
Subject: Re: De-normalizing weights and svyset command in Stata

Posted by [Reduced-For\(u\)m](#) on Wed, 14 Jan 2015 20:53:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

Not sure I can help on all of these, but I have a few suggestions:

1/2 - I don't know much about the HIV weights or whether they represent individuals or households, but in general I use the following type of procedure. For each survey (country-by-survey-round), I calculate the total sum of weights (W_c) and then take the individual weight (W_i) and divide it by the total weight (W_i/W_c). Now the sum of weights for that country/round equals 1, while preserving the relative probability across people. Then, if I want my weights to be population-level representative, I multiply those weights by the country's population (or, in the case of a sub-population (say, Men age 25-49 or something) by that sub-population). That information comes from somewhere else (UN, World Bank, etc). Now you have weights that, within-country, distribute the population weight by probability of sampling, and across country factor in population differences. If the weights are representative of households and not individuals, the relevant population would be households in the country.

Note though: if you are using, say, West Africa as a region, Nigeria will basically swamp everything else. This may or may not be desirable, but you should look at your relative populations and decide whether or not you want one or two countries to dominate your estimates (you may want that, you may not).

3 - I know very little about the HIV testing, but if you are pooling people who were and were not tested, you'd get the wrong prevalence (because you don't know if the people in the other recodes are positive or not). Maybe I'm just worrying about nothing, but didn't immediately understand why you were pooling those to calculate HIV prevalence.

4/5 -you can do the svyset stuff mostly mechanically too. Binreg supports weights, so you can directly use those (`pw=weight`) and then it also supports clustering. You'd want something like:

```
binreg Y X [pw=weight], vce(cluster clustervar)
```

This won't get you the efficiency gains (read smaller standard errors) that accounting for stratification might get you, but that should be a small effect*. Note, when generating "clustervar" you want to do it so that each country has its own clusters (so cluster 14 in one country is different than cluster 14 from another, for instance). You might just generate a country/survey two-digit huber, and then `gen clustervar = countrynum*1000 + clusternum` - something like that.

Hope some of that helps.

*If you want to get fancy, you could probably bootstrap that using the stratification and drawing clusters with replacement, but my guess is the difference is going to be very small, and if you already have enough precision (are already statistically significant at acceptable levels) it might not be worth it, you can just say your SEs are "conservative".
