addition, the construction of weights is itself an uncodified process.)"

https://www.nber.org/papers/w18859
[What Are We Weighting For? Gary Solon, Steven J. Haider, Jeffrey M. Wooldridge, 2015]

"When estimating population descriptive statistics, weighting is called for if needed to make the analysis sample representative of the target population. With regard to research directed instead at estimating causal effects, we discuss three distinct weighting motives: (1) to achieve precise estimates by correcting for heteroskedasticity; (2) to achieve consistent estimates by correcting for endogenous sampling; and (3) to identify average partial effects in the presence of unmodeled heterogeneity of effects. In each case, we find that the motive sometimes does not apply in situations where practitioners often assume it does."

https://journals.sagepub.com/doi/pdf/10.1177/0049124194023002004 or else https://journals.sagepub.com/doi/10.1177/0049124194023002004]
[Sampling Weights and Regression Analysis. By Winship and Mare, 1994.]

"When a researcher is going to perform a regression analysis with data that have sampling weights, what should be done? First, the analyst should estimate two models: one with unweighted data (OLS) and one using the sampling weights (WOLS). If the parameter estimates are substantively similar, then the OLS estimates are preferable because they are more efficient and the estimated standard errors will be correct... When OLS and WOLS produce different parameter estimates, the researcher needs to carefully consider the possible reasons. One possibility is that the model may be missing linear, nonlinear, or interaction terms."

In that 1994 article, several statistical packages, including Stata, were criticized for the way they handle weighting. In 2005, I asked Winship whether he was still critical of Stata. He replied "I am very happy with how Stata handles weighting. It does provide a lot of different options so one can do it wrong in any particular case. However, it does calculate the standard errors correctly. It is still the case that if weighted and unweighted differ, this is evidence that you have a mis-specified model. If they are the same, one should use the unweighted because they have smaller standard errors."

Conclusion. Having said all that, I tend to just go ahead and weight, partly because I do not think I am smart enough to figure out how and when to not weight. But, I do think it may be a good idea to check whether weighted and unweighted OLS coefficient estimates differ, since such differences may indicate problems with model specification.